BEYOND PHISH: Toward Detecting Fraudulent e-Commerce Websites at Scale

Marzieh Bitaab*, Haehyun Cho†, Adam Oest‡, Zhuoer Lyu*, Wei Wang§, Jorij Abraham¶,
Ruoyu Wang*, Tiffany Bao*, Yan Shoshitaishvili*, Adam Doupé*
*Arizona State University, †Soongsil University, ‡PayPal, Inc., §Palo Alto Networks, ¶Scam Adviser
{mbitaab, zlyu15, fishw, tbao, yans, doupe}@asu.edu, haehyun@ssu.ac.kr, aoest@paypal.com,
wewang@paloaltonetworks.com, jorij.abraham@ecommercefoundation.org

2023 IEEE Symposium on Security and Privacy (SP) | 978-1-6654-9336-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/SP46215.2023.10179461

***Abstract*—Despite recent advancements in malicious website detection and phishing mitigation, the security ecosystem has paid little attention to Fraudulent e-Commerce Websites (FCWs), such as fraudulent shopping websites, fake charities, and cryptocurrency scam websites. Even worse, there are no active large-scale mitigation systems or publicly available datasets for FCWs.**

In this paper, we first propose an efficient and automated approach to gather FCWs through crowdsourcing. We identify eight different types of non-phishing FCWs and derive key defining characteristics. Then, we find that anti-phishing mitigation systems, such as Google Safe Browsing, have a detection rate of just 0.46% on our dataset. We create a classifier, BEYOND PHISH, to identify FCWs using manually defined features based on our analysis. Validating BEYOND PHISH on never-before-seen (untrained and untested data) through a user study indicates that our system has a high detection rate and a low false positive rate of 98.34% and 1.34%, respectively. Lastly, we collaborated with a major Internet security company, Palo Alto Networks, as well as a major financial services provider, to evaluate our classifier on manually labeled real-world data. The model achieves a false positive rate of 2.46% and a 94.88% detection rate, showing potential for real-world defense against FCWs.

I. INTRODUCTION

Internet users are under constant attacks from cybercriminals who want to defraud them of their hard-earned money. While phishing is a well-known and well-studied phenomenon where the ultimate goal of the attacker is to steal *information* from the user such as passwords and social security numbers [1–6], other types of cybercrime attempt to *directly defraud* the user.

For instance, Carpineto and Romano [7] studied the problem of how to detect *fake online shopping* scams, where cybercriminals create realistic looking online shopping websites and trick users into purchasing goods that never arrive. Another similar category of scam is *pet scams*, described by Price [8], where victims purchase or adopt pets on a fake website that never arrive. Recently, Bitaab et al. [9] identified fake charity websites that took advantage of the COVID-19 pandemic.

We believe that these types of scams represent a larger category of threats, and we call these *fraudulent e-commerce websites* (*FCWs* for short). The key difference that distinguishes these attacks from phishing attacks is the goal of the attacker—miscreants lure users into spending money on *non-existing or misleading items or services*. Furthermore, FCWs do *not* necessarily impersonate well-known brands, but, instead, they mimic the behavior and user experience of legitimate e-commerce websites.

Modern defenses against phishing attacks are well-studied and ubiquitous in the web ecosystem [1–6]. Major web browsers have incorporated client-side anti-phishing heuristics along with server-side blocklists to detect phishing websites. Unfortunately, existing phishing defenses do not work for FCWs because the detection heuristics typically use the familiarity between the phishing website and the authentic websites.

Despite prior research on some categories of FCWs, such as fake online stores [7, 10–12], pet scams [8], and charity scams [9], detecting FCWs at the ecosystem level remains an open problem, with multiple challenges holding back large-scale, ecosystem-level detection similar to what is seen for phishing attacks. First, to the best of our knowledge, there is no up-to-date, publicly available dataset for detecting FCWs, which raises obstacles for using machine-learning approaches for detection. Previous research used spam datasets such as spamscatter [13] or datasets that are not public [14, 15]. These spam-sourced datasets are insufficient for detecting FCWs as fewer than 30% of URLs in spam emails represent FCWs [13], and without techniques for filtering them, the resulting classifiers are trained on noisy data.

Therefore, we need to collect an up-to-date, clean dataset of FCWs to correctly reflect the status quo. However, curating a list of FCWs is a challenging task as such websites disappear quickly. In addition, FCWs are not limited to one source of distribution such as emails, and thus, are difficult to collect. Furthermore, because the content of FCWs plays an important role in detecting them [16, 17], employing naive methods such as blind web crawling can result in an unsuitable dataset to study FCWs. Lastly, the data must consist of actual FCWs. Manually verifying each FCWs is a time-consuming task that requires expertise in this area.

Another challenge is that FCWs evolve over time. For example, we observed that, in the past, fake online shopping websites used exceedingly low prices to attract customers. However, newer fraudulent shopping websites quote reasonable prices (perhaps because users are wary of deals that are "too good to be true").

In this paper, we first focus on collecting a comprehensive dataset containing different types of FCWs. This dataset reveals additional categories of FCWs: fake investing websites, fake delivery websites, fake educational services, fake adult content and dating, as well as other, less prominent examples. Then, we leverage our observations to design discriminative detection features. Finally, we propose a method to detect FCWs. Specifically, we aim to answer the following research questions:

DOI 10.1109/SP46215.2023.00152

- RQ1: How can we collect and label a comprehensive dataset to study the characteristics of FCWs?
- RQ2: How effective are the current defense systems in protecting users from FCWs?
- RQ3: How can we detect FCWs at scale by leveraging their unique characteristics?

To answer these research questions, we first designed a crowdsourced approach to gather both legitimate e-commerce websites and FCWs from a popular forum aimed at manual FCW detection. Then, we use sentiment analysis techniques to automatically detect FCWs based on users' responses. Next, we study the effectiveness of current mitigation systems on the collected data. Due to the lack of defenses against FCWs, as also confirmed by Bitaab et al. [9], we study the common characteristics of FCWs to define indicative features. For instance, we find that having social media logos without links to a valid social media account is a strong indicator of FCWs.

Using these features, we then propose a mechanism to detect FCWs. To evaluate our detection, we perform extensive experiments, including a user study with human participants by deploying a social media bot to collect users' feedback on the classification decisions. The user study demonstrates that our system has a high detection rate of 98.34% and a low false positive rate of 1.34% We then collaborated with Palo Alto Networks and a major financial organization to validate our model and the collected data.

In summary, the contributions of this paper are as follows:

- We leverage social media to collect 6,127 FCWs that are actively luring victims. We further apply sentiment analysis techniques on the collected dataset to label the URLs and build a comprehensive dataset.
- We identify common features among FCWs and build a neural network model called BEYOND PHISH to detect them.
- We perform empirical evaluations including a user study to illustrate the performance of BEYOND PHISH on previously unseen FCWs. BEYOND PHISH achieves low false positive rates of $0.47\%, 1.37\%$ and high accuracy of $93.41\%, 98.38\%$ on the evaluation set and user study, respectively. Additionally, through our collaboration with Palo Alto Networks and a major financial organization, we validate both the collected dataset and the model.

This analysis of FCWs benefits many parties: (1) researchers can further investigate mitigating this type of scam, (2) payment processors can protect users by taking appropriate actions, and (3) browsers can incorporate FCW mitigation systems to warn users. To further our goals of reproducible science, we release our collected datasets (though not the proprietary dataset provided by Palo Alto Networks), source code, and the BEYOND PHISH model[1].

II. BACKGROUND

Phishing is a well-known type of scam where miscreants masquerade as trustworthy entities. The objective of phishing is to deceive people into releasing sensitive information such as social security number or account credentials. Phishing attacks usually involve deceptive websites and are initiated by lures that trick victims into visiting the phishing websites, and these lures typically use email or private messages [1].

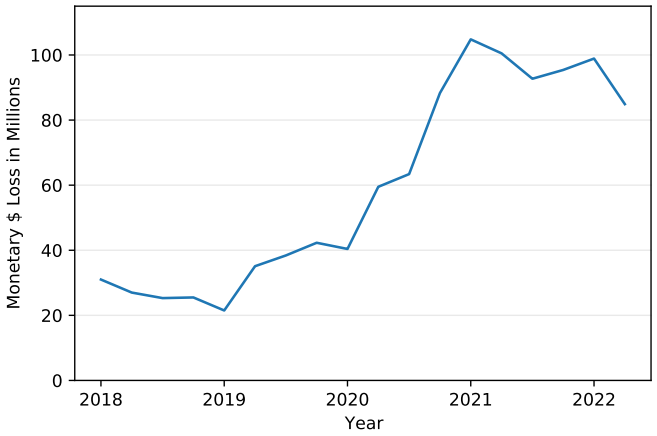


Fig. 1: Total monetary loss to online shopping websites per quarter according to the FTC [18].

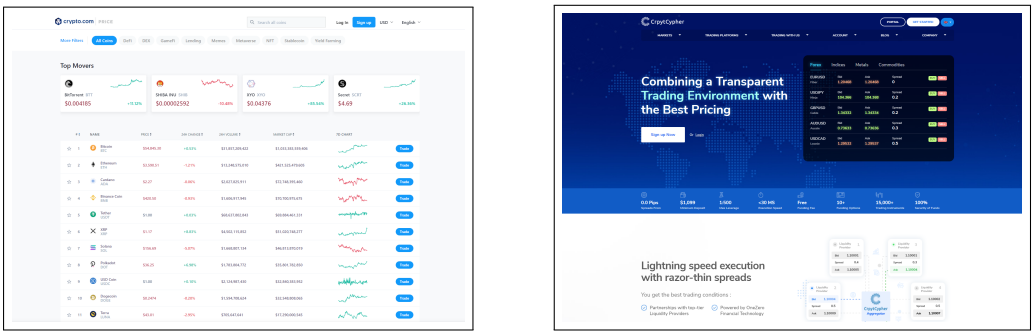
Fraudulent e-Commerce Websites (FCWs), however, have a different objective and *monetization* approach compared to phishing. Rather than impersonating known e-commerce entities/brands as in phishing, FCWs attackers create fraudulent websites that appear to be legitimate e-commerce websites. The goal of FCWs is to trick victims into paying for bogus goods or services that never arrive.

Figure 1 shows the overall increase in the amount of financial loss reported by the FTC due solely to fake online shopping websites (one category of FCWs) [18]. With peak monetary loses at above $\$100$ million in the first quarter of 2021 alone, this motivates our study of FCWs.

Because phishing websites impersonate well-known brands, they can be detected using features such as their URLs or visual similarity to legitimate websites [3, 19, 20]. Anti-phishing systems have matured as prior research has proposed myrial mitigation systems [19]. Content take-downs [21], certificate intelligence [22], URL and content blocklisting, and classification [2, 23, 24] are several examples of anti-phishing systems that protect users. In addition, Internet users heavily rely on browser-based phishing detection systems such as Google Safe Browsing [25] and Microsoft Windows Defender. Due to their scale and always-on nature [19], they can be viewed as the first line of defense against phishing websites [26]. In contrast, there are no general fraudulent e-commerce detection systems to protect users.

FCWs, unlike phishing websites, create a facade with similar *behavior* to legitimate websites. To do so, they have characteristics such as a well-defined website theme, social media logos, proper contact page, and valid payment gateways. Therefore, most FCWs cannot be easily detected by their appearance or URL related features alone. Impersonating a

[1]github.com/mbitaab/beyondphish



(a) Legitimate website.

(b) Scam website (FCW).

Fig. 2: Example of a legitimate website and a similar FCW. The fraudulent website has a similar *user experience* to the legitimate one by using a convenient website template, payment method, and having social media icons.

legitimate website's *behavior* and *user experience*, instead of replicating the exact look, is the main difference between phishing and FCWs. The monetization goal of FCWs is to trick the user into purchasing a product or service that never arrives or is substantially different than what was advertised, whereas the monetization goal of phishing is to steal credentials (that can later be monetized) [1]. Figure 2 shows two eCommerce websites: Figure 2a is legitimate and Figure 2b is an FCW. Both websites have satisfying user interfaces: both appear to have social media and a well-made, responsive design.

A. Known FCWs examples

Fake online shopping. These websites mimic legitimate shopping website to lure users, appearing to sell rare, desirable, or discounted items [7, 10–12]. Unique items make fake online shopping websites more visible to users when they try to search for the items, and discounted items attract more visitors. Specifically, miscreants use different techniques such as limited time offers or showing recent purchases as notifications to pressure users into buying the items [9].

Pet scams. Pet scams claim to sell pets below market price [8]. Miscreants aim to make victims emotionally attached to a fictitious pet. These websites appear legitimate at first glance because most of them do not have any on-site payment option. Rather, miscreants ask users to fill out an application to be reviewed for eligibility.

Fake charity websites. These scam websites use real world scenarios to take advantage of users' empathy. Fake charity or donation websites deceive users into thinking that they are helping people in need while in reality a miscreant receives the money. Usually miscreants exploit recent crisis. For example, during the COVID-19 pandemic, attackers used the pandemic to create fake charity websites pretending to help victims [9].

Fake cryptocurrency or stock market scams. People consider cryptocurrency and/or the stock market as a desirable investment opportunity [27]. This creates an opportunity for miscreants to lure users into investing in their fraudulent cryptocurrency or high yield stocks, and Xia et al. [28] identified several fake cryptocurrency exchanges.

III. Goal and Scope

The dominant practice to acquire ground truth on malicious domains is to extract them from various blocklists such as Spamhaus [34], PhishTank [35], and OpenPhish [36], or more general malicious domains such as VirusTotal [37]. However, these blocklists have a number of subtle issues [38, 39]: Previous research uses different techniques to acquire the truth labels for various datasets, and they revealed high false positive and false negative rates regarding the aforementioned datasets by comparing the dataset's labels [40, 41]. Moreover, the data collected for one approach usually would not transfer to another approach in a different domain. Considering the lack of reliable data of fraudulent e-commerce websites (FCWs), we design an automated method for gathering a crowdsourced dataset (Section IV-A).

Prior work on detecting different classes of scams is shown in Table I. Our work is different from prior work because we focus on the novel problem of analyzing and detecting the broad category of FCWs, while prior work focused on detecting spam, phishing, or only fake online shopping websites. To this end, we first collect a comprehensive dataset of FCWs through crowdsourcing. Then, we analyze different FCWs variants to discover their characteristics. Considering legitimate e-commerce websites and fraudulent e-commerce websites' characteristics, we design a detection method. One of the detection challenges is feature extraction, which has a significant impact on the performance of the detection model [42]. Therefore, we leverage our observations to define features uniquely suited for this domain.

IV. Data Collection, Validation, and Analysis

To build a supervised machine learning model to detect fraudulent e-commerce websites (FCWs), we first collect a labeled dataset with both fraudulent e-commerce and legitimate websites. To this end, we first collect likely FCWs posted to a social media forum (a subreddit dedicated to discussing scam websites). We first label the sites in this dataset as fraudulent e-commerce or legitimate through a sentiment analysis on the corresponding forum discussion. Then, we evaluate the labeling process using input from Palo Alto Networks and a

TABLE I: Comparison of prior work on detecting different classes of scams, including datasets and sizes. Everything above the dashed line focuses on phishing, spam, malicious websites, or other scams where the goal is to trick users into revealing sensitive information. The work below the dashed line is related to BEYOND PHISH, and BEYOND PHISH has the largest dataset by far (and includes types of FCWs that are not included in prior work).

Previous Research	Datasets	# Samples	Targeted Domain
Garera et al. [29]	Google Toolbar Dataset	1,263	Phishing
Kolari et al. [30]	BHOME, BSUB, SPLOG	10,800	Spam
PREDATOR [14]	Spamhaus + URIBL + Spam trap	1,284,664	Spam
Ma et al. [31]	DMOZ + Yahoo + Spamscatter + PhishTank	15,000	Phishing and spam
Ma et al. [32]	Yahoo + Webspam	20,000	Phishing and spam
Choi et al. [16]	DMOZ + jwSpamSpy + PhishTank + DNS-BH	32,000	Phishing, spam, and malware
Delta [17]	Wepawet	12,464,920	Malicious websites
Surveylance [15]	Alexa + Google Search Results	5,173	Scam survey websites
Srinivasan et al. [33]	Google Search Results	124,003	Scam technical support websites
Wadleigh et al. [10]	Google Search Results	6,979	Fake online shopping
Carpineto et al. [7]	Alexa + Google Search Results	1,000	Fake online shopping
Beltzung et al. [11]	Watchlist Internet	5,919	Fake online shopping
Mostard et al. [12]	Thuiswaarborg	3,332	Fake online shopping
Proposed BEYOND PHISH	Reddit + Palo Alto Networks	18,549	Fraudulent eCommerce websites

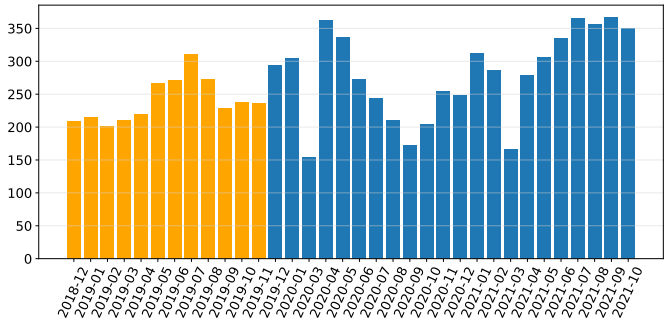


Fig. 3: Posts per month in the /r/Scams subreddit that include URLs. Bars in yellow are archived posts that we crawled in December 2019, while the bars in blue were crawled live.

manual validation in Section IV-C to understand the quality of the dataset (in terms of false positive and false negative rates). Next, we dig into the data to understand the FCWs categories. Finally, we discuss common characteristics among FCWs in Section IV-E.

A. Data Collection

A major challenge in analyzing FCWs is that there are no up-to-date or publicly available datasets (discussed in Section II). To address these challenges, we take advantage of the social media platform Reddit, where a significant number of $\approx$ 330 million users discuss various topics in dedicated *subreddits* [43]. We used Reddit as it is a structured and monitored (by moderators) social media platform where discussions are categorized into different areas of interest called a subreddit [44]. Within each subreddit, users discuss specific topics through postings called *submissions*. We collect a dataset of users' submissions and the corresponding comments based on a subreddit dedicated to discussing FCWs.

As a preliminary step in our study, we continuously crawled the /r/Scams subreddit from December 2019 to October 2021. This subreddit has $354,000$ members and users discuss if suspicious websites, emails, and calls are fraudulent or legitimate, and also share their experience. We collected $16,072$ submissions of which $6,233$ contained live URLs. For each collected URL, we saved the full HTML source of the webpage and its domain registration information through WHOIS [45]. In addition, we also retroactively crawled the prior year of posts from December 2018 to December 2019, resulting in $17,442$ additional submissions with $2,881$ live (at the time of our crawl) URLs. In total, we analyzed $33,514$ submissions and acquired $9,114$ live URLs. The number of submissions for each month is provided in Figure 3. To measure the current ecosystem-level protection against FCWs, we study the effectiveness of widely used mitigation systems such as Google Safe Browsing in Section VI-B.

B. Data Labeling

The most important step in curating our dataset is to label the actual FCWs among the $9,114$ suspicious URLs we crawled. To this end, we automate the labeling process by analyzing the users' comments on each URL.

After manually examining /r/Scams posts, we noticed that users' comments can be used to understand the legitimacy of a suspicious URL. For example, if the shared URL is an FCW, users may comment *"don't buy"* or *"common scam, move on."* Because each submission and its comments are rigorously monitored by the moderators of the /r/Scams subreddit, we consider the comments credible. The moderators remove deceptive posts and comments to protect users [44]. All of the submissions have at least one comment, with an average of 9 and a median of 4 comments per submission.

To automate the labeling process based on users' comments, we train and use a Natural Language Processing (NLP) model that classifies each comment as *positive* or *negative* indicating whether or not each comment is a positive sentiment. Figure 4 shows an overview of the dataset labeling process.

To create an NLP model capable of classifying users' comments, we use a neural network classifier on top of the BERT model [46]. BERT is a language model that can be

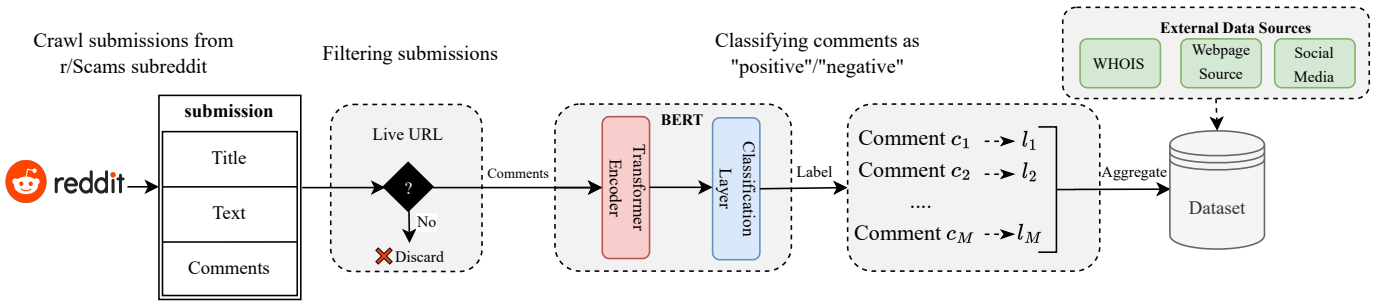


Fig. 4: The fraudulent website gathering and labeling process.

used to perform various NLP tasks such as text generation, sentiment analysis, and question answering. To use BERT for sentiment analysis, we first convert each comment to a context vector $\mathbf{g}$, containing important information about the comment. Then, we use the neural network classifier to label the context vector $\mathbf{g}$ as positive or negative. To train the model, we use the Stanford Sentiment Treebank binary classification dataset [47] that contains $215,154$ phrases along with $\{positive, negative\}$ labels. We use this dataset to train a general sentiment classification model that can be used to accurately predict the sentiment of the comments.

To classify and assign a label to every URL, we first classify each comment on a submission. Then, we label a submission's URL as fraudulent if there are more negative comments than positive comments. Appendix D provides an example of the labeling process. Considering $\mathbf{c}_i$ as the context of i^{th} comment and $\hat{y}_i^c \in \{positive, negative\}$ as its predicted label, we use the following equation to determine the URL's label in submission y:

$$y = \begin{cases} fraudulent & \sum_i^m I(\hat{y}_i^c = n) > \sum_i^m I(\hat{y}_i^c = p) \\ legitimate & otherwise \end{cases}$$

where $I(.)$ is the indicator function and m shows the number of comments of the submission.

C. Data Validation

We design two experiments to validate and assess the label noise in our labeled data. First, we collaborated with Palo Alto Networks to test our model against a real-world dataset. We place the details of this experiment in Section VI-D because it uses our completed model (which we describe later). Ultimately, validating our model on this previously unseen (and manually verified dataset) had a false positive rate of 2.46% and a detection rate of 94.88%.

In the second validation step, we selected a subset of 2,000 random websites within our dataset and hired three security experts. We trained them to detect the legitimacy of a website by manually interacting with the website, reading Reddit threads, and using search engines. The participants did not collaborate and did not have any information about the assigned labels. We used a majority vote to label each URL. We provide details of manual labeling process in Appendix E. We calculated a 1.98% false positive rate (FPR) and 1.63% false negative rate (FNR), accounting for 1.86% label noise in our dataset.

TABLE II: Distribution of FCWs in our Reddit dataset.

#	Shopping Website Category	% of submissions
1	Fake Online Shopping	60.38%
2	Pet Scam	20.52%
3	Charity Websites	6.04%
4	Cryptocurrency and Stock Market	3.91%
5	Delivery Websites	3.36%
6	Education Related Websites	2.92%
7	Adult Content and Dating	2.76%
8	Other	0.10%

This label noise can be caused by both human or machine error during the labeling process of the dataset. According to Bishop [48], one way to prevent overfitting in neural networks is to add label noise to the data. Moreover, recent studies have shown that neural networks are robust to even more than 20% of label noise [49, 50]. We, therefore, believe that the label noise of our dataset is acceptable.

One may argue that using this dataset biases the machine learning model toward detecting websites that *human users* detect as FCWs. We try to answer this concern from two aspects. First, the Reddit data also includes URLs that users share after they fell victim to the scam. Secondly, to demonstrate the model's generalizability, we validated the trained model on an unseen dataset obtained from Palo Alto Networks, presented in Section VI-D.

D. Categorizing fraudulent e-commerce websites

After collecting the dataset, we categorized the labeled FCWs to understand the different types of fraudulent e-commerce. To this end, we analyzed the source code of the collected websites (Section IV-A) to assign each website to a category using a manually curated set of keywords provided in Appendix C. Table II shows the percentage of each fraudulent website category in the collected dataset.

Of the FCWs categories previously discussed in Section II-A, *fake online shopping* scams are the most common at 60.38% in our dataset, *pet scams* are 20.52% of our dataset, *fake charity websites* are third at 6.04%, and *cryptocurrency and stock market scams* are fourth at 3.91%. We also find other categories of FCWs:

Delivery Websites. Fraudulent delivery websites act as a support website for fraudsters who want to sell items to users. They can be used in pet scams and fraudulent online shopping websites to show fake tracking history for non-existent packages, and may even seek to collect additional personal information. The delivery websites can prolong the longevity of other FCWs, making their users believe that the problem is within the shipping company and not the FCWs.

Education Related Websites. These fraudulent websites sell services that target students who need help in writing essays, research papers, and other types of homework assignments. In some of these websites, rather than *only* taking the victim's money and not delivering the service, the miscreants additionally extort the students in exchange for not reporting them to their school [51].

Adult Content and Dating. This category includes websites related to adult content (providing fake adult content) and fake dating websites.

Other. Other types of fraudulent websites such as job offer scams and credit services are included in this category.

E. Characteristics of fraudulent e-commerce websites

Now that we have a labeled dataset of FCWs, we manually examine them to identify features that distinguish FCWs from legitimate e-commerce websites.

Historically, one of the common characteristics among fake shopping websites was cheap prices: several prior studies consider the discount amount as a feature that can distinguish between fraudulent and legitimate websites [7, 10]. However, we observe that most of the recent FCWs do not offer significant discounts. We believe that having a typical price range for items blends fraudulent e-commerce with legitimate websites.

People rely on social media to discover unknown brands. Hence, a social media presence can increase a brand's authority [52]. Both popular brands and also new ones (even non-shopping websites) use social media to increase visibility. In the beginning of our data collection, December 2018, the lack of social media logos presence was a common trend in FCWs. However, newer FCWs are more likely to include social media on their website, yet, the added social media logos do not link to the FCWs' social media account. They either add only logos of various social media websites (with no links), or they include logos with an invalid link. The invalid link can be any link to a social media page that is not the website's actual business profile.

In Figure 10 in Appendix B we provide examples of social media icon misuse in FCWs. Among our collected data, 81.47% of FCWs do not have any social media link in their content as indicated in Figure 10a, and 16.80% of them contain an invalid social media link similar to Figure 10b. In contrast, only 33.85% of legitimate websites do not have any social media links. We believe the reason that FCWs do not include valid social media links is to avoid their identity being revealed by victims and shared on social media.

Another characteristic of FCWs is associated with their top-level domain (TLD). Miscreants want to spend less money to acquire domains, and they tend to use cheap TLDs [14, 53]. In our dataset, 29.46% of FCWs use cheap TLDs such as *.xyz*, *.store*, and *.shop*, in comparison to 3.76% such TLDs among legitimate websites. Moreover, we noticed that fraudulent websites use cheap registrars more often. Considering popular cheap registrars (i.e., *Namecheap*, *GoDaddy*, *Porkbun*, *NameSilo*, *Danesco* and *Hostinger* [8]), 57.16% of FCWs use cheap registrars, whereas 27.76% of legitimate websites use them.

Shopify[2] is an e-commerce platform that simplifies creating an online shopping website. Sellers can easily create a shopping website by uploading their products, payment information, and choosing a theme to make their online store. Within our collected Reddit dataset, 61.35% of the Shopify stores were FCWs. The high rate of Shopify FCWs reveals the fact that miscreants attempt to take advantage of such platforms to create their fake shops [54].

V. Fraudulent E-Commerce Website Classification

In Section IV, we categorized Fraudulent e-Commerce Websites (FCWs) and found commonalities among them. In this section, we propose a detection method based on the identified common characteristics. Our goal is to create a model, which we call BEYOND PHISH, that can be used to detect FCWs from websites in the wild. We manually define features based on our analysis of the collected dataset (described in Section IV-E). We then create a model for detecting FCWs by leveraging features from the website's content, DNS records, website's URL, and its social media.

Figure 5 shows the high-level overview of our system. The dataset (Section IV-A) is passed through a feature extraction module which makes use of the content, DNS records, URLs, and social media links. This process outputs a feature vector for each URL, enabling us to learn a classifier $\mathcal{F}$ capable of separating fraudulent e-commerce from legitimate websites. To evaluate BEYOND PHISH, we monitor the /r/Scams subreddit to extract new posts containing URLs. Next, we design a human evaluation study to validate the performance of the trained classifier $\mathcal{F}$ using participants' feedback.

A. Feature Selection and Extraction

Each type of website, fraudulent or legitimate, has different characteristics that helps the classifier to distinguish between them. We categorize features into four main groups: content-based, DNS-based, URL-based, and social media-based, and Table III summarizes the features.

Content-based Features refer to the features which are based on the HTML source code of the website:

Valid social media links: Legitimate websites commonly use social media for marketing purposes, and miscreants mimic legitimate websites by including a logo or links to social

[2]We have disclosed the list of suspicious shopping websites that were detected by our proposed method.

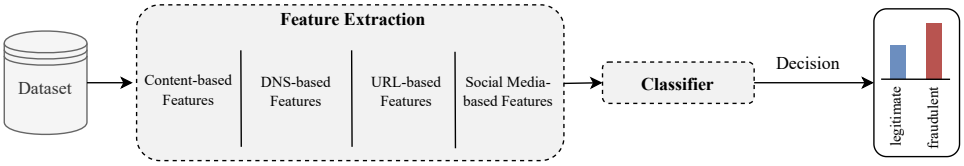


Fig. 5: The training process of BEYOND PHISH.

TABLE III: Summary of defined features. We consider DNS, URL, website's content, and its social media to define features.

Feature category	Feature name	Feature type
Content-based	Valid social media	Categorical
	# of external links	Quantitative
	# of script tags	Quantitative
DNS-based	Domain age	Quantitative
	Registration period	Ordinal
	Domain country	Categorical
	Host country	Categorical
	Same host & domain country	Ordinal
	Cheap registrar	Categorical
	Domain privacy	Ordinal
	Top 100K	Ordinal
URL-based	Cheap TLD	Ordinal
	Includes hyphen	Ordinal
	Includes digits	Ordinal
	Sub-domain level	Quantitative
Social Media-based	# of followers	Categorical
	Account age	Quantitative
	# of likes	Quantitative

media sites such as Twitter. However, these links are mostly invalid. An invalid link refers to a social media URL that points to another business, login page, or any other part of the social media website which is not an actual profile. We use the three most popular social media sites: Twitter, Instagram, and Facebook. For each social media we consider a separate feature and the feature's value is set to -1 if the website does not contain any link to the social media, 0 if there is an invalid social media link, and 1 if there is a valid social media link. Because this feature is extracted from the source code of a website, we categorize it under the content-based features.

Number of external links: FCWs often create closed environments (no outgoing links), while legitimate sites have large numbers of outgoing links. We observe that, on average, legitimate websites have 1.91 times more outgoing links than FCWs. This feature is a discrete value greater than 0.

Number of script tags: We extract the number of `script` tags from the HTML source of the website, including both external (with a `src` attribute) and inline. A high number of `script` tags can indicate malicious activity [55, 56]. In our dataset we observe an average of 23 tags for FCWs, while legitimate websites contain 13 script tags on average.

DNS-based Features are based on the public WHOIS information regarding the URL's most recent domain registration:

Age of domain: Miscreants abandon their domains after blocklisting or being reported. Thus, it is common for FCWs to have a recent registration date. We assign the number of days between the current date and the website's creation date to this feature. According to our data, FCWs have a short average age of 2 years, while legitimate ones have an average age of 13 years.

Registration period: The "Expiration Date" in the WHOIS information indicates when the domain is going to expire. A domain can be registered for one to ten years. Because longer registrations cost more, miscreants tend to register domains for short periods. The value of this feature is the number of years between the expiration and the creation date. Our data indicates that most of the FCWs are only registered for 1 year.

Domain registration country: According to our observations, 56.49% of the FCWs use a country other than US, CA, and UK as their registered country. Among these FCWs, Panama ("PA") is the most commonly used country accounting for 42.86%. This trend has not changed since December 2018. To include this information in the feature vector, we consider a one-hot vector indicating the registrar country based on the domain's WHOIS record.

Host country: Similar to the domain registrar country, we one-hot encode the country, based on the website's IP address.

Same host and domain country: The website's host country does not need to be the same as the registrar country. This feature indicates whether the host and domain country of a website are the same or not. Our analysis shows that legitimate websites have 3 times greater odds of having the same host and domain country than FCWs.

Cheap registrar: Miscreants mostly register their domains through registrars with the cheapest price. *Namecheap*, *GoDaddy*, *Porkbun*, *NameSilo*, *Danesco*, and *Hostinger* are among the top registrars that offer domains at low prices. If the website's registrar is one of the aforementioned registrars, this feature will be set to 1, otherwise 0.

Domain privacy: Although private WHOIS is used for legitimate websites [57], it is widely used among FCWs. A third of FCWs use private WHOIS to hide their identity in our dataset. This binary feature indicates whether a website is using private WHOIS.

Alexa top 100K: We expect FCWs to draw less traffic than legitimate ones, adding a binary feature indicating if the website's Alexa rank is below 100,000.

URL-based Features consider parts of the URL:

Cheap TLD: Our observations show that 28.21% of FCWs use cheap TLDs, such as *.xyz* and *.store*. According to the average price of domain registrations, we consider the 50 cheapest TLDs to construct a binary feature indicating if the URL uses cheap TLD name or not. We chose the top 50 TLDs as their average price is less than $\$2.00$.

Hyphen in domain: One way to obfuscate well-known domain names is to add a hyphen ("-") in the name. This binary feature shows if the domain name includes a hyphen [14].
Digit in domain: Another method of domain name obfuscation is to use digits in the domain name. This binary feature shows if the domain name includes digits [14].
Sub-domain level: Our analysis shows that legitimate websites have 8.86 sub-domains on average while FCWs have 19.87. We believe this is because some FCWs websites attempt to confuse users by placing brand names in subdomains (such as nike.shoes.example.com) [1].

Social Media-based Features provide information about the social media profiles related to a website:
Age of social media account: As more and more fraudulent websites try to mimic legitimate websites and include social media links in their websites, it is important to check the credibility of the corresponding social media accounts. The age of a social media account is one of the indicators of credibility of a business. We crawled the creation date of Facebook and Twitter accounts to calculate their age. Because Instagram does not provide the creation date, we cannot consider the age of Instagram accounts.
Number of followers: Another feature that provides insight regarding the credibility of social media is the number of followers. The number of followers could have a wide range depending on different factors such as the popularity of a brand, so we bucketed this feature into five categories as follows: 1: $1 \geq f < 5000$, 2: $5000 \geq f < 15000$, 3: $15000 \geq f < 50000$, 4: $50000 \geq f < 100000$, and 5: $100000 \geq f$.
Number of likes: This feature is only applicable to websites with Facebook links, which shows how many users have liked the website's Facebook page. To quantize this feature we use the same approach as the number of followers.

B. Model Architecture

We create our FCWs detection model BEYOND PHISH based on the defined features. The input to the classifier is a feature vector $\mathbf{x}$ containing attributes described in Section V-A, and the output is the probability p of a website being legitimate or fraudulent. We consider several classifiers to detect FCWs including random forest [58], XGBoost [59], SVM [60], and a feed-forward neural network. The details of the classifier and the implementation details are provided in Appendix A.

VI. EVALUATION

To evaluate the effectiveness of BEYOND PHISH we seek to answer the following questions regarding fraudulent e-commerce websites (FCWs):

- Q1 How reliable is BEYOND PHISH in detecting FCWs?
- Q2 What is the performance of BEYOND PHISH on real-world and unknown data?
- Q3 What are important features for the classifier to detect FCWs?

To answer Q1, we split our gathered data into training and testing sets, then we report the performance metrics on the testing set. We answer Q2 by designing a user study to evaluate the reliability of our detection model on real world data using Reddit users' feedback. Then, we perform another experiment using Alexa domains to further investigate the false positive rate of BEYOND PHISH. We classify $10K$–$20K$ rank, and bottom the $10K$ rank Alexa domains using BEYOND PHISH, then we manually check the websites that are detected as fraudulent. Finally, we evaluate our model on data provided by Palo Alto Networks. We then answer Q3 by leveraging DeepLift [61] on our neural network-based model to analyze the impact of features on its decisions.

To report the evaluation metrics, we use 5-fold cross validation. The training parameters for trained models are provided in Appendix A.

TABLE IV: Statistics and source of dataset for each experiment.

Purpose	Dataset	Legitimate	FCWs
Training and Testing	Reddit	2,365	6,127
	Palo Alto Networks	9,965	0
Classification in the Wild	User Study	298	1,925
	Palo Alto Networks	13,982	10,054

A. Dataset

We collaborated with Palo Alto Networks to add legitimate e-commerce websites to our dataset for training and testing purposes. They deploy various web scanners and use state-of-the-art URL filtering techniques to categorize URLs. Palo Alto Networks provided us with e-commerce websites that are verified as legitimate by security experts. While we cannot disclose all the details of this proprietary dataset, it was collected over 10 years based on the popularity of websites. Newly registered domains have also been labeled by human experts and added to the dataset. The purpose of adding newly registered domains is to include less reputable e-commerce websites as well. By combining legitimate e-commerce from Palo Alto Networks and collected URLs from Reddit, our dataset includes $12,330$ legitimate and $6,127$ fraudulent URLs and corresponding features. Training and testing dataset statistics are in Table IV.

B. Current Ecosystem Defenses

We demonstrate the ecosystem's lack of defense for FCWs by analyzing the performance of existing mitigation systems. Google Safe Browsing (GSB) is the most impactful blocklist that protects 81.42% of the Internet traffic [25] from phishing and malware. In this paper, we use the GSB API to check the detection status of the URLs in our dataset. Another blocklist provider is the Anti-Phishing Working Group (APWG) that focuses on responses to cybercrime [62]. We evaluate our dataset on both blocklists. APWG and GSB detect only 25 and 10 FCWs within our dataset, respectively. This indicates that current blocklists do not mitigate FCWs, and users are at risk of being exposed to FCWs.

TABLE V: Performance of BEYOND PHISH (BP) with comparison to baselines. This indicates our model is capable of detecting FCWs and also shows the difference between FCWs and phishing domains. As an example, Cantina+ has a low FPR for phishing domains, however, it has a high FPR on our domain, indicating the difference between the two domains.

Model	FPR (↓ better)	Detection Rate (↑ better)	F1 (↑ better)
BP+Random Forest	1.84%	72.79%	0.810
BP+XGBoost	1.84%	91.92%	0.906
BP+SVM	3.33%	78.92%	0.825
BP+NN	1.68%	87.14%	0.924
CheckPhish	0.68%	18.87%	0.012
Cantina+	22.22%	79.72%	0.769
RealTime	3.66%	69.07%	0.773
HAN	22.22%	20.21%	0.295

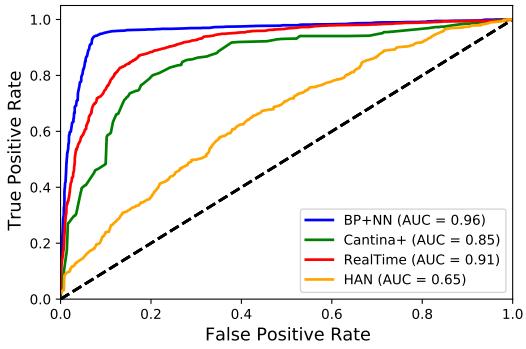


Fig. 6: Receiver Operating Characteristic (ROC) curve of BEYOND PHISH and the baselines.

C. Detection Performance

We evaluate the performance of BEYOND PHISH against the following baselines:

CheckPhish [63]: is a method that claims to detect both phishing and fraudulent websites using computer vision and NLP techniques. This method uses Convolutional Neural Networks (CNN) to analyze a website's appearance and extracts different features from the webpage's source code to concatenate with its visual features [63].

HAN: Inspired by Saxe et al. [64] HAN is content-based. Because we did not have access to the original implementation, we created a similar model to detect FCWs based only on the webpage's source code. In this baseline, we use Hierarchical Attention Networks [65] to classify webpage source code in the collected dataset as fraudulent or legitimate. This method's main advantage is its ease of interpretation, as we can visualize the essential words or parts of the webpage source code that made the classifier choose its label.

Cantina+ [2]: is a content-based machine learning classifier that detects phishing websites based on page, URL, and domain-based features. We use Cantina+ only to highlight the limitations of phishing detections for FCWs, rather than a direct comparison to BeyondPhish. To ensure a fair evaluation, we trained Cantina+ on our collected dataset.

RealTime [11]: detects fake shopping websites using the content of websites for classification. It first extracts the features by calculating TF-IDF for all words in the dataset. Then, it uses XGBoost to classify TF-IDF based vectors.

We planned to include more baselines from prior work that focus on a specific type of FCWs: We contacted the respective authors and unfortunately, we could not obtain the necessary source code or data [14, 33] or the provided code/data was not enough to create a fair baseline [15].

We evaluate the models in terms of false positive rate, detection rate, F1 score, and Receiver Operating Characteristic (ROC) curve. The false positive rate indicates the percentage of benign websites incorrectly labeled as fraudulent. The F1 score is the harmonic mean of precision and recall, while the detection rate shows the percentage of fraudulent websites which are labeled correctly.

The macro-average ROC curve in Figure 6 indicates the diagnostic ability of a binary classifier, created by plotting the true positive rate against the false positive rate at different threshold settings. The content-based baseline is expected to have lower performance as detecting FCWs only based on the source code of the website is not reliable. FCWs have very similar content to legitimate websites and detecting them requires a deeper understanding of semantics. This observation is consistent with the RealTime [11] method that only uses the content of websites to classify fake shopping websites.

As the baselines do not detect most of the FCWs, they have a very low false positive rate; however, labeling fraudulent websites as legitimate leads to a low detection rate that can miss fraudulent websites, exposing users to scams. The high detection rate and low false positive rate of models trained on our designed features makes it more effective at detecting FCWs. Although the Cantina+ model is suitable for detecting phishing websites [2], it does not detect FCWs well. From the results of Cantina+, we conclude that phishing detection models are not suitable for detecting FCWs because of the feature sets that they use. Comparing the results from different methods on our designed features indicates that BP+NN is the best performing at detecting FCWs due to its lower false positive rate and higher F1 score.

To further investigate the effectiveness of BP+NN, we design another experiment. We evaluate BP+NN on a new split of the test-set using a training set of 80% of each of the categories and then testing on the remaining 20% in each category. Note that the detection rate differs from previous experiments due to the nature of the training set. Table VI indicates the results for this experiment. It can be inferred that BP+NN achieves more reliable results compared to the baselines. Although BP+NN does not achieve the highest detection rate in some categories, it has lower FPR and/or higher F1 score.

D. Classification in the Wild

We perform four different experiments to validate the performance of BEYOND PHISH on never-seen, in-the-wild samples. We choose BP+NN as the model because it outperforms other variants of BEYOND PHISH models.

TABLE VI: Performance of classifiers on each FCW category.

Category	Model	FPR (↓ better)	Detection Rate (↑ better)	F1 (↑ better)
Fake Online Shopping	BP+NN	**0.78%**	61.47%	**0.747**
	CheckPhish	0.82%	1.45%	0.028
	RealTime	3.18%	**62.92%**	0.746
Pet Scam	BP+NN	0.83%	**70.58%**	**0.815**
	CheckPhish	**0.29%**	0.0%	0.0
	RealTime	2.18%	62.92%	0.746
Charity Websites	BP+NN	1.51%	**59.25%**	**0.711**
	CheckPhish	**0.79%**	1.98%	0.036
	RealTime	8.33%	54.54%	0.600
Cryptocurrency and Stock Market	BP+NN	2.04%	69.15%	**0.754**
	CheckPhish	**0.61%**	0.0%	0.0
	RealTime	12.76%	**69.56%**	0.711
Delivery Websites	BP+NN	4.28%	**67.64%**	**0.767**
	CheckPhish	**0.0%**	0.0%	0.0
	RealTime	4.16%	62.50%	0.727
Education Related Websites	BP+NN	1.42%	**58.34%**	**0.769**
	CheckPhish	**0.0%**	0.0%	0.0
	RealTime	5.88%	38.46%	0.500
Adult Content and Dating	BP+NN	**1.47%**	**98.52%**	0.583
	CheckPhish	2.95%	13.95%	0.222
	RealTime	4.68%	50.00%	**0.611**

TABLE VII: BEYOND PHISH's performance on in-the-wild samples over 14 months (with /r/Scams user specified labels).

Total # data	FPR	FNR	TPR	TNR	Accuracy
2,223	1.34%	1.66%	98.34%	98.65%	98.38%

User Study. To validate BEYOND PHISH's reliability on real world and unseen (untrained and untested) data, we perform a human study on the /r/Scams subreddit. We followed an IRB-approved protocol, and we did not collect any other information of the participants.

First, we created a Reddit bot which monitors new submissions on /r/Scams. Each new submission is checked against a regular expression to process only those containing URL(s). Then, according to the testing phase in Figure 5, the URLs are passed to the feature extraction module followed by the classifier. The resulting label is posted as a comment on the submission by the bot to inform users about its decision. Moreover, /r/Scams users can submit their feedback by clicking on the *agree* or *disagree* links on the bot's comments.

In the posted comment we informed participants of the purpose of the Reddit bot, explaining its task and the information it collects. Table IV shows the statistics of the dataset used in this experiment. Over the course of 14 months, we collected $13,917$ responses from $8,174$ users on $2,223$ submissions where 298 URLs are legitimate and $1,925$ are fraudulent. The responses are solely collected based on the users who click on the *agree* or *disagree* links on the bot's comments. After examining the results, we find that BEYOND PHISH predicted the correct label 98.38% of the time. For each of the bot's response, we consider a majority vote on users' feedback as the true label. The detailed results are shown in Table VII. Comparing Table VII and Table V indicates the consistency of BEYOND PHISH's performance on unknown and untrained data. Comparing the user study's 98.38% accuracy to BEYOND PHISH's accuracy of 93.41% on the test data shows that BEYOND PHISH can deliver performance close to that of manual analysis.

Alexa Domains. In the prior experiment, the previously unseen samples were submitted to /r/Scams, and these samples are potentially biased because a human user has already decided that they might be FCWs (indicated by the submission to /r/Scams). Therefore, to understand BEYOND PHISH's false positive rate on legitimate websites (which are unlikely to be submitted to /r/Scams), we design another experiment using Alexa domains with a rank between $10,000$ and $20,000$. In this experiment we first train BEYOND PHISH without considering the *Alexa top 100K* feature (as this feature would bias the results) and classify each domain's URL using BEYOND PHISH. Each URL that is labeled as fraudulent by the classifier is manually labeled by three subject matter experts. The participants did not have any information about the classifier's predicted label. Comparing BEYOND PHISH's predictions and the experts' assigned labels indicates a false positive rate of 1.21%. To understand the impact of Alexa ranking, we repeat this experiment for the bottom $10,000$ websites from the $1M$ Alexa websites. Comparing BEYOND PHISH's prediction to the experts' assigned labels yields a false positive rate of 0.92%.

Palo Alto Networks Data. In this experiment, we run BEYOND PHISH on a previously unseen dataset of 13,982 legitimate and 10,054 FCWs provided by Palo Alto Networks. Table IV indicates statistics of this dataset. Palo Alto Networks collects real-world FCWs and its security researchers verify the labels of collected data manually. Testing our model on the Palo Alto Networks data indicates a false positive rate of 2.46% and a high detection rate of 94.88%. The model's performance on these unseen samples validates our collected dataset, the generalizability of designed features, and the ability of BEYOND PHISH to identify FCWs in a realistic setting.

Major Financial Organization Data. This experiment focuses on real-world validation, where we compare our classifier against expert curation of scams by a financial organization. First, we collect domains submitted to Scam Advisor over the course of two weeks. Then, we collaborate with a major financial organization, which identified 2,229 websites that it believed to be fraudulent based on suspicious transaction records. BP+NN flagged 1,879 of them as FCWs. This result indicates that our system agrees on FCWs for a significant number of the reported websites (note that the human experts only identify scams; they do not deem websites as legitimate). However, comparing the results on the Reddit dataset and the major financial organization dataset shows a decrease in the model's performance. We believe several factors contribute to this regression. First, the presence of label noise in the data can affect the model's accuracy. While the Palo Alto Networks and Reddit datasets we collected are manually labeled, we do not control our partners' data labeling. For instance, some websites may be labeled as scams by financial organizations because of usage in money laundering, which we do not consider as FCW. Second, our partner's dataset is 10 months newer than the training dataset, which may have resulted in a data shift

TABLE VIII: Feature analysis of BEYOND PHISH considering **legitimate** and **fraudulent** as target label.

	Legitimate		Fraudulent	
Rank	**Feature**	**Score**	**Feature**	**Score**
1	Valid social media	10.80	Valid social media	10.80
2	# of likes	10.61	Account age	7.43
3	Account age	7.44	Sub-domain level	5.69
4	# of followers	7.15	Registration period	5.38
5	Sub-domain level	5.69	Domain age	4.69
6	Registration period	5.43	Top 100K	4.05
7	Domain age	4.66	Host country	4.01
8	Top 100K	4.12	# external links	3.96
9	Host country	4.02	Domain country	3.78
10	# external links	4.00	# script tags	3.66
11	Domain country	3.78	Cheap TLD	3.64
12	# script tags	3.66	Domain privacy	3.61
13	Cheap TLD	3.63	# of followers	3.61
14	Domain privacy	3.60	# of Facebook page likes	3.60
15	Same host & domain country	3.60	Includes hyphen	3.60
16	Includes hyphen	3.59	Same host & domain country	3.60
17	Cheap registrar	3.56	Cheap registrar	3.56
18	Includes digits	3.56	Includes digits	3.56

in FCW trends that our model does not capture. Ultimately, we believe that the active monitoring of FCWs by security experts at major organizations demonstrates that FCWs are an important problem for the broader industry to tackle.

E. Analysis of Features

Understanding BEYOND PHISH's prediction based on a given input is important as it enables us to explain which features are inherent to FCWs. In this section, we perform feature analysis to find the importance of every feature in the BP+NN model. We also conducted an ablation study to analyze the effect of every feature category provided in Section F.

Neural networks act as black boxes when it comes to interpretability. Various methods have been proposed to help interpret the predictions of neural network models [61, 66]. SHAP [61] is a unified framework for interpreting a model's predictions. For a specific prediction, SHAP assigns each feature an importance value, known as Shapley values. Considering each feature as a variable, Shapley values measure the impact of each variable taking into account the interaction with other variables. SHAP computes these values based on a comparison of what a model predicts *with* and *without* the feature.

We use DeepSHAP [61], a variant of the SHAP framework that uses back-propagation values in a neural network to find important features. We use a baseline distribution consisting of $1,000$ random valid samples. Table VIII shows the score and rank of each feature in our model considering the target label as legitimate and fraudulent, respectively. The scores are normalized absolute Shapley values. Valid social media, cheap TLD, and the number of external links are among the most important detection features. Except for these three features, we can see different features are important to assign each label. For example, domain age is more important for assigning the fraudulent label.

Neural networks, due to their non-linear property, consider different features when classifying different inputs. Hence, for each input data we may encounter different feature importance values as illustrated in Table IX. Table IX indicates the most important features for detecting five random samples as both legitimate and fraudulent. Furthermore, to indicate the importance of features in general, we calculate the importance of each feature by averaging all input data Shapley values.

TABLE IX: Each row represents the most important feature of a sample considering both legitimate and fraudulent as the target label. This table shows that BEYOND PHISH is non-linear and considers different features for classifying different inputs.

Rank	**Legitimate as target**	**Fraudulent as target**
1	# of followers	# of followers
2	Valid social media	Valid social media
3	Social media age	Social media age
4	Domain age	Sub-domain level
5	Sub-domain level	Top 100K

F. Model Robustness

Similar to typical machine learning-based detection systems, attackers may attempt to evade BEYOND PHISH after it is trained. In this section, we show that trying to evade BEYOND PHISH will alter the economics for miscreants to create a legitimate-looking fraudulent website and consequently harm their profitability.

We consider the same feature categories as in Section V-A and, from an attacker's point-of-view, we try to alter features in every category to evade BEYOND PHISH. Our proposed neural network-based model is a non-linear detection model and the effects of changing a feature cannot be studied directly. In other words, we cannot simply change the value of a feature without considering its dependencies. For example, changing the *valid social media* feature affects all the features in the social media-based category. To make the robustness analysis feasible, we do not consider the effects of features on each other. To calculate the effectiveness of every attack, we use *success rate* metric, which is the percentage of FCWs that successfully bypass the classifier. Except for the content-based features category, to evade the classifier, attackers must spend more money to change the feature values.

Content-based features: Content-based features are the easiest category for attackers to change because there is no cost in altering these features. However, changing the website's content by adding a valid social media link raises the risk of exposure through social media.

Three values $\{-1, 0, 1\}$ can be assigned to the *valid social media* feature. In the best-case scenario, where the attacker creates a valid social media link with a similar account name to the domain name, the success rate is 13%. We further evaluate the performance of BEYOND PHISH by excluding the *valid social media* feature in Figure 7, named BP+NN\Social Media. According to our dataset, 79.08% of the FCWs do not include a social media link because maintaining social media pages is costly, time-consuming, and it increases the risk of being exposed as fraudulent by social media users and bots. As

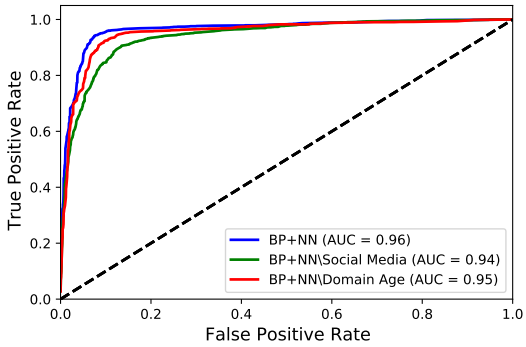


Fig. 7: Comparing BEYOND PHISH with its variants where one of the features is removed.

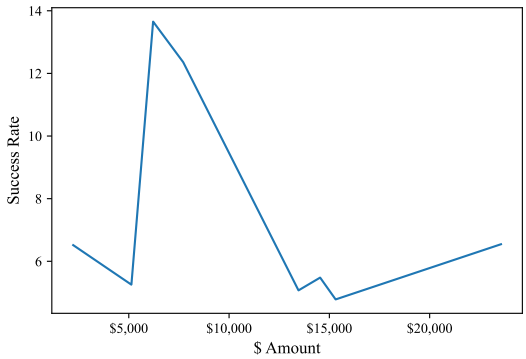


Fig. 8: Success rate of using an aged domain per amount cost.

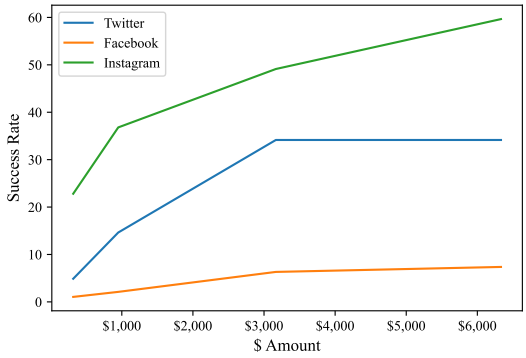


Fig. 9: The amount of money that miscreants should spend to bypass the classifier.

shown in Figure 7, using this feature increases the performance of BEYOND PHISH by 3% in AUC.

DNS-based features: Altering DNS-based features requires more effort from attackers compared to altering content-based features. As shown in Table VIII, DNS-based features such as domain age play an important role in detecting FCWs. To analyze the robustness of DNS-based features, we consider *domain age*, *cheap registrar*, *registration period*, and *private WHOIS* features.

For the *domain age* feature, we modified the creation date of FCWs to vary from 1995–2021 to see how it affects the decision of the classifier. To estimate the cost of aged domains we use data from expireddomains.net. Figure 8 shows the success rate of using an aged domain for different prices. According to the prices of aged domains, we conclude that attackers need to spend at least $\$1,000$ to buy an aged domain that may bypass BEYOND PHISH. Using an aged domain affects other features such as domain name, country, and URL-based features. We believe the aforementioned features are the cause of non-linearity in Figure 8. Although domain age plays an important role in detecting FCWs, we train a variant of BEYOND PHISH without considering the *domain age* feature named BP+NN\Domain Age, shown in Figure 7, which indicates negligible performance loss. Altering the *cheap registrar*, *registration period*, and *private WHOIS* have similar results: attackers can bypass the classifier by changing these features with a success rate of 1.21%, 1.20%, and 2.39%, respectively.

URL-based features: Attackers can alter all of the features in this category by choosing a domain name that changes the feature values. Among these features we consider *cheap TLD* and *sub-domain level*. We change the *cheap TLD* feature in FCWs to analyze the change in BEYOND PHISH's output. The success rate for using a non-cheap domain is only 0.34%, which is negligible. Next, we perturb the *sub-domain level* feature by changing it within range 1–4. We observe that 1.86% FCWs can bypass the classifier by using only one sub-domain.

Social Media-based features: Although most FCWs do not have a valid social media link, analyzing this feature helps us to broaden our outlook on how fraudulent websites operate their social media. To analyze this feature, we use FCWs having a valid social media link. Then, by perturbing the social media-based features, we study how attackers can bypass the classifier. Figure 9 indicates the trade-off between the ability to bypass the classifier, and the amount of money miscreants should spend. We used viralyft.com to estimate the cost of purchasing followers or likes. As the price for buying followers is different for various social media sites, we used the average cost. Figure 9 indicates the number of Instagram followers has the most impact on the decision of the classifier. However, to bypass the decision of the classifier, miscreants must spend at least $\$280$ to purchase the required amount of followers.

VII. DISCUSSION

In this section, we discuss possible deployment scenarios, as well as some limitations of our work.

Deployment Scenarios. Our classification approach can be used to protect users from FCWs. Our experimental results in Section VI demonstrate that current blocklists cannot detect FCWs. Thus, the proposed approach could be used as a complementary system alongside current blocklists such as GSB or Microsoft Windows Defender. This way, not only will users be protected against phishing and malware, but they can also be warned about possible FCWs.

According to the Whois Domain Search, around $100,000$ domains are registered daily, and some of these will result in FCWs. Another possible deployment scenario is for domain

registrars or website building platforms, such as Shopify, to use the proposed approach as a screening method to scan newly created websites and take action against possible FCWs.

Limitations. The limitations of our work include subtleties in our data collection and the detection methods.

Data-related limitations. Data collection is the first step in analyzing and classifying FCWs. Despite previous research in this area [10, 67], there is a lack of robust and relevant datasets. Most previous studies crawl URLs from spam emails to find potential fraudulent URLs. Although we overcome this problem by using Reddit, there are still some limitations. Despite the large amount of collected URLs, we had limited URLs to train and evaluate our model, due to the short lifetime of FCWs.

One of the fundamental challenges in FCWs classification is the lack of ground truth data. To overcome this problem, we use sentiment analysis on feedback related to each URL. Our experiment in Section IV-C illustrates an error of 4.88%. We automated the labeling process to increase efficiency. However, error in both feedback and classifier's performance may skew the data, and manually labeled data by experts may suffer from the same issue. Moreover, as there might be FCWs that humans are not very good at detecting, these will not be posted on /r/Scams. Therefore, collecting URLs from /r/Scams can be biased towards FCWs that raise human's suspicion.

Finding meaningful features is challenging in the field of FCWs detection. The feature extraction component in our proposed method can be improved by collaborating with organizations that have additional data sources, website traffic, or registered email reputation.

Our training dataset was collected from /r/Scams over 2 years in 2018–2021, and the validation dataset from /r/Scams was collected over 14 months in 2020–2021. Despite having an older training dataset, our experiments on never-before-seen samples over time demonstrate the robustness of the features and model. BEYOND PHISH achieves 98.38% accuracy on potential FCWs submitted to /r/Scams, and 83.41% accuracy on the August 2022 dataset from a financial company. We believe that these results show that currently there is little "concept drift" in FCWs, and perhaps this indicates that miscreants do not need to significantly alter their techniques due to poor ecosystem level defenses (as shown in Section VI-B).

Detection method limitations. Content-based features are important in detecting FCWs. However, as many newly registered domains do not have content, it is difficult to classify them at an *early stage*. To quickly determine domain reputation, Hao et al. [14] considers only URL and DNS-based features. Although their method shows promising results, our experiments in Section VI-E indicate the significance of content-based features. Moreover, neglecting content-based features can lead to classifier bias against certain registrars [68].

VIII. RELATED WORK

One track of research on detecting malicious websites uses features obtained from URL properties. The motivation behind these methods is that miscreants try to create URLs similar to well-known brands or names. Kolari et al. [30] were the first to extract words from URLs to detect malicious blogs. They proposed a method that considers URL's tokens to extract features. The features are then passed to an SVM classifier to detect malicious blogs [30]. Improving upon this, Garera et al. proposed a method that considers URL, domain and path features to analyze URLs and detect phishing websites [29]. Also, URLs can be used to extract host-based features [31, 32]. Ma et al. proposed a phishing detection approach by including host-based features such as IP address, WHOIS, domain name, and geographic properties [31]. A similar work uses a different set of host-based features to detect spam URLs [32]. Xu et al. defined malicious websites as websites that can cause download and execution of malware in browsers. They extracted application-layer and network-layer features from packets to detect malicious websites [69].

Other approaches used content-based features which can be obtained from the HTML and JavaScript webpage source. Compared to URL and host-based features, content-based features are considered heavy-weight. However, they can provide information and thus create better prediction models [56]. Choi et al. proposed a method for detecting malicious web pages by considering DNS, webpage content, domain name, and network features [16]. They used simple content-based features such as HTML tag count, iframe count, line count, and hyperlink count. Another approach, Delta, was proposed by Borgolte et al. where they considered change-related features between two versions of the same website [17]. Change-related features are considered as the difference between the Document Object Models (DOM) trees of a website with its base version. This detection method is built to check if the change of the website is malicious. A recent approach [11] based on the source code of the website, applies a TF-IDF vectorizer to classify the fake shops. As our experiments indicates, this method achieves a low false positive and detection rate. The low detection rate reveals the inadequacy of using only content-based features.

In the domain of detecting fraudulent e-commerce websites (FCWs), Carpineto et al. proposed a method to detect counterfeit brand shopping websites [7]. Fist, they gathered a dataset using search queries on well-known brands. Then, they extracted features to build an SVM classifier capable of detecting fraudulent shopping websites. The features include discount rates on displayed products, brands of items, and if the webpage URL's path contains a brand name. Mostard et al. combined both visual and contextual features to detect FCWs [12]. The contextual features are similar to ones that Carpineto et al. used [7]. Kharraz et al. proposed Surveylance to detect malicious survey websites by using both visual and content-based features [15]. Similarly, Srinivasan et al. proposed a method based on DNS and content-based features to detect scam technical support websites [33]. Following the detection of malicious surveys and technical support websites, Vadrevu et al. proposed a method for tracking social engineering attack campaigns based on their online advertisements [70]. Hao et al. proposed an approach called PREDATOR to classify

malicious websites at the time of domain registration [14]. PREDATOR considers URL and host-based features such as the registration history, domain name length and weekday of registration to detect malicious websites at an early stage. However, they did not consider content-based features which play an important role in detecting FCWs.

Relying heavily on content-based features can harm the classifier's performance because such features are easily controllable by miscreants. For example, [12] considers features such as the presence of an address, copyright text, shopping cart, and phone number. For detecting FCWs, our approach differs in the considered features as we use new features (along with URL, host, content, and social media based features) related to our problem domain and omit deprecated features. Previous research used features such as iframe count, amount of items' discount and number of items [7, 55]. According to our observations, these features are no longer present in modern FCWs and are specific to counterfeit shopping websites only. We used generic features that are not time-sensitive and can be used for different kinds of FCWs. Moreover, our work differs from others in terms of the targeted domain and dataset. Several previous studies while claiming to work on FCWs, used spam datasets [31, 32] such as spamscatter that contains different kinds of malicious websites [13]. Due to advances in spam detection methods, it is very difficult to successfully advertise FCWs through email [71]. Our observations in Section IV-E indicate that such websites use other techniques (e.g., social media advertisements) than spam emails. Therefore, we leveraged social media to curate a dataset of FCWs.

IX. Conclusion

Miscreants take advantage of users' increased reliance on online services to defraud them. Despite advances in anti-phishing systems, current mitigation methods cannot effectively protect users against fraudulent e-commerce websites as scam websites mimic the behavior and user experience of legitimate websites. We created an automated approach to detecting FCWs using a set of well-defined features based on a crowd-sourced dataset. Our experiments indicated the insufficiency of current mitigation systems and illustrated the promising performance of the BEYOND PHISH classifier for detecting FCWs in the wild. We consider the collected data and designed classifier as stepping stones to further research in understanding the nature of, and mitigating, FCWs at scale. We release the dataset, models, and implemented baselines upon publication.

X. Ethics Statement

Although all data used in this study and the source code of the model will become publicly available, we are committed to preventing damage to new businesses by the Reddit fraudulent e-commerce websites detection bot (the study design was approved by our institution's IRB). To this end, we review the bot's decisions daily and provided several contacts for users to report legitimate websites that may have been affected by the Reddit bot's decision. During this study, we received only one such report. We immediately responded to the report, investigated the site in question and resolved the issue by removing the bot's response. Moreover, we have reported all collected fraudulent e-commerce websites to the relevant organizations.

Acknowledgements

We thank the anonymous reviewers for their valuable feedback. Our appreciation also extends to the two industry organizations for their insightful contributions and collaboration. This work was supported in part by the Defense Advanced Research Projects Agency (DARPA) CHESS (No. FA8750-19C-0003), the NSF grant 2000792, the Korea Internet & Security Agency (KISA) grant funded by the Personal Information Protection Commission (PIPC) (No. 1781000003), and the Department of Defense. We gratefully acknowledge their support.

References

[1] A. Oest, Y. Safei, A. Doupé, G.-J. Ahn, B. Wardman, and G. Warner, "Inside a phisher's mind: Understanding the anti-phishing ecosystem through phishing kit analysis," in *2018 APWG Symposium on Electronic Crime Research (eCrime)*. IEEE, 2018, pp. 1–12.

[2] G. Xiang, J. Hong, C. P. Rose, and L. Cranor, "Cantina+ a feature-rich machine learning framework for detecting phishing web sites," *ACM Transactions on Information and System Security (TISSEC)*, vol. 14, no. 2, pp. 1–28, 2011.

[3] C. Ardi and J. Heidemann, "Auntietuna: Personalized content-based phishing detection," in *NDSS Usable Security Workshop (USEC)*, 2016.

[4] T. K. Panum, K. Hageman, R. R. Hansen, and J. M. Pedersen, "Towards adversarial phishing detection," in *13th USENIX Workshop on Cyber Security Experimentation and Test (CSET 20)*, 2020.

[5] S. Marchal and N. Asokan, "On designing and evaluating phishing webpage detection techniques for the real world," in *11th USENIX Workshop on Cyber Security Experimentation and Test (CSET 18)*, 2018.

[6] A. Oest, Y. Safaei, P. Zhang, B. Wardman, K. Tyers, Y. Shoshitaishvili, and A. Doupé, "{PhishTime}: Continuous longitudinal measurement of the effectiveness of anti-phishing blacklists," in *29th USENIX Security Symposium (USENIX Security 20)*, 2020, pp. 379–396.

[7] C. Carpineto and G. Romano, "Learning to detect and measure fake ecommerce websites in search-engine results," in *Proceedings of the International Conference on Web Intelligence*, 2017, pp. 403–410.

[8] B. Price, "Resource Networks of Pet Scam Websites," in *2020 APWG Symposium on Electronic Crime Research (eCrime)*, 2020.

[9] M. Bitaab, H. Cho, A. Oest, P. Zhang, Z. Sun, R. Pourmohamad, D. Kim, T. Bao, R. Wang, Y. Shoshitaishvili, A. Doupé, and G.-J. Ahn, "Scam Pandemic: How Attackers Exploit Public Fear through Phishing," in *2020 APWG Symposium on Electronic Crime Research (eCrime)*, 2020.

[10] J. Wadleigh, J. Drew, and T. Moore, "The e-commerce market for" lemons" identification and analysis of websites selling counterfeit goods," in *Proceedings of the 24th International Conference on World Wide Web*, 2015, pp. 1188–1197.

[11] L. Beltzung, A. Lindley, O. Dinica, N. Hermann, and R. Lindner, "Real-time detection of fake-shops through machine learning," in *2020 IEEE International Conference on Big Data (Big Data)*. IEEE, 2020, pp. 2254–2263.

[12] W. Mostard, B. Zijlema, and M. Wiering, "Combining visual and contextual information for fraudulent online store classification," in *IEEE/WIC/ACM International Conference on Web Intelligence*, 2019, pp. 84–90.

[13] D. S. Anderson, C. Fleizach, S. Savage, and G. M. Voelker, "Spamscatter: Characterizing internet scam hosting infrastructure," Ph.D. dissertation, University of California, San Diego, 2007.

[14] S. Hao, A. Kantchelian, B. Miller, V. Paxson, and N. Feamster, "Predator: proactive recognition and elimination of domain abuse at time-of-registration," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 1568–1579.

[15] A. Kharraz, W. Robertson, and E. Kirda, "Surveylance: automatically detecting online survey scams," in *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2018, pp. 70–86.

[16] H. Choi, B. B. Zhu, and H. Lee, "Detecting malicious web links and identifying their attack types." *WebApps*, vol. 11, no. 11, p. 218, 2011.

[17] K. Borgolte, C. Kruegel, and G. Vigna, "Delta: automatic identification of unknown web-based infection campaigns," in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, 2013, pp. 109–120.

[18] F. T. Commission, "Consumer sentinel network data book2019," 2022, https://www.ftc.gov/reports/consumer-sentinel-network-data-book-2021.

[19] A. Oest, P. Zhang, B. Wardman, E. Nunes, J. Burgis, A. Zand, K. Thomas, A. Doupé, and G.-J. Ahn, "Sunrise to sunset: Analyzing the end-to-end life cycle and effectiveness of phishing attacks at scale," in *29th {USENIX} Security Symposium ({USENIX} Security 20)*, 2020.

[20] C. Ardi and J. Heidemann, "Precise detection of content reuse in the web," *ACM SIGCOMM Computer Communication Review*, vol. 49, no. 2, pp. 9–24, 2019.

[21] E. Alowaisheq, P. Wang, S. Alrwais, X. Liao, X. Wang, T. Alowaisheq, X. Mi, S. Tang, and B. Liu, "Cracking the wall of confinement: Understanding and analyzing malicious domain."

[22] C. Nykvist, L. Sjöström, J. Gustafsson, and N. Carlsson, "Server-side adoption of certificate transparency," in *International Conference on Passive and Active Network Measurement*. Springer, 2018, pp. 186–199.

[23] C. Whittaker, B. Ryner, and M. Nazif, "Large-scale automatic classification of phishing pages," 2010.

[24] Y. Zhang, J. I. Hong, and L. F. Cranor, "Cantina: a content-based approach to detecting phishing web sites," in *Proceedings of the 16th international conference on World Wide Web*, 2007, pp. 639–648.

[25] S. G. Stats, "Desktop vs mobile vs tablet market share worldwide," 2019, https://gs.statcounter.com/platform-market-share/desktop-mobile-tablet.

[26] A. Oest, Y. Safaei, A. Doupé, G.-J. Ahn, B. Wardman, and K. Tyers, "Phishfarm: A scalable framework for measuring the effectiveness of evasion techniques against browser phishing blacklists," in *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2019, pp. 1344–1361.

[27] E. Mnif, A. Jarboui, and K. Mouakhar, "How the cryptocurrency market has performed during covid 19? a multifractal analysis," *Finance research letters*, vol. 36, p. 101647, 2020.

[28] P. Xia, H. Wang, B. Zhang, R. Ji, B. Gao, L. Wu, X. Luo, and G. Xu, "Characterizing Cryptocurrency Exchange Scams," *Computers & Security*, vol. 98, p. 101993, 2020.

[29] S. Garera, N. Provos, M. Chew, and A. D. Rubin, "A framework for detection and measurement of phishing attacks," in *Proceedings of the 2007 ACM workshop on Recurring malcode*, 2007, pp. 1–8.

[30] P. Kolari, T. Finin, A. Joshi *et al.*, "Svms for the blogosphere: Blog identification and splog detection," in *AAAI spring symposium on computational approaches to analysing weblogs*, 2006.

[31] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond blacklists: learning to detect malicious web sites from suspicious urls," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009, pp. 1245–1254.

[32] ——, "Identifying suspicious urls: an application of large-scale online learning," in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 681–688.

[33] B. Srinivasan, A. Kountouras, N. Miramirkhani, M. Alam, N. Nikiforakis, M. Antonakakis, and M. Ahamad, "Exposing search and advertisement abuse tactics and infrastructure of technical support scammers," in *Proceedings of the 2018 World Wide Web Conference*, 2018, pp. 319–328.

[34] "Spamhaus," https://www.spamhaus.org/.

[35] "PhishTank," http://phishtank.org/index.php.

[36] "OpenPhish," https://openphish.com/.

[37] "VirusTotal," https://www.virustotal.com/.

[38] P. Peng, L. Yang, L. Song, and G. Wang, "Opening the blackbox of virustotal: Analyzing online phishing scan engines," in *Proceedings of the Internet Measurement Conference*, 2019, pp. 478–485.

[39] N. Kheir, F. Tran, P. Caron, and N. Deschamps, "Mentor: positive dns reputation to skim-off benign domains in botnet c&c blacklists," in *IFIP International Information Security Conference*. Springer, 2014, pp. 1–14.

[40] S. Sinha, M. Bailey, and F. Jahanian, "Shades of grey: On the effectiveness of reputation-based "blacklists"," in *2008 3rd International Conference on Malicious and Unwanted Software (MALWARE)*. IEEE, 2008, pp. 57–64.

[41] A. Ramachandran, D. Dagon, and N. Feamster, "Can dns-based blacklists keep up with bots?" in *CEAS*. Citeseer, 2006.

[42] Y. Zhauniarovich, I. Khalil, T. Yu, and M. Dacier, "A survey on malicious domains detection through dns data analysis," *ACM Computing Surveys (CSUR)*, vol. 51, no. 4, pp. 1–36, 2018.

[43] F. Team, "Reddit Statistics For 2021 (Demographics, Usage & Traffic Data)," 2021, https://foundationinc.co/lab/reddit-statistics/.

[44] J. Baumgartner, S. Zannettou, B. Keegan, M. Squire, and J. Blackburn, "The pushshift reddit dataset," in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 14, 2020, pp. 830–839.

[45] "ICANN," 2019, https://www.icann.org/resources/pages/governance/bylaws-en.

[46] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[47] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 conference on empirical methods in natural language processing*, 2013, pp. 1631–1642.

[48] C. M. Bishop, "Training with noise is equivalent to tikhonov regularization," *Neural computation*, vol. 7, no. 1, pp. 108–116, 1995.

[49] D. Rolnick, A. Veit, S. Belongie, and N. Shavit, "Deep learning is robust to massive label noise," *arXiv preprint arXiv:1705.10694*, 2017.

[50] H. Song, M. Kim, D. Park, and J.-G. Lee, "Learning from noisy labels with deep neural networks: A survey," *arXiv preprint arXiv:2007.08199*, 2020.

[51] B. B. Bureau, "bbb scam alert: Cheating on homework leads to extortion scam," 2021, https://www.bbb.org/article/news-releases/24032-bbb-scam-alert-students-hire-homework-help-and-end-up-in-extortion-con.

[52] R. W. Naylor, C. P. Lamberton, and P. M. West, "Beyond the like button: The impact of mere virtual presence on brand evaluations and purchase intentions in social media settings," *Journal of Marketing*, vol. 76, no. 6, pp. 105–120, 2012.

[53] H. Liu, K. Levchenko, M. Félegyházi, C. Kreibich, G. Maier, G. M. Voelker, and S. Savage, "On the effects of registrar-level intervention." in *LEET*, 2011.

[54] B. News, "Shopify stores riddled with fakes and fraudsters," 2020, https://www.bbc.com/news/business-55420445.

[55] D. Canali, M. Cova, G. Vigna, and C. Kruegel, "Prophiler: a fast filter for the large-scale detection of malicious web pages," in *Proceedings of the 20th international conference on World wide web*, 2011, pp. 197–206.

[56] D. Sahoo, C. Liu, and S. C. Hoi, "Malicious url detection using machine learning: A survey," *arXiv preprint arXiv:1701.07179*, 2017.

[57] R. Clayton and T. Mansfield, "A study of whois privacy and proxy service abuse," in *13th Workshop on the Economics of Information Security*, 2014.

[58] T. M. Oshiro, P. S. Perez, and J. A. Baranauskas, "How many trees in a random forest?" in *International workshop on machine learning and data mining in pattern recognition*. Springer, 2012, pp. 154–168.

[59] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[60] L. Wang, *Support vector machines: theory and applications*. Springer Science & Business Media, 2005, vol. 177.

[61] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in neural information processing systems*, 2017, pp. 4765–4774.

[62] "APWG," 2020, https://apwg.org/.

[63] "CheckPhish," 2020, https://checkphish.ai/.

[64] J. Saxe, R. Harang, C. Wild, and H. Sanders, "A deep learning approach to fast, format-agnostic detection of malicious web content," in *2018 IEEE Security and Privacy Workshops (SPW)*. IEEE, 2018, pp. 8–14.

[65] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, 2016, pp. 1480–1489.

[66] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," *arXiv preprint arXiv:1704.02685*, 2017.

[67] N. Christin, "Traveling the silk road: A measurement analysis of a large anonymous online marketplace," in *Proceedings of the 22nd international conference on World Wide Web*, 2013, pp. 213–224.
[68] J. Saxe and K. Berlin, "expose: A character-level convolutional neural network with embeddings for detecting malicious urls, file paths and registry keys," *arXiv preprint arXiv:1702.08568*, 2017.
[69] L. Xu, Z. Zhan, S. Xu, and K. Ye, "Cross-layer detection of malicious websites," in *Proceedings of the third ACM conference on Data and application security and privacy*, 2013, pp. 141–152.
[70] P. Vadrevu and R. Perdisci, "What you see is not what you get: Discovering and tracking social engineering attack campaigns," in *Proceedings of the Internet Measurement Conference*, 2019, pp. 308–321.
[71] T. A. Almeida and A. Yamakami, "Advances in spam filtering techniques," in *Computational Intelligence for Privacy and Security*. Springer, 2012, pp. 199–214.
[72] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[73] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

APPENDIX

A. *Details of Trained Models*

The compared approaches include using Random Forest, SVM, and XGBoost. We detail the parameter settings for each of these approaches in Table X.

TABLE X: Parameter details of baselines.

Model	Parameter	Value
SVM	kernel	rbf
	gamma	scale
RF	max_depth	10
	min_samples_split	2
	n_estimators	100
XGBoost	objective	binary:logistic
	eta	1
	n_estimators	100
	max_depth	10

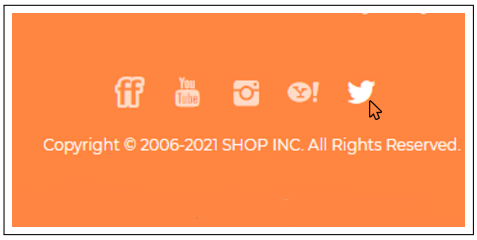
As mentioned in Section V-B, the designed neural network classifier consists of six layers. The input x is a feature vector that will be passed through the network to output the probability of the input being an FCW:

$$\begin{aligned} \mathbf{o_1} &= \tanh(\mathbf{W}^{(1)}\mathbf{x} + \mathbf{b}^{(1)}) \\ \mathbf{o_2} &= \tanh(\mathbf{W}^{(2)}\mathbf{o_1} + \mathbf{b}^{(2)}) \\ \mathbf{o_3} &= \tanh(\mathbf{W}^{(3)}\mathbf{o_2} + \mathbf{b}^{(3)}) \\ p &= \text{sigmoid}(\mathbf{W}^{(3)}\mathbf{o_3} + \mathbf{b}^{(3)}) \end{aligned} \tag{1}$$

where $\{\mathbf{o}^{(i)}\}_{i=0}^{3}$ is the output of each layer, and $\{\mathbf{W}^{(i)}, \mathbf{b}^{(i)}\}_{i=0}^{3}$ are learnable network parameters.

For this task, we use the weighted cross-entropy loss function during training. We use $w_{legitimate} = 0.8$ and $w_{scam} = 0.2$ for legitimate and FCW samples, respectively. This weight assignment penalizes the model for mis-classifying the legitimate samples more than mis-classifying the FCW samples.

Given the formulation of the MLP classifier $\mathcal{F}$, we aim to find the optimal network parameters $\{\mathbf{W}^{(i)}, \mathbf{b}^{(i)}\}_{i=0}^{4}$. To this end, we use Adam optimizer [72] to minimize the loss function L and optimize the network parameters. Moreover, we use



(a) FCW without any social media link.

(b) An invalid social media link.

Fig. 10: Examples of FCWs misusing social media icons.

Batch Normalization (BN) [73] which enables us to accelerate the learning process and solve the vanishing gradient problem when using the *sigmoid* activation function. BN is applied to each data batches $\mathcal{B} = \{x_1, x_2, ..., x_b\}$ with size b during the training process. It transforms $\mathcal{B}$ to a new data batch $\mathcal{B}' = \{x'_1, x'_2, ..., x'_b\}$ as $(x' = x_i - \mu_\mathcal{B})/\sqrt{\sigma^2_\mathcal{B} + \epsilon}$, where $\mu_\mathcal{B} = \frac{1}{m}\sum_{i=1}^{b} x_i$ indicates the batch mean, $\sigma^2_\mathcal{B} = \frac{1}{m}\sum_{i=1}^{b}(x_i - \mu_\mathcal{B})^2$ is the batch variance, and ϵ is a constant small number added to the batch variance for numerical stability.

We use the popular machine learning framework PyTorch to implement the model. Each hidden layer of the MLP classifier has $\{2048, 1024, 512, 256\}$ neurons, respectively. During training, a batch size of 32 is used to sample data from the training set. The batch is passed through the classifier to output the probability, thus, calculating the loss value using **??**. Then, the MLP classifier is updated using Adam optimizer with a learning rate of 0.0001. During the testing phase, we consider the output label as fraudulent if the output probability is greater than 0.5.

B. *Example of Social Media Link*

Figure 10 shows two different examples of social media icons' misuse in FCWs.

C. *Categorizing Websites Source Codes*

We used the following terms to identify websites in each category:

- **Online Shopping:** cart, shop, shoe, bag, ps4, ps5, xbox, nintendo, game, sale, and discount.
- **Education Related Websites:** course, education, essay, study, school, assignment, paper, and tutor.
- **Adult Content:** porn, pornography, sex, nude, and xxx.

TABLE XI: An example of our labeling process on a Reddit submission.

Comment	Label	Probability
Common scam, ignore and move on. Also, use an ad blocker and ignore things you see on Facebook.	NEGATIVE	0.99
Don't do this :)	NEGATIVE	0.99
My mother uses Facebook...	NEGATIVE	0.91
I bought this product, it is very good. Good quality. I think it's not scam as you say.	POSITIVE	0.99
URL Label	NEGATIVE	

- **Pet Scam:** cat, dog, pet, breed, kitten, puppy, and puppies.
- **Delivery Websites:** shipment, tracking, cargo, and delivery.
- **Charity Websites:** charity, donation, donate, hungry, and hunger.
- **Cryptocurrency and Stock Market:** bitcoin, crypt, BTC, and blockchain, stock, trade, and trading.
- **Business Related Websites:** credit, own, home, business, and finance.

D. *Example of Labeling Process*

We present an example of our labeling process on a Reddit post in Table XI. In this example, we first assign the label of every comment using BERT, then we aggregate the results as in Section IV-B.

E. *Manual Labeling Process*

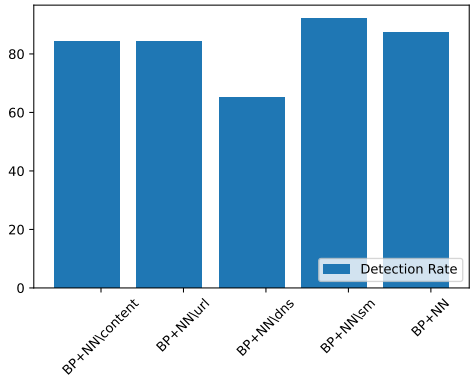
To classify the 2,000 websites, we enlisted the help of three security experts. During the labeling process, these experts were not allowed to discuss their evaluations with each other. Instead, each expert independently investigated the features of each website and evaluated relevant Reddit threads and conducted web searches to determine the website's reputation. If an expert could not make a conclusive decision, they labeled the website as "unknown." The final label for each website was determined using a majority vote among the three experts. Any samples with a majority of "unknown" labels were removed from our dataset.

F. *Feature Category Importance*

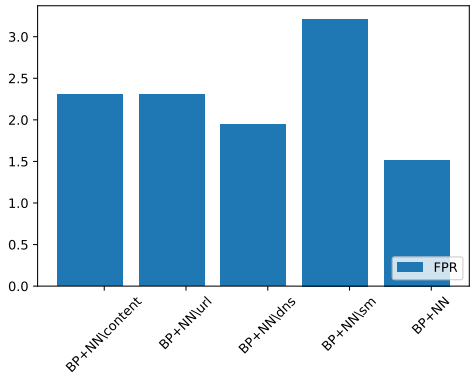
We perform an ablation study to show the importance of each features category. Table III shows the category of each feature. In this experiment, we design four variants of our model that excludes one feature category out during the training and testing process:

- **BP+NN\content**: removing content-based features.
- **BP+NN\url**: removing URL-based features.
- **BP+NN\dns**: removing DNS-based features.
- **BP+NN\sm**: removing social media-based features.

Figure 11 shows the false positive rate and true positive rate (i.e., detection rate) of the BP+NN model on exclusion of each feature category. The results indicate that using all features we can get better performance in comparison to the baselines. Although removing social media-based features increases the detection rate by a small margin, it increases the false positive rate by a large margin, making this category essential to having a model with low false positive rate.



(a) Detection rate of BP+NN



(b) False positive rate of BP+NN

Fig. 11: Ablation study on excluding every feature category. BP+NN\C indicates we have excluded all features from category C during training and testing process.

G. *ECDF of Features*

We plot the Empirical Commutative Distribution plot in Figure 12 to show the difference of the designed features between legitimate and fraudulent websites. While most of the features have a meaningful difference, some of them have a similar distribution such as including hypen and digits. However, some classifiers such as neural network-based classifiers combine features to better distinguish between labels. We believe that the combination of such features with other ones can create a fine-grained decision boundary.

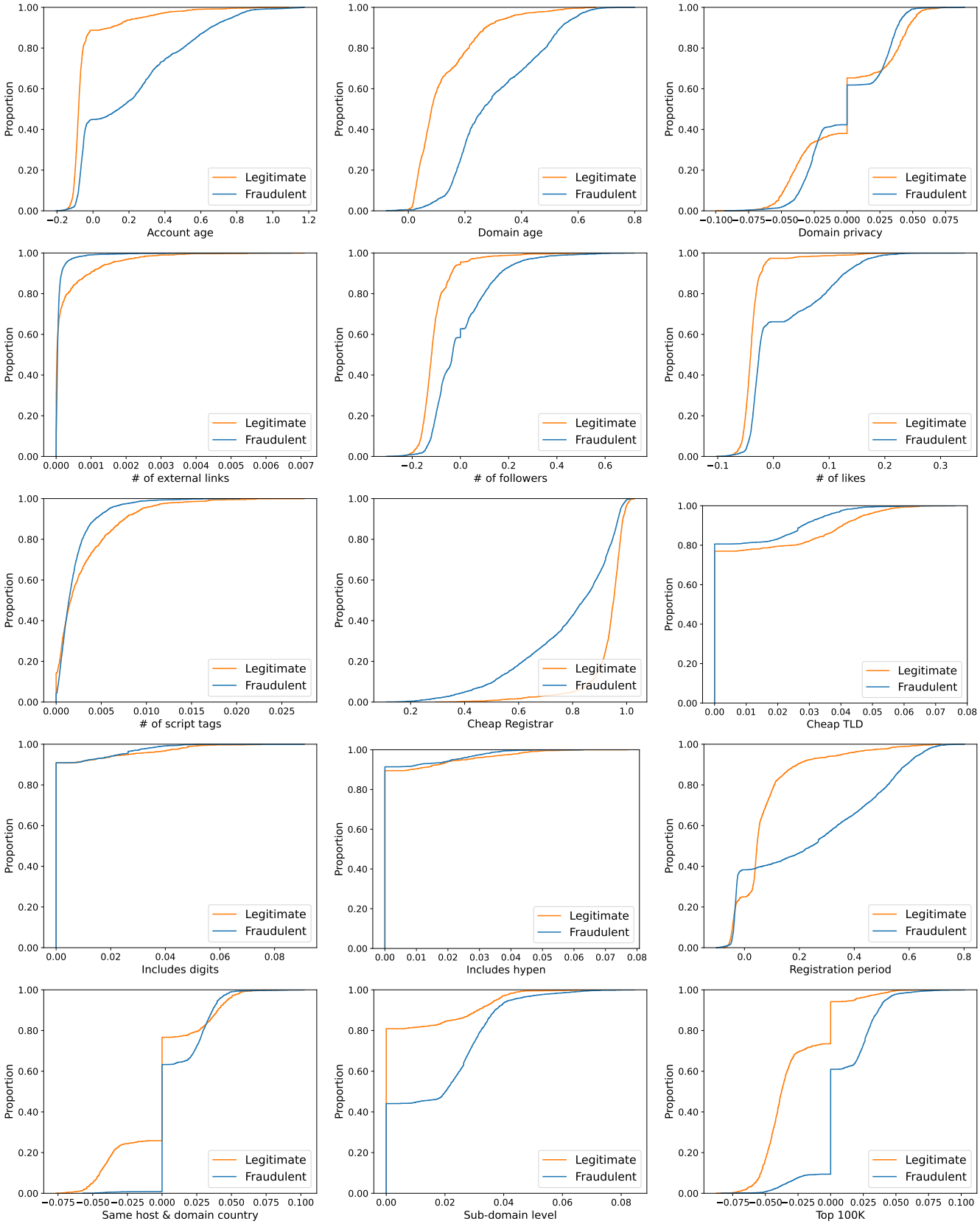


Fig. 12: ECDF plots of features showing distribution of each feature according to different labels.