TBTrackerX: Fantastic Trigger Bots and Where to Find Malicious Campaigns on X

Mohammad Majid Akhtar*, Rahat Masood*, Muhammad Ikram†, and Salil S. Kanhere*
*School of Computer Science and Engineering
University of New South Wales, Sydney, NSW 2052, Australia
Emails: {majid.akhtar, rahat.masood, salil.kanhere}@unsw.edu.au
†School of Computing, Macquarie University, Sydney, NSW 2109, Australia
Email: muhammad.ikram@mq.edu.au

***Abstract*—Malicious actors on online social networks (OSNs) use script-controlled social bots that engage users through replies or comments. These bots are programmed to activate only when specific trigger keywords appear in posts. We refer to such advanced context-aware campaigners as *trigger bot* (TB) agents, which aim to deceive users into making payments for illicit products or revealing sensitive financial credentials. This paper presents a systematic and data-driven study on the detection and characterization of TB agents. We introduce TBTrackerX, a novel framework designed to collect and analyze TB activity. Using this system, we captured 4,452 TB agent replies from 2,647 unique TB agents, targeting our honeytrap account, and uncovered interactions with over 84K users on 𝕏. Our results show that TB agents evade detection by using contextually similar replies (with similarity scores up to 0.97), exhibiting intermittent posting patterns (in bursts ranging from 15 seconds to 5 minutes), and adopting dormant behavior after peak campaign activity. Furthermore, we identify a coordinated TB ecosystem, characterized by fake TB followers and shared TB masters. This study underscores the pressing need for better moderation and detection mechanisms to combat these sophisticated forms of social media manipulation.**

I. Introduction

Online social networks (OSNs) have emerged as the predominant medium for communication, support, and influence. While their openness facilitates rapid information sharing, it also introduces systemic vulnerabilities. Adversaries exploit these through increasingly sophisticated social media manipulation (SMM) campaigns [1], often driven by automated agents, or social bots, whose tactics, techniques, and procedures have evolved over the past decade [2], [3].

Social bots are automated, script-controlled accounts that are designed to mimic real users and perform targeted tasks. They are now widespread across major OSN platforms, including 𝕏 (formerly Twitter) [4], [5], YouTube [3] and Facebook [6], [7]. Once simple and easy to detect, bots have grown more sophisticated, with advanced profile customization, nuanced content strategies, and coordinated behavior [1]. The emergence of large language models (LLMs) [8], [9], has further advanced bot capabilities, enabling real-time context-aware user engagement. Some bots now use LLMs to reply to tweets, simulate human-like conversations, and produce persuasive content that seamlessly blends into political, financial, or commercial discourse [10]–[13].

The growing sophistication of social bots blurs the line between human and automated behavior, driving reply-based malicious campaigns. On 𝕏, many such campaigns promoting cryptocurrency scams [2], [11], phishing links [14], and political propaganda [15] use bots to impersonate support agents or push fake giveaways, tricking users into revealing sensitive financial details. What makes these campaigns unique is their use of *trigger-based engagement*: bots remain dormant until a user's post contains specific keywords (e.g., "`metamask`", "`cashapp`", "`hacked`"). We refer to these as *trigger bots* (TBs): automated entities that monitor posts in real time and respond selectively when trigger terms appear (see §II-A). While many TBs are malicious, others use similar tactics for legitimate purposes (e.g., freelance support), making it essential to study the full spectrum of TB behavior, from *benign* to *malicious*.

The risk deepens when TBs coordinate and use semantic variation to evade detection. As shown in Figure 1, multiple TB agents often coordinate to target the same victim with semantically similar, but not identical replies, circumventing duplicate content filters. Prior work [2], [11] has shown that TB agents operating deceptive support scams have collectively stolen at least 38.40 BTC. The scale, sophistication, and impact of these operations underscore the need for a measurement-driven investigation of the TB ecosystem across multiple campaigns.

Despite causing significant harm, such as confirmed cryptocurrency theft [2], the detection of TB agents remains largely reactive and insufficient. This study addresses that gap by conducting the first large-scale, empirical analysis of TB agents, examining their behavior, campaign structures, and detection strategies. To guide our study, we investigate the following research questions:

RQ1: TB ecosystem discovery. *What observable components within the TB agent ecosystem enable their detection and discovery?*

RQ2: Profile and behavioral characteristics of TBs. *How*

Network and Distributed System Security (NDSS) Symposium 2026
23-27 February 2026, San Diego, CA, USA
ISBN 979-8-9919276-8-0
https://dx.doi.org/10.14722/ndss.2026.241239
www.ndss-symposium.org

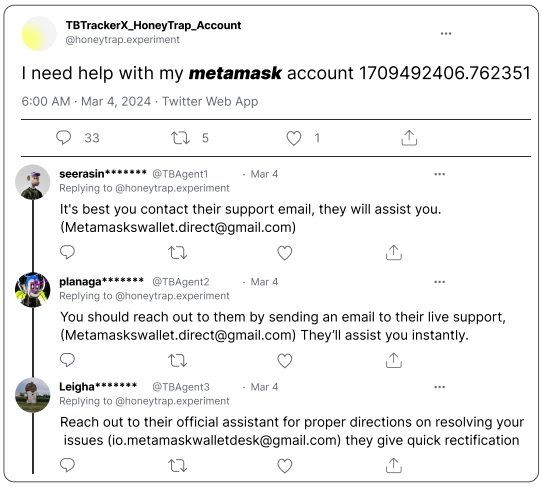

Fig. 1: Example of Trigger Bot (TB) agents replying to trigger keyword (in bold) in Tweet on 𝕏. While appear benign, their semantically similar replies are crafted to lure users to *malicious* campaigns.

do the actions and characteristics of malicious TB agents differ from benign ones?

RQ3: Platform intervention and longitudinal evaluation. *How successful is 𝕏 in identifying and suspending malicious TBs? What longitudinal patterns explain why some TB agents remain active despite platform enforcements?*

RQ4: Detection strategies and stakeholder recommendations. *Which detection methods are most effective? What practical recommendations can be provided to stakeholders?*

To address these questions, we developed TBTrackerX, a honeytrap-driven measurement and detection framework that uses *trigger keywords* to attract TB agents. From March 1–30, 2024, we deployed TBTrackerX using a controlled 𝕏 account and captured over 4,452 TB replies from 2,647 distinct TB agents across 10 trigger keywords (see §III-A). We analyzed both malicious and benign TB agents across four dimensions: profile characteristics, textual content, temporal patterns, and social network structure. Our results show that while 57% - 67% of malicious TB agents were suspended by 𝕏, significant gaps remain in the platform's mitigation efforts. This highlights the importance of a deeper understanding of malicious bot behavior and improved detection of SMM.

Our findings are summarized below as key contributions.

1) **Measurement of TB agents.** We present the first large-scale, empirical study of TB agents, a novel and growing class of reply-based, script-controlled, and campaign-driven social bots on 𝕏. To uncover these agents, we develop TBTrackerX and introduce a novel honeytrap methodology (cf. §III) to identify reply-based TB agents. Our deployment identified 2,647 TB agents involved in deceptive financial scams, illicit products/services, and misleading romance schemes.
2) **Multi-dimensional TB agent dataset.** We compile a comprehensive dataset (cf. §III-C) of TB agents, including their replies, tweet history, metadata, and network footprints. For each agent, we collect 𝕏 profile metadata, the latest 100 tweets and replies, and a list of followers and friends, enabling detailed analysis of coordinated bot campaigns.
3) **Ecosystem and strategy analysis of TB agents.** We conduct multi-dimensional analysis of TB agents from four vantage points: profile characteristics, reply content, temporal patterns, and network structures, to address our research questions. This includes quantifying suspension rates (**RQ3**), identifying evasion strategies (**RQ4**), and uncovering key differences between malicious and benign TB agents (**RQ1**, **RQ2**), to inform effective detection strategies.
4) **Detection of TB agents.** To benchmark detection performance, we evaluate 15 diverse baselines across two key tasks: *malicious TB agent detection* and *campaign classification*. Additionally, we conduct an ablation study to evaluate feature importance and a generalizability study to assess the model's robustness against unseen TB agents. Our findings demonstrate that a classical XGBoost model achieves strong performance with F1-scores of 0.88 for malicious agent detection and 0.92 for campaign classification (cf. §V).

Outline. Our study reveals how TB agents' profile attributes change across campaigns (cf. §IV-A), modify content to evade detection (cf §IV-B), and exhibit irregular posting patterns (cf §IV-C). Our research reveals that TBs are not just isolated nuisances but part of a coordinated, evolving ecosystem that uses timing, content obfuscation, and campaign-level orchestration to evade detection at scale (cf §IV-D, §IV-E, §IV-F). By dissecting their behavioral and temporal fingerprints, we identify systemic gaps in current mitigation efforts. Our insights provide a data-driven foundation for strengthening automated defenses on OSNs (cf §V). Based on TBTrackerX's findings, we aim to advance research and guide stakeholders—including OSN platforms, end users, and the web security community to improve detection, transparency, and accountability in combating sophisticated SMM activities, as discussed in §VI-B.

II. Background and Related Work

This section provides the background and related work to contextualize our work within existing literature and highlight gaps our methodology aims to address.

A. Overview of TB Ecosystem Components

Definition 2.1 (Trigger Bots (TB)). *A TB is a social bot that interacts with and promotes a campaign to an OSN user based on specific (trigger) keywords used by the OSN user. It mimics humans and often engages in comments (in the form of replies) with other OSN users. Depending on their nature, campaigns can be classified as malicious or benign.*

A close analysis of the collected TB replies reveals an ecosystem composed of interconnected components. In particular, we observed that these TBs function as agents that actively promote specific campaigns to regular OSN users, aiming to direct them to the campaign masters behind the TB activity. TB agents sustain an extensive network of coordinated followers to enhance their credibility and operate covertly.

Interestingly, although TBs target different keywords, the behavior of both agents and their followers appears to be systematically organized and directly controlled by TB masters or TB farms (account marketplaces), as illustrated in Figure 2.

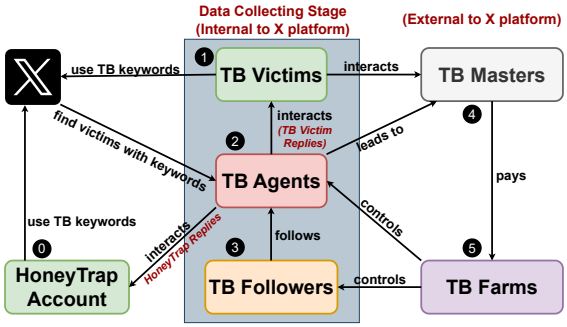

Fig. 2: TB ecosystem and components of TB operations.

Below, we outline each component of the TB ecosystem:

HoneyTrap Account. As the name suggests, we created a controlled $\mathbb{X}$ account to attract TB agents by periodically posting tweets containing trigger keywords. The account was active for one month, posting once every 12 hours to comply with the $\mathbb{X}$'s policies and rate limits. Importantly, we did not engage with TB agents through replies or direct messages during the experiment and data collection.

TB Victims. We observed that OSN users frequently receive replies from TB agents after including trigger keywords in their tweets. These keywords serve as entry points for many modern TB campaigns, signaling TB agents' interest. As a result, unsuspecting users who mention these keywords may inadvertently become potential TB victims.

TB Agents. TB agents are programmable social bots activated based on the trigger keywords and tasked with interacting and engaging with OSN users. They are designed to mimic normal or legitimate user behavior, often posing as support representatives or service providers. Our analysis of their replies revealed (see §IV-B) that the messages are often contextually similar and appear to be paraphrased variations, likely to evade detection while maintaining relevance (§IV-F).

TB Followers. These are followers of TB agents. Our initial assumption was that TB agents solely promoted the campaigns. However, our findings (see §IV-D) revealed that TB followers, *specialized* supporting accounts, play a crucial role in making the TB agents appear more legitimate. These followers typically show minimal activity, such as an empty profile or a single random tweet on their timeline, to hide suspicion and avoid detection on OSN.

TB Masters. At the top of the ecosystem are the TB masters, malicious campaigners who ultimately interact with the victims. TB agents are deployed to manipulate users into initiating contact with these campaigners, typically through external channels such as email. Notably, we discovered (see §IV-E) that TB agents and followers associated with different trigger keywords often share similar characteristics, suggesting that multiple campaigners may outsource their operations to the same TB farms. In some instances, to increase campaign efficiency, TB campaigners may also directly operate as TB agents and interact through the direct messaging feature on $\mathbb{X}$.

TB Farms. To maximize agents reach to potential TB victims, TB masters utilize existing public or underground online marketplaces like TB farms, entities that offer specialized services to buy, sell, and manage profiles of TB agents and TB followers on $\mathbb{X}$ [16]. These farms mimic 'like farms' [7] or distributed call centers [6], coordinating accounts that simulate diverse geographical locations and behavior to enhance authenticity and credibility.

B. Related Work

Figure 2 illustrates the core components for addressing the TB ecosystem on OSNs. This subsection compares related work across these components, highlighting how prior approaches differ from or align with ours.

Previous work has focused on specific entities such as hashtags [17], lists [18], or mentions, whereas our use of trigger words generalizes these, allowing broader applicability. Methods like [1], [2] resemble our honeytrap but are limited to fake cryptocurrency campaigns, overlooking other domains such as misleading giveaways and illicit product campaigns. Work in [17] only partially detects TB agents based on hashtags and static control (e.g., post/follower counts and friends), which are unstable across accounts and campaigns. Other efforts on fake follower campaigns [19] focus on inflating popularity without addressing campaign context. Overall, prior studies lack integration of TB agents and followers, a central focus of our research.

Most work focuses only on bot detection, often including benign bots, without capturing trigger behaviour or malicious intent [20]. In contrast, our work distinguishes between malicious and benign TB agents, going beyond standard detection. While earlier research notes that bots reply during specific hours [3], it lacks an analysis of temporal posting patterns. We find that TB agents post intermittently to evade detection (see §IV-C), a behavioral trait not previously reported.

Coordinated TB agent behaviour suggests the influence of shared TB masters or farms, aligning with evidence of account commodification on online marketplaces [16]. Although some research has explored malicious TB agents on platforms such as YouTube [3], [21] and Facebook [6], [7], little attention has been given to TB followers and masters. We focus our study on $\mathbb{X}$, due to its large user base and its role as an initial target for adversaries [1], [2]. To date, no study has comprehensively mapped the full TB ecosystem, as shown in Figure 2.

III. TBTrackerX

This section presents the TBTrackerX methodology for deploying a honeytrap and tracking TB accounts on $\mathbb{X}$. We detail our TB data collection process and thoroughly explain how ground truth data is established for precise measurement and analysis.

A. TB-Related Keywords Selection

Identifying suitable trigger keywords is a key challenge in attracting TB agents. Therefore, we conducted a systematic investigation of TB-prone keywords. We examined an initial list of 30 keywords previously reported in the academic literature [1], [2], journalistic investigations, and limited disclosures from OSN platforms [11], [14]. Determining which keywords remain effective (or active) over time requires ongoing validation, as TB masters constantly evolve their tactics. To this end, we closely monitored the 𝕏 platform and performed preliminary tests to validate each keyword, ensuring its responses reliably attracted TB agents. From the initial list, 19 keywords were active and ongoing. After reviewing motivations of TB agents from previous literature [1], [2], [22], [23], the scambait forum and database [24], and internal discussions, we grouped active keywords into four types of campaigns, discussed later in this subsection. Then we narrowed the keywords from 19 to 10 as some of them overlap. We designed our methodology to isolate the impact of each keyword, meaning each honeytrap post included one keyword to observe keyword-specific TB agents' behavior independently.

Diversification of TB agents' Campaigns. Responses from TB agents may vary depending on the type of campaign, such as fake giveaways, tech support, or product-related schemes. To capture a representative range of TB agent behaviors, we select keywords across three broad categories of malicious campaigns, unlike those that are benign.

- **Malicious Campaign 1:** ***Deceptive support campaigns.*** In this campaign, TB agents impersonate legitimate personnel to deceive end users for financial gains [2]. Trigger keywords such as '`metamask`', '`trustwallet`', and '`hacked`' indicate illegitimate tech support profiles promoting these campaigns. Specifically, both '`metamask`' and '`trustwallet`' are closely linked to cryptocurrency wallets. Meanwhile, TBs posing as tech support under the guise of '`hacked`' falsely present themselves to help OSN users regain account access. These keywords have also been noted in previous studies [1], [2].
- **Malicious Campaign 2:** ***Illicit product campaigns.*** Recent studies show that adversaries have extensively misused 𝕏 to promote illicit goods and services [23]. In our list of potential keywords, we found that TB agents responding to '`IPTV`' and '`shrooms`' appear as sellers or affiliates of prohibited products. IPTV refers to the technology that delivers over-the-top (OTT) media services. `IPTV` TB agents typically point to third-party IPTV sellers, often associated with pirated or redistributed content without proper licensing. Conversely, '`shrooms`' TB agents act as questionable resellers of various illegal psychedelic drugs such as shrooms (abbreviated for psilocybin magic mushroom) or DMT (dimethyltryptamine) [25].
- **Malicious Campaign 3:** ***Misleading giveaway campaigns.*** During our keyword analysis, we identified a community of fake giveaway profiles on 𝕏, that respond to keywords such as '`sugar daddy`' and '`cashapp`'. These TB agents lure users with false promises of gifts and build online relationships, only to later demand and extort money. These are commonly known as pig-butchering or romance scams [26], [27].
- **Collecting Benign TB Agents.** To distinguish *malicious* TB agents' behavior, we also consider agents that offer legitimate services (e.g., freelance services), drawn to *benign* trigger keywords. To gather such data, we select three keywords: '`logo`', '`graphic designer`', and '`essay`'. TB agents in this category often offer help with arts or digital services. Their benign nature is evidenced by their tendency to share creative portfolios or discuss their work. We also observed that these TB agents stopped engaging once they realized the inauthenticity of our honeytrap account.

With these additions, as shown in Table I, our data collection captures agents based on $N = 10$ distinct trigger keywords. We acknowledge that while not exhaustive, our measurement exercise encompasses a diverse spectrum of types of TB agents. For additional details on how we address sampling bias, please see Appendix VII-B. Finally, the overlapping keywords (e.g., `robux` similar to `metamask`) we excluded from our selection also highlight how malicious campaigns have increasingly broadened potential TB-prone keywords in recent years.

Abbr. as	TB Agent Campaign Name	TB-prone Keywords

Mal. 1	Deceptive Support Campaigns	`metamask`
		`trustwallet`
		`hacked`

Mal. 2	Illicit Product Campaigns	`IPTV` `shrooms`
Mal. 3	Misleading Giveaway Campaigns	`cashapp` `sugar daddy`

Benign	Benign Campaigns	`logo`
		`graphic designer`
		`essay`

TABLE I: Overview of campaigns and underlying TB keywords.

B. Data Crawling

This section outlines the data crawling process, detailing the modules of TBTrackerX used to set up a honeytrap and collect the dataset of TB agents for our analysis.

HoneyTrap Tweet Generator Module. This module generates honeytrap tweets, each containing a single trigger keyword. Our honeytrap tweet consists of a single sentence that contains a brief context, for example "`I need help with <trigger keyword>`" before the trigger keyword to *bait* the TB agents. Furthermore, we append unique UNIX timestamps to each post to avoid appear consecutive identical tweets as shown in Figure 1. Our tweet generator module corroborates previous findings that TB agents respond to specific keywords while ignoring the broader context.

Crawling Module. To study TB agents' modus operandi on 𝕏, our crawler focuses on the primary location where TB agents operate: the comment sections of tweets containing trigger keywords (cf. Figure 1). We used our Honeytrap account to post tweets twice daily (every 12 hours) for 30 days–once per keyword–adhering to 𝕏's rate limits.

After each tweet, we waited 12 hours before initiating data collection. The crawler captured replies and profile metadata from all accounts engaging with the tweet. When a TB agent

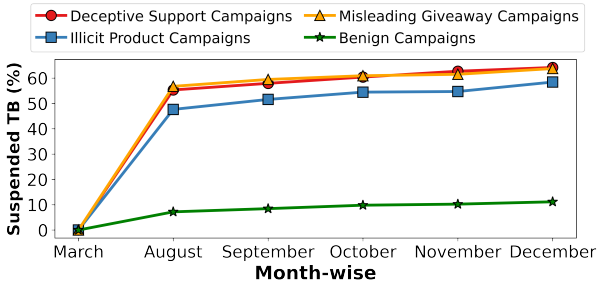

Fig. 3: Monitoring of TB agents suspension by 𝕏. The plot shows that as many as 57% to 67% of TBs have been suspended across malicious campaigns by the end of December 2024.

Data Overview	Mal. 1	Mal. 2	Mal. 3	Benign
TB Agent Count	1,201	724	504	247
TB HoneyTrap Replies	2,019	1,072	977	384
TB-Suspended	802 (67%)	411 (57%)	330 (65%)	31 (13%)
TB-Active	346 (29%)	179 (25%)	115 (23%)	196 (79%)
TB-Deleted	41 (3%)	131 (18%)	55 (11%)	18 (7%)
TB-Changed Screen Name	21 (2%)	24 (3%)	20 (4%)	13 (5%)
TB Victim Replies Count	105,416	60,189	48,002	18,235
TB Victim Unique Count	30,824	14,372	29,277	10,235
TB Followers Count	106,147	102,286	61,419	280,580
TB Friends Count	31,437	256,572	88,901	184,414

TABLE II: Overview of TB agent data collection. Here, Mal. (1, 2, 3), and Benign refer to deceptive support, illicit product, misleading giveaway, and benign campaigns, respectively.

replied, we collected their 100 most recent replies and network metadata (including followers and followees). This provides a broader view of the campaign behavior from four key perspectives for analysis: profile characteristics, reply content, temporal patterns, and network structures. For clarity, we refer the replies made directly to our honeytrap tweets as "*honeytrap replies*" and the most recent 100 replies by the same agent to other OSN users' tweets as "*TB victim replies*". Collecting only 100 recent replies is a trade-off balancing rate limit constraints, storage, and processing time. Prior work has shown that 100 replies are sufficient to capture account behavior [28].

C. Dataset and Ground Truth Labeling

After configuring the crawling module and defining a set of TB-prone keywords, we deployed TBTrackerX on the 𝕏 platform. Over 30 days, from March 1 to March 30, 2024, we collected a dataset comprising 4,452 honeytrap reply messages and 2,647 TB agents. Table II summarizes the distribution of TB agents across different TB campaigns. Overall, the TB agents interacted with over 84K unique potential TB victims, which includes victims gathered from the recent 100 replies. Additionally, the TB agents had an aggregate follower count of over 500K, indicating their potential for wide dissemination. Figure 10 (in Appendix §VII-D) presents a word cloud of TB agent replies from our dataset, highlighting the diversity in their content across campaigns. More screenshots of the TB honeytrap replies can be found in the Appendix (§VII-H), and keyword-specific collected TB agent data overview in Appendix Table X.

Interestingly, we found that between 57% (illicit product agents) and 67% (deceptive support agents) of malicious TB agents that interacted with our honeytrap accounts across different campaigns have been suspended by 𝕏's enforcement mechanisms, as shown in Figure 3. We use this suspension data as part of our ground truth labeling strategy. However, 640 malicious TB agents remained active at the time of our final snapshot (i.e., December 2024), suggesting that these actors successfully evaded 𝕏's detection systems (cf. §IV-F).

After preprocessing, we compiled the dataset for manual labeling. Each active TB agent was assigned one of two labels: (TB agent) bot or human. Three annotators, proficient in English and with a postgraduate college education, carried out the labeling process. The average inter-annotator agreement was 85.9%, demonstrating high consistency and reliability. Overall, 96% of the active TB agents were labeled as bots and 4% as humans. The small proportion of human-labeled accounts indicates a minimal false positive rate for including legitimate human accounts in the dataset. These results show that our honeytrap module is highly effective in attracting malicious TB agents. Please refer to Appendix VII-E for additional information about the labeling process.

IV. TB Ecosystem Measurement

In this section, we comprehensively analyze the collected TB agents by examining their behavior across four key dimensions: profile characteristics, reply content, temporal patterns, and social network structures. Moreover, our objective is to identify consistent behavioral markers that distinguish malicious TB agents from their benign counterparts.

A. Analysis of TB Agents Profile Characteristics

This section explores the TB agent's profile across five attributes: profile setup behavior, creation year, device preference, languages used, and TB agent's reply preference (i.e., reply-to-tweet or reply-to-reply).

1) ***Profile Setup Behavior****:* Our results in Table III show that over ≈95% of TB agents in all four campaigns have custom profile images, indicating an effort to appear genuine. TB agents show differences in other parts of their profile setup across campaigns. For example, deceptive support campaigns exhibit significantly less manual intervention, suggesting more script-controlled activity, compared to the misleading giveaway campaign. They are less likely to have filled out the location field (21.2% vs. 51.4%), the description field (20.9% vs. 71.2%), and to allow direct messages (8.8% vs. 88.5%). This behavior highlights that this group of agents tends to be anonymous and covert to avoid tracking. Conversely, agents from misleading giveaway campaigns are more likely to stay on 𝕏 for further interaction with potential victims through direct messages.

In contrast, the illicit products campaign resembles the deceptive support and misleading giveaway campaigns. From Table III, we observe that these TB agents tend to keep their DM disabled, while many provide a description (52.1%) and location information (71.1%). Conversely, benign TB agents show greater authenticity, with a significant proportion including descriptions and location details, and are DM-enabled.

Attribute	Mal. 1	Mal. 2	Mal. 3	All Mal.	Benign
Profile Setup Behavior					
- With Description	20.9%	52.1%	71.2%	40.4%	79.0%
- Location-Filled	21.2%	71.1%	51.4%	42.1%	55.5%
- DM-Enabled	8.8%	6.6%	88.5%	24.7%	47.0%
- Custom Profile Image	96.8%	99.0%	94.8%	97.0%	95.1%
- Verified	1%	0%	0%	0.5%	10.9%
Creation Year					
- 2024 (January-February)	54.5%	43.7%	44.5%	49.3%	29.2%
- 2023	27.9%	48.7%	22.2%	32.7%	23.9%
- 2022-2009	17.7%	7.7%	33.3%	18.0%	47.0%
Device Preference					
- Android	83.0%	87.8%	19.8%	70.2%	39.0%
- iPhone	16.4%	9.9%	77.1%	28.2%	7.5%
- Web App	0.6%	2.2%	3.1%	1.6%	45.4%
- TweetDeck	0%	0%	0%	0%	0.2%
- TweetDeck Web App	0%	0%	0%	0%	7.9%
Language Usage					
- English	98.1%	98.2%	85.4%	95.3%	89.1%
- Undetermined (e.g., und)	0.1%	0.3%	9.4%	2.3%	4.6%
- Non-Standard (e.g., qme)	0.1%	0.8%	0.4%	0.4%	3.4%
- Others (e.g., tr, ja, fr, es)	1.7%	0.7%	5.8%	2.0%	2.9%
Reply Preference					
- Direct (Reply-to-Tweet)	60.1%	77.2%	71.6%	57.5%	86.7%
- Threaded (Reply-to-Reply)	39.9%	22.9%	28.4%	42.5%	13.3%
TB Agents Count	1,201	724	504	2,411	247

TABLE III: Overview of TB agent profile analysis. Here, Mal. 1, Mal. 2, Mal. 3, All Mal., and Benign refer to deceptive support, illicit product, misleading giveaway, combined malicious campaigns, and benign campaigns, respectively.

This indicates greater customization and user engagement. Lastly, 𝕏 verified profiles are more prevalent among benign campaigners than among malicious ones, with 10.9% of benign agents being verified compared to hardly any among the malicious agents.

*2) **TB Agents Creation Year**:* Our results in Table III show that of all 2,411 malicious TB agents, 49.3% (1260) agents were created in the first two months of year 2024 (before data collection started in March), followed by 32.7% (836) in 2023, indicating that more than 82% of TB agents (combined for 2023 and 2024) were newly established within the past two years. In contrast, the remaining 18% TB agents were created between 2009 and 2022, which could represent older, repurposed, or sleeper accounts. The data reveals the importance of monitoring newer agents for evolving tactics while acknowledging that older agents may serve strategic purposes.

Conversely, out of 247 benign agents, 53.1% (140) agents were created in 2023 and 2024 (combined), indicating a trend of creating newer accounts among benign agents. However, a significant number of agents, 47.0% (107), date back to 2022 or earlier, indicating the continued presence of older, possibly long-standing users. In addition, most of these agents remain active (79.2%), with only a small proportion suspended (13.3%) or deleted independently (6.8%). The pattern of agent creation and longevity underscores the platform's overall compliance and legitimacy among benign TB agents.

*3) **Preference of Devices Operated by TB Agents**:* Our results in Table III show that all three malicious campaigns exhibit high mobile device usage, particularly Twitter for Android, which dominates in two campaigns: deceptive support (83.0%) and illicit product campaigns (87.8%). It is plausible to expect such behavior in bot farms that control automation and emulation [2], [7]. Interestingly, the third malicious campaign for misleading giveaways is heavily based on Twitter for iPhone (77.1%), indicating a different operational style and behavior. Among them, the use of the Twitter Web App remains minimal, implying limited manual interaction.

In contrast, benign campaigns use devices in a balanced way. For example, Twitter for Web App (45.4%) followed by Android (39%), with some use of TweetDeck (8.1% combined with their Web App) and iPhone (7.5%). The diverse use of devices suggests a natural and varied user behavior, involving probable manual intervention and scheduling within the campaign. The distribution indicates the organic use of TB agents by real individuals, unlike the coordinated, uniform TB farm patterns observed in malicious campaigns.

*4) **Language Usage of TB Agents**:* Our analysis of language attributes reveals that the TB agents use 38 languages. We observe that across both campaigns (malicious and benign), English (en) overwhelmingly dominates responses, accounting for 95.3% and 89.1% of the total replies by the agents, respectively. Notably, a small fraction (each less than 1%) of malicious agent replies appears in languages such as Tagalog (tl), Turkish (tr), Japanese (ja), and French (fr). Meanwhile, benign agents exhibit minor engagement in Spanish (es), Indonesian (in), and Danish (da). This suggests that these accounts primarily target an English-speaking audience. However, the usage of low-resource languages suggests occasional multilingual shifts by TB agents. We speculate that TB agents might use LLMs, text spinners, or machine translation to generate multilingual replies.

Additionally, we note a small proportion of unspoken languages. These are non-standard, potentially obfuscated languages such as und (undetermined), (qme), and (qam). They appear in both campaigns, such as und (2.3% in malicious and 4.6% in benign) and non-standard (0.4% in malicious and 3.4% in benign). The unconventional language suggests that benign campaign agents are more prone to language-detection uncertainty, likely because genuine individuals promoting their work use shorter or more ambiguous tweets (containing only URLs, mentions, media, or hashtags). The finding corroborates an existing study indicating that the presence of und, qme, and qam often signifies a post composed of very brief text or containing only mentions or URLs [29].

*5) **Reply Preference (Direct vs. Threaded)**:* This analysis compares how TB agents engage with trigger keywords in conversations on 𝕏. For example, by replying *directly* to tweets (i.e., reply-to-tweet) or by continuing threads called *threaded* replies (i.e., reply-to-reply). The reply preference attribute across both campaigns reveals notable behavioral differences. Both illicit product agents and misleading giveaway agents favor direct replies (77.2% and 71.6%, respectively).

In contrast, deceptive support agents have (39.9%) of replies made to existing replies (i.e. reply-to-reply), indicating more nested conversations. This suggests that deceptive support agents more actively follow TB-prone keywords even within replies. On the other hand, benign campaigns rely heavily on direct replies (86.7%). This variation suggests that benign TB agents are more likely to respond directly to original tweets

(perhaps through targeted engagements) than to engage with unrelated tweets or to deeper multi-level interactions.

New Insights: *The key takeaways are that (i) A significant portion of the malicious TB agents are fresh accounts (49.3% only created in the first two months of 2024), with a strong preference for mobile platforms, such as Android (70.2%) or iPhone (28.2%). (ii) We found 39.9% of deceptive support agents engage more deeply in threaded conversations (reply-to-reply). (iii) The primary language of all agents is English, while some show multilingual capability in up to 38 languages.*

B. Analysis of TB Agents Replies Content

In this section, we examine the replies from honeytrap and TB victims to analyze three attributes. First, identify the preferred contact methods (e.g., email, URLs) promoted in replies. Second, assess content variation across campaigns, and third, check linguistic consistency within campaigns using syntactic (TF-IDF) and contextual (RoBERTa) representations.

*1) **Promoted Contact Methods in HoneyTrap Replies**:* To facilitate further manipulation, TBs often include contact methods, such as email addresses or URLs, in their responses. It is important to note that the choice of contact method suggests how TB victims are lured to continue the manipulation–through the platform or external to the platform. For example, sharing an email address or URL directs users outside of 𝕏, while the DM feature keeps the interaction within the platform.

Our results in Table IV show that all four campaigns favor different contact methods in honeytrap replies. The agents in deceptive support campaigns primarily use email addresses likely to be disguised as legitimate tech support operators. On the other hand, illicit product campaign agents typically mention the names of other OSN accounts (internal to 𝕏 or external, such as Instagram, Telegram, and Facebook) more often than they do other contact methods.

Interestingly, the misleading giveaway agents seek interaction through direct messaging, suggesting a more direct mode of operation. In contrast, benign agents post URLs, highlighting an interest in showcasing their external work portfolios (on websites such as `fiverr.com`). In summary, these communication methods reveal the varying openness and intent of TB agent-driven interactions on the 𝕏 platform.

Methods	Mal. 1	Mal. 2	Mal. 3	Benign C.
Email	**1,225**	3	4	5
𝕏 Direct Message	46	68	**686**	76
URLs	89	85	14	**284**
Mentions	563	**955**	77	7
Phone Number	28	9	8	8
Hashtags	63	169	20	25
All Methods	2,014	1,289	809	405
HoneyTrap Replies	2,019	1,072	977	384
Template-based Reply	1,031	6	3	11

TABLE IV: Distribution of communication methods used by TB agents. Bold denotes the most used contact method.

*2) **Reply Content Variation Between Campaigns**:* We examine the text representations of response content (honeytrap and TB victim) to examine how they vary across campaigns. We filter out non-English replies for each campaign and preprocess them (to remove email, URLs, etc.). Next, we use RoBERTa, a transformer-based language model, to capture contextual meanings that convert each sentence into a dense semantic embedding. Finally, we visualize the high-dimensional embeddings in 2D using t-distributed Stochastic Neighbor Embedding (t-SNE).

Our result, shown in Figure 4, shows that all four campaigns form well-separated clusters in the embedding space for both (honeytrap and TB victim) reply types. Ideally, this difference in textual content is expected as content is tailored specifically to the campaign's needs and motives. However, we note that (i) the left (honeytrap replies) plot of the Figure 4 shows two sub-clusters of benign agents (in green), and (ii) the plot on the right (TB victim replies) in Figure 4 shows a broader spread of benign agent content (in green) in the embedding space. The two insights reveal the heterogeneity and wider embedding dispersion of benign TB agents, underscoring the variability in responses generated by authentic human users.

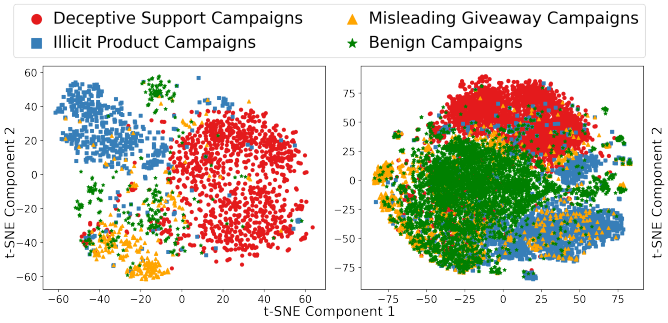

Fig. 4: The t-SNE visualization of (a) *honeytrap replies* (on the left), and (b) *TB victim replies* (on the right) between campaigns. The two plots show well-separated clusters and a broad (heterogeneous) content diversity in benign campaigns.

*3) **Evaluating Linguistic Consistency Within Campaign**:* We quantify linguistic consistency across all TB agents' textual responses within their campaigns. For this, we employ two complementary text representation methods: a contextual embedding model and a lexical-based vectorizer. We use RoBERTa for the contextual approach, as it captures nuanced meanings and context beyond surface-level word overlap. We then compute the pairwise cosine similarity scores for all sentence pairs, yielding a symmetric similarity matrix. To avoid redundancy, we extract only the upper triangular values (excluding the diagonal), representing the unique pairwise semantic similarities. In parallel, we apply the traditional TF-IDF (Term Frequency–Inverse Document Frequency) method to encode each sentence into sparse lexical vectors, emphasizing the frequency and uniqueness of word usage. Cosine similarity was similarly computed over these TF-IDF vectors to measure syntactic similarity.

In the left plot of Figure 5, our results show that all campaigns exhibit high context similarity (most similarities > 0.90). About 50% of replies in all campaigns have a RoBERTa similarity between 0.95 and 0.97. In contrast, up to

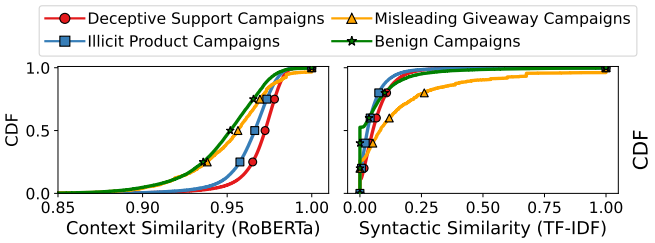

Fig. 5: The distribution of (a) context (on the left) and (b) syntactic similarity (on the right) of honeytrap replies. The plots show contextual consistency within campaigns, but low word overlap, highlighting *different words–same meaning* behavior.

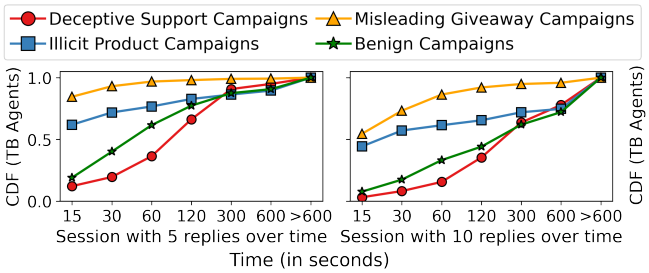

Fig. 6: Temporal session analysis of TB agent's replies.

95% of campaign replies have a TF-IDF similarity < 0.30, indicating minimal word overlap. This contrasting finding suggests extensive use of paraphrasing, highlighting *different words–same meaning* behavior among TB agents, possibly to evade content filters on OSNs. Additionally, the misleading giveaway agents' TF-IDF curve (in yellow) is notably flatter, indicating a high level of vocabulary match. This aligns with the observation that this group of TB agents frequently replies with syntactically similar text pairs, often using brief messages such as 'DM me' or 'message me'.

Our previous subsection demonstrated that TB agents actively promote specific contact methods within their replies. In this subsection, we observe that TB agents frequently employ paraphrasing. We hypothesize that these agents rely on a generation mechanism that rephrases content while consistently embedding contact methods such as email addresses, URLs, or mentions enclosed in braces within sentences. To test this hypothesis, we conducted an empirical analysis using a regular expression to extract honeytrap replies that contain {}, [], or () braces. Our experiment found that 1,031 replies (51.0%) in deceptive support campaigns included these enclosed braces to insert distinct email addresses. This finding suggests that TB agents engaged in deceptive support campaigns follow a template-based approach as shown in Table IV. In contrast, other campaigns provided minimal evidence of similar structured template patterns. We further confirmed this observation through manual analysis across all campaigns.

New Insights: The key takeaways are: *(i) TB agents exhibit a strong tendency to post contextually similar replies (similarities > 0.90). These replies exhibit low-word overlap (< 0.30), indicating systematic paraphrasing intended to evade detection on* 𝕏*. (ii) Different campaigns focus on specific primary contact methods: deceptive support agents employ email (1,225 times), illicit product use mentions (955 times), and misleading giveaway agents request to direct message (686 times) to facilitate subsequent stages of manipulation.*

C. Analysis of TB Agents Temporal Patterns

In this subsection, we perform fine-grained session-based temporal profiling of agents. By examining the timestamps of TB victim replies, we uncover distinct session-level dynamics, such as short, bursty activity versus prolonged idle periods, which constitute a critical part of a TB agent's temporal signature. We define a *short-bursty session* as a sequence of five or more consecutive replies posted with inter-reply intervals of 15 seconds or less. To facilitate this analysis, TBTrackerX normalizes each agent's timeline by assigning the most recent reply a timestamp of zero and expressing all prior reply times as relative offsets (in seconds). This transformation enables consistent temporal feature extraction across accounts.

Subsequently, we apply a time-based sliding window (t) approach with a varied range of window lengths $t \in \{15, 30, 60, 120, 300, 600, > 600\}$ seconds–selected based on prior work in social bot detection [1]. For each window t, we identify sessions that contain at least n consecutive replies (where n is experimentally validated) and compute the number of TB agents exhibiting such behavior over time. This profiling yields features such as inter-reply intervals, burst durations, and session lengths, which are instrumental for capturing the underlying automation traits of TB agents.

In Figure 6, we present the results of session-based temporal profiling using two configurations: n=5 (on the left) and n=10 (on the right), where n denotes the minimum number of consecutive replies within a session. Using our time-based session detection approach, we observe that over 75% of TB agents associated with misleading giveaway campaigns exhibit short-burst behavior—posting at least five consecutive replies with inter-reply intervals no greater than 15 seconds. This bursty activity pattern is strongly indicative of automation, likely orchestrated through scripts. Such uniform and rapid behavior provides clear signals for detection and appears correlated with 𝕏's high suspension rates of these TB agents (cf. Fig. 3).

Notably, increasing the threshold to $n = 10$ does not substantially alter the temporal signature of these TB agents, underscoring the pervasive nature of their bursty activity. In contrast, deceptive support agents display a markedly different temporal pattern: more than 80% TBs in this category exhibit inter-reply intervals of 300 seconds (5 minutes) or longer. This behavior suggests a deliberate strategy to mimic benign activity and evade detection systems by avoiding rapid posting. In particular, we observe that increasing the session window size (t) consistently captures a higher proportion of TBs. Specifically, TB agents across campaigns exhibit *intermittent* activity patterns, oscillating between dense short bursts of replies and prolonged idle periods. This variability in posting behavior is a distinctive temporal trait of malicious automation.

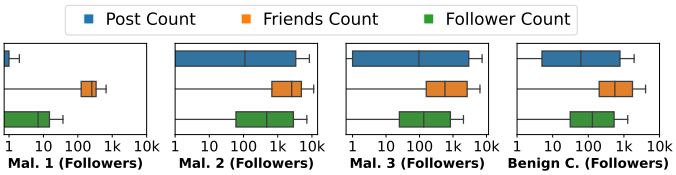

Fig. 7: Profile attributes of TB followers in terms of count of followers, friends, and lifetime posts across campaigns.

D. Analysis of TB Followers

A critical element of the TB ecosystem involves inauthentic TB followers. These TB followers do not promote TB agents' campaigns, as such overt coordination would reveal suspicious behavior indicative of a botnet. Instead, these TB followers operate as online sleeper agents capable of dynamically associating with or detaching from TB agents as needed. We unveil the existence of such TB followers in deceptive support campaigns using the metadata attributes of the follower accounts linked to TB agents. As shown in Figure 7, these followers are primarily designed to follow a large number of deceptive support TB agents, with a median nearing 250 TB agents (highlighted in orange in the first plot). Simultaneously, they exhibit minimal posting activity, often limited to a single tweet (shown in blue), which provides just enough activity to avoid detection while avoiding the red flag of an empty profile.

In contrast, TB followers in other campaigns exhibit more moderate behavior with a balanced follower-to-following ratio (shown in orange and green). However, we speculate that a blend of inauthentic and benign followers exists in the other two malicious campaigns. It should be noted that many online tactics artificially inflate the OSN account metrics. A typical example is *#follow4follow* and *#like4like* reciprocal tactics, which bad actors may misuse to exploit less experienced users seeking rapid online popularity. These reciprocal engagement schemes create the illusion of popularity and trustworthiness, which malicious agents can weaponize to boost follower counts. Attracted by the prospect of quick visibility, naive users may unknowingly amplify malicious agents, contributing to scams, misinformation, or the expansion of inauthentic networks [15]. This form of SMM undermines platform integrity and introduces artificial engagement across the ecosystem.

E. Analysis of TB Masters

In this subsection, we dive deep into the contact method used in honeytrap replies to profile TB masters. Our analysis unfolds in two steps to substantiate the presence of TB masters in orchestrating manipulation campaigns. First, we treat each unique contact method (e.g., email address, URL) as representing a distinct TB master, waiting to initiate subsequent stages of manipulation. Second, we assess how many of these TB masters are orchestrating campaigns with at least n coordinating TB agents.

In Table V, we reveal extensive ongoing malicious activity, with over 300 distinct TB masters each controlling at least one TB agent across various campaigns. These findings indicate that TB agents initiate their manipulation on 𝕏, before redirecting users to other platforms. In deceptive support and illicit product campaigns, a majority of TB masters (356 and 302, respectively) operate outside the 𝕏 platform. In contrast, misleading giveaway campaigns are primarily driven by internal actors with TB masters (around 355) engaging TB victims through direct messages on the platform.

Unique Count	Mal. 1	Mal. 2	Mal. 3	Benign C.
#TB Master ($\geq$ 1 TB Agent)	356	302	42	121
#TB Master ($\geq$ 2 TB Agent)	194	122	8	0
#TB Master ($\geq$ 5 TB Agent)	80	41	0	0
#TB Master ($\geq$ 10 TB Agent)	20	9	0	0
#TB Master/Agent (via DM)	32	47	355	54
#TB Agent	1,201	724	504	247

TABLE V: Overview of the TB master operating campaigns with at least n agents promoting unique contact methods in honeytrap replies. The results demonstrate the varying degrees of campaign coordination and reach in the TB ecosystem.

Notably, as n increases, we observe heightened coordination. For instance, 20 TB masters manage at least 10 TB agents in deceptive support campaigns, while 9 TB masters do the same in illicit product campaigns. This reflects significant operational coordination. Meanwhile, in benign campaigns, we observe no such coordination, even as n varies, indicating each contact method corresponds to a distinct genuine individual. Overall, our contact method effectively quantifies the scale and coordination of malicious activity within the TB ecosystem.

F. Longitudinal Analysis of TB Agents

This section details a nearly year-long longitudinal analysis of TB agents. Previous studies [1], [2], [23] suggest that malicious TB agents have a short active life cycle of two to six months before suspension. Based on these findings, we periodically check the availability of the captured TB agents four months after the initial data collection to assess evasion rates (see Figure 3 in §III-C). We find 640 malicious TB agents escaped suspension by bypassing 𝕏's detection systems. To fully understand the TB agent's evasion strategy, we revisited the active TB agents on 𝕏 in Jan. 2025 to collect two profile attributes: the time of the TB agent's last reply and the latest reported follower count.

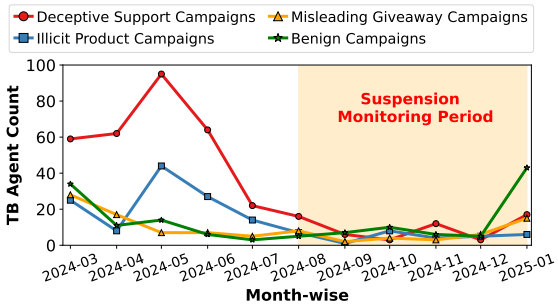

Fig. 8: Longitudinal analysis of active TB agents on 𝕏

In Figure 8, we illustrate the monthly count of TB agents, categorized by the time of their most recent activity. Our result highlights the following insights: (i) Most agents entered *dormancy* periods. Across all malicious campaigns, TB agents

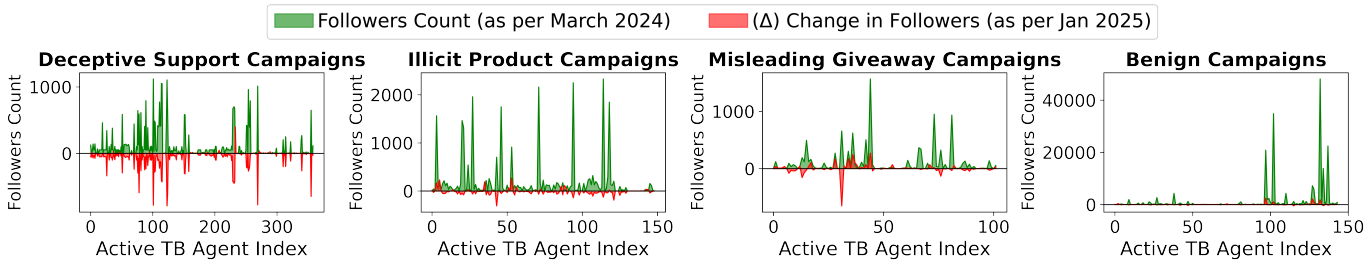

Fig. 9: Longitudinal analysis of active TB agents in terms of change in count of *followers* compared between March 2024 and January 2025. The red area in the plot highlights that active malicious TB agents *disassociate* from inauthentic TB followers to evade detection.

observed a peak for a short period (till Jul. 2024) and stopped activity by Aug. 2024, highlighting the time we started our periodic suspension monitoring. (ii) Based on the TB agent's latest follower counts in Jan. 2025, we observe that many malicious TB agents in deceptive support campaigns disassociate or completely separate from TB followers to increase longevity on the platform. This is highlighted by the (red) change in the follower count plot illustrated in Figure 9. (iii) In contrast, benign agents show stable or increasing followers demonstrating a regular, continued activity.

V. Detection of TB Agents

In this section, we systematically compare various models for detecting *malicious TB agents* and their *campaigns*. We begin by outlining the problem statement (§V-A), followed by our approach to feature selection (§V-B) and evaluation metrics used (§V-C). Next, we present the experimental study and the baseline models considered (§V-D). Our primary objective is to identify the model that performs best with our selected features, including comparisons against state-of-the-art (SOTA) baselines (§V-E1). Moreover, we conduct an ablation study to determine the importance of individual features (§V-E2). Lastly, we evaluate the generalizability of the best-performing model by testing its detection performance on previously unseen malicious TB agents (§V-E3).

A. Problem Statement

Given training data that consists of TB agent profiles (TB) and their corresponding triggered replies R, where $TB = \{tb_1, tb_2, tb_3..., tb_i\}$ represents the set of TB agents and $R_i = \{R^1\}$ denotes the single triggered reply for each agent tb_i, we aim to perform two evaluation tasks in the testing data – malicious TB agent detection and campaign classification:

- *Malicious TB agent detection*: This task involves predicting a binary label indicating whether the agent is malicious or benign. Formally, the output is $y(TB) \in \{0, 1\}$, where $y(TB) = 1$ indicates that the TB agent tb_i is malicious.
- *Campaign detection*: This is a multiclass classification task where each TB agent is assigned to one of four campaign categories. The output is $y(TB) \in \{0, 1, 2, 3\}$, where $y(TB) = 0$ indicates that TB agent $tb_i \in$ deceptive support, $y(TB) = 1$ indicates $tb_i \in$ illicit product, $y(TB) = 2$ indicates $tb_i \in$ misleading giveaway or $y(TB) = 3$ indicates $tb_i \in$ benign campaigns.

B. Feature Selection

We extract four types of features for each TB agent: numerical, boolean, categorical, and textual. Numerical features include various profile attributes such as *number of followers, friend count, total status posts*, and *tweet frequency* (computed as the ratio of status posts to the account age in days). Boolean features capture the presence or absence of specific profile attributes such as the *geographical location information, biography description*, and *the direct message permission status*. Categorical features include the *type of contact method* and *the preferred device* used by the agent. Textual features are derived from raw reply content using RoBERTa, which generates *campaign-aware embeddings*. Finally, we concatenate all extracted features to form a unified TB feature representation.

C. Evaluation Metric

We evaluate the performance of different models using standard metrics such as *F1-score*, *recall*, and *Matthews Correlation Coefficient* (MCC). The F1-score captures the balance between precision and recall, reflecting the average classification performance and the trade-off between correctly and incorrectly identified TB agents. Recall, also called sensitivity, is the detection rate, measuring the proportion of actual TB agents correctly detected. The MCC assesses the overall quality and stability of the classification by considering all components of the confusion matrix (i.e., TP, TN, FP, and FN). Given the class imbalance in both evaluation tasks, we report the mean and standard deviation of the macro-averaged results over 10 iterations, each using a different seed to ensure robustness and reliability of the evaluation.

D. Baselines and Experimental Setup

We compare the detection performance of 15 SOTA baselines, including classical ML, ensemble-based, generative, transformer, feature-based, and rule-based approaches:

- *Classical ML-based*: We include simple models, such as support vector machine (SVM), logistic regression (LR), and random forest (RF), as noted in prior works [1], [4], [30]. We also consider XGBoost due to its fast and scalable architecture [31]. Unless otherwise stated, we use the default parameters for all models.
- *Ensemble-based*: We adopt a Mixture-of-Expert (MoE) approach, inspired by its recent success in scaling LLMs [32] and social bot detection [33], [34]. MoE models use a gating

mechanism to activate only a subset of "expert" components per input, improving computational efficiency.

- *LLM-based*: Motivated by recent work on LLMs for classification tasks, such as misinformation detection and troll identification [35], [36], we evaluate five open-source LLMs available via Unsloth [37] as baselines: Gemma (gemma-7b-bnb-4bit, gemma-2-9b) [38], Llama (Llama3.1-8B-Instruct, Llama3.2-3B-Instruct) [39] and Phi-4 [40]). In this paper, we refer to these models (listed in Table XI) as Llama-3.1, Llama-3.2, Gemma, Gemma-2, and Phi-4, respectively.
- *Transformer-based models*: We include RoBERTa [41], a general-purpose transformer-based model, and BERTweet, specifically pre-trained on tweets [42]. Both models apply self-attention mechanisms to capture contextual dependencies and are fine-tuned using reply text from TB agents.
- *Feature-based approaches*: We include Botometer Lite [30] and BotHunter [43], which classify accounts based on profile features. Both use Random Forest classifiers to handle diverse input feature sets effectively.
- *Anomaly or rule-based methods*: We consider Swatting [17]. This recent lightweight and interpretable rule-based approach identifies malicious agents by applying fixed boundary thresholds to selected feature values.

We intentionally excluded GNN- and CNN-based detectors [44], [45], as these models require extensive data collection involving spatial or neighborhood-level information. While we acknowledge that this may be perceived as a limitation of our work, such data is typically only accessible to the platform operator. In scenarios where the detector is deployed by an entity external to the OSN (e.g., government or fact-checking agencies), access to the underlying structure is often restricted or unavailable. Consequently, our decision to focus on the selected baseline models is motivated by both practical feasibility and domain-specific constraints.

Experimental Settings: We use the same dataset for both evaluation tasks, but implement separate training and evaluation pipelines for each. In the first pipeline, we treat the *malicious TB agent detection* as a binary classification task by merging all three malicious campaign agents into a single malicious class, resulting in a *benign* versus *malicious* classification setup. In the second pipeline, we formulate campaign detection as a multiclass classification task, where the model is trained to distinguish among four classes: the three individual malicious campaign types and the benign class. For both pipelines, we used a consistent 80:20 train-test split across all models under a holdout evaluation strategy. Each selected model is evaluated on both tasks using the corresponding pipeline. For LLMs, we adopt in-context learning (ICL), a widely used evaluation approach that is particularly effective for few-shot and zero-shot settings [46]. We report details on prompt engineering, LLM implementation, ICL results, and other baseline settings in Appendix §VII-F.

E. Evaluation and Results

*1) **Performance Comparison**:* Table VI shows the performance of multiple models on the tasks of malicious TB agent detection and campaign classification. XGBoost achieves the highest performance across both tasks, with F1-scores of 0.88 and 0.92, respectively. As a gradient boosting method with built-in regularization, XGBoost is well-suited for handling datasets with complex relationships and feature types. The input in both evaluation pipelines includes concatenated numerical, categorical, boolean, and textual embedding features, formats that XGBoost effectively handles. Interestingly, despite being more complex, the Phi-4 model (LLM-based) achieves the second-best performance in both tasks with 3-shot F1-score of 0.87 and 0.91, respectively.

Task →	Malicious TB Agent Detection			Campaign Detection		
Baselines	**F1**	**Recall**	**MCC**	**F1**	**Recall**	**MCC**
SVM	$0.51_{\pm.03}$	$0.52_{\pm.01}$	$0.11_{\pm.08}$	$0.31_{\pm.03}$	$0.33_{\pm.02}$	$0.22_{\pm.04}$
LR	$0.86_{\pm.02}$	$\mathbf{0.93}_{\pm.02}$	$\underline{0.74}_{\pm.04}$	$0.82_{\pm.02}$	$0.83_{\pm.02}$	$0.81_{\pm.02}$
RF	$0.80_{\pm.02}$	$0.75_{\pm.03}$	$0.62_{\pm.05}$	$0.87_{\pm.02}$	$0.86_{\pm.02}$	$0.87_{\pm.02}$
XGBoost	$\mathbf{0.88}_{\pm.02}$	$0.86_{\pm.02}$	$\mathbf{0.77}_{\pm.03}$	$\mathbf{0.92}_{\pm.01}$	$\mathbf{0.92}_{\pm.02}$	$\underline{\mathbf{0.92}}_{\pm.01}$
MoE	$0.83_{\pm.03}$	$0.80_{\pm.03}$	$0.66_{\pm.05}$	$0.88_{\pm.02}$	$0.88_{\pm.02}$	$0.89_{\pm.02}$
Llama-3.1	$0.73_{\pm.02}$	$0.86_{\pm.02}$	$0.52_{\pm.04}$	$0.90_{\pm.01}$	$\underline{0.91}_{\pm.02}$	$0.90_{\pm.01}$
Llama-3.2	$0.45_{\pm.01}$	$0.67_{\pm.02}$	$0.19_{\pm.03}$	$0.72_{\pm.02}$	$0.72_{\pm.03}$	$0.70_{\pm.03}$
Gemma	$0.79_{\pm.02}$	$\underline{0.87}_{\pm.02}$	$0.61_{\pm.04}$	$0.82_{\pm.02}$	$0.82_{\pm.02}$	$0.80_{\pm.02}$
Gemma-2	$0.73_{\pm.02}$	$\underline{0.87}_{\pm.02}$	$0.53_{\pm.04}$	$0.89_{\pm.02}$	$0.90_{\pm.02}$	$0.90_{\pm.02}$
Phi-4	$\underline{0.87}_{\pm.02}$	$\underline{0.87}_{\pm.02}$	$\underline{0.74}_{\pm.04}$	$\underline{0.91}_{\pm.02}$	$0.90_{\pm.02}$	$\underline{\mathbf{0.92}}_{\pm.01}$
RoBERTa	$0.82_{\pm.14}$	$0.81_{\pm.13}$	$0.64_{\pm.30}$	$0.86_{\pm.02}$	$0.86_{\pm.10}$	$0.89_{\pm.05}$
BERTweet	$0.82_{\pm.14}$	$0.80_{\pm.13}$	$0.66_{\pm.25}$	$0.85_{\pm.09}$	$0.84_{\pm.07}$	$0.88_{\pm.05}$
BotHunter	$0.82_{\pm.02}$	$0.79_{\pm.02}$	$0.64_{\pm.03}$	$0.80_{\pm.02}$	$0.79_{\pm.02}$	$0.75_{\pm.03}$
Botometer	$0.75_{\pm.02}$	$0.69_{\pm.02}$	$0.54_{\pm.04}$	$0.73_{\pm.01}$	$0.71_{\pm.01}$	$0.69_{\pm.02}$
Swatting	$0.49_{\pm.02}$	$0.51_{\pm.01}$	$0.07_{\pm.08}$	NA	NA	NA
Model Wins	(4/15)	(3/15)	(0/15)	(10/15)	(9/15)	(14/15)

TABLE VI: Performance of baselines on *malicious TB agent* and *campaign* detection. The bold and underlined values indicate the best and second-best performance, respectively. Standard deviations are given for 10 random seeds. The 3-shot performance is reported for all LLM-based detectors. NA indicates that the model cannot perform well on a multiclass evaluation task. "Model wins" counts baselines where the model outperforms the other task.

The results in Table VI yield five key insights: (i) XGBoost, a classical ML-based model, consistently outperforms all SOTA baselines across both evaluation tasks, demonstrating its robustness and effectiveness in handling diverse feature types. (ii) We observe noticeable differences in performance across different LLM families and model sizes, suggesting that the extent of general prior knowledge varies significantly. Notably, Phi-4 consistently outperforms other LLMs under both few-shot and zero-shot settings (For comprehensive ICL results, refer to Table XII in Appendix §VII-F). (iii) Transformer-based models such as RoBERTa and BERTweet, which rely solely on textual inputs, underperform in detecting malicious TB agents. Misclassifications are especially common for TB agents whose reply test falls outside the token length range (10-60 tokens) used during BERTweet's pretraining [42]. This limitation highlights the importance of incorporating non-textual features for robust classification.

(iv) Multi-dimensional behavioral feature-based methods such as Botometer Lite [30] and BotHunter [43] offer simplicity and interpretability. However, their lack of textual input limits their ability to detect sophisticated TB agents that closely resemble benign accounts. Without leveraging reply content, these models struggle to capture the nuanced seman-

Ablation Setting	Malicious TB Agent Detection			Campaign Detection		
	F1-score	Recall	MCC	F1-score	Recall	MCC
full feature set (baseline)	**0.88**	**0.86**	**0.77**	**0.92**	**0.92**	**0.92**
w/o emb	0.85 (-3.41%) ↓	0.83 (-3.49%) ↓	0.71 (-7.79%) ↓	0.84 (-8.70%) ↓	0.84 (-8.70%) ↓	0.81 (-11.96%) ↓
w/o emb-cat	0.67 (-23.86%) ↓*	0.64 (-25.58%) ↓*	0.36 (-53.25%) ↓*	0.71 (-22.83%) ↓*	0.70 (-23.91%) ↓*	0.69 (-25.00%) ↓*
w/o emb-cat-bool	0.66 (-25.00%) ↓	0.63 (-26.74%) ↓	0.33 (-57.14%) ↓	0.62 (-32.61%) ↓	0.61 (-33.70%) ↓	0.57 (-38.04%) ↓

TABLE VII: Feature ablation study showing absolute values and percentage decrease from the full feature set (baseline). The marker ↓* indicates a significant decrease from the baseline and previous ablation setting.

Type	Training Setting	Test on	Metric	Baseline	+1% Unseen	+5% Unseen	+10% Unseen	+20% Unseen
OvR	Benign vs. (Mal. 1 + Mal. 2)	Mal. 3	**F1**	0.69	0.72 (+0.03) ↑	0.82 (+0.13) ↑	0.85 (+0.16) ↑	0.90 (+0.21) ↑
			Rec.	0.57	0.60 (+0.03) ↑	0.76 (+0.19) ↑	0.82 (+0.25) ↑	0.89 (+0.32) ↑
	Benign vs. (Mal. 1 + Mal. 3)	Mal. 2	**F1**	0.93	0.93 (+0.00) →	0.93 (+0.00) →	0.94 (+0.01) ↑	0.95 (+0.02) ↑
			Rec.	0.94	0.94 (+0.00) →	0.94 (+0.00) →	0.96 (+0.02) ↑	0.97 (+0.03) ↑
	Benign vs. (Mal. 2 + Mal. 3)	Mal. 1	**F1**	0.88	0.96 (+0.08) ↑	0.97 (+0.09) ↑	0.97 (+0.09) →	0.97 (+0.09) →
			Rec.	0.83	0.96 (+0.13) ↑	0.98 (+0.15) ↑	0.98 (+0.15) →	0.99 (+0.16) ↑
OvO	Benign vs. Mal. 1	Mal. 2	**F1**	0.93	0.94 (+0.01) ↑	0.94 (+0.01) →	0.94 (+0.01) →	0.95 (+0.02) ↑
			Rec.	0.90	0.92 (+0.02) ↑	0.92 (+0.02) →	0.92 (+0.02) →	0.94 (+0.04) ↑
		Mal. 3	**F1**	0.61	0.73 (+0.12) ↑	0.79 (+0.18) ↑	0.85 (+0.24) ↑	0.92 (+0.31) ↑
			Rec.	0.46	0.60 (+0.14) ↑	0.69 (+0.23) ↑	0.79 (+0.33) ↑	0.91 (+0.45) ↑
	Benign vs. Mal. 2	Mal. 1	**F1**	0.66	0.93 (+0.27) ↑	0.96 (+0.30) ↑	0.97 (+0.31) ↑	0.97 (+0.31) →
			Rec.	0.50	0.89 (+0.39) ↑	0.95 (+0.45) ↑	0.97 (+0.47) ↑	0.98 (+0.48) ↑
		Mal. 3	**F1**	0.55	0.71 (+0.16) ↑	0.81 (+0.26) ↑	0.86 (+0.31) ↑	0.87 (+0.32) ↑
			Rec.	0.39	0.57 (+0.18) ↑	0.72 (+0.33) ↑	0.79 (+0.40) ↑	0.82 (+0.43) ↑
	Benign vs. Mal. 3	Mal. 1	**F1**	0.38	0.84 (+0.46) ↑	0.95 (+0.57) ↑	0.97 (+0.59) ↑	0.97 (+0.59) →
			Rec.	0.24	0.73 (+0.49) ↑	0.93 (+0.69) ↑	0.97 (+0.73) ↑	0.98 (+0.74) ↑
		Mal. 2	**F1**	0.78	0.89 (+0.11) ↑	0.93 (+0.15) ↑	0.93 (+0.15) →	0.94 (+0.16) ↑
			Rec.	0.66	0.84 (+0.18) ↑	0.91 (+0.25) ↑	0.93 (+0.27) ↑	0.94 (+0.28) ↑

TABLE VIII: Generalizability study across `OvR` and `OvO` settings with incremental exposure to previously unseen variants of malicious TB agents alongside benign agents during training. Results show absolute values and an increase from the *baseline* result highlighted in green. The up arrow ↑ indicates an increase from the previous column and the right arrow → shows no increase. Mal. (1, 2, 3), and Benign refer to deceptive support, illicit product, misleading giveaway, and benign campaigns, respectively.

tics often embedded in deceptive campaign replies. (v) The *campaign* detection task achieves stronger performance than the *malicious TB agent* detection task across most competing baselines (10 model wins, 4 model losses) in terms of F1-score. This outcome is expected, as fine-grained classification introduces less ambiguity than binary classification, where multiple malicious behaviors are collapsed into a single class. Distinguishing among specific malicious campaign types allows models to learn more discriminative patterns.

For simplicity and consistency, we refer–XGBoost, the best-performing model, as the main model in subsequent evaluations of robustness and generalizability in the rest of the paper.

*2) **Feature Ablation Study**:* To understand the contribution of each feature category (numerical, boolean, categorical, and textual), we conduct an ablation study using the XGBoost model. Table VII illustrates the result, where we evaluate the model's performance as individual feature types are systematically removed from the full feature set. For clarity, we define the ablation variants as follows: w/o emb: excludes textual embeddings, w/o emb-cat: excludes both textual embeddings and categorical features, and w/o emb-cat-bool excludes textual embeddings, categorical, and boolean features.

Across both detection tasks, performance drops significantly when features are removed—especially in the F1-score, as highlighted in red in Table VII. This degradation is likely due to the model's reduced representational capacity, which hinders its ability to effectively distinguish between classes when deprived of necessary input signals. Notably, the w/o emb-cat variant shows the most substantial decline, with F1-scores dropping by 23.9% and 22.9% on the two tasks, respectively. This finding underscores that the model derives most of its predictive power from the textual embeddings (i.e., campaign-aware content) and categorical features (e.g., the campaign's preferred contact method). These features capture essential semantic and behavioral cues that are critical for distinguishing between different types of TB agents and campaigns.

*3) **Generalizability Study**:* In this subsection, we evaluate the generalizability of the model, its ability to maintain detection performance when encountering previously unseen malicious TB agents that differ from those seen during training. To assess this, we conduct experiments using two strategies: (i) *one vs. rest* (OvR): each class is evaluated against all others, and (ii) *one vs. one* (OvO): each pair of classes is evaluated independently. These approaches allow us to test the model's robustness to variant malicious TB agents. Additionally, we examine performance trends as we incrementally increase the proportion of unseen TB agents included in the training data (1%, 5%, 10%, and 20%).

Table VIII reveals three main insights: (i) Under the OvO setting, training with deceptive support agents (mal. 1) results in higher generalizability to other unseen types (e.g., illicit product and misleading giveaway agents). This suggests that mal. 1 may exhibit greater within-class diversity or more representative patterns, enabling the model to better generalize across unseen malicious behaviors. (ii) As the proportion of unseen TB samples increases in the training data, F1-score and recall show a steady upward trend. This improvement is expected, as additional data enables the model to learn more discriminative features. Notably, once the unseen sample size reaches 20%, the F1-score stabilizes above 0.90, indicating

strong generalization. A similar trend is observed for recall. (iii) Incorporating a wider variety of subclasses during training significantly improves generalizability. This emphasizes the importance of capturing a diverse distribution of agent behaviors to detect new or evolving TB agents more effectively.

VI. Discussion

In this section, we summarize the main findings of our work and discuss their practical and real-life implications.

A. Lessons Learned

Hidden Components in the TB Agents Ecosystem. This paper provides the first large-scale empirical analysis of $\approx 2.6K$ TB agents on X, collectively attracting over 500K TB followers and impacting 84K unique TB victims. Additionally, the study identifies common contact method-based campaigns that suggest the presence of TB masters or TB farms, offering key insights into the underlying structure and operational dynamics of the TB ecosystem.

The Operational Taxonomy of Campaigns. We categorize 4.4K honeytrap replies into four distinct campaign types: deceptive support, illicit product, misleading giveaways, and benign campaigns. We observe that malicious TB agents target specific user categories (e.g., cryptocurrency users, psychedelic interest groups, pseudo-romantic victims), demonstrating refined tactics and a purposeful exploitation of niche communities. These patterns highlight a high degree of tactical sophistication within the TB ecosystem.

Deceptive Practices: Evasion Strategies. By analyzing TB profiles and their replies, we identify coordinated evasion strategies employed by malicious TB agents. These include highly paraphrased replies (up to 0.95 and 0.97 context similarity) across campaigns, the use of multilingual responses, and irregular temporal activity patterns, all indicative of shared tactics aimed at avoiding detection. Moreover, many TB agents are pre-configured with fake followers, strategically crafted to enhance perceived credibility and further evade detection.

Dangerous to Dormant Account Longevity. We provide a novel account creation timeline and a nearly year-long longitudinal analysis of TB agents, revealing that 50% of these accounts were created before 2024. Many of these older accounts leveraged their longevity to evade detection by disassociating from TB followers or entering periods of dormancy. Conversely, TB agents created in early 2024 continue to dominate active campaign operations, highlighting the ecosystem's rapid adaptation to evolving platform defenses.

Behavioral Profiling of TB Agents. We identify the use of contact-method-based operations, where agents embed email addresses or URLs in replies to guide users to interactions on external platforms. This tactic enables seamless pivoting from 𝕏 to other ecosystems, often where TB masters await to continue deceptive engagements. These findings illustrate the end-to-end profiling of TB agents and provide crucial insights into how cross-platform deception is coordinated and sustained.

OSN Detection Gaps. Our results show that, despite the platform's efforts, only 57% to 67% of TB agents involved in malicious campaigns were actioned by 𝕏, leaving 640 malicious TB agents still active. This highlights a significant enforcement gap and the urgent need for earlier and more effective detection strategies. TBTrackerX demonstrates promising advancements, achieving detection accuracies of $\approx 88\%$ for *malicious TB agents* and $\approx 92\%$ for *campaign* detection.

B. Recommendations

Based on our findings, we offer targeted recommendations to support future mitigation efforts across these groups.

Online Social Network Platforms. We recommend that OSN platforms enforce stricter multi-factor authentication to deter the creation of autonomous accounts. This includes but is not limited to (i) monitoring inconsistencies in browser agents, headers, or IP locations often linked to Selenium use to detect TB masters or farms, (ii) tightening checks on temporary mail server use for email verification and *CAPTCHA* circumvention during sign-up to suppress easy TB agent deployment, and (iii) favoring shadowbanning over outright suspension to enable ongoing tracking of fake follower networks. Platforms should also implement robust detection systems to flag TB-prone keywords and sanitize posts before publishing. Since users can mute keywords without disengaging from related content, proactive filtering is crucial. Finally, OSNs should run awareness campaigns about these SMM and enhance information sharing across platforms.

OSN End Users. Follower growth tactics among OSN users often fall into a grey area. While platforms prohibit rapid, inauthentic follower growth, practices like *#follow4follow* and *#like4like* unintentionally support TB agent networks. Users should also refrain from using known TB-prone keywords in replies, as these can attract TB agents and increase the risk of denial-of-OSN (DoSN) attacks against potential TB victims. TBTrackerX shows that even test accounts using TB keywords experience a surge of 145+ followers in 30 days. To mitigate such risks, platforms should enforce penalties for individuals or organizations engaging in inauthentic follower growth or intentional misuse of TB keywords, especially when such behavior facilitates manipulation or exploitation.

Security Community. We urge security researchers to pursue key directions that advance detection, transparency, and accountability in addressing increasingly sophisticated SMM. Specifically: (i) During TB data collection, analysts should identify TB-prone keywords used by TB agent accounts. For example, a shrooms-related TB agent may engage with multiple related terms such as `shrooms` and `DMT`, both from the family of psychedelic drugs (refer to Appendix §VII-G for systematic keyword exploration). Establishing a multilingual TB keyword corpus (e.g., メタマスク–the keyword `metamask` in Japanese) could also be helpful, as our preliminary research suggests that TB agents flock to the keyword in both languages (see Appendix §VII-G). (ii) The community should also prioritize reverse engineering of TB agent reply generator

modules. TB agents often use paraphrasing, multilingual, and cross-lingual responses tailored according to the original TB victim's post, indicating the possible use of LLMs (please see Appendix VII-C). Collaborative research with the broader security community and LLM providers can help uncover the underlying mechanisms used by TB masters or farms, enabling campaign-agnostic detection strategies.

C. Limitations and Future Work

Limitations. We acknowledge that our dataset does not capture all TBs on 𝕏. Our findings represent a conservative estimate, establishing a lower bound on the scale and impact of TB campaigns. Moreover, TBTrackerX may miss some interactions with our test account due to platform rate limits. Our crawler captures TB replies every 12 hours to maximize coverage, which may overlook short-lived TB agents (n=19) that were deleted shortly after activity. Nevertheless, our data collection framework remains adaptable for future refinements.

Due to 𝕏 platform's lack of visible and reliable profile attributes such as age, gender, or location, our test account could not be customized to reflect diverse user characteristics. As a result, we emulate TB victim behavior through a single test account, and the role of demographic factors in TB targeting remains out of scope for this study.

Future Work. Our experiments relied on a single test account for TBTrackerX during data collection. To minimize keyword selection bias and isolate the behavior of TB agents responding to specific triggers, we intentionally avoided using multiple trigger keywords in a single post. Exploring the combined effects of trigger keywords remains a direction for future study. We also limited our focus on optimizing the text template-based serialization method. While our current approach uses simple templates suitable for interaction with LLMs [47], future work will explore more advanced serialization strategies, such as feature ranking or emphasizing key features with prefixed cues (e.g.,'critically') [48]. Finally, while some work has examined the adversarial robustness of TB detectors [4], evaluating TBTrackerX under such scenarios and enhancing its resilience is a key avenue for future research.

VII. Concluding Remarks

This paper proposes TBTrackerX, a comprehensive framework for detecting TB agents that autonomously operate on OSNs such as 𝕏. We adopt a measurement-driven methodology to understand the tactics, techniques, and procedures employed by both benign and malicious TB agents. To benchmark detection performance, we compared 15 diverse baseline models across two key tasks: *malicious TB agent detection* and *campaign classification*. Additionally, we conduct an ablation study to evaluate feature importance and a generalizability study to assess the model's robustness to unseen TB agents. Our findings demonstrated that a classical XGBoost model achieved strong performance with F1-scores of 0.88 and 0.92 for malicious agent and campaign detection, respectively. These findings show that well-engineered yet straightforward models can outperform more complex baselines when leveraging rich multimodal features. Overall, TBTrackerX contributes new insights into the evolving TB agent ecosystem, revealing a growing presence of coordinated SMM activities and offering a practical path forward for detection in real-world settings.

References

[1] M. M. Akhtar, R. Masood, M. Ikram, and S. S. Kanhere, "Sok: False information, bots and malicious campaigns: Demystifying elements of social media manipulations," in *Proceedings of the 19th ACM Asia Conference on Computer and Communications Security*, ser. ASIA CCS '24. New York, NY, USA: Association for Computing Machinery, 2024, p. 1784–1800. [Online]. Available: https://doi.org/10.1145/3634737.3644998

[2] B. Acharya, M. Saad, A. E. Cina, L. Schonherr, H. D. Nguyen, A. Oest, P. Vadrevu, and T. Holz, " Conning the Crypto Conman: End-to-End Analysis of Cryptocurrency-based Technical Support Scams ," in *2024 IEEE Symposium on Security and Privacy (SP)*. Los Alamitos, CA, USA: IEEE Computer Society, May 2024, pp. 17–35. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/SP54263.2024.00156

[3] X. Li, A. Rahmati, and N. Nikiforakis, "Like, comment, get scammed: Characterizing comment scams on media platforms," in *Proceedings Network and Distributed System Security Symposium*, 2024.

[4] M. M. Akhtar, N. S. Bhuiyan, R. Masood, M. Ikram, and S. S. Kanhere, "Botsscl: Social bot detection with self-supervised contrastive learning," *Online Social Networks and Media*, vol. 48, p. 100318, 2025.

[5] M. M. Akhtar, I. Karunanayake, B. Sharma, R. Masood, M. Ikram, and S. S. Kanhere, " Towards Automatic Annotation and Detection of Fake News ," in *2023 IEEE 48th Conference on Local Computer Networks (LCN)*. Los Alamitos, CA, USA: IEEE Computer Society, Oct. 2023, pp. 1–9. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/LCN58197.2023.10223359

[6] J. Liu, P. Pun, P. Vadrevu, and R. Perdisci, "Understanding, measuring, and detecting modern technical support scams," in *2023 IEEE 8th European Symposium on Security and Privacy*. IEEE, 2023, pp. 18–38.

[7] M. Ikram, L. Onwuzurike, S. Farooqi, E. D. Cristofaro, A. Friedman, G. Jourjon, M. A. Kaafar, and M. Z. Shafiq, "Measuring, characterizing, and detecting facebook like farms," *ACM Transactions on Privacy and Security (TOPS)*, vol. 20, no. 4, pp. 1–28, 2017.

[8] K.-C. Yang and F. Menczer, "Anatomy of an ai-powered malicious social botnet," *arXiv preprint arXiv:2307.16336*, 2023.

[9] C. Grimme, J. Pohl, S. Cresci, R. Lüling, and M. Preuss, "New automation for social bots: from trivial behavior to ai-powered communication," in *Multidisciplinary International Symposium on Disinformation in Open Online Media*. Springer, 2022, pp. 79–99.

[10] J. Piao, Z. Lu, C. Gao, and Y. Li, "Social bots meet large language model: Political bias and social learning inspired mitigation strategies," in *Proceedings of the ACM on Web Conference*, 2025, pp. 5202–5211.

[11] L. Abrams, "Twitter bots pose as support staff to steal your cryptocurrency," https://www.bleepingcomputer.com/news/security/twitter-bots-pose-as-support-staff-to-steal-your-cryptocurrency/, 2023, accessed: 1 November 2023.

[12] Y. Zhang, K. Sharma, L. Du, and Y. Liu, "Toward mitigating misinformation and social media manipulation in llm era," in *Companion Proceedings of the ACM Web Conference 2024*, 2024, pp. 1302–1305.

[13] C. Chen and K. Shu, "Combating misinformation in the age of llms: Opportunities and challenges," *AI Magazine*, vol. 45, no. 3, pp. 354–368, 2024.

[14] CoinCodeCap, "Warning: Metamask phishing scams proliferate on x (twitter)," https://coincodecap.com/warning-metamask-phishing-scams-proliferate-on-x-twitter, 2024, accessed: 5 January 2024.

[15] L. Vargas, P. Emami, and P. Traynor, "On the detection of disinformation campaign activity with network analysis," in *Proceedings of the 2020 ACM SIGSAC Conference on Cloud Computing Security Workshop*, ser. CCSW'20. New York, NY, USA: ACM, 2020, p. 133–146.

[16] M. Beluri, B. Acharya, S. Khodayari, G. Stivala, G. Pellegrino, and T. Holz, "Exploration of the dynamics of buy and sale of social media accounts," *arXiv preprint arXiv:2412.14985*, 2024.

[17] C. Brokate, M. Richard, L. Giordani, and J. Liénard, "Swatting spambots: Real-time detection of malicious bots on x," in *Companion Proceedings of the ACM Web Conference 2024*, 2024, pp. 818–821.

[18] K. Li, D. Lee, and S. Guan, "Understanding the cryptocurrency free giveaway scam disseminated on twitter lists," in *2023 IEEE International Conference on Blockchain (Blockchain)*. IEEE, 2023, pp. 9–16.
[19] Y. Zouzou and O. Varol, "Unsupervised detection of coordinated fake-follower campaigns on social media," *EPJ Data Science*, vol. 13, no. 1, p. 62, 2024.
[20] L. H. X. Ng and K. M. Carley, "What is a social media bot? a global comparison of bot and human characteristics," *arXiv preprint arXiv:2501.00855*, 2025.
[21] S. H. Na, S. Cho, and S. Shin, "Evolving bots: The new generation of comment bots and their underlying scam campaigns in youtube," in *Proceedings of the 2023 ACM on IMC*, 2023, pp. 297–312.
[22] F. Blocker, "Twitter spam bots: Faking comments & engagement," https://fraudblocker.com/articles/twitter-spam-bots, 2025, accessed: 12 July 2025.
[23] H. Wang, Y. Li, R. Huang, and X. Mi, "Detecting and understanding the promotion of illicit goods and services on twitter," in *Proceedings of the ACM on Web Conference 2025*, 2025, pp. 3389–3404.
[24] Scammer.info Community, "Scammer.info: Scam reporting and scambaiting forum," Online platform, 2025, available online at: https://scammer.info/, accessed 2025-08-06.
[25] C. P. R. Andrea Chalfin, "Authorities warn of psilocybin scam that appears to originate in manitou springs, but is linked overseas," https://www.cpr.org/2025/02/26/manitou-springs-psilocybin-scam/, 2025, accessed: 5 March 2025.
[26] B. Acharya and T. Holz, "An explorative study of pig butchering scams," *arXiv preprint arXiv:2412.15423*, 2024.
[27] M. T. Whitty and T. Buchanan, "The online romance scam: A serious cybercrime," *CyberPsychology, Behavior, and Social Networking*, vol. 15, no. 3, pp. 181–183, 2012.
[28] L. H. X. Ng, D. C. Robertson, and K. M. Carley, "Stabilizing a supervised bot detection algorithm: How much data is needed for consistent predictions?" *OSNEM*, vol. 28, p. 100198, 2022.
[29] D. Uniyal and R. Nayak, "Twitter's pulse on hydrogen energy in 280 characters: a data perspective," *Social Network Analysis and Mining*, vol. 14, no. 1, p. 37, 2024.
[30] K.-C. Yang, O. Varol, P.-M. Hui, and F. Menczer, "Scalable and generalizable social bot detection through data selection," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 01, 2020, pp. 1096–1103.
[31] 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.
[32] A. Q. Jiang, A. Sablayrolles, A. Roux, A. Mensch, B. Savary, C. Bamford, D. S. Chaplot, D. d. l. Casas, E. B. Hanna, F. Bressand *et al.*, "Mixtral of experts," *arXiv preprint arXiv:2401.04088*, 2024.
[33] L. H. X. Ng and K. M. Carley, "Botbuster: Multi-platform bot detection using a mixture of experts," in *Proceedings of the international AAAI conference on web and social media*, vol. 17, 2023, pp. 686–697.
[34] Y. Liu, Z. Tan, H. Wang, S. Feng, Q. Zheng, and M. Luo, "Botmoe: Twitter bot detection with community-aware mixtures of modal-specific experts," in *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2023, pp. 485–495.
[35] L. Luceri, E. Boniardi, and E. Ferrara, "Leveraging large language models to detect influence campaigns on social media," in *Companion Proceedings of the ACM Web Conference 2024*, ser. WWW '24. New York, NY, USA: ACM, 2024, p. 1459–1467. [Online]. Available: https://doi.org/10.1145/3589335.3651912
[36] S. Feng, H. Wan, N. Wang, Z. Tan, M. Luo, and Y. Tsvetkov, "What does the bot say? opportunities and risks of large language models in social media bot detection," *arXiv preprint arXiv:2402.00371*, 2024.
[37] M. H. Daniel Han and U. team, "Unsloth," 2023. [Online]. Available: http://github.com/unslothai/unsloth
[38] G. Team, T. Mesnard, C. Hardin, R. Dadashi, S. Bhupatiraju, S. Pathak, L. Sifre, M. Rivière, M. S. Kale, J. Love *et al.*, "Gemma: Open models based on gemini research and technology," *arXiv preprint arXiv:2403.08295*, 2024.
[39] A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan *et al.*, "The llama 3 herd of models," *arXiv preprint arXiv:2407.21783*, 2024.
[40] M. Abdin, J. Aneja, H. Behl, S. Bubeck, R. Eldan, S. Gunasekar, M. Harrison, R. J. Hewett, M. Javaheripi, P. Kauffmann *et al.*, "Phi-4 technical report," *arXiv preprint arXiv:2412.08905*, 2024.
[41] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.
[42] D. Q. Nguyen, T. Vu, and A. T. Nguyen, "Bertweet: A pre-trained language model for english tweets," *arXiv preprint arXiv:2005.10200*, 2020.
[43] D. M. Beskow and K. M. Carley, "Bot-hunter: a tiered approach to detecting & characterizing automated activity on twitter," in *Conference paper. SBP-BRiMS: International conference on social computing, behavioral-cultural modeling and prediction and behavior representation in modeling and simulation*, vol. 3, no. 3, 2018.
[44] Y. Yang, Q. Wu, B. He, H. Peng, R. Yang, Z. Hao, and Y. Liao, "Sebot: Structural entropy guided multi-view contrastive learning for social bot detection," in *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, 2024, pp. 3841–3852.
[45] S. Feng, H. Wan, N. Wang, and M. Luo, "Botrgcn: Twitter bot detection with relational graph convolutional networks," in *Proceedings of the 2021 IEEE/ACM international conference on advances in social networks analysis and mining*, 2021, pp. 236–239.
[46] G. Chochlakis, A. Potamianos, K. Lerman, and S. Narayanan, "The strong pull of prior knowledge in large language models and its impact on emotion recognition," *arXiv preprint arXiv:2403.17125*, 2024.
[47] W. Wei, L. Na, Z. Lei, L. Fang, C. Hao, Y. Xiuying, H. Lei, Z. Min, W. Gang, Z. Jie *et al.*, "An extensive study on text serialization formats and methods," *arXiv preprint arXiv:2505.13478*, 2025.
[48] S. Jaitly, T. Shah, A. Shugani, and R. S. Grewal, "Towards better serialization of tabular data for few-shot classification with large language models," *arXiv preprint arXiv:2312.12464*, 2023.
[49] C. M. Rivers and B. L. Lewis, "Ethical research standards in a world of big data," *F1000Research*, vol. 3, 2014.
[50] S. Hegselmann, A. Buendia, H. Lang, M. Agrawal, X. Jiang, and D. Sontag, "Tabllm: Few-shot classification of tabular data with large language models," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2023, pp. 5549–5581.
[51] Y. Jiang, W. Zhang, X. Zhang, X.-Y. Wei, C. W. Chen, and Q. Li, "Prior knowledge integration via llm encoding and pseudo event regulation for video moment retrieval," in *Proceedings of the 32nd ACM International Conference on Multimedia*, ser. MM '24. New York, NY, USA: ACM, 2024, p. 7249–7258.
[52] L. Ai, T. Kumarage, A. Bhattacharjee, Z. Liu, Z. Hui, M. Davinroy, J. Cook, L. Cassani, K. Trapeznikov, M. Kirchner *et al.*, "Defending against social engineering attacks in the age of llms," *arXiv preprint arXiv:2406.12263*, 2024.
[53] Y. Zhang, K. Sharma, L. Du, and Y. Liu, "Toward mitigating misinformation and social media manipulation in llm era," in *Companion Proceedings of the ACM Web Conference 2024*, 2024, pp. 1302–1305.

APPENDIX

A. Ethical Considerations

Our research is strictly non-commercial, and complies with 𝕏's Terms and Conditions for research use. Data collection was limited to tweets from public user profiles, and no data is shared with third parties for commercial purposes. All results are anonymized and do not allow for the identification or tracking of individual users. We follow ethical guidelines as outlined in [49] and obtained ethics approval from our institution for experiments involving potentially human-generated content. While we propose a honeytrap framework for identifying TB agents, our focus is on measurement, detection, and improving the robustness of the detection system. We do not endorse the utilization of this framework for unethical purposes, including intentionally enticing TB agents or facilitating DoSN attacks against OSN users. Our role remained strictly observational, focused on passive data collection for research purposes, with no actions taken to manipulate platform dynamics or influence user experiences.

- *Minimal involvement of human users*: We ensured at multiple stages that coverage only baited TB agents, not human users.
  1) First, during the preliminary keyword monitoring stage, we (all authors) manually confirmed that each selected keyword only attracted TB agents. We used quantitative evidence of bot activity, such as high reply-to-tweet ratios (close to 1, indicating only reply behavior), repeated contact methods, and semantic consistency across TB replies, all characteristic indicators of automated behavior.
  2) Second, after data collection, external expert annotators independently validated that the honeytrap setup primarily engaged TB agents across campaigns.
- *Avoiding impact on legitimate users*: We distinguish malicious from benign agents to avoid harming legitimate use. We used harmless honeytrap Tweets like "I need help with <trigger keyword>". Finally, all account details in any screenshots are obscured.

B. Addressing sampling bias

We acknowledge that sampling bias remains an inherent challenge in this type of study.

1) *Platform-bias*: We selected 𝕏 because it supports direct user interactions through Tweets and follower relationships. Our methodology and findings (e.g., keyword-based honeytrap setup, evasion tactics, recommendations) are generalizable and can inform defenses across other OSNs vulnerable to reply-based attacks where users or bots can directly engage with posts.
2) *Account-bias*: We initially attempted to create multiple profiles representing different demographic factors. However, 𝕏 platform settings were found unreliable for accurately capturing actual demographic attributes, as age and gender are concealed and the location field is self-declared, often containing arbitrary or fictional text. To maintain methodological validity and avoid misleading inferences, we excluded TB-targeting by demographics. Furthermore, we used a single test account to minimize interaction and operational footprint on 𝕏.
3) *Timing-bias*: To mitigate time-of-day bias, we varied the posting schedule every 10 days during data collection. Our sampling methodology was carefully designed to balance feasibility and coverage, ensuring that observed TB behaviors were representative and not tied to specific posting times.
4) *Keyword-bias*: To our knowledge, this is the first study to analyze the TB ecosystem. While the 10 manually selected keywords serve as a starting point, our findings reflect broader underlying campaigns, thereby minimizing keyword-specific bias.

C. TB Reply generator module and role of LLMs in TB replies

In preliminary tests, we examined the TB agent reply generator module by posting cross-lingual Tweets (e.g., non-English text with an English trigger keyword and vice versa). The results indicate that replies consistently match the language of the tweet rather than the keyword. This suggests that the generator likely operates in three stages: (i) identifying TB-related tweets via trigger keyword search, (ii) detecting the tweet's language, and (iii) generating a contextually aligned reply in that language, typically embedding a contact element (e.g., email or mention).

We further evaluated TB agents using two OpenAI detectors (fine-tuned RoBERTa-base and RoBERTa-large) to identify AI-generated replies. As shown in Table IX, the result indicates (i) at least 70% TB agents across campaigns may be likely to use LLM to produce replies, except for the misleading giveaway campaigns, as they produce short text, typically inviting only DMs. (ii) As expected, a larger model (RoBERTa-large) performs better than RoBERTa-base in detecting AI-generated replies.

Campaign	# TBs	# TB Agents (RoBERTa-base-OpenAI)	# TB Agents (RoBERTa-large-OpenAI)
Mal. 1	1201	72.44 % (870)	84.43 % (1041)
Mal. 2	724	78.87 % (571)	82.04 % (594)
Mal. 3	504	19.44 % (98)	22.62 % (114)
Benign	247	70.85 % (175)	77.33 % (191)

TABLE IX: TB agents likely producing AI-generated replies

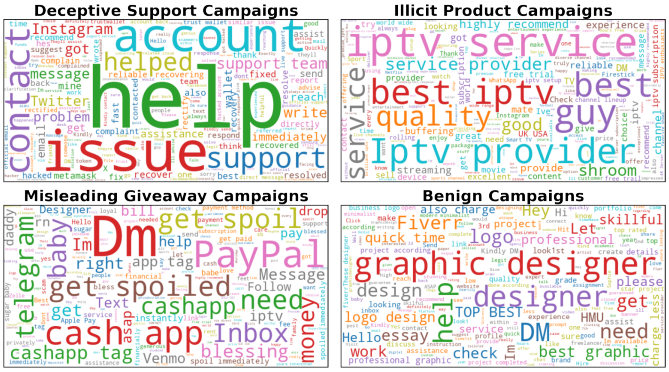

Fig. 10: Word cloud of content of TB agents' replies, see §III-C.

D. TB Agents Replies Word Cloud

Figure 10 presents word clouds generated from different campaigns to highlight content diversity in TB agents' honeytrap replies. These visualizations provide qualitative insight into the distinct content domain to target specific user categories (e.g., cryptocurrency users, psychedelic interest groups, pseudo-romantic victims), as discussed in Section III-A.

E. Additional information on expert labeling process

Annotators were instructed to classify accounts as either bot or human based on two main criteria:

1) *Reply content*: Each account's responses to our honeytrap tweets, along with up to 100 of its most recent replies (including timestamps), were reviewed. All sensitive information was redacted and anonymised.
2) *Behavioural patterns*: Annotators manually evaluated signs of automation such as repetitive phrasing, shared reply templates, and consistent posting times.

Before labeling, annotators were provided with a clear definition of bots. All annotators held a postgraduate qualification

Data Collected	metamask	trustwallet	hacked	IPTV	shrooms	cashapp	sugar daddy	logo	graphic designer	essay
TB Agent Count	606	284	401	625	109	61	456	93	93	78
TB HoneyTrap Replies	1,011	499	509	915	157	76	901	140	147	97
TB-Suspended	425	152	285	353	63	36	307	15	8	12
TB-Active	160	113	94	153	31	16	103	71	80	58
TB-Deleted	15	13	20	119	15	10	52	6	4	8
TB-Changed Screen Name	8	6	7	20	4	3	17	4	2	7
TB Victim Replies	51,004	22,025	32,387	50,624	9,565	5,345	42,657	6,608	5,449	6,178
TB Victim Unique Count	14,777	6,903	10,637	9,255	5,338	4,188	25,089	4,077	2,799	4,651
TB Followers	62,253	30,453	13,441	92,419	9,867	3,415	58,004	33,974	18,534	228,072
TB Friends	11,974	8,255	11,208	247,233	9,339	7,823	81,078	57,264	26,681	100,469

TABLE X: Overview of keyword-based TB agent data collection, (cf. §III-C).

```
def format-data-row-to-campaign-aware-prompt(row):
    return (
        f"User has [[row['followers-count']]] followers,
        [[row['friends-count']]] friends, and has posted
        [[row['statuses-count']]] statuses. "
        f"'The user can be direct messaged.' if
        [[row['can-dm']]] else 'The user cannot be direct
        messaged.' "
        f"The tweet was sent from [[row['source']]]. "
        f"'Description is filled.' if
        [[row['with-description']]] else 'No description
        provided.' "
        f"'Location is filled.' if
        [[row['filled-location']]] else 'Location not
        provided.' "
        f"Tweet frequency:
        [[row['tweet-frequency']:.2f]] tweets/day. "
        f"User has replied: '[[row['reply-text']]]'. "
        f"'User prefers to include ' +
        repr([[row['cue-type']]]) + ' as communication
        method in the reply.' if [[row['cue-type']]] and
        [[row['cue-type']]].lower() != 'empty' else 'User
        does not use any communication method.'"
    )
```

Fig. 11: Campaign-aware Serialization Method

(Master's or PhD) and had prior experience or familiarity with OSN or web security research. The *average inter-annotator agreement* was calculated as the percentage of instances where all three annotators agreed, based on each keyword. Disagreements were resolved by a majority vote, with the final label assigned as bot or human accordingly.

F. Baseline Implementation and LLM Prompt Details

As shown in Table XI, the chosen LLMs (Llama, Phi, and Gemma-family LLMs) are released in 2024 (the same year as our dataset collection), each employing a different model parameter size. Notably, during the feature engineering phase for LLMs, the prompt generator generates and serializes the TB agent features (row-wise) into a campaign-aware[1] natural-language representation. Inspired by the findings of Stefan et al. [50], we adopt a *text template-based* serialization method that serializes feature names and values into natural-language sentences. Simple templates for sentences are preferable as LLM effectively utilizes prior knowledge in the LLM and uses feature names and their relationships to the values for

[1]Campaign-specific information is preserved as embeddings are extracted using RoBERTa on the free-text (raw text) replies posted by TB agents.

```
You are a classification assistant.
### Instruction:
Classify each user as either 'deceptive support
campaigns', 'illicit product campaigns', 'misleading
giveaway campaigns', 'benign campaigns'.

    Definitions of labels:
    - 'deceptive support campaigns': Accounts that
    promote technical support staff deceptive scams.
    - 'illicit product campaigns': Accounts that promote
    fake or illicit product promotions.
    - 'misleading giveaway campaigns': Accounts that
    appear fake sugar daddy and conduct financial scams.
    - 'benign campaigns': Accounts that promote genuine
    graphic designing or assignment completion service.

    Output must be one of these labels and do not change
    spelling or miss words and do not add any
    explanation, punctuation, quotes, brackets or extra
    text.

[Provide either no sample (zero-shot), one sample, three
samples, or five samples of respective `campaigns']
### Input: [[Zero-shot or Few-shot Samples (1, 3 or 5)]]
### Response: [[Label]]

### Input:
[[Queried Sample]]

### Response:
[[Answer]]
```

Fig. 12: Prompt Template for LLM (Zero-shot and Few-shot Setting)

classification [47], [50], [51]. These sentences are inputs that the LLMs will subsequently utilize during the evaluation stage for classification. Please find the serialization method in the Figure 11. When setting the output generation parameters for each LLM, we set *temperature* T=0.0) to generate a deterministic response for a fixed prompt state.

Short Form	Name and Version	Size	Release Month
Llama-3.1	Llama-3.1-8B-Instruct	8B	July 2024
Llama-3.2	Llama-3.2-3B-Instruct	3B	September 2024
Gemma	gemma-7b-bnb-4bit	7B	February 2024
Gemma-2	gemma-2-9b	9B	June 2024
Phi-4	Phi-4	14B	December 2024

TABLE XI: Specific model versions used as part of our experiments. For each model, we define the exact `Version` of the model accessed and the `Release Date` to facilitate fair comparison with traditional models (cf. §V-D).

Task	Method	Number of Shots (K)			
		Zero-shot	1-shot	3-shot	5-shot
Malicious TB	**Llama-3.1**	$0.67_{\pm 0.02}$	$0.52_{\pm 0.02}$	$0.73_{\pm 0.02}$	$0.78_{\pm 0.02}$
	Llama-3.2	$0.69_{\pm 0.03}$	$0.49_{\pm 0.02}$	$0.45_{\pm 0.01}$	$0.68_{\pm 0.03}$
	Gemma	$0.11_{\pm 0.01}$	$0.41_{\pm 0.01}$	$0.79_{\pm 0.02}$	$0.84_{\pm 0.03}$
	Gemma-2	$0.49_{\pm 0.01}$	$0.52_{\pm 0.02}$	$0.73_{\pm 0.02}$	$0.90_{\pm 0.02}$
	Phi-4	$0.59_{\pm 0.02}$	$0.64_{\pm 0.02}$	$0.87_{\pm 0.02}$	$0.87_{\pm 0.02}$
Campaign	**Llama-3.1**	$0.66_{\pm 0.02}$	$0.77_{\pm 0.02}$	$0.90_{\pm 0.01}$	$0.89_{\pm 0.02}$
	Llama-3.2	$0.44_{\pm 0.02}$	$0.63_{\pm 0.02}$	$0.72_{\pm 0.02}$	$0.80_{\pm 0.02}$
	Gemma	$0.27_{\pm 0.02}$	$0.78_{\pm 0.02}$	$0.82_{\pm 0.02}$	$0.52_{\pm 0.02}$
	Gemma-2	$0.28_{\pm 0.01}$	$0.75_{\pm 0.02}$	$0.89_{\pm 0.02}$	$0.53_{\pm 0.02}$
	Phi-4	$0.67_{\pm 0.02}$	$0.86_{\pm 0.01}$	$0.91_{\pm 0.02}$	$0.92_{\pm 0.02}$

TABLE XII: F1 score performance of zero-shot and few-shot LLMs in *malicious TB agent* and *campaign* detection task (cf. §V-E1).

Current LLMs may already have been trained on social-engineering attacks and misinformation campaigns [52], [53], incorporating valuable prior knowledge [46], sparing us the need to run extensive operations such as model training or fine-tuning. Therefore, we randomly select a few TB agent data as held-out training samples for a few (K-shot) in-context learning scenarios. Detailed zero-shot and K-shot (K = 1-shot, 3-shot, and 5-shot) performance is shown in Table XII, and the corresponding zero-shot and few-shot prompts are listed in Figure 12.

We find the following insights from the full results: (i) prior knowledge (zero-shot setting) is better for some models than for others (e.g., Llama models in the first task). (ii) We reported 3-shot results in Table VI in Section V-E1 in this work, as it consistently performed better than 5-shot on many models (such as the Gemma models in the second task). Even for models where 5-shot worked better, we noticed that the reason was primacy bias and recency bias, where the model only outputs the last seen or initial memorized labels–rather than learning context from the data. We even qualitatively test our hypothesis by fine-tuning the instructions by changing the order of the 5-shot samples. We find that the LLM outputs the last seen label in the fine-tuned instruction, confirming our hypothesis. Therefore, 5-shot results are unreliable, as providing more samples increases the context window, hindering the model's capability. Thus, in Table VI, only 3-shot results are provided for all LLM-based detectors, as we found that three feature representations can generalize between distributional drift of the campaign to capture structurally dissimilar keywords within campaigns (refer to Table I in Section §III-A).

G. TB-prone Keyword Discovery and Multilingual TB agents

This section introduces two additional aspects of TB agents. First, incorporating a broader range of TB-prone keywords is valuable. In post-hoc analyses, we identified and validated new trigger keywords using NLP and regex matching. For a systematic (qualitative) exploration, we leveraged original Tweets (from the TB victim) to which TB agents replied and removed those that contained the selected TB-prone keywords and stopwords, and computed word–user frequency counts from the remaining set, indicating how many users (TB victims) used each common word. For example, in the illicit product campaigns, we found new keywords such as `weed`, `adderall`, `DMT`, and `xanax`, all enticing TB agents. Similarly, in the deceptive support campaigns, we found new keywords such as `support`, `recover`, and `contact`. Secondly, we observed certain TB agents posting in multiple languages, highlighting their diverse engagement strategies. The preliminary investigation is illustrated in Fig. 13, highlighting the variety of languages used by these accounts.

Fig. 13: Example of a multilingual TB agent replying in multiple different languages.

H. HoneyTrap Tweets Screenshots

This section provides screenshots showcasing TB agent's replies across campaigns. These figures highlight how responses vary across campaigns, offering insights into content diversity and engagement patterns. Figures 14 show the corresponding screenshots for each campaign, respectively. Please see section III-C for further details.

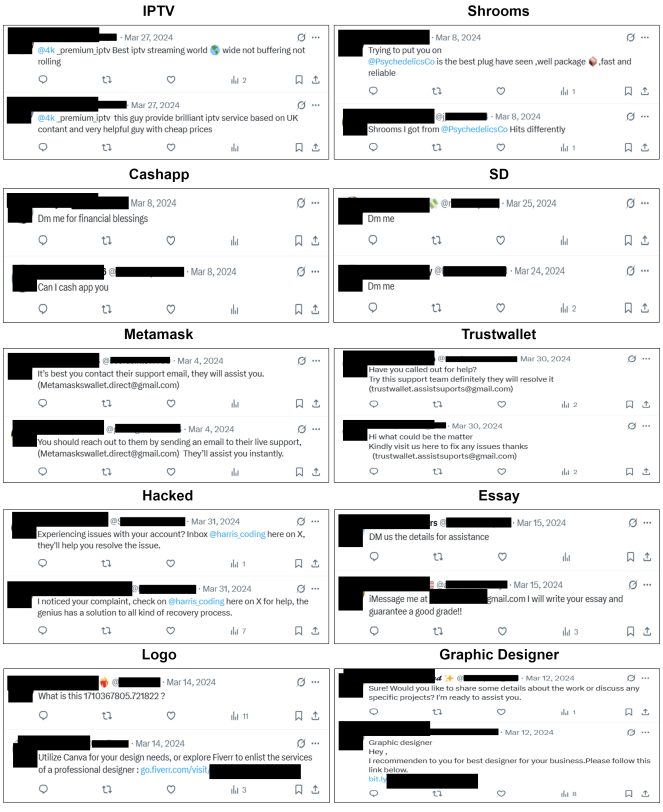

Fig. 14: Examples of malicious and benign TB agents. Interestingly, a benign agent (bottom left) can be seen asking about the UNIX timestamp in our honeytrap tweet.