Individual and Group Dynamics in the Reality Mining Corpus

Charlie K. Dagli and William M. Campbell
MIT Lincoln Laboratory
Lexington, MA 02420
{dagli,wcampbell}@ll.mit.edu

Abstract—Though significant progress has been made in recent years, traditional work in social networks has focused on static network analysis or dynamics in a large-scale sense. In this work, we explore ways in which temporal information from sociographic data can be used for the analysis and prediction of individual and group behavior in dynamic, real-world situations. Using the MIT Reality Mining corpus, we show how temporal information in highly-instrumented sociographic data can be used to gain insights otherwise unavailable from static snapshots. We show how pattern of life features extend from the individual to the group level. In particular, we show how anonymized location information can be used to infer individual identity. Additionally, we show how proximity information can be used in a multilinear clustering framework to detect interesting group behavior over time. Experimental results and discussion suggest temporal information has great potential for improving both individual and group level understanding of real-world, dense social network data.

I. INTRODUCTION

As a consequence of changing economic and social realities, the increased availability of large-scale, real-world sociographic data presents great opportunities for advances in social computing.

Traditional work in social networks has focused on static situations (i.e., classical social network analysis) or dynamics in a large-scale sense (e.g., disease propagation). As the availability of large, dynamic data sets continues to grow, so will the value of automatic approaches that leverage temporal aspects of social network analysis. Dynamic analysis of social networks is a nascent field that has great potential for algorithm research and development.

In this paper, we explore ways in which sociographic data can be used for the analysis and prediction of individual and group behavior in dynamic real-world situations. Specifically, we explore identity, organizational structure and patterns of life in dynamic social networks (DSN [1]) through analysis of the highly-instrumented Reality Mining data set gathered at the Human Dynamics Laboratory at the MIT Media Lab [2].

II. RELATED WORK

Analysis of time-varying social network data is an area of growing interest in the research community. Dynamic social network analysis (DSNA) seeks to analyze the behavior of social networks over time [1], detecting re-curring patterns [3][4], community structure (either formation or dissolution), [5][6] and typical [7][8][9] and anomalous [10] behavior.

Previous studies of data of this type have looked into general analysis of coarse-level social dynamics. Hierarchical Bayesian topic models [11], Hidden Markov Models [12] and eigen-modeling [13] have been used for individual and group routine discovery in the Reality Mining data at the “work,” “home,” “elsewhere” granularity level. In [2], relational dynamics in the form of spatial proximity and calling data were used to infer both friendship and job satisfaction. Call data is also used in [14] to cluster aggregate activity in different regions of metropolitan Rome.

In this work, we elaborate on work in this area by studying how fine-scale pattern-of-life features can be used at both the individual and group level to gain insight into the behavior of social interactions in a social community.

Specifically, our main contributions can be summarized as follows:

- We show how individual participant identity can be detected to a high degree of accuracy from patterns learned only from cell tower location sequences.
- We show how group dynamics can be co-clustered in time using computationally tractable multilinear algebra techniques which help highlight otherwise unseen temporal behavior.

The organization of this paper is as follows. In Section III, we describe the Reality Mining data set. In Section IV, we begin the dynamic analysis of the Reality Mining data by considering if (and to what degree) cell tower locations can be used to determine the pattern of life, and thus identity, of individual study participants. Section V explores how multilinear algebra techniques can be used to investigate group dynamics and patterns of life using fine-level features in the form of bluetooth detections. Finally, in Section VI, we summarize our approach and findings and suggest directions for future work.

III. REALITY MINING DATA SET

The Reality Mining project was conducted from 2004-2005 by the Human Dynamics Laboratory (HDL) at the MIT Media Laboratory. The study followed ninety-four subjects using mobile phones pre-installed with several pieces of software that
recorded and sent the researcher data about call logs, Bluetooth devices in proximity of approximately five meters, location in the form of cell tower IDs, application usage, and phone status. Subjects were observed using these measurements over the course of nine months and included students and faculty from two programs within the institute. Self-report relational data from each individual was collected, where subjects were asked about their proximity to, and friendship with, others.

The subjects from the Reality Mining study consisted of students and staff at MIT during the months between September 2004 and June 2005. For this paper, 94 subjects that had completed the survey conducted in January 2005 were used for analysis. Of these 94 subjects, 68 were colleagues working in the same building on campus (90% graduate students, 10% staff) while the remaining 26 subjects were incoming students at the universitys business school. The subjects volunteered to become part of the experiment in exchange for the use of a high-end smartphone for the duration of the study. Interested readers are referred to [2] for a more detailed description of the Reality Mining project.

The Reality Mining data was obtained from the MIT HDL in anonymized form. All personal data such as phone numbers were one-way hashed (MD5) generating unique ids used in the analysis. MIT HDL found that although subjects were initially concerned about the privacy implications, less than 5% of the subjects ever disabled the logging software throughout the 9-month study. Data was reformatted into a MySQL database to enable easier querying and anomalous information filtering.

IV. IDENTITY VIA DYNAMIC LOCATION INFORMATION

Our first set of experiments from the Reality Mining dataset considered individual patterns of life. Our investigations looked at the uniqueness of these patterns. In other words, can individual identity be determined through pattern of life? Note this also has implications for anonymizing data and privacy.

The Reality Mining dataset, at a top level, is an interesting challenge for this problem. The subjects in the study are in close proximity on a university campus and thus are good “foils” for each other. Individuals in the study might have confusable location histories because of similarities of life patterns (classes, group meetings) and close proximity of services (dorms, cafeterias, etc.).

A. Data Setup

As discussed in Section III, the Reality Mining data set included location data in the form of both Bluetooth and cell phone data. All of this data was used in a relative sense; i.e., no knowledge of the absolute location of the sensors was assumed. We focused on cell phone data for our experiments because of its sparseness and also to contrast with our later work with fine-grain, noisy bluetooth detection data in Section V.

A typical SQL query of the location data,

```sql
select * from locs where (myhn=44) order by d, t limit 4;
```

might return the following:

<table>
<thead>
<tr>
<th>myhn</th>
<th>d</th>
<th>t</th>
<th>areaid</th>
<th>cellid</th>
</tr>
</thead>
<tbody>
<tr>
<td>44</td>
<td>2004-11-01</td>
<td>18:25:30</td>
<td>5119</td>
<td>40813</td>
</tr>
<tr>
<td>44</td>
<td>2004-11-01</td>
<td>18:26:01</td>
<td>5119</td>
<td>40332</td>
</tr>
<tr>
<td>44</td>
<td>2004-11-01</td>
<td>18:26:18</td>
<td>5119</td>
<td>40811</td>
</tr>
<tr>
<td>44</td>
<td>2004-11-01</td>
<td>18:27:13</td>
<td>5119</td>
<td>40332</td>
</tr>
</tbody>
</table>

In the table, “myhn” is the hash number of the individual, “d” is the date of the observation, “t” is the time, “areaid” is the cell area, and “cellid” is the cell tower identification. We can see from the short example, that the update frequency can be quite frequent.

We organized the data from the Reality Mining according to date. The span of the data was from September 2004 to June 2005. The data was partitioned into two training sets:

- **Training Set 1 (TR1)** Data collected between September 2004 and December 1, 2004
- **Training Set 2 (TR2)** Data collected between March 1, 2005 and March 31, 2005

We also partitioned the data into multiple test sets. The test sets were **TST1** Jan 1–Feb 1, 2005; **TST2** Feb 1–March 1, 2005; **TST3** March 1–April 1, 2005, and **TST4** April 1–June 1, 2005. Only a modest amount of data was available after May 1 and this was pooled into TST4.

Both training instances and test instances were based upon a windowing of the data of a variable number of days (1-5). For example, suppose a test of identity is conducted with a 2 day window using TST4. Then, for every 2 day period (non-overlapping), April 1-2, April 3-4, April 5-6, etc., a test instance is constructed. This process allowed us to understand the role of variation of the number of days of observation necessary to achieve a certain error rate.

B. Algorithms for Identity Determination from Location Data

1) Features: A typical data stream from an individual ordered over time would look like the following,

(5119,40813), (5119,40332), (5119,40811), (5119,40332), ...

where each pair is of the form (areaid, cellid).

To extract features from the data stream, we first performed the following sequence of operations. First, windowing was applied to section the sequence data into $k$ day non-overlapping intervals (as explained earlier). Second, for each windowed region, run-length encoding was used to eliminate identical symbols (i.e., duplicates in the time series were removed). Third, $n$-gram analysis of the discrete sequence was performed. Both the cellid and areaid were combined to a single symbol. As an example for bigrams (2-grams), the sequence just presented would be transformed to,

5119_40813|5119_40332, 5119_40332|5119_40811, 5119_40811|5119_40332, ...

We then use the idea “bag of $n$-grams” approach to extract features and perform classification [15]. For each window, we
calculate the probability of an $n$-gram in the window,

$$p(d_i) = \frac{\#(t_k = d_i)}{\sum_k \#(t_k = d_k)} \tag{1}$$

where $\#(t_k = d_i)$ indicates the number of tokens in the window equal to $d_i$. These probabilities then become entries in a vector $v$,

$$v = [p(d_1) \ldots p(d_M)]^T. \tag{2}$$

In general, the vector $v$ will be sparse since a given window will not contain all potential $n$-grams. The vector $v$ is then weighted as in information retrieval with a document component. We weighted by the following function,

$$\log \left( \frac{1}{p(d_i|bkg)} \right) + 1 \tag{3}$$

In the weight equation, $p(d_i|bkg)$, is the probability of an $n$-gram calculated over all instances in the training set. The combination of (2) and (3) produces a feature that is large only if the feature is unusual (i.e., not common in the whole data set) and is highly probable in the window.

2) Classification Method: Classification was performed using a standard method—a support vector machine (SVM) with a linear kernel. An SVM is a two-class classifier constructed from sums of a kernel function $K(\cdot, \cdot)$,

$$f(x) = \sum_{i=1}^{L} \alpha_i t_i K(x, x_i) + d, \tag{4}$$

where the $t_i$ are the ideal outputs, $\sum_{i=1}^{L} \alpha_i t_i = 0$, and $\alpha_i > 0$. The vectors $x_i$ are support vectors and obtained from the training set by a quadratic optimization process using a default soft-margin parameter $C$. The ideal output are either 1 or $-1$, depending upon whether the corresponding support vector is in class 0 or class 1, respectively. For classification, a class decision is based upon whether the value, $f(x)$, is above or below a threshold.

The focus of the SVM training process is to model the boundary between classes. For a separable data set, SVM optimization chooses a hyperplane in the expansion space with maximum margin. The data points from the training set lying on the boundaries are the support vectors in equation (4).

C. Experiment Setup and Results

For our experiments, we extracted features using the method described in Section IV-B1 and the SVM classifier in Section IV-B2 using a linear classifier. For a given experiment, the window size was fixed at a value of $k$ days ($k$ was in the range 1-5).

We initially applied our technique to TRN1 and TST1. The task was detection of an individual based upon his or her cell phone profile over $k$ days. Another variable swept was the order of the $n$-gram. Performance of the system was measured in terms of $P_m$ (probability of a miss) and $P_{fa}$ (probability of a false alarm). Since the output of the SVM is a continuous value, a threshold can be set and the output compared to the threshold to detect whether an individual is present or not. Both $P_m$ and $P_{fa}$ can be displayed with detection error tradeoff.

Fig. 1. Identity Detection via Location Information. Results for the detection of identity of an individual using cell tower location data from one day on the TST1 dataset using an SVM and the TRN1 training set are shown in Figure 1(a). Comparison of different temporal contexts for location features show trigrams as the highest performing temporal context. Given this, results for the detection of identity of an individual using cell tower location data from $k$ days on the TST1 dataset using an SVM, trigrams, and the TRN1 training set are shown in Figure 1(b). Best performing test window is for data aggregated across 3 days.
(DET) curves (a form of ROC curve). Results for a 1-day test
as the order of the n-gram is varied is shown in Figure 1(a). We
can see from the figure that trigram and 4-gram performance
is best.

Our next sweep was of the number of days of testing in the
window. We varied k from 1 to 5 and used trigrams. Similar
results, following the trends of Figure 1(a), were observed for
other n-gram orders, see Figure 1(b). An interesting result to
note in Figure 1(b) is that an increasing number of days does not
necessarily mean improved performance. This phenomena
probably occurs because the location of an individual varies
from weekday to weekend and aggregating too much data
blurs the distinction.

We also considered changing the amount of training data
from the original 3-months in TRN1 to only 1-month in TRN2.
One of the difficulties encountered was that the data set has a
data collection change during spring break that causes the data
from before March 1 to be incompatible with the data after this
time. Therefore, we compare TRN1-TST1, TRN1-TST2 with
TRN2-TST4. From Figure 2, we can see that TRN2-TST4 has
excellent performance at the equal error rate (P_{FA} = P_{MA}) point
of about 9%. The additional training data provides significantly
better performance for the low false alarm region of the curve
(P_{FA} < 1%). This trend is common in identification tasks.

D. Discussion

Our simple experiments with the Reality Mining cell phone
location data showed that we could detect individuals with
reasonable accuracy, greater that 90%, with just a few days
of data. Given that this data is a subset of the data available
in a cell phone and we used no absolute location information,
this simple method shows that the identifiability of individuals
from personal device behavior is possible and indicative of
pattern of life.

A detailed examination of our technique showed that tri-
grams worked well for the task. Reasonable performance could
be obtained with any training or testing window of 1-5 days.
Performance improved at low false alarm rates if several
months of data were available.

V. INFERRING DYNAMIC COMMUNITY BEHAVIOR

Next we look at dynamics at the group level and show
how pattern of life can be detected. In this section, we seek a
general approach for dynamic social network analysis that will
allow for fast, high-level summaries of group dynamics in real-
world social networks. To this end, we investigate the use of
multilinear semantic indexing (MLSI) [10][16] in the context of
dynamic social networks. Multi-modal co-clustering tools,
based on tensor modeling and analysis, can be successfully used
to provide summaries of community structure and behaviour.
We first begin with a brief overview of MLSI.

A. Multilinear Semantic Indexing

Multilinear Semantic Indexing is a generalization of tra-
ditional Latent Semantic Indexing (LSI). To see this con-
nection, consider the use-case of text document clustering.
Traditional LSI relies on the rank-k SVD decomposition of the
term-document matrix, a matrix of (weighted) term-frequency
(rows) as a function of corpus document (columns). This de-
composition creates topic-term and topic-document clustering
through the respective sets of k left and right singular vectors.
These vectors are called aspect-profiles. Additionally, large
singular values weight the strength of correlation between
pairs of document / term aspect-profiles, serving to co-cluster
documents and terms within the corpus.

MLSI generalizes this idea to create multi-modal co-
clustering from higher-order tensor representations of the data:
term-document-author tensors for the text document cluster-
ing example. Through High-Order SVD (HOSVD), MLSI
produces M multi-modal aspect-profiles (topic-term, topic-
document, topic-author) as well as multi-modal co-clustering
between M-tuples of aspect-profiles through the multi-linear
equivalent of singular values.

In the next section we briefly discuss HOSVD via Tucker
Decomposition, introducing concepts in multilinear algebra as
necessary.

B. Tensors and the Tucker Decomposition

A tensor of order M can be thought as a multiway array
with M indices. Formally, an M-th order tensor can be written as
\[ X = (x_{i_1 \cdots i_M}) \in \mathbb{R}^{N_1 \times N_2 \times \cdots \times N_M} \] (5)
where \( \{N_i\}_{i=1}^M \) are the dimensions for each of the M indices.
Matrices and vectors are considered second and first-order
tensors, respectively.
Given an input tensor, $\mathcal{X}$, as in (5), the most general tensor decomposition is the Tucker decomposition:

$$\mathcal{X} \approx \sum_{j_1=1}^{K_1} \cdots \sum_{j_M=1}^{K_M} y_{j_1, \ldots, j_M} u_{1,j_1} \odot \cdots \odot u_{M,j_M}$$

(6)

where $u_{i,j} \in \mathbb{R}^{N_i}$ for all $j = 1 \ldots K_i$ and $K_i \leq N_i$ for all $i$.

The tensor $\mathcal{Y} = (y_{1, \ldots, i_M})$ is called the core tensor. The $M$-way outer product

$$Z = (z_{k_1, \ldots, k_M}) = u_{1,j_1} \odot \cdots \odot u_{M,j_M}$$

$$z_{k_1, \ldots, k_M} = u_{1,j_1}(k_1) \cdots u_{M,j_M}(k_M)$$

(7)

is a generalization of the vector outer product and produces a rank-1 tensor.

In compact multilinear notation, the Tucker Decomposition is represented as $\mathcal{X} \approx \{\mathcal{Y}; \{U_i\}_{i=1}^M\}$ where each $U_i \in \mathbb{R}^{N_i \times K_i}$ is a matrix whose columns correspond to $\{u_{i,j} \in \mathbb{R}^{N_i}\}_{j=1}^{K_i}$.

The projection matrices, $\{U_i\}_{i=1}^M$, need not be orthogonal, but if they are, the Tucker decomposition is considered a higher-order singular value decomposition (HOSVD) of the input tensor $\mathcal{X}$ [17].

HOSVD algorithms seek to find the best approximation to the input tensor by minimizing the least-squares reconstruction error, $\text{argmin}_{\{U_i\}_{i=1}^M} \|\mathcal{X} - \mathcal{X}\|_F^2$, where $\|\|_F$ is the Frobenius norm and $\mathcal{X}$ is as given in (6). A standard solution to this problem can be obtained (approximately) by an alternating least-squares algorithm [16].

C. Interpreting Tucker Decomposition as MLSI

In summary, a Tucker decomposition is a linear combination of $M$-tuples of column vectors from matrices $\{U_i\}_{i=1}^M$ weighted by the value of the core tensor $\mathcal{Y}$ at the same $M$-tuple index.

Accordingly, larger absolute values in the core tensor mean a larger contribution to the final reconstruction, which therefore implies the relative importance of the $M$-tuple of subspace vectors (as a group) in approximating the input tensor.

In this way, the vectors in these $M$-tuples can be considered strongly correlated. Values in the core tensor highlight these correlations, and as a result provide meaningful co-clustering between $M$-tuples of vectors from each modality’s projection space. In this way, the core tensor serves an analogous role as the singular values in traditional LSI. The columns of the projection matrices correspond to a multi-modal version of traditional LSI aspect-profiles.

D. Time-Profile-Specific Sub-Networks

In the case of time-varying social network interactions (where we consider time as the third index in an order-3 data tensor where indices 1 and 2 are relationships between actors) we propose Tucker decomposition maybe more easily interpreted through time-profile specific sub-networks.

In the case of an order-3 tensor, as in the dynamic social network analysis problem, consider re-writing Equation (6) for $M=3$ as follows:

$$\mathcal{X} = \sum_{j_1=1}^{K_1} \sum_{j_2=1}^{K_2} \sum_{j_3=1}^{K_3} y_{j_1, j_2, j_3} u_{1,j_1} \odot u_{2,j_2} \odot u_{3,j_3}$$

$$= \sum_{j_3=1}^{K_3} X_{j_3} \odot u_{3,j_3}$$

(8)

where we define the matrix $X_{j_3}$ as the time-profile-specific sub-network correlated with time-profile, $u_{3,j_3}$. The polarity of the social interactions present in the sub-network $X_{j_3}$ implies positive and negative correlation with the time-profile, respectively.

This matrix can be interpreted as the adjacency matrix of the social interactions correlating most strongly to a given time-profile. When time is the third index in the data tensor, this may be a more natural way to interpret a Tucker decomposition as opposed to MLSI where the list of most active participants from each correlating pair (or tuple) of indices is returned to a user in list form.

As a final comment, the polarity of values in a time-profile extracted via tensor decomposition may sometimes introduce ambiguity into the interpretation of the produced summaries. This may be especially true for negative values in the time-profile as seen in Figure 4(c). Traditional MLSI handles this issue by returning the absolute value of all values in the decomposition. This may not always be an appropriate strategy, however, because it ignores information we may not want to disregard.

This is most easily seen if we consider the reconstruction of the original data. In this context, a time-profile can be seen as the signal that modulates the corresponding sub-network over different time-stamps throughout the tensor. The lower-order sub-network / time-stamps will represent the high-energy, “low frequency” information in the tensor. Accordingly, they will give a “smoothed” estimate of the original data as seen in Figure 4(a). As higher-order pairs are introduced, they may need to “subtract” out some of the smoothed signal at particular times to introduce more “high frequency” content back into the reconstruction. Accordingly, the time-profile values at these time stamps will be negative as seen in Figure 4(c). These observations are similar to those in traditional matrix decomposition.

To gain insight on the ability of this approach to summarize group behavior from dynamic social network data, we analyze close-range social interactions from the Reality Mining corpus.

E. Data Set Pre-Processing

For our experiments, we used Bluetooth proximity information from periodic scans of nearby devices from each of the 106 participants in the Reality Mining study. We found the proximity information resulted in more dense interaction networks than call data when considering small time increments.

We use the number of detections between study participants, per hour as the social interaction feature. Restricting the time
range to the academic year (September 1, 2004 to May 15, 2005) and removing participants lacking Bluetooth data, this results in a mode-3 data tensor of size 95 x 95 x 6168. That is, for input tensor \( X \), \((x_{i,j,k})\) corresponds to the number of times participant \( i \) was detected by participant \( j \) at hour \( k \).

We normalize the values in the input tensor by \( \log(1 + x_{i,j,k}) \) to prevent large values from dominating the tensor decomposition.

For ease of interprebility, we re-order the indices of the input tensor using spectral clustering. To do so, we calculate the second-largest eigen-vector of the normalized graph-Laplacian of the time-marginalized social network (Figure 3(a)). The values in this eigenvector are then sorted, and the resulting re-ordering of indices is used to re-order indices in our input tensor.

\[ F. \text{Experimental Protocol} \]

For each data set, we perform MLSI via a rank-(25,25,25) Tucker decomposition of the pre-processed input tensor, keeping the original time-resolution on samples. The size of the decomposition was chosen heuristically to represent a reasonable number of profiles users could sort through when interpreting the results. Both the size of the decomposition as well as the time-resolution of the input tensor are both open issues when using this approach. Results are generally evaluated qualitatively.

\[ G. \text{Results} \]

Looking at the social profile-vectors, as represented by the columns of the matrix shown in Figure 3(b)), we observe the first two vectors correspond loosely to the two main communities seen in the Reality Mining data, namely the Media Lab and Sloan participants. We observe subsequent profile vectors correspond to the larger, more active Media Lab participants. This makes sense as the Tucker decomposition seeks to best represent the data in a least-squares sense, and in so doing biases its representation toward the largest, most active community, which in this case is the Media Lab. The Sloan community is more succinctly represented by profile 2. In general, we observe reasonable community clustering and co-clustering where it exists.

Temporally, we observe the expected gaps in the time-profiles corresponding to Thanksgiving (MIT calendar [18]: November 25-28, observed: November 25-29 and Winter Break (MIT calendar [18]: December 18 - January 2 \(^2\), observed: December 24 - January 2). We were able to detect the disbanding of the Sloan community as a whole after the Fall semester, which was easily observable in the sharp drop-off of signal energy in time-profile 2 and higher-order time-profiles.

Additionally, through spectral analysis, we observe the time-profiles also exhibit behaviour corresponding to daily and weekly routines: the two largest spectral components (on average) are 1/24 hours and 1/163 hours (1/6.8 days). This specific result was also observed by [19].

\(^2\)Last day before Independent Activities Period

The time-profiles seen in Figure 3(c), are more clearly interpreted in the context of their correlated sub-networks which are shown in Figure 4(a) and elaborated on in the following section.

1) Time-Profile Specific Sub-Networks: We divide this section into community-specific results, namely for the two main communities represented among the participants in the study: the Media Lab and Sloan Graduate Students. We begin first with the Media Lab.

Medidd Lab Community

The first time-profile-specific sub-network, Figure 4(a), is the most relevant sub-network for the Media Lab community, and in general, a compact first-order summary of the data. The time-profile exhibits a fairly uniform structure across all time instances, with general periodic features consistent with work and school related activities throughout the academic year, most noticeably the gap in the time-profile corresponding to the break between Fall and Spring semesters. The sub-network shows the majority of the social interactions occur between the larger Media Lab community (with a relatively smaller proportion occurring in the Sloan community).

This differs from the information conveyed in the average social-network derived from marginalizing interactions over time (seen in Figure 3(a)). When time information is removed, it appears the Sloan community is equally active as the Media Lab community during this time period. In reality, as this sub-network and subsequent sub-networks arising from tensor analysis will show, this is not the case.

Looking more closely, we observe a small core of stronger interactions among a sub-set of Media Lab participants. This smaller core can be more clearly seen in the sub-network most closely correlated to time-profile 3, Figure 4(c). This smaller core of Media Lab participants is active throughout the study, but as seen in Figure 4(c), most strongly correlated to the beginning of the data collection. We know from Figure 4(a) that this is the most active group among the Media Lab participants, other participants aren’t as active in the beginning of the study, and that many of these participants self-identify as being more senior than other members. Accordingly, we hypothesize this sub-network may correspond to project leads.

Sloan Graduate Students

The most interesting results of our tensor analysis of the Reality Mining data were sub-networks (and associated time-profiles) correlating with the Sloan graduate student community.

As can be seen in Figure 4(b), both the Sloan student sub-network and the corresponding time-profile are detected cleanly. From the time-profile, we observe this sub-network appears strongly only in the Fall semester.

Figure 5(b) lists the most prominent time-stamps from each peak cluster (sorted chronologically) from this time-profile. Interestingly, the strongest periodic behavior can be observed on Tuesdays and Thursdays at 11am from the middle of October to the beginning of December.
Marginal Adjacency and Tucker Decomposition of Reality Mining Data. Distinct community structure is observed in the time-marginalized adjacency matrix of the Reality Mining data set, Figure 3(a). The three delineated blocks correspond to “General Graduate,” “MIT Media Lab,” and “Sloan” student clusters respectively. The Tucker decomposition is shown in Figure 3(b) and 3(c). Due to the symmetric property of the adjacency matrix, indices 1 and 2 produce identical projection matrices and so we include only one copy here for clarity. It is easy to see clear community structures in the social network profiles, most notably the distinction between Media Lab and Sloan students.

If we zoom into the time-profile, we observe local peaks consistent with this result. Figure 5(a) shows a three-week window between October 24 and November 22, 2004. The red bars on the time-axis delineate weekday from weekend. At this finer time-resolution, we also observe an additional weekly sub-peak associated with Thursdays at 4pm. These same Tuesday/Thursday 11am, Thursday 4pm spikes appear locally throughout the profile, not just in weeks associated with the global peaks.

Since spikes are clearly periodic, persist through the Fall semester then disappear in the Spring, the evidence suggests this behavior could be due to course attending activity. Of all Sloan course offerings in Fall 2004, only two courses are consistent with this behavior [20]. Of these, only the first-year core course 15.515 Financial Accounting [21], Figure 5(d), fits all the observed evidence. As seen in Figure 5(c), 15.51 has three lecture sessions which meet Tuesdays and Thursdays from 10:30-12, and two recitation sessions which meet Thursdays at 4pm. As a result, one could make the uncertain, yet probable, claim that the behavior observed in this time-profile corresponds to Sloan students attending 15.515 Financial Accounting.

These are examples of scenarios ideally suited for this approach, namely those situations where actors co-cluster cleanly in space and time. Given the appropriate context, tensor analysis allowed us to establish (with some degree of uncertainty) a causal link between an observed behavior and a generating event, a result which was not readily apparent from the input data.

H. Caveats

Tensor-based tools for analyzing dynamic social networks are most useful in situations requiring unsupervised, first-order summarizations of data with dense and complete interaction information. As in the case of the possible course-attending behavior in the Sloan community, when patterns exist these techniques can enable uncued discovery of interesting behavior.

When social networks are overly sparse, highly noisy, or have small communities with ephemeral behavior, these techniques may not be able to provide meaningful insights. Since tensor decomposition approaches are driven by minimizing the Froebinus norm of reconstruction error, they are driven by preserving “signal” energy. In cases where communities have dense interaction information between individuals and persistent temporal behavior, enough energy exists to be represented by the decomposition. When communities are incomplete or have negligible temporal extent, they will end up being “washed-out,” or smoothed-away in the low-rank approximation of the data a tensor decomposition provides.

It is in these situations were more fine-grain techniques are more appropriate. As a reviewer brought to our attention, recent work applying probabilistic topic modeling approaches to time slices of bluetooth proximity information allows for fine-grain inference of time-based social interactions [22]. Such approaches help cast computationally light, high-level summarization techniques such as MLSI as more appropriate as a pre-filtering stage for finer-resolution yet more computationally intensive techniques such as [22].

VI. CONCLUSIONS

In this paper, we explored the analysis of dynamic sociographic data. In particular, we have shown temporally-driven analysis of the Reality Mining data allows us to gain unique insights into both individual and group behavior unobservable from static snapshots.

Our experiments with the Reality Mining cell phone location data showed that individuals could be detected with accuracy of greater that 90% with just a few days of data. We show how relative location information is indicative of pattern of life and that the identifiability of individuals from personal device behavior is possible.
Fig. 4. Time-Profile Specific Sub-Networks for Reality Mining Data Set. Sub-networks and their correlated time-profiles are shown in pairs for time-profiles 1 (Figure 4(a)), 2 (Figure 4(b)) and 3 (Figure 4(c)) respectively. Positive and negatively correlated sub-networks are shown by plotting the absolute values of the positive and negative parts of $X_j$, respectively. This representation shows the same information as Figure 3, but in a more natural manner.
Using trigrams of location context, reasonable performance could be obtained with any training or testing window of 1-5 days. If several months of data are available, performance at low false alarm rates improve significantly.

We have shown how tensor representation and decomposition of dynamic social network data can be used to find fast, first-order summarizations of communities and their behavior over time. We have shown that when communities are dense (complete interaction information) and persistent in their behavior, high-order summarizations are able to capture meaningful information, in some cases with remarkable detail. This behavior persists throughout the Fall semester, and disappears in the spring. From this analysis, we hypothesize the sub-network corresponding to this time-profile could be those Sloan students attending the first-year core course, 15.515 Financial Engineering, Figure 5(d).

ACKNOWLEDGMENT

The authors would like to thank Nathan Eagle and Professor Sandy Pentland of the Human Dynamics Laboratory at MIT Media Lab for access to, and helpful discussions regarding the Reality Mining Data Set.

REFERENCES


70