Towards an Information Theoretic Framework for Location-Based Data Linkage

Bradley Malin and Edoardo Airoldi
Data Privacy Laboratory, Institute for Software Research International

November 2005
CMU-ISRI-05-131

Data Privacy Laboratory
Institute for Software Research International
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

A long-standing challenge for data management is the ability to correctly relate information corresponding to the same entity distributed across databases. Traditional research into record linkage has concentrated on string comparator metrics for records with common, or relatable, attributes. However, spatially distributed data are often devoid of such crucial information for database schema integration. Rather than directly relate schemas, spatially distributed data can be related through *location-based* linkage algorithms, which link *patterns* in location-specific attributes (e.g. visit). In this paper we focus on two fundamental algorithms for location-based linkage and we investigate how different distributions of how entities visit locations influence linkage performance. We begin by studying algorithm accuracy for linking real-world data. We then outline a theoretical framework rooted in information theory that allows us to provide insight into observed phenomena. Our framework also provides a useful basis for studying the performance of location-based linkage algorithms: we analyze two opposing cases where location visit patterns arise from uniform and power distributions of entities to locations. We carry out our investigations under both the assumption of complete and incomplete information. Our findings suggest that low skew distributions are more easily linked when complete information is known. In contrast, when information is incomplete high skew distributions lead to higher linkage rates.

Keywords: databases, data management, record linkage, trails, information theory

Contents

1 Introduction

The ability to link, merge, and integrate data corresponding to a particular entity is a fundamental criterion for successful data management. Support and discovery of linkages assists various critical processes such as data fusion, cleaning, profiling, and deduplication. [2, 10, 23, 31] To facilitate this process, research in data warehousing and relational database management has produced sound architectures for storage, relational modelling, retrieval, and the aggregation of mass amounts of entity-specific data [9]. Yet, these models tend to concentrate on databases where schemas are fully specified, or fuzzy relational schemas are supplied by a user or learned from the databases' attributes. [5, 17, 18, 14] Yet, technological advances have sustained a continuing increase in our abilities to gather and store information at the entity-specific[1] level. [27] As a result, an entity's data is often scattered across databases maintained at a large number of disparate locations with a vast range of schemas. Current methods for database schema matching are time consuming, error prone, and subject to semantic constraints which need to be supplied on a case-by-base basis, and as a result do not scale well in large distributed environments. A less expensive alternative is to take advantage of attributes which express simple events, such as location-based attributes (e.g. visit made to location x). The linkage of an entity's data is achieved via location access patterns (i.e. who went where). An outline of the main goals for this paper are:

- Introduce several simple algorithms for location-based record linkage and provide intuition for algorithm performance in real world scenarios, and
- Abstract real world location access strategies and design a simple framework where such strategies can be studied formally in a controlled setting.

1.1 Record Linkage

An entity's information can be distributed across databases for a number of reasons, including 1) temporal: collection at different times, 2) attributional: collection on different sets of attributes, and 3) spatial: collection at different locations. In the first two data distribution types, linkage methods must be designed to account for incomplete, distorted, or possibly corrupt entity-specific information. The latter can manifest in the form of typographical errors, such as when the name "John" is represented by "Jon". In addition, such methods need to be wary of ambiguous values, as well as variations of an entity's information. For example, the name "Jonathan Doe" and "Jon Doe" could refer to the same entity.

Research into methods which resolve these confusions tend to be studied under the label of "record linkage" [13, 29, 30] or "identity uncertainty". [25] Most techniques for linkage of this type of distribution, concentrate on string comparator metrics [11, 32], and methods for comparing tuples over common [3, 24, 4], or relatable [28], attributes. In certain cases, such as free text analysis, the goal remains the construction of a common set of semantic attributes to input into record linkage techniques. [6, 12] In general, these methods are based on a rich history into the study and characterization of linguistic regularities and syntactic-specific aspects of natural language distributions.

1.2 Location-Based Linkage

However, if an entity's data is distributed across tuples such that the tuples' attributes are unrelated, then the traditional methods of record linkage and string comparators are of no assistance for linkage. In such environments, location-based linkage methods function as a complement to traditional record linkage and provide linkage capabilities that previously appeared impossible. [20] There are two main factors of data

[1] In this paper an *entity* is a unit object of interest which a database records data on, such as a person, a company, a country.

and data collection/release strategies which location-based linkage methods employ. The first factor is related to traditional to record linkage. Specifically, an entities' information of a certain type, such as that studied by the previous methods (e.g. strings, names, etc.) is traceable across locations. The second factor is that an individual's information is collected or released from location where the entity visited. Thus, though there is no common attribute between an data types of say "name" and "ip address", there exists an implicit relationship in the location where the information was collected. Subsequently, as the number of locations collecting and sharing data increases, the number of location-visit patterns, or trails, for an entity's data increase and each specific entity's pattern tends toward uniqueness. It is through these visit patterns that linkages can occur.

Unlike record linkage over common string-based attributes, the formalization of trail linkage has only recently been introduced. As a result, there has been little research to further our understanding into how different distributions regarding an entity's ability for mobility in a multiple location environment affects the linkage capabilities of basic location-based linkage methods. In this paper, we begin to provide insight into the underlying processes governing location-based linkage and how we can formally model its process. We investigate several fundamental types of visit strategies employed by individuals in real world populations. These strategies have been observed in both the physical and the virtual medium (i.e. Internet users), which in comparison yield distinct access strategies. In general, this paper investigates the problem of trail linkage through several types of entity to location distributions.

To study location access in a more controlled fashion, based on observed and known behaviors of specific populations, we simulate location visit distributions for different types of strategies. Based on readily available data, for the physical world we consider the sets of hospitals that diseased individuals choose to visit for treatment. In this environment, we find that individuals tend to visit locations on according to a uniform or weak Gaussian distribution. In comparison, online users tend to visit locations in a pattern of high skew that adhere to power function (*i.e.* self similarity) laws. We simulate both of these environments and consider how they can be modelled by intuitive and by information theory and simple metrics.

The remainder of this paper is organized as follows. In the following section we review the formal concept of location-based linkage, as well as several simple methods for the linkage process. The methods introduced are algorithmic, and theoretical analysis suggests that linkage capability grows at a certain rate given the number of subjects and locations. However, this theoretical rate is not guaranteed to be observed in the real-world. This we demonstrate with a simple proof of concept using the population of hospital visiting patients. In addition, we show the power law feature of online environments, as well as how highskew populations are generated. Next, we simulate and perform linkage analysis on several types of simulated datasets corresponding to a range of distributions, including various parameterizations of power-law, such as the Zipf distribution, and uniform distributions. In addition, we investigate and formalize the relationship of location-based linkage to the concept of entropy in information theory. Finally, this work addresses limitations and possible extensions to our findings.

2 Problems and Issues

In this section we describe the terminology, data structures, and algorithms studied in this paper.

We review and generalize a simple model of location-based linkage for an environment where multiple locations collect similar types of data as follows. [20] Let L be a set of locations collecting data and E be a set of entities contributing data. Each location $l \in L$ stores data in a table $\tau_l^A(A_1, \ldots, A_n)$, with attributes $\{A_1, \ldots, A_n\}$. Each tuple $t[a_1, \ldots, a_n] \in \tau_l^A$ references a unique entity in E, where $a_1 \in A_1$, $\ldots, a_n \in A_n$. Without loss of generality, we consider an environment where all locations' tables have an equivalent of attributes. The attributes of disparate locations tables need not be equivalent, but they must be relatable.

E	Set of entities.
L	Set of locations.
τ_l^A	Location l's table, with attribute set A.
$t[a_1, \ldots, a_n]$	Tuple t's values for attributes $A_1 \ldots A_n$.
X	Table with location specific attributes.
$trail(x, \mathsf{X})$	Trail of tuple y in location-based table X.
$x \preceq y$	x is a subtrail of y
$x \succeq y$	x is a supertrail of y

Table 1: Summary of symbols.

τ_1^X

Name
John
Mary

τ_1^Y

IP Address
128.2.41.234
167.92.182.1
114.32.70

τ_2^X

Name
Bob
Kate

τ_2^Y

IP Address
128.2.41.234
32.221.5.15
167.92.182.1

τ_3^X

Name
John
Mary
Kate

τ_3^Y

IP Address
32.221.5.15
167.92.182.1
114.32.70

τ_4^X

Name
Bob
Kate

τ_4^Y

IP Address
32.221.5.15
128.2.41.234
114.32.70

Table 2: Sample tables for four locations and two types of information.

Several tables are depicted in Table 2. All locations can collaborate and share their tables to construct a location-representative table X, referred to as a track. Basically, track X is the resulting join from linking all locations tables over a set of related attributes. This join can be constructed from traditional record linkage algorithms for tables with common attributes. [31, 29] In addition to being linked, a notable aspect of this table is that the latter $|L|$ attributes are "location-specific" attributes, such that the value of a particular tuple specifies the presence or absence of the tuple's underlying entity's presence or absence at a location. For a tuple t, the location-based attributes comprise the trail of the data, or $trail(x, \mathsf{X})$. Examples of tracks and trails are depicted in Table 3. The number of tuples in a track is depicted by its cardinality $|\mathsf{X}|$.

The basis behind trail linkage is there exists two different types of data collected at the set of locations in the environment. Thus two different types of tracks, X and Y, can be constructed, which consist of the following necessary conditions. The first condition is that both tracks are drawn from the same population of entities E. The second condition is the non-location-based attributes from X are unrelatable to those from Y . The main distinguishing feature of trail linkage algorithms is their characterization of data completeness. A track X is said to be *unreserved* to a track Y, if for every entity, the data trails corresponding to the entity in both tracks are equivalent.

In some situations, a location can collects data of both types, but it undercollects (or underreports) data of one type (i.e. the data is not in the location's table). In this case, track X is said to be *reserved* to track Y if the trail of each entity in matrix X, $trail(x, \mathsf{X})$ can be transformed into the entity's corresponding $trail(y, \mathsf{Y})$ in matrix Y by flipping only Boolean values of 0 to 1. When this transformation can be performed, $trail(x, \mathsf{X})$ is said to be a subtrail (represented with the $\preceq$ symbol) of $trail(y, \mathsf{Y})$. Similarly, $trail(y, \mathsf{Y})$ is said to be the supertrail of $trail(x, \mathsf{X})$, or $trail(y, \mathsf{Y}) \succeq trail(x, \mathsf{X})$. Figure 3 provides an example of location-based tracks where track X is reserved to matrix Y. In this example, both *trail*(*Mary*, X) $\preceq$ *trail*(*167.92.182.1*, Y) and *trail*(*Mary*, X) $\preceq$ *trail*(*114.32.40.81*, Y).

X

Name	l_1	l_2	l_3	l_4
John	1	0	1	1
Mary	1	0	1	1
Bob	1	0	1	1
Kate	1	0	1	1

Y

IP Address	l_1	l_2	l_3	l_4
128.2.41.234	1	1	0	1
167.92.182.1	1	1	1	0
32.221.5.15	0	1	1	1
114.32.70.81	1	0	1	1

Table 3: Two sample location-based tracks from four location's tables. X **is reserved to** Y.

Here we summarize two basic linkage problems: *unreserved trail linkage* and *reserved trail linkage*.

Problem 1 *(Unreserved Trail Linkage) Given two unreserved tables* X, Y with location-based attributes referencing a common set of entities E, find the set of tuple pairs $\langle x, y \rangle$, $x \in \mathsf{X}$, $y \in \mathsf{Y}$, such that x and y reference the same entity $e \in E$. □

Intuitively, unreserved trail linkage corresponds to the case when both types data of the entity is recorded at every location visited. Thus, a linkage in this scenario is established when every location-based value in the trails of tuples x and y are equivalent.

Problem 2 *(Reserved Trail Linkage) Given two tables* X, Y with location-based attributes referencing a common set of entities E, such that X is reserved to Y , find the set of tuple pairs $\langle x, y \rangle$, $x \in \mathsf{X}$, $y \in \mathsf{Y}$, such that x and y reference the same entity $e \in E$. □

Basically, the reserved trail linkage problem is similar to the previous problem, with one caveat. Since one table is reserved, the trails need not be equivalent to be linked. Specifically, trail x can be a subtrail of trail y and a correct link can still be established.

3 Methods

In this section we describe two fundamental algorithms to perform location-based record linkage, that deal with both the case of complete information and the case of incomplete information, and we present an application to medical data. We then subsume location-based linkage problems under a general framework with roots in information theory, and we study the theoretical performance of our algorithms under uniform and power law location access strategies and different assumption about the quality of data.

3.1 Location-Based Record Linkage Algorithms

Linkage of $trail(x, \mathsf{X})$ to $trail(y, \mathsf{Y})$ occurs when $trail(x, \mathsf{X})$ is correctly matched with $trail(y, \mathsf{Y})$. The REIDIT (RE-identification of Data in Trails) links only true linkages when location-based attributes consist of Boolean values, where 1 and 0 represent the presence and absence of information at a location, respectively. [20]

The first trail linkage algorithm is known as REIDIT-Complete (REIDIT-C), the pseudocode of which is provided in Algorithm 1. REIDIT-C performs exact matching on the trails in tables X and Y. It assumes that

Algorithm 1 REIDIT-C (X, Y)

```
Assumes:  X and Y are unreserved
$REID \Leftarrow \emptyset$
for $x = 1$ to |X| do
  if there is one and only one $y$, such that $(trail(x, \mathsf{X}) \equiv trail(y, \mathsf{Y}))$ then
    //Link x and y and remove both from further consideration
    $REID \Leftarrow \langle(y, \mathsf{Y})$, $(n, \mathsf{X})\rangle \cup$ REIDIT-C(X-$x$, Y-$y$)
  end if
end for
return $REID$
```

Algorithm 2 REIDIT-I (X, Y)

```
Assumes:  X is reserved to Y
$REID \Leftarrow \emptyset$
for $n = 1$ to |X| do
  if there is one and only one $y$, such that $trail(n, \mathsf{X}) \preceq \mathrm{trail(y, \mathsf{Y})}$ then
    //Remove n and y from further consideration
    $REID \Leftarrow \langle(y, \mathsf{Y})$, $(n, \mathsf{X})\rangle \cup$ REIDIT-I(X- $n$, Y- $y$)
  end if
end for
if |X| $\equiv$ |Y| then
  for $m = 1$ to |Y| do
    if there is one and only one $x$, such that $trail(m, \mathsf{Y}) \succeq \mathrm{trail(x, \mathsf{X})}$ then
      //Remove x and m from further consideration
      $REID \Leftarrow \langle(m, \mathsf{Y})$, $(x, \mathsf{X})\rangle \cup$ REIDIT-I(X- $x$, Y- $m$)
    end if
  end for
end if
return $REID$
```

both X and Y are unreserved. For every tuple $x \in X$, REIDIT-C determines if there is only one equivalent trail in Y. If there is, then a unique linkage is made, however, when there is ambiguity no linkage is made.

The REIDIT-Incomplete (REIDIT-I) algorithm, pseudocode of which is provided in Algorithm 2, is applicable when one table is reserved to the other. For each trail in the table containing incomplete trails, the set of its supertrails from the table containing complete trails are found[2]. If there is only one supertrail, then a correct trail linkage has occurred. The linked trails from X and from Y are removed. Processing continues until no more linkages can be made.

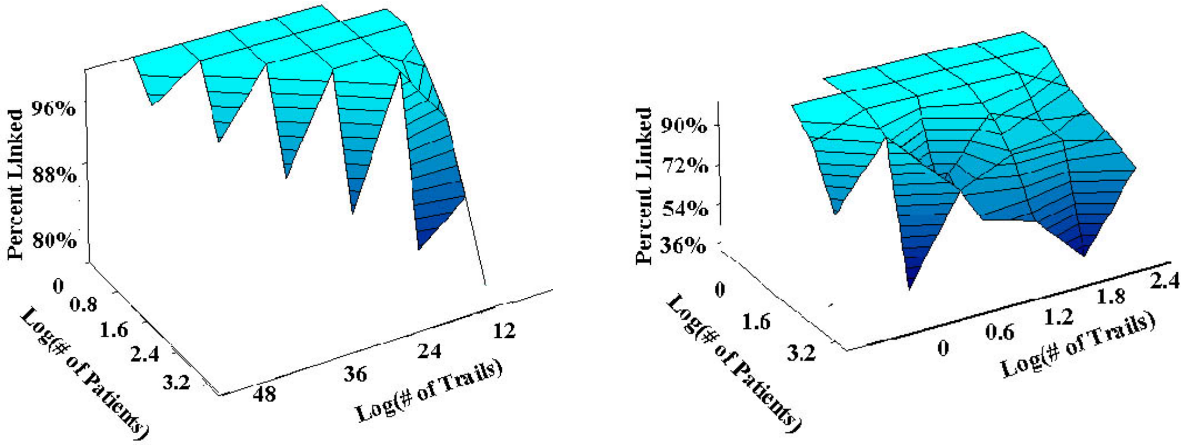

Figure 1: Maximum (left) and observed (right) linkage capability as a function of system size for genetic data and hospital discharge data trails. As the shade of the surface plot becomes darker, linkage capability decreases.

4 Experiments

For both REIDIT-C and REIDIT-I, the maximum number of trail linkages is dependent both on the number of unique permutations of a binary string and, continuing the examples above, on the location access profiles of individuals. Given a table X, containing references to entities and the locations visited, and a set of data-collecting locations L, the number of linkages is bounded above by the maximum number of unique patterns, $2^{|L|} - 1$. If the number of records in the table X is $|X| \leq 2^{|L|}$, then the maximum number of trail linkages is at most $|X|$, the number of subjects, which implicates that all trails could possibly be linked. However, when $|X| > 2^{|L|}$, the maximum number of trail linkages is bounded by the exponential, or $2^{|L|} - 1$ and it is be impossible to link all trails.

Though linkage capability scales exponentially in a perfect environment, such growth is not guaranteed to be achieved in the real world. One of the main reasons is that individual entities are not random agents who generate binary strings with uniform probabilities. On the contrary, research in many diverse areas, including demography, e-commerce, and web personalization suggests that there are trends in the manner that individuals choose locations to visit. This aspect of non-random trails we validate with two real world different datasets.

[2] We assume that one of the tales may have incomplete information, but reports it properly, e.g., a zero means absence and another code is used to report missing data.

4.1 Case Study: Illinois Medical Records

The first dataset consists of hospital discharge data for the state of Illinois from 1990 to 1997. [15] In this dataset, the entities are hospital patients and the locations are in-patient hospitals. In the medical community, these algorithms have been applied to link data on entities distributed across hospitals. For the proof of concept analysis presented below we validate these findings on a sampled entity population where the number of entities is on the order of 10,000 patients and the total number of locations visited is over 200 hospitals.

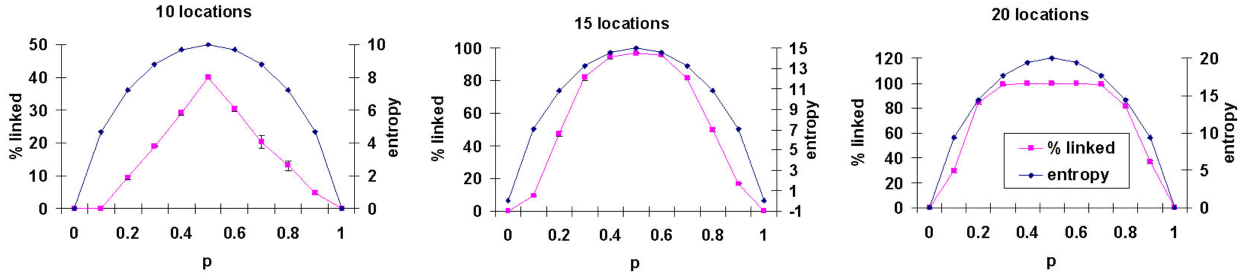

Figure 2: REIDIT-C mean linkability of simulated entities distributed to 10 (left), 15 (center), and 20 (right) locations according to uniform distribution. Error bars correspond to one standard deviation of the simulated populations. The upper and lower lines correspond to entropy and percent of entities linked, respectively.

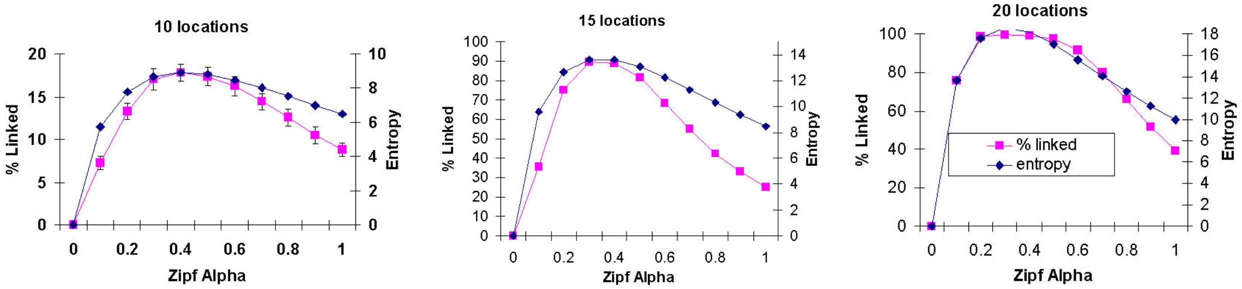

Figure 3: REIDIT-C mean linkability of simulated entities distributed to 10 (left), 15 (center), and 20 (right) locations according to Zipf distribution. Error bars correspond to one standard deviation of the simulated populations. The upper and lower lines correspond to entropy and percent of entities linked, respectively.

The ability for a trail linkage algorithm to discover a unique trail is dependent on more than the distribution of entities to locations. In addition, there is a dependence on the actual numbers of entities and locations in the system being analyzed. Intuitively, one would expect that a system with 50 patients and 50 hospitals would yield more linkages than in a system with 50 patients and 20 hospitals. A similar result would be expected if we varied the number of patients while holding the number of hospitals constant. This was analyzed as follows and found to be true. The number of patients visiting a certain number of hospitals was analyzed with respect percent of the subpopulation linked. We considered the number of patients with respect to the number of available trails that could exist given the number of hospitals. Specifically, the number of trails was computed as the number of possible trails for the number of hospitals visited. For example, consider a population that could visit any of 150 hospitals. If only one hospital were visited, then $\binom{150}{1}$, or 150, possible unique trails exist. If two hospitals were visited, then $\binom{150}{2}$, or 11175, trails possible trails exist. In actuality, the number of unique trails observed is less than the number available to the population.

The general result is depicted in Figure 1, where both the number of patients and the number of trails are considered, on a logarithmic scale for conciseness. While all possible trails are available for each patient to satisfy, only a small fraction are actually observed in the dataset. This is due to the fact that as the number of hospitals visited increases, less patients actually visit this increased number. So, while the number of available trails increases, less trails are used than when a lesser number were available.

In the hospital discharge dataset, the distribution of individuals to locations varies from uniform to approximately Guassian. Yet, these distributions consist of relatively low skew and do not account for the range of distributions in the real world.

4.2 Case Study: Online Browsing Profiles

In addition, we also consider entities in an online environment, where a completely different type of location access phenomena is found. For instance, it has been observed that the popularity of webpages within a particular website varies widely with high-skew [7] - much higher than that observed in the discharge data. The distribution of the "popularity" of locations (*i.e.* the number of distinct people visiting a location) adheres to a power-law function, namely the Zipf distribution. In general for a Zipf, the probability of occurrence of an event (which we consider to be data collection at a location), f_i, is inversely proportional to the event's rank (as determined by its frequency) r_i via the equation $Z \times f_i = r_i^{-\alpha}$, where α is a constant between [0,1] and Z is the number of observations. Given the rank of a set of pages X and the number of users Y who visit those pages, the two dimensional plot of log(X) vs. log(Y) will follow an inverse linear trend. A similar finding has been observed with traffic over websites and, subsequently, has been employed for more efficient search engines [8].

The dataset used to study this phenomena with respect to online information was compiled by the Homenet project at Carnegie Mellon University, who provide families in the Pittsburgh area with Internet services in exchange for the monitoring and recording of the families' online services and transactions. [16] We studied URL access data collected over a two-month period that included 86 households and 144 individuals. Each individual was provided with a unique login and password for fine-grained monitoring. Overall, approximately 5000 distinct website domains and 66,000 distinct pages were accessed. For the following analysis, we employ the simplifying assumption that all websites collect two types of data: 1) identifying information, such as name or address on the purchaser at the time of purchase; and, 2) the IP address of computers visiting their site on each visit. For our studies, we consider f_i to be the probability that an individual visits website i and Z is the set of households in the dataset. To determine if the online dataset is representative of a real world environment, we analyzed the traffic at each domain with respect to the number of distinct visitors. In comparison to a Zipf distribution of $\alpha = 0.6$, a linear fit of observed log frequencies to expected log frequencies yields a correlation coefficient of 0.98. As such, it can be inferred that the high-skew location access trend holds in the online environment.

4.3 Simulation Study: Uniform and Power-Law Location Access Strategies

Zipf distributions explain high skew environments. For example, consider an environment where L is a set of locations and S is a population of subjects visiting those locations. The probability that any particular entity visits location $l_i \in L$ is equal to $r_i^{-\alpha}$, where r_i is the rank of l_i's popularity and α is a coefficient between 0 and 1. When α equals 1, then the distribution is a true Zipf and when $\alpha < 1$ the Zipf distribution is said to be in a generalized form. Given the high skew of the distribution, the log-log plot of "number of visitors" to "location rank" is linear, while the coefficient functions as a dampening factor on the slope of the plotted curve.

There are many aspects of location-based information which influence the linkage capability of a system.

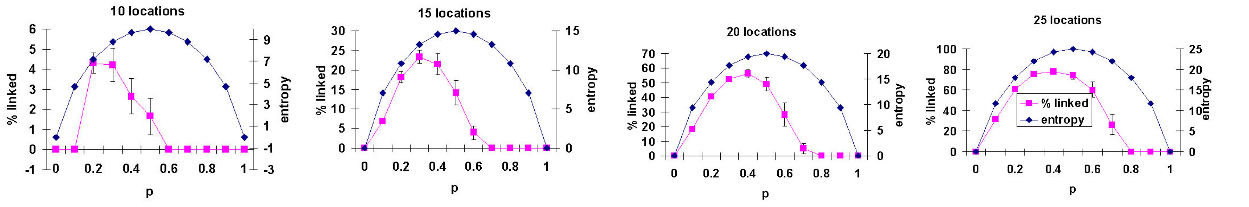

Figure 4: REIDIT-I mean linkability of simulated entities distributed to according to uniform distribution with probability p.

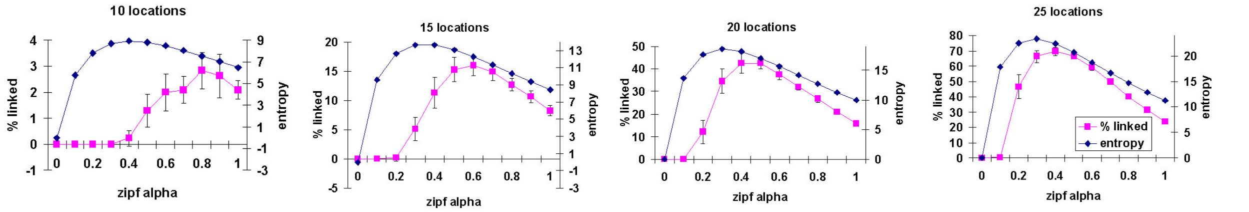

Figure 5: REIDIT-I mean linkability of simulated entities distributed to according to Zipf distribution with probability p.

The main contributing components include the number of subjects, the number of locations, the distribution of subjects to locations, as well as the parameters controlling said distributions. For this research, we concentrate on the number of locations and the distributions guiding subject access to these locations. Thus, for the analyses herein, the number of subjects is fixed as 1000. For these synthetic populations, we generate two types of systems, the first according to uniform access and the second using a Zipf access distribution.

Am entity's trail in a uniform distribution is controlled by a single parameter p. Basically, the probability that any arbitrary value $trail(l, t, \mathsf{X})$ equals 1 is p. For our experiments we sample p from the range $[0, 1]$ at equidistant intervals of 0.1 (*i.e.* $p = \{0.1, 0.2, \ldots, 1\}$).[3] Similarly, populations that are guided by the Zipf distribution are generated using the formula described above. As with the uniform distribution, the Zipf is studied by varying the parameter α over the same interval $[0, 1]$, and sample points, as the p parameter of the uniform distribution. For each tested data point, such as $\langle |L| = 10, p = 0.3\rangle$, we generate 100 populations. Each population is subjected to either the REIDIT-C or REIDIT-I algorithm. For each distribution type and parameterization, these populations are allocations to sets of locations over the range of 3 to 40 locations.

For the analysis of the REIDIT-I algorithm, we only consider the trails as generated by the aforementioned distributions. In other words, while the reserved to track can contain a greater number of subjects' trails than the other track, we work under the assumption that it is a closed population and that the track sizes are equivalent. This aspect of the simulation has certain limitations, but is useful for an exploratory investigation of the REIDIT-I response to varying distributions. Several limitations and concerns of this simplification will be discussed in the following section.

The resulting 10-point plots for REIDIT-C and REIDIT-I are depicted in Figures 2 through 5. In these plots the mean percentage and plus/minus one standard deviation of mean for the 100 simulated populations are depicted in the lower of the two plotted curves. The x-axis corresponds to the parameter of the distribution in question, while the left y-axis corresponds to values of the mean percentage linked. For completeness, and to dispel confusion, the upper curve corresponds to entropy - which will be addressed in

[3] In theory, any number of points on the $[0, 1]$ range will suffice. We choose 10 equidistant points for equal coverage of the distributions in consideration for this research.

a moment. Our analyses are reported and depicted for both the mean and the standard deviation for each point.

4.4 Distributional Effect on Linkage

From the linkage plots, though there is no direct way to compare the parameterizations of the uniform and Zipf distribution, there are several interesting observations that can be made. First with respect to both the REIDIT-C and REIDIT-I linkage algorithms, it is apparent that the uniform distribution consistently yields a larger number of linkages than the Zipf distribution. This is observable, even by visual inspection, by considering the maximum linkability of the distribution type. For example, when considering 10 locations, REIDIT-C links a maximum of approximately 40% of the subjects distributed uniformly (which occurs when p = 0.5), as opposed to around 16% of the subjects that are distributed in Zipf high skew (which occurs when α = 0.4). This finding is consistent across all systems as the number of the locations in consideration is increased.

Second, we consider a less readily observable feature that directly relates to the general linkage capability of a distribution type. To compare distributional types (*i.e.* uniform vs. Zipf), we consider the area under the linkage curve. This is calculated as the total area under the 10-point mean linkage curve (average number of linkages in 100 simulated populations). The results of this calculation with respect to distributions and algorithm results are presented in Figures 6 and 7. Tough the uniform distribution always yields the larger maximum number of linkages, the Zipf distribution is almost always the more linkable when considering all parameterizations. This is obviously so in the case of REIDIT-I linkage, where Figure 7 shows that the Zipf always dominates. Similarly, under REIDIT-C, Zipf is both the initial and inevitable dominant. However, this analysis reveals an unanticipated and intriguing finding. In certain ranges, the uniform distribution is dominant to the Zipf! In Figure 6, this finding is observed between approximately 8 and 18 locations.

The flip in distribution linkage capability dominance occurs for two reasons. First, Zipf dominates when there are not many locations in consideration because it is more difficult to realize complete vectors of all 1's. Second, Zipf dominates as the number of locations increase because it is easier for lesser accessed locations, which is what the newly considered locations are, to convert an unlikely trail into an extremely unlikely trail.

4.5 Calibrating Information Theory for System Linkage

The synthetic trails generated for the experiments are Boolean vectors of 0's and 1's. As such, it seems feasible that each trail can be likened to a measure of information available on a subject. Continuing along this line of thought, it is plausible that the trail linkage capability of a system is related to the Shannon entropy, as defined in information theory. [26] From a general standpoint, the entropy provides a characterization of the total amount of randomness in the distribution of 1's and 0's for a variable. In a sense, the entropy of the track (subjects to locations) is a general predictor of linkage capability of a system.

For our purposes, let X be the track that maps a population of subjects S to a set of locations L. Also, let f_l be the fraction of subjects in S that visit location l. This term is calculated as

$$f_l = \frac{\sum_s^{|X|} trail(l, s, |\mathsf{X}|)}{|\mathsf{X}|}$$

Given this information, the entropy for a single location l, $H(l)$, is equivalent to:

$$H(l) = -f_l log(f_l) - (1 - f_l) log(1 - f_l)$$

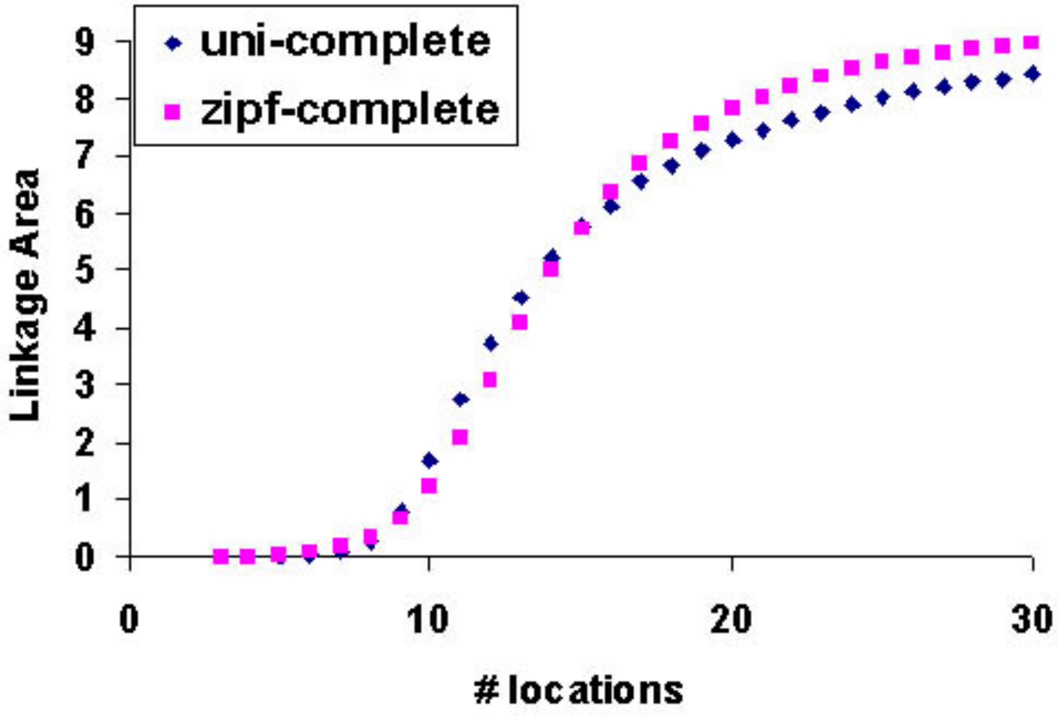

Figure 6: Area under the mean linkage curves for simulated populations and REIDIT-C linkage. “Zipf” and “uni” correspond to the Zipf and uniform distributions, respectively.

For synthetic populations generated during the experiments, each location is allocated individuals independently. Thus, the entropy measure for the entire system X is computed as $H(\mathsf{X}) = \sum_{l=1}^{|L|} H(l)$.

Both the entropy of the system and the linkage of populations over different distributions produce response curves in terms of how linkage capability is influenced. Visually, the results can be observed in Figures 2 and 4. As stated above, the entropy is the upper line, while actual linkages is the lower line. The scale for the entropy is provided on the right y-axis.

It is apparent from these graphs that there exists a general relational trend between the actual linkage curve (R) and the expected linkage curve as predicted by entropy (E). At the most general level, it is visually verifiable that, as the number of locations increases, the actual linkage curve tends towards the entropy prediction. From a mathematical standpoint, we consider these curves as functions, such that $R(x) = y$.

To determine how E and R relate to each other, we define several basic metrics for comparison. Though it is desirable to use known techniques for comparison, the curves generated for linkage analysis do not relate to standard probability (or cumulative) distribution function. As a result, there is no statistical or numerical test to compare the resulting curves to one another. Thus, we define several metrics based on intuitive and observable features relating the curves. The first measure is called the *shift* σ of the curves, which measures the distance along the x-axis between the maximum y-value peaks of the two curves. The second measure is called *shape* ψ, which relates the general shape of the two curves to one another. Shape is calculated as the scaled difference between the 10-point plot of E and R. More formally, both metrics are computed as:

$$\sigma(E, R) = |max_x R(x) - max_x E(x)|$$

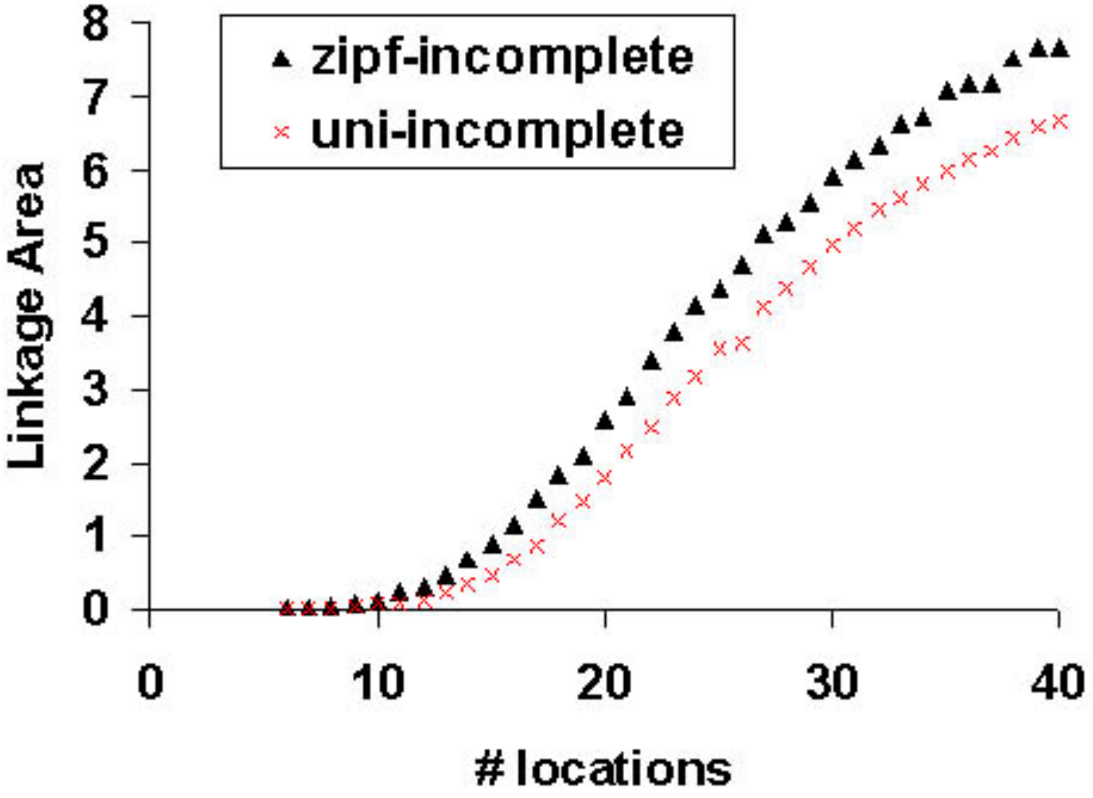

Figure 7: Area under the mean linkage curves for simulated populations and REIDIT-I linkage.

and

$$\psi(E,R) = -\sum_{i=1}^{10} |E(i)\frac{max(R)}{max(E)} - R(i)|$$

The resulting information from these metrics is summarized in Figure 4.5. Both of these metrics are a characterization of features that measure the distance between the distributions. As values for the metrics tends toward 0, the curves converge. As expected, the curves tend toward convergence as the number of locations increase. Yet after convergence begins to come into the line of sight, a counter-intuitive phenomenon occurs. Specifically, phenomenon is that, after a certain number of locations are considered for a particular distribution and trail linkage algorithm, the E and R curves begin to diverge from each other. This is an artifact of the limits of linkage. Notice that in Figures 2 through 5, when a lesser number of locations are considered the linkage curve has a well defined peak. This peak corresponds to the parameter at which the distribution is most amenable to linkage. But this peak is only discernible when less than all of the trails are linked. Thus, when the system is fully linked at multiple parameterizations of the distribution, the linkage curve plateaus at 100% at its peak, while the entropy continues to be well defined. This limit to linkage causes the observed linkage curve to be improperly matched to the entropy of the system. So, in a sense, there is no divergence observed, but rather a limit to independent use of the entropy metric.

The shape metric allows for the discovery of another notable feature that captures how the distribution type influence different trail linkage algorithms. Note that via REIDIT-C, the uniform distribution converges earlier than the Zipf distribution. In contrast, when subjected to the REIDIT-I algorithm, the uniform distribution converges after the Zipf distribution. Ah, a paradox! At first consideration, one would expect that one distribution type, either uniform or Zipf, would converge earlier in both algorithms. However, this paradox results from both how trails are generated under the two distributions as well as how the trail linkage algorithms leverage information. First, consider the linkage algorithms. REIDIT-C looks for a unique bit pattern. In this sense, both 1's and 0's are contribute evenly to the trail linkage process. This is why the linkage curve for the uniform distribution is balanced, or has no shift around the midpoint of p . In other words, the % linked is approximately equivalent for +/-x around the parameterization of p = 0.5. With respect to REIDIT-I tough, a 0 value in a trail functions as fuzzy bit, since it can be used as either a 0 or a 1. Thus, as p tends toward 1, trails with a lesser number of 1's than the $p * |L|$ become extremely difficult to link, and the linkage curve shifts away from high values of p which allow for trails with large amounts of 1's. This is only one part of the problem though. In effect, the Zipf distribution should be hindered by this problem as well. But because the Zipf distribution allows for locations to have different entropy values (due to being a system of single uniform distributions), the Zipf systems ends up revealing more linkages. Thus, the total amount of linkage the Zipf is capable of tends to be greater. If one wanted to validate this claim, it is simple to observe that the average linkage, but not the maximum, for the Zipf is greater than the uniform.

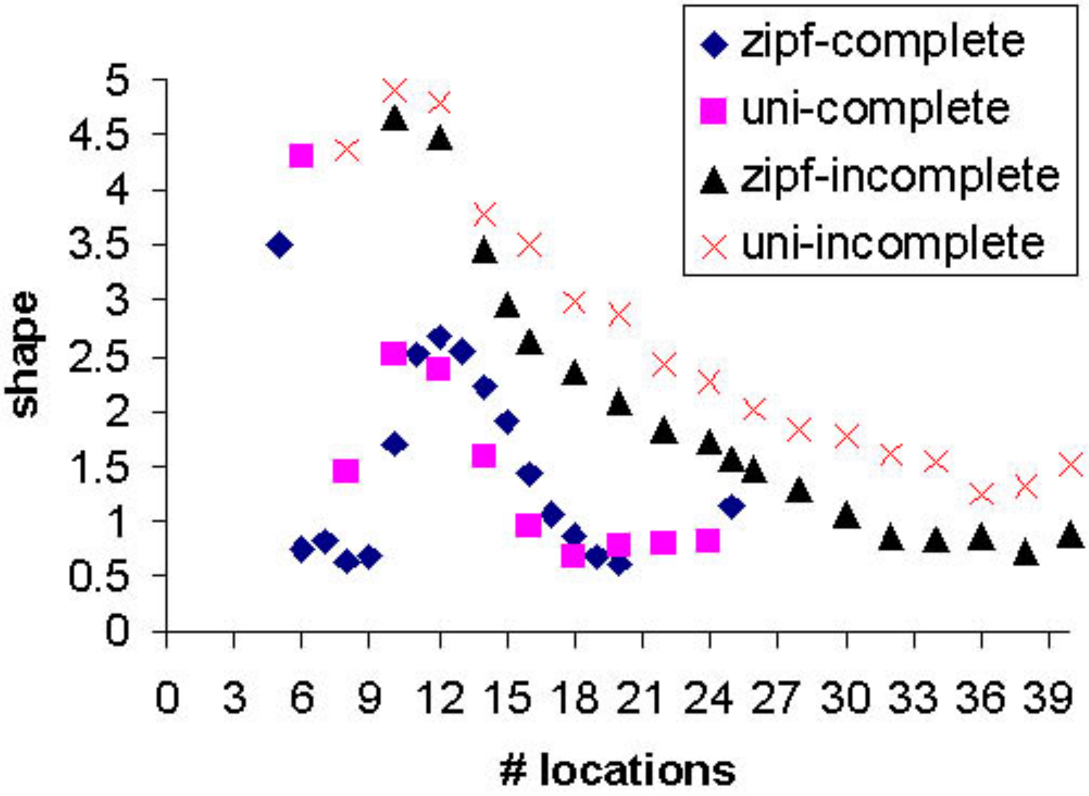

Figure 8: Shape metric for similarity in actual and entropy linkage curves. The valley characterizes when there the actual number of linkages curve begins to plateau.

4.5.1 Probabilistic Intuitions for REIDIT-C

In order to gain some intuition on what goes on during the simulations consider the case of uniform location access strategy with complete information. In this case the parameter governing the access behavior is p, that

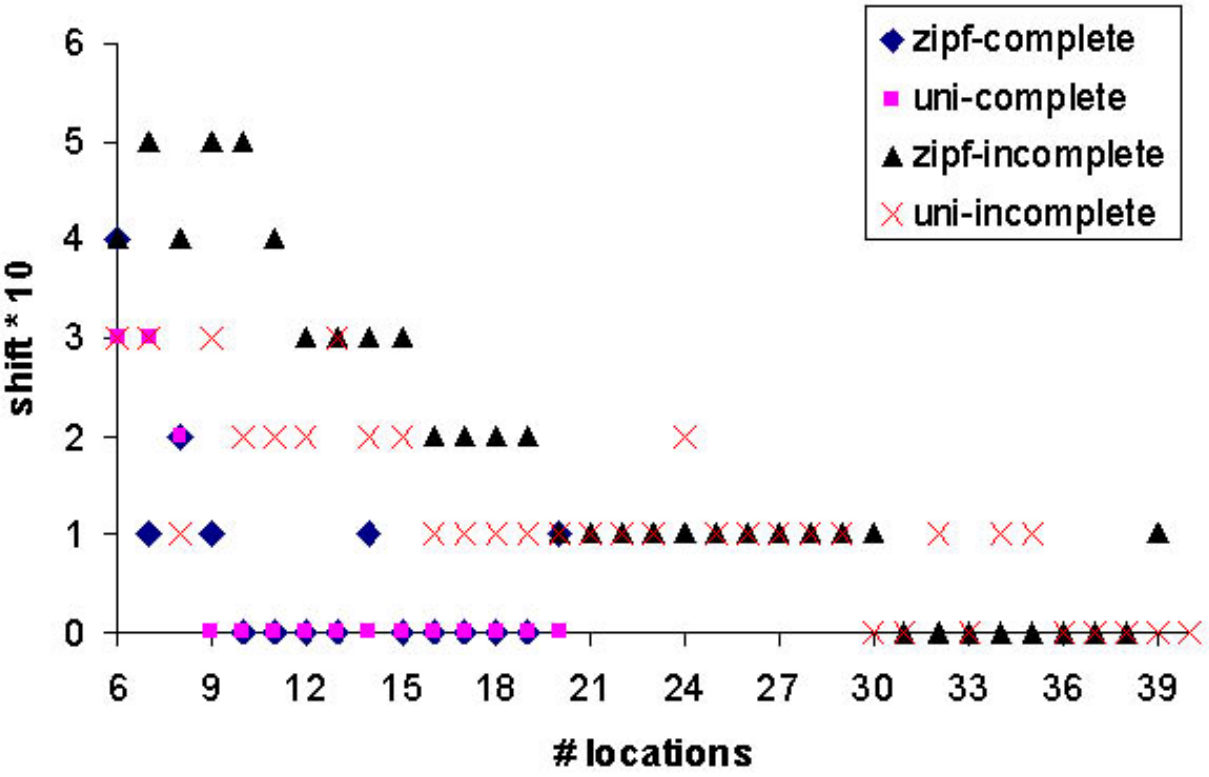

Figure 9: Shift metric for distance between max peak of re-identifiability curves.

is, the probability that a given entity visits a given location. An entity visits different locations independently and its behavior does not depend on what other entities do. Then, the number of locations visited by any given entity, say V, follows a Binomial distribution with parameters L, the number of locations, and p, and the expected number of visits is $E(V) = L \cdot p$. In this scenario, the quantity of interest is the expected number of unique assignments, $E(U)$, since REIDIT-C will accurately link those.

Before getting into the computation of the quantity of interest is worth noting an interesting trade-off, which qualitatively explains the results we observed during the simulations. Specifically, the expected number of unique trails—not assignments—is given by the binomial coefficient $\binom{L}{E(V)} = \binom{L}{Lp}$. The trade-off is between the number of locations, more locations means more possible trails for given a number of visits, and the entropy of the location access strategy as captured by p, as $p \rightarrow 0$ ($p \rightarrow 1$) the expected number of visits decreases (increases) which means less possible trails for a given number of locations. In the simulations above $\binom{L}{Lp}$ grows fast with L for a relatively stable access behavior p and a fixed number of entities, leading to a very sparse mapping of entities to trails, and eventually to an increasing performance of location-based linkage for every value of p.

In the simple case of uniform access and complete information it is possible to analytically characterize the simulated behavior of REIDIT-C and compute the expected number of unique assignments, $E(U)$. In order to do so, we need to define a two-dimensional Markov process that mimics the assignment process of trails to individuals. Define U as the number of trails uniquely assigned to an entity, and define N as the number of trails non-uniquely assigned to an entity, that is, trails assigned to two or more entitiesIt is now sufficient to write down the transition matrices that govern the process $P(U_t, N_t | U_{t-1}, N_{t-1})$ and use that properly to obtain $E(U)$.

For example, for $L = 10$, $p = 0.5$ and $|\mathsf{P}| = 1000$, the expected number of visits for each entity is $E(L) = 5$ and this leads to $\binom{10}{5} = 252$ possible trails. In order to compute the expected number of unique trails assignment, $E(U)$, we have to define the two stationary transition matrices

$$P_U = \begin{bmatrix} 0 & 1 & 0 & \dots & \dots & 0 & 0 \\ 0 & \frac{1}{U} & 0 & \frac{U-1}{U} & \dots & \dots & 0 \\ \vdots & & \ddots & & \dots & \dots & 0 \\ & & & & & & \\ 0 & \dots & \dots & \frac{U-1}{U} & 0 & \frac{1}{U} & 0 \\ 0 & 0 & \dots & \dots & 0 & 1 & 0 \end{bmatrix}$$

and

$$P_N = \begin{bmatrix} 0 & 1 & 0 & \dots & \dots & 0 & 0 \\ 0 & \frac{1}{N} & 0 & \frac{N-1}{N} & \dots & \dots & 0 \\ \vdots & & \ddots & & \dots & \dots & 0 \\ & & & & & & \\ 0 & \dots & \dots & \frac{N-1}{N} & 0 & \frac{1}{N} & 0 \\ 0 & 0 & \dots & \dots & 0 & 1 & 0 \end{bmatrix}.$$

Both P_U and P_N are 253×253 matrices and contain the probabilities of passing from $U = u_t$ to $U = u_{t+1}$ and of passing from $N = n_t$ to $N = n_{t+1}$, respectively. After obtaining the stable distribution of this system, $P(U_\infty, N_\infty)$ we can sum out N_∞ to obtain the desired marginal expectation, $E(U_\infty)$.

5 Discussion

The above analyses provide a wealth of insight into the capabilities of the REIDIT linkage algorithms. In this section we briefly address some findings of particular interest. After discussing revelations from our investigations, we consider some of the limitations of our framework and how future research can extend the framework.

5.1 Distribution Parameter Estimation and Linkage Certainty

One of the most interesting of our findings is that high-skew distributions yield higher overall linkage capability in comparison to low-skew distributions. This is especially the case in light of the fact that low-skew distributions always provide the potential for a larger number of linkages for any given number of locations in the data collection environment. This finding has profound implications on the design of a system of locations that collects multiple types of data, unrelated in their attributes, from a population of entities. This implication is directly related to risk management theory. For instance, if it can be validated that data collection will always be complete, and thus susceptible to the REIDIT-C algorithm, then setting up locations in a manner that allow for high-skew distribution of data to location distributions is always a superior choice to low-skew distributions. Regardless of the parameterization of the high-skew, it will always yield more linkages than the corresponding low-skew distribution.

When the data collection environment provides less certainty in the relationship between tracks of different data types, our strategy for optimizing linkage capability changes. Firstly, the optimal choice for location allocation can be to distribute locations such that data collection is low-skew distributed. However, it should be noted that low-skew distributions should be approached with a bit of trepidation. The main reason is because low-skew distributions bear a greater risk regarding the ability to capture data from the population of entities. Consider an incomplete data collecting environment, in which case REIDIT-I is employed for trail linkage. If the system reverts into a best case location access scenario, such that the parameterization of the distribution maximizes linkage capability, then the low-skew distribution will permit more linkages. Yet, when there is uncertainty as to whether the parameterization will actually yield maximum linkage capability, then the locations are actually better off capturing data according to a high-skew distribution. This is justified by the finding that the average number of linkages, across the range of parameterizations, is greater in the high-skew distributions. Thus, it appears that the question of which type of distribution will yield more linkages is a matter of how confidently the parameter of the entity to location data distribution can be estimated.

The latter concern, regarding certainty in parameter estimation of location access and completeness of data collection poses several complications. These aspects of the problem are not directly addressed in this paper, but they can be considered in extensions to this research. In the following section, we shed some light on these areas and provide suggestions on directions for future research.

5.2 Limitations and Extensions

Though this research provides a theoretical investigation regarding how particular distributions influence trail linkage potential, there are certain caveats of the simulation design which limit the extension of these results. First, to a certain extent, this research is biased in that it does not completely represent real world populations in our simulations. This is because in the real world most entities are not random agents visiting locations independent, but rather they can play an active role in choosing which locations to visit which manifests in the form of correlations between locations in the patterns of access. These patterns can be different than the unique features we exploit in the REIDIT algorithms. Instead, entities tend to visit multiple locations in co-location patterns. As a result of such location access, the linkage capabilities of the synthetic populations used in this research may be inflated.

Therefore, one clear extension to this research is to investigate linkage under different types of collation patterns. Generating synthetic datasets to adhere to complex patterns is still very much an open question in the statistics and data mining communities. However, there is some research that points in the direction of utility, including multivariate distributional theory, genetic algorithms for binary string evolution, and market basket data synthesis. Expanding on the latter, several groups have introduced an approaches for generating synthetic market baskets [1, 19] and, in some respects, the Boolean trail linkage problem can be framed as a market basket problem. Each location can be considered a different product that an individual decides to purchase or not. Thus, if one was to define a set of purchasing patterns, possibly as association rules, then it is possible that useful tracks could be constructed using synthetic market baskets. The foreseeable limitations lie in the fact that market basked generation is useful for studying interesting patterns, but may not facilitate the consideration of outliers, which the trail linkage algorithms are interested in studying. Therefore, before synthetic market basket data can be used, there needs to be some validation performed on the feasibility of how realistically outliers can be represented.

Second, the distributions used in this study consist of homogenous populations, such that visit access to all locations adheres to a single distribution. However, we should ask, “What is the effect of mixture models of populations on linkage?” For instance, to what extent is linkage facilitated when half the population is uniformly distributed while the other half is Zipf distributed? It is possible, and one could speculate on the results, but it is a complex problem that is difficult to reason. As a result, this offers another feasible direction for research into the fundamentals of trail linkage.

6 Conclusions

This research introduced methods and metrics for studying the effect of location access distributions on location-based, or trail, linkage algorithms for data with non-common attributes. Our finding reveal that completeness of information (i.e. whether an entity's leaves data behind at the location) plays a critical role in an algorithm's linkage capabilities. Specifically, low-skew distributions always provide greater potential for linkage when trails are completely truthful (i.e. for each location, the trail reveals if an entity did or did not visit the location). Moreover, when a trail is incomplete, then high-skew distributions yield greater linkage potential. The previous statements account for all parameterizations of location access distributions, but when there exists certainty in the parameterization, then low-skew distributions can yield a greater linkage capabilities. This research was situated in an information theoretic framework, which explicitly models how trail linkage can be represented as a communication problem. Though our models are theoretical and based on simulation, this work provides a foundation for both basic and applied trail linkage research. To extend our models, future researchers should build on our theoretical findings. One direct extension to this work is to study distributions with location dependencies, as well as mixture models of location access distributions.

Acknowledgements

The author thanks insightful discussions with Kathleen Carley and Latanya Sweeney, as well as the members of the Data Privacy Laboratory at Carnegie Mellon University. Bradley Malin is partially supported by a National Science Foundation IGERT Program grant. Both authors are partially supported by the Data Privacy Laboratory in the Institute for Software Research International, a department in the School of Computer Science at Carnegie Mellon University. The opinions expressed in this research are solely those of the authors and do not necessarily reflect those of the National Science Foundation.

References

[1] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proc* 20^{th} *International Conference on Very Large Databases*. Santiago, Chile. 1994.

[2] R. Ananthakrishna, S. Chaudhuri, and V. Ganti. Eliminating fuzzy duplicates in data warehouses. In *Proc* 28^{th} *International Conference on Very Large Databases*. Hong Kong, China. 2002.

[3] I. Bhattacharya and L Getoor. Iterative reord linkage for cleaning and integration. In *Proc* 9^{th} *ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*. Paris, France. 2004.

[4] R. Baxter and P. Christen and T. Churches. A comparison of fast blocking methods for record linkage. In *Proc ACM SIGKDD Workshop on Data Cleaning, Record Linkage, and Object Consolidation.* Washington, DC. 2003.

[5] J. Berlin and A. Motro. Database schema matching using machine learning with feature selection. In *Proc Conference on Advanced Information Systems Engineering*. Toronto, Canada. 2002: 452-466.

[6] V. Borkar, K. Deshmukh, and S. Sarawagi. Automatic segmentation of text into structured records. In *Proc ACM SIGMOD International Conference on Data Management*. Santa Barbara, CA. 2001: 175-186.

[7] L. Breslau, P. Cao, L. Fan, G. Phillip, and S. Shenker. Web caching and Zipf-like distributions: Evidence and implications. In *Proc IEEE INFOCOM - The Conference on Computer Communications*. March 1999.

[8] S. Brin and L. Page. The anatomy of a large-scale hypertextual web search engine. In *Proc* 7^{th} *World Wide Web Conference*. New York, NY. 1998.

[9] S. Chaudhuri and U. Dayal. An overview of data warehousing and OLAP technology. *ACM SIGMOD Record*. 1997; 26(1): 65-74.

[10] S. Chaudhuri, K. Gamjam, V. Ganti, and R. Motwani. Robust and efficient match for on-line data cleaning. In *Proc ACM SIGMOD International Conference on Data Management*. San Diego, CA. 2003: 313-324.

[11] W.W. Cohen, P. Ravikumar, and S.E. Fienberg, A comparison of string distance metrics for name-matching tasks. In *Proc International Joint Conference on Artificial Intelligence*. Acapulco, Mexico. 2003.

[12] W.W. Cohen. Integration of heterogeneous databases without common domains using queries based on textual similarity. *ACM SIGMOD Record*. 1998; 27(2): 201-212.

[13] I.P. Fellegi and A.B. Sunter. A theory for record linkage. *Journal of the American Statistical Association*. 1969; 64: 1183-1210.

[14] N. Friedman, L. Getoor, D. Koller, and A. Pfeffer. Learning probabilistic relational models. In *Proc. International Joint Conference on Artificial Intelligence*. Stockholm, Sweden. 1999: 1300-1307.

[15] State of Illinois Health Care Cost Containment. Data release overview. Springfield. 1998.

[16] R. Kraut, T. Mukhopadhyay, J. Szczypula, S. Kiesler, and B. Scherlis. Information and communication: Alternative uses of the Internet in households. *Information Systems Research*. 2000; 10: 287-303.

[17] A. Laurent. Querying fuzzy multidimensional databases: unary operators and their properties. *International Journal of Uncertainty, Fuzziness, and Knowledge-Based Systems*. 2003; 11: 31-45.

[18] C. Li and X. Wang. A data model for supporting on-line analytical processing. In *Proc ACM Conference on Information and Knowledge Management*. Rockville, MD. 1996.

[19] Y. Li, P. Ning, X.S. Wang, and S. Jajodia. Generating market basket data with temporal information. In *Proc ACM SIGKDD Workshop on Temporal Data Mining*. San Francisco, CA. 2001.

[20] B. Malin. Compromising privacy with trail re-identification: the REIDIT algorithms. *Center for Automated Learning and Discovery Technical Report CMU-CALD-02-118, School of Computer Science, Carnegie Mellon University*. Pittsburgh, PA. 2002.

[21] B. Malin and L. Sweeney. Re-identification of DNA through an automated linkage process. In *Proc of the 2001 American Medical Informatics Association Annual Symposium*. Washington, DC. 2001: 423-427.

[22] B. Malin and L. Sweeney. How (not) to protect genomic data privacy in a distributed network: using trail re-identification to design and evaluate anonymity protection systems. *Journal of Biomedical Informatics*. 2004; 37(3): 179-192.

[23] M. Neiling and H.J. Lenz. Data fusion and object identification. In *Proc SSGRR 2000, International Conference on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet*. L'Aquila, Italy. 2000

[24] Parag and P. Domingos. Multi-relational record linkage. In *Proc ACM SIGKDD Workshop on Multi-Relational Data Mining*. Seattle, CA. 2004: 31-48.

[25] H. Pasula, B. Marthi, B. Milch, S. Russell, and I. Shpitser. Identity uncertainty and citation matching. In *Proc Advances in Neural Information Processing Systems 15*. Vancouver, Canada. 2002.

[26] C.E. Shannon and W. Weaver. The mathematical theory of communication. University of Illinois Press. Urbana, IL. 1949.

[27] L. Sweeney: Information explosion. In: L. Zayatz, P. Doyle, J. Theeuwes, and J. Lane (eds): Confidentiality, disclosure, and data access: theory and practical applications for statistical agencies. Urban Institute. Washington, DC. 2001.

[28] V. Torra. OWA operators in data modeling and re-identification. *IEEE Trans on Fuzzy Systems*. 2004; 12(5): 652-660.

[29] W.E. Winkler. Matching and record linkage. In B.G. Cox et al. (ed) Business Survey Methods. J. Wiley. New York. 1995; 355-384.

[30] W.E. Winkler. Machine learning, information retreival, and record linkage. In *Proc Section on Survey Research Methods, American Statistical Association*. 2000: 20-29.

[31] W.E. Winkler. Data cleaning methods. In *Proc ACM SIGKDD Workshop on Data Cleaning, Record Linkage, and Object Consolidation.* Washington, DC. 2003.

[32] W. Yancey. An adaptive string comparator for record linkage. In *Proc Section on Survey Research Methods, American Statistical Association*. 2004.