Linking Users Across Domains with Location Data: Theory and Validation

Chris Riederer, Yunsung Kim,
Augustin Chaintreau
Columbia University
New York, NY
{mani,augustin}@cs.columbia.edu
yunsung.kim@columbia.edu

Nitish Korula, Silvio Lattanzi
Google Research
New York, NY
{nitish,silviol}@google.com

ABSTRACT

Linking accounts of the same user across datasets – even when personally identifying information is removed or unavailable – is an important open problem studied in many contexts. Beyond many practical applications, (such as cross domain analysis, recommendation, and link prediction), understanding this problem more generally informs us on the privacy implications of data disclosure. Previous work has typically addressed this question using either different portions of the same dataset or observing the same behavior across thematically similar domains. In contrast, the general cross-domain case where users have different profiles independently generated from a common but unknown pattern raises new challenges, including difficulties in validation, and remains under-explored.

In this paper, we address the reconciliation problem for location-based datasets and introduce a robust method for this general setting. Location datasets are a particularly fruitful domain to study: such records are frequently produced by users in an increasing number of applications and are highly sensitive, especially when linked to other datasets. Our main contribution is a generic and self-tunable algorithm that leverages any pair of sporadic location-based datasets to determine the *most likely* matching between the users it contains. While making very general assumptions on the patterns of mobile users, we show that the maximum weight matching we compute is provably correct. Although true cross-domain datasets are a rarity, our experimental evaluation uses two entirely new data collections, including one we crawled, on an unprecedented scale. The method we design outperforms naive rules and prior heuristics. As it combines both sparse and dense properties of location-based data and accounts for probabilistic dynamics of observation, it can be shown to be robust even when data gets sparse.

WWW 2016, April 11–15, 2016, Montréal, Québec, Canada.
ACM 978-1-4503-4143-1/16/04.
http://dx.doi.org/10.1145/2872427.2883002 .

1. INTRODUCTION

Almost every interaction with technology creates digital traces, from the cell tower used to route mobile calls to the vendor recording a credit card transaction; from the photographs we take, to the "status updates" we post online. The idea that these traces can all be merged and connected is both fascinating and unsettling. The ability to merge different datasets across domains can provide individuals with enormous benefits, as seen by increasingly widespread adoption of apps that learn multi-domain user behavior and provide helpful recommendations and suggestions. However, when done by third parties that a user may not interact with directly, this raises fundamental questions about data privacy. In this paper, we focus on location data and show that this type of data is privacy sensitive. More formally, we focus on the following technical question: Is it possible to link accounts of the same user across datasets using just location data? The answer to that question points both to algorithmic feasibility but also our ability to maintain seemingly distinct identities or personas until one chooses to reveal they belong to the same user.

Increasingly often, as shown in recent studies, the location of a smartphone owner is captured and recorded for a majority of mobile apps even in the absence of geographical personalization. This considerably expands the number of parties who can collect and exploit the knowledge of a user's whereabouts. Even when data is recorded sporadically, these datasets are very rich and intimately connected to one's everyday life; they may present or at least partially reflect our most recognizable patterns. Recently, even a small amount of location information was shown sufficient to either render most users distinguishable [7, 25], or infer multiple sociological traits such as race [18], friendship [4, 6], gender, or marital status when combined with domain semantic information [27].

In spite of this work, determining when and how two accounts belong to the same mobile user in *different* domains remains an open problem, primarily for three reasons: First, identity reconciliation is harder than both classifying and distinguishing users. As an example of the former, one may not be able to connect two profiles exactly, but can still be quite certain that both belong to a high-income American, for instance. For the latter, uniqueness of an individual in one dataset does not imply that they will be easily recognized in another one. For instance, in a simple case where individuals produce location records randomly and independently in two domains, users will likely be unique but it is

provably impossible to link them across datasets. Second, as a consequence, many previous methods are domain specific and typically focus on clean and dense parts of the data. In contrast, most of our motivating examples above are sparse, and we aim at leveraging locations in the general case without additional information attached. Third, with almost no exceptions, identity reconciliation was always considered for different parts of the exact same data set, or at best domains that are semantically similar. In contrast, our goal is to address the most general case in which records across domains are separately generated but share an underlying pattern: The user's physical location. Since one cannot occupy two locations at the same time, the common pattern of our physical mobility creates fertile ground to notice events that coincide, and those that are incompatible. The main question is how to use those observations (ideally in a provably optimal manner), under which conditions they are sufficient to link accounts, and how to collect data to empirically validate any related claims.

Exploiting rare coincidences to de-anonymize users is now a classic problem, with a sparsity based method available for almost a decade [15]. While we defer a more detailed comparison with our work to the next section, we would like to point out the main ingredient of our algorithm: a new use of misses and repetitions to interpret coincidental records that exploits the sparse property of coupling between Poisson processes. We note that sporadic collection of records typically resembles such statistics for rare events. This method, which is proved optimal and correct under these simple assumptions, is hence particularly effective in various datasets. Another advantage of our scheme is that it relies on only three parameters[1] that are initially unknown but easy to approximate. We prove empirically that simple methods to estimate these parameters are robust even when starting from imperfect observations.

We now present the following contributions.

- A new generic and self-tunable algorithm which combines positive and negative signals from co-incident events to build a new type of maximum weight matching. In practice this algorithm is compatible with a parameter tuning step exploiting a previously proposed density-based method. In spite of no domain-specific tuning, our algorithm outperforms the state of the art.
- A rigorous interpretation of our algorithm justifying its correctness. In particular we provide a simple model of mobility that encompasses various cases of location-based data. This is, to the best of our knowledge, the first mathematical model for observed location traces across multiple domains. We prove the ideal correct matching maximizes our algorithm's score *and conversely*, that only correct matching achieves maximum score in expectation.
- An empirical evaluation of this problem in three distinct scenarios that significantly extends beyond previous studies in both realism and scope. The first dataset, already publicly available, allows immediate comparison with prior results. For the second scenario considered, we collected data from two current live services, gathering considerably more locations, and proving that our method achieves near perfect accuracy. Finally, our method is shown superior in a commercial scenario that is significantly more heterogeneous and challenging[2].

As we explained above, linking anonymous profiles across domains is considerably more challenging than either establishing users' distinguishability or classifying users into different groups. As such, it may have been considered impractical at scale. The fact that we can link users, sometimes with high precision and recall, shines new light on the protection offered by even the most complete anonymity. Our results are, to the best of our knowledge, the first example of a cross domain analysis of this problem to prove an algorithm's correctness, together with the first validation at scale of location based reconciliation in real cases. As more data are available, and different patterns or domain specific properties are discovered, we believe that more algorithms could be designed and evaluated against the technique we present as a benchmark for the most general case.

2. RELATED WORK

It has been shown that most users in location based datasets are unique, either through a few of their most visited places [25] or based on a few timed visits chosen at random [7, 8]. This property follows a tradition of work specifying the risk of releasing even anonymized datasets [21]. What this shows is that users can be re-identified *in theory*, for instance in one of the following two cases: if an adversary has access to auxiliary information (*e.g.*, the real identity of all users who visited a place at a given time, or an original set of seed nodes which are already re-identified) [7], or alternatively if a public data set is known to intersect the anonymized one [21]. What those works do *not* show, however, is how to exploit this uniqueness in the common case we consider: two *distinct* datasets with no auxiliary information that is known *a priori*.

Identity reconciliation so far has leveraged three principles: *Ad-hoc identifying features* such as matching username, email addresses, or unique tags. Those are ignored here; as recently measured in [10] they are rarely available and accurate. *Information propagation*, where starting from a seed set of identified nodes, a graphical structure such as a social network is exploited to expand the set of matched nodes in static [12, 13, 16, 17, 24] or mobile [20, 11] datasets. Again, those techniques cannot be applied in the general case where no preexisting graph and seeds are known[3]. Finally, *identification of nearest neighbors* using similarity metrics [15, 9] generalizes the first method to leverage non-identifying features and imperfect matches. Data sparsity plays an important role, which is typically included in the design of the similarity metric. This approach suffers from the opposite problem: it applies so broadly that it is very

[1]Two are related, so estimation has two degrees of freedom.

[2]This dataset was not released in raw form to any researcher in the team; the evaluation was run on a remote server with a non-exclusive agreement that other academic researchers can replicate in the spirit of reproducing and improving future reconciliation methods. Note that the authors from Google did not have even remote access to this data.

[3]In the most ambitious information propagation where seeds may be noisy and structures, initially unknown, are inferred, the differences between this approach and one based on similarity starts to fade. We experimented with it but found no improvement from information propagation to report.

loosely defined. Indeed, most successful reconciliations using this technique report on the art of deciding upon informative similarity features – or often the subtlety of their combined effects [9] – without necessarily providing a unified justification. Moreover, a closer look showed that the accuracy of similarity methods for static features (e.g., name, home location, friends) are typically overestimated in practice [10]. Our work addresses this important need: Our inference method interprets location datasets, however different in their domains, as sporadic observations of the same hidden mobility processes. We generalize data sparsity from a static viewpoint to a dynamic viewpoint, leveraging naturally misses and repetitions in the observed processes. In spite of a considerable amount of prior work on Entity Resolution [5], we did not find similar analysis and algorithms, probably because mobile datasets are relatively new and exhibit specific dynamics. Similarly, the related literature on network alignment [1] rarely considers the bipartite case [14] and it centers on static graphs. We empirically found that our method yields superior accuracy to those previously proposed, while being more robust and easy to use.

Other attempts at re-identifying users using mobility data *only* have typically expressed similarity between users with *density based methods* [23, 9]. Those rely on a user having a discriminative pattern in the frequency she visits various places. In [23] author aims at reconciling users in the same domain but at different periods, hence ignoring the time of the visits themselves. In situations where datasets overlap in time, those techniques leave much information unused.[4] Another technique, somewhat diametrically opposed, uses *specific visit times* [19]. Prior to this paper, this was only validated in a single domain (by randomly extracting a subset of each user's profile to recognize). We empirically show that none of those methods extend to the more demanding cross domain case without incurring large inaccuracy. This confirms previous observations that density and time based similarities can reduce the scope of re-identification attacks by removing a lot of dissimilar accounts [9], but cannot be used as is for reconciliation as they lead to low accuracy in practice [10]. Finally, we should mention a statistical learning approach based on Dirichlet distribution used to relate anonymous CDR data with publicly available social network data [2, 3]. It remains, however, difficult to judge its effectiveness as it is used without further theoretical justification and validated without ground truth in the data. Our method, in contrast, is tailored from scratch to location based datasets, its correctness is proved under simple assumption on nodes' visits, and it has been evaluated on three data-sets with ground truth, among the largest available to date, including two that have never appeared in this context. Whether more generic statistical learning reproduces some of the strength of our method remains an interesting question to explore beyond the scope of this paper.

3. LOCATION-BASED RECONCILIATION

3.1 Problem Formulation and Model

We use U and V to denote the set of n user accounts in the two domains, with accounts to be linked using location-based data. Let σ_I denote the true ("identity") mapping that correctly links the two accounts of the each user. The users may visit locations at various times and perform an *action* (such as a checkin), which results in the creation of a record in one of the datasets. Each such record is associated with the location and time-stamp, and possibly additional semantic information that is relevant to this dataset, but may not make sense in a different domain. Therefore, in our algorithm, we *only* use the time-stamped location data. Note that locations and times may be recorded at a different granularity and levels of precision in the two different datasets to be reconciled (for instance, one may only record the nearest cell tower, the other has GPS coordinates). To account for this, we divide locations and times into *bins*, corresponding to a geographical region or interval of time; For a fixed bin corresponding to location region ℓ and time interval t, any action recorded in region ℓ during time interval t is associated with bin (ℓ, t). We use L to denote the set of all location regions and T the set of time intervals in the union of our datasets.

As shown in Figure 1, although each user u or v physically follows a continuous time trajectory M_t (shown on the left), her *mobility record* $r(u)$ in each domain is defined as the multi-set of (location, time) bins in which she took an action: $r(u) = \{(\ell_1, t_1), (\ell_2, t_2), \dots\}$. Note that it is important that this is a multiset: if a user records 2 actions in the same bin, this bin is present twice in the mobility record. Given a specific (location, time) pair (ℓ, t) we denote the number of actions in domain 1 that user u took by $a_1(u, \ell, t)$ (i.e., the number of occurrences of (ℓ, t) in the multiset $r^1(u)$). We define $a_2(u, \ell, t)$ similarly for domain 2. For ease of notation, we use a_1 (respectively a_2) to denote $a_1(u, \ell, t)$ (resp. $a_2(u, \ell, t)$) when u, ℓ, t are clear from the context.

In this paper, we focus on reconciling users across two domains based only on their mobility records, which we refer to as $r^1(u)$ and $r^2(u)$ respectively. In other words, given a collection of mobility records $\{\ r^1(u) \mid u \in U\ \}$ and $\{\ r^2(u) \mid u \in V\ \}$ for the same population but with no identity attached, our goal is to return the true mapping σ_I which maps the record belonging to one user to the record of the same user in the other collection.

3.2 Mobility Model and Assumptions

In order to formally analyze algorithms applying to the cross-domain reconciliation problem defined above, it is necessary to work under a given *mobility model* which governs how users produce records. Without such assumption, only worst-case performance can be measured, which is arbitrarily bad for any algorithm since one can devise instances where the set of locations with actions in domain 1 is completely disjoint from the set of locations with actions in domain 2. Providing the first such model and proving it leads to a practical method is one of our key contributions.

We assume the mobility records follow a simple generation process: First, for each (location, time) pair, the number of visits of each user to this location during this time period follows a Poisson distribution, with rate parameter $\lambda_{\ell,t}$ and this choice is independent of the visits produced for any other pair. It is a rather crude but effective assumption, as it combines mathematical simplicity (critical later to justify our method), and a form of robustness. Indeed, Poisson distributions are known to be good approximations of rare event processes and to combine gracefully when summed, al-

[4]It is, for instance, entirely ineffective in a homogeneous population where each user follows the same location distribution for her visits. Our method, in contrast, is proved to correctly handle that case.

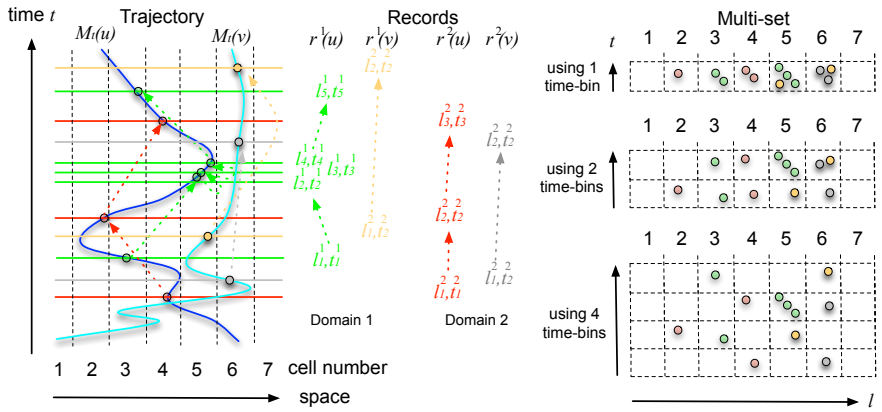

Figure 1: Two space-time trajectories with associated footprints in two domains.

lowing multiple granularity levels to be combined. They are quite commonly used to handle robust parameter estimation, which is important as the parameter $\lambda_{\ell,t}$ is unknown to the algorithm.

The characterization above describes how visits are produced, but does not specify how users perform actions that are observed. We assume that each time the user visits a location, an action in domain 1 and domain 2 occurs, independently of each other, with probabilities p_1 and p_2 respectively. Thus, the mobility records are random variables, which we denote by $R^1(u)$ and $R^2(u)$ respectively, with the number of actions in a given bin (ℓ, t) being random variables denoted by $A_1(u, \ell, t)$ and $A_2(u, \ell, t)$ respectively. The process of visits and action in each domain is also assumed to be independent among users.

Possible extensions: While we keep the model to its simplest form for the sake of a clear exposition, the arguments we provide in this paper generalize to multiple other cases. First among them, all results apply as well when the probability p_1 and p_2 could depend on l and t as well. One could also analyze our algorithms when those parameters are not constant among users. After experimenting with those more general models, we found that they do not yield significant practical improvement in the scenarios we evaluated. We also note that one can adopt different generative models, but many of these do not change the problem significantly, or the analysis of our algorithm. For instance, the number of visits to a particular location may be generated by a binomial distribution, instead of Poisson.

Other extensions are interesting topics for further study: For example, our model does not currently account for geographical proximity between different locations; in reality, users who visit a location ℓ are also likely to visit a nearby location ℓ'. One advantage is that this keeps our model general and robust to variations in formats and resolution across datasets that are quite common in space-time data. For instance, actions 1km apart may be considered close in a rural setting but far in an urban area. Our method is agnostic to such relative change of distance. We also note that our model ignore dependencies between users. For instance, members of a family may travel together and the presence of friends in a location may render a visit by a given person more likely. On the other hand, our model can accommodate frequency of visits that vary between users and hence create communities that on average visit frequently similar places. With larger and richer data, it is likely that more realistic models than ours may give additional insights and better exploit users' true mobility patterns. However, the simple case we define above leads to a simple algorithm that captures mobility of users sufficiently well to beat the state of the art and present a reasonable benchmark for future use.

4. ALGORITHM AND ANALYSIS

In this Section, we present an algorithm tailored to the location record model introduced above. Our main contribution is a proof that under these assumptions, there is a tight correspondence between the maximum weight matching that we define and the 'true' matching between users, even exhibiting a positive gap. Later, Section 5 will demonstrate that this correspondence generalizes in practice to make this algorithm a superior alternative to multiple known approaches.

4.1 Algorithm

Our algorithm works in two phases: The first phase is to compute a score for every candidate pair of users $(u, v) \in U \times V$ (see below for more details). In a second phase, we first define a complete bipartite graph on (U, V) where the weight of the edge (u, v) is given by the score for (u, v) aforementioned. We then compute the matching in this bipartite graph that has maximum weight[5]. The algorithm then claims that records that are connected by an edge belong to the same user. Under the assumptions introduced above, we can prove that this procedure is always correct.

In the rest of this section, we provide more details on how the scores of a pair (u, v) are determined: For each (location, time) bin (ℓ, t), we compute $\text{Score}(u, v, \ell, t) = \ln(\phi_{\ell,t}(a_1, a_2))$, where the term $\phi_{\ell,t}$ in the logarithm is:

$$\frac{P\left[A_1(u,\ell,t) = a_1 \wedge A_2(v,\ell,t) = a_2 \mid \sigma_I(u) = v\right]}{P\left[A_1(u,\ell,t) = a_1\right] \cdot P\left[A_2(v,\ell,t) = a_2\right]}.$$

[5]If some edges have negative weight it is possible in theory for a maximum weight matching not to match all users. However, under our assumptions it does not happen.

The numerator of ϕ measures the probability that the *same* user performs a_1 actions in domain 1 and a_2 actions in domain 2 in the bin (ℓ, t). The two terms in the denominator are the probability that an arbitrary user performs a_1 actions in domain 1 in bin (ℓ, t), and another user performs a_2 actions in domain 2 in this bin. Since we assume that user performs actions independently, $\phi_{\ell,t}(a_1, a_2)$ measures how much *more* likely it is to observe a_1 actions in domain 1 by account u and a_2 actions in domain 2 by account v if these accounts belong to the *same user* than if these are two different users.

Note that, in the above definition of $\phi_{\ell,t}$, the probability is taken in the model we introduce (*i.e.*, that of independent actions taken conditioned on Poisson visits). This yields multiple equivalent formulas to compute the ratio $\phi_{\ell,t}$:

LEMMA 1. *The value of $\phi_{\ell,t}(a_1, a_2)$ in the model we introduce is equal to any of the following expressions (where $\lambda_{\ell,t}$ is denoted by λ for ease of notation):*

(i) $$\frac{P\left[A_1(u,\ell,t) = a_1 \wedge A_2(v,\ell,t) = a_2 \mid \sigma_I(u) = v\right]}{P\left[A_1(u,\ell,t) = a_1\right] \cdot P\left[A_2(v,\ell,t) = a_2\right]}.$$

(ii) $$\frac{e^{-\lambda} \sum_{k \geq \max(a_1,a_2)} \frac{\lambda^k \binom{k}{a_1}(1-p_1)^{k-a_1}\binom{k}{a_2}(1-p_2)^{k-a_2}}{k!}}{\sum_{k \geq a_1} \frac{\lambda^k \binom{k}{a_1}(1-p_1)^{k-a_1}}{k!} \cdot \sum_{k \geq a_2} \frac{\lambda^k \binom{k}{a_2}(1-p_2)^{k-a_2}}{k!}}.$$

(iii) $$\frac{e^{-\lambda(1-p_1-p_2)}}{(\lambda(1-p_1))^{a_1}(\lambda(1-p_2))^{a_2}} \sum_{k \geq \max(a_1,a_2)} \frac{(\lambda(1-p_1)(1-p_2))^k k!}{(k-a_1)!(k-a_2)!}.$$

(iv) $$\frac{e^{-(\lambda p_1 p_2)}(1-p_1)^{a_2}(1-p_2)^{a_1}}{(\lambda(1-p_1)(1-p_2))^{\min(a_1,a_2)}} \mathbb{E}\left[\frac{(X+\max(a_1,a_2))!}{(X+|a_1-a_2|)!}\right],$$
for expectation taken over X a Poisson variable with parameter $r = \lambda(1-p_1)(1-p_2)$.

PROOF. (i) becomes (ii) once we develop each probability by conditioning on the number of visits k that u and/or v make to the bin (ℓ, t), and we observe that a few terms simplify. To obtain (iii) one should observe by the Poisson sampling property that $A_1(u, \ell, t)$ is also distributed according to a Poisson variable, with parameter (λp_1). This simplifies the denominator which then yields this expression. Finally, to obtain (iv), it suffices to introduce the change of variables $k' = k - \max(a_1, a_2)$ and notice that the series becomes this expectation taken over all possible values taken by X. □

Our algorithm, formalized immediately below, can leverage any of the above formulas to approximate ϕ. Expression (i) is the most general (and holds even for non-Poisson visits). Using (iv) with $p_1 = p_2$ and $a_1 = a_2 = a$ we see that the score is especially large when λ is small (as this visit is rare) and a is large (the common observations occurs more than once). For each pair of records, the algorithm computes all the scores associated with the (location,time) bins. It sums them across all bins to compute the weight of the edge between this pair.

While the algorithm is conceptually well defined, there are two things to note about its implementation. First, the input includes the set of parameters of the Poisson distribution, $\{\lambda_{\ell,t}\}$; these are not known, but can be estimated (see discussion in Section 5). Second, the definition of ϕ involves infinite sums over all values of $k \geq a_1, a_2$. We prove below that this can be approximated to arbitrary precision by taking the sum over a limited number of terms.

We now justify our algorithmic approach, and prove that the expected score is highest for the true matching.

Algorithm 1: Our reconciliation algorithm

Require: $\forall u \in U : r^1(u), \forall v \in V : r^2(v), \{\lambda_{\ell,t}\}$
  for $(u, v) \in (U \times V)$ **do**
    $w(u, v) = \sum_{t \in T} \sum_{\ell \in L} \ln \phi_{\ell,t}\left(a_1(u, \ell, t), a_2(v, \ell, t)\right)$
  end for
  Let $E = \{w(u, v) : (u, v) \in (U \times V)\}$
  Compute the maximum weighted matching on the bipartite graph $B(U, V, E)$
  return the function that maps matched vertices.

4.2 Relation to Maximum Likelihood

We explain our choice of the function ϕ (and hence our specific weight function $w(u, v)$) by showing that the weight of a matching is proportional to its log likelihood, and the matching with maximum expected weight (i.e. maximum expected likelihood) is indeed the true matching σ_I.

The observed inputs to the algorithm are the mobility records r^1, r^2. Taking a maximum likelihood estimation (MLE) approach, our goal is to find the matching or permutation σ that maximizes the likelihood $P[\sigma \mid r^1, r^2]$. As is standard, we have:

$$P[\sigma \mid r^1, r^2] = \frac{P[R^1 = r^1, R^2 = r^2 \mid \sigma] \cdot P[\sigma]}{P[R^1 = r^1, R^2 = r^2]}$$

Assuming a uniform prior over all permutations σ, it is easy to see that we are trying to find the permutation σ maximizing $P[R^1 = r^1, R^2 = r^2 \mid \sigma]$.

Assuming σ is the true permutation / mapping, since mobility of different users is independent, the probability of observing various actions for u depends only on the actions of $\sigma(u) = v$. Therefore, we have: $P[R^1 = r^1, R^2 = r^2 \mid \sigma]$

$$= \prod_{u,v:\sigma(u)=v} \prod_{\ell \in L} \prod_{t \in T} P[a_1(u, \ell, t), a_2(v, \ell, t) \mid \sigma_I(u) = v] \quad (1)$$

To normalize this probability, we divide by the overall probability of observing r^1 and r^2 in the two domains. Since $P[R^1 = r^1] = \prod_u \prod_{(\ell,t) \in L \times T} P[A_1(u, \ell, t) = a_1(u, \ell, t)]$ and $P[R^2 = r^2] = \prod_v \prod_{(\ell,t) \in L \times T} P[A_2(v, \ell, t) = a_2(v, \ell, t)]$ we note in particular that $P[R^1 = r^1] \cdot P[R^2 = r^2]$ does not depend on σ. Hence dividing Eq.(1) by it does not change which σ maximizes the likelihood.

Combining these, it is easy to observe that the likelihood of σ is proportional to:

$$\frac{P[R^1 = r^1, R^2 = r^2 \mid \sigma]}{P[R^1 = r^1] \cdot P[R^2 = r^2]} = \prod_{u,v:\sigma(u)=v} \prod_{(\ell,t) \in L \times T} \phi_{\ell,t}(a_1(u, \ell, t), a_2(v, \ell, t))$$

Taking the logarithm of both sides, we see that the log likelihood is proportional to:

$$\sum_{u,v:\sigma(u)=v} \sum_{(\ell,t) \in L \times T} \ln \phi_{\ell,t}(a_1(u, \ell, t), a_2(v, \ell, t)) = \sum_{u,v:\sigma(u)=v} w(u, v)$$

To put it differently, this proves that the log likelihood of σ is exactly the weight of the matching it defines in the bipartite graphs that our algorithms constructs. Hence, constructing a maximum-weight matching as our algorithm does is equivalent to computing the maximum-likelihood permutation σ given our observations.

What remains to be shown is that maximum likelihood exhibits a gap, *i.e.*, the correct permutation σ_I reconciling identity of all users has an expected weight that is higher than any other permutation by a positive margin. Note that, since ϕ involves infinite sums, we need to prove this result for the approximated expected weight that we obtain after truncating each sum in the definition of ϕ.

4.3 Proof of Correctness

Recall that for each location ℓ and time t, we compute a score for a pair of users u and v based on the number of observed actions $a_1(u,\ell,t)$ and $a_2(v,\ell,t)$ as the logarithm of the function $\phi_{\ell,t}$. Fixing ℓ, t, we drop the subscripts and simply write $\lambda = \lambda_{\ell,t}$ and $\phi = \phi_{\ell,t}$. We defined $\phi(a_1, a_2)$ as:

$$\frac{e^{\lambda}\sum_{k\geq\max\{a_1,a_2\}}\frac{\lambda^k}{k!}\binom{k}{a_1}(1-p_1)^{k-a_1}\binom{k}{a_2}(1-p_2)^{k-a_2}}{\sum_{k\geq a_1}\frac{\lambda^k}{k!}\binom{k}{a_1}(1-p_1)^{k-a_1}\cdot\sum_{k\geq a_2}\frac{\lambda^k}{k!}\binom{k}{a_2}(1-p_2)^{k-a_2}}$$

Note that this requires taking three infinite sums, but to define a practical algorithm, we cannot sum over an infinite number of terms. We now argue that for any C, we can efficiently approximate ϕ to within $\pm 1/C$. More formally

THEOREM 1. *Let $C \geq e^7$ and $\phi'(a_1, a_2)$ be defined using the above definition of $\phi(a_1, a_2)$ by truncating the numerator after $\max\{\ln C, 2\max\{a_1, a_2\}\}$ terms, and each factor in the denominator after $\ln C$ terms. We then have*

$$1 - \tfrac{1}{C} \leq \tfrac{\phi'(a_1,a_2)}{\phi(a_1,a_2)} \leq 1 + \tfrac{1}{C}\,.$$

We now show that the expected weight of the true / identity permutation is larger than the expected likelihood of any other permutation by a constant, even after truncating the calculation of $\phi(a_1, a_2)$.

LEMMA 2. *For any bin (ℓ, t) and any pair of users (u, v), then $v \neq \sigma_I(u)$ implies $E[Score(u, v, \ell, t)] \leq 0$. On the other hand, $v = \sigma_I(u)$ implies $E[Score(u, v, \ell, t)] > \lambda_{\ell,t} p_1^2 p_2^2 K$, where $K = \frac{1}{2}\lambda(p_1 + p_2 - p_1p_2)^2$.*

PROOF. Since we have a fixed ℓ, t, we use ϕ to denote $\phi_{\ell,t}$, λ to denote $\lambda_{\ell,t}$, and $A_1(u), A_2(v)$ to denote $A_1(u,\ell,t)$ and $A_2(v,\ell,t)$ respectively. First, consider the case $v \neq \sigma_I(u)$. The expected value of ϕ, *i.e.*, $E[\phi(A_1(u), A_2(v))]$ can be rewritten:

$$\begin{aligned}
&\sum_{a_1,a_2} P[A_1(u)=a_1]P[A_2(v)=a_2]\cdot\phi(a_1,a_2)\\
=&\sum_{a_1,a_2} P[A_1(u)=a_1]P[A_2(v)=a_2]\\
&\qquad\times\left(\frac{P[A_1(u)=a_1\wedge A_2(v)=a_2 \mid v=\sigma_I(u)]}{P[A_1(u)=a_1]\cdot P[A_2(v)=a_2]}\right)\\
=&\sum_{a_1,a_2} P\left[A_1(u)=a_1\wedge A_2(v)=a_2 \mid v=\sigma_I(u)\right] = 1
\end{aligned}$$

where the final equality comes from summing probabilities over the entire domain of the joint distribution. By Jensen's inequality:

$$\begin{aligned}
E[\text{Score}(u,v,\ell,t)] &= E[\ln\phi(A_1(u),A_2(v))]\\
&\leq \ln E[\phi(A_1(u),A_2(v))] = \ln 1 = 0
\end{aligned}$$

We now consider the harder case, when $v = \sigma_I(u)$.

$$\begin{aligned}
&E[\text{Score}(u,v,\ell,t)] = E[\ln\phi(A_1(u),A_2(v))]\\
=&\sum_{a_1,a_2} P[A_1(u)=a_1\wedge A_2(v)=a_2 \mid v=\sigma_I(u)]\cdot\ln\phi(a_1,a_2).
\end{aligned}$$

To simplify notation below, we use $X(a_1, a_2)$ to denote $P[A_1(u) = a_1 \wedge A_2(v) = a_2 \mid v = \sigma_I(u)]$, and $Y(a_1, a_2)$ to denote $P[A_1(u) = a_1] \cdot P[A_2(v) = a_2]$. The distributions X and Y give the probabilities of observing a_1 and a_2 actions in the two domains assuming the users are the same, and are not the same respectively. Using this notation, we have:

$$E[\text{Score}(u,v,\ell,t)] = \sum_{a_1,a_2} X(a_1,a_2)\ln\frac{X(a_1,a_2)}{Y(a_1,a_2)} = I(A_1;A_2)$$

where $I(A_1; A_2)$ denotes the mutual information between A_1 and A_2, which is also equal to $D_{KL}(X \parallel Y)$, the Kullback-Leibler (KL) divergence of Y from X; this quantity is always non-negative.

We have already shown that for $v \neq \sigma(u)$, the expected score is at most 0. On the other hand, for $v = \sigma(u)$, we have the expected score being non-negative. However, we wish to go further and prove that $E[\text{Score}(u, v, \ell, t)]$ is lower bounded by a positive constant in the latter case.

To do this, we apply the following lower bound:

$$\begin{aligned}
I(A_1;A_2) &= X(0,0)\ln\frac{X(0,0)}{Y(0,0)} + \sum_{a_1,a_2\neq(0,0)} X(a_1,a_2)\ln\frac{X(a_1,a_2)}{Y(a_1,a_2)}\\
&\geq X(0,0)\ln\frac{X(0,0)}{Y(0,0)} + (1-X(0,0))\ln\frac{(1-X(0,0))}{(1-Y(0,0))}\,.
\end{aligned}$$

We now evaluate $X(0, 0)$ and $Y(0, 0)$ respectively.

$$\begin{aligned}
X(0,0) &= \sum_{k\geq 0} e^{-\lambda}\frac{\lambda^k}{k!}(1-p_1)^k(1-p_2)^k\\
&= e^{-\lambda(p_1+p_2-p_1p_2)}\sum_{k\geq 0} e^{-\lambda(1-p_1)(1-p_2)}\frac{(\lambda(1-p_1)(1-p_2))^k}{k!}\\
&= e^{-\lambda(p_1+p_2-p1p_2)} \geq 1-\lambda(p_1+p_2-p_1p_2)\,,
\end{aligned}$$

where the last equality is because the preceding sum contains all probabilities from a Poisson distribution with rate parameter $\lambda(1 - p_1)(1 - p_2)$, and the final inequality comes from the Taylor series expansion of e^{-x}. Similarly, we have:

$$\begin{aligned}
Y(0,0) &= \left(\sum_{k\geq 0} e^{-\lambda}\frac{\lambda^k}{k!}(1-p_1)^k\right)\cdot\left(\sum_{k\geq 0} e^{-\lambda}\frac{\lambda^k}{k!}(1-p_2)^k\right)\\
&= e^{-\lambda p_1}e^{-\lambda p_2} = e^{-\lambda(p_1+p_2)}\,,
\end{aligned}$$

This yield a lower bound on the mutual information above:

$$\begin{aligned}
\text{First,}\quad & X(0,0)\ln\frac{X(0,0)}{Y(0,0)}\\
&\geq (1-\lambda(p_1+p_2-p_1p_2))\ln\frac{e^{-\lambda(p_1+p_2-p_1p_2)}}{e^{-\lambda(p_1+p_2)}}\\
&= (1-\lambda(p_1+p_2-p_1p_2))\lambda p_1p_2\,.\\
\text{Then}\quad & (1-X(0,0))\ln\frac{(1-X(0,0))}{(1-Y(0,0))}\\
&\geq \lambda(p_1+p_2-p_1p_2)\ln\frac{(1-e^{-\lambda(p_1+p_2-p_1p_2)})}{(1-e^{-\lambda(p1+p_2)})}
\end{aligned}$$

Combining these terms and applying considerable algebraic manipulation yields the desired result with the appropriate value of K. Please refer to the appendix for this final step. □

5. COMPARISON AND CASE STUDIES

Having established the theoretical guarantees for our algorithm, we now compare its performance to alternative reconciliation algorithms, inspired by the state of the art. We

Dataset	Domain	Users	Checkins	Median Checkins	Locations	Date Range
FSQ-TWT	Foursquare	862	13,177	8	11,265	2006-10 – 2012-11
	Twitter	862	174,618	60.5	75,005	2008-10 – 2012-11
IG-TWT	Instagram	1717	337,934	93	177,430	2010-10 – 2013-09
	Twitter	1717	447,366	89	182,409	2010-09 – 2015-04
Call-Bank	Phone Calls	452	∼200k	∼550	∼3500	2013-04 – 2013-07
	Card Transactions	452	∼40k	∼60	∼3500	2013-04 – 2013-07

Table 1: Overview of datasets used in study. For FSQ-TWT and IG-TWT, number of locations refers to locations at a 4 decimal GPS granularity (position within roughly 10m).

describe our datasets, the baselines we compared against, some of our real-world implementation, and our results.

5.1 Datasets

Studying the cross domain problem is challenging due to the difficulty in obtaining ground truth. We used a total of three datasets (each from different pairs of spatio-temporal domains) to evaluate the performance of Algorithm 1.

Foursquare–Twitter.

Our first dataset, labeled **FSQ-TWT**, links checkins on the location-based social network, Foursquare, to geolocated tweets. This dataset was collected previously in [26]. After selecting users with locations present in both dataset, we obtain 862 users with 13,177 Foursquare checkins and 174,618 Twitter checkins.

This dataset presents an interesting challenge. There is a large imbalance in data, with many more tweets than Foursquare checkins.

Additionally, the domains are somewhat different– whereas Foursquare checkins are typically associated with a user showing what they are currently doing (in particular, eating at a restaurant), tweets are more general and associated with more behaviors. To verify that tweets and checkins were usually not one event forwarded by software across both services, which could make this dataset artificially easy, we looked at if checkins matched *exactly* on time place. Only 260 pairs of checkins (less than 0.3%) had exactly matching GPS coordinates, and of those, none were within 10 seconds of each other. Beyond this, we reduced all coordinates to 4 digits of accuracy (around 10m), removing low level GPS digits that could be used as a "signature".

Instagram–Twitter.

Our second dataset, referred to as **IG-TWT**, links users on the photo-sharing site, Instagram, to the microblogging service, Twitter. We obtained this data in the following manner: First, we download publicly available location data from Instagram, saving user metadata if he or she had at least 5 geotagged photos in their 100 most recently uploaded photos. For each photo, we did not download or save any images, instead only using latitude-longitude pairs, times, and a user identifier. To find more profile IDs to crawl, we used the profile IDs of anyone who commented or "liked" a crawled user's photos. We started this process with the founder of Instagram, a central node whose photos are commented on or receive "like" from a diverse set of users. This process yielded 120K users with 35M checkins (i.e. time, latitude-longitude pairs from a geolocated photo).

On Instagram, a user can associate a single URL with their profile. We analyzed these URLs, looking for URLs which matched Twitter accounts. Of these, we manually examined 50, finding that all profiles were correct matches based on profile name, profile picture, and/or posted photos, when available. Then, using Twitter's API, we crawled all publicly available tweets for those users, again saving latitude-longitude pairs, time, and user identifier for geolocated tweets. This process left us with 1717 matched users, with a total of 337,934 Instagram checkins and 447,366 Twitter checkins.

This dataset promises to be the "easiest", due to the large number of photos and tweets per user (median 93 and 89, respectively). Picture-taking and tweeting appear to be somewhat different behaviors, but related in the sense that both are actions whereby a user communicates an action or message to a larger, public audience. To again verify that tweets and Instagram posts were not one event forwarded to both services via software, we again looked at exact matches in low-level GPS coordinates and time. Only 2415 pairs of checkins (around 0.6% of all checkins) had exactly matching GPS coordinates, and of those, only 2 were within 10 seconds of each other. Again, all coordinates were then reduced to 4 digits.

Cell Phone – Credit Card Record.

Our third and final dataset contains a log of phone calls (referred to as call detail records or "CDR") linked to credit card transactions (referred to as "bank" data) made by 452 users from a G20 country over 4 months from April 1st through July 31st, 2013. We will refer to this dataset as **Call-Bank**. The linking was made by two companies who originated the data, a telecommunications and credit card company, respectively. Each record of a phone call in the CDR data consisted of a phone number, time, and cell tower ID with its latitude-longitude coordinates. Each record of a credit card transaction in the bank data consisted of the latitude and longitude of the geolocated business at which the transaction was made, along with the time and phone number of the credit card owner. These transactions only included in-person visits, as opposed to online or over-the-phone transactions. The two companies hashed the phone number using the same hash function, and associated this hash with the information for that user. This information was then passed to a third party. The researchers from Columbia University accessed this information on a secure, remote server.[6] At no time were the real phone numbers or credit card numbers available or utilized.

[6]The researchers from Google never had access to this data.

The two datasets log location in different ways. For the CDR data, a user could have been anywhere within range of the associated cell tower. The bank data, however, have a more precise localization. To link the two, we compute the Voronoi diagram generated by cells' locations. We then say that a business location is the same as a cell tower if it is contained in this tower's Voronoi cell. Note that this is a clear demonstration of the need for location *bins* (in this case, the Voronoi cells), as introduced in our model.

The original data is extremely sparse, and contains above 70k users common to the two datasets. However, many users have no calls or bank transactions in the same location, because about 80% of users have fewer than 10 transactions, meaning they use their credit card on average roughly once every two weeks. To make the problem more tractable, we used a smaller subset of active users, by discarding those that made fewer than 50 bank transactions throughout the entire span (*i.e.*, keeping those making a transaction on average every 2-3 days). It amounts to a total of 452 users, whose transactions and calls are dispersed throughout a total of over 3500 cell towers.

This dataset promises to be extremely challenging. Phone calls and credit card transactions are very different activities, and it is not expected that they occur for a user in the same place at the same time. Indeed, only 294 of our 452 active users had even at least one location in common across domains.

Summary.

We summarize the statistics on the datasets in Table 1. Note that although our datasets have the same set of users in both domains, our algorithm can run *without* this requirement– our algorithm will simply leave some users unmatched. Although by some standards these datasets are small, their size is comparable to previous studies [26, 19] and it is difficult to obtain cross-domain datasets of greater magnitude while still maintaining high levels of accuracy.

5.2 Prior Algorithms

We compare our algorithm with three state of the art reconciliation techniques, which we briefly describe in the rest of this subsection.

Exploiting Sparsity: The "Netflix Attack".

The first reconciliation technique that we consider is a variation of the algorithm used to de-anonymize the Netflix prize dataset [15]. The Netflix algorithm cannot be applied directly to our setting, but is not hard to adapt. The algorithm first defines a score between users u and v as follows:

$$S(r^1(u), r^2(v)) = \sum_{(l,t)\in r^1(u)\cap r^2(v)} w_l f_l(r^1(u), r^2(v)),$$

where $w_l = \frac{1}{\ln\left(\sum_{v,t} a_2(v,l,t)\right)}$ and $f_l(r^1, r^2)$ is given by

$$e^{\frac{\sum_t a_1(u,l,t)}{n_0}} + e^{-\frac{1}{\sum_t a_1(u,l,t)} \sum_{t:(l,t)\in r^1} \min_{t':(l,t')\in r^2} \frac{|t-t'|}{\tau_0}}.$$

Note that n_0 and τ_0 are unspecified parameters of the algorithms. This score function considers the visits of u to the locations near v's trajectories. In resemblance to the score function in [15], it favors locations that are visited less often, as they are considered more discriminative just like in [9], frequent visits to the same location, and visits that occur shortly before or after v's traces. The algorithm declares a user u with the best score to be a match for a user v if the score of the best candidate and the score of the second best candidate differ by no less than ε standard deviations of all candidate scores - otherwise the user is unmatched. Intuitively, this algorithm is designed to exploit *sparsity*, using unique, rare occurrences in two datasets to link users. For future use, we refer to this algorithm as NFLX.

Exploiting Density: Histogram Matching.

In [23] the authors leverage frequency of visits to location as a fingerprint of individuals across datasets. Let $\Gamma^1_l(u)$ be the fraction of time that user u is in location l in the first dataset and $\Gamma^1(u)$ be the distribution across different locations. For each pair of user u and v the weight $w(u,v)$ between them is defined using the Kullback-Leibler divergence:

$$D\left(\Gamma^1(u)\left\|\frac{\Gamma^1(u)+\Gamma^2(v)}{2}\right.\right) + D\left(\Gamma^2(v)\left\|\frac{\Gamma^2(v)+\Gamma^1(u)}{2}\right.\right).$$

Each edge weight reflects the degree of disparity between two users. This algorithm computes a minimum weight matching for the complete bipartite graph drawn between individuals, as a way to minimize that disparity. In contrast to NFLX, this algorithm relies on the *density* of data, assuming that over time even in different periods a unique histogram of user visits will emerge from a user's behavior. In the remaining we refer to this technique as HIST. Note that other methods use frequency of visits to define similarity, such as [9]. It can be shown under similar assumptions to our model that within the categories of algorithms that only leveraging density, HIST provably provides the minimum error and that it decreases fast as more data are available [22].

Alternative: Frequency-Based Likelihood.

As a third comparison we consider the reconciliation technique introduced in [19], which approximates the likelihood of a visit made in one domain by the frequency of visits for that user in the other domain, hence assuming:

$$P\left(l \mid r^1(u)\right) = \frac{\sum_t a_1(u,l,t) + \alpha}{\sum_{l',t} a_1(u,l',t) + \alpha|L|},$$

where $\alpha > 0$ is a parameter. This regularization, sometimes referred to as Laplacian smoothing, prevents null empirical frequencies from leading to an infinite score. The mapping (that we denote by WYCI after the title of the paper) is then computed as $\sigma(u) = \arg\max_v \prod_{(l,t)\in r^2(v)} P\left(l \mid r^1(u)\right)$. The paper introduces another distance parameter, but later claims it has negligible impact, as we also observe ourselves.

5.3 Implementing Algorithm 1 in Practice

Parameter Estimation.

In our experiments we partition the time interval into 1024, 2048, 3072 and 4096 time bins. In each time bin we de-duplicate visits to the same locations. In the rest of the paper we describe the results for 4096 time bins, although as we show, similar results hold for different binning.

Our algorithm requires knowing the three main parameters p_1, p_2 and $\lambda_{l,t}$ for each bin (l,t). Unfortunately, using single domain observations separately, the problem is ill posed. For instance parameters (p_1, p_2, λ) and $(\frac{p_1}{2}, \frac{p_2}{2}, 2\lambda)$

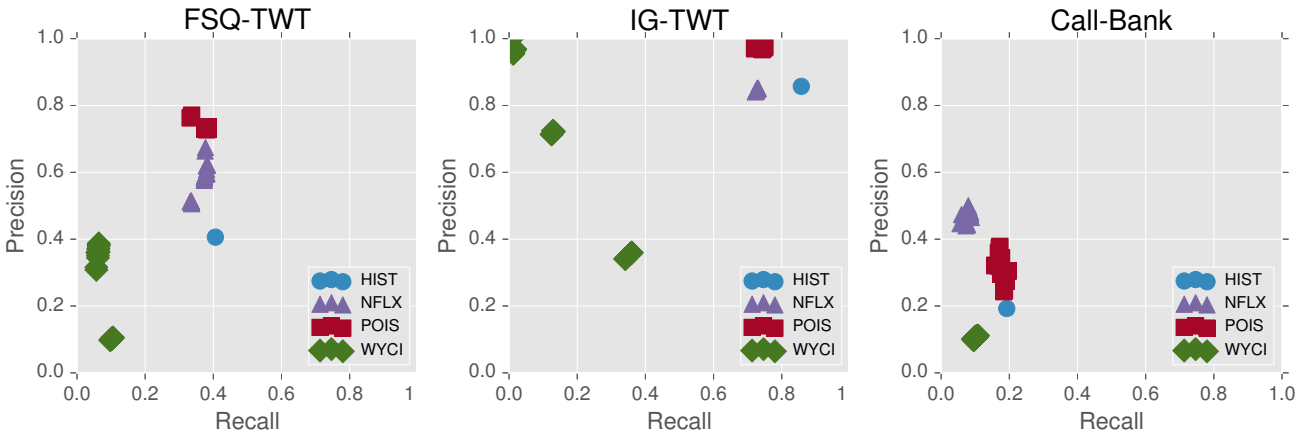

Figure 2: Precision and Recall plots for each dataset.

are simply indistinguishable from a marginal standpoint. On the other hand, by conditioning on bins (l, t) where an action in domain 1 is observed, we have

$$p_2 \approx \frac{\sum_u \sum_t \sum_l \min(a_1(u,l,t), a_2(\sigma_I(u),l,t))}{\sum_u \sum_t \sum_l a_1(u,l,t)},$$

at least in expectation. But this formula requires knowing σ_I, which is precisely the unknown we aim to find. A critical observation we make is that approximating p_1 and p_2 is good enough. All we need is a candidate permutation σ to match user across different domains only for the sake of parameter estimation. In our experiment we use the output of the HIST as our candidate permutation σ. While it is possible to iterate once a new permutation is found to refine even further, we observe in practice that it is not necessary.

Finally, we have to estimate $\lambda_{l,t}$. Unfortunately most datasets are sparse and do not allow separate estimation of $\lambda_{l,t}$ accurately at each time and location. However, we found that assuming that $\lambda_{l,t}$ is constant across time allows a first estimate of a location-normalized popularity given by $\rho_l \approx \frac{\sum_u \sum_t a_i(u,l,t)}{\sum_u \sum_t \sum_l a_i(u,l,t)}$. The parameter λ can then be computed by aggregating observations on all locations together with normalizing factors removed:

$$\lambda \approx \frac{1}{(|U|+|V|)|T|} \sum_l \left(\frac{\sum_{u,t} a_1(u,l,t)}{p_1 \rho_l} + \frac{\sum_{v,t} a_2(v,l,t)}{p_2 \rho_l} \right).$$

Later, we show that estimated parameters are quite robust and resemble ground truth estimated from the true matching.

Additional Feature.

Finally, we introduce for practical settings an "eccentricity" factor ε, which works as follows. After a matching is computed, we only output this edge if the matched candidate's score differs from the second-best by more than ε times the standard deviation of all candidates.

5.4 Comparison on Real Cases

We now turn our attention to experimental performances of our algorithm. In Figure 2, we show the precision recall plots for our algorithm (for different eccentricity values) and for the other three reconciliation techniques: HIST, NFLX and WYCI. For our algorithm, we used estimated parameters and for the other techniques, we used optimal parameters (found via exhaustive search).

There are several interesting observations that we can make on Figure 2. First, on the public dataset FQ-TWT our algorithm outperforms all prior methods (especially in precision). Nevertheless it is interesting to note that the precision of all methods is not ideal, probably due to sparsity of the data.

A second interesting observation is that our algorithm achieves very high precision when the dataset is more rich. In fact when we then turn our attention to our second dataset, the live service (IG-TWT) that we crawled, we obtain almost perfect precision. Note that not all the other techniques, for example NFLX, are able to leverage the denser data, as much.

Finally we test our method on a much more heterogeneous dataset (Call-Bank) that is also more realistic and sensitive. In this setting our algorithm outperforms previous techniques, with none of the previous algorithms able to achieve good precision and recall at the same time.

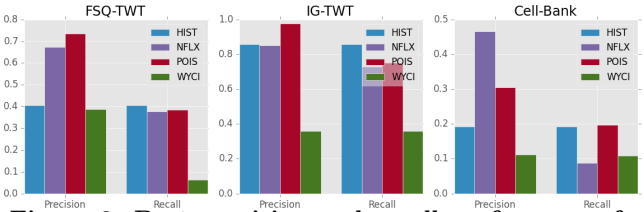

Figure 3: Best precision and recall performance for each technique in various datasets.

In Figure 3 we present the best performances of the four techniques in the three dataset. It is interesting to notice that our algorithm gives the best trade-off between precision and recall. In particular, even if other techniques achieve sometimes better precision or recall our algorithm is not dominated by other algorithms. In fact it is always Pareto optimal in respect of the precision recall curve, and the only algorithm for which this is true.

We now investigate the impact of the number of user checkins on accuracy. In Figure 4, by binning users into quartiles based on number of checkins, and observing the accuracy, we can see that that our algorithm is able to leverage both the amount of the data and its uniqueness. In fact

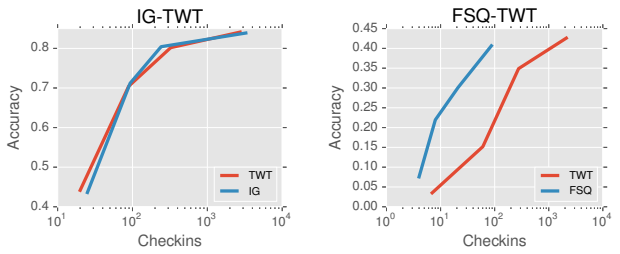

Figure 4: Number of checkins vs. our algorithm's accuracy.

the performance of our algorithm are positively correlated both with the number of checkins and with the entropy of the visited location.

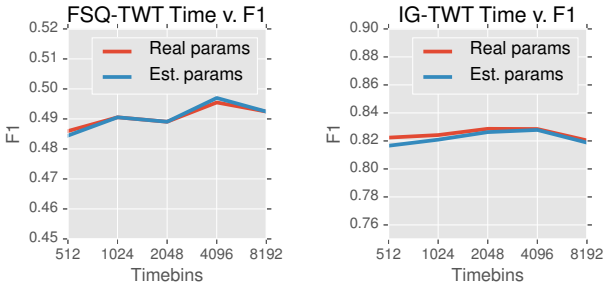

Figure 5: Effect of parameter estimation and time binning on algorithm performance.

We next turn our attention to the impact of our estimated parameters. As mentioned in Sec. 5.3, we cannot know the exact values of p_1, p_2, and $\lambda_{l,t}$. When running our algorithm, we first found a guess at a permutation, and used that matching to estimate the parameters. Comparing this with using the *true* permutation, we can see how far off our guess was and the impact on the algorithm. Fig. 5 shows two lines, one using parameters derived from the real permutation and one using an estimate. Clearly, using the estimate is as good as using the real permutation, and is in fact better at certain time levels. Additionally, this figure shows that there is only a small boost in performance when using differently sized time bins. This is helpful in that it seems the algorithms performance is largely unaffected by choice of parameters.

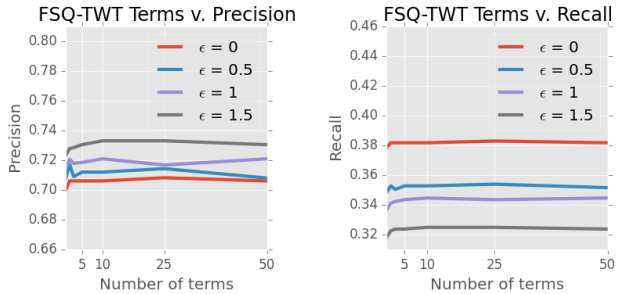

Figure 6: Precision and recall for the FSQ-TWT datasets for different values of the eccentricity and varying numbers of terms of the infinite sum.

Finally we show in Figure 6 the effect of eccentricity and number of terms (of the infinite sum) on performances of our algorithm. The eccentricity is a term that rejects links if other candidates are also very likely. A higher eccentricity should thus correspond with greater precision at the cost of lower recall. In these figures, we can see that this relationship indeed holds, allowing users to potentially find only the strongest matches, perhaps as "seed" links for other algorithms. The number of terms appears to have little effect on algorithm performance, empirically validating our proof that our approximation appears to have little impact on the final result.

6. CONCLUSION

User data is constantly multiplying across an increasing array of websites, apps and services, as they are eager to share part of their behavior with service providers to receive personalized (and free) services. Users may attempt to deal with the privacy implications through partially or inaccurately filled profile information (such as entering a fake name, age, etc.), or using the privacy settings to "lock down" access. However, such methods are of limited use, because commonly collected fields (such as location) that are integral to the service provided may *in themselves* be sufficient to link this account with other accounts of the same user.

In this paper, we present a new approach to characterize when and how such linking is possible. We theoretically justify our algorithm and empirically validate it on real datasets. The results we present, most of them shown for the first time in a cross-domain setting, demonstrate that simple conditions may be sufficient for correct reconciliation and highlight the sensitivity of location data. Several avenues for further research are suggested by these results: Our model assumes very simple behavior by users, modeling them as generating location records independently, and is already quite effective. Can one further exploit patterns inherent to human mobility, such as sleep schedule, commute patterns, working days, and other time dependencies? Is location special, or are there other universal characteristics that are equally meaningful?

7. ACKNOWLEDGEMENTS

This work was supported by NSF under grant CNS-1254035. The authors gratefully acknowledge Mat Travizano and Carlos Sarraute of Grandata for their help with this work.

8. REFERENCES

[1] M. Bayati, M. Gerritsen, D. F. Gleich, A. Saberi, and Y. Wang. Algorithms for Large, Sparse Network Alignment Problems. *Data Mining, 2009. ICDM '09. Ninth IEEE International Conference on*, pages 705–710, 2009.

[2] A. Cecaj, M. Mamei, and N. Bicocchi. Re-identification of anonymized CDR datasets using social network data. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2014 IEEE International Conference on*, pages 237–242. IEEE, 2014.

[3] A. Cecaj, M. Mamei, and F. Zambonelli. Re-identification and information fusion between anonymized CDR and social network data. *Journal of Ambient Intelligence and Humanized Computing*, 7(1):1–14, 2015.

[4] E. Cho, S. A. Myers, and J. Leskovec. Friendship and mobility: user movement in location-based social networks. In *KDD '11: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1082–1090. ACM Request Permissions, 2011.

[5] P. Christen. *Data Matching, Concepts and Techniques for Record Linkage, Entity Resolution, and Duplicate Detection* . Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[6] D. J. Crandall, L. Backstrom, D. Cosley, S. Suri, D. Huttenlocher, and J. M. Kleinberg. Inferring social ties from geographic coincidences. *Proceedings of the National Academy of Sciences*, 107(52):22436–22441, 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, and A. S. Pentland. Unique in the shopping mall: on the reidentifiability of credit card metadata. *Science*, 347(6221):536–539, 2015.

[9] O. Goga, H. Lei, S. Parthasarathi, and G. Friedland. Exploiting innocuous activity for correlating users across sites. In *WWW '13: Proceedings of the 22nd international conference on World Wide Web*, pages 447–458, 2013.

[10] O. Goga, P. Loiseau, R. Sommer, R. Teixeira, and K. Gummadi. On the Reliability of Profile Matching Across Large Online Social Networks. In *KDD '15: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1799–1808. ACM Request Permissions, 2015.

[11] S. Ji, W. Li, M. Srivatsa, J. S. He, and R. Beyah. Structure Based Data De-Anonymization of Social Networks and Mobility Traces. In *ISC Proceedings of the 17th International Information Security Conference*, pages 237–254. Springer International Publishing, 2014.

[12] E. Kazemi, S. H. Hassani, and M. Grossglauser. Growing a graph matching from a handful of seeds. *Proceedings of the VLDB Endowment*, 8(10):1010–1021, 2015.

[13] N. Korula and S. Lattanzi. An efficient reconciliation algorithm for social networks. *Proceedings of VLDB*, 7(5):377–388, 2014.

[14] D. Koutra, H. Tong, and D. Lubensky. BIG-ALIGN: Fast Bipartite Graph Alignment. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 389–398, 2013.

[15] A. Narayanan and V. Shmatikov. Robust De-anonymization of Large Sparse Datasets. *Security and Privacy, 2008. SP 2008. IEEE Symposium on*, pages 111–125, 2008.

[16] A. Narayanan and V. Shmatikov. De-anonymizing Social Networks. *Security and Privacy, 2009 30th IEEE Symposium on*, pages 173–187, 2009.

[17] P. Pedarsani and M. Grossglauser. On the privacy of anonymized networks. In *KDD '11: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1235–1243. ACM Request Permissions, 2011.

[18] C. J. Riederer, S. Zimmeck, C. Phanord, A. Chaintreau, and S. M. Bellovin. I don't have a photograph, but you can have my footprints.: Revealing the Demographics of Location Data. In *COSN '15: Proceedings of the third ACM conference on Online social networks*, pages 185–195. ACM, 2015.

[19] L. Rossi and M. Musolesi. It's the Way you Check-in: Identifying Users in Location-Based Social Networks. *COSN '14: Proceedings of the 2nd ACM conference on Online social networks*, pages 215–226, 2014.

[20] M. Srivatsa and M. Hicks. Deanonymizing Mobility Traces: Using Social Networks as a Side-Channel. *CCS '12: Proceedings of the 2012 ACM conference on Computer and communications security*, pages 628–637, 2012.

[21] L. Sweeney. k-anonymity: a model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5):557–570, 2002.

[22] J. Unnikrishnan. Asymptotically Optimal Matching of Multiple Sequences to Source Distributions and Training Sequences. *Information Theory*, 61(1):452–468, 2015.

[23] J. Unnikrishnan and F. M. Naini. De-anonymizing private data by matching statistics. In *Communication, Control, and Computing (Allerton), 2013 51st Annual Allerton Conference on*, pages 1616–1623. IEEE, 2013.

[24] L. Yartseva and M. Grossglauser. On the performance of percolation graph matching. In *COSN '15: Proceedings of the third ACM conference on Online social networks*, pages 119–130. ACM Request Permissions, 2013.

[25] H. Zang and J. Bolot. Anonymization of location data does not work: a large-scale measurement study. In *MobiCom '11: Proceedings of the 17th annual international conference on Mobile computing and networking*, pages 145–156. ACM Request Permissions, 2011.

[26] J. Zhang, X. Kong, and P. S. Yu. Transferring heterogeneous links across location-based social networks. In *WSDM '14: Proceedings of the 7th ACM international conference on Web search and data*

mining, pages 303–312. ACM Request Permissions, 2014.

[27] Y. Zhong, N. J. Yuan, W. Zhong, F. Zhang, and X. Xie. You Are Where You Go. In *WSDM '15: Proceedings of the 8th ACM international conference on Web search and data mining*, pages 295–304. ACM Press, 2015.

9. APPENDIX

9.1 Proof of Theorem 1

We first show that each of the 2 factors in the denominator of $\phi(a_1, a_2)$ can be replaced by the corresponding truncated sum while affecting its value by at most $1 + 1/C^2$. Since the numerator is decreased by truncation, this establishes the upper bound on $\phi'(a_1, a_2)$. We then show that for the numerator of $\phi(a_1, a_2)$, the difference between the infinite sum and its truncated version is at most $1/C$ times the first term in this sum. Since the denominator is decreased by truncation, this establishes the lower bound on ϕ'.

To obtain the upper bound, we first consider the factor $\sum_{k=a_1}^{\infty} \frac{\lambda^k}{k!}\binom{k}{a_1}(1-p_1)^{k-a_1}$ in the denominator. Expanding the binomial coefficient and pulling common terms outside the summation, this factor can be written as:

$$\frac{\lambda^{a_1}}{a_1!}\sum_{k\geq a_1}\frac{\lambda^{k-a_1}(1-p_1)^{k-a_1}}{(k-a_1)!} = \frac{\lambda^{a_1}}{a_1!}\sum_{k\geq 0}\frac{\lambda^k(1-p_1)^k}{k!}$$

Note that first term in this revised sum evaluates to 1, the term of index $\ln C$ evaluates to $\lambda^{\ln C}(1-p_1)^{\ln C}/(\ln C)! \ll \frac{1}{C^2}$, and the sum of all terms from $\ln C$ onward are at most $\frac{\lambda^{\ln C}(1-p_1)^{\ln C}/(\ln C)!}{(1-\lambda)}$ (upper bounding the infinite sum with a geometric series). Since $\lambda < 1/2$, we conclude that the sum of all terms from index $\ln C$ onward are less than $1/C^2$ times the first term.

The truncated sum for the second factor in the denominator can be bounded identically, giving us the desired upper bound on $\phi'(a_1, a_2)$.

It remains only to establish the lower bound by bounding the truncated numerator. We assume without loss of generality that $a_1 \geq a_2$. Expanding the binomial coefficients in the definition of the numerator of $\phi(a_1, a_2)$ and pulling common terms outside the summation, we can rewrite the numerator as:

$$\frac{\lambda_1^a(1-p_2)^{(a_1-a_2)}}{a_1!\ a_2!}\sum_{k\geq a_1}\frac{\lambda^{k-a_1}\left((1-p_1)(1-p_2)\right)^{k-a_1}\cdot k!}{(k-a_1)!(k-a_2)!}$$

The first term inside the revised sum is simply $a_1!/(a_1 - a_2)! > 1$. Let i denote the final index in the truncated sum, $a_1 + \max\{\ln C, 2a_1\}$. The ith term is upper bounded by $\lambda^{i-a_1}\cdot\frac{i!}{(i-a_1)!(i-a_2)!}$. If $a_1 \geq 4$, then since $i \geq 3a_1$, it is easy to see that $\frac{i!}{(i-a_1)!^2} < 1/2$. If $a_1 \leq 4$, then since $i - a_1 \geq \ln C \geq 7$, we can note that $\frac{i!}{(i-a_1)!^2} < 1/2$. As $\lambda < 1/2$ and $i > a_1 + \ln C$, the ith term is less than $1/C \cdot 1/2$. Again upper bounding the infinite sum with a geometric series, the sum of all terms from index i onward is less than the ith term divided by $(1-\lambda)$, and hence $< 1/C$. Therefore, the sum of all terms from the ith term onward is less than $1/C$ times the first term, completing the proof.

9.2 Proof of Lemma 2

Recall that in Lemma 2, we proved that $E[\text{Score}(u, v, \ell, t] \leq 0$ for any pair of users u, v such that $v \neq \sigma_I(u)$. For $v = \sigma_I(u)$, we showed that the expected score is lower bounded by:

$$\begin{aligned}
&X(0,0)\ln\frac{X(0,0)}{Y(0,0)} + (1-X(0,0))\ln\frac{(1-X(0,0))}{(1-Y(0,0))}\\
&= X(0,0)\ln\frac{X(0,0)}{Y(0,0)} - (1-X(0,0))\ln\frac{(1-Y(0,0))}{(1-X(0,0))}\\
&\geq (1-\lambda(p_1+p_2-p_1p_2))\lambda p_1p_2 -\\
&\qquad \lambda(p_1+p_2-p_1p_2)\ln\frac{(1-e^{-\lambda(p1+p_2)})}{(1-e^{-\lambda(p_1+p_2-p_1p_2)})}
\end{aligned}$$

To prove that this expression is lower bounded by $(\lambda p_1 p_2)^2 K$, it suffices to prove that:

$$\begin{aligned}
&(1-\lambda(p_1+p_2-p_1p_2))\lambda p_1p_2 -\\
&\qquad \lambda(p_1+p_2-p_1p_2)\ln\frac{(1-e^{-\lambda(p1+p_2)})}{1-e^{-\lambda(p_1+p_2-p_1p_2)})}\\
&\geq (\lambda p_1p_2)^2K
\end{aligned}$$

or equivalently:

$$\begin{aligned}
&(1-\lambda(p_1+p_2-p_1p_2))p_1p_2 - \lambda(p_1p_2)^2K\\
&- \quad (p_1+p_2-p_1p_2)\ln\frac{(1-e^{-\lambda(p1+p_2)})}{(1-e^{-\lambda(p_1+p_2-p_1p_2)})} \geq 0 \quad (2)
\end{aligned}$$

We can simplify the final factor in this inequality as follows:

$$\begin{aligned}
&\ln\frac{(1-e^{-\lambda(p1+p_2)})}{(1-e^{-\lambda(p_1+p_2-p_1p_2)})} = \ln e^{-\lambda(p_1p_2)}\frac{(e^{\lambda(p_1+p_2)}-1)}{(e^{\lambda(p_1+p_2-p_1p_2)}-1)}\\
&= \left(\ln\frac{(e^{\lambda(p_1+p_2)}-1)}{(e^{\lambda(p_1+p_2-p_1p_2)}-1)}\right) - \lambda p_1p_2
\end{aligned}$$

where the first equality came from multiplying the numerator and denominator by $e^{\lambda(p_1+p_2-p_1p_2)}$.
Substituting into Inequality (2), our lemma reduces to:

$$\begin{aligned}
&(1-\lambda(p_1+p_2-p_1p_2))p_1p_2 - \lambda(p_1p_2)^2K\\
&\quad (p_1+p_2-p_1p_2)\left(\ln\frac{(e^{\lambda(p_1+p_2)}-1)}{(e^{\lambda(p_1+p_2-p_1p_2)}-1}) - \lambda p_1p_2\right) \geq 0
\end{aligned}$$

or, equivalently:

$$\begin{aligned}
&p_1p_2(1-\lambda(p_1p_2)K) -\\
&\qquad (p_1+p_2-p_1p_2)\ln\frac{(e^{\lambda(p_1+p_2)}-1)}{(e^{\lambda(p_1+p_2-p_1p_2)}-1)} \geq 0 \quad (3)
\end{aligned}$$

This is hard to simplify directly, so we introduce the following upper bound:

$$\lambda p_1p_2 = \ln\frac{1}{e^{-\lambda p_1p_2}} = \ln\frac{e^{\lambda(p_1+p_2)}}{e^{\lambda(p_1+p_2-p_1p_2)}} \leq \ln\frac{e^{\lambda(p_1+p_2)}-1}{e^{\lambda(p_1+p_2-p_1p_2)}-1}$$

Using Z to represent the quantity $\ln\frac{e^{\lambda(p_1+p_2)}-1}{e^{\lambda(p_1+p_2-p_1p_2)}-1}$ and substituting the new inequality in Inequality (3), we are try-

ing to prove:

$$\begin{aligned}
&p_1p_2(1-ZK)-(p_1+p_2-p_1p_2)Z\geq 0\\
&\Leftrightarrow p_1p_2\geq (p_1+p_2-p_1p_2(1-K))Z\\
&\Leftrightarrow \frac{p_1p_2}{p_1+p_2-p_1p_2(1-K)}\geq Z\\
&\Leftrightarrow e^{\frac{p_1p_2}{p_1+p_2-p_1p_2(1-K)}}\geq \frac{e^{\lambda(p_1+p_2)}-1}{e^{\lambda(p_1+p_2-p_1p_2)}-1}
\end{aligned}$$

Now to conclude the proof we use two inequalities that follows from the Taylor expansions. In particular we have:

$$e^x\geq 1+x+\frac{1}{2}x^2$$

and for $x\in o(1)$:

$$e^x\leq 1+x+x^2$$

Now by assuming that $\lambda\in o(1)$ and by fixing $K=\frac{1}{2}\lambda(p_1+p_2-p_1p_2)^2$ we get:

$$\begin{aligned}
& e^{\frac{p_1p_2}{p_1+p_2-p_1p_2(1-K)}}\geq \frac{e^{\lambda(p_1+p_2)}-1}{e^{\lambda(p_1+p_2-p_1p_2)}-1}\\
\Leftrightarrow\quad & 1+\frac{p_1p_2}{p_1+p_2-p_1p_2+\frac{1}{2}\lambda(p_1+p_2-p_1p_2)^2}+\\
& \frac{p_1^2p_2^2}{2(p_1+p_2-p_1p_2+\frac{1}{2}\lambda(p_1+p_2-p_1p_2)^2)^2}\geq\\
& \frac{\lambda(p_1+p_2)+\lambda^2(p_1+p_2)^2}{\lambda(p_1+p_2-p_1p_2+\frac{1}{2}\lambda(p_1+p_2-p_1p_2)^2)}\\
\Leftrightarrow\quad & 1+\frac{p_1p_2}{p_1+p_2-p_1p_2+\frac{1}{2}\lambda(p_1+p_2-p_1p_2)^2}+\\
& \frac{p_1^2p_2^2}{2(p_1+p_2-p_1p_2+\frac{1}{2}\lambda(p_1+p_2-p_1p_2)^2)^2}\geq\\
& 1+\frac{p_1p_2+\lambda(p_1+p_2)^2}{p_1+p_2-p_1p_2+\frac{1}{2}\lambda(p_1+p_2-p_1p_2)^2}\\
\Leftrightarrow\quad & \frac{\frac{1}{2}p_1^2p_2^2}{p_1+p_2-p_1p_2+\frac{1}{2}\lambda(p_1+p_2-p_1p_2)^2}\geq\lambda(p_1+p_2)^2
\end{aligned}$$

Now by fixing $\lambda<\frac{1}{8}\frac{p_1^2p_2^2}{(p_1+p_2)^2}$ we get:

$$\begin{aligned}
& \frac{\frac{1}{2}p_1^2p_2^2}{p_1+p_2-p_1p_2+\frac{1}{2}\lambda(p_1+p_2-p_1p_2)^2}\geq\lambda(p_1+p_2)^2\\
\Leftrightarrow\quad & \frac{\frac{1}{2}p_1^2p_2^2}{p_1+p_2-p_1p_2+\frac{1}{16}p_1^2p_2^2}\geq\frac{1}{8}p_1^2p_2^2\\
\Leftrightarrow\quad & \frac{1}{4}p_1^2p_2^2\geq\frac{1}{8}p_1^2p_2^2
\end{aligned}$$

So the claim follows.