Cite this research: *Telo, J. (2022). Supervised Machine Learning for Detecting Malicious URLs: An Evaluation of Different Models*

SSRAML SageScience, 5(2), 30–46.

Article history:
Received:
April/13/2022
Accepted:
Nov/10/2022
Published:
Nov/15/2022

.

Supervised Machine Learning for Detecting Malicious URLs: An Evaluation of Different Models

Joan Telo
https://orcid.org/0009-0004-5101-8064

Abstract

Malicious URLs are often used to distribute malware, steal personal information, or engage in phishing attacks. Traditional approaches for identifying these URLs are often ineffective, and as such, researchers are exploring new methods to address this problem. In this study, we investigate the use of supervised machine learning models to detect malicious URLs. Our dataset consisted of 651191 URLs, which were classified into four different categories: Benign, defacement, phishing, and malware. We employed several machine-learning algorithms, including Decision Tree, Random Forest, Ada Boost, K Neighbors, SGD, Extra Trees, and Gaussian NB, to evaluate their ability to classify URLs into these categories accurately. Our results show that the accuracy scores range from 0.789548 to 0.914718, indicating that the models perform reasonably well in detecting malicious URLs. The Random Forest Classifier and Extra Trees Classifier achieved the highest accuracy scores of 0.914718 and 0.914711, respectively, indicating that they performed the best on the dataset. In contrast, the Gaussian NB model had the lowest accuracy score of 0.789548, suggesting that it performed the worst on the dataset. This research demonstrates that supervised machine learning models can effectively detect malicious URLs. The results indicate that Random Forest and Extra Trees classifiers may be particularly useful for this task. This research may provide a foundation for further development and improvement of machine learning-based systems for detecting malicious URLs, enhancing online security for individuals and organizations.

Keywords: *Accuracy, Classification, Extra Trees Classifier, Malicious URLs, Random Forest Classifier, Supervised machine learning*

I. Introduction

Malicious URLs are one of the most common and dangerous forms of cyber attacks today. They are a type of online threat that is designed to trick users into clicking on a link that leads to a fake website or a site that contains malware, phishing scams, or other harmful content. Malicious URLs can be found in a variety of places online, including email messages, social media posts, online ads, and even search engine results. Malicious URLs are links that are designed to look like legitimate links, but they actually lead to websites that are designed to steal information from users or infect their devices with malware. These links can be sent via email, social media, or other online channels, and they can be disguised as messages from reputable companies or organizations, like banks or government agencies.

Malicious URLs work by tricking users into clicking on them, which then leads them to a fake website or a site that contains malware. These links are often disguised as legitimate links, and they can be very convincing. When a user clicks on a malicious URL, they are redirected to a fake website that looks just like the real one. This fake website may ask the user to enter their login credentials, credit card information, or other sensitive information, which is then stolen by the attacker. Malicious URLs can also lead users to websites that contain malware, which can infect their devices with viruses or other harmful software. Malware can be used to steal sensitive information, like passwords and banking information, or it can be used to control users' devices or steal data from them.

In some cases, malicious URLs may be used to launch a phishing attack. Phishing attacks are designed to trick users into entering their login credentials or other sensitive information into a fake website. Once the attacker has this information, they can use it to access the user's accounts or steal their identity. Phishing URLs are a common tactic used by cybercriminals to steal sensitive information, such as login credentials or financial data. Phishing URLs typically mimic legitimate websites in order to trick users into providing their information. These URLs are often sent via email, text message, or social media, and can also be found on fake websites or advertisements.

One of the most common ways that phishing URLs are used is through email scams. Cybercriminals will create emails that appear to be from a legitimate source, such as a bank or social media platform, and will include a link to a fake website. Once the user enters their information on the fake website, the cybercriminals can use it to access the user’s accounts or steal their identity. Another common use of phishing URLs is through fake websites. Cybercriminals will create websites that look similar to legitimate websites, such as a bank or online retailer, and will use these sites to steal sensitive information. They may also use advertisements to direct users to these fake websites, making it even more difficult for users to identify the scam. Phishing URLs can also be found on social media platforms. Cybercriminals will create fake profiles or pages and will use them to send messages to users that contain phishing URLs. These messages may be disguised as legitimate offers or promotions, but in reality, they are simply attempts to steal the user’s information.

Malware URLs are links that lead to websites that contain malicious software, also known as malware. These URLs can be used by cybercriminals to infect a user's computer with malware, such as viruses, spyware, or ransomware [1]. Malware URLs can be found in various forms, including email attachments, malicious pop-up ads, or links shared on social media or instant messaging platforms.

One of the most common ways that malware URLs are used is through email phishing scams. Cybercriminals will send an email with a link to a malicious website disguised as a legitimate one, such as a bank or online retailer. Once the user clicks on the link, malware can be downloaded onto their computer, giving the cybercriminal access to their personal information. Malware URLs can also be found on fake websites or advertisements. Cybercriminals may create a fake website that looks similar to a legitimate one and entice users to click on links or download software. They may also use pop-up ads that direct users to websites containing malware.

Spam URLs are links that lead to websites that contain unwanted or unsolicited content, typically in the form of advertisements, scams, or other types of unwanted content. These

URLs can be sent via email, text message, or social media, and can also be found on fake websites or advertisements. Spam URLs are often used to promote fake products, services, or scams, and can be used to trick users into giving away personal information or downloading malware.

Cybercriminals will send emails to large numbers of users, often using deceptive subject lines or other tactics to entice users to click on links. Once the user clicks on the link, they may be directed to a fake website, or a website containing unwanted content or malware. Spam URLs can also be found on social media platforms, where they may be shared by bots or fake accounts [2]. These URLs may be disguised as legitimate content, such as news articles or videos, but in reality, they are simply attempts to trick users into clicking on links.

Adware URLs are links that lead to websites that contain unwanted or malicious advertisements. Adware is software that displays advertisements on a user's computer, often without their consent or knowledge. Adware URLs can be found in various forms, including email attachments, pop-up ads, or links shared on social media or instant messaging platforms. One of the most common ways that adware URLs are used is through pop-up ads [3-5]. Cybercriminals will create ads that look like legitimate ones, such as antivirus software or system updates, and entice users to click on them. Once the user clicks on the ad, adware can be downloaded onto their computer, displaying unwanted ads and potentially slowing down their system.

Adware URLs can also be found on fake websites or advertisements. Cybercriminals may create a fake website that looks similar to a legitimate one and entice users to click on links or download software. They may also use pop-up ads that direct users to websites containing adware.

Clicking on a malicious URL can have serious consequences for users. Depending on the type of malicious URL, users may be at risk of having their personal information stolen, their devices infected with malware, or their accounts hacked. Phishing URLs can be particularly dangerous, as they can lead to the theft of login credentials and other sensitive information [6]. If a user falls for a phishing scam, their accounts may be compromised, and they may be at risk of identity theft or other forms of financial fraud. Malware URLs can also be very harmful. Malware can infect users' devices and steal sensitive information, like passwords and banking information [7]. It can also be used to control users' devices or steal data from them. In addition to the immediate consequences of clicking on a malicious URL, there can also be long-term effects. If a user's personal information is stolen, they may be at risk of identity theft or other forms of financial fraud. This can have a lasting impact on their credit score and financial security. If a user's device is infected with malware, it may be difficult or even impossible to remove it completely. This can lead to ongoing security vulnerabilities and a decreased level of trust in online security [8].

Another negative consequence of malicious URLs is the risk of falling victim to a scam. Scammers can use malicious URLs to trick individuals into believing they are purchasing a legitimate product or service, only to end up losing their money without receiving anything in return. In recent years, there has been a significant shift towards buying and selling online, as more people are choosing to purchase goods and services through e-commerce platforms and online marketplaces [9]. This trend is driven by a combination of

factors, including convenience, competitive pricing, and the availability of a vast array of products and services from different sellers, all accessible from the comfort of one's own home. As a result, online marketplaces have become an essential part of the modern retail landscape, with many traditional brick-and-mortar retailers expanding their online offerings to remain competitive in the ever-evolving digital marketplace [10]. Cybercriminals may create fake online marketplaces or product listings that look identical to legitimate ones but are designed to steal money from unsuspecting buyers. Scammers can also use malicious URLs to deliver ransomware, which can encrypt an individual's files and demand payment in exchange for the decryption key. In addition to financial losses, falling victim to a scam can lead to feelings of embarrassment, frustration, and loss of trust in online marketplaces and e-commerce platforms.

Machine learning (ML) is a subset of artificial intelligence (AI) that enables computers to learn and adapt from experience without being explicitly programmed. ML algorithms are designed to learn from large volumes of data and identify patterns or insights that can be used to make predictions or decisions. Machine learning is rapidly evolving and has become a critical component of many modern technologies, including self-driving cars, speech recognition systems, and image recognition applications.

One of the key benefits of machine learning is that it can be used to automate decision-making processes. For example, in the financial sector, machine learning algorithms are used to analyze large volumes of financial data and make recommendations for trading decisions. In healthcare, machine learning is used to analyze patient data to identify potential health risks and develop personalized treatment plans [10]. By automating decision-making processes, machine learning can help to increase efficiency, reduce costs, and improve accuracy. Moreover, machine learning has the ability to detect patterns and anomalies in data that might not be apparent to human analysts. This can be especially useful in areas such as fraud detection, where machine learning algorithms can analyze large volumes of financial data and identify patterns that suggest fraudulent activity. Similarly, in cybersecurity, machine learning can be used to analyze network traffic and detect patterns of suspicious behavior that might indicate a cyber attack [11].

II. Feature selection: Indicators to detect potentially malicious URLs

One effective way to detect potentially malicious URLs is to look for indicators that suggest the URL may be harmful. These indicators can include the use of non-standard characters or symbols, misspelled words, or suspicious domains that have recently been registered. Additionally, URLs that contain multiple redirects, have long strings of numbers and letters, or appear to mimic legitimate websites may also be indicators of malicious intent [12-15].

HTTP:

The Hypertext Transfer Protocol (HTTP) is the underlying protocol used by the World Wide Web to transfer data between web servers and clients. URLs that use HTTP are often considered to be less secure than those that use the more secure HTTPS protocol. The main reason for this is that HTTP does not provide any encryption or authentication mechanisms, which means that data transmitted over HTTP can be intercepted and manipulated by third parties.

In contrast, URLs that use HTTPS provide end-to-end encryption, which ensures that data transmitted between the web server and client is secure and cannot be accessed by unauthorized parties. Additionally, HTTPS also provides authentication mechanisms, which verify that the website being accessed is legitimate and has not been compromised by attackers. Despite the clear benefits of HTTPS, many websites still use HTTP, either due to technical limitations or a lack of awareness of the risks. This can make it difficult for users to identify whether a URL is malicious or not [16-18]. However, there are some indicators that can help users determine whether a URL is likely to be safe or not.

One such indicator is the presence of a padlock icon in the address bar, which indicates that the website is using HTTPS and that the connection is secure. Additionally, some web browsers may display a warning message if a user tries to access a website over HTTP, informing them that the connection is not secure and that data may be intercepted by attackers.

Number of digits:

URLs that contain a large number of digits may be an indicator of potentially malicious activity, as attackers often use numbers to obfuscate the true nature of the URL. In some cases, attackers may use long strings of random digits to create URLs that look legitimate but actually lead to malicious content or phishing sites. Attackers may also use numbers to try to bypass web filters or other security mechanisms that are designed to block known malicious URLs. By using a large number of digits, attackers can create URLs that are unique and not yet blacklisted by security systems, making it more difficult to detect and block malicious content.

However, not all URLs that contain a large number of digits are malicious. Some legitimate websites may use numbers in their URLs for various reasons, such as tracking user behavior or organizing content. As such, users should not automatically assume that a URL is malicious simply because it contains a large number of digits.

Number of letters:

URLs that contain a large number of letters may be less suspicious than those that contain a large number of digits or special characters. This is because attackers often use digits and special characters to create URLs that look legitimate but actually lead to malicious content or phishing sites. By contrast, URLs that contain only letters may be more indicative of legitimate websites, as they are more likely to use human-readable words and phrases.

However, it is important to note that not all URLs that contain a large number of letters are legitimate. Attackers may still use letters to try to trick users into clicking on malicious links or entering personal information on phishing sites. For example, attackers may create URLs that mimic legitimate websites but contain subtle misspellings or variations in the spelling of the domain name.

Shortening service:

URLs that have been shortened by a service such as bit.ly or goo.gl may be used to hide the true destination of the URL and may be used in phishing attacks. Shortening services take a long URL and create a shorter, condensed version that can be more easily shared on social media platforms or in email messages. However, the shortened URL may not reveal

the true destination of the link, making it difficult for users to determine whether the link is safe to click. Attackers may use shortened URLs to redirect users to phishing sites or to distribute malware through spam emails or other means.

Shortening services may also be used to create URLs that mimic legitimate websites, making it difficult for users to detect that they are being directed to a phishing site. For example, an attacker may use a shortened URL that contains a misspelled version of a popular website name or domain.

IP address:

URLs that have an IP address instead of a domain name may be more suspicious, as legitimate websites typically use domain names. An IP address is a unique numerical identifier assigned to each device on a network, such as a computer or server. While some websites may use IP addresses for testing or internal purposes, it is unusual for a public-facing website to use an IP address as its primary identifier.

Attackers may use IP addresses to host malicious content or to create phishing sites that mimic legitimate websites. By using an IP address instead of a domain name, attackers can create URLs that are difficult to detect and block using traditional security measures. This can make it easier for attackers to carry out phishing attacks, distribute malware, or steal sensitive information from unsuspecting users.

URL length:

Sure, URLs that are longer in length may be indicative of suspicious activity. In general, shorter URLs are easier to read, remember, and share, which is why many legitimate websites use them. Longer URLs may be used to obfuscate the true destination of the URL or to include additional parameters that can be used to carry out attacks.

Attackers may use longer URLs to create phishing sites that mimic legitimate websites or to distribute malware through spam emails or other means. By using longer URLs, attackers can make it more difficult for users to determine whether the link is safe to click or not. Additionally, longer URLs may contain misspellings or other typos that can make them appear more convincing to unsuspecting users. However, not all longer URLs are malicious in nature. Some legitimate websites may use longer URLs for various reasons, such as including tracking information or session IDs.

Symbols

@: URLs that contain an '@' symbol may be more suspicious as it can be used in phishing attacks, where attackers try to trick users into entering their login credentials on a fake website.

?: URLs that contain a question mark '?' may be more suspicious as it can be used to add parameters to a URL and may be used to carry out malicious activities such as injecting malware or stealing sensitive information.

-: URLs that contain a hyphen '-' may be more suspicious as it can be used to create homoglyphs or typosquatting domains that can be used to impersonate legitimate domains.

=: URLs that contain an equal sign '=' may be more suspicious as it can be used to pass parameters or execute malicious code.

.: URLs that contain multiple dots '.' or have a very long domain extension (e.g. .xyz or .top) may be more suspicious as they can be used to impersonate legitimate domains or hide the true destination of the URL.

#: URLs that contain a hash symbol '#' may be more suspicious as it can be used to add a fragment identifier to a URL and may be used to carry out malicious activities such as injecting scripts or executing code.

%: URLs that contain a percent symbol '%' may be more suspicious as it can be used to encode characters or execute malicious code.

+: URLs that contain a plus sign '+' may be more suspicious as it can be used to concatenate multiple parameters or execute malicious code.

$: URLs that contain a dollar sign '$' may be more suspicious as it can be used to encode characters or execute malicious code.

!: URLs that contain an exclamation mark '!' may be more suspicious as it can be used to execute malicious code or carry out phishing attacks.

: URLs that contain an asterisk " may be more suspicious as it can be used to carry out wildcard attacks or execute malicious code.

,: URLs that contain a comma ',' may be more suspicious as it can be used to separate parameters or carry out malicious activities such as code injection.

//: URLs that contain a double slash '//' may be more suspicious as it can be used to carry out path traversal attacks or execute malicious code.

III. Results

Figure 1 represents the distribution of different types of website types based on their count. The figure has four types of activities namely "benign", "defacement", "phishing", and "malware". The number of websites detected with "benign" activity is the highest among all the types, with a count of 428,103. "Defacement" has the second-highest count, with 96,457 websites detected with this activity. "Phishing" is also prevalent among websites, with 94,111 websites detected to have this activity. The count of websites with "malware" activity is the least among all the types, with only 32,520 websites detected with this activity. able 2 shows the results of feature extraction for several URLs. The table contains information about the URL length, domain, and the presence of several special characters in the URLs. The first column of the table indicates whether the URL is classified as belonging to a certain category (1) or not (0).

The second column shows the length of each URL in characters. The third column provides the domain of the URL. The following columns indicate the presence (1) or absence (0) of various special characters in the URL, such as question marks, hyphens, equal signs, periods, hashtags, percentage signs, plus signs, dollar signs, exclamation marks, asterisks, commas, double slashes, and at signs.

Figure 1. Distribution of different types of website types

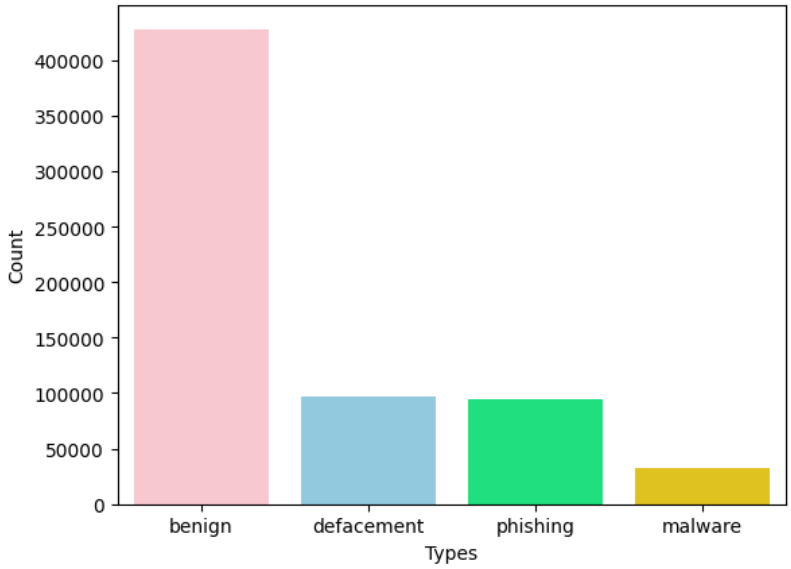
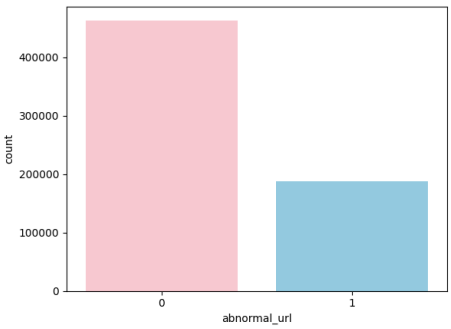
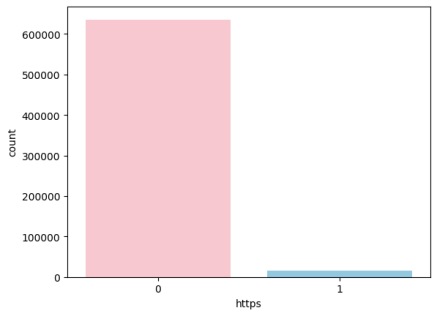


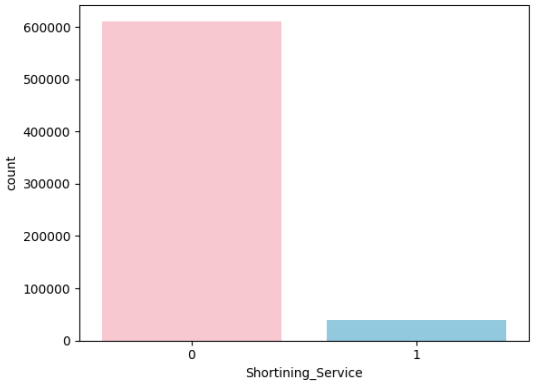
Table 1. sample of dataset after deleting www. subdomains	
url	type
collaboration.cadbury.com/allaboutus/ourbrands...	benign
http://9779.info/%E5%88%9D%E4%B8%AD%E7%A7%91%E...	malware
lcp0rkyg-site.1tempurl.com	phishing
optivasecurity.000webhostapp.com	phishing
moviefilmcenter.com/raw-justice	benign
http://secure.runescape.com.d.weblagon.loginfo...	phishing
citiesarchitecture.com/Architecture/6/1942/Ell...	benign
http://pomorskie-lzs.pl/index.php?option=com_f...	defacement
http://61.52.144.240:48330/Mozi.m	malware
http://appleid.apple.co.uk.cgi-bin.webobjects....	phishing

Table 2. Feature extraction															
url category	URL length	domain	?	-	=	.	#	%	+	$	!	*	,	//	@
1	128	howdypartnersmedia.com.au	0	1	4	5	3	0	0	0	0	0	0	0	1
0	61	findticketsfast.com	0	0	0	0	2	0	0	0	0	0	0	0	0
1	38	javadoplant.nl	0	1	0	1	2	0	0	0	0	0	0	0	1
0	13	hutchgov.com	0	0	0	0	1	0	0	0	0	0	0	0	0
1	80	destro.nl	0	0	0	0	3	0	0	0	0	0	0	0	1
0	60	animea.net	0	0	4	0	2	0	0	0	0	0	0	0	0
3	65	huangjintawujin.cn	0	0	1	0	2	0	0	0	0	0	0	0	1
0	45	questia.com	0	0	0	0	2	0	0	0	0	0	0	0	0
0	25	kthorjensen.blogspot.com	0	0	0	0	2	0	0	0	0	0	0	0	0
0	27	coloradorapids.com	0	0	0	0	1	0	0	0	0	0	0	0	0

Figure 2. Feature distribution







The correlations between various variables in the dataset are shown in figure 3. In this case, several noteworthy correlations have been identified. For instance, the variable url_len has a moderately strong positive correlation with symbols like (dot) and = (equal sign), as well as with the Shortening_Service variable. Similarly, having_ip_address has a moderate positive correlation with the abnormal_url, https, and digits variables. The (dot) symbol also has a moderate positive correlation with = (equal sign) and Shortening_Service, and a weak positive correlation with letters. The equal sign (=) has a moderate positive correlation with the Shortening_Service variable. Additionally, abnormal_url has a moderate positive correlation with https and digits. Finally, https has a moderately strong positive correlation with digits and a weak positive correlation with letters. The digits variable, in turn, has a weak positive correlation with letters.

Figure 3. Correlation heatmap

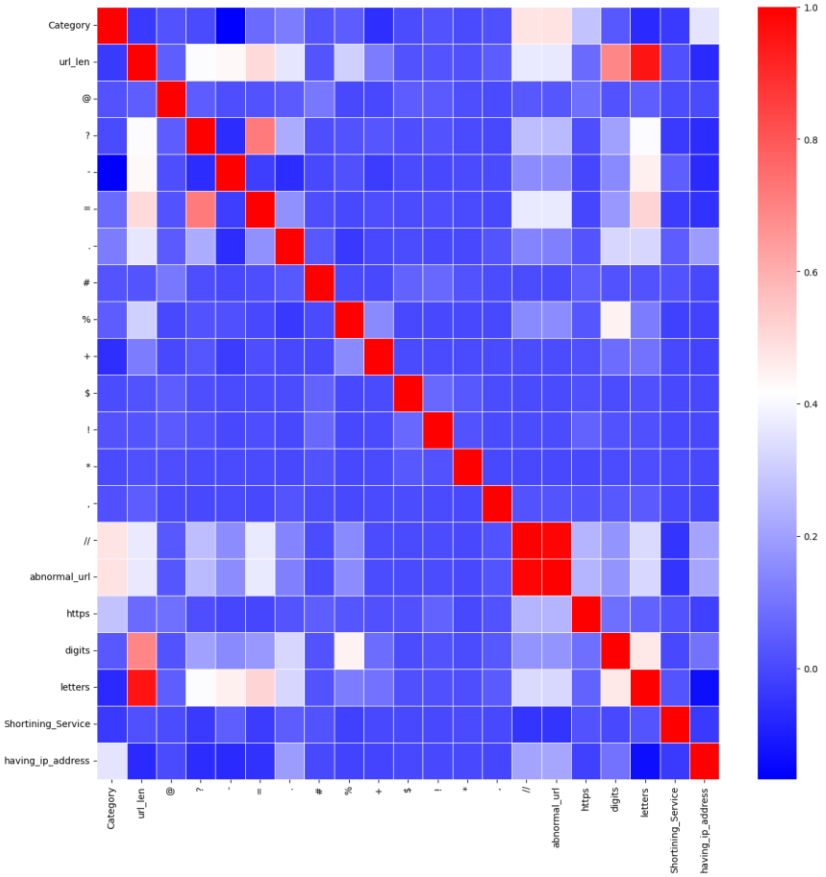


Table 3 (a). comparison among classifiers					
Classifier	Test Accuracy	Precision (0)	Precision (1)	Precision (2)	Precision (3)
DecisionTreeClassifier	90.95%	0.92	0.93	0.81	0.95
RandomForestClassifier	91.47%	0.92	0.94	0.83	0.96
AdaBoostClassifier	82.01%	0.84	0.82	0.45	0.91
KNeighborsClassifier	89.04%	0.91	0.89	0.74	0.94
SGDClassifier	81.85%	0.83	0.83	0.43	0.88
ExtraTreesClassifier	91.47%	0.92	0.93	0.83	0.97
GaussianNB	78.95%	0.85	0.66	0.6	0.61

Table 3 (b). comparison among classifiers					
Classifier	Test Accuracy	Recall (0)	Recall (1)	Recall (2)	Recall (3)
DecisionTreeClassifier	90.95%	0.97	0.96	0.57	0.91
RandomForestClassifier	91.47%	0.98	0.96	0.58	0.91
AdaBoostClassifier	82.01%	0.98	0.89	0.15	0.46
KNeighborsClassifier	89.04%	0.96	0.95	0.52	0.87
SGDClassifier	81.85%	0.99	0.86	0.1	0.56
ExtraTreesClassifier	91.47%	0.98	0.97	0.57	0.91
GaussianNB	78.95%	0.92	1	0.02	0.7

The tables 3, (table 3 (a), and table 3, (b)) provide performance metrics for various machine learning algorithms, including DecisionTreeClassifier, RandomForestClassifier, AdaBoostClassifier, KNeighborsClassifier, SGDClassifier, ExtraTreesClassifier, and GaussianNB, evaluated on a test dataset. These algorithms have been trained to classify samples into four different classes represented by 0, 1, 2, and 3, and the performance metrics include accuracy, precision, recall, f1-score, and support.

Accuracy is a measure of how well the algorithms perform in predicting the correct class for the samples. In this case, all the algorithms have achieved an accuracy of over 78%, with the highest accuracy achieved by RandomForestClassifier and ExtraTreesClassifier at 91.47%. The DecisionTreeClassifier and KNeighborsClassifier have also performed well with accuracy scores of 90.95% and 89.04%, respectively. On the other hand, AdaBoostClassifier and GaussianNB have achieved lower accuracy scores of 82.01% and 78.95%, respectively, indicating that they may not be the best choice for this classification problem.

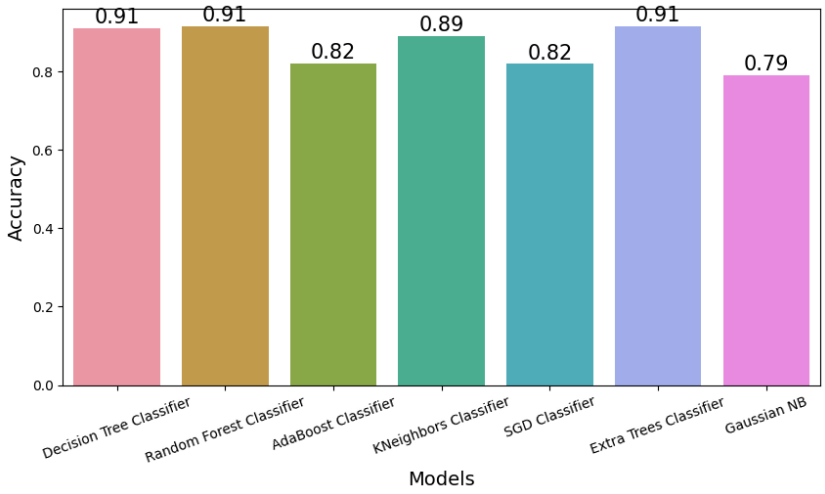
Precision measures how well the algorithms classify samples belonging to a particular class correctly. The precision scores for each class range from 0.45 to 0.97, with RandomForestClassifier and ExtraTreesClassifier having the highest precision scores across all classes. DecisionTreeClassifier, KNeighborsClassifier, and SGDClassifier also have relatively high precision scores. However, AdaBoostClassifier and GaussianNB have relatively lower precision scores, indicating that they may not be the best algorithms for predicting samples belonging to certain classes.

Recall measures how well the algorithms can identify samples belonging to a particular class. The recall scores for each class range from 0.10 to 1.00. RandomForestClassifier, ExtraTreesClassifier, and KNeighborsClassifier have the highest recall scores across all classes, indicating that they can effectively identify samples belonging to any class. AdaBoostClassifier and GaussianNB have lower recall scores, indicating that they may not be effective in identifying samples belonging to certain classes.

The f1-score is a weighted average of precision and recall and provides a measure of the overall performance of the algorithms. The f1-scores for each class range from 0.04 to 0.95,

with RandomForestClassifier and ExtraTreesClassifier having the highest f1-scores across all classes. DecisionTreeClassifier, KNeighborsClassifier, and SGDClassifier also have relatively high f1-scores. However, AdaBoostClassifier and GaussianNB have relatively lower f1-scores, indicating that they may not be the best algorithms for predicting samples belonging to certain classes. Support indicates the number of samples belonging to each class. The support values range from 6,550 to 85,565, with the highest number of samples belonging to class 0 and the lowest number of samples belonging to class 3. The performance metrics suggest that RandomForestClassifier and ExtraTreesClassifier are the best algorithms for this classification problem, achieving the highest accuracy, precision, recall, and f1-score across all classes. DecisionTreeClassifier and KNeighborsClassifier also perform well, achieving high accuracy, precision, recall, and f1-score scores, while AdaBoostClassifier and GaussianNB have relatively lower scores, indicating that they may not be the best choice for this classification problem.

Figure 4. Classifier comparison



The figure shows the accuracy scores of seven different classification models, namely Decision Tree Classifier, Random Forest Classifier, AdaBoost Classifier, KNeighbors Classifier, SGD Classifier, Extra Trees Classifier, and Gaussian NB. The accuracy score is a metric that indicates how well the model predicts the correct output class for the given input data. According to the figure, the Random Forest Classifier and Extra Trees Classifier have the highest accuracy scores of 0.914718 and 0.914711, respectively. These two models seem to perform better than the other models in the list. The Decision Tree Classifier also has a high accuracy score of 0.909489. On the other hand, the AdaBoost Classifier has the lowest accuracy score of 0.820077. This model appears to be the least

accurate among the seven models. The Gaussian NB has the second-lowest accuracy score of 0.789548.

IV. Conclusion

The rapid growth of the internet has led to an increase in cybercrime. Cybercriminals use various methods to exploit vulnerabilities and steal sensitive information. One of the most common methods used by cybercriminals is the use of malicious URLs. Malicious URLs are URLs that are designed to look legitimate, but when clicked on, they can infect a computer with malware, steal sensitive information, or redirect the user to a fake website. To protect users from these threats, machine learning algorithms can be used to detect and block malicious URLs [19-23].

Malicious URLs are URLs that are designed to look legitimate but are actually designed to harm the user. Malicious URLs can be used to spread malware, steal sensitive information, or redirect the user to a fake website [24]. Malicious URLs can be found in various places such as email, social media, and search engine results. Cybercriminals use various methods to make malicious URLs look legitimate, such as using a legitimate-looking domain name or using a URL that appears to be from a trusted source.

Machine learning has emerged as a powerful tool for detecting malicious URLs, which are one of the most common ways that attackers use to spread malware or phishing attacks. Malicious URLs often hide behind seemingly harmless links and can easily trick unsuspecting users into clicking on them [25-27]. By analyzing various features of a URL, machine learning algorithms can identify patterns that are associated with malicious content and use these patterns to detect such URLs [28-31].

The domain name is the part of the URL that identifies the website or server that hosts the content. Malicious URLs often use domain names that are misspelled or have slight variations from legitimate domain names, such as using the number "1" instead of the letter "l". Machine learning algorithms can detect such patterns and flag them as potentially malicious. Another feature that machine learning algorithms analyze when detecting malicious URLs is the length of the URL. Malicious URLs often have longer URLs that contain many subdomains, making them difficult to read and understand. By analyzing the length of the URL, machine learning algorithms can identify patterns that are associated with malicious content and use these patterns to detect such URLs.

The presence of certain keywords in a URL is also an important feature that machine learning algorithms consider when detecting malicious URLs. Malicious URLs often contain keywords that are related to popular topics such as celebrities, news events, or political issues, as these topics are more likely to attract users' attention. Machine learning algorithms can analyze the presence of such keywords and use them as an indicator of potentially malicious content. The use of URL encoding is another feature that machine learning algorithms analyze when detecting malicious URLs. URL encoding is a technique that is used to convert certain characters in a URL into a special format that can be safely transmitted over the internet [32-34]. Malicious URLs often use URL encoding to hide their true content or to evade detection by security tools. Machine learning algorithms can detect the use of URL encoding and use it as an indicator of potentially malicious content.

The detection of malicious URLs is a significant challenge in today's digital landscape. One of the main challenges is the sheer volume of URLs present on the internet, which makes it virtually impossible to manually check each URL for malicious content. Additionally, URLs can be complex, with cybercriminals using techniques like URL encoding to hide their malicious intent. This complexity makes it harder to detect malicious content, even with the use of automated tools. Another challenge is the emergence of zero-day attacks, which are designed to exploit vulnerabilities that have not yet been discovered or patched. Traditional detection methods may not be effective in detecting these attacks, making it necessary to develop new detection techniques that can keep up with the evolving threat landscape. Furthermore, machine learning algorithms can sometimes produce false positives, incorrectly identifying legitimate URLs as malicious. This can result in significant disruptions and costs, as legitimate traffic is blocked or flagged as suspicious.

Reference

[1] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Identifying suspicious URLs: an application of large-scale online learning," in *Proceedings of the 26th Annual International Conference on Machine Learning*, Montreal, Quebec, Canada, 2009, pp. 681–688.

[2] J. Jiang *et al.*, "A Deep Learning Based Online Malicious URL and DNS Detection Scheme," in *Security and Privacy in Communication Networks*, 2018, pp. 438–448.

[3] Y.-L. Zhang *et al.*, "POSTER: A PU Learning based System for Potential Malicious URL Detection," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, Dallas, Texas, USA, 2017, pp. 2599–2601.

[4] H. Choi, B. B. Zhu, and H. Lee, "Detecting malicious web links and identifying their attack types," 2011. .

[5] D. Huang, K. Xu, and J. Pei, "Malicious URL detection by dynamically mining patterns without pre-defined elements," *World Wide Web J. Biol.*, vol. 17, no. 6, pp. 1375–1394, Nov. 2014.

[6] A. C. Bahnsen, I. Torroledo, L. D. Camacho, and S. Villegas, "Deepphish: simulating malicious ai," in *2018 APWG symposium on electronic crime research (eCrime)*, 2018, pp. 1–8.

[7] R. K. Nepali and Y. Wang, "You Look Suspicious!!: Leveraging Visible Attributes to Classify Malicious Short URLs on Twitter," in *2016 49th Hawaii International Conference on System Sciences (HICSS)*, 2016, pp. 2648–2655.

[8] A. A. Mughal, "Cybersecurity Hygiene in the Era of Internet of Things (IoT): Best Practices and Challenges," *Applied Research in Artificial Intelligence and Cloud Computing*, vol. 2, no. 1, pp. 1–31, 2019.

[9] V. Garcia, "The Impact of Presence in Global Online Marketplace on Global Brand Awareness," *Journal of Modern Issues in Business Research*, vol. 1, no. 1, pp. 1–12, 2021.

[10] V. Garcia, "Do Online Marketplaces Play a Significant Role in Shaping Entrepreneurial Intention? An Empirical Investigation," *Empirical Quests for Management Essences*, 2021.

[11] B. Cui, S. He, X. Yao, and P. Shi, "Malicious URL detection with feature extraction based on machine learning," *Int. J. High Perform. Comput. Networking*, vol. 12, no. 2, pp. 166–178, Jan. 2018.

[12] P. Burnap, A. Javed, O. F. Rana, and M. S. Awan, "Real-time Classification of Malicious URLs on Twitter using Machine Activity Data," in *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, Paris, France, 2015, pp. 970–977.
[13] A. D. Gabriel, D. T. Gavrilut, B. I. Alexandru, and P. A. Stefan, "Detecting Malicious URLs: A Semi-Supervised Machine Learning System Approach," in *2016 18th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*, 2016, pp. 233–239.
[14] C. Liu, L. Wang, B. Lang, and Y. Zhou, "Finding effective classifier for malicious URL detection," in *Proceedings of the 2018 2nd International Conference on Management Engineering, Software Engineering and Service Sciences*, Wuhan, China, 2018, pp. 240–244.
[15] P. Zhao and S. C. H. Hoi, "Cost-sensitive online active learning with application to malicious URL detection," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, Chicago, Illinois, USA, 2013, pp. 919–927.
[16] A. A. Mughal, "Building and Securing the Modern Security Operations Center (SOC)," *International Journal of Business Intelligence and Big Data Analytics*, vol. 5, no. 1, pp. 1–15, 2022.
[17] M. Aldwairi and R. Alsalman, "Malurls: Malicious urls classification system," in *Annual International Conference on Information Theory and Applications*, 2011.
[18] M. Akiyama, T. Yagi, and M. Itoh, "Searching Structural Neighborhood of Malicious URLs to Improve Blacklisting," in *2011 IEEE/IPSJ International Symposium on Applications and the Internet*, 2011, pp. 1–10.
[19] D. R. Patil and J. B. Patil, "Malicious URLs Detection Using Decision Tree Classifiers and Majority Voting Technique," *Cybern. Inf. Technol.*, vol. 18, no. 1, pp. 11–29, Mar. 2018.
[20] A. A. Mughal, "Cyber Attacks on OSI Layers: Understanding the Threat Landscape," *Journal of Humanities and Applied Science Research*, vol. 3, no. 1, pp. 1–18, 2020.
[21] B. Eshete and V. N. Venkatakrishnan, "WebWinnow: leveraging exploit kit workflows to detect malicious urls," in *Proceedings of the 4th ACM conference on Data and application security and privacy*, San Antonio, Texas, USA, 2014, pp. 305–312.
[22] S. G. Selvaganapathy and M. Nivaashini, "Deep belief network based detection and categorization of malicious URLs," *Security Journal: A …*, 2018.
[23] R. Verma and A. Das, "What's in a URL: Fast Feature Extraction and Malicious URL Detection," in *Proceedings of the 3rd ACM on International Workshop on Security And Privacy Analytics*, Scottsdale, Arizona, USA, 2017, pp. 55–63.
[24] A. A. Mughal, "Cybersecurity Architecture for the Cloud: Protecting Network in a Virtual Environment," *International Journal of Intelligent Automation and Computing*, vol. 4, no. 1, pp. 35–48, 2021.
[25] M.-S. Lin, C.-Y. Chiu, Y.-J. Lee, and H.-K. Pao, "Malicious URL filtering — A big data application," in *2013 IEEE International Conference on Big Data*, Silicon Valley, CA, USA, 2013, pp. 589–596.
[26] A. Vazhayil, R. Vinayakumar, and K. P. Soman, "Comparative Study of the Detection of Malicious URLs Using Shallow and Deep Networks," in *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 2018, pp. 1–6.
[27] A. A. Mughal, "A COMPREHENSIVE STUDY OF PRACTICAL TECHNIQUES AND METHODOLOGIES IN INCIDENT-BASED APPROACHES FOR CYBER FORENSICS," *TJSTIDC*, vol. 2, no. 1, pp. 1–18, Jan. 2019.

[28] Q. Ye, Z. Zhang, and R. Law, "Sentiment classification of online reviews to travel destinations by supervised machine learning approaches," *Expert Syst. Appl.*, vol. 36, no. 3, Part 2, pp. 6527–6535, Apr. 2009.
[29] A. A. Mughal, "Artificial Intelligence in Information Security: Exploring the Advantages, Challenges, and Future Directions," *Journal of Artificial Intelligence and Machine*, 2018.
[30] P. Márquez-Neila, C. Fisher, R. Sznitman, and K. Heng, "Supervised machine learning for analysing spectra of exoplanetary atmospheres," *Nature Astronomy*, vol. 2, no. 9, pp. 719–724, Jun. 2018.
[31] L. J. Jensen and A. Bateman, "The rise and fall of supervised machine learning techniques," *Bioinformatics*, vol. 27, no. 24, pp. 3331–3332, Dec. 2011.
[32] A. A. Mughal, "The Art of Cybersecurity: Defense in Depth Strategy for Robust Protection," *International Journal of Intelligent Automation and Computing*, vol. 1, no. 1, pp. 1–20, 2018.
[33] S. B. Kotsiantis, I. Zaharakis, and P. Pintelas, "Supervised machine learning: A review of classification techniques," *intelligence applications in …*, 2007.
[34] F. Y. Osisanwo *et al.*, "Supervised machine learning algorithms: classification and comparison," *International Journal of Computer Trends and Technology (IJCTT)*, vol. 48, no. 3, pp. 128–138, 2017.