STEvent: Spatio-Temporal Event Model for Social Network Discovery

HADY W. LAUW
Institute for Infocomm Research
EE-PENG LIM and HWEEHWA PANG
Singapore Management University
and
TECK-TIM TAN
Nanyang Technological University

Spatio-temporal data concerning the movement of individuals over space and time contains latent information on the associations among these individuals. Sources of spatio-temporal data include usage logs of mobile and Internet technologies. This article defines a spatio-temporal event by the co-occurrences among individuals that indicate potential associations among them. Each spatio-temporal event is assigned a weight based on the precision and uniqueness of the event. By aggregating the weights of events relating two individuals, we can determine the strength of association between them. We conduct extensive experimentation to investigate both the efficacy of the proposed model as well as the computational complexity of the proposed algorithms. Experimental results on three real-life spatio-temporal datasets cross-validate each other, lending greater confidence on the reliability of our proposed model.

Categories and Subject Descriptors: H.3.4 [**Information Storage and Retrieval**]: Systems and Software—*Information networks*; E.1 [**Data**]: Data Structures—*Graphs and networks*; J.4 [**Computer Applications**]: Social and Behavioral Sciences

General Terms: Algorithms, Experimentation, Verification

Additional Key Words and Phrases: Data mining, spatio-temporal databases, social network

ACM Reference Format:

Lauw, H. W., Lim, E.-P., Pang, H., and Tan, T.-T. 2010. STEvent: Spatio-temporal event model for social network discovery. ACM Trans. Inform. Syst. 28, 3, Article 15 (June 2010), 32 pages.
DOI = 10.1145/1777432.1777438 http://doi.acm.org/10.1145/1777432.1777438

This work was done while H. W. Lauw and Ee-Peng Lim were at Nanyang Technological University.
Contact authors' address: H. W. Lauw, Institute for Infocomm Research; email: hwlauw@izr.a-star.edu.sg.

DOI 10.1145/1777432.1777438 http://doi.acm.org/10.1145/1777432.1777438

1. INTRODUCTION

1.1 Motivation and Background

With greater use of mobile computing and Web technologies, comes a greater amount of data about users on the move or on the Internet. Knowingly or unknowingly, users are tracked when they carry wireless devices or when they visit Web pages at different sites. Such user data having both location and time properties are known as movement data. In this article, we aim to mine the phenomenon where movement data suggests social associations among users. We focus on one movement behavior known as co-occurrence, where the fact that two or more users are collocating around the same time implies that there may be some association among them.

Knowledge of social networks finds useful applications in diverse fields such as law enforcement [Krebs 2002; Xu and Chen 2005], business [Domingos and Richardson 2001; Kempe et al. 2003], and social networking [Boyd 2004; Kumar et al. 2004]. It also aids social network-based information-seeking, such as searching for a piece of information held by a friend of a friend, or finding referral to a human expert [Lampe et al. 2006; Yu and Singh 2003; Zhang and van Alstyne 2004]. The social network discovery problem addressed in this article produces social networks that may feed into these diverse applications.

We observe that the term "social network" has been loosely defined. In different fields or applications, the semantics of social associations could be different (e.g., friendship, family, criminal collaboration). Our objective here is not to discover any and all types of social associations; rather, we confine our study to "associations" that can be mined from spatio-temporal data.

Moreover, we only consider a specific type of spatio-temporal data. We use D to denote the collection of tuples. Each tuple $d = \langle a, t, s \rangle$ in D codes for a time $d.t$ and a location $d.s$ at which an actor $d.a$ is observed. Each time value is expressed at a particular atomic unit (e.g., seconds). It is not necessary that the actors' locations are tracked at regular intervals. We use semantic locations, whereby each location has a coarse granularity and has some semantic meaning. Examples include physical locations (e.g., rooms) and cyber locations (e.g., URL addresses). Such locations can be tracked more easily due to their coarse granularity, and other location models such as *xyz* or GPS-based coordinates could be transformed into semantic locations with the help of suitable mapping.

In addition to allowing us to discover social connections based on physical collocations, such data may also reveal social connections based on common interests, as in the case of frequent co-occurrences in cyber locations (i.e., visiting similar URL addresses). Moreover, both the temporal and spatial components of the data can lend further context to the discovered associations. For instance, a temporal analysis of the discovered associations could potentially reveal which associations are strengthening and which others are breaking up. A spatial analysis may reveal that a given person has location-based associations, for example around work, home. The discovered social network could also feed into various social network analysis techniques [Wasserman and Faust 1994], such as to identify the most centrally connected nodes, or subgroups within the network, or other interesting connections between nodes. These analyses would be useful for real-life applications such as contact tracing in the case of an epidemic.

1.2 Objective and Contributions

Our objective is to discover a social network graph $G(G_V, G_E)$ from D, in which the nodes in G_V represent actors and the edges in G_E represent weighted associations between pairs of actors. In addition, the basis upon which these associations are to be inferred is spatio-temporal co-occurrence among interacting actors.

Our contributions in this research can be summarized as follows.

(1) We propose a novel model called *STEvent* to discover social associations based on spatio-temporal co-occurrences. While co-occurrence provides a sound basis for inducing social networks [Faloutsos et al. 2004; Lin and Chalupsky 2003], our specific criterion of co-occurrence based on spatio-temporal events is novel.

(2) We automate social network discovery with the *STEvent* model by designing efficient algorithms to derive the spatio-temporal events and to compute the strength of associations among actors. The social network discovery problem has a quadratic complexity with respect to the number of actors. When computing the link between each pair of actors involves processing a long time series of location data, it is vital to have efficient computational solutions for discovering the overall social network.

(3) Our model and algorithms have been extensively tested through experiments on two proprietary real-life datasets, as well as one publicly available dataset, with encouraging results. The two proprietary datasets are collected from the usage of wireless networks in our campus. The first, *Cyber Location Data*, captures users' movement behavior over cyber locations. The second, *Physical Location Data*, captures movement behavior over physical locations. The public *Reality Mining Data* also captures movement behavior over physical locations.

1.3 Article Outline

The rest of this article is organized as follows. In Section 2, we relate our current work to various prior work in social networks. In Section 3, we describe and formalize our spatio-temporal event model. Following that, in Section 4, we develop a two-phase algorithm based on the event framework. Subsequently, we report our experimental results in three sections: Cyber Location Data in Section 5, Physical Location Data in Section 6, and Reality Mining Data in Section 7. Section 8 concludes the article.

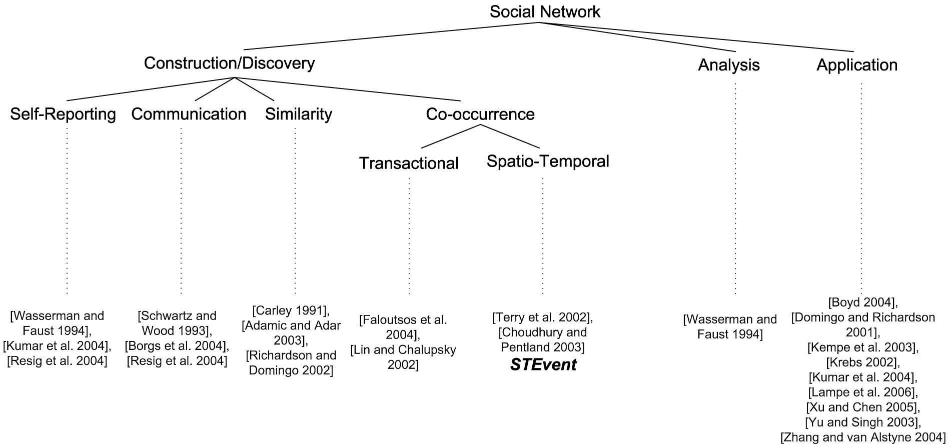

Social Network
Construction/Discovery
Analysis
Application
Self-Reporting
Communication
Similarity
Co-occurrence
Transactional
Spatio-Temporal
[Wasserman and Faust 1994], [Kumar et al. 2004], [Resig et al. 2004]
[Schwartz and Wood 1993], [Borgs et al. 2004], [Resig et al. 2004]
[Carley 1991], [Adamic and Adar 2003], [Richardson and Domingo 2002]
[Faloutsos et al. 2004], [Lin and Chalupsky 2002]
[Terry et al. 2002], [Choudhury and Pentland 2003]
STEvent
[Wasserman and Faust 1994]
[Boyd 2004], [Domingo and Richardson 2001], [Kempe et al. 2003], [Krebs 2002], [Kumar et al. 2004], [Lampe et al. 2006], [Xu and Chen 2005], [Yu and Singh 2003], [Zhang and van Alstyne 2004]

Fig. 1. Taxonomy of work in social networks.

2. RELATED WORK

First, we review terminology frequently used in social network literature. An *actor* is a social entity (e.g., a person). The relationship between a pair of actors is called a *link*, which may be directed or undirected, and binary (present or absent) or weighted. Links could be of various types (e.g., friendship, familial). *Social network* encompasses a set of actors and all the links that could be defined on them. A social network with *n* types of actors is identified as an *n*-mode network. These terms will be used frequently throughout this article.

The variety of work in social networks can be classified according to the taxonomy given in Figure 1.

Construction/discovery. Social network discovery, which encompasses our current work, involves inferring links based on some indicators of potential associations. From our survey, there are four major criteria used in prior work to infer social associations, as listed here.

(1) *Self-reporting* accepts only the links reported by the actors themselves. Reporting links could mean revealing the associates in questionnaires or interviews [Wasserman and Faust 1994], acknowledging the associates in personal profile or home pages [Kumar et al. 2004], or including these associates in Instant Messaging buddy lists [Resig et al. 2004]. Self-reported links are not always mutual or equally weighted in both directions.

(2) *Communication* is another strong expression of relationship. Internet-based communication tools, such as emails [Schwartz and Wood 1993], newsgroups [Borgs et al. 2004], and Instant Messaging [Resig et al. 2004], often leave electronic trails that can be traced and mined. Communication-based links may be directed or undirected (if an exchange is required).

(3) *Similarity* borrows the idea from sociology that people who are more closely related tend to have greater similarity to each other [Carley 1991]. The

problem of finding links between pairs of actors can then be reduced to finding similar pairs. Similarity may be defined in various ways, such as having similar content and linkages in home pages [Adamic and Adar 2003], or sharing similar opinions on common areas of interest [Richardson and Domingos 2002].

(4) *Co-occurrence* is based on the idea that entities occurring together at a frequency higher than that of random chance are likely to have some association between them. One type of co-occurrence is *transactional* co-occurrence, which is supported by discrete transactions. For example, if a Web page is a transaction, two names frequently co-occurring on the same Web pages may be considered related [Faloutsos et al. 2004]. Alternatively, two authors who co-author papers frequently are also likely to be related [Lin and Chalupsky 2003].

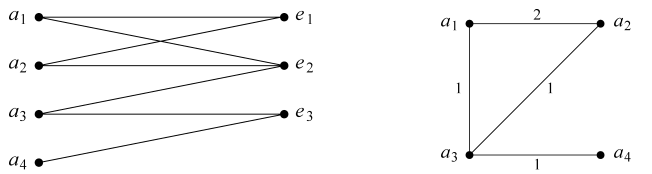
Our approach is based on *spatio-temporal* co-occurrence, which is co-occurrence defined over space and time. That movement data is a possible indicator of social association has been suggested in prior work [Terry et al. 2002; Choudhury and Pentland 2003]. Our current work (*STEvent*) is distinguished from the prior work in the following ways.

(1) It focuses on the analysis of movement data and algorithm development to infer associations, while the others focus on the development of movement-tracking devices.
(2) It generalizes the spatio-temporal co-occurrence beyond movement over physical locations to include other location types such as cyber locations.
(3) It is the first to attempt at verification of the inferred associations through analysis of demographic data.

Analysis/application. The analysis and application of social networks typically assume that a social network has been constructed or discovered beforehand. In social network analysis [Wasserman and Faust 1994], we may attempt to find central actors (those well-positioned within the network), cohesive subgroups of actors, and so on. The analytical outputs of social network analysis are useful in various applications in such diverse fields as law enforcement, business, social networking, and information-seeking. In this article, we focus solely on the construction/discovery aspect, which produces a social network that could feed into any of these analysis or application techniques.

3. STEVENT: SPATIO-TEMPORAL EVENT MODEL

Social network theory on mining relations from events is grounded on the study of 2-mode affiliation networks [Wasserman and Faust 1994]. An event is any social collectivity that actors are affiliated to, including clubs, organizations, companies, social events, and so on. Each instance of affiliation is captured as an actor-event link. The collection of all such links makes up the affiliation network, as depicted by a bipartite graph given in Figure 2(a). In this figure, we have actors a_1 and a_2 affiliated to events e_1 and e_2, a_3 to e_2 and e_3, and a_4 to e_3.



(a) 2-Mode Affiliation Network

(b) 1-Mode Social Network

Fig. 2. Constructing social network from basic events.

Affiliation networks may be used to build a social network among actors. By bringing actors together, an event serves as a basis for interaction to take place. For instance, conferences gather academicians to exchange knowledge and contacts. This leads to the notion of event-supported links between two actors, where the link weight is the number of events common to the two actors. In Figure 2(b), we give the social network consisting of these event-supported links, derived from the affiliation network in Figure 2(a). In this case, only a_1 and a_2 are linked by two events (e_1 and e_2), while the other pairs have only one event each.

We now extend this basic event model to spatio-temporal events. Spatio-temporal events are neither readily given in the data nor clearly defined by standard definitions. Thus, we propose a novel model called *STEvent* that first discovers spatio-temporal events from the data and then uses the events to build a network of associations among actors. In describing the model, we begin with the definition of events, followed by a description of how events may be weighted according to four novel criteria. Subsequently, we explain how locations of multiple levels of granularity may be accommodated in our event model. Finally, we show how our model infers links from the events.

3.1 Event Definition

Given a spatio-temporal database D of actors' whereabouts over time, we define events in terms of interaction between actors that are captured by their co-occurrences. If actors continually move around in a large expanse of space over an extended period of time, instances where actors co-occur at a particular time and location may reveal rendezvous acts. Presumably, rendezvous facilitates interaction and association. Such co-occurrences of two or more actors in time and space are what we term as spatio-temporal events.

Formally, we define a spatio-temporal event as follows. Here, we assume certain user-specified values of semantic location granularity and time duration δ_{max}.

Definition 3.1. A spatio-temporal event is a subset of tuples, $e \subseteq D$, meeting all of the following conditions.

—$\forall d_i, d_j \in e,\ d_i.s = d_j.s$,
that is, tuples are of the same location

—$\forall d_i, d_j \in e,\ |d_i.t - d_j.t| \leq \delta_{max}$,
that is, tuples are separated in time by at most δ_{max}

Table I. Table of Notation

Notation	*Description*
$\mathcal{E}$	the set of all events
e	an event
$e.\mathcal{A}$	event *e*'s actors
$e.s$	event *e*'s location
$e.t^-$	event *e*'s start time
$e.t^+$	event *e*'s end time
$e.\delta$	event *e*'s duration
$e.\triangle$	event *e*'s area
$e.w$	event *e*'s weight
$e.w_{p-s}$	event *e*'s spatial precision
$e.w_{p-t}$	event *e*'s temporal precision
$e.w_{u-s}$	event *e*'s spatial uniqueness
$e.w_{u-t}$	event *e*'s temporal uniqueness
$e.l$	event *e*'s location level

—$|\{d.a \mid d \in e\}| \geq 2$,
that is, tuples involve two or more actors

—for any event $e' \subseteq D$, $(e' \subseteq e) \vee (e \subseteq e') \Rightarrow (e' = e)$,
that is, each event is maximal

As required by the first two conditions, the semantic location granularity and time interval δ_{max} specify the constraints of a co-occurrence. Respectively, they limit the furthest that actors could be separated in space and time to be considered as co-occurring with one another. They could be adjusted to render a co-occurrence meaningful in the sense of inducing some association among actors. The third condition requires that an event must involve more than one actor; it is obvious that a co-occurrence must involve at least two actors.

Finally, requiring each event to be maximal places a constraint on the number of times that a tuple may be included in events. Its purpose is to ensure, as much as possible, that each event stands for a single underlying interaction. Generally, to be distinct, events may overlap in terms of tuples, but they ought not to be subsets of one another. Overlapping events may arise from either a chain of interactions or a long-running interaction split by the constraint of δ_{max}. The latter case is rectifiable by combining highly overlapping events involving the same actors into a long-running event.

We now enumerate some notation related to events. For ease of reference, this notation is listed in Table I. The set of all events defined over database D is denoted as $\mathcal{E}$. An event $e \in \mathcal{E}$ has several properties. The set of distinct actors represented by tuples in an event is its *actor set*, $e.\mathcal{A} = \{d.a \mid d \in e\}$. An event's *start time*, $e.t^- = min_{d \in e}\{d.t\}$, and *end time*, $e.t^+ = max_{d \in e}\{d.t\}$, are the times of its earliest and latest tuples respectively. Correspondingly, its *duration* is defined by $e.\delta = |e.t^- - e.t^+|$. The *area* $e.\triangle$ of an event measures the scope of its semantic location $e.s$. We do not specify the exact form of this property, other than that for two locations, where one contains the other, the area value should be monotonic with respect to the granularity of the semantic location, that is, the containing location should have no smaller area than the contained location. Lastly, its *weight* $e.w$ is a goodness measure related to the quality of relationship among actors of that event. The remaining notation will be explained when introduced in subsequent sections.

3.2 Event Weight

In assigning weight to events, we use an event's spatial and temporal properties to gauge its adeptness in representing an actual interaction. This adeptness is expressed in two ways. First, a higher weight indicates a greater *likelihood* of an underlying interaction. Second, this weight also measures the *intensity* of the interaction, which is indicative of the strength of association. Towards this extent, we adopt the measures of *precision* and *uniqueness*.

Precision measures how "exact" a co-occurrence is. While harder to attain, a finer-granularity co-occurrence lends more confidence that an interaction has actually taken place, in the same way that we are more confident that two people are friends if they stand very closely together than if they stand wide apart. Precision can be separately defined for the time and space dimensions.

—*Spatial precision* of an event, denoted $e.w_{p-s}$, measures how closely in space actors are from each other when participating in an event. A finer location granularity should have a higher spatial precision value. We define the spatial precision $e.w_{p-s} \in (0, 1]$ of an event e as the inverse of the event's area $e.\triangle$, normalized with respect to the maximum such value, as shown in Eq. (1).

$$e.w_{p-s} = \frac{\frac{1}{e.\triangle}}{\max_{e' \in \mathcal{E}} \{\frac{1}{e'.\triangle}\}} \tag{1}$$

—*Temporal precision* of an event, denoted $e.w_{p-t}$, measures the closeness in time between the occurrences of the earliest and the latest actors. While it is possible that an event should take place for a certain minimum duration [Wang et al. 2003], given that the data is a set of snapshots, we may not know for certain how long an actor stays at each location. On the other hand, when several actors are spotted in quick succession to each other, they are more likely to have been related. We define the temporal precision $e.w_{p-t} \in (0, 1]$ of an event e in terms of the event's duration as shown Eq. (2). Addition of a unit of time δ_{unit} to the denominator is meant to ensure a nonzero minimum value for the case of $e.\delta = \delta_{max}$.

$$e.w_{p-t} \;= 1 - \frac{e.\delta}{(\delta_{max} + \delta_{unit})} \tag{2}$$

Uniqueness is based on the idea that co-occurrences on more unique properties are more indicative of association because of the lower probability of sharing these rarer properties. For instance, it has been suggested in prior work that two Instant Messaging users who are online together during period of relative inactivity (as opposed to peak periods) are more likely to be related [Resig et al. 2004]; that unique features are better than commonly-shared features in predicting similarity-based association [Adamic and Adar 2003]; and that novel, exclusive connections are more interesting than common ones [Lin and Chalupsky 2003].

—*Spatial uniqueness* measures the uniqueness of an event's location among other events. Intuitively, co-occurrences at unique locations are more predictive of meaningful interaction. The function for spatial uniqueness $e.w_{u-s} \in (0, 1]$ is given in Eq. (3). Counting only events other than itself ensures a nonzero minimum value.

$$e.w_{u-s} = 1 - \frac{|\{e' \in \mathcal{E} \mid (e' \neq e) \wedge (e'.s = e.s)\}|}{|\mathcal{E}|} \tag{3}$$

—*Temporal uniqueness* has an effect that is similar to spatial uniqueness. An event that takes place when few other events are taking place are less likely to have been due to chance. Two events overlap each other temporally if they share at least a nonzero period of time. This is reflected in the function for temporal uniqueness $e.w_{u-t} \in (0, 1]$ given in Eq. (4).

$$e.w_{u-t} = 1 - \frac{|\{e' \in \mathcal{E} \mid (e' \neq e) \wedge (e'.[t^-, t^+] \cap e.[t^-, t^+] \neq \emptyset)\}|}{|\mathcal{E}|} \tag{4}$$

Finally, *event weight* $e.w \in (0, 1]$ is the product of the four preceding measures, as shown in Eq. (5). Having nonzero value for each measure prevents any one measure from nullifying the contribution of the other measures. An event's weight can be interpreted as the probability that the event predicts an actual association between participating actors or the strength of such a predicted association.

$$e.w = e.w_{p-s} \times e.w_{p-t} \times e.w_{u-s} \times e.w_{u-t} \tag{5}$$

3.3 Supporting Locations with Multilevel Granularity

Earlier, we define a an event in terms of locations at a single, user-specified level of granularity. Now, we extend that definition to include locations with multiple levels of granularity.

We code granularity levels as $l \in \{1, 2, \ldots, l_{max}\}$, with 1 and l_{max} representing the coarsest and finest levels of granularity, respectively. For example, a physical location may have several levels of granularity, for example, building (level 1), floor (level 2), and room (level 3). For a tuple d with location $d.s$, we refer to its level of granularity as $d.s.l$. We represent a containing location at a level l' coarser than $d.s.l$ as $d.s(l')$. A tuple that supports an event at a particular location level also supports all coarser levels.

We magnify the database D into another database D' such that $D' = \{d_j = \langle a, t, d_i.s(l')\rangle \mid d_i \in D,\ 1 \leq l' \leq d_i.s.l\}$. Each tuple $d_i \in D$ may produce up to l_{max} number of tuples $d_j \in D'$, for the same actor and the same time value but with locations expressed at various levels of granularity. Thus, the only necessary change to Definition 3.1 involves using D' in place of D.

The multilevel granularity also lends itself to a natural function for area. A location of a coarser granularity should have an area at least as large as those locations of finer granularity that it contains. One possible function for an event's area is the inverse of the granularity level of its location. If, for an event e, its location level is $e.l$, then we may express its area as $e.\triangle = \frac{1}{e.l}$. Using this area function, the spatial precision can be rewritten as in Eq. (6).

$$e.w_{p-s} = \frac{e.l}{\max_{e' \in \mathcal{E}} \{e'.l\}} \tag{6}$$

Deriving events from the magnified database D' may have several consequences. For one, knowing that two actors are in the same city is redundant if we know they are at the same home unit. If several events involve the same set of actors at around the same time, then we should take into account only the one with the finest location granularity. To formally capture the relationship between events that arises from the multilevel granularity structure of semantic locations, we give the following definition of *subevent* and *superevent*.

Definition 3.2. An event e_{sub} is a *subevent* of another event e_{sup}, or alternatively e_{sup} is a *superevent* of e_{sub}, if the following conditions are met.

—$(e_{sup}.\triangle > e_{sub}.\triangle) \wedge (e_{sup}.s$ contains $e_{sub}.s)$,
that is, the superevent's location has a coarser granularity and contains the subevent's

—$(e_{sup}.t^{-} \leq e_{sub}.t^{-}) \wedge (e_{sub}.t^{+} \leq e_{sup}.t^{+})$,
that is, the subevent's time period sits within the superevent's

—$e_{sub}.\mathcal{A} \subseteq e_{sup}.\mathcal{A}$,
that is, actors participating in the subevent participate in the superevent as well.

The first condition captures the essence of the subevent-superevent relationship as having arisen from locations of different granularity levels. If a subevent and its superevent have arisen from the same tuples in the original database D, then the latter two conditions are natural consequents of the first condition. As it is, a subevent is a more restrictive instance of a more general superevent, involving fewer actors congregating at a smaller location over a shorter duration. Note that this subevent-superevent relation is only defined between events already constructed from the database D'. It is used to determine events associated with a pair of actors, as described in the following section.

3.4 Event-Based Links

We have seen that spatio-temporal events are assigned weight over a continuous range of 0 to 1. If we interpret this weight as the probability that an event predicts an association, we may want to impose a certain threshold (*min_event_weight*) on the minimum weight that an event should have to support links between actors.

Definition 3.3. An event e supports a link $\langle a_x, a_y \rangle$ between two actors, a_x and a_y, if $(\{a_x, a_y\} \subseteq e.\mathcal{A}) \wedge (e.w \geq min\_event\_weight)$, for a given threshold *min_event_weight*.

For any given pair, there may be more than one such event. We can then group together all such events as the *event set* of the pair. Furthermore, owing to the multilevel granularity of semantic locations, we should take care to only include the most restrictive subevents supporting a linkage between the pair.

Definition 3.4. For a given link $\langle a_x, a_y \rangle$, its event set is $\mathcal{E}_{\langle a_x, a_y \rangle} \subseteq \mathcal{E}$, such that

—$\mathcal{E}_{\langle a_x, a_y \rangle} = \{e \in \mathcal{E} \mid (\{a_x, a_y\} \subseteq e.\mathcal{A}) \wedge (e.w \geq min\_event\_weight)\}$
—$\forall e \in \mathcal{E}_{\langle a_x, a_y \rangle}\ \nexists e' \in \mathcal{E}_{\langle a_x, a_y \rangle},\ e'$ is a subevent of e.

The size of the event set of a link hints at the strength of relationship between the pair of actors. Intuitively, the greater the cardinality of an event set, the more events profess to establish the linkage between the concerned pair, and correspondingly not only the linkage between the pair is more likely, it is also likely to be stronger. In order to factor this in quantifying the relationship strength of a pair, we define the *link weight* for a pair of actors $\langle a_x, a_y \rangle$ by the summation of the weight of the events in its event set, as given in Eq. (7).

$$\langle a_x, a_y \rangle.w = \sum_{e \in \mathcal{E}_{\langle a_x, a_y \rangle}} (e.w) \tag{7}$$

To control the number of links to be included in the output social network, we may impose a threshold *min_link_weight*.

Definition 3.5. A link $\langle a_x, a_y \rangle$ exists if $\langle a_x, a_y \rangle.w \geq min\_link\_weight$, for a given threshold *min_link_weight*.

We refer to the above model for constructing social network links from spatio-temporal events as *STEvent*. Based on this model, we can now restate the spatio-temporal event-based social network discovery problem more concretely.

Given database D, maximum duration δ_{max}, thresholds *min_event_weight* and *min_link_weight*, find social network graph $G(G_V, G_E)$, where

—$G_V = \{a \mid \exists \langle a_x, a_y \rangle \in G_E,\ a \in \{a_x, a_y\}\}$
—$G_E = \{\langle a_x, a_y \rangle \mid \langle a_x, a_y \rangle.w \geq min\_link\_weight\}$

4. COMPUTATIONAL ALGORITHMS

In this section, we present algorithms to solve the above-mentioned problem in two phases, namely: (1) *construction of events* and (2) *construction of links*.

4.1 Phase 1: Construction of Events

This phase deals with parsing the database, creating events, and assigning tuples to these events. Algorithm 4.1 lists the required steps. It takes as input the database D and the maximum duration δ_{max}. It returns as output the set of all events $\mathcal{E}$ constructed from D.

First, two sets of events, $\mathcal{E}_{cand}$ and $\mathcal{E}$, are initialized as empty sets. $\mathcal{E}_{cand}$ is a temporary store of recently created events that may still be affected by incoming tuples. $\mathcal{E}$ is the output set of events.

Tuples are traversed in chronological order (line 2). A new event is created whenever a new location or time stamp is seen (lines 10–13). Moreover, one event is created or updated for each location granularity (line 9). Events in the temporary store $\mathcal{E}_{cand}$ of the same location as the incoming tuple d are updated (lines 14–16). $\mathcal{E}_{cand}$ is continually cleared of events whose temporal properties

Algorithm 4.1. CONSTRUCTION OF EVENTS

—**Input**: database D, maximum duration δ_{max}
—**Output**: events $\mathcal{E}$
—**Algorithm**:

```
1:  $\mathcal{E} = \emptyset,\ \mathcal{E}_{cand} = \emptyset$
2:  **for** each tuple $d \in D$ in the order of $d.t$ **do**
3:      **for** each event $e \in \mathcal{E}_{cand},\ (d.t - e.t^- > \delta_{max})$ **do**
4:          **if** $(|e.\mathcal{A}| \geq 2) \wedge (\nexists e' \in \mathcal{E}, (e \subseteq e'))$ **then**
5:              $\mathcal{E} = \mathcal{E} \cup \{e\}$
6:          **end if**
7:          $\mathcal{E}_{cand} = \mathcal{E}_{cand} - \{e\}$
8:      **end for**
9:      **for** each location granularity level $l = 1$ to $l_{max}$ **do**
10:         **if** $\nexists e \in \mathcal{E}_{cand},\ (e.s = d.s(l)) \wedge (e.t^- = d.t)$ **then**
11:             create new event $e = \{d\}$ with $e.s = d.s(l)$ and $e.t^- = d.t$
12:             $\mathcal{E}_{cand} = \mathcal{E}_{cand} \cup \{e\}$
13:         **end if**
14:         **for** each event $e' \in \mathcal{E}_{cand},\ e' \neq e,\ (e'.s = d.s(l))$ **do**
15:             $e' = e' \cup \{d\}$
16:         **end for**
17:     **end for**
18: **end for**
19: return $\mathcal{E}$
```

do not allow them to accept more tuples, that is, whose duration would breach the limit of δ_{max} (lines 3–8). If such events are well-constructed, namely, they consist of at least two actors, and are not just subsets of another event, they are transferred to the output set $\mathcal{E}$. After all the tuples have been traversed, the set of events $\mathcal{E}$ is returned as output of this phase (line 19).

To gauge the complexity of the algorithm, we look at the most deeply-nested iteration, which is the updating of events with the current tuple (lines 14–16 of Algorithm 4.1). This step is done once for every event in the temporary store with the same location as the current tuple (up to δ_{max} iterations), for every level of location granularity (up to l_{max} iterations), for every tuple of the database ($|D|$ iterations). In the worst case, the complexity of this phase is $O(|D| \times l_{max} \times \delta_{max})$.

4.2 Phase 2: Construction of Links

In this phase (Algorithm 4.2), the events $\mathcal{E}$ generated in the previous phase are evaluated, and links are generated from them. As output, this phase returns the nodes G_V and the links G_E of the desired social network graph $G(G_V, G_E)$.

First, we initialize G_V, G_E, and G_{Ecand} as empty sets. G_{Ecand} is a temporary store of links. In the first outermost loop iterating over each event (lines 2–25), the algorithm computes event weights to determine which events could support links. Computing spatial and temporal precisions ($e.w_{p-s}$ and $e.w_{p-t}$) is trivial if the maximum duration and area are known beforehand (line 3). However, computing spatial and temporal uniqueness ($e.w_{u-s}$ and $e.w_{u-t}$) requires looping

Algorithm 4.2. CONSTRUCTION OF LINKS

—**Input**: events $\mathcal{E}$, *min_event_weight*, *min_link_weight*
—**Output**: nodes G_V, links G_E
—**Algorithm**:

```
 1: $G_V = \emptyset,\ G_E = \emptyset,\ G_{Ecand} = \emptyset$
 2: for each event $e \in \mathcal{E}$ do
 3:   compute $e.w_{p-s}$ and $e.w_{p-t}$
 4:   $countEventsSharingLocation = 0$, $countEventsSharingTime = 0$
 5:   for each event $e' \in \mathcal{E}$, $e' \neq e$ do
 6:     if $e'.s = e.s$ then
 7:       $countEventsSharingLocation + +$
 8:     end if
 9:     if $e'.[t^-, t^+] \cap e.[t^-, t^+] \neq \emptyset$ then
10:       $countEventsSharingTime + +$
11:     end if
12:   end for
13:   $e.w_{u-s} = 1 - countEventsSharingLocation/|\mathcal{E}|$
14:   $e.w_{u-t} = 1 - countEventsSharingTime/|\mathcal{E}|$
15:   $e.w = e.w_{p-s} \times e.w_{p-t} \times e.w_{u-s} \times e.w_{u-t}$
16:   if $e.w \geq min\_event\_weight$ then
17:     for each pair $a_x, a_y \in e.\mathcal{A}$ do
18:       $G_{Ecand} = G_{Ecand} \cup \{\langle a_x, a_y\rangle\}$
19:       if $\nexists e' \in \mathcal{E}_{\langle a_x, a_y\rangle}$, ($e'$ subevent of $e$) then
20:         remove superevents of $e$ from $\mathcal{E}_{\langle a_x, a_y\rangle}$
21:         $\mathcal{E}_{\langle a_x, a_y\rangle} = \mathcal{E}_{\langle a_x, a_y\rangle} \cup \{e\}$
22:       end if
23:     end for
24:   end if
25: end for
26: for each link $\langle a_x, a_y\rangle \in G_{Ecand}$ do
27:   $\langle a_x, a_y\rangle.w = \sum_{e \in \mathcal{E}_{\langle a_x, a_y\rangle}}(e.w)$
28:   if $\langle a_x, a_y\rangle.w \geq min\_link\_weight$ then
29:     $G_E = G_E \cup \{\langle a_x, a_y\rangle\}$
30:     $G_V = G_V \cup \{a_x, a_y\}$
31:   end if
32: end for
33: return $G_V$, $G_E$
```

through events in $\mathcal{E}$ to count other events sharing the same spatial or temporal properties, which takes $|\mathcal{E}|$ iterations (lines 4–14). Event weight is the product of the above four measures (line 15). If an event's weight is above the threshold *min_event_weight*, this event can support links between pairs of actors (lines 16–24). An event of n actors supports $n(n-1)/2$ links. These links are candidate links entered into the temporary store G_{Ecand}, as the ultimate weight of these links is not yet known. For each candidate link $\langle a_x, a_y\rangle$, its event set $\mathcal{E}_{\langle a_x, a_y\rangle}$ is updated while keeping watch of subevent-superevent relationships in its event set; only the most restrictive subevents are accepted. The outcome is the set of candidate links G_{Ecand}.

In the second outermost loop (lines 26–32), the weight of each candidate link is evaluated by summing up the weights of events due to the pair of actors. Links whose weights are beyond the required threshold *min_link_weight* are entered into the output set of links G_E. The corresponding actors are also included in the set of nodes G_V. The two sets, which is a graph representation of the desired social network, are then returned as output of this phase (line 33).

Complexity-wise, the first outermost loop iterates through $|\mathcal{E}|$ events. For each event, the computation of uniqueness measures may yet require $|\mathcal{E}|$ iterations. The generation of candidate links is more difficult to estimate. For an event e, the number of pairs generated would be $|e.\mathcal{A}|(|e.\mathcal{A}| - 1)/2$, but the average value of $|e.\mathcal{A}|$ is not known beforehand. A simplifying assumption is that each event introduces an equal number of pairs into the candidate set, in which case the number of candidate links per event is $|G_{Ecand}|/|\mathcal{E}|$. The estimated complexity of the first outermost loop is then $O(|\mathcal{E}| \times (|\mathcal{E}| + |G_{Ecand}|/|\mathcal{E}|)) = O(|\mathcal{E}|^2 + |G_{Ecand}|)$. Given that the second outermost loop has a complexity of $O(|G_{Ecand}|)$, the overall complexity of this phase is $O(|\mathcal{E}|^2 + |G_{Ecand}|)$.

4.3 Algorithm Enhancements

In this section, we present several enhancements to the previous algorithms. In general, they improve the computational complexity at the cost of memory complexity due to new data structures being introduced to achieve the speedup. The enhanced algorithms for Phase 1 and Phase 2 are given in Algorithms 4.3 and 4.3, respectively.

Lazy construction of events. The first enhancement is aimed at removing the term δ_{max} from the complexity of Phase 1 by avoiding the immediate creation of events. In Algorithm 4.1, every time a new time stamp is seen, a new event is created, leading to the existence of up to δ_{max} number of events in $\mathcal{E}_{cand}$ to update with the incoming tuple.

A better approach is not to create an event for every new time stamp, but only to replace a prior, expired event of the same location. In the enhanced Algorithm 4.3, when an event e expires, that is, cannot be updated with the incoming tuple d as $|d.t - e.t^-| > \delta_{max}$, the event e is shelved in $\mathcal{E}$ and a new *child event* is created to replace e in $\mathcal{E}_{cand}$ (lines 4–14). The child event e_{child} is "descended" from e, being a subset of e containing tuples less than δ_{max} apart from the incoming tuple d, such that $|d.t - e_{child}.t^-| \leq \delta_{max}$. A brand new event is created only if currently there is no event in $\mathcal{E}_{cand}$ with the same location (lines 18–21). Therefore, at any point of time, there will be only one event of a particular location in $\mathcal{E}_{cand}$. At each iteration, there is only one event to update with the incoming tuple, and the term δ_{max} disappears from Phase 1's complexity, leaving $O(|D| \times l_{max})$.

Indexing events' locations and time periods. The second enhancement is aimed at improving Phase 2's complexity by more efficiently evaluating the spatial and temporal uniqueness of events. In Section 4.2, we relate that evaluating the spatial and temporal uniqueness of an event may require iterating through all events to count how many events share the same location or time period ($|\mathcal{E}|$ complexity).

Algorithm 4.3. Enhanced Construction of Events

—**Input**: database D, maximum duration δ_{max}, *min_link_weight*

—**Output**: events $\mathcal{E}$, index events by location $I_{s\to e}$, index of events by time $I_{t\to e}$, pruned actors $\mathcal{A}_{pruned}$

—**Algorithm**:

```
 1: initialize indices $I_{s\to e}$ and $I_{t\to e}$
 2: $\mathcal{E} = \emptyset,\ \mathcal{E}_{cand} = \emptyset,\ \mathcal{A}_{all} = \emptyset,\ \mathcal{A}_{pruned} = \emptyset$
 3: for each tuple $d \in D$ in the order of $d.t$ do
 4:   for each event $e \in \mathcal{E}_{cand},\ (d.t - e.t^- > \delta_{max})$ do
 5:     if $(|e.\mathcal{A}| > 1) \wedge (\nexists e' \in \mathcal{E}, (e \subseteq e'))$ then
 6:       $\mathcal{E} = \mathcal{E} \cup \{e\}$
 7:       $updateIndex(I_{s\to e},\ e.s,\ e)$
 8:       $updateIndex(I_{t\to e},\ e.[t^-, t^+],\ e)$
 9:       $\mathcal{A}_{all} = \mathcal{A}_{all} \cup e.\mathcal{A}$
10:     end if
11:     $\mathcal{E}_{cand} = \mathcal{E}_{cand} - \{e\}$
12:     $e_{child} = \{d_i \in e \mid d.t - d_i.t \leq \delta_{max}\}$
13:     $\mathcal{E}_{cand} = \mathcal{E}_{cand} \cup \{e_{child}\}$
14:   end for
15:   for each location granularity level $l = 1$ to $l_{max}$ do
16:     if $\exists e \in \mathcal{E}_{cand},\ (e.s = d.s(l))$ then
17:       $e = e \cup \{d\}$
18:     else
19:       create new event $e = \{d\}$ with $e.s = d.s(l)$
20:       $\mathcal{E}_{cand} = \mathcal{E}_{cand} \cup \{e\}$
21:     end if
22:   end for
23: end for
24: for each actor $a \in \mathcal{A}_{all}$ do
25:   if $\sum_{e\in\mathcal{E}_a}(e.w_{p-s} \times e.w_{p-t}) < min\_link\_weight$ then
26:     $\mathcal{A}_{pruned} = \mathcal{A}_{pruned} \cup a$
27:   end if
28: end for
29: return $\mathcal{E},\ I_{s\to e},\ I_{t\to e},\ \mathcal{A}_{pruned}$
```

We do away with the above brute force approach by building two indices: an index of events by location $I_{s\to e}$ and an index of events by time $I_{t\to e}$. These two indices are constructed in Phase 1 (lines 7–8 of Algorithm 4.3). With indices, the same task of evaluating spatial and temporal uniqueness (lines 4–7 of Algorithm 4.4) can be done with $\log|\mathcal{E}|$ complexity. Phase 2's complexity can be reduced to $O(|\mathcal{E}| \times \log|\mathcal{E}| + |G_{Ecand}|)$, at the cost of increased memory complexity due to the indices.

Pruning by actors. The third enhancement attempts to reduce the size of $|G_{Ecand}|$, the number of candidate links to be examined in Phase 2, by pruning actors that are unlikely to achieve the required threshold *min_link_weight*. The intuition is that if the combined weight of all an actor's events is not beyond *min_link_weight*, then none of this actor's links (supported by a subset of events) will meet the *min_link_weight* threshold.

Algorithm 4.4. Enhanced Construction of Links

```
—Input: $\mathcal{E}$, $min\_event\_weight$, $min\_link\_weight$, $I_{s\to e}$, $I_{t\to e}$, $\mathcal{A}_{pruned}$
—Output: nodes $G_V$, links $G_E$
—Algorithm:
 1: $G_V = \emptyset,\ G_E = \emptyset,\ G_{Ecand} = \emptyset$
 2: for each event $e \in \mathcal{E}$ do
 3:    compute $e.w_{p-s}$ and $e.w_{p-t}$
 4:    $countEventsSharingLocation = queryIndex(I_{s\to e},\ e.s)$
 5:    $countEventsSharingTime = queryIndex(I_{t\to e},\ e.[t^-, t^+])$
 6:    $e.w_{u-s} = 1 - countEventsSharingLocation/|\mathcal{E}|$
 7:    $e.w_{u-t} = 1 - countEventsSharingTime/|\mathcal{E}|$
 8:    $e.w = e.w_{p-s} \times e.w_{p-t} \times e.w_{u-s} \times e.w_{u-t}$
 9:    if $e.w \geq min\_event\_weight$ then
10:       for each pair $a_x, a_y \in e.\mathcal{A}$, where $a_x, a_y \notin \mathcal{A}_{pruned}$ do
11:          $G_{Ecand} = G_{Ecand} \cup \{\langle a_x, a_y\rangle\}$
12:          if $\nexists e' \in \mathcal{E}_{\langle a_x,a_y\rangle}$, ($e'$ subevent of $e$) then
13:             remove superevents of $e$ from $\mathcal{E}_{\langle a_x,a_y\rangle}$
14:             $\mathcal{E}_{\langle a_x,a_y\rangle} = \mathcal{E}_{\langle a_x,a_y\rangle} \cup \{e\}$
15:          end if
16:       end for
17:    end if
18: end for
19: for each link $\langle a_x, a_y\rangle \in G_{Ecand}$ do
20:    $\langle a_x, a_y\rangle.w \ = \ \sum_{e\in\mathcal{E}_{\langle a_x,a_y\rangle}}(e.w)$
21:    if $\langle a_x, a_y\rangle.w \geq min\_link\_weight$ then
22:       $G_E = G_E \cup \{\langle a_x, a_y\rangle\}$
23:       $G_V = G_V \cup \{a_x, a_y\}$
24:    end if
25: end for
26: return $G_V$, $G_E$
```

The event set due to a link ($\mathcal{E}_{\langle a_x,a_y\rangle}$) is defined as the set of events that both a_x and a_y participate in. In a similar way, we can define the event set $\mathcal{E}_a$ due to a single actor a as the set of events that a participates in with any other actor. The weight due to the actor a alone is the sum of weights of events in its event set, $a.w = \sum_{e\in\mathcal{E}_a}(e.w)$. For any actor a, there does not exist a link $\langle a_x, a_y\rangle$, such that $(a \in \{a_x, a_y\}) \ \wedge \ (a.w < \langle a_x, a_y\rangle.w)$. This is because event weight is always positive and for any link $\langle a_x, a_y\rangle$, $e \in \mathcal{E}_{\langle a_x,a_y\rangle} \iff (e \in \mathcal{E}_{a_x}) \ \wedge \ (e \in \mathcal{E}_{a_y})$. Thus if an actor's weight does not meet the threshold, neither will any of this actor's links.

The set of actors to be pruned, $\mathcal{A}_{pruned}$, can be determined in Phase 1 (lines 24–28 in Algorithm 4.3). Since $\sum_{e\in\mathcal{E}_a}(e.w_{p-s} \times e.w_{p-t})$ is an upper-bound value for $a.w$, actors for whom the condition $(\sum_{e\in\mathcal{E}_a}(e.w_{p-s} \times e.w_{p-t}) < min\_link\_weight)$ holds can be excluded from G_{Ecand} (line 10 of Algorithm 4.4). The higher the specified threshold min_link_weight, the more actors are pruned and the smaller $|G_{Ecand}|$ is. Given that $|G_{Ecand}|$ is a term in Phase 2's complexity, reducing the number of candidate links would make the algorithm run faster.

The achieved speedup by the three algorithm enhancements is quite significant, as will be shown in Section 5.5.

5. EXPERIMENTS ON CYBER LOCATION DATA

The main objective of experiments is to verify the validity of the links discovered by *STEvent*, which we carry out using demographic information. In addition, we also study the behavior of our proposed algorithms with different parameter settings. To verify the consistency of the results, we repeat the experiments over several location granularity levels and several overlapping two-month periods, respectively. We also test whether the proposed algorithm enhancements improve computational efficiency. Finally, we compare *STEvent* with spatial- and temporal-only models, to underline the necessity of both space and time in *STEvent*'s event definition.

5.1 Dataset

This data is collected as a log of Web pages (given by URL addresses) accessed by users of the wireless network in our campus. These users include undergraduate and graduate students, as well as members of the university staff. We call this the *Cyber Location Data* (or *cyberdata*). Each tuple $\langle a, t, s \rangle$ consists of a user login name a, a time stamp t, and a URL address s. To protect privacy, all user login names were anonymized. While not a location in the geographical sense of the word, a URL address possesses some semantic meaning as coded by the words forming the address as well as by the content of the Web page that it points to. When several people access the same Internet resources, they could be driven by recommendation, collaboration, common affiliation, or shared interests, all of which are themselves indicators of association. This data, with irregularly-spaced time stamps, identifiable users, and semantic locations, complies with the expected characteristics of spatio-temporal data assumed by our model.

We preprocessed this data in the following way. Although the data spanned the period from August 2004 to March 2005, we did not use data for November and December 2004, as the usage level was very low during this period, which was the university's holiday period. We retained only data concerning users who appeared at least once in each of the six remaining months (August to October 2004 and January to March 2005). There were a total of 533 such users. We also opted to use a single level of location granularity, choosing the URL domain for most experiments. There were about 131 thousand unique URL domains. The data size after preprocessing was about 9.5 million tuples or 550MB.

5.2 Demographic Similarity

The ideal scheme to verify the relationships extracted by the proposed model is to seek confirmation from the actors concerned. However, the provider of the data ruled out approaching the actors concerned for privacy reasons. An alternative verification scheme is to measure the similarity between two related actors. A result from sociology is that people who are more closely related tend to have greater similarity to each other [McPherson et al. 2001]. Given the availability of limited demographic data on each actor, we look at whether strongly related pairs of actors are more likely to be similar than any pair of two picked at random actors.

Table II. Parameter Values (*cyberdata* Aug-Sep04)

Parameter	Default Value	Range
$\|D\|$	2.5 million	0 – 2.5 million
δ_{max}	2 hours	10 minutes – 16 hours
min_link_weight	0	0 – 100
min_event_weight	0	0

Table III. Demographic Similarity (*cyberdata* Aug-Sep04)

		STEvent (%)			*Spatial*	
Common Features	*Random* (%)	*Level 1*	*Level 2*	*Level 3*	*Level 1*	*Temporal*
at least 1	55.5	86.0	84.8	84.8	78.0	80.0
at least 2	18.5	22.0	23.2	25.3	17.6	27.0
at least 3	3.3	5.3	5.9	9.4	2.3	9.7

As a baseline, we draw 100 links at random from the same population of actors and call it the *Random* result set. To represent our proposed model, we run the *STEvent* algorithms on the portion of *cyberdata* for the two-month period (or 61 days) *Aug-Sep04* with the default parameter values given in Table II ($\delta_{max} = 2h, min\_event\_weight = 0$). We extract the top 100 links in terms of weight and call this our *STEvent* result set. Here, we repeat the experiments across three levels of the URL directory structure, at level 1 (URL domain), level 2 (first directory after the domain), and level 3 (one more directory below). On average, *STEvent*'s top 100 links involve 35 unique users, as opposed to 164 unique users for *Random*.

Demographic similarity. For each actor in a result set, we obtain information on up to three attributes, namely: *department* (e.g., business, biology, civil engineering); *status* (e.g., undergraduate, postgraduate, staff); and *year of admission* (e.g., 2004). 515 out of the 533 users in the data have at least one attribute value known. We count the number of attribute values that each pair of actors have in common (0 to 3). It follows that higher values indicate higher similarity.

Table III shows the distribution of the number of common attribute values among the 100 pairs in and *Random* and *STEvent* result sets. Since not all pairs of actors can be compared on all three attributes, we present the number of pairs with at least *n* common attributes as a fraction of all pairs that can be compared on *n* attributes. For instance, in Table III, for *STEvent*, out of the pairs who have all three attributes present, 5.3% have all three attribute values in common.

Intuitively, we suspected that stronger relationship could be detected at more specific locations. However, this expectation is not supported by the demographic similarity distributions in Table III, which are relatively uniform across the three levels. The χ^2 homogeneity test at 5% level [Walpole et al. 2002] confirms that there is insufficient evidence to conclude otherwise, at least for this particular *cyberdata* dataset. This hints at the adequately high quality of the URL domain (level 1) in representing the deeper levels of location granularity. Hereinafter, we will use level 1 by default to represent *STEvent*.

Compared to the corresponding distribution for the *Random* result set, *STEvent*'s distribution (at any level) shows greater similarity between pairs than the *Random*'s. Statistical χ^2 goodness-of-fit test [Walpole et al. 2002] at 5% level of significance also suggests that the *STEvent*'s distribution is sufficiently dissimilar from *Random*'s to imply that the improvement by *STEvent* over *Random* is significant.

Spatial and temporal. The proposed *STEvent* model is based on a definition of an event that uses both spatial and temporal dimensions (see Section 3.1). Disregarding either space or time would generate different results. Here, we also compare *STEvent* against two models.

—*Spatial* considers only the spatial dimension. It is computed using the algorithms in Section 4, after replacing the time value of all tuples with a constant.

—*Temporal* considers only the temporal dimension. It is also computed using the algorithms in Section 4, after replacing the location value of all tuples with a constant.

Comparing these values with those for *Random* and *STEvent* in Table III, we see that both *Spatial* and *Temporal* still outperform *Random*. Interestingly, *Temporal* has higher demographic similarity than *Spatial*, and in certain cases also higher similarity than *STEvent*. It is probable that for *cyberdata*, spatial co-occurrence is less likely among socially related individuals due to the diversity of possible cyber locations. Instead, users' patterns of activity, as given by temporal co-occurrence, may be more correlated with demographic similarity. Note that this is likely more indicative of the underlying data set than the actual merits of the various methods, given that demographic similarity is not the gold standard and that for a different data set the methods may perform differently (as shown in Section 6.2).

Common URLs. Ultimately, similarity alone is insufficient to verify the links. More importantly, these links should also be supported by reasonable events that hint at the probable relationships among the actors. As such, we empirically look at event locations for a select set of top-ranked pairs. In Table IV, we list 12 pairs who are among the top 50 links in terms of weight within the *STEvent* (level 1) result set. For each pair, we provide their common demographic attribute values, a number of URL locations that they have in common, as well as their link weight. The 12 pairs are not sorted by weight, but rather are organized around the subsets of actors involved for ease of discussion.

Pairs 1 to 6. The first six pairs involve four civil engineering graduate students (a_1, a_2, a_3, and a_4). Their interest in Chinese universities (South East China University and Xi'an Jiaotong University) indicates probable prior affiliation to these institutions. In addition, other China-based URLs such as BJPTA.gov.cn, Chinese Software Developer Network, and Sohu Sports

Table IV. Highly Similar Event-Based Pairs (*cyberdata* Aug-Sep04 at level 1)

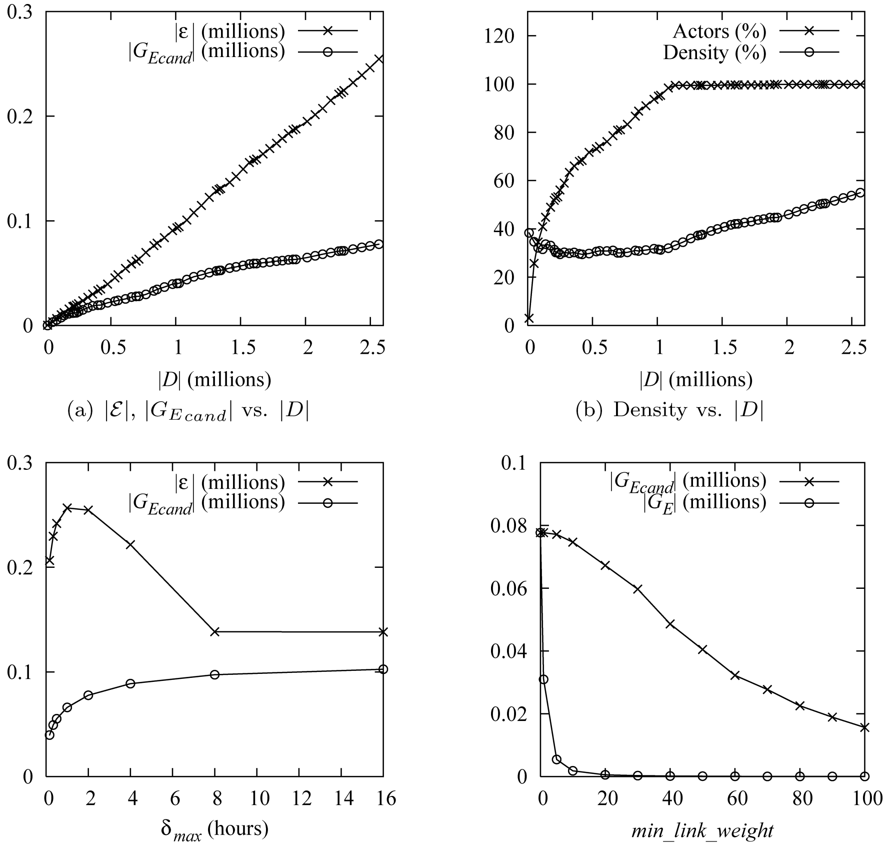
Pairs		*Common Attributes*	*Sample URL Locations*	*Weight*
1	a_1 a_2	Postgraduate Civil Engineering 2003	Center for Aerospace Structures, Univ. of Colorado South East University (China) ScienceDirect Digital Library	228.9
2	a_2 a_3	Postgraduate Civil Engineering	Singapore Millennium Foundation Scholarship South East University (China) ScienceDirect Digital Library	201.1
3	a_1 a_3	Postgraduate Civil Engineering	BJPTA.gov.cn (China) South East University (China) ScienceDirect Digital Library	180.3
4	a_1 a_4	Postgraduate Civil Engineering	Xi'an Jiaotong University (China) US Naval Facilities Engineering Command US Federal Real Property Management	148.5
5	a_3 a_4	Postgraduate Civil Engineering	Sohu Sports (China) ScienceDirect Digital Library	86.1
6	a_2 a_4	Postgraduate Civil Engineering	Chinese Software Developer Network ScienceDirect Digital Library	85.1
7	a_5 a_6	Postgraduate Electrical Engineering	Sina Entertainment, Finance (China) BlogCN	145.1
8	a_6 a_7	Postgraduate Electrical Engineering	Sina Entertainment, Sports (China) IEEE Xplore National Kidney Foundation (Singapore)	136.8
9	a_5 a_7	Postgraduate Electrical Engineering	Sina Entertainment, Finance (China) IEEE Xplore HardwareZone (Singapore)	99.3
10	a_8 a_9	Postgraduate Biology 2003	Nucleic Acids Research Journal (NAR) National Center for Biotechnology Information (NCBI) ScienceDirect Digital Library	96.0
11	a_9 a_{10}	Postgraduate Biology	National Center for Biotechnology Information (NCBI) Blizzard Entertainment	91.3
12	a_{11} a_{12}	Postgraduate Mechanical Engineering	AsiaOne (Singapore) Zaobao (Singapore) ScienceDirect Digital Library	84.0

suggest their common country of origin (China). Their access of ScienceDirect reveals their research occupation.

Pairs 7 to 9. The next three pairs involve three electrical engineering graduate students (a_5, a_6, and a_7). Their access of entertainment, finance, and sports sections of the China-based Sina portal indicates their common interests and probable country of origin. Their research occupation is evidenced by their access of IEEE Xplore, an established digital library for electrical engineering.

Pairs 10 and 11. The next two pairs involve three biology graduate students (a_8, a_9, and a_{10}). Nucleic Acid Research journal and National Centre for Biotechnology Information database indicate their common research interests. a_9 and a_{10} are likely to have a common interest in gaming as well, as evidenced by their access of the Web site Blizzard Entertainment (an American computer game developer).

Pair 12. The last pair involves two mechanical engineering graduate students (a_{11}, and a_{12}). Their common URLs include newspaper portals (AsiaOne and Zaobao) as well as the ScienceDirect digital library.



(a) $|\mathcal{E}|$, $|G_{Ecand}|$ vs. $|D|$

(b) Density vs. $|D|$

(c) $|\mathcal{E}|$, $|G_{Ecand}|$ vs. δ_{max}

(d) $|G_{Ecand}|$, $|G_E|$ vs. *min_link_weight*

Fig. 3. Algorithmic behavior (*cyberdata* Aug-Sep04).

In each of the 12 pairs above, there is a strong impression of the probable existence of some sort of relationship supported by common occupation, interest, or country of origin.

5.3 Algorithmic Behavior

Here, we study the effects of variation of parameters on the behavior of the algorithms. At any one time, we vary one parameter and keep the rest fixed. When fixed, the parameters have the following values.

—*D* will be the portion of *cyberdata* for the two-month period (or 61 days) *Aug-Sep04*, with a single location granularity level (URL domain).

—δ_{max} will be two hours, which is a reasonable time window for a meaningful co-occurrence. As we see in Figure 3(c), this setting also generates near the peak number of events.

—*min_event_weight* will be fixed at 0, assuming that all discovered events matter.

—*min_link_weight* will also be fixed at 0 for simplicity. In practice, we expect *min_link_weight* to be significantly above 0 in order to derive only the strong links. We explore the variation of *min_link_weight* in Figure 3(d).

These default parameter values, as well as the range that we explore later, are given in Table II.

Vary $|D|$. While keeping the other parameter values fixed, the data size is varied from 0 to 2.5 million tuples by starting with an empty set and then incrementally adding one day's worth of data, resulting in 61 experimental readings. Figure 3(a) suggests that the growth in the number events $|\mathcal{E}|$ is approximately linear to the data size $|D|$. This makes sense, as the rate at which events take place in real life should be more or less constant. The discovered events support potential links between any pair of actors participating in events together. As more events are discovered, the number of candidate links $|G_{Ecand}|$ also increases, as shown in Figure 3(a).

Vary density. Figure 3(b) shows the growth in the number of actors with at least one event. Not every actor is active every day. As we incrementally increase the data size by one day's worth of data, the number of participating actors initially increases. After a month, each actor has participated in at least one event. Thereafter the number of actors is stable. We also track the density of the graph formed by the candidate links, defined as the ratio of the number of candidate links to the maximum possible such number or $\frac{|G_{Ecand}|}{n(n-1)/2}$ for n actors. Initially, the density is rather flat, as the number of links increases with the number of actors. By the end of August, the number of actors is stable, but as more events occur, more pairs of actors can be connected by at least one event. This is evident in the later increase of density. We expect that the density would flatten again once all the possible links have been discovered.

Vary δ_{max}. Varying the maximum event duration δ_{max} would directly affect the formation of each event, as it is integral to determining what constitutes an event. In Figure 3(a), as this value is varied from 10 minutes to 16 hours, initially we see a minor surge in the number of events, which reaches the peak around one and two hours. Apparently, longer δ_{max} makes it easier for several tuples to belong to an event together. However, it then declines and eventually levels off. An exceedingly long δ_{max} would "combine" several shorter events at the same location into a long-running event. On the other hand, the number of candidate links keeps increasing, though at an increasingly slower pace, as it gets less and less restrictive for two tuples to join together in an event. Nevertheless, the top 100 pairs remain relatively consistent even at different δ_{max} settings. For instance, the top 100 pairs for $\delta_{max} = 2$hr share a significant proportion of the top 100 pairs produced by other δ_{max} settings, that is, 91% with $\delta_{max} = 1$hr, 86% with $\delta_{max} = 4$hr, and 82% $\delta_{max} = 8$hr.

Vary min_link_weight. Previously, with *min_link_weight* set at 0, the candidate links are equivalent to the discovered links. Figure 3(d) shows that as we increase this value to 100, the number of candidate links $|G_{Ecand}|$ is always greater than the number of links $|G_E|$, as only those candidate links whose

Table V. Demographic Similarity across Periods (*cyberdata* at level 1)

Common Features		*STEvent* (%)				
	Random (%)	*Aug-Sep04*	*Sep-Oct04*	*Oct04-Jan05*	*Jan-Feb05*	*Feb-Mar05*
at least 1	55.5	86.0	84.0	78.0	70.0	65.0
at least 2	18.5	22.0	23.5	19.4	18.0	18.0
at least 3	3.3	5.3	6.5	11.5	10.5	6.1

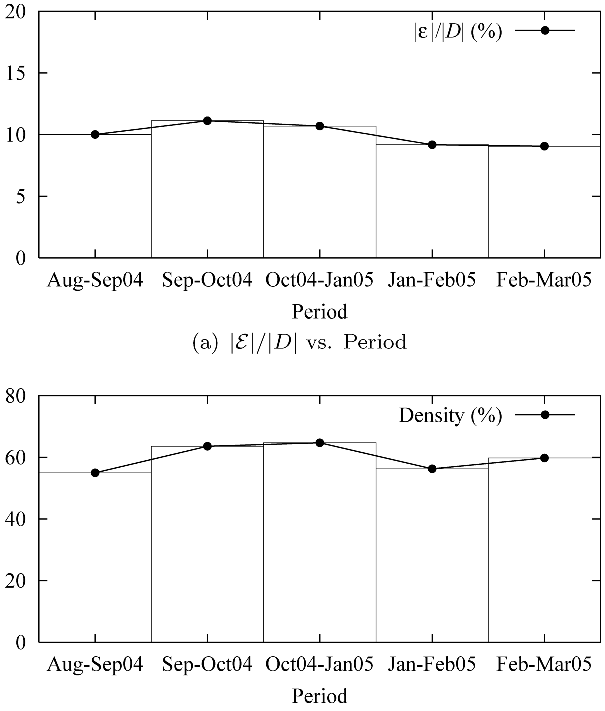
weight meets the required *min_link_weight* qualify as links. The number of candidate links gradually falls off due to the *pruning by actor* algorithm enhancement, in which an actor whose aggregate weight of all its links is still below *min_link_weight* is excluded from the set of candidate links. In turn, the number of links drops more precipitously from 77677 at *min_link_weight* = 0, to 573 at *min_link_weight* = 20, to 28 at *min_link_weight* = 100. It is expected that using a lower *min_link_weight* threshold will increase the number of links produced, but these links are also likely to be weaker. For instance, going from top 100 pairs to top 200 pairs will decrease the demographic similarity at 1 attribute from 86% to 72%, and demographic similarity at 2 attributes from 22% to 17%.

5.4 Variation across Time

Previously, we have used data from the two-month period (*Aug-Sep04*). These experiments are repeated for four more overlapping two-month periods: *Sep-Oct04*, *Oct04-Jan05*, *Jan-Feb05*, and *Feb-Mar05*. Note that *Oct04-Jan05* does not include the holiday months of November and December 2004. We again use the default parameter values (other than data size) as given in Table II.

Demographic similarity. The demographic similarity distributions shown in Table V vary slightly among these periods. Nevertheless, the χ^2 goodness-of-fit test at 5% level [Walpole et al. 2002] confirms that *STEvent* distribution for each period is sufficiently dissimilar from that of *Random* (thus marking each period's improvement over *Random* significant). On average, every two consecutive periods share about 60% of the top 100 links in common. This indicates a remarkable consistency in the identification of the strongest links. In the short term, the top links in one period are likely to feature as top links in the next period. However, within a campus environment, it is expected that in the long term the set of strongest links would likely change as new students arrive and current students graduate.

Events and density. Figure 3(a) suggests that the relationship between the number of events and data size is approximately linear, so in Figure 4(a) we track how the ratio of events to tuples varies with time periods. Apparently, this ratio remains stable in the range of 9-11%. Yet another measure that is expected to be relatively stable is the density of the candidate links graph, an expectation which is met in Figure 4, showing only slight variations with values in the range of 55-65%.



(a) $|\mathcal{E}|/|D|$ vs. Period

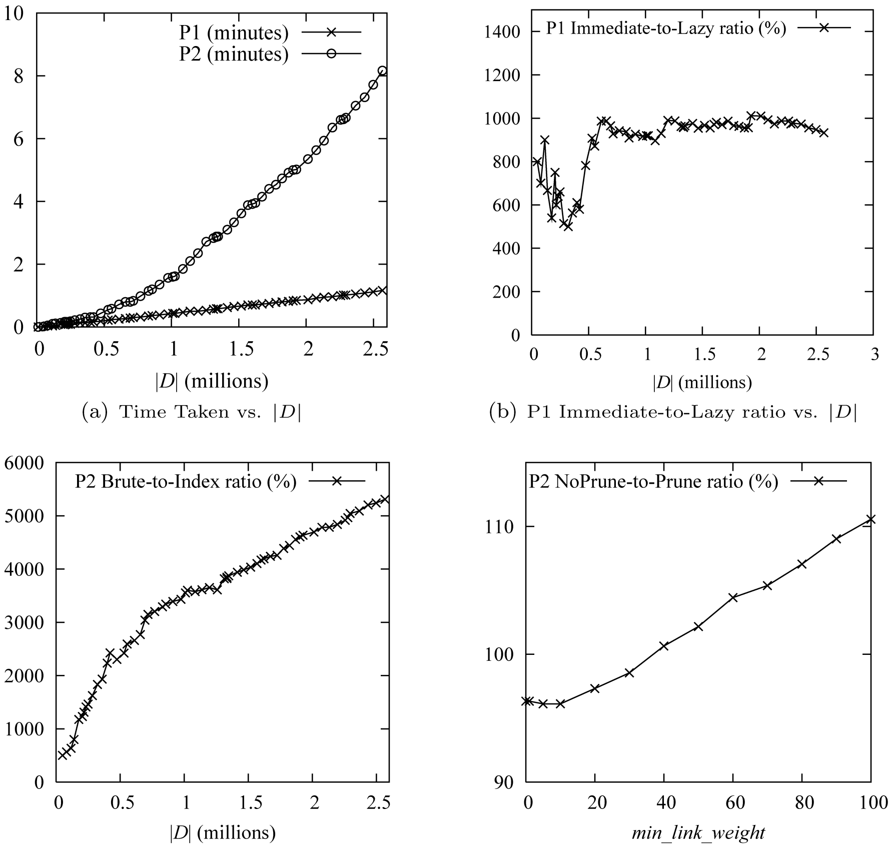
(b) Density vs. Period

Fig. 4. Algorithmic behavior across periods (*cyberdata*).

5.5 Time Complexity

Here, we explore the time complexity of our algorithms. It is also instructive to see how each of the three algorithm enhancements has improved computational efficiency. While keeping the other parameters fixed at the default values (see Table II), we vary the data size and chart the time taken for each phase of the algorithm, Phase 1 being the construction of events and Phase 2 being the construction of links. To get these timings, the algorithms were implemented in C++ and run on an Intel Pentium 4 1.7GHz machine with 512MB RAM running Windows XP.

Fully enhanced. Figure 5(a) shows that the time complexity of Phase 1 (P1) and Phase 2 (P2) when all three algorithm enhancements are implemented. P1 grows linearly with the data size, confirming the theoretical complexity of $O(|D|)$. In turn, P2 is slightly above linear, feasibly approaching the theoretical complexity of $O(|\mathcal{E}| \times \log |\mathcal{E}| + |G_{Ecand}|)$. Overall, P2 is running at a multiple of up to 7 times of P1, highlighting the disparity between the two phases. The longest total time taken for the two phases approaches 10 minutes; this is when *min_link_weight* is 0. It is expected that for real usage, higher values of *min_link_weight* will be set, leading to shorter completion times.



(a) Time Taken vs. $|D|$

(b) P1 Immediate-to-Lazy ratio vs. $|D|$

(c) P2 Brute-to-Index ratio vs. $|D|$

(d) P2 NoPrune-to-Prune ratio vs. *min_link_weight*

Fig. 5. Time complexity (*cyberdata* Aug-Sep04).

Selectively enhanced. The performance improvement given by an algorithm enhancement can be measured by how much running time is saved by implementing the enhancement. We track the ratio of the time taken when an algorithm enhancement *is not* implemented (numerator) to the time taken when it *is* implemented (denominator). The more time saved by the algorithm enhancement, the higher the ratio. The performance ratios of the three enhancements (lazy construction of events, indexing of events locations and times, and pruning by actors) are plotted in Figures 5(b), 5(c), and 5(d), respectively.

—In Figure 5(b), we see that the lazy construction produces a stable improvement in the time complexity even as we increase the data size.

—By far, the best improvement is given by the indexing enhancement, as shown in Figure 5(c), with steadily increasing improvement with larger data sizes.

—Figure 5(d) tracks the performance improvement from pruning by actors along the *min_link_weight* axis. It shows modest improvement even at the highest *min_link_weight* values.

Thus, the proposed algorithm enhancements are shown to produce measurable improvements in the performance of the algorithms.

6. EXPERIMENTS ON PHYSICAL LOCATION DATA

The objectives of experiments on this dataset are to cross-validate the experiments on *cyberdata* to see if the results stand, as well as to find new insights peculiar to *Physical Location Data* (or *physicaldata*).

6.1 Dataset

This data is collected as a log of base stations (situated at known physical locations) that each user connects to in order to gain access to the wireless network. Each tuple $\langle a, t, s\rangle$ of this data consists of a user a, a time stamp t, as well as a location s. A location may not correspond precisely to where a user is located physically, but rather to the closest base station to which this user's wireless device is connected to. Each base station serves an area within a 25–100m radius, with more base stations placed in crowded areas. As the locations most frequented by our users are relatively crowded, it was likely that the closest base station would be nearby. This degree of uncertainty is tolerable, given that our model considers not only spatial proximity, but also temporal, proximity and repetitive co-occurrences over space and time. A location in this data consists of three levels, from the most general to the most specific: building (level 1), floor (level 2), and room (level 3). Unless otherwise specified, we use all three levels of location granularity.

Each tuple originally referred to a wireless device, identified by a device id. With the availability of other types of data from DHCP and firewall servers, a device id could be mapped back to a real user, with some data loss. DHCP data allowed mapping a device id to an IP address allocated to that device at a particular point of time. Firewall data allowed mapping an IP address to a user name, which identified a real user. The mapping was time-sensitive, that is, a device id could be successfully mapped to a user name only if there were matching DHCP and firewall records within a small time window (5 minutes). Unsuccessful mapping resulted in data loss. Increasing the mapping time window would reduce the data loss, but would also reduce the mapping confidence.

We again retained only data concerning users who appeared at least once in each of the six months (August to October 2004 and January to March 2005). There were 75 such users, moving over 63 unique level-3 locations. Of these users, 73 have at least one known demographic attribute value. The data size after preprocessing was 34 thousand tuples or 1.41MB, which was much smaller than *cyberdata*. Because of the data loss due to mapping and the much smaller size of *physicaldata*, we decided to use *cyberdata* as the primary data for experiments and *physicaldata* as a secondary data for verification.

6.2 Demographic Similarity

Using *physicaldata* in place of *cyberdata*, we repeat the experiments in Section 5.2 (demographic similarity) with the parameter settings given in

Table VI. Parameter Values (*physicaldata* Aug-Sep04)

Parameter	Default Value	Range
$\vert D\vert$	14 thousand	0 – 14 thousand
δ_{max}	2 hours	10 minutes – 16 hours
$\mathit{min\_link\_weight}$	0	0 – 25
$\mathit{min\_event\_weight}$	0	0

Table VII. Demographic Similarity (*physicaldata* Aug-Sep04)

		STEvent (%)			*Spatial*	
Common Features	*Random* (%)	*Level 1*	*Level 2*	*Level 3*	*Level 3*	*Temporal*
at least 1	53.5	91.0	97.0	100.0	94.0	69.0
at least 2	19.3	66.0	74.0	87.0	77.6	31.0
at least 3	13.2	64.1	66.7	71.1	75.6	20.3

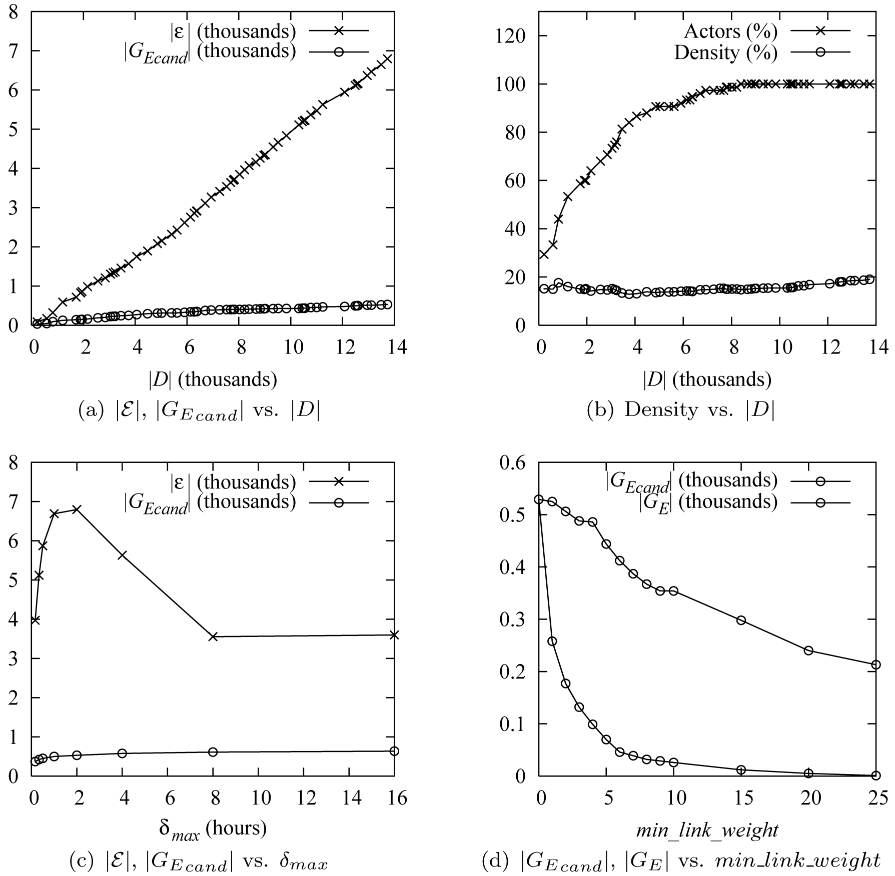
Table VIII. Demographic Similarity across Periods (*physicaldata*)

		STEvent (%)				
Common		*Aug-*	*Sep-*	*Oct04-*	*Jan-*	*Feb-*
Features	*Random* (%)	*Sep04*	*Oct04*	*Jan05*	*Feb05*	*Mar05*
at least 1	53.5	100.0	100.0	100.0	98.0	100.0
at least 2	19.3	87.0	90.0	84.0	78.0	83.8
at least 3	13.2	71.1	74.4	69.1	61.8	59.3

Table VI. We again compare *STEvent*'s distributions to the *Random*'s distribution drawn from the set of 75 actors. This *Random* distribution is slightly different from that for *cyberdata*, especially for pairs sharing three common attribute values. This difference is not surprising, as these pairs are drawn from a smaller set of actors (75 vs. 533 actors), with a different underlying distribution in terms of proportions of attribute values. On the other hand, for *physicaldata*, the *STEvent* similarity values are much higher as compared to *Random*. Applying a similar χ^2 goodness-of-fit test at 5% level [Walpole et al. 2002], as before, reveals that all *STEvent* distributions, for each level or each period, are sufficiently dissimilar from *Random* distribution, confirming the significant improvement that visual inspection alone has suggested. As for *cyberdata*, *STEvent*'s top 100 links involves a smaller number of users (38), as compared to *Random* (70).

Table VII also shows the demographic similarity for *Spatial* and *Temporal*. Comparing these values with those for *Random* and *STEvent*, we see that generally *STEvent* has higher demographic similarity than *Spatial* and *Temporal*, which in turn have higher demographic similarity than *Random*. In contrast to the finding in Section 5.2, *Spatial* now has higher similarity than *Temporal*, and has the highest "at least 3" similarity among all methods. For *physicaldata*, spatial co-occurrence is more correlated with similarity in demographic attributes such as *department*, which explains *Spatial*'s higher demographic similarity.

While the χ^2 homogeneity test at 5% level [Walpole et al. 2002] for values in Table VIII suggests that the *STEvent* distribution is more or less uniform for all periods, that for values in Table VII suggest that the distributions of the different levels are not uniform; in fact they seem to be improving at deeper



(a) $|\mathcal{E}|$, $|G_{Ecand}|$ vs. $|D|$

(b) Density vs. $|D|$

(c) $|\mathcal{E}|$, $|G_{Ecand}|$ vs. δ_{max}

(d) $|G_{Ecand}|$, $|G_E|$ vs. *min_link_weight*

Fig. 6. Algorithmic behavior (*physicaldata* Aug-Sep04).

levels (more specific locations). The latter is a result that *cyberdata* is not able to produce.

6.3 Algorithmic Behavior

Using *physicaldata* in place of *cyberdata*, we repeat the experiments in Section 5.3 (algorithmic behavior) with the parameter settings given in Table VI. The results are illustrated in Figure 6. A quick visual inspection comparing Figure 3 and Figure 6 reveals a remarkably similar set of trends in all four subfigures, despite the significant difference in data sizes between the two datasets. Although the density line in Figure 6(b) seems flat, in fact it increases very slightly as the number of actors stabilizes, displaying a trend similar to the density line in Figure 3(b). Other than noting this consistency between the two datasets, we would not go into any further detail, as we believe the previous discussion on *cyberdata* still applies here.

A comparable figure to Figure 5 on time complexity is not shown here, because given the relatively much smaller data size, the time taken for experiments on *physicaldata* is almost negligible and does not show any clear or notable trend.

6.4 Comparison: Cyber Location vs. Physical Location

To summarize the relationship between the two datasets, we highlight their differences and similarities, as follows.

The two datasets have data sizes (tuples, actors, locations) of different orders of magnitude, with *cyberdata* being the much larger one. Seeking overlap between the top 100 pairs of the two datasets in each period does not yield many common pairs (10%, 11%, 11%, 9%, 7%). Moreover, the demographic similarity distributions of *physicaldata* is generally much higher than those of *cyberdata*. These hint at the different kinds of semantics of relationship that can be mined from those datasets. *physicaldata* co-occurrences seem to correlate more with the notion of similarity than *cyberdata* co-occurrences.

On the other hand, it has also been shown affirmatively how the algorithmic behaviors of the two datasets are consistent in their trends; how the demographic similarity for each respective dataset is similar across the five periods of interest; and how their respective demographic similarity distributions (across location levels and periods) consistently produce statistically significant improvements over *Random* distributions.

7. EXPERIMENTS ON REALITY-MINING DATA

The objective of experiments on this dataset is to validate our experiments on a separate, publicly available spatio-temporal dataset, namely *Reality Mining Data* (or *realitydata*) [Eagle and Pentland 2006].

7.1 Dataset

realitydata is a data set collected by a MIT group, recording the activities of 100 subjects at MIT who each carried a Nokia 6600 smart phone over the course of 2004-2005 academic year. The 75 subjects were staff and students of the MIT Media Lab, while the other 25 were incoming students at the MIT Sloan School of Management. A component of this dataset, which is of interest to us, is the *cellspan* table, where each tuple $\langle starttime, endtime, person\_id, celltower\_oid\rangle$ logs the *starttime* and *endtime* when a person identified by *person_id* was connected to a celltower identified by *celltower_oid*. In total, this table contains 2.5 million records representing 89 persons and 32 thousand cell tower locations.

Note that *cellspan* is similar to *physicaldata*, with the exception that the time values are intervals instead of time points. To fit our algorithms, we convert *cellspan*'s tuples into the form of $\langle time, person\_id, celltower\_oid\rangle$ by creating one tuple for every 60 seconds between the *starttime* and *endtime*.

While the data spans from January 2004 to May 2005, the level of activity is uneven for different months. In particular, we selected only the five months featuring at least 50 distinct *person_id* values, namely September 2004 to January 2005. We further retained only data concerning the 41 persons who appeared at least once in each of the five months.

Table IX. Demographic Similarity Across Periods (*realitydata*)

		STEvent (%)			
Common Features	*Random* (%)	*Sep-Oct04*	*Oct-Nov04*	*Nov-Dec04*	*Dec-Jan05*
at least 1	62.1	83.7	76.9	74.4	75.3
at least 2	42.6	54.9	54.2	53.7	53.8
at least 3	14.9	34.4	40.4	42.3	41.8
at least 4	3.9	12.5	16.7	12.5	11.1

Table X. Demographic Similarity for Spatial and Temporal (*realitydata* Sep-Oct04)

Common Features	*Spatial* (%)	*Temporal* (%)
at least 1	69.1	100.0
at least 2	51.5	60.9
at least 3	36.5	34.2
at least 4	10.5	10.7

7.2 Demographic Similarity

We conduct a similar experiment as in Section 5.2 to measure demographic similarity. In the case of *realitydata*, its *person* table contains some person attributes obtained from a survey of the participants. We selected the following four attributes. *Position* indicates whether a person is a staff or a student at the Media Lab, or belonging to the School of Management. *Regular* indicates how regular a person's hours are (e.g., not at all, somewhat, very). The other two attributes are *predictable life* (e.g., not at all, somewhat, very), and *travel* (e.g., rarely, sometimes, often, very often). These attributes roughly represent the work/lifestyle of the participants, in contrast to the attributes in Section 5.2 that represent user affiliations.

Table IX compares the demographic similarity of *Random* and *STEvent* for four different two-monthly periods (*Sep-Oct04*, *Oct-Nov04*, *Nov-Dec04*, and *Dec-Jan05*). For each of the four periods, *STEvent* has significantly higher similarity than *Random*, with 11% to 16% of *STEvent* pairs having equality for all four attributes compared to 3.9% for *Random*. The better performance than *Random* is consistent with the outcome of similar experiments with *cyberdata* and *physicaldata* (see Sections 5.4 and 6.2, respectively).

Table X shows the corresponding demographic similarity for *Spatial* and *Temporal* methods for the *Sep-Oct04* period. The three methods are generally comparable, with *STEvent* having the highest "at least 4" similarity, *Spatial* having the highest "at least 3" similarity, and *Temporal* scores very well for "at least 1". This probably implies that for this dataset, both the spatial and temporal dimensions are relatively important, and neither is extremely dominant. Hence *STEvent*, which incorporates both dimensions, perform similarly to *Spatial* and *Temporal*.

In conclusion, the experimental results for *realitydata* data generally support the outcome of our experiments on the proprietary *cyberdata* and *physicaldata* datasets.

8. CONCLUSION

In this article, we study the problem of discovering social associations from spatio-temporal data. We propose a spatio-temporal event model, called *STEvent*, to discover the existence as well as to weigh the strengths of these associations. Experimental results on two proprietary and one public real-life spatio-temporal datasets cross-validate each other to a large extent.

Importantly, trends produced by our experimental results on the real-life spatio-temporal datasets could be explained satisfactorily by either empirical observation of the results or theoretical analysis of the proposed algorithms. Furthermore, the ability to produce results that correlate with the notion of similarity, as well as the reliability in producing such results consistently for different datasets and for different time frames, lend credence to *STEvent*.

We also realize that spatio-temporal data could reveal only certain aspects of associations. Other aspects of social associations could also be mined from other types of data such as emails, cocitation, or coauthorship data. Integrating these heterogeneous datasets in order to discover richer associations would be an interesting problem for future work.

REFERENCES

ADAMIC, L. A. AND ADAR, E. 2003. Friends and neighbors on the Web. *Social Netw. 25*, 3, 211–230.

BORGS, C., CHAYES, J., MAHDIAN, M., AND SABERI, A. 2004. Exploring the community structure of newsgroups. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, 783–787.

BOYD, D. M. 2004. Friendster and publicly articulated social networking. In *Extended Abstracts of the Conference on Human Factors and Computing Systems*. 1279–1282.

CARLEY, K. 1991. A theory of group stability. *Amer. Soc. Rev. 56*, 3, 331–354.

CHOUDHURY, T. AND PENTLAND, A. 2003. Sensing and modeling human networks using the Sociometer. In *Proceedings of the 7th IEEE International Symposium on Wearable Computing*. IEEE, Los Alamitos, CA, 216–222.

DOMINGOS, P. AND RICHARDSON, M. 2001. Mining the network value of customers. In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, 57–66.

EAGLE, N. AND PENTLAND, A. 2006. Reality mning: Sensing complex social systems. *Person Ubiquitous Comput. 10*, 4, 255–268.

FALOUTSOS, C., MCCURLEY, K. S., AND TOMKINS, A. 2004. Fast discovery of connection sub- graphs. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, 118–127.

KEMPE, D., KLEINBERG, J., AND TARDOS, E. 2003. Maximizing the spread of influence through a social network. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, 137–146.

KREBS, V. E. 2002. Mapping networks of terrorist cells. *Connections 24*, 3, 43–52.

KUMAR, R., NOVAK, J., RAGHAVAN, P., AND TOMKINS, A. 2004. Structure and evolution of blogspace. *Comm. ACM 47*, 12, 35–39.

LAMPE, C., ELLISON, N., AND STEINFIELD, C. 2006. A face(book) in the crowd: Social searching vs. social browsing. In *Proceedings of the 20th ACM Conference on Computer Supported Cooperative Work*. ACM, New York, 167–170.

LIN, S. AND CHALUPSKY, H. 2003. Unsupervised link discovery in multi-relational data via rarity analysis. In *Proceedings of the 3rd IEEE International Conference on Data Mi*ning, IEEE, Los Alamitos, CA, 171–178.

MCPHERSON, M., SMITH-LOVIN, L., AND COOK, J. M. 2001. Birds of a feather: homophily in social networks. *Annual Rev. Sociology 27*, 415–444.

RESIG, J., DAWARA, S., HOMAN, C. M. , AND TEREDESAI, A. 2004. Extracting social networks from instant messaging populations. In *Proceedings of the Workshop on Link Analysis and Group Detection* (in conjunction with *the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*).

RICHARDSON, M. AND DOMINGOS, P. 2002. Mining knowledge-sharing sites for viral marketing. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, 61–70.

SCHWARTZ, M. F. AND WOOD, D. C. M. 1993. Discovering shared interests using graph analysis. *Comm. ACM 36*, 8, 78–89.

TERRY, M., MYNATT, E., RYALL, K., AND LEIGH, D. 2002. Social net: Using patterns of physical proximity over time to infer shared interests. In *Extended Abstracts of the ACM Conference on Human Factors in Computing Systems*. ACM, New York, 816–817.

WALPOLE, R. E., MYERS, R. H., MYERS, S. L., AND YE, K. 2002. *Probability & Statistics for Engineers & Scientists* 7th Ed., Prentice-Hall, Englewood Cliffs, NJ.

WANG, Y., LIM, E., AND HWANG, S. 2003. On mining group patterns of mobile users. In *Proceedings of the 14th International Conference on Database and Expert Systems Applications*. 287–296.

WASSERMAN, S. AND FAUST, K. 1994. *Social Network Analysis: Methods and Applications*. Cambridge University Press.

XU, J. J. AND CHEN, H. 2005. CrimeNet explorer: A framework for criminal network knowledge discovery. *ACM Trans. Inform. Syst. 23*, 2, 201–226.

YU, B. AND SINGH, M. P. 2003. Searching social networks. In *Proceedings of the 2nd Joint Conference on Autonomous Agents and Multiagent Systems*. ACM, New York, 65–72.

ZHANG, J. AND VAN ALSTYNE, M. 2004. SWIM: Fostering social network based information search. In *Extended Abstracts on Human Factors in Computing Systems*. ACM, New York, 1568–1568.

Received December 2008; revised June 2009; accepted July 2009