De-anonymizing Web Browsing Data with Social Networks

Jessica Su
Stanford University
jtysu@stanford.edu

Ansh Shukla
Stanford University
anshukla@stanford.edu

Sharad Goel
Stanford University
scgoel@stanford.edu

Arvind Narayanan
Princeton University
arvindn@cs.princeton.edu

ABSTRACT

Can online trackers and network adversaries de-anonymize web browsing data readily available to them? We show—theoretically, via simulation, and through experiments on real user data—that de-identified web browsing histories can be linked to social media profiles using only publicly available data. Our approach is based on a simple observation: each person has a distinctive social network, and thus the set of links appearing in one's feed is unique. Assuming users visit links in their feed with higher probability than a random user, browsing histories contain tell-tale marks of identity. We formalize this intuition by specifying a model of web browsing behavior and then deriving the maximum likelihood estimate of a user's social profile. We evaluate this strategy on simulated browsing histories, and show that given a history with 30 links originating from Twitter, we can deduce the corresponding Twitter profile more than 50% of the time. To gauge the real-world effectiveness of this approach, we recruited nearly 400 people to donate their web browsing histories, and we were able to correctly identify more than 70% of them. We further show that several online trackers are embedded on sufficiently many websites to carry out this attack with high accuracy. Our theoretical contribution applies to any type of transactional data and is robust to noisy observations, generalizing a wide range of previous de-anonymization attacks. Finally, since our attack attempts to find the correct Twitter profile out of over 300 million candidates, it is—to our knowledge—the largest-scale demonstrated de-anonymization to date.

CCS Concepts

•**Security and privacy** → **Pseudonymity, anonymity and untraceability;** •**Information systems** → *Online advertising; Social networks;*

DOI: TBD

1. INTRODUCTION

Online anonymity protects civil liberties. At an abstract level, it enables intellectual freedom: research shows that users change their behavior when they know they are being surveilled online [23], resulting in a chilling effect [32]. Concretely, users who have their anonymity compromised may suffer harms ranging from persecution by governments to targeted frauds that threaten public exposure of online activities [6].

The online advertising industry builds browsing histories of individuals via third-party trackers embedded on web pages. While a small number of companies admit to attaching user identities to these browsing-history datasets, most companies promise users that the histories are pseudonymous and not linked to identity. Privacy advocates have argued that such data can be de-anonymized, but we lack conclusive evidence. It has remained unclear what type of identified auxiliary information could be used in a de-anonymization attack, whether an attack could work at the scale of millions of users, and what the success rate of such an attack would be.

In this paper we show that browsing histories can be linked to social media profiles such as Twitter, Facebook, or Reddit accounts. We begin by observing that most users subscribe to a distinctive set of other users on a service. Since users are more likely to click on links posted by accounts that they follow, these distinctive patterns persist in their browsing history. An adversary can thus de-anonymize a given browsing history by finding the social media profile whose "feed" shares the history's idiosyncratic characteristics.[1]

Such an attack is feasible for any adversary with access to browsing histories. This includes third-party trackers and others with access to their data (either via intrusion or a lawful request). Network adversaries—including government surveillance agencies, Internet service providers, and coffee shop eavesdroppers—also see URLs of unencrypted web traffic. The adversary may also be a cross-device tracking company aiming to link two different browsing histories (e.g., histories generated by the same user on different devices). For such an adversary, linking to social media profiles is a stepping stone.

We make three key contributions. First, we develop a general theoretical framework for de-anonymization. We assume there is a background probability of clicking on links, and that a link appearing in a user's feed increases its probability of appearing in their browsing history by a user-

[1]A user's feed or timeline contains the aggregated content posted by all accounts to which the user subscribes.

specific factor. We then derive a maximum likelihood estimate, which lets us identify the feed in the system most likely to have generated the observed history. This general framing applies to a variety of other de-anonymization attacks (c.f. Section 8).

Our second contribution is implementing and evaluating this technique. We chose Twitter as the source of auxiliary information for several reasons: its real-time API, which avoids the need for large-scale web-crawling; the fact that most activity is public; and finally, the fact that links are wrapped in the `t.co` shortener, which simplifies details of our attack. We assume that either due to the referer header or by exploiting timing information, the adversary knows which links in the user's history resulted from clicks on Twitter. By employing a variety of caching and approximation techniques, we built a system capable of de-anonymizing web browsing histories in real-time, typically in under one minute. To test the performance of this system, we picked 60 active Twitter users at random, obtained their feeds, and simulated browsing histories using a simple behavioral model. Given a synthetic history containing 30 Twitter links, we identified the correct Twitter profile—out of over 300 million active Twitter users—over 50% of the time. We show that our maximum likelihood estimate achieves better accuracy than intersection size and Jaccard similarity, two approaches that have been previously studied in the context of similar de-anonymization tasks [15, 35].

Finally, our third contribution is creating an experiment to test this attack on real browsing histories.[2] We built an online tool to allow users to donate their browsing history; upon which we executed our attack and showed the result to the user so they could confirm or deny. The attack worked correctly for 72% of the 374 users who completed the experiment. We present these results as a proof of concept, noting that our sample of users is not representative.

There are many ways in which users may be de-anonymized when browsing the web (see Section 2). However, our attack is notable for its generality and for the variety of adversaries who may employ it. Any social media site can be used for such an attack, provided that a list of each user's subscriptions can be inferred, the content is public, and the user visits sufficiently many links from the site. For example, on Facebook subscriptions can be inferred based on "likes," and on Reddit based on comments, albeit incompletely and with some error. Further, it is inherent in the web's design and users' behavior, and is not due to specific, fixable vulnerabilities by browsers or websites, unlike previous de-anonymization attacks. It simultaneously confirms the fingerprintability of browsing profiles and the easy availability of auxiliary information. Application-layer de-anonymization has long been considered the Achilles' heel of Tor and other anonymity systems, and our work provides another reason why that is the case.

The increasing adoption of HTTPS on the web diminishes the strength of an attack by network adversaries, but not by third-party trackers. However, network adversaries still see the domain of encrypted requests, even if the URL is hidden. We hypothesize that the attack will still work in this scenario but will require a greater number of links per user. Users can mitigate attacks by installing tracker-blocking tools such as Ghostery, uBlock Origin, or Privacy Badger, as well as HTTPS everywhere to increase the use of encryption. Of course, not revealing one's real-world identity on social media profiles also makes it harder for the adversary to identify the user, even if the linking is successful. Nascent projects such as Contextual Identity containers for Firefox help users more easily manage their identity online [5]. None of these solutions is perfect; ultimately, protecting anonymity online requires vigilance and awareness of potential attacks.

[2]This experiment was approved by Stanford University's Institutional Review Board (Protocol No. 34095).

2. RELATED WORK

The de-anonymization literature is vast, but linkage attacks (and demonstrations of uniqueness) based on *behavior* are especially relevant to our work. These include transactional records of movie viewing [28], location traces [7, 22], credit-card metadata [8], and writing style [27]. Attacks on anonymous communication systems such as long-term intersection attacks and statistical disclosure attacks employ similar principles [24].

To our knowledge, the only previous work that studies the uniqueness of browsing history is by Olejnik et al. [31]. Based on a large online experiment, they report that testing 50 links is sufficient to uniquely fingerprint 42% of users in a sample of about 370,000 users, and that these behavioral fingerprints are stable over time. Their results are not directly comparable with ours: the browsing histories in their experiment were obtained via "history sniffing" (which uses a browser bug that has long been fixed). As a result, they are only able to test for the presence or absence of URLs from a selected list, rather than analyze the entire browsing history of participating users. Further, the work leaves open the question of whether these behavioral fingerprints can actually be linked to auxiliary information.

Wondracek et al. [38] present an online de-anonymization attack that is conceptually similar to ours, although it again involves history sniffing. Further, it relies on additional privacy vulnerabilities on social media sites: specifically, taking an action (such as commenting) results in a request to a URL specific to that comment or other action. From a scientific perspective, the paper shows that group memberships in social media sites tend to be unique, but does not shed light on the uniqueness or de-anonymizability of browsing histories in general (in the absence of the now-fixed privacy vulnerabilities).

Our work also directly relates to third-party online tracking. Such tracking has grown tremendously in prevalence and complexity over the past two decades [20, 25, 34, 4]. Today Google can track users across nearly 80% of sites through its various third-party domains [21]. Web tracking has expanded from simple HTTP cookies to include more persistent tracking techniques, such as the use of flash cookies to "respawn" or re-instantiate HTTP cookies [36], the use of cache E-Tags and HTML5 localStorage for the same purpose [3], and "cookie syncing" between different third parties [11, 1]. Device fingerprinting attempts to identify users by a combination of the device's properties [9, 19]. New fingerprinting techniques are continually discovered [26, 30, 13], and are subsequently used for tracking [29, 2, 1, 10]. These techniques allow trackers to more effectively compile unique browsing histories, but they do not by themselves link histories to identity.

Leaks of PII from first parties to third parties are rampant, and this is one way in which an identity may be attached to pseudonymous browsing histories [18, 17]. Further, the NSA

is known to piggyback on advertising cookies for surveillance; Englehardt et al. [11] show that this technique can be effective and that such a network eavesdropper may also be able to learn users' identities due to usernames and other PII transmitted by websites in the clear. Our work presents a new way in which eavesdroppers may connect web traffic to identities, and it will work even if the PII leaks are fixed.

3. DE-ANONYMIZATION STRATEGY

Our de-anonymization strategy proceeds in three steps. First, we posit a simple model of web browsing behavior in which a user's likelihood of visiting a URL is governed by the URL's overall popularity and whether the URL appeared in the user's Twitter feed. Next, for each user, we compute their likelihood (under the model) of generating a given anonymous browsing history. Finally, we identify the user most likely to have generated that history. A similar likelihood-based approach was used by Ma et al. [22] to de-anonymize location traces.

We construct a stylized model of web browsing by first assuming that a user's web history is generated by a sequence of independent, identically distributed random variables $H_1, \ldots, H_n$, where H_t corresponds to the t-th URL visited by the user. We further assume that each user i has a personalized set of recommended links R_i. For example, on a social media site like Twitter, we can take this recommendation set to be those links that appear in the user's feed (i.e., links posted by the user's friends on the network). Finally, we assume that a user is more likely to visit a link if it appears in the user's recommendation set, where there is a user-specific multiplicative factor r_i that describes each user's responsiveness to the recommendation set. A user's web browsing behavior is thus controlled by two parameters: the recommendation set R_i (which is a set of links), and the recommendation factor r_i.

Theorem 1 below formalizes this generative model of web browsing behavior. Further, given a browsing history and a set of candidate users with recommendation sets $\mathcal{C} = \{R_1, \ldots, R_k\}$, it derives the maximum likelihood estimates $\hat{R}$ and $\hat{r}$. In particular, $\hat{R}$ is the recommendation set (and hence user) most likely associated with a given, de-identified browsing history.

THEOREM 1. *Let $\Omega = \{\omega_1, \ldots, \omega_N\}$ be a universe of items, and suppose $\{p_j\}$ gives a probability distribution on Ω (i.e., $p_j \geq 0$ and $\sum_{j=1}^{N} p_j = 1$). Let $\mathcal{C} = \{R_1, \ldots, R_k\}$ be a collection of recommendation sets, where $R_i \subseteq \Omega$. For any $R \in \mathcal{C}$ and $r > 0$, define a random variable $H_t(R, r)$ taking values in Ω such that*

$$\Pr(H_t = \omega_j) = \begin{cases} rp_j/z & \text{if } \omega_j \in R \\ p_j/z & \text{if } \omega_j \notin R \end{cases}$$

where z is a normalizing factor: $z = r\sum_{\omega_j \in R} p_j + \sum_{\omega_j \notin R} p_j$. Then, given i.i.d. draws $H_1(R, r), \ldots, H_n(R, r)$, the maximum likelihood estimates $(\hat{R}, \hat{r})$ of the underlying parameters are

$$\hat{R} = \operatorname*{argmax}_{R \in \mathcal{C}} \left[q_R \log\left(\frac{q_R}{p_R}\right) + (1 - q_R) \log\left(\frac{1 - q_R}{1 - p_R}\right) \right] \quad (1)$$

and

$$\hat{r} = \left(\frac{q_{\hat{R}}}{1 - q_{\hat{R}}}\right) \Big/ \left(\frac{p_{\hat{R}}}{1 - p_{\hat{R}}}\right) \quad (2)$$

where $q_R = |\{t \mid H_t \in R\}|/n$ and $p_R = \sum_{\omega_j \in R} p_j$.

In the theorem above, q_R is the fraction of links in the observed browsing history that are in the recommendation set R (e.g., the fraction of links appearing in the associated user's Twitter feed). Similarly, p_R is the generalized size of the recommendation set, where it accounts both for the total number of items in the set and the popularity of those items. Intuitively, $\hat{R}$ is a recommendation set for which $q_{\hat{R}}$ is large and $p_{\hat{R}}$ is small; that is, many of the links in the observed history appear in $\hat{R}$ and $\hat{R}$ is not too big. The theorem allows for $r_i < 1$, in which case R_i is an "anti-recommendation" set (e.g., a list of malware links one should not visit). However, in the cases we consider here, $r_i > 1$.

PROOF. Let $X_R(\omega) = 1$ if $\omega \in R$, and $X_R(\omega) = 0$ otherwise. Furthermore, suppose $H_t = \omega_{a_t}$. Then the log-likelihood $\mathcal{L}(R, r)$ of $H_1, \ldots, H_n$ is

$$\begin{aligned} \mathcal{L}(R, r) &= \sum_{t=1}^{n} \left[X_R(\omega_{a_t}) \log\left(\frac{rp_{a_t}}{z}\right) + (1 - X_R(\omega_{a_t})) \log\left(\frac{p_{a_t}}{z}\right) \right] \\ &= \sum_{t=1}^{n} \left[X_R(\omega_{a_t}) \log r + \log p_{a_t} - \log z \right]. \end{aligned}$$

Now, note that

$$\begin{aligned} z &= r \sum_{\omega_j \in R} p_j + \sum_{\omega_j \notin R} p_j \\ &= (r - 1) \sum_{\omega_j \in R} p_j + \sum_{\omega_j \in \Omega} p_j \\ &= (r - 1)p_R + 1. \end{aligned}$$

Consequently,

$$\begin{aligned} \mathcal{L}(R, r) &= \sum_{t=1}^{n} \left[X_R(\omega_{a_t}) \log r + \log p_{a_t} - \log((r - 1)p_R + 1) \right] \\ &= nq_R \log r - n \log\left((r - 1)p_R + 1\right) + \sum_{t=1}^{n} \log p_{a_t}. \end{aligned}$$

Differentiating $\mathcal{L}$ with respect to r, we have

$$\frac{\partial}{\partial r} \mathcal{L}(R, r) = \frac{nq_R}{r} - \frac{np_R}{(r - 1)p_R + 1}$$

and so $\frac{\partial}{\partial r} \mathcal{L}(R, r) = 0$ when

$$r = \frac{q_R}{p_R} \cdot \frac{1 - p_R}{1 - q_R}.$$

At these critical points,

$$\begin{aligned} z &= (r - 1)p_R + 1 \\ &= \frac{1 - p_R}{1 - q_R}. \end{aligned}$$

Substituting these values into the original expression, we find that the value of R at which $L(R, r)$ attains its maximum must also maximize the function

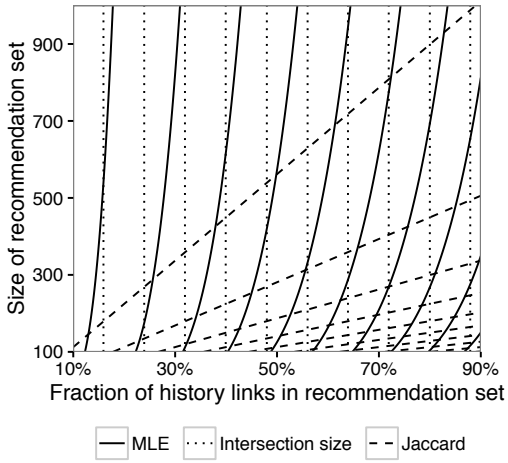

Figure 1: Contour plot that shows how each ranking method scores candidates, as a function of the proportion of history links that appear in the candidate's recommendation set and the size of the recommendation set. Contours of the MLE method are steeper than Jaccard but not as steep as intersection size.

$$
\begin{aligned}
& q_R \log r - \log z \\
&= q_R \log \left[\frac{q_R}{p_R} \cdot \frac{1-p_R}{1-q_R} \right] - \log \left[\frac{1-p_R}{1-q_R} \right] \\
&= q_R \log \frac{q_R}{p_R} - (1-q_R) \log \left[\frac{1-p_R}{1-q_R} \right].
\end{aligned}
$$

Therefore, we find that

$$
\hat{R} = \operatorname*{argmax}_{R \in \mathcal{C}} \left[q_R \log \left(\frac{q_R}{p_R} \right) + (1-q_R) \log \left(\frac{1-q_R}{1-p_R} \right) \right]
$$

and

$$
\hat{r} = \left(\frac{q_{\hat{R}}}{1-q_{\hat{R}}} \right) \Bigg/ \left(\frac{p_{\hat{R}}}{1-p_{\hat{R}}} \right).
$$

□

To help provide intuition about our MLE approach, we compare it to two natural alternative strategies. In the first—which we term the intersection size method [33, 38]—one associates a given history H with the recommendation set R that contains the most URLs from the history (i.e., R that maximizes $|R \cap H|$). In contrast to the MLE, such a strategy does not explicitly adjust for the size of the recommendation set, and one worry is that the intersection size method is biased toward larger recommendation sets. To account for the size of a recommendation set, a second alternative is to associate a history with the recommendation set that has the greatest Jaccard similarity with the history: $|H \cap R|/|H \cup R|$. In many cases, the recommendation set R is much larger than the history H (e.g., the number of links appearing in one's Twitter feed is often much larger than the number of links one visits), and so maximizing Jaccard similarity approximately amounts to finding the candidate set R that maximizes $|H \cap R|/|R|$.

Figure 1 shows that our MLE approach penalizes the size of a candidate recommendation set more than the intersection size method and less than Jaccard similarity. For each of the three de-anonymization methods, the lines in the contour plot indicate level curves, along which candidate recommendation sets receive equal scores. In computing the MLE, for simplicity we assume each item in the recommendation set has constant probability $p_j = 1/10^6$. Because the intersection size method is independent of the size of the recommendation set, its level curves are vertical. Jaccard, in contrast, has linear contours (where we approximate Jaccard by $|H \cap R|/|R|$, so that we do not need to consider dependence on history size). Finally, the MLE has contours that are close to those of intersection size, but not quite vertical.

4. SYSTEM DESIGN

The MLE estimate described above requires calculating q_R, the number of history links appearing in a candidate's recommendation set, and p_R, the generalized size of the recommendation set. Furthermore, it requires maximizing an expression of these quantities across *all* users in the network. For a typical network with hundreds of millions of users, acquiring and calculating these quantities is computationally challenging. Here, we outline a system that addresses these hurdles for the Twitter social network and enables real-time linking of web browsing histories to user profiles.

4.1 Candidate Ranking

We begin by expressing the MLE estimate concretely in terms of the Twitter network: recommendation sets, R_i, are the links posted by friends of a particular user i,[3] and the anonymous history $H = \{H_1, \ldots, H_n\}$ is a set of links some user has clicked on from Twitter.[4] In this formulation, q_{R_i} is the number of links from H posted by user i's friends, and p_{R_i} is a measure of the user's feed size.

An immediate simplification is to reduce our candidate set $\mathcal{C}$ to only feeds that have at least one link from the history, $\hat{\mathcal{C}} = \{R_i \mid R_i \cap H \neq \emptyset\}$, since the MLE is approximately zero outside of this set. Because the complete list of posters for a link is obtainable through a search query, we can find all recommendation sets in $\hat{\mathcal{C}}$. Specifically, we first search Twitter to determine all the users who posted a link in the history. Then, for every poster, we add their followers' feeds to our reduced set of candidates $\hat{\mathcal{C}}$. The search results also let us calculate q_{R_i} scores for every candidate by counting the number of distinct links in H posted by their friends.

The score p_{R_i} is the total background probability of clicking on links in R_i. In practice, it is impossible to determine the exact distribution of these probabilities. We approximate p_{R_i} by assuming that each of user i's friends tweets links with a fixed total probability and estimate $\hat{p}_{R_i} = \lambda \cdot (\#$ of friends of user $i)$, where $\lambda = e^{-15}$. The parameter λ was loosely estimated based on the volume of Twitter links; it was fixed prior to any empirical evaluation.

[3]Twitter defines the friends of a user as the set of people the user follows.

[4]The exact method used to restrict an entire browsing history to links clicked on from Twitter depends on the mode of the attack and is described later in more detail.

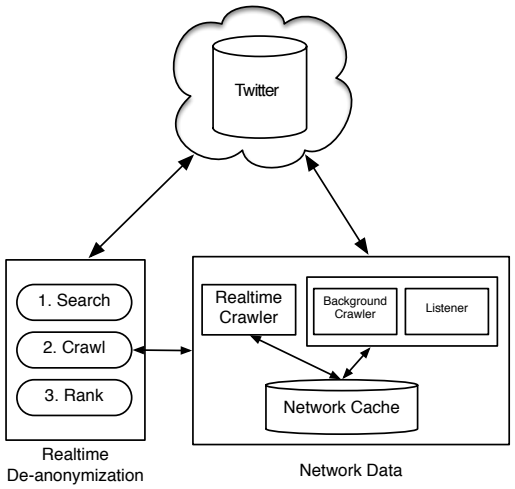

Figure 2: System architecture for real-time de-anonymization of web browsing histories.

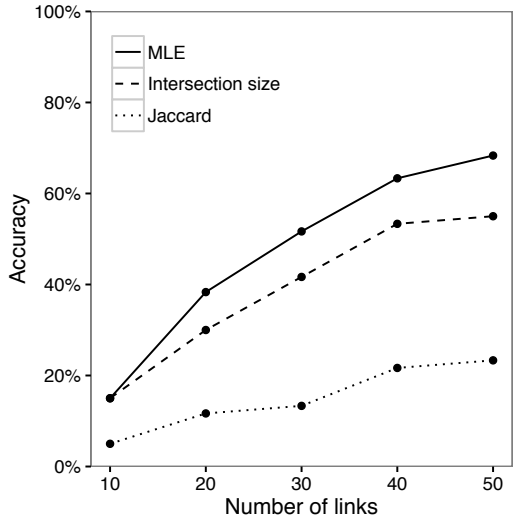

Figure 3: De-anonymization accuracy on synthetic browsing histories for three candidate ranking methods: the MLE of Theorem 1, the number of history links appearing in the candidate's feed (intersection size), and Jaccard similarity.

4.2 System for Real-Time De-Anonymization

Our strategy to generate candidates queries Twitter data sources for search results and network information (i.e., follower lists). There is, however, a limit to how many queries we can make in real-time. To address this issue, we first observe that links appearing in a large number of candidate sets provide little signal, and so one can calculate the MLE on a reduced history $\hat{H} \subseteq H$ of *informative links* whose network information is obtainable within some set amount of time. If any part of a link's network is too expensive to obtain, we disregard the link entirely. This ensures we calculate q_{R_i} exactly for a given history $\hat{H}$. Our approximation $\hat{H}$ can be thought of as strategically throwing away some signal about links clicked by the user in order to exactly and tractably calculate q_R.

To efficiently assemble network information, we use an eager caching strategy: in the background, we run a service that listens to the stream of tweets and finds users with between 10,000 and 100,000 followers; these users are then added to a cache and crawled by a different process. Our de-anonymization strategy thus relies on both pre-fetched data and information obtained in real-time.

Figure 2 outlines the system architecture used for real-time de-anonymization. Our de-anonymization service starts by receiving an anonymous history H. It then searches Twitter for all tweets containing links found in the history. The search results are passed on to the crawling system which attempts to crawl the followers of every poster in the search results. If a poster is not cached and is too expensive to crawl in real-time, we omit the links they posted from our history set to produce $\hat{H}$. At the end of the crawling stage, we have a list of recommendation sets and q_{R_i} scores. The final step of the de-anonymization calculates an MLE using Theorem 1 with the approximations $\hat{\mathcal{C}}$, $\hat{H}$, and $\hat{p}_{R_i}$ described in this section.

5. SIMULATIONS

5.1 Constructing synthetic histories

To evaluate our de-anonymization strategy, we start by examining performance on a set of synthetically generated histories. The simulated histories are constructed to follow a simple behavioral model: a user clicks on links mostly from their own feed and sometimes clicks on links posted by a friend of a friend. A user might be encouraged to visit such friend-of-friend links by Twitter's algorithmic recommendation system [37] or simply visit these links due to organic exploration. In either case, these friend-of-friend URLs test a de-anonymization model's resilience to noise.

Based on the behavioral model, we construct these histories in three steps. First, we monitor the real-time stream of Twitter activity and randomly select a user who posted a tweet, with two caveats: we exclude users with fewer than 20 friends or followers and those with more than 300 friends. The former restriction ensures that our sample includes reasonably active users, and the latter limit is chosen so that we can efficiently construct histories within rate limits. We note that this sample of users is not representative of the overall population of Twitter users, in part because those who appear in the real-time stream of tweets tend to be significantly more active than average.

Next, for each of the selected users, we generate *friend* links and *friend-of-friend* links. Friend links—posted by a friend of the user and thus appearing in the user's feed—are generated by randomly selecting one of the user's friends and then randomly selecting a URL posted by that friend in the last 30 days. We sample links posted by friends-of-friends by first sampling a friend of the user uniformly at random, then sampling a friend of that friend at random, and finally sampling a link posted by that friend-of-friend.

In total we generate 50 friend URLs and 10 friend-of-friend URLs for 60 users.[5] We blend the friend and friend-of-friend URLs to create a collection of synthetic histories for each user with various sizes and link compositions.

[5]We selected 90 users from the stream, but 30 had too little obtainable feed activity to simulate 50 URLs.

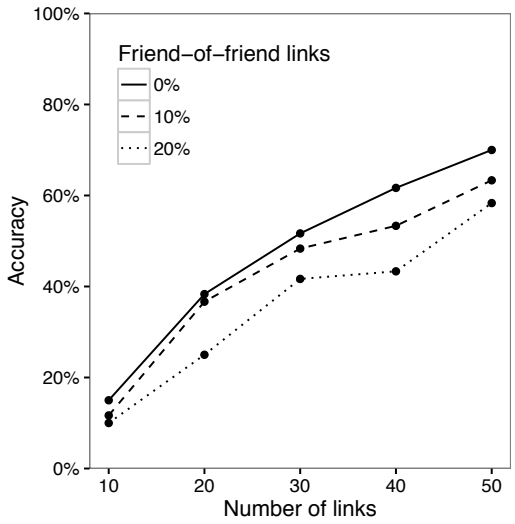

Figure 4: De-anonymization accuracy on synthetic browsing histories generated with varying levels of noise.

5.2 Analysis

Figure 3 compares the accuracy of the MLE of Theorem 1 to that of the intersection and Jaccard methods for pure friend histories. The plot illustrates two points. First, even with a relatively small history of links, the MLE method successfully de-anonymizes a substantial proportion of users. For example, given 30 URLs, the MLE correctly identified 52% of users in our sample. Second, at each history size, the MLE matches or outperforms the other two de-anonymization methods. In contrast to the MLE's 52% de-anonymization rate with 30 URLs, the intersection method correctly identified 42% of users, and Jaccard identified just 13% of users.

We next examine the robustness of our approach to histories containing friend-of-friend URLs. Figure 4 shows these results for blended histories containing 10% and 20% friend-of-friend links at various history sizes. As expected, histories with friend-of-friend links are harder to de-anonymize. In particular, with 30 URLs, 20% of which are friend-of-friend links, we successfully identify 40% of users in our sample, compared to 52% for histories of size 30 containing only friend links. Nevertheless, de-anonymization accuracy is still relatively high.

6. REAL-WORLD EVALUATION

6.1 Collecting web browsing data

The above results on synthetic browsing histories point to the potential for our de-anonymization approach. We next evaluate this method on real, user-contributed web browsing histories, which we collected via an online experiment.

The experiment was open to users running the Google Chrome web browser on a desktop computer. As shown in Figure 5, when users first visited our site, they were provided with a brief description of the experiment and then asked to install a Chrome extension to send us their recent web browsing history. The extension extracted up to 100 Twitter links—marked with domain name `t.co`—visited within the past 30 days to generate the Twitter history H for de-anonymization. If fewer than five links were found in their history, we told users that we would not be able to de-anonymize them and sent no data to our servers. Users with at least five `t.co` links were given an opportunity to verify their data and confirm that they wanted to share it.

The uploaded history was processed by the real-time de-anonymization system described in Section 4.2. The system constructed the reduced history $\hat{H}$ of links by defining informative links as those which were: (1) tweeted or retweeted at most 100 times; and (2) had only been tweeted or retweeted by people with at most 100,000 followers. If a user did not have at least four informative links (i.e., if $|\hat{H}| < 4$), we again told the user that we did not have enough information to successfully run the de-anonymization procedure. Overall, 84% of users who submitted their browsing history passed this filter; among those with at least 10 links, 92% passed; and among those with at least 20 links, 97% passed the filter.

The de-anonymization procedure produced a list of candidates ranked by MLE score. Users were shown the top 15 candidates and prompted to inform us which, if any, corresponded to their Twitter profile. After responding to this question, users were offered an optional opportunity to disclose their identity by signing into Twitter, in which case we would know their identity even if none of our top 15 candidates were correct.

We recruited participants by advertising the experiment on a variety of websites, including Twitter, Facebook, Quora, Hacker News, and Freedom to Tinker. In total, 649 people submitted web browsing histories. In 119 cases (18%), our application encountered a fatal error (e.g., because the Twitter API was temporarily unavailable), and we were unable to run the de-anonymization algorithm. Of the 530 remaining cases, 87 users (16%) had fewer than four informative links, and so we did not attempt to de-anonymize them; we thus attempted to de-anonymize 443 users. Of these, 374 users (84%) confirmed whether or not our de-anonymization attempt was successful. And of these 374 users, 77 (21%) additionally disclosed their identity by signing into Twitter.

We note that the users who participated in our experiment are not representative of the Twitter population. In particular, they are quite active: the users who reported their identity had a median number of 378 followers and posted a median number of 2,041 total tweets.

6.2 Analysis

Of the 374 people who confirmed the accuracy of our de-anonymization attempt, 268 (72%) were the top candidate generated by the MLE, and 303 participants (81%) were among the top 15 candidates.[6] Consistent with our simulation results, we were able to successfully de-anonymize a substantial proportion of users who contributed their web browsing histories.

Figure 6 adds detail to this result, showing accuracy as a function of the size of a participant's submitted history. As expected, performance is strongly related to history size. We correctly identified 86% of users with 50–75 URLs whereas our accuracy falls to 71% for participants with 25–50 URLs.

We also compare the performance of our de-anonymization approach to the intersection method and Jaccard similarity.

[6]In part we achieve this performance because t.co links are uniquely generated every time a link is posted to Twitter. However, even if we consider only the original, unshortened links, we still achieve 49% de-anonymization accuracy.

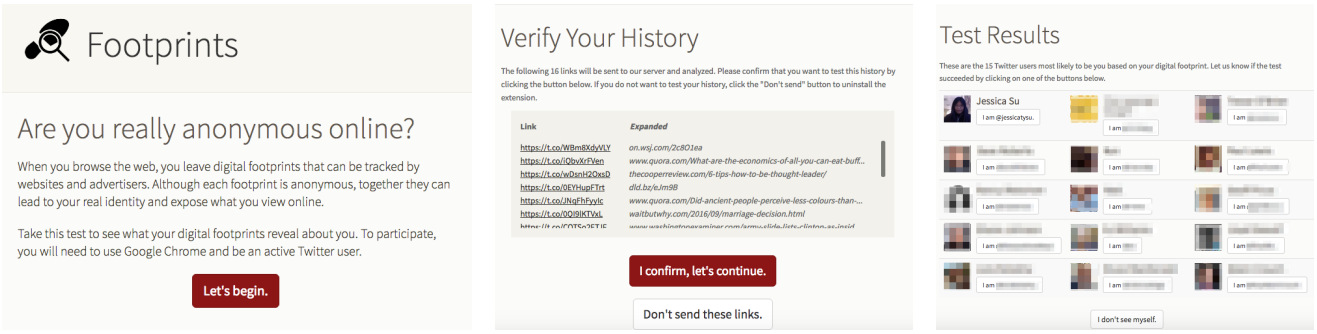

Figure 5: Screenshots of the online experiment.

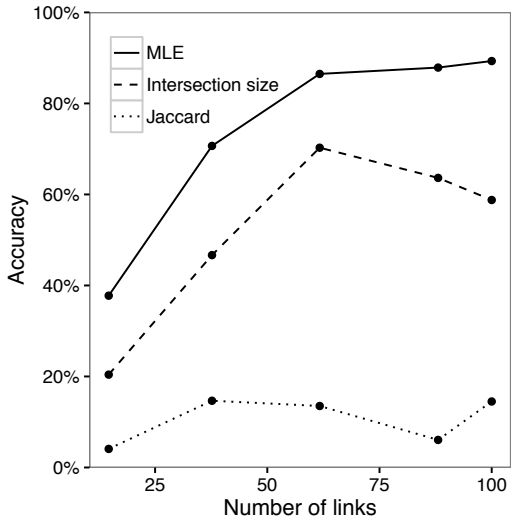

Figure 6: De-anonymization accuracy for three candidate ranking methods on user-contributed web browsing histories. Accuracy for intersection size and Jaccard rankings are approximate, as ground-truth answers are typically only available for users who were ranked in the top 15 by the MLE.

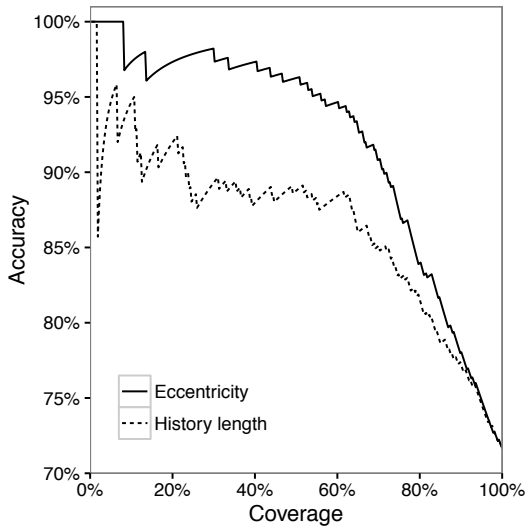

Figure 7: De-anonymization accuracy on the top-k histories ranked by eccentricity and history length.

Unfortunately, because of the experiment's design, we typically only know ground truth for the individuals who ranked in the top 15 by our approach, and it is possible in theory that the other two methods succeed precisely where the MLE fails. To assess this possibility, we consider the 11 cases in which an individual did not appear in our list of top 15 candidates but disclosed their identity by signing into Twitter. In all of these 11 cases, both the intersection method and Jaccard failed to successfully identify the user. Thus, while based on a small sample, it seems reasonable to assume that if a participant is not ranked in the top 15 by the MLE method, then other de-anonymization methods would also have failed. Based on this assumption, Figure 6 compares the performance of all three de-anonymization methods on the full set of 374 users. As on the simulated data, we find that our method outperforms Jaccard similarity and intersection size, often by a substantial margin.

We can further use the MLE scores to estimate the confidence of our predictions. Given ordered candidate scores $s_1 \geq s_2 \geq \cdots \geq s_n$ for an anonymous browsing history H, the *eccentricity* [28] of H is $(s_1 - s_2)/\text{std-dev}(\{s_i\})$. Figure 7 shows prediction accuracy on the top-k instances ranked by eccentricity. The right-most point on the plot corresponds to accuracy on the full set of 374 histories (72%); if we limit to the 50% of histories with the highest eccentricity, accuracy increases to 96%. For comparison, the plot also shows accuracy as a function of history length, and indicates that eccentricity is the better predictor of accuracy.

7. THREAT MODELS

Our de-anonymization strategy assumes access to an individual's Twitter browsing history. Such data are available to a variety of organizations with commercial or strategic incentives to de-anonymize users. In this section, we describe two such possible attackers and evaluate the efficacy of our approach on data available to them.

Third-party trackers are entities embedded into some websites for the purpose of collecting individual user browsing habits. Trackers can determine whether a user arrived from Twitter to a site where they are embedded by examining the page's `document.referrer` property. We estimate the de-anonymization capabilities of four common third-party trackers: Google, Facebook, ComScore, and AppNexus. For each user-contributed history, and for each organization, we first determine which URLs in the history they are likely able to track by checking if the organization has a tracker

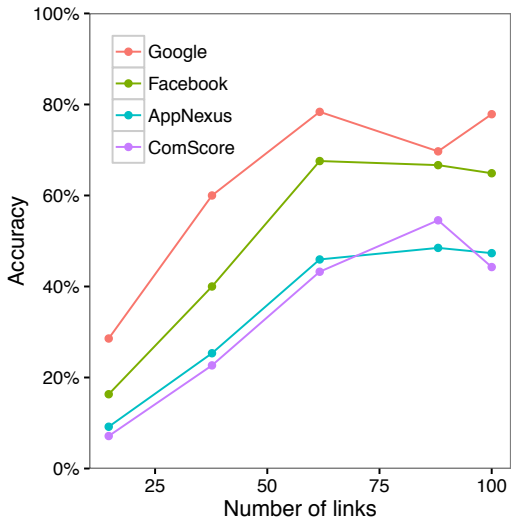

Figure 8: De-anonymization accuracy on the subset of links visible to various organizations that track online behavior, based on user-submitted histories.

installed on the top-level domain of the URL [10]. We then attempt to de-anonymize each history based only on the subset of URLs visible to each tracking organization. Figure 8 shows the results of this analysis and illustrates that all four organizations are pervasive enough to successfully carry out de-anonymization using our method.

Finally, we examine the de-anonymization capabilities of a network eavesdropper. These are attackers capable of sniffing network traffic (e.g., state actors with access to backbone servers) and monitoring server requests from anonymous users. Due to security features of the `https` protocol, such attacks can only determine the full URL of requests made over `http`. Therefore, to simulate the data available to them, we run our de-anonymization strategy using only `http` links submitted in our real-world experiment. We find that network attackers can be fairly successful: 31% of participants in our experiment were identified using only their `http` links.

8. DISCUSSION AND CONCLUSION

We have shown theoretically and empirically that web browsing histories can be linked to social media profiles using only public auxiliary information. The form of Theorem 1 applies to any bounded set of items from which an anonymous actor makes selections influenced by some affinity mechanism. For example, paper citations are likely selected from a universe of relevant work where authors have an affinity for their own work or past citations, as shown by [14]. With this framing, our MLE may be used to de-anonymize papers with author names stripped for double-blind review. Similarly, our model is applicable to the problem of de-anonymizing a movie rental record based on reviews posted on the web [28], as well as a long-term intersection attack against an anonymity system based on, say, the timing of a user's tweets or blog posts [24]. All of these can be seen as behavioral fingerprints of a user, and our analysis helps explain why such fingerprints tend to be unique and linkable. Exploring other problems where our model applies is one direction for future work.

Our statistical approach yielded a useful algorithm that we were able to validate in simulations and with real browsing histories. Our quantitative estimates of accuracy might overestimate the effectiveness of a real-life attack in some ways but underestimate it in other ways. For example, a third-party tracker may not always be able to learn if the current page visit originated from the social media site in question. On the other hand, the adversary may fruitfully make use of other fingerprinting information available through URLs, such as UTM codes. Thus, the main lesson of our paper is qualitative: we present multiple lines of evidence that browsing histories may be linked to social media profiles, even at a scale of hundreds of millions of potential users. Furthermore, our attack has no universal mitigation outside of disabling public access to social media sites, an act that would undermine the value of these sites.

There are many ways in which browsing history may be de-anonymized online. Most straightforwardly, both Facebook and Google—exploiting the fact that they are prominent first parties as well as third parties—track users under their real identities. However, our attack is significant for its broad applicability. The technique is available to *all trackers*, including those with whom the user has no first-party relationship. Our findings are relevant in various other settings. One example is the Federal Communications Commission's recently adopted privacy rule for Internet service providers: the FCC requires that to store and use customer information, ISPs ensure that the information is "not reasonably linkable" to individuals. Our results suggest that pseudonymous browsing histories fail this test, and call for more research into privacy-preserving data mining of browsing histories [16, 12].

Acknowledgments

We thank Twitter for supporting this research by providing free access to the Gnip search API. We also thank Henri Stern for his help building the online experiment. Finally, we thank Jonathan Mayer and the anonymous reviewers for their helpful feedback. Narayanan is supported by NSF award CNS 1526353.

9. REFERENCES

[1] G. Acar, C. Eubank, S. Englehardt, M. Juarez, A. Narayanan, and C. Diaz. The web never forgets: Persistent tracking mechanisms in the wild. In *Proceedings of ACM CCS*, pages 674–689. ACM, 2014.

[2] G. Acar, M. Juarez, N. Nikiforakis, C. Diaz, S. Gürses, F. Piessens, and B. Preneel. Fpdetective: dusting the web for fingerprinters. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pages 1129–1140. ACM, 2013.

[3] M. D. Ayenson, D. J. Wambach, A. Soltani, N. Good, and C. J. Hoofnagle. Flash cookies and privacy II: Now with html5 and etag respawning. 2011.

[4] C. Budak, S. Goel, J. Rao, and G. Zervas. Understanding emerging threats to online advertising. In *Proceedings of the ACM Conference on Economics and Computation*, 2016.

[5] M. Chew and S. Stamm. Contextual identity: Freedom to be all your selves. In *Proceedings of the Workshop on Web*, volume 2. Citeseer, 2013.

[6] N. Christin, S. S. Yanagihara, and K. Kamataki. Dissecting one click frauds. In *Proceedings of the 17th ACM conference on Computer and communications security*, pages 15–26. ACM, 2010.

[7] Y.-A. De Montjoye, C. A. Hidalgo, M. Verleysen, and V. D. Blondel. Unique in the crowd: The privacy bounds of human mobility. *Scientific reports*, 3, 2013.

[8] Y.-A. De Montjoye, L. Radaelli, V. K. Singh, et al. Unique in the shopping mall: On the reidentifiability of credit card metadata. *Science*, 347(6221), 2015.

[9] P. Eckersley. How unique is your web browser? In *International Symposium on Privacy Enhancing Technologies Symposium*, pages 1–18. Springer, 2010.

[10] S. Englehardt and A. Narayanan. Online tracking: A 1-million-site measurement and analysis. In *ACM Conference on Computer and Communications Security*, 2016.

[11] S. Englehardt, D. Reisman, C. Eubank, P. Zimmerman, J. Mayer, A. Narayanan, and E. W. Felten. Cookies that give you away: The surveillance implications of web tracking. In *Proceedings of the 24th Conference on World Wide Web*, 2015.

[12] Ú. Erlingsson, V. Pihur, and A. Korolova. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the Conference on Computer and Communications Security*, 2014.

[13] D. Fifield and S. Egelman. Fingerprinting web users through font metrics. In *International Conference on Financial Cryptography and Data Security*, 2015.

[14] S. Hill and F. Provost. The myth of the double-blind review?: Author identification using only citations. *SIGKDD Explor. Newsl.*, 5(2):179–184, Dec. 2003.

[15] M. Korayem and D. J. Crandall. De-anonymizing users across heterogeneous social computing platforms. In *ICWSM*, 2013.

[16] A. Korolova, K. Kenthapadi, N. Mishra, and A. Ntoulas. Releasing search queries and clicks privately. In *Proceedings of the 18th international conference on World wide web*. ACM, 2009.

[17] B. Krishnamurthy, K. Naryshkin, and C. Wills. Privacy leakage vs. protection measures: the growing disconnect. In *Proceedings of the Web*, 2011.

[18] B. Krishnamurthy and C. E. Wills. On the leakage of personally identifiable information via online social networks. In *Proceedings of the 2nd ACM workshop on Online social networks*, pages 7–12. ACM, 2009.

[19] P. Laperdrix, W. Rudametkin, and B. Baudry. Beauty and the beast: Diverting modern web browsers to build unique browser fingerprints. In *37th IEEE Symposium on Security and Privacy*, 2016.

[20] A. Lerner, A. K. Simpson, T. Kohno, and F. Roesner. Internet jones and the raiders of the lost trackers: An archaeological study of web tracking from 1996 to 2016. In *25th USENIX Security Symposium*, 2016.

[21] T. Libert. Exposing the invisible web: An analysis of third-party http requests on 1 million websites. *International Journal of Communication*, 9:18, 2015.

[22] C. Y. Ma, D. K. Yau, N. K. Yip, and N. S. Rao. Privacy vulnerability of published anonymous mobility traces. *IEEE/ACM Transactions on Networking*, 21(3):720–733, 2013.

[23] A. Marthews and C. Tucker. Government surveillance and internet search behavior. *Available at SSRN 2412564*, 2015.

[24] N. Mathewson and R. Dingledine. Practical traffic analysis: Extending and resisting statistical disclosure. In *International Workshop on Privacy Enhancing Technologies*, pages 17–34. Springer, 2004.

[25] J. R. Mayer and J. C. Mitchell. Third-party web tracking: Policy and technology. In *2012 IEEE Symposium on Security and Privacy*. IEEE, 2012.

[26] K. Mowery and H. Shacham. Pixel perfect: Fingerprinting canvas in HTML5. *W2SP*, 2012.

[27] A. Narayanan, H. Paskov, N. Z. Gong, J. Bethencourt, E. Stefanov, E. C. R. Shin, and D. Song. On the feasibility of internet-scale author identification. In *IEEE Symposium on Security and Privacy*, 2012.

[28] A. Narayanan and V. Shmatikov. Robust de-anonymization of large sparse datasets. In *2008 IEEE Symposium on Security and Privacy (sp 2008)*, pages 111–125. IEEE, 2008.

[29] N. Nikiforakis, A. Kapravelos, W. Joosen, C. Kruegel, F. Piessens, and G. Vigna. Cookieless monster: Exploring the ecosystem of web-based device fingerprinting. In *Security and privacy (SP), 2013 IEEE symposium on*, pages 541–555. IEEE, 2013.

[30] L. Olejnik, G. Acar, C. Castelluccia, and C. Diaz. The leaking battery A privacy analysis of the HTML5 Battery Status API. Technical report, 2015.

[31] L. Olejnik, C. Castelluccia, and A. Janc. Why Johnny can't browse in peace: On the uniqueness of web browsing history patterns. In *5th Workshop on Hot Topics in Privacy Enhancing Technologies*, 2012.

[32] J. Penney. Chilling effects: Online surveillance and wikipedia use. *Berkeley Technology Law Journal*, 2016.

[33] A. Ramachandran, Y. Kim, and A. Chaintreau. "I knew they clicked when I saw them with their friends". In *Proceedings of the 2nd Conference on Online Social Networks*, 2014.

[34] F. Roesner, T. Kohno, and D. Wetherall. Detecting and defending against third-party tracking on the web. In *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, pages 12–12. USENIX Association, 2012.

[35] K. Sharad and G. Danezis. An automated social graph de-anonymization technique. In *Proceedings of the 13th Workshop on Privacy in the Electronic Society*, pages 47–58. ACM, 2014.

[36] A. Soltani, S. Canty, Q. Mayo, L. Thomas, and C. J. Hoofnagle. Flash cookies and privacy. In *AAAI spring symposium: intelligent information privacy management*, volume 2010, pages 158–163, 2010.

[37] J. Su, A. Sharma, and S. Goel. The effect of recommendations on network structure. In *Proceedings of the 25th Conference on World Wide Web*, 2016.

[38] G. Wondracek, T. Holz, E. Kirda, and C. Kruegel. A practical attack to de-anonymize social network users. In *IEEE Symposium on Security and Privacy*, 2010.