1 Introduction

Search rank fraud, i.e., the posting of large numbers of fake activities for products hosted in commercial peer-opinion services such as those provided by Google, Apple, Amazon, seeks to give the illusion of grassroots engagement, and boost financial gains [1–4], promote malware [5–9] and even assist censorship efforts [10, 11]. Search rank fraud continues to be a significant problem [2, 12, 13], after years of investment from service providers [14–16] and the academic community (see § 2 for related work).

We posit that one reason for this failure stems from our misunderstanding and underestimation of the capabilities, behaviors, and strategies of the professional raters recruited to perform search rank fraud: existing work is built on assumptions about professional raters, that are either extracted from small datasets of fraud, made based on intuition, or revealed by commercial site insiders. We have recently challenged these assumptions, in qualitative studies that we performed with professional raters that target Google services [17–19]. We found raters who evolved fraud-posting strategies that circumvent and even exploit key assumptions made by fraud detection work (§ 2). This makes some raters particularly successful. For instance, 90% of 1,164 Google Play accounts that 39 professional raters revealed to provably control, were still active one year later.

Figure 1: Photo taken by a participant in a qualitative study we conducted with professional raters [17], with the premises and employees of his business. Photo reproduced with permission.

In this project we envision that knowledge of the authentic capabilities, behaviors and strategies employed by empirically validated raters, will enable us to develop solutions that efficiently manage and contain search rank fraud, by detecting, classifying and neutralizing its effects. To realize this vision however, we need to address several challenges:

• **Fraud diversity**. Fraud detection and classification solutions need to flexibly target diverse types of fraud organizations, behaviors and strategies, such as the ones that we found in preliminary studies [17, 18]. Examples include (1) *federated fraud*, carried out by raters who organize in mostly static teams (see Figure 1 for a photo of a team's brick-and-mortar offices) and post fraud from hundreds of mobile devices and tens of thousands of user accounts that they pool, and (2) *organic fraud*, generated by individual operators with personal accounts and devices, who mix fraud among genuine activities, and form ad-hoc teams.

• **Binary classification is not enough**. The remarkable success of fraud suggests that the current, binary classification of activities, e.g., fake vs. honest reviews, fraudulent vs. genuine accounts, followed by the removal of detected fraud, fails to stop prolific federated raters, who can easily create new accounts and post new fraud. Further, the decentralized nature of organic fraud enables it to elegantly evade status quo assumptions, e.g., that fraud produces synchronized, lockstep behaviors or suspicious activity spikes.

• **Training and evaluation of developed solutions**. Commercial platforms are close-sourced, and their Terms of Service (ToS) prohibit posting fraudulent activities. However, fraud detection and classification solutions need to be trained using large sets of ground truth data, and, importantly, need to be evaluated in production-like environments, under real-time fraud posting conditions.

Project Contributions. We build this project on the thesis that to be effective, fraud detection and classification efforts need to *involve the organizations and individuals who contribute to search rank fraud*. Therefore, we organize the project in research modules that engage with professional raters to (1) collect ground truth knowledge and evaluate defenses, (2) develop fraud detection and classification solutions that adapt to rater strategy changes, and (3) attribute fraud to the organizations that posted it.

More specifically, in Research Module A we leverage our finding that the behavior patterns of professional raters extend beyond the sites that they target, to notably include their use of mobile devices and user

accounts. To address the first of the above challenges, we posit that such behavior patterns can be used to classify fraud according to the type of rater posting it. We will build RacketStore, an app market-inspired site and mobile app platform, to collect detailed ground truth behavior data from professional raters and honest users.

In Research Module B we build on results from Module A to develop and evaluate the first solutions that disentangle organic from federated fraud, and from honest behaviors. We will build realistic adversary models that emulate observed and reported rater behaviors and constraints, and successfully circumvent existing fraud detection solutions. We leverage the novel conjecture (based on preliminary data) that the operation constraints of federated and organic raters impose distinguishable patterns of use of the devices that they control, to develop an adversarial learning process that (1) trains a discriminator network to identify fraudulent online activities, and classify them according to the characteristics of the raters who posted them, and (2) iteratively improves a fraud generation network to produce synthetic fraud that is hard to distinguish from honest activities.

To address the second challenge, in Research Module C we complement and extend Module B to develop fraud de-anonymization solutions that further classify detected federated fraud, by attributing it to the organizations responsible for posting it (e.g., see Figure 1), and enable us to identify the resourceful and influential rater federations.

We will further use RacketStore to address the third challenge: recruit professional raters to install and use RacketStore, thus evaluate developed solutions online with real raters, and monitor the evolution of their behaviors and strategies. In addition, we will leverage RacketStore to propose a departure from the standard retaliation approach (e.g., closing of accounts, removal of reviews), to instead develop solutions that neutralize the effects of search rank fraud on its intended victims, i.e. the users.

1.1 Intellectual Merit

In this project we develop solutions to study, detect and prevent fraud in online services, and techniques and platforms to evaluate developed solutions. We introduce the following novel research contributions:

RacketStore: Evaluation and fraud profiling platform. Introduce the hypothesis that professional rater behaviors can be used to identify fraud and classify it according to the rater type. Build the first platform, techniques and protocols, to (1) collect data about the behaviors of professional raters and honest users in peer-opinion sites from their mobile devices, and (2) evaluate developed solutions online, using live raters, with proven expertise in search rank fraud.

Adversarial learning based fraud classification. Model fraud generation as a constraint optimization problem, and introduce *activity sequences*, timelines of constraint-satisfying actions. Develop techniques to extract activity sequence *embeddings*, and develop a deep, adversarial learning approach to iteratively train convolutional and recurrent neural networks, to disentangle organic and federated fraud from honest behaviors. Develop fraud generators to produce benchmark datasets of synthetic, fraudulent activity sequences that defeat state-of-the-art defenses.

Fraud de-anonymization. Introduce *fraud de-anonymization* and *pseudonymous rater discovery* problems. Leverage identified rater behavior patterns, the unique opportunity provided by the detailed, device-level data collected through RacketStore, and network representation learning techniques, to develop predictors that attribute fraud, and identify accounts and devices controlled by the same organization.

Fraud vaccines. Develop RacketStore-based solutions to neutralize the effects of search rank fraud, by *nudging* users toward making safer decisions when acquiring products with peer-opinion feedback.

2 Related Work

Fraud Detection and Adversary Assumptions. State-of-the-art research on detecting peer-opinion fraud uses machine learning to classify fake vs. honest reviews [20–36], and fraudulent vs. genuine accounts [36–45]. Solutions are built on key assumptions about adversarial behaviors and capabilities, which include (i)

bursty activity [22, 26, 27, 34], i.e., that raters post reviews in quick, suspicious sequences, (ii) lockstep behaviors [31–33, 46], i.e., that raters synchronize the user of their accounts when posting reviews, (iii) review plagiarism [20, 21, 24, 28] and distinguishability of machine vs. human generated reviews [47], i.e., that due to human limitations, raters either copy-and-paste their reviews or use review generators, (iv) extreme reviews and deviation [28–30, 35], i.e., that raters seek to minimize their work thus post only extreme ratings, and (v) ratio of singleton accounts [34–36], i.e., that a promoted product often receives many reviews from accounts specifically created for this task. However, in previous work [17], we found that professional raters have evolved strategies to bypass defenses built based on these assumptions. Instead, in this project we will develop fraud detection and prevention solutions that are consistent with the capabilities, behaviors and strategies reported and inferred from real professional raters.

Modeling Online Fraud. Previous work has studied the creation of fraud in a variety of online services. For instance, in Twitter, Thomas et al. [48, 49] investigated fraudulent account markets to monitor prices, availability, and perpetrated fraud. They also identified suspended accounts, and studied the behavior and lifetime of spam accounts, and the campaigns they execute. Stringhini et al. [50] studied follower markets by purchasing followers from different merchants, and discovered patterns and detected market-controlled accounts in the wild. In Facebook, De Cristofaro et al. [51] studied page "likes" performed by fraudster "farms" using honeypot pages, and analyzed temporal, social and demographic characteristics of the likers. Critical operational details of fraud markets have however remained mostly unstudied. In this project we will instead directly engage and seek insights from professional raters, and use them to build realistic models of fraud, synthesize evaluation data and develop next generation fraud detection and prevention techniques.

Other similar studies have different goals. To highlight the methods and prevalence of scammers, specific to Nigeria, Park et al. [52] collected three months of data using an automated system which posts honeypot ads on Craigslist, and interacted with scammers. For instance, Portnoff et al. [53] used NLP and ML-based methods to determine post type, product and price on cybercriminal market offerings. Further, Wang et al. [54] used empirical crawled data to identify SEO campaigns and documented their impact on promoting search results for several luxury brands. In contrast, the protocols that we will design to interact with raters will seek to identify online service vulnerabilities that raters exploit, their strategies to avoid detection, and their intrinsic weaknesses, to be exploited by the next generation of fraud detection solutions.

Collecting Training Data. Previous work has used crowdsourcing to recruit raters to write reviews for existing venues [55, 56], purchase Twitter followers from specialized markets [50], or deploy Facebook honeypot pages to collect fake likes [57]. Such an approach raises ethical and ToS concerns, as peer-opinion sites forbid the creation of fraud. Instead, Yang et al. [58] collected Twitter spammers who posted links to phishing and malware sites, while Seneviratne et al. [59] collected apps removed by app markets due to being spam. Such techniques do not provide ground truth assurances, since they build on solutions with inherent false positive rates. Further, we are not aware of sustained academic efforts to identify and collect ground truth honest activities and accounts. Others, e.g., [60, 61], use datasets of fraudulent and honest activities revealed by commercial services. The well-documented failure of commercial services to prevent fraud casts doubts on the quality of such datasets. Instead, in this project we will build RacketStore, an app market platform that will enable us to ethically collect fraudulent and honest data, and evaluate fraud detection solutions that we develop, in a live environment, with real professional raters.

3 The Model

System Model. We consider general, online services with *peer-opinion* functions, that host accounts for products, their developers, and users. Such systems include app markets (e.g., Google Play, Apple Store), crowdsourced review forums (e.g., Google Maps, Amazon, Yelp, TripAdvisor), and social networks (e.g., Facebook, Twitter). Developers use their accounts to upload information about their products, e.g., apps, pages, physical goods, or venues such as restaurants. Users can interact with the product from a registered device through an *activity*, e.g., install, view, review, like. Certain activities, e.g., reviews or likes, are

expected to be performed by users only for products they have previously installed, used, viewed or visited. Most online services provide an *account validation* functionality, where they request the account owner to prove control of a mobile phone, e.g., by providing its calling number, then retrieving a code sent to it through SMS.

Search Rank. Products with higher *engagement*, e.g., that receive more positive reviews, installs, views, or likes, achieve a higher search rank, become more influential, and are acquired more frequently and generate more revenue, either through direct payments, ads or impact on public opinion. In Yelp for instance, it was shown that a one star boost in rating, helps restaurants increase revenue by a 5-9% margin [62, 63]; in Facebook, the top 20 fake news stories about the 2016 elections received more engagement than the top 20 real election stories from major media outlets [64].

Adversary Model. Adversarial product developers hire *professional raters* (or raters), i.e., specialized organizations and/or individuals, to perform *search rank fraud* campaigns, i.e., promote their products by posting many stellar reviews and ratings, and create the illusion of grass-roots engagement with their products.

We build on knowledge we acquired in previous investigations [17, 19, 65], to consider a diverse ecosystem of often ingenious professional raters, and avoid making strong, restrictive assumptions about their organization structures, capabilities, skills and strategies. For instance, we found that many raters control multiple accounts (also known as sockpuppets [38–44, 66]), ranging from only a few to thousands [17]. We further found raters that are federated, organic, and hybrid. *Federated raters* organize in static teams, with hierarchies and sometimes even brick-and-mortar offices (see Figure 1), and pool resources such as accounts and devices. *Organic raters* are lone individuals who use their own devices and accounts to post commissioned reviews. We have infiltrated 16 groups hosted in Facebook (with a total of 86,717 members), that are used by professional raters to organize and communicate. We found substantial evidence of flexible, *hybrid rater organizations*: federated raters further crowdsource their search rank fraud work, by broadcasting job details on such groups; organic raters collaborate through *exchange reviews*, by committing to write the same number of reviews for the products promoted by others.

4 Research Plan

We introduce and develop fraud management solutions that target validated fraud posting strategies of professional raters. We organize this project into 3 main modules, illustrated in Figure 2: In Research Module A we develop the RacketStore platform, to collect detailed, ground truth behavior data from professional raters and honest users, and provide an evaluation environment for the fraud detection solutions that we develop in Research Module B and the fraud attribution solutions that we develop in Research Module C. We employ an iterative, adversarial-training approach that (1) integrates fraud posting strategies discovered in Module A into the solutions of Module B and C and (2) expands ground truth datasets from evaluation efforts of developed solutions onto the platform of Module A. In the following, we detail our plans for each module.

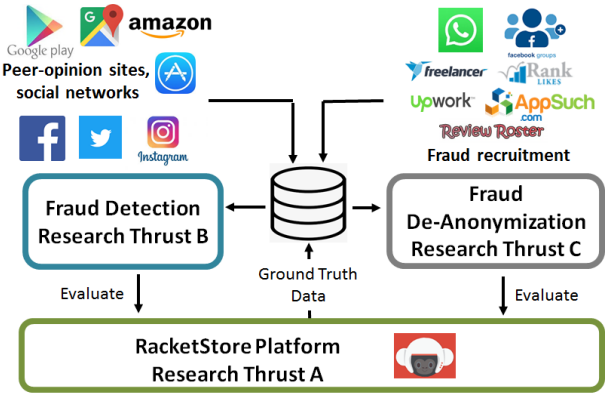

Figure 2: Research plan. Research Modules B and C develop fraud detection and classification solutions for sites with peer-opinion functionality. Research Module A builds RacketStore, the platform to collect ground truth data and evaluate developed solutions.

4.1 Research Module A: Build a Training and Evaluation Platform

We will build a platform to collect training data and evaluate developed solutions.

The Problems and Preliminary Results. The current academic approach to train fraud detectors, is to use data available from commercial peer-opinion sites. However, the documented inability of commercial

peer-opinion sites to contain fraud [1–3, 5–11] suggests that such data is not sufficient. For instance, in previous work on Google Play [17] we found that 90% of 1,164 accounts we verified to be controlled by 39 professional raters, were still active one year after the raters revealed them.

Further, we currently have an *inaccurate and incomplete understanding of fraud*: Most fraud detection solutions are built on a few key assumptions about adversarial behaviors and capabilities (§ 2), that are either based on intuition, extracted from small datasets of fraud, or that have been revealed by collaborators within commercial sites, and need to be taken on faith. In a qualitative study [17] with recruited professional raters, through semi-structured interviews consisting of 116 questions about fraud-posting capabilities and strategies in Google Play, we found participant-revealed behaviors that circumvent key assumptions, e.g., lockstep behaviors [22, 29, 32–34, 67–71], suspicious activity spikes [20–22, 24, 26, 28, 31, 34, 51, 69, 72–81]) (§ 3). We also found and validated participant-revealed techniques to bypass Google-imposed verifications, strategies to avoid detection and even leverage fraud detection to enhance fraud efficacy.

In addition, while good data is essential to any machine learning task [82–84], we do not have efficient solutions to collect feature-rich ground truth data about the behaviors of both professional raters and honest users in peer-opinion sites. In previous work we have collected ground truth fraud data [17, 18, 65, 85, 86], however, not honest user activities. Further, data currently available for collection, and most existing fraud datasets, consist only of the end-result of user activities, i.e., (fake) reviews or (sockpuppet) accounts, and do not include background information on the users and their activities, e.g., the device used to post reviews or how the device was used prior to posting an activity.

Lastly, but importantly, we are not aware of any existing platform that can be used to evaluate and compare developed fraud detection and classification solutions under the daily operating conditions encountered by commercial peer-opinion sites. Such an evaluation and comparison is vital to the development of new fraud management solutions.

Approach Overview, Intuition and Novelty. In this module we will build RacketStore, the first app market-inspired platform dedicated to collecting detailed information about the activities of honest users and professional raters, in particular *from the devices that they use to access peer-opinion sites*. The intuition behind this effort is that the fraud-posting constraints imposed on various types of professional raters are likely to result in distinguishable patterns of device use. For instance, we expect and will investigate that (1) federated raters use the devices that they pool, mostly to post fraudulent activities, (2) honest users use their devices solely for personal purposes, while (3) organic raters use them to perform a mix of personal and fraudulent activities. We will investigate whether this will impose observable differences between honest users, organic raters and federated organizations, in terms of, e.g., the apps that they have installed on their devices, and the patterns of their app and device usage.

A second fundamental goal of RacketStore, is to be the first online evaluation platform for fraud detection and classification solutions, with fraud posted in real-time by recruited raters (see § 4.2.3 and § 4.3.2).

4.1.1 Details of Proposed Work

Similar to commercial peer-opinion services, RacketStore will consist of an online site and a mobile app. The site will implement the basic functionality of a commercial system, e.g., an app market like Google Play (see § 3), but will not host real products and apps, including executables (i.e., apks), thus cannot be used to distribute malware or misinformation. Instead, we will use existing tools [87, 88] to invent new product concepts, names and logos. An early prototype of the RacketStore site is available [89].

We will further build the RacketStore *mobile app*, to be installed on the mobile devices of users, and be a portal to the RacketStore site. The RacketStore app will periodically collect snapshots of device use details, e.g., (1) the apps installed, (2) the currently used app, (3) the accounts logged in, and (4) the device status, e.g., battery level, available memory, screen on/off, sleep mode activation, SIM card in, etc. Early experiments with a preliminary app version, suggest that we can collect the currently used app once every few seconds, while keeping the compressed, daily collected data under 400Kb. We will experiment with

a broad range of mobile devices to ensure minimal impact of RacketStore on device operations, including CPU, battery, bandwidth and required permissions. We will seek participant feedback to ensure that they are comfortable with the required permissions and keeping RacketStore installed for several weeks.

We will recruit professional raters through specialized groups in Facebook, Whatsapp and Telegram as we have done in [17], and also from crowdsourcing sites that specialize in fraud, e.g., [90–95], as we have done in [19, 65, 85, 86]. In a pilot study, we contacted 16 members in the Facebook groups that we infiltrated (see § 3), and 8 agreed to participate and installed the alpha version RacketStore app. This, together with the high number of members in these groups (total of 86,717, § 3) provides early evidence for our ability to recruit professional raters. Further, we will develop protocols to recruit and distinguish ground truth honest users, who use peer-opinion sites but have never been paid to promote products. For instance, to access a broad demographic segment, we will advertise our study using ads on sites like Instagram. We will develop and include a short questionnaire into the RacketStore app, to determine if participants satisfy the above conditions (e.g., education background, sites used, participation in campaigns, including white-hat ones). We will deliver the questionnaire at the end of the study, to reduce cognitive bias.

We will develop features that capture if honest users and raters differ in the types of apps they install (e.g., promoted apps vs. malware vs. apps that they actually use), the duration for which they keep them installed and how they interact with them. In our previous work [17], all raters claimed to interact with an app before promoting it, and perform some form of *retention installs*, i.e., keep the app installed after reviewing it. Further, one of the above pilot study participants, had more than 10 browser apps installed on his device. We will also explore whether the number of accounts logged in on a device differentiates honest users and organic raters from federated raters. In our preliminary study [17], federated raters claimed that they login in up to 5 accounts on any device that they control. However, in the above pilot study we found raters who had up to 40 Gmail accounts logged in on a single device. We will design and conduct semi-structured interviews with participants, after the completion of the data collection process, to help us interpret the collected data and associate it with specific user behaviors and strategies.

In Task B.2 (§ 4.2) we detail our plan to use the collected information to train supervised learning models to detect and classify fraud. Further, in § 4.2.3 and § 4.3.2 we describe plans to use RacketStore to evaluate and compare developed fraud detection and de-anonymization solutions, while in § 5 we discuss plans to use RacketStore to neutralize the effects of fraud.

4.1.2 Ethical Considerations

We will follow the best ethical practices for conducting sensitive research with vulnerable populations [96]. We will clearly declare our identity, research objective, and potential impact on the participant work without following any sort of deception. We have IRB approval for most studies that we will conduct in this project.

RacketStore will not host real products, thus any reviews or ratings posted by participants will not impact users. To prevent non-consenting users from using the RacketStore app, the app will ask the user upon startup, to type a unique, 6-digit code, sent only to recruited participants. Consistent with Google and IRB policies, the RacketStore app will present to the user a list of permissions that we request, the data that we collect and its intended use. The user will need to consent and grant permissions, in order for the app to start collecting data. We will use GDPR [97] and NIST [98] recommended pseudonymisation for data processing and statistics, and other generally accepted good practices for privacy preservation.

4.2 Research Module B: Fraud Detection and Classification

The Problem and Preliminary Results. In previous work [85, 86] we have developed fraud detection techniques that correlate detected review relations with linguistic and behavioral signals extracted from longitudinal Google Play app data that we collected [99]. However, in [17] we have shown that some raters have evolved behaviors that evade detection. Notably, we have documented the popularity of organic fraud, which, due to its asynchronous nature, is much harder to detect than federated and inorganic fraud. For

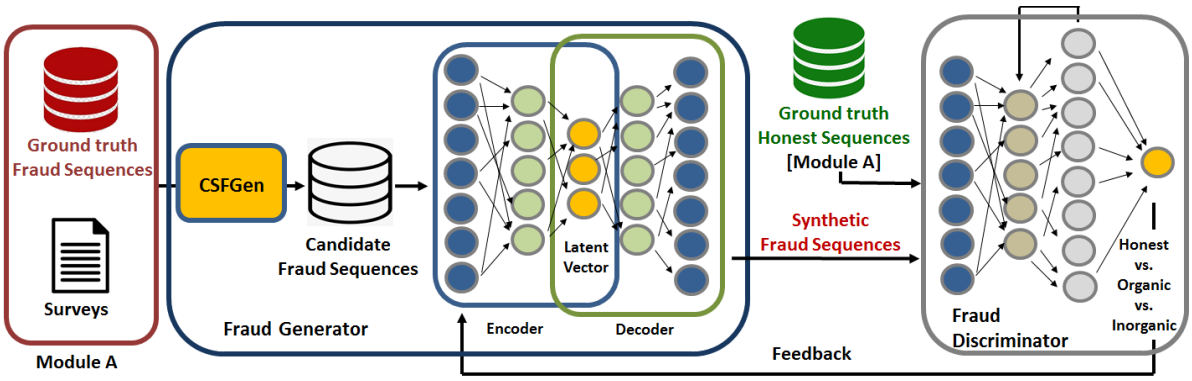

Figure 3: Fraud detection: adversarial learning approach. The constraint-satisfying fraud generation (CSFGen) block of Task B.1 produces synthetic fraud sequences that satisfy requirements identified in Module A. These sequences help train FraudGen, an autoencoder, to efficiently generate realistic, hard-to-detect fraud sequences. We use FraudGen sequences and ground truth honest sequences (Module A) to train the fraud discriminator network (FDNet, Task B.2), then iterate to use FDNet to improve FraudGen.

instance, organic raters are less likely to exhibit lockstep behaviors, and the products that they target are less likely to exhibit suspicious activity spikes. Fraud posted from organic accounts and devices is further camouflaged among, and needs to be disentangled from real, honest activities of the rater.

Thus, the goal of this module is to develop solutions to detect and classify fraud-posting behaviors, that remain effective with the evolution in rater behaviors and strategies. One challenge in building accurate fraud classifiers is that we do not have enough data, which is essential for any machine learning task [82–84]. Alone, the ground truth data that we collect in Module A will not be sufficient to train complex models.

Overview of Approach and Novelty. We leverage review-posting constraints reported by professional raters and timelines of rater behaviors collected from RacketStore, to build realistic adversary models. We will use these models to generate large sets of synthetic fraud data. Given our need to design detection solutions that go beyond state-of-the-art fraud, we seek to generate synthetic data that is (1) realistic, i.e., emulates observed behaviors of validated raters, (2) satisfies requirements reported by raters in semi-structured interviews [17], and (3) is not detected by existing defenses. We will share generated fraud data with the research community, as a benchmark to compare newly developed fraud detection solutions.

We will use adversarial learning [100–102] to pit the developed adversary models against fraud detection networks, and iteratively refine and improve both. Figure 3 outlines our proposed approach, a minmax game between two main components: FraudGen, a generator that mimics professional rater activities, and FDNet, a fraud discriminator network. The following tasks describe plans to develop these components.

4.2.1 Task B.1: FraudGen: The Fraud Generator

Previous work [47, 103] has proposed deep learning-based generative solutions for fabricating review text. In this task we recognize that the review text creation process is only the tip of the iceberg. We propose instead a holistic generative fraud approach that considers all aspects of the fraud creation process, thus a spectrum of constraints encountered by genuine professional raters when performing search rank fraud activities. For this, we introduce and use the concept of *activity sequences*, timelines of activities performed by a user, e.g., opening a user account, registering a mobile device, validating the account, product acquisition (e.g., installing, uninstalling or purchasing a product), reviewing the product.

CSFGen: Constraint Satisfaction Fraud Generator. We will first use data collected and rater strategies discovered in Module A (§ 4.1) to develop optimization problems that model the real-life constraints of raters in their daily fraud-posting routine (e.g., job specs, finite resources, detection avoidance), and their quest to maximize financial gain. For instance, we define a *device management optimization problem* for a professional rater who controls n devices and needs to simultaneously promote m products. Given x_{ij},

the number of accounts from device i used to review app j, R_j the number of reviews to be posted for product j, and $p_{i,j}$ the payment to be received after successfully posting a review from device i to product j, the problem will seek to maximize gain $\sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}x_{ij}$, subject to constraints such as (1) post all required reviews: $\sum_{i=1}^{n} x_{ij} = R_j \; j = 1..m$, and (2) post at most t_i reviews from device i to any product: $0 \leq x_{ij} \leq t_i, i = 1..n, j = 1..m$).

In addition, we define a *lockstep behavior avoidance optimization problem* that seeks to optimize the rater use of his controlled user accounts when targeting a product. Let W_{ij} be the number of products reviewed in common by two accounts $i, j \; i \neq j$. Let u_i be decision variables for account use, that is 1 if user account i is used to promote the product (0 otherwise), and let y_k be decision variables that model new account creation, i.e., y_k is 1 if account k is created to post review (0 otherwise). Since raters reported different costs for posting reviews from old accounts (p) vs. from new accounts (q), we seek to maximize profit $p_i \sum_i u_i + q_k \sum_k y_k$ subject to constraints such as (1) previous lockstep of any two accounts i and j used to review the product should be below a threshold: $W_{ij}u_iu_j \leq t, i, j = 1..n, i \neq j$, (2) all purchased reviews should be posted from either old or new accounts: $\sum_i u_i + \sum_k y_k = R$, and (3) using only the client-imposed number of old and new accounts: $l_x \leq \sum_i u_i \leq t_u, l_y \leq \sum_k y_k \leq t_y$.

We will use CSP solvers, e.g., [104–107] to generate *fraud sequences*, i.e., activity sequences that a professional rater needs to perform to satisfy constraints, e.g., purchase a new device, open a new user account, install or purchase a product, post a review.

FraudGen: Deep Learning Fraud Generator. In previous work [17] we confirmed that raters adjust their strategies to avoid detection. Thus, a static, CSP-based approach to model fraud does not allow us to model the evolution of fraud, after new defenses are deployed. To enable the exploration of such fraud strategy adjustments, we will build FraudGen, a DNN-based fraud generator, to synthesize fraud sequences with two objectives: satisfy the constraints defined above, and be labeled as honest by the fraud discriminator of Task B.2. For instance, we will use variational autoencoder [108] networks, trained using synthetic fraud sequences generated by the above CSP-based generator. Thus, we will train FraudGen encoder layers to crystallize the above constraints into a latent vector, such that random latent vectors passed through trained decoder layers, will produce fresh fraud sequences that satisfy the above constraints. In preliminary work [109, 110], we used an autoencoder to extract a core of invariant features from images, and provide a camera-based authentication using personal objects. In this project, to train the encoder and decoder layers of FraudGen, we will define cost functions that include a reconstruction loss that models the number and importance of constraints that are not satisfied by fraud sequences reconstructed by the decoder.

Activity Embeddings. To convert activity sequences into values that can be used as inputs by neural networks, we leverage the similarity of the above activity sequences to natural language to generate *activity embeddings* [111, 112], that capture the characteristics of the neighbors of an activity and/or word. We will pre-train word embeddings by optimizing an auxiliary objective (e.g., predicting a word based on its context) in a large unlabeled corpus of activity sequences consisting of the above CSFGen-generated fraud sequences and ground truth honest activity sequences (Module A).

4.2.2 Task B.2. FDNet: The Fraud Discriminator

Manual Feature Investigation. We will use the mobile device-level data that we collect in Module A, to identify features that model differences between the various types of professional raters and honest users. Examples include the *app rotation frequency*, i.e., how often apps are installed/uninstalled, the number of installed apps that are flagged as suspicious (e.g., VirusTotal, Google SafetyNet API [113]), the number of installed apps that the user has reviewed, and statistics over the times that the apps were kept installed, before and after they were reviewed. Other feature sources include the accounts simultaneously logged in on a device, the timelines of the device being idle, in power save mode, or with the screen off, battery and available memory, and review typing style of the user (e.g., cut-and-paste vs. manual typing vs. mix).

We will use these features to identify fundamental conflicts in fraud posting strategies, and inform

and evaluate fraud detection and classification solutions that we develop in this project. For instance, we conjecture (and will verify) that the time that an honest user spends on apps, has a long tail distribution (i.e., a few apps are used much more frequently than most others). This conflicts with product developer requirements that hired raters should increase (and make uniform) their time to interact with the apps they promote.

FDNet: DNN-Based Fraud Discriminator. We will investigate the ability of the extracted features, to train standard supervised learning algorithms to differentiate between fraud posted by organic and federated raters, and from honest activities. However, a static discriminator will become obsolete with evolution in professional rater strategies. To address this problem, we will build FDNet, a deep neural network-based fraud detection approach, that can be adversarially trained as described above, to adapt to rater changes.

We will first explore a convolutional neural network (CNN) based FDNet, that takes as input the above activity embeddings pre-trained on large unlabeled corpora (see Task B.1) and outputs probabilities that the sequence is honest, or contains fraud. The intuition behind our exploration of CNNs for activity sequences, are location invariance and compositionality: the same constraints apply at various locations in an activity sequence. FDNet's convolutions will "slide" over contiguous activity sub-sequences. To address the difficulty of training from scratch all the parameters of a CNN, we will leverage transfer learning [114, 114, 115] to reuse layers and weights from networks performing similar tasks. For instance, we will use the first layers of CNNs dedicated to NLP tasks such as sentiment, subjectivity and question type classification [116, 117]. To address the fixed-sized input restriction of CNNs we will further investigate recurrent neural networks (RNN), including LSTM [118, 119] and GRU [120] networks, that handle input (i.e., activity sequence embeddings) of arbitrary length. The sequential processing of RNNs is further suitable to capture the inherent sequential nature of activity sequences, where units are words or even entire activities, and develop their semantic meaning based on previously processed units.

We will further investigate if existing fraud detection solutions (§ 3) are able to flag activities in FraudGen sequences, and use successful solutions as a secondary discriminator, to further train FraudGen.

4.2.3 Evaluation Plan

In addition to cross-validation experiments using data collected in Module A and generated in Task B.1, we will also implement the developed fraud detection solutions in the RacketStore platform (§ 4.1), and evaluate them online, with professional raters recruited following the protocols developed in Module A. We will use standard metrics (precision, recall, F1-measure) to evaluate performance.

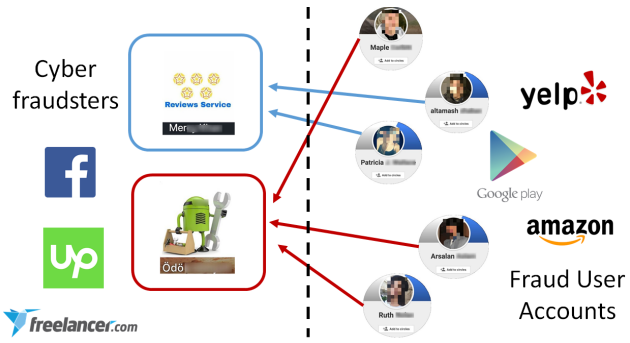

Figure 4: Fraud de-anonymization illustration. Fraud detection only identifies suspicious user accounts on the right. Fraud de-anonymization also finds the crowdsourcing account (left side) that controls them. Arrows signify control.

We will design online experiments where we monitor recruited raters in order to observe their approach to bypass fraud detection solutions. For instance, we will ask a subset of recruited raters to post *persistent* reviews, that are not filtered by the RacketStore site for a certain number of days. In addition, we will ask another subset of raters to post reviews that keep the product's average rating above a specified value. Both job types are standard for professional raters. We will then document the strategies employed by the participant raters, to re-post reviews after our fraud detector filters them. We expect strategies that include (1) raters re-posting their reviews from different accounts and/or devices, (2) changing the review text and/or rating, and (3) waiting longer before posting subsequent reviews. We will use observed behaviors (and collected data) to inform the FraudGen model that we construct in Task B.1, thus keep our defenses relevant to newly observed adversary behaviors.

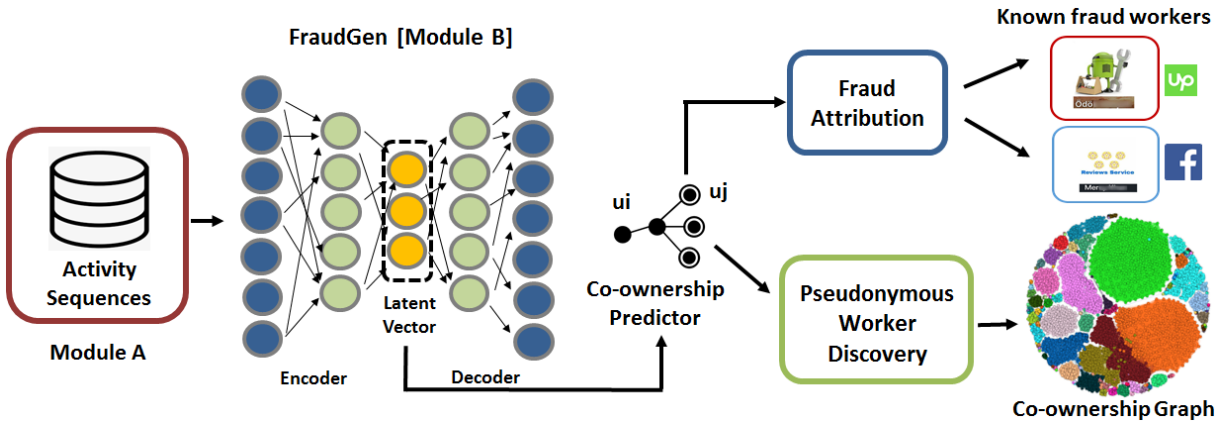

Figure 5: Fraud de-anonymization and pseudonymous rater discovery: Develop a *co-ownership predictor* with features extracted in Module B, and build (1) a *fraud attribution* engine to assign fraudulent accounts to known raters, and (2) a *co-ownership graph*, to discover new groups of rater-controlled accounts.

We discuss ethical aspects of our experiments, in § 4.1.2.

4.3 Research Module C: Attribution of Federated Fraud

The Problem, Approach Overview, Novelty and Preliminary Results. The solutions that we develop in Module B will classify resources (e.g., accounts, devices, reviews) as honest, organic or federated, but will not be able to detect the large fraud federations, who control many resources. To address this, we introduce the *fraud de-anonymization problem*, illustrated in Figure 4: Given a user account or device detected in Module B to be controlled by a rater, determine which of a set of previously identified professional raters actually controls them. Since the set of known raters will be only a subset of all professional raters, we further define the *pseudonymous rater discovery* problem: Given a set of suspected fraud-promoting accounts and/or devices that could not be attributed to any of the known raters, identify new groups of accounts and/or devices, such that each group is controlled by a different organization.

We are the first to propose to attribute discovered fraud to raters, and discover the federated raters responsible for posting large amounts of fraud. In early work [19] we conjectured that professional raters have a unique writing style that we can use to attribute fraud. This however was effective only for a subset of the raters; in later work [17] we found that this assumption does not always hold, due to plagiarism, the organizational structure of federated raters, and specific developer requirements. Instead, in this project we leverage the unique device-and-site usage data that we collect in Module A (§ 4.1) to build a function (illustrated in Figure 5) that predicts if two devices and/or accounts are controlled by the same organization, then use the function to attribute discovered fraud, or group it by source.

4.3.1 Proposed Work

Probabilistic review-posting model. We consider a probabilistic review-posting model from devices and accounts controlled by raters, inspired by Su et al. [121], and validated using data that we collected from professional raters [17, 18]. For instance, given $\mathcal{U}$ the set of accounts and $\mathcal{S}$ the set of products hosted in the online system, we assume that an account u controlled by a rater, is likely to review subjects in a pairwise-disjoint family of sets over $\mathcal{S}$, $\mathcal{F}_W = \{\Omega_1, \Omega_2, \ldots, \Omega_m\}$ with different multiplicative factors $r_1, r_2, \ldots, r_m$ describing u's responsiveness to each Ω_i. Ω_i denotes a subset of products with a common set of features. We assume that the review history of an account is described by a sequence of independent and identically distributed random variables $R_1, R_2, \ldots, R_n$ where $R_k \in \mathcal{S}$ represents the k-th subject reviewed from the account. Therefore, an account's review posting behavior is characterized by $\mathcal{F}_W$ and r_i for all $i = 1 \ldots m$.

Let $\{p_j\}$ be a probability measure over the sample space $\mathcal{S}$, related to the popularity of the subjects: $p_j \geq 0, \sum_{j=1}^{|\mathcal{S}|} p_j = 1$. For any fraudster profile $(W, U, S) \in W^*$, we define random variable $R_k(\mathcal{F}_\mathbf{W}, \mathbf{r})$

with values in $\mathcal{S}$ and with the probability distribution $\mathbb{P}(R_k = s_j) = \frac{r_q p_j}{c}$ if $s_j \in \Omega_q$, for all $q = 1..m$, and $\mathbb{P}(R_k = s_j) = \frac{p_j}{c}$ if $s_j \in \bigcap_{i=1}^{m} \Omega_i^C$. $c = \sum_{i=1}^{m} r_i \sum_{s_j \in \Omega_i} p_j + \sum_{s_j \in \bigcap_{i=1}^{m} \Omega_i^C} p_j$ and $\mathbf{r} = [r_1, \ldots, r_m]^\intercal$ is the vector of multiplicative factors. Specifically, we assume that the probability that the k-th review targets subject s_j is higher, i.e., proportional to factor r_m, if subject s_j satisfies Ω_m's membership properties. Otherwise, this probability is simply given by the ratio p_j/c. Then, let $R_1(\mathcal{F}_{\mathbf{W}}, \mathbf{r})$, $R_2(\mathcal{F}_{\mathbf{W}}, \mathbf{r})$, ..., $R_n(\mathcal{F}_{\mathbf{W}}, \mathbf{r})$, be a review history suspected to be fraudulent. To build a solution for the fraud de-anonymization problem, given a set of candidate raters, each described by a family of sets $\mathcal{F}_{\mathbf{W}}$, we will derive the maximum likelihood estimates $\hat{\mathbf{r}}$ and $\hat{\mathcal{F}_{\mathbf{W}}}$ of the function $\mathcal{L}(\mathcal{F}_{\mathbf{W}}, \mathbf{r}) = \left(\prod_{i=1}^{m} \prod_{R_k \in \Omega_i} \mathbb{P}(R_k \mid \mathcal{F}_{\mathbf{W}}, \mathbf{r}) \right) \prod_{R_k \in \bigcap_{i=1}^{m} \Omega_i^C} \mathbb{P}(R_k \mid \mathcal{F}_{\mathbf{W}}, \mathbf{r})$, where $\hat{\mathcal{F}_{\mathbf{W}}}$ is the family of sets associated with the rater most likely associated with the given review history.

Co-activity Relationships. To define the Ω_i sets of the above MLE-based de-anonymization approach, we will explore and develop features that model the similarity of devices, accounts and products. For instance, we will use the device-and-site data collected in Module A to develop features that model common activity (co-activity) relationships between any two user accounts (or devices) u_i and u_j. Example features include (1) the similarity of the devices, e.g., as statistics over the Jaccard distance of the sets of apps installed and/or used on these devices, over time, (2) the similarity of the accounts, e.g., the products that they have reviewed in common, the order in which they reviewed them, their *inter-review times*, and the similarity of the devices on which they were logged in over time, and (3) the similarity of their reviews, e.g., in terms of text and star rating. While valuable for providing a human-interpretable intuition of similarity, a manual choice of features may also generate blind-spots that can be exploited by raters. Instead, we propose to further leverage the FraudGen autoencoder of Task B.1 (§ 4.2.1, see Figure 5), and its latent vector that contains the principal features extracted from input activity sequences. Further, we will use them to calculate a *co-activity weight* of accounts (or devices) u_i and u_j, e.g., as the dot product or cosine similarity of the latent vectors extracted by FraudGen from their activity sequences.

Predict Account Co-Ownership. We propose to build *co-activity graphs* over sets of accounts and/or devices. For instance, let $\mathcal{U}_s$ be the set of accounts that have reviewed a product s. For a threshold value ϵ, we define the ϵ-co-activity graph of product s to be the weighted graph $G_s = (\mathcal{U}_s, E_s)$, where $(u_i, u_j) \in E_s$ iff. the above computed co-activity value of u_i, u_j exceeds ϵ. We will use network representation learning to extract vector representations of nodes that enable node classification and graph clustering. We will extract embeddings for each account in co-activity graphs of products targeted by raters using network embedding techniques, including based on multi-hop similarity [122, 123], adjacency similarity [124], random walks [125, 126]), and multi layer networks [127]. We will use extracted embeddings to train supervised learning algorithms that predict co-ownership relationships between accounts and/or devices. Another features that we will use to train this classifier builds on the intuition that accounts and/or devices controlled by the same rater are likely to review products in common. We will then use the above embeddings to cluster accounts, using e.g., K-means, then extract a *co-cluster weight*, i.e., the number of times that two user accounts have appeared in the same cluster, in the co-activity graphs of different products.

Fraud De-Anonymization and Pseudonymous Rater Discovery. To de-anonymize fraud, we will use knowledge about accounts in a cluster being controlled by a rater (see Module A), to attribute the other accounts in that cluster to the same rater. However, such information may be conflicting and thus not straightforward to use, since accounts in a cluster may be controlled by multiple raters, e.g., who have previously collaborated on jobs. To address this challenge, we will use the above co-ownership predictor. For instance, we will process each un-attributed account u in each cluster that has both attributed and un-

attributed accounts. For each account u_w in u's cluster that is controlled by a rater w, the co-ownership predictor will determine if u and u_w share the same owner, then attribute u to w. Since account u may appear in multiple clusters, for multiple products that u has promoted, the co-ownership predictor may determine that u can be attributed to multiple raters. Instead of using majority voting to break the tie, we will explore a supervised learning approach, to identify and extract features that measure the attribution strength of u to each known rater, e.g., the number of times it was attributed to the rater, the confidence of the co-ownership predictor, and use ground truth data (§ 4.1) to train a classifier.

We will further use the above co-ownership predictor to develop pseudonymous rater discovery solutions that group previously un-attributed user accounts into communities, each controlled by a different, albeit not yet discovered rater. We propose and will build a *co-ownership graph* $G_c = (V_c, E_c)$, where nodes are rater-controlled, but un-attributed user accounts, while an edge in E_c exists between two nodes if the accounts are predicted by the co-ownership predictor to be controlled by the same rater. Figure 5 shows a co-ownership graph that we have built from data we collected from 5,548 user accounts who reviewed 640 apps involved in fraud. We will investigate and adapt graph partitioning algorithms, e.g., the Karger [128] weighted min-cut, to recursively partition the co-ownership graph into subgraphs that are connected through links of minimal total weight, but each is more densely connected than the original graph. We will also investigate and adapt to the fraud de-anonymization problem, generalizations of dense subgraph identification, e.g., [129, 130] and loopy belief propagation [131].

4.3.2 Evaluation Plan

We will implement and deploy the fraud attribution solutions developed in this module, on the RacketStore platform (§ 4.1). We will recruit raters to post reviews for RacketStore-hosted apps, and will develop IRB-approved protocols to train and evaluate developed solutions. For instance,

• **Training**. We will implement a sequential rater recruitment process, where we only recruit one rater at a time: All the user accounts that post reviews in the specified interval will then be controlled by that rater. We will train the developed fraud de-anonymization solutions using such ground truth attributed accounts. We have IRB approval for this process; In [17] we have collected the first *attributed fraud dataset*, from 39 raters who revealed a total of 1,664 Google accounts that they used to post fraud.

• **Testing with rater oracles**. We will subsequently recruit multiple raters, then use our trained de-anonymizer to attribute each review-posting account to one of the raters recruited. During this live evaluation, we will lack ground truth account attribution information. To address this, we propose and will evaluate a *rater oracle* technique : use recruited raters to validate the outcome of our de-anonymizer. For instance, we will ask each rater to confirm which accounts they have used to post their reviews. To verify oracle honesty, we will include in the queried list also *test* accounts, for which we know the answer, e.g., earlier revealed accounts, and *synthetic* accounts, that we know are not controlled by the rater. We will use collected data to further expand our datasets of ground truth attributed sockpuppet accounts.

We will evaluate developed solutions on measures that include scalability, processing speed, precision and recall of de-anonymization. We discuss ethical considerations of our experiments, in § 4.1.2.

4.4 Generalization of Approach and Results

We consider a general system that models a broad range of peer-opinion platforms (e.g., app markets, crowd-sourced review forums), and a flexible adversary model that aligns with the capabilities, behaviors and strategies that we have seen advertised on specialized groups and forums, and have documented through studies with, and have validated using data that we collected from, professional raters. We believe that the federated rater model, thus also our solutions, apply in both commercial peer-opinion campaigns, and state-sponsored, fake news distribution settings in social networks [132–135]. We conjecture and will investigate that the constraints imposed by the limited, pooled resources of federated raters generate device-use patterns that are both difficult to avoid, and apply to such organizations across the spectrum of influence campaigns

in peer-opinion sites and social networks.

4.5 Plans to Address Envisioned Difficulties

We discuss several problems that can occur during the execution of this project, and plans to tackle them.
Rater Willingness to Participate. There are no direct local legal policies to criminalize search rank fraud in many countries of the Global South, where most raters in our studies reside. In our previous studies [17], participating raters said that they have never faced any kind of legal issues. Such work is also not stigmatized in most countries in the Global South [136]. Hence, the jobs we will post, are not illegal or unethical according to their own law of the land. Further, we expect that participants will be willing to disclose data and strategies despite risks of countermeasures: in conversations with participants in previous studies [17, 18], we found that they are not convinced of the threat, and are confident to be able to bypass any defenses.
Exploiting RacketStore. Participants in our studies may attempt to inject incorrect information, e.g., by using accounts and strategies that differ from the ones that they employ in real fraud jobs. RacketStore may encourage a "fake fraud" market where people create accounts only to make money from jobs on RacketStore. Raters may also use RacketStore as a training platform, to hone their skills and even prove expertise to prospective employers. We will explore techniques to verify the authenticity of claimed fraud. For instance, raters will be required to login with accounts (e.g., using OAuth 2.0) that they also use in commercial sites, enabling us to inspect the available activities of such accounts on commercial sites. Gaining popularity implies at least partial success of our efforts, providing the research community with access to newly developed fraud strategies and an adversarial training playground, *diverted from commercial sites*.
Strategies to Thwart Our Defenses. In § 4.2.3 and § 4.3.2 we detail activities to monitor strategy changes imposed by our solutions on professional raters. Our deep learning approach will not provide human-interpretable cues that help detect fraud, but will ensure that our solutions evolve along with the adversary. We do foresee strategy changes to delay fraud attribution, e.g., raters dividing their accounts into disjoint subsets, and promoting any product from only one subset of accounts. Our de-anonymization solutions introduce however a tradeoff between the fraud operation's efficiency and its detectability: decreasing account reuse decreases profits, and reputable accounts are often preferred in search rank fraud jobs [38, 50, 137].

5 Project Evaluation Plan

We tightly integrate our evaluation into the proposed work: we dedicate Module A to building RacketStore, an app-market inspired training and evaluation platform, then in Modules B (§ 4.2.3) and C (§ 4.3.2) we detail our plans to use RacketStore to evaluate the fraud detection and attribution solutions that we develop.
Fraud Vaccines: User Nudges and Training. In addition, we will use RacketStore to evaluate our ability to influence users to adopt safer behaviors in peer-opinion services, e.g., to raise awareness to search rank fraud and incorporate it into the process of deciding which products to acquire. This differs from the current approach of commercial sites [14, 15, 78], that remove detected fraud. In preliminary studies [17, 99], we observed that fraud removal only further encourages the creation of even more fraud, where raters create more user accounts to replenish closed ones, and post more reviews to replace filtered ones.

Instead, inspired by the recently introduced concept of fake news vaccines [138], we introduce and will implement and evaluate *fake review vaccination* solutions to train users to avoid products promoted through search rank fraud campaigns. We will design UIs to *nudge* [139–141] users toward incorporating concerns about reviews, their quality and perceived authenticity, during their product acquisition decision process. For instance, we will extend the product page to include information such as the number of reviews suspected to have been posted by professional rater organizations, the number of detected organizations, the time interval of the campaigns, and the impact that these reviews have on the rating of the product.

We will design protocols to ask participants to use the RacketStore app to search for a desired product, then choose one of the results. We will include among the results, products claiming similar functionality, but some clean and others with indicators of search rank fraud. We will evaluate the impact of the above

nudges (and the rank of the product) on the user product choice. We will use metrics such as the number of users who inspect the presented details (see above), and the number of users who choose to acquire a suspicious product. These protocols will also allow us to train users to include search rank fraud concerns into their decision process, and to avoid products that are suspected of search rank fraud, e.g., by notifying users who choose to install products that include indicators of search rank fraud.

6 Relevance To Secure and Trustworthy Cyberspace (SaTC)

This project is relevant to the information authenticity track: we vertically integrate (1) a study of professional raters who post fraud for products hosted in peer-opinion online services, with (2) the development of solutions to detect, prevent and neutralize the effects of fraud, and (3) realistic evaluation efforts. The covert, detection-avoidance nature of search rank fraud, further relates our project to intrusion detection.

7 Broader Impacts

Our study of professional raters who post fake fraud in peer-opinion sites will help develop and validate, a new generation of fraud detection solutions. This has the potential to help protect millions of online users from misleading information, substandard products, malware and even censorship [3, 4, 8–11, 137, 142]. The proposed approach is also relevant to the study and detection of state-sponsored political trolls, who organize into "armies" [143–146], and use complex strategies to distribute and promote fake news from large numbers of accounts that they control in social networks [132–135].

Broadening Participation In Computing. Given the project's study of professional rater behaviors and strategies, and the development of protocols to recruit participants to evaluate developed solutions, we expect that underrepresented groups who might otherwise be intimidated by traditional computational work, will find valuable learning experiences in the CaSPR lab.

The PI will leverage the unique opportunities provided by FIU to recruit, mentor and involve in research, underrepresented students. According to the American Society for Engineering Education (ASEE) Profiles, FIU ranks first in the continental U.S. in Hispanic engineering Bachelors degrees awarded. FIU is also ranked 8th in the country in the percentage of graduated African-Americans with B.S. degrees, and educates four times more female minority students than the national average. Over 50% of the undergraduate students in the School of Computing and Information Sciences at FIU are minorities.

The PI has a history of mentoring women and other minority students. He has graduated a female Ph.D. student in Spring 2019. He has also supported and mentored a minority Hispanic undergrad at FIU for the past 2 years, with a year-round REU. The student has graduated in the Summer of 2019 with an ARCH-award winning thesis, and will continue to be advised by the PI, upon his return to FIU as a graduate student in the Spring 2020. The PI has also been a senior personnel on two NSF REU projects, and has mentored 8 undergrad and 2 K-12 students over the past 5 summers, see e.g., [147]).

This project will support 2 Ph.D. students, one expected to be a female Ph.D. candidate. The PI will continue to recruit and support minority undergraduates with separate NSF REU grants. Each supported Ph.D. student will mentor one year-round REU student that we will recruit in this project. At least 1 of the 2 REUs will be Hispanic; both REUs will graduate with a thesis.

Education Plan. The PI believes in a very close hands-on approach for students, making an active effort to engage them individually and as a group to discuss problems, ideas, and papers. The PI has a history of successful integration of research in teaching, regularly including modules on his social network security research, in his graduate and undergrad level computer security courses (e.g., *CIS-5373: System Security*, *CIS-5374: Information Security and Privacy* and *CEN-5079: Secure Application Programming*). The PI will continue to incorporate findings from this project into graduate courses, transform the proposed prototypes and tools into new hands-on instruction labs to be used in graduate level classes, and further stimulate graduate students to get involved in information authentication research.

Community Outreach. The PI has participated in workshops organized by FIU, e.g., the "Burn your Brain" workshop, a 1-day event for local high school students, and has an active collaboration with the Norman S. Edelcup K-8 school in Sunny Isles Beach, advising the research group led by Dr. Christine Todd, comprising 5 female K-8 students. The PI will build on this expertise to organize 1-week long (Monday - Saturday) summer workshops during the 2nd and 3rd summers of the project. To maximize the impact of the workshop, we will recruit 10-15 local K-12 teachers. We will use local contacts and online boards to advertise the workshop.

We will use material from this project to involve the participants in fraud detection, machine learning and design research projects. We will divide the participants accordingly into 3 teams, of 3-5 members per team. Each team will be given an assignment, and will be mentored by the PI and the 2 Ph.D. students. During the first 2 days of the workshop, each mentor will cover background material, including slides and hands-on examples, on fraud detection, supervised learning concepts, and participant recruitment, questionnaire development and ethical concerns during user studies. During the remaining days, each team will work on a specific assignment, inspired by the tasks of this project.

Example assignments include (1) design studies to evaluate human perception of fake reviews, then further split into sub-teams (researchers and subjects) to simulate participant studies, (2) develop a supervised learning classifier of reviews as fake and honest, and (3) develop an unsupervised learning solution to group accounts and reviews by ownership.

Dissemination. We will release RacketStore and the proposed fraud detection and prevention solutions that we develop, as open source, with code available through our lab GitHub account [148] and the CPS-VO. We will publish synthetic, benchmarking data generated in Module B; we will anonymize the data collected in Module A, before making it public. We will report findings via periodic articles and status reports.

8 Project Management

PI Carbunar directs the Cyber Security and PRivacy (CaSPR) lab at FIU, and has relevant expertise in online fraud detection and prevention in Google Play [17–19, 65, 85, 86, 99, 149] and Yelp [150–152], abuse detection and prevention in Facebook [153, 154], and deep learning for mobile authentication [109, 110].

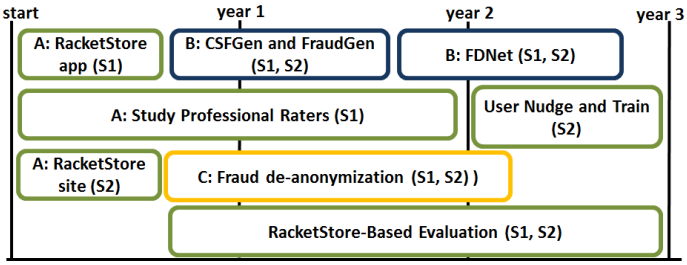

Figure 6: Project timeline, student (S1, S2) involvement.

CaSPR lab has exclusive access to the computational resources required to deliver this project: 3 GPU servers (each with 8 GeForce GTX 1080 Ti GPUs) for the deep learning based fraud detection, 4 servers (each with 40 Intel Xeon CPUs@2.20GHz) and 1 storage server (24 x Seagate 10TB) for hosting RacketStore and storing collected and synthetic fraud data. Figure 6 shows the prospective research timeline.

9 Prior NSF Support

#1527153, PI Carbunar, "TWC: Small: Collaborative: Cracking Down Online Deception Ecosystems", 9/1/2015-8/31/2019, $261,652. *Intellectual Merit*: The project designs fraud detection techniques for online systems. *Broader Impacts*: The impact of review based online services makes the problem of fraud detection of significant importance to victims and to the credibility of the services. This project yielded several publications [18, 19, 65, 85, 86, 99, 150, 153, 154] and publicly released several datasets [155].

#1840714, PI Carbunar, "EAGER: An Open Mobile App Platform to Support Research on Fraudulent Reviews", 8/15/2018-8/14/2020, $149,999). *Intellectual Merit*: The project investigates, develops and evaluates an online framework (precursor of the RacketStore site but not the app) to study search rank fraud in app markets. *Broader Impacts*: This platform will advance the evaluation of fraud detection research. The project resulted in publications [17, 18], and releases the app market and its data.

References

[1] Amazon flooded with thousands of fake reviews, report claims. CNBC, https://www.cnbc.com/2019/04/16/amazon-flooded-with-thousands-of-fake-reviews-report-claims.html, April 2019.

[2] Fake Reviews Are Silicon Valleys Next Fake News Problem. https://streetfightmag.com/2019/08/15/fake-reviews-are-silicon-valleys-next-fake-news-problem/#.XdVV4dVKjZ5, August 2019.

[3] Brian Reigh. Fake reviews on the Play store reportedly growing and getting smarter. *Android Authority*, April 2017.

[4] Emma Woollacott. Amazon's fake fake review problem is now worse than ever, study suggests. *Forbes*, Sept. 2017.

[5] 172 malicious apps with 335M+ installs found on Google Play. The Next Web, https://thenextweb.com/apps/2019/10/01/google-play-android-malware-2/, October 2019.

[6] Kate O'Flaherty. New Google Android Malware Warning Issued To 8 Million Play Store Users. Forbes, https://www.forbes.com/sites/kateoflahertyuk/2019/10/24/new-google-android-malware-warning-issued-to-8-million-play-store-users/#5e0e68421235, October 2019.

[7] Apps Infected with Malware on Google Play Store. Alert Logic, https://www.alertlogic.com/resources/threat-reports/apps-infected-with-malware-on-google-play-store/.

[8] Stephanie Mlot. Top Android App a Scam, Pulled From Google Play. PCMag, 2014.

[9] Zach Miners. Report: Malware-infected Android apps spike in the Google Play store. PCWorld, 2014.

[10] Jan Youngren. How to beat Google Plays algorithm and get 280 million installs. https://vpnpro.com/blog/how-to-manipulate-google-play-rankings-and-get-280-million-installs/, 2019.

[11] Zak Doffman. New Google Warning: 280M+ Android Users At Risk As China Manipulates Play Store. Forbes, https://tinyurl.com/y4ne7th7, 2019.

[12] Simon Parkin. The Never-Ending War on Fake Reviews. The New Yorker, https://www.newyorker.com/tech/annals-of-technology/the-never-ending-war-on-fake-reviews, March 2018.

[13] Mike Blumenthal. How Google Helped Foster the Fake Review Economy and Benefits From It. http://blumenthals.com/blog/2019/01/26/how-google-helped-foster-the-fake-review-economy-and-benefits-from-it/, January 2019.

[14] Jason Cipriani. Google starts filtering fraudulent app reviews from Play Store. ZDNet, https://tinyurl.com/hklb5tk, 2016.

[15] Sarah Perez. Amazon bans incentivized reviews tied to free or discounted products. Tech Crunch, https://tinyurl.com/zgn9sq3, 2016.

[16] Adam Mosseri. Working to Stop Misinformation and False News.

[17] Mizanur Rahman, Nestor Hernandez, Ruben Recabarren, Syed Ishtiaque Ahmed, and Bogdan Carbunar. The Art and Craft of Fraudulent App Promotion in Google Play. In *The 26th ACM Conference on Computer and Communications Security*, 2019.

[18] Nestor Hernandez, Mizanur Rahman, Ruben Recabarren, and Bogdan Carbunar. Fraud de-anonymization for fun and profit. In *Proceedings of the 25th ACM Conference on Computer and Communications Security*, 2018.

[19] Mizanur Rahman, Nestor Hernandez, Bogdan Carbunar, and Duen Horng Chau. Search rank fraud de-anonymization in online systems. In *Proceedings of ACM Conference on Hypertext and Social Media*, 2018.

[20] Parisa Kaghazgaran, James Caverlee, and Anna Squicciarini. Combating crowdsourced review manipulators: A neighborhood-based approach. In *Proceedings of the ACM International Conference on Web Search and Data Mining*, pages 306–314, 2018.

[21] Parisa Kaghazgaran, James Caverlee, and Majid Alfifi. Behavioral analysis of review fraud: Linking malicious crowdsourcing to amazon and beyond. In *Proceedings of the AAAI International Conference on Web and Social Media*, 2017.

[22] Huayi Li, Geli Fei, Shuai Wang, Bing Liu, Weixiang Shao, Arjun Mukherjee, and Jidong Shao. Bimodal distribution and co-bursting in review spam detection. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1063–1072, 2017.

[23] Bryan Hooi, Hyun Ah Song, Alex Beutel, Neil Shah, Kijung Shin, and Christos Faloutsos. Fraudar: Bounding graph fraud in the face of camouflage. In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 895–904, 2016.

[24] Atefeh Heydari, Mohammadali Tavakoli, and Naomie Salim. Detection of fake opinions using time series. *Expert Syst. Appl.*, 58(C):83–92, October 2016.

[25] Bryan Hooi, Neil Shah, Alex Beutel, Stephan Günnemann, Leman Akoglu, Mohit Kumar, Disha Makhija, and Christos Faloutsos. BIRDNEST: bayesian inference for ratings-fraud detection. *CoRR*, abs/1511.06030, 2015.

[26] Geli Fei, Arjun Mukherjee, Bing Liu, Meichun Hsu, Malu Castellanos, and Riddhiman Ghosh. Exploiting Burstiness in Reviews for Review Spammer Detection. In *Proceedings of the AAAI International Conference on Web and Social Media*, pages 175–184, 2013.

[27] Huayi Li, Zhiyuan Chen, Arjun Mukherjee, Bing Liu, and Jidong Shao. Analyzing and detecting opinion spam on a large-scale dataset via temporal and spatial patterns. In *Proceedings of the AAAI International Conference on Web and Social Media*, pages 634–637, 2015.

[28] Arjun Mukherjee, Abhinav Kumar, Bing Liu, Junhui Wang, Meichun Hsu, Malu Castellanos, and Riddhiman Ghosh. Spotting Opinion Spammers Using Behavioral Footprints. In *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining*, pages 632–640, 2013.

[29] Zhen Xie and Sencun Zhu. Grouptie: Toward hidden collusion group discovery in app stores. In *Proceedings of the ACM Conference on Security and Privacy in Wireless and Mobile Networks*, pages 153–164, 2014.

[30] Guan Wang, Sihong Xie, Bing Liu, and Philip S. Yu. Review graph based online store review spammer detection. In *Data Mining, IEEE International Conference on*, pages 1242–1247, 2011.

[31] Alex Beutel, Wanhong Xu, Venkatesan Guruswami, Christopher Palow, and Christos Faloutsos. CopyCatch: Stopping Group Attacks by Spotting Lockstep Behavior in Social Networks. In *Proceedings of the 22Nd International Conference on World Wide Web*, pages 119–130, 2013.

[32] Gianluca Stringhini, Pierre Mourlanne, Gregoire Jacob, Manuel Egele, Christopher Kruegel, and Giovanni Vigna. EVILCOHORT: Detecting communities of malicious accounts on online services. In *Proceedings of USENIX Security Symposium*, pages 563–578, 2015.

[33] Tian Tian, Jun Zhu, Fen Xia, Xin Zhuang, and Tong Zhang. Crowd fraud detection in internet advertising. In *Proceedings of the 24th International Conference on World Wide Web*, pages 1100–1110, 2015.

[34] Junting Ye, Santhosh Kumar, and Leman Akoglu. Temporal opinion spam detection by multivariate indicative signals. In *Proceedings of the AAAI International Conference on Web and Social Media*, pages 743–746, 2016.

[35] Shebuti Rayana and Leman Akoglu. Collective opinion spam detection: Bridging review networks and metadata. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 985–994, 2015.

[36] Vlad Sandulescu and Martin Ester. Detecting singleton review spammers using semantic similarity. In *Proceedings of the 24th International Conference on World Wide Web*, pages 971–976, 2015.

[37] Srijan Kumar, Bryan Hooi, Disha Makhija, Mohit Kumar, Christos Faloutsos, and V. S. Subrahmanian. REV2: fraudulent user prediction in rating platforms. In *Proceedings of the ACM International Conference on Web Search and Data Mining*, pages 333–341, 2018.

[38] Haizhong Zheng, Minhui Xue, Hao Lu, Shuang Hao, Haojin Zhu, Xiaohui Liang, and Keith Ross. Smoke Screener or Straight Shooter: Detecting Elite Sybil Attacks in User-Review Social Networks. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2018.

[39] Srijan Kumar, Justin Cheng, Jure Leskovec, and V.S. Subrahmanian. An army of me: Sockpuppets in online discussion communities. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, pages 857–866, Republic and Canton of Geneva, Switzerland, 2017. International World Wide Web Conferences Steering Committee.

[40] Muhammad AL-Qurishi, Mabrook Alrakhami, Atif Alamri, Majed Alrubaian, Sk Md Mizanur Rahman, and M Hossain. Sybil defense techniques in online social networks: A survey. PP:1–1, 01 2017.

[41] Changchang Liu, Peng Gao, Matthew Wright, and Prateek Mittal. Exploiting temporal dynamics in sybil defenses. In *Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security*, CCS '15, pages 805–816, New York, NY, USA, 2015. ACM.

[42] Zhi Yang, Christo Wilson, Xiao Wang, Tingting Gao, Ben Y Zhao, and Yafei Dai. Uncovering social network sybils in the wild. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 8(1):2, 2014.

[43] Haifeng Yu, Phillip B. Gibbons, Michael Kaminsky, and Feng Xiao. Sybillimit: A near-optimal social network defense against sybil attacks. *IEEE/ACM Trans. Netw.*, 18(3):885–898, June 2010.

[44] George Danezis and Prateek Mittal. Sybilinfer: Detecting sybil nodes using social networks. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2009.

[45] Qiang Cao, Michael Sirivianos, Xiaowei Yang, and Tiago Pregueiro. Aiding the detection of fake accounts in large scale social online services. In *Presented as part of the 9th USENIX Symposium on Networked Systems Design and Implementation (NSDI 12)*, pages 197–210, San Jose, CA, 2012. USENIX.

[46] Meng Jiang, Peng Cui, Alex Beutel, Christos Faloutsos, and Shiqiang Yang. Inferring strange behavior from connectivity pattern in social networks. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 126–138. Springer, 2014.

[47] Yuanshun Yao, Bimal Viswanath, Jenna Cryan, Haitao Zheng, and Ben Y. Zhao. Automated Crowdturfing Attacks and Defenses in Online Review Systems. In *Proceedings of the ACM Conference on Computer and Communications Security*, pages 1143–1158, 2017.

[48] Kurt Thomas, Chris Grier, Dawn Song, and Vern Paxson. Suspended accounts in retrospect: An analysis of twitter spam. In *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*, IMC '11, pages 243–258, New York, NY, USA, 2011. ACM.

[49] Kurt Thomas, Damon McCoy, Chris Grier, Alek Kolcz, and Vern Paxson. Trafficking fraudulent accounts: The role of the underground market in twitter spam and abuse. In *Proceedings of the 22Nd USENIX Conference on Security*, SEC'13, pages 195–210, Berkeley, CA, USA, 2013. USENIX Association.

[50] Gianluca Stringhini, Gang Wang, Manuel Egele, Christopher Kruegel, Giovanni Vigna, Haitao Zheng, and Ben Y. Zhao. Follow the green: Growth and dynamics in twitter follower markets. In *Proceedings of the 2013 Conference on Internet Measurement Conference*, pages 163–176, 2013.

[51] Emiliano De Cristofaro, Arik Friedman, Guillaume Jourjon, Mohamed Ali Kaafar, and M. Zubair Shafiq. Paying for likes?: Understanding facebook like fraud using honeypots. In *Proceedings of the 2014 Conference on Internet Measurement Conference*, IMC '14, pages 129–136, New York, NY, USA, 2014. ACM.

[52] Youngsam Park, Jackie Jones, Damon McCoy, Elaine Shi, and Markus Jakobsson. Scambaiter: Understanding targeted nigerian scams on craigslist. In *21st Annual Network and Distributed System Security Symposium, NDSS*, 2014.

[53] Rebecca S. Portnoff, Sadia Afroz, Greg Durrett, Jonathan K. Kummerfeld, Taylor Berg-Kirkpatrick, Damon McCoy, Kirill Levchenko, and Vern Paxson. Tools for automated analysis of cybercriminal markets. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, 2017.

[54] David Y. Wang, Matthew Der, Mohammad Karami, Lawrence Saul, Damon McCoy, Stefan Savage, and Geoffrey M. Voelker. Search + seizure: The effectiveness of interventions on seo campaigns. In *Proceedings of the 2014 Conference on Internet Measurement Conference*, pages 359–372, 2014.

[55] Stephanie Gokhman, Jeff Hancock, Poornima Prabhu, Myle Ott, and Claire Cardie. In search of a gold standard in studies of deception. In *Proceedings of the Workshop on Computational Approaches to Deception Detection*, pages 23–30, 2012.

[56] Myle Ott, Yejin Choi, Claire Cardie, and Jeffrey T. Hancock. Finding deceptive opinion spam by any stretch of the imagination. In *Proceedings of the Human Language Technologies*, HLT '11, 2011.

[57] Emiliano De Cristofaro, Arik Friedman, Guillaume Jourjon, Mohamed Ali Kaafar, and M. Zubair Shafiq. Paying for likes?: Understanding facebook like fraud using honeypots. In *Proceedings of the 2014 Conference on Internet Measurement Conference*, IMC '14, pages 129–136, 2014.

[58] Chao Yang, Robert Harkreader, and Guofei Gu. Die free or live hard? empirical evaluation and new design for fighting evolving twitter spammers. In *Recent Advances in Intrusion Detection*, pages 318–337. Springer, 2011.

[59] Suranga Seneviratne, Aruna Seneviratne, Mohamed Ali Kaafar, Anirban Mahanti, and Prasant Mohapatra. Early detection of spam mobile apps. In *Proceedings of the 24th International Conference on World Wide Web*, pages 949–959. ACM, 2015.

[60] Dong Yuan, Yuanli Miao, Neil Zhenqiang Gong, Zheng Yang, Qi Li, Dawn Song, Qian Wang, and Xiao Liang. Detecting fake accounts in online social networks at the time of registrations. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, pages 1423–1438, 2019.

[61] Zenghua Xia, Chang Liu, Neil Zhenqiang Gong, Qi Li, Yong Cui, and Dawn Song. Characterizing and detecting malicious accounts in privacy-centric mobile social networks: A case study. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2012–2022. ACM, 2019.

[62] Michael Luca and Georgios Zervas. Fake it till you make it: Reputation, competition, and yelp review fraud. *Management Science*, 62(12), 2016.

[63] Michael Anderson and Jeremy Magruder. Learning from the crowd: Regression discontinuity estimates of the effects of an online review database. *Economic Journal*, 122(563), 2012.

[64] Juju Chang, Jake Lefferman, Claire Pedersen, and Geoff Martz. When Fake News Stories Make Real News Headlines. Nightline, ABC News, November 2016.

[65] Mizanur Rahman, Ruben Recabarren, Bogdan Carbunar, and Dongwon Lee. Stateless puzzles for real time online fraud preemption. In *Proceedings of the ACM Web Science Conference (WebSci)*, 2017.

[66] Haifeng Yu, Michael Kaminsky, Phillip B. Gibbons, and Abraham D. Flaxman. Sybilguard: Defending against sybil attacks via social networks. *IEEE/ACM Trans. Netw.*, 16(3):576–589, June 2008.

[67] Chang Xu and Jie Zhang. Combating product review spam campaigns via multiple heterogeneous pairwise features. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, pages 172–180. SIAM, 2015.

[68] Meng Jiang, Peng Cui, Alex Beutel, Christos Faloutsos, and Shiqiang Yang. Inferring strange behavior from connectivity pattern in social networks. In Vincent S. Tseng, Tu Bao Ho, Zhi-Hua Zhou, Arbee L. P. Chen, and Hung-Yu Kao, editors, *Advances in Knowledge Discovery and Data Mining*, pages 126–138. Springer International Publishing, 2014.

[69] Qiang Cao, Xiaowei Yang, Jieqi Yu, and Christopher Palow. Uncovering large groups of active malicious accounts in online social networks. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, CCS '14, pages 477–488, 2014.

[70] Jonghyuk Song, Sangho Lee, and Jong Kim. Crowdtarget: Target-based detection of crowdturfing in online social networks. In *Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security*, CCS '15, pages 793–804, New York, NY, USA, 2015. ACM.

[71] Zhen Xie and Sencun Zhu. Appwatcher: Unveiling the underground market of trading mobile app reviews. In *Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, WiSec '15, pages 10:1–10:11, New York, NY, USA, 2015. ACM.

[72] Bryan Hooi, Neil Shah, Alex Beutel, Stephan Günnemann, Leman Akoglu, Mohit Kumar, Disha Makhija, and Christos Faloutsos. BIRDNEST: bayesian inference for ratings-fraud detection. In *Proceedings of the 2016 SIAM International Conference on Data Mining, Miami, Florida, USA, May 5-7, 2016*, pages 495–503, 2016.

[73] Prudhvi Ratna Badri Satya, Kyumin Lee, Dongwon Lee, Thanh Tran, and Jason (Jiasheng) Zhang. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, CIKM '16, pages 2365–2370, New York, NY, USA, 2016. ACM.

[74] Zhen Xie, Sencun Zhu, Qing Li, and Wenjing Wang. You can promote, but you can't hide: Large-scale abused app detection in mobile app stores. In *Proceedings of the 32nd Annual Conference on Computer Security Applications*, ACSAC '16, pages 374–385, New York, NY, USA, 2016. ACM.

[75] Chang Xu. Detecting collusive spammers in online review communities. In *Proceedings of the Sixth Workshop on Ph.D. Students in Information and Knowledge Management*, PIKM '13, pages 33–40, New York, NY, USA, 2013. ACM.

[76] Stephan Günnemann, Nikou Günnemann, and Christos Faloutsos. Detecting anomalies in dynamic rating data: A robust probabilistic model for rating evolution. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 841–850, New York, NY, USA, 2014. ACM.

[77] Ee-Peng Lim, Viet-An Nguyen, Nitin Jindal, Bing Liu, and Hady Wirawan Lauw. Detecting product review spammers using rating behaviors. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 939–948, New York, NY, USA, 2010. ACM.

[78] Arjun Mukherjee, Vivek Venkataraman, Bing Liu, and Natalie Glance. What Yelp fake review filter might be doing. In *Proceedings of the AAAI International Conference on Web and Social Media*, 2013.

[79] Shanshan Li, James Caverlee, Wei Niu, and Parisa Kaghazgaran. Crowdsourced app review manipulation. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, pages 1137–1140, New York, NY, USA, 2017. ACM.

[80] Amir Fayazi, Kyumin Lee, James Caverlee, and Anna Squicciarini. Uncovering crowdsourced manipulation of online reviews. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, pages 233–242, New York, NY, USA, 2015. ACM.

[81] Sihong Xie, Guan Wang, Shuyang Lin, and Philip S. Yu. Review spam detection via temporal pattern discovery. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 823–831, 2012.

[82] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pages 843–852, 2017.

[83] Michele Banko and Eric Brill. Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 26–33, 2001.

[84] Alon Halevy, Peter Norvig, and Fernando Pereira. The unreasonable effectiveness of data.

[85] Mahmudur Rahman, Mizanur Rahman, Bogdan Carbunar, and Polo Chau. Search rank fraud and malware detection in google play. *IEEE Transactions on Knowledge and Data Engineering*, 29(6), 2017.

[86] Mahmudur Rahman, Mizanur Rahman, Bogdan Carbunar, and Polo Chau. Fairplay: Fraud and Malware Detection in Google Play. In *Proceedings of the SIAM International Conference on Data Mining (SDM)*, 2016.

[87] App Idea Generator. `https://appideagenerator.com/`.

[88] Iconizer. `https://icons8.com/iconizer`.

[89] RacketStore App Market. `http://www.monkeyrocket.review`.

[90] The Social Marketeers. `http://www.thesocialmarketeers.org/`, Last accessed November 2016.

[91] Review Roster. `http://www.reviewroster.com/`, Last accessed November 2016.

[92] Rank Likes. `http://www.ranklikes.com/`, Last accessed November 2016.

[93] Apps Viral. `http://www.appsviral.com/`, Last accessed November 2016.

[94] App Such. `http://www.appsuch.com`, Last accessed November 2016.

[95] App Reviews. `http://www.app-reviews.org`, Last accessed November 2016.

[96] Dearbhail Bracken-Roche, Emily Bell, Mary Ellen Macdonald, and Eric Racine. The concept of vulnerabilityin research ethics: an in-depth analysis of policies and guidelines. *Health research policy and systems*, 15(1):8, 2017.

[97] TE Parliament. Regulation (eu) 2016/679 of the european parliament and of the council. *Official Journal of the European Union*, 2016.

[98] Guide to Protecting the Confidentiality of Personally Identifiable Information (PII). NIST, `https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-122.pdf`.

[99] Rahul Potharaju, Mizanur Rahman, and Bogdan Carbunar. A Longitudinal Study of Google Play. *IEEE Transactions on Computational Social Systems*, 4(3), 2017.

[100] Goodfellow, Ian and Pouget-Abadie, Jean and Mirza, Mehdi and Xu, Bing and Warde-Farley, David and Ozair, Sherjil and Courville, Aaron and Bengio, Yoshua. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[101] Martín Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein Generative Adversarial Networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 214–223, 2017.

[102] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. SeqGAN: Sequence Generative Adversarial Nets with Policy Gradient. In *AAAI*, pages 2852–2858, 2017.

[103] Parisa Kaghazgaran, Majid Alfifi, and James Caverlee. Wide-ranging review manipulation attacks: Model, empirical study, and countermeasures. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 981–990, 2019.

[104] Greg J. Badros, Alan Borning, and Peter J. Stuckey. The cassowary linear arithmetic constraint solving algorithm. *ACM Trans. Comput.-Hum. Interact.*, 8(4):267–306, December 2001.

[105] Christian Schulte and Peter J Stuckey. Speeding up constraint propagation. In *International Conference on Principles and Practice of Constraint Programming*, pages 619–633, 2004.

[106] IBM ILOG CP Optimizer. `https://www.ibm.com/analytics/cplex-cp-optimizer`.

[107] OptaPlanner. `https://www.optaplanner.org/`.

[108] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *Proceedings of the International Conference on Representation Learning (ICLR)*, 2014.

[109] Mozhgan Azimpourkivi, Umut Topkara, and Bogdan Carbunar. A Secure Mobile Authentication Alternative to Biometrics. In *Proceedings of the 33rd Annual Computer Security Applications Conference (ACSAC)*, pages 28–41, 2017.

[110] Mozhgan Azimpourkivi, Umut Topkara, and Bogdan Carbunar. Camera Based Two Factor Authentication Through Mobile and Wearable Devices. *Proceedings of Ubicomp and the ACM Interact. Mob. Wearable Ubiquitous Technol.*, 1(3), 2017.

[111] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[112] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[113] Protect against security threats with SafetyNet. `https://developer.android.com/training/safetynet`.

[114] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.

[115] Rajat Raina, Andrew Y Ng, and Daphne Koller. Constructing informative priors using transfer learning. In *Proceedings of the 23rd international conference on Machine learning*, pages 713–720, 2006.

[116] Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

[117] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*, 2014.

[118] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[119] Felix A. Gers, Jürgen A. Schmidhuber, and Fred A. Cummins. Learning to Forget: Continual Prediction with LSTM. *Neural Comput.*, 12(10):2451–2471, October 2000.

[120] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

[121] Jessica Su, Ansh Shukla, Sharad Goel, and Arvind Narayanan. De-anonymizing web browsing data with social networks. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17, pages 1261–1269, 2017.

[122] Shaosheng Cao, Wei Lu, and Qiongkai Xu. Grarep: Learning graph representations with global structural information. In *Proceedings of the 24th ACM international on conference on information and knowledge management*, pages 891–900, 2015.

[123] Mingdong Ou, Peng Cui, Jian Pei, Ziwei Zhang, and Wenwu Zhu. Asymmetric transitivity preserving graph embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1105–1114, 2016.

[124] Amr Ahmed, Nino Shervashidze, Shravan Narayanamurthy, Vanja Josifovski, and Alexander J Smola. Distributed large-scale natural graph factorization. In *Proceedings of the 22nd international conference on World Wide Web*, pages 37–48, 2013.

[125] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the International Conference on Knowledge Discovery and Data Mining*, pages 701–710, 2014.

[126] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016.

[127] Marinka Zitnik and Jure Leskovec. Predicting multicellular function through multi-layer tissue networks. *Bioinformatics*, 33(14):i190–i198, 2017.

[128] David R Karger. Global min-cuts in rnc, and other ramifications of a simple min-cut algorithm. In *SODA*, volume 93, 1993.

[129] Moses Charikar. Greedy approximation algorithms for finding dense components in a graph. In *Proceedings of the Workshop on Approximation Algorithms for Combinatorial Optimization*, 2000.

[130] Charalampos E. Tsourakakis. The k-clique densest subgraph problem. In *Proceedings of the Conference on World Wide Web (WWW)*, 2015.

[131] Leman Akoglu, Rishi Chandy, and Christos Faloutsos. Opinion Fraud Detection in Online Reviews by Network Effects. In *Proceedings of the AAAI International Conference on Web and Social Media*, 2013.

[132] Michael Riley, Lauren Etter, and Bibhudatta Pradhan. A Global Guide to State-Sponsored Trolling. Bloomberg, `https://www.bloomberg.com/features/2018-government-sponsored-cyber-militia-cookbook/`, year=.

[133] Neil MacFarquhar. Inside the Russian Troll Factory: Zombies and a Breakneck Pace. New York Times, InsidetheRussianTrollFactory:ZombiesandaBreakneckPace, 2018.

[134] Tom McCarthy. How Russia used social media to divide Americans. The Guardian, https://www.theguardian.com/us-news/2017/oct/14/russia-us-politics-social-media-facebook, 2017.

[135] Heres Everything The Mueller Report Says About How Russian Trolls Used Social Media. BuzzFeed, https://www.buzzfeednews.com/article/ryanhatesthis/mueller-report-internet-research-agency-detailed-2016, 2019.

[136] Lilly Irani, Janet Vertesi, Paul Dourish, Kavita Philip, and Rebecca E. Grinter. Postcolonial computing: A lens on design and development. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 1311–1320, 2010.

[137] Nicholas Confessore, Gabriel Dance, Richard Harris, and Mark Hansen. The follower factory. *The New York Times*, Jan 2018.

[138] J. Roozenbeek and S. Linden. Fake news game confers psychological resistance against online misinformation. *Palgrave Commun*, 65, 2019.

[139] Yang Wang, Pedro Giovanni Leon, Kevin Scott, Xiaoxuan Chen, Alessandro Acquisti, and Lorrie Faith Cranor. Privacy nudges for social media: An exploratory facebook study. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 763–770. ACM, 2013.

[140] Richard H Thaler and Cass R Sunstein. *Nudge: Improving Decisions About Health, Wealth, and Happiness*. Penguin, 2009.

[141] Rebecca Balebako, Pedro G Leon, Hazim Almuhimedi, Patrick Gage Kelley, Jonathan Mugan, Alessandro Acquisti, Lorrie Faith Cranor, and Norman Sadeh. Nudging users towards privacy on mobile devices. In *Proceedings of the CHI Workshop on Persuasion, Nudge, Influence and Coercion*, pages 193–201, 2011.

[142] Ezra Siegel. Fake Reviews in Google Play and Apple App Store. http://www.apptentive.com/blog/fake-reviews-google-play-apple-app-store/, 2014.

[143] Russian web brigades. https://en.wikipedia.org/wiki/Russian_web_brigades.

[144] Proyecto de Formacin del Ejercito de Trolls De La Revolution Bolivariana Pana Enfrentar Guerra Mediatica. https://www.bloomberg.com/features/2018-government-sponsored-cyber-militia-cookbook/data/Ejercito_De_Trolls_Venezuela.pdf.

[145] 50 cent army. https://en.wikipedia.org/wiki/50_Cent_Party.

[146] Public opinion brigades. https://en.wikipedia.org/wiki/Public_opinion_brigades.

[147] Ian Terry, Anita Wu, Sebastian Ramirez, Niki Pissinou, Sitharama Iyengar, and Bogdan Carbunar. Geofit: Verifiable fitness challenges. In *First National Workshop for REU Research in Networking and Systems (REUNS)*, 2014.

[148] Cyber Security and Privacy Research (CaSPR) Lab GitHub page. https://github.com/casprlab/.

[149] Bogdan Carbunar and Rahul Potharaju. A Longitudinal Study of the Google App Market. In *Proceedings of the IEEE/ACM International Conference on Advances in Social Network Analysis and Mining (ASONAM)*, 2015.

[150] Mahmudur Rahman, Bogdan Carbunar, Jaime Ballesteros, and Duen Horng (Polo) Chau. To catch a fake: Curbing deceptive yelp ratings and venues. *Statistical Analysis and Data Mining*, 8(3):147–161, 2015.

[151] Mahmudur Rahman, Bogdan Carbunar, Jaime Ballesteros, George Burri, and Duen Horng (Polo) Chau. Turning the Tide: Curbing Deceptive Yelp Behaviors. In *Proceedings of the SIAM International Conference on Data Mining (SDM)*, 2014.

[152] Jaime Ballesteros, Bogdan Carbunar, Mahmudur Rahman, and Naphtali Rishe. Yelp Events: Making Bricks Without Clay? In *Proceedings of the 5th International Workshop on Hot Topics in Peer-to-peer Computing and Online Social Networks (HotPOST)*, 2013.

[153] Sajedul Talukder and Bogdan Carbunar. AbuSniff: Automatic Detection and Defenses Against Abusive Facebook Friends. In *Proceedings of the International Conference on Web and Social Media*, 2018.

[154] Bogdan Carbunar, Mizanur Rahman, Mozhgan Azimpourkivi, and Debra Davis. GeoPal: Friend Spam Detection in Social Networks Using Private Location Proofs. In *Proceedings of the IEEE International Conference on Sensing, Communication and Networking (SECON)*, 2016.

[155] Search Rank Fraud Detection in Online Services. CaSPR Lab@FIU, `http://www.casprlab.com/socialfraud.html`.