PairFac: Event Analytics through Discriminant Tensor Factorization

Xidao Wen
School of Information Sciences
University of Pittsburgh
xidao.wen@pitt.edu

Yu-Ru Lin
School of Information Sciences
University of Pittsburgh
yurulin@pitt.edu

Konstantinos Pelechrinis
School of Information Sciences
University of Pittsburgh
kpele@pitt.edu

ABSTRACT

The study of disaster events and their impact in the urban space has been traditionally conducted through manual collections and analysis of surveys, questionnaires and authority documents. While there have been increasingly rich troves of human behavioral data related to the events of interest, the ability to obtain hindsight following a disaster event has not been scaled up. In this paper, we propose a novel approach for analyzing events called `PairFac`. `PairFac` utilizes discriminant tensor analysis to automatically discover the impact of a major event from rich human behavioral data. Our method aims to (i) uncover the persistent patterns across multiple interrelated aspects of urban behavior (e.g., when, where and what citizens do in a city) and at the same time (ii) identify the salient changes following a potentially impactful event. We show the effectiveness of `PairFac` in comparison with previous methods through extensive experiments. We also demonstrate the advantages of our approach through case studies with real-world traffic sensor data and social media streams surrounding the 2015 terrorist attacks in Paris. Our work has both methodological contributions in studying the impact of an external stimulus on a system as well as practical implications in the area of disaster event analysis and assessment.

Keywords

Urban Computing; Tensor Factorization; Event Analytics

1. INTRODUCTION

Analyzing the impact of disastrous events has been central in understanding and responding to crises. Effective crisis management requires not only careful planning and preparation for disaster relief operations, but also a timely assessment of an event's impact to facilitate actions that will bring the society back its normal operations as fast as possible [16]. In this work, we introduce a novel event analysis framework that can automatically reveal the changes of human behavioral patterns associated with an event through mining context-rich urban activity data.

Traditionally, the assessment of disaster impact has primarily relied on the manual collection and analysis of surveys and questionnaires as well as the review of authority reports [17], which can be costly and time-consuming. Today, in the era of mobile and pervasive computing, increasingly rich digital human traces of routine transactions generated by citizens, businesses, and organizations, can be collected through online activities (e.g., activities on social media), sensing technologies (e.g., mobile phones and wireless sensors) and other means (e.g., crowdsourcing platforms). These rich troves of human behavioral data provide an unprecedented opportunity to closely examine (both qualitatively and quantitatively) the changes in urban activity due to events of interest. While much progress has been made in predictive event analytics, e.g., detecting or forecasting event outbreaks [19, 2, 22], automatically quantifying and capturing the impact of an event has been neglected despite its aforementioned importance.

In this paper, we introduce a novel approach that aims to automatically discover the impact of an exogenous event on multiple aspects of urban human activities, i.e., how does the event change *when*, *where* and *what* citizens normally do in a city? Our approach, called `PairFac`, formulates this as a discriminant tensor analysis problem and solves it through joint factorization of a *pair* of tensors. Specifically, given two tensors capturing urban activity data *before* and *after* a potentially impactful event, `PairFac` simultaneously learns the shared and discriminative subspaces from the tensor pairs. These reveal both the persistent and changing patterns across multiple interrelated aspects of urban activity data. Our method differs from existing literature [14, 8, 11] by introducing the discriminative weight vector that allows for automatically discerning the discriminative components. Extensive experiments on both synthetic and real-world event datasets demonstrate the effectiveness of our approach.

Our main contributions can be summarized as follows:

- We formally introduce a problem for capturing the impact of an exogenous event on the normal operations of a system using discriminant tensor analysis.
- We develop a new joint tensor factorization framework that aims to simultaneously learn the shared and discriminative components from a pair of high-dimensional data sources. Our method is able to automatically identify the discriminative components without a predefined number of shared and/or discriminative components.

CIKM'16 , October 24-28, 2016, Indianapolis, IN, USA

DOI: http://dx.doi.org/10.1145/2983323.2983837

- We conduct extensive experiments on synthetic datasets, which shows the superior performance of our method in comparison with existing work.
- We use `PairFac` to analyze the impact of the 2015 terrorist attacks in Paris through mining multiple relevant datasets including Paris traffic sensor data, Twitter data, and Foursquare check-ins. The analysis reveals several distinctive patterns of activity surrounding the shocking event.

The rest of this paper is organized as follows. Section 2 discusses related work, while Section 3 presents the problem formulation with the essential background. We introduce multiple solutions to the tensor factorization problem, including a novel algorithm that automatically learns the discriminative weights of the components in Section 4. Section 5 shows detailed quantitative results on synthetic data sets, while Section 6 presents the Paris attack case study application of `PairFac`. Finally, Section 7 concludes our work.

2. RELATED WORK

In this section we briefly introduce relevant literature to our methodology as well as to event and urban analytics.

2.1 Shared and discriminative subspace learning

The increasing availability of data from a diverse set of sources has given rise to the study of joint analysis of heterogeneous data. There exist studies for simultaneously discovering shared and discriminative subspace, using Non-Negative Matrix Factorization (NNMF). For example, Gupta et al. [7] propose a joint NNMF on two data sources through a shared subspace, while maintaining their unique variations through individual subspaces. Gupta et al. [8] further impose mutually orthogonal regularizations to separate the common and discriminative subspaces, ensuring that the shared subspace and the discriminative subspaces are mutually exclusive. Following the same idea, Kim et al. [11] relax the framework by requiring the shared subspaces to be *similar* while not necessarily being strictly identical. Regarding the shared and discriminative subspace learning in the context of tensor factorization, the framework by Liu et al. [14] - similar to [7] - separates the subspace into shared and individual subspaces. To the best of our knowledge, there has not been any work to date imposing regularization on the shared and discriminative spaces or extending a more flexible framework to the factorization of higher dimensional data sources.

2.2 Event Analytics

With the growing volume of social media that become rapidly available, there has been a rising interest in the area of event analytics through microblogs (e.g., Twitter). Researchers have approached this field from three perspectives. One line of research is geared towards event detection. A common technique is to monitor the frequency of all words and look for a sudden burst in the frequency of (a subset of) them [15]. The other line of research is to make sense of the event storyline through statistical or visual analytics. Diakopoulos et al. [6] design a visual analytic tool to help journalists and media professionals to extract news-worthy content from a large volume of social media around the events. Our work falls into a third category, which aims at studying the impact of an event on the affected population. Related work includes Lin and Margolin [13] that explores the response of Twitter users in different cities to the bombing attacks in Boston. However, the authors are particular focused on the emotional response towards the attacks. Furthermore, Bagrow et al. [4] provide a quantitative view of the behavioral changes in human activity under extreme conditions, such as bomb attacks and earthquakes, through the analysis of the mobile phone records. Also, Song et al. [18] mine the GPS records of 1.6 million users and build a system to automatically discover, analyze, and simulate the mobility of large population in severe disasters in Japan. The shortcoming of using cell phone and GPS data is that the activity context is absent. The latter significantly complicates the analysis through the increased dimensionality of the data.

2.3 Urban Computing

During the last years, there has been a significant rise in the attention to the research of urban computing and informatics. For instance, Zhang et al. [26] propose a data-driven system that reveals the real-time sensing of individual refueling behavior as well as the city-wide petrol consumption, using the GPS data from taxis. Kaltenbrunner et al. [10] use the amount of bikes available in the stations in the Barcelona to detect temporal and geographical mobility patterns within the city. One closely related work is that of Wang and Taylor [21] that uses Twitter data to study the perturbation and resilience of human mobility patterns in New York City during and after the hurricane Sandy.

Our study contributes in the urban computing area since `PairFac` is a generic framework that can be used to study the impact of various exogenous events – them being naturally or imposed by the local government. For example, the impact of a long-term construction project on the dwellers' mobility and activities can be quantified using `PairFac`.

3. PRELIMINARIES AND PROBLEM FORMULATION

In this section, we provide some basic background and preliminaries for the study, followed by the problem formulation. In particular, we provide the notations and essential background on tensors and their basic operations.

3.1 Preliminaries

3.1.1 Tensors

A tensor is a mathematical representation of a multidimensional array, i.e., an extension of concepts such as scalars, vectors and matrices to higher dimensions. Table 1 presents the notation we use in the rest of the paper. We use x to represent a scalar, $\mathbf{x}$ a vector, $\mathbf{X}$ a matrix, and $\boldsymbol{\mathcal{X}}$ a tensor. We use $\mathbf{x}_i$ to denote the i-th entry of vector $\mathbf{x}$, $\mathbf{X}_{ij}$ to denote the element of matrix $\mathbf{X}$ at position $\{i, j\}$ and $\boldsymbol{\mathcal{X}}_{ijk}$ to denote the element of tensor $\boldsymbol{\mathcal{X}}$ at position $\{i, j, k\}$. The *order* of a tensor is the number of dimensions (modes, or ways). The *dimensionality* of a mode is the number of elements in that mode. We use I_q to denote the dimensionality of the q-th mode. For example, the three-way tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}_+^{I_1 \times I_2 \times I_3}$ has three modes with dimensionality of I_1, I_2, and I_3, respectively. $\mathbb{R}_+$ indicates that all the elements of $\boldsymbol{\mathcal{X}}$ obtain nonnegative values.

3.1.2 Basic Operations

Symbol	Description
x	a scalar (lower-case letter)
$\mathbf{x}$	a vector (boldface lower-case letter)
$\mathbf{X}$	a matrix (boldface capital letter)
$\boldsymbol{\mathcal{X}}$	a tensor (boldface Euler script letter)
$\mathbf{X}_{i,j}$	the scalar at the $\{i,j\}$ position of matrix $\mathbf{X}$
$\boldsymbol{\mathcal{X}}_{i,j,k,\ldots}$	the scalar at the $\{i,j,k\ldots\}$ position of $\boldsymbol{\mathcal{X}}$
$\mathbf{X}_{(n)}$	mode-n unfolding of tensor $\boldsymbol{\mathcal{X}}$
$\mathbf{U}^{(n)}$	mode-n factor matrix of tensor $\boldsymbol{\mathcal{X}}$
$\mathbf{U}_r^{(n)}$	the r-th column in mode-n factor matrix of tensor $\boldsymbol{\mathcal{X}}$
$I_1,\ldots,I_M$	the dimensionality of mode 1, ..., M
$\mathbf{R}$	the desired rank (boldface captial letter)
$\mathbf{K}$	# of shared components (boldface captial letter)

Table 1: Description of Notations.

Mode-n matricization or unfolding: Matricization is the process of reordering the elements of an M-way array into a matrix. A mode-n matricizataion of a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_M}$ is denoted by $\mathbf{X}_{(n)} \in \mathbb{R}^{I_n \times \prod_{q \neq n}^{M} I_q}$.

Mode-n product: The mode-n matrix product of a tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_M}$ with a matrix $\mathbf{U} \in \mathbb{R}^{J \times I_n}$ is denoted by $\boldsymbol{\mathcal{X}} \times_n U$ and is a new tensor of size $I_1 \times \cdots \times I_{n-1} \times J \times I_{n+1} \times \cdots \times I_N$ with $(\boldsymbol{\mathcal{X}} \times_n \mathbf{U})_{i_1 \cdots i_{n-1} j i_{n+1} \cdots i_N} = \sum_{i_n=1}^{I_n} x_{i_1 i_2 \cdots i_N} u_{j i_n}$.

Tensor Decomposition: Given an input tensor, tensor factorization decomposes it into a smaller/core tensor multiplied by a matrix along each mode. For the case of a three-way tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I \times J \times K}$, we have $\boldsymbol{\mathcal{X}} \approx \boldsymbol{\mathcal{Z}} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}$. Matrices $\mathbf{A} \in \mathbb{R}^{I \times P}$, $\mathbf{B} \in \mathbb{R}^{J \times Q}$, and $\mathbf{C} \in \mathbb{K}^{K \times R}$ are called *factor matrices*, or *factors*, while tensor $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{I \times J \times K}$ is called the *core tensor*. In this process, each element of the tensor $\boldsymbol{\mathcal{X}}$ is the product of the corresponding factor matrix elements multiplied by a weight z_{pqr}, i.e., $x_{ijk} \approx \sum_{p=1}^{P} \sum_{q=1}^{Q} \sum_{r=1}^{R} z_{ijk} a_{ip} b_{jq} c_{kr}$.

CP Decomposition: CANDECOMP/PARAFAC [9] decomposition is often referred as CP. The CP decomposition of tensor $\boldsymbol{\mathcal{X}}$ could be expressed as $x_{ijk} \approx \sum_{r=1}^{R} z_r a_{ir} b_{jr} c_{kr}$. Let $[\mathbf{z}]$ denote a superdiagonal tensor, where $[\cdot]$ is the operation that transforms vector $\mathbf{z}$ to a superdiagonal tensor by setting tensor element $z_{k\ldots k} = \mathbf{z}_k$ and other elements as 0. Thus the CP decomposition of a three-way tensor can be written as $\boldsymbol{\mathcal{X}} \approx [\mathbf{z}] \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}$. Following Kolda [12], the CP model can be concisely expressed as $\boldsymbol{\mathcal{X}} \approx [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!] \equiv \sum_{r=1}^{R} \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r$.

3.2 Problem Formulation

Simultaneous Discovery of Common and Discriminative Mobility Patterns: The central problem in this paper can be formally expressed as follows. Given two nonnegative input tensors, $\boldsymbol{\mathcal{X}}_B \in \mathbb{R}^{I \times J \times K}$ and $\boldsymbol{\mathcal{X}}_A \in \mathbb{R}^{I \times J \times K}$ representing the sets of urban activities ***Before*** and ***After*** an exogenous shock, where each entry in the tensors represents the ***Location***, ***Time*** and ***Venue*** of each activity, we seek to obtain a nonnegative tensor factorization (NTF) to approximate both input tensors, as:

$$\boldsymbol{\mathcal{X}}_B \approx [\![\mathbf{U}_B^{(L)}, \mathbf{U}_B^{(T)}, \mathbf{U}_B^{(V)}]\!]$$

and

$$\boldsymbol{\mathcal{X}}_A \approx [\![\mathbf{U}_A^{(L)}, \mathbf{U}_A^{(T)}, \mathbf{U}_A^{(V)}]\!],$$

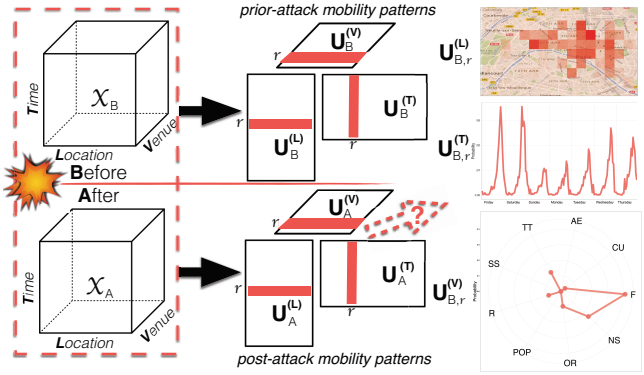

Figure 1: **Problem illustration of the proposed discriminant tensor analysis.** $\boldsymbol{\mathcal{X}}_B$ and $\boldsymbol{\mathcal{X}}_A$ represents the data tensor ***Before*** and ***After*** a terrorist attack event. Matrix $\mathbf{U}^{(L)}$, $\mathbf{U}^{(T)}$, and $\mathbf{U}^{(V)}$ represent the three factor matrices, ***Location***, ***Time***, and ***Venue***, respectively. The corresponding column in each factor matrix jointly represents a behavioral pattern. The goal is to find similar and discriminative behavioral patterns across the event.

respectively, where $U_q^{(L)} \in \mathbb{R}_+^{I \times R}$, $U_q^{(T)} \in \mathbb{R}_+^{I \times R}$, $U_q^{(V)} \in \mathbb{R}_+^{I \times R}$, $q \in \{A, B\}$, represent the factor matrices corresponding to *location*, *time*, and *venue*, respectively, for $\boldsymbol{\mathcal{X}}_B$ and $\boldsymbol{\mathcal{X}}_A$. Note that in this work we focus on three-mode tensors but the proposed tensor analysis framework can be used to deal with data with higher dimensions.

As shown in Fig.1, the corresponding column (red) of each factor matrix together define a mobility pattern that associates specific areas, time, and types of venues. Given our interest in disastrous events, such as terrorist attacks that inject an intense psychological instability in the targeted population, the mobility or behavior patterns of this population are likely to change after the event. The goal here is to discover the shared and discriminative components of the tensors describing urban activities before and after an event of interest. Each component here represents a coherent set of mobility pattern before and/or after the event of interest.

4. SOLUTIONS

In this section, we describe our solution for the problem described in Section 3.2.

4.1 Shared and Discriminative Subspace approach

To learn the shared and discriminative subspace, Liu et al. [14] proposed the Common and Discriminative subspace Non-negative Tensor Factorization (CDNTF) which takes a set of labeled tensor as its input and computes both their common and discriminative subspaces simultaneously as the output. Following their work, the objective of CDNTF can be re-written as the following simultaneous factorization of two input tensors:

$$\boldsymbol{\mathcal{X}}_B \approx [\![[\mathbf{U}_C^{(L)} | \mathbf{U}_{D:B}^{(L)}], \mathbf{U}_B^{(T)}, \mathbf{U}_B^{(V)}]\!]$$

and

$$\boldsymbol{\mathcal{X}}_A \approx [\![[\mathbf{U}_C^{(L)} | \mathbf{U}_{D:A}^{(L)}], \mathbf{U}_A^{(T)}, \mathbf{U}_A^{(V)}]\!],$$

where the columns of matrix $\mathbf{U}_B^{(L)}$ and $\mathbf{U}_A^{(L)}$ are segmented into two parts: $\mathbf{U}_C^{(L)}$ represents the common subspace, while $\mathbf{U}_{D:B}^{(L)}$ and $\mathbf{U}_{D:A}^{(L)}$ represents the discriminative components to each tensor $\boldsymbol{\mathcal{X}}_B$ and $\boldsymbol{\mathcal{X}}_A$. The above common and discriminative subspace discovery is the solution to the minimization of the following objective function:

$$J_0 = \frac{1}{n_B}\left\|\boldsymbol{\mathcal{X}}_B - [\![[\mathbf{U}_C^{(L)}|\mathbf{U}_{D:B}^{(L)}], \mathbf{U}_B^{(T)}, \mathbf{U}_B^{(V)}]\!]\right\|^2 + \frac{1}{n_A}\left\|\boldsymbol{\mathcal{X}}_A - [\![[\mathbf{U}_C^{(L)}|\mathbf{U}_{D:A}^{(L)}], \mathbf{U}_A^{(T)}, \mathbf{U}_A^{(V)}]\!]\right\|^2, \tag{1}$$

where $\mathbf{U}^{(L)}$, $\mathbf{U}^{(T)}$, and $\mathbf{U}^{(V)}$ are defined as above. n_A and n_B are the Frobenius norm of each tensor, and $\|\cdot\|^2$ stands for the Frobenious norm.

4.2 Regularized Shared and Discriminative Subspace Approach

Shared and discriminative subspace learning have also been explored in the context of nonnegative matrix factorization. In fact, CDNTF can be thought of as the extension of nonnegative shared subspace learning (JSNMF [7]) to higher dimensions. Under this framework, Gupta et al. [8] propose regularized nonnegative shared subspace learning that further imposes a mutual orthogonality constraint on the constituent subspace, which segregates the common patterns from those that are source-specific. In the context of discovering common and discriminative mobility patterns, we extend the framework to **R**egularized **J**oint **S**ubspace **N**onnegative **T**ensor **F**actorization (RJSNTF) and derive the following minimization problem:

$$J_1 = J_0 + \sum_{m\in\{L,V,T\}} J_{R1}(\mathbf{U}_C^{(m)}, \mathbf{U}_{D:B}^{(m)}, \mathbf{U}_{D:A}^{(m)}), \tag{2}$$

where $J_{R1}(\cdot)$ is a regularization function used to penalize the "similarity" between subspaces spanned in $\{\mathbf{U}_B^{(m)}\}$ and $\{\mathbf{U}_A^{(m)}\}$. Following [8], the mutually orthogonal constraints are defined as:

$$J_{R1}(\mathbf{U}_C^{(m)}, \mathbf{U}_{D:B}^{(m)}, \mathbf{U}_{D:A}^{(m)}) = \alpha\left\|{\mathbf{U}_C^{(m)}}^T\mathbf{U}_{D:B}^{(m)}\right\|^2 + \beta\left\|{\mathbf{U}_C^{(m)}}^T\mathbf{U}_{D:A}^{(m)}\right\|^2 + \gamma\left\|{\mathbf{U}_{D:A}^{(m)}}^T\mathbf{U}_{D:B}^{(m)}\right\|^2, \tag{3}$$

where α, β and γ are the regularization parameters. When $J_{R1}(\mathbf{U}_C^{(m)}, \mathbf{U}_{D:B}^{(m)}, \mathbf{U}_{D:A}^{(m)}) = 0$, the model is reduced to CDNTF.

RJSNTF enforces the shared components to be strictly identical, which is a hard constraint. This might result in distortion when factorizations the tensors. Kim et al. [11] have proposed the simultaneous discovery of common and discriminative topics via joint nonnegative matrix factorization where this constraint is relaxed by redefining the regularization term. The definition enforces the similar components to be even more similar while the discriminative parts even more so as well. Following the same idea and replacing $\mathbf{U}_C^{(m)}$ with $\mathbf{U}_{C:B}^{(m)}$ and $\mathbf{U}_{C:A}^{(m)}$ to represent the similar components of tensors $\boldsymbol{\mathcal{X}}_B$ and $\boldsymbol{\mathcal{X}}_A$ respectively we derive **S**imultaneous **D**iscovery of **C**ommon and **D**iscriminative **N**onnegative **T**ensor **F**actorization (SDCDNTF) as the following minimization function:

$$J_2 = J_0 + \sum_{m\in\{L,V,T\}} J_{R2}(\mathbf{U}_{C:B}^{(m)}, \mathbf{U}_{C:A}^{(m)}, \mathbf{U}_{D:B}^{(m)}, \mathbf{U}_{D:A}^{(m)}), \tag{4}$$

and

$$J_{R2}(\mathbf{U}_{C:B}^{(m)}, \mathbf{U}_{C:A}^{(m)}, \mathbf{U}_{D:B}^{(m)}, \mathbf{U}_{D:A}^{(m)}) = \alpha\left\|\mathbf{U}_{C:B}^{(m)} - \mathbf{U}_{C:A}^{(m)}\right\|^2 + \beta\left\|{\mathbf{U}_{D:B}^{(m)}}^T\mathbf{U}_{D:A}^{(m)})\right\|_{1,1}, \tag{5}$$

where $\|\cdot\|_{1,1}$ denotes the absolute sum of all the matrix entries.

4.3 Automatic Discovery of Discriminative Components

The above approaches fall under the same framework that splits the tensors' components into common and discriminative parts in advance, learning them with different regularization. These approaches require the number of shared (or distinct) components to be determined beforehand, which is difficult in practice. In this paper, we propose a novel factorization method, which we term `PairFac`, that does not require a manual input for the distinction between these two parts. In a nutshell, we assign a weight to each component that reflects the *discriminative coefficient* or *score* of the corresponding component.

To do so, we introduce two auxiliary data tensors $\boldsymbol{\mathcal{Z}}_B$ and $\boldsymbol{\mathcal{Z}}_A$ that represent the aggregated unique patterns found in each tensor respectively. We first define the following function to compute these auxiliary tensors.

Definition 4.1. Given a data tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I\times J\times K}$, $G(\boldsymbol{\mathcal{X}})$ is a *clamping* function that finds a tensor $\boldsymbol{\mathcal{Z}} \in \mathbb{R}^{I\times J\times K}$ that restricts the entries in $\boldsymbol{\mathcal{X}}$ to a given range, so that we have $\boldsymbol{\mathcal{Z}} = G(\boldsymbol{\mathcal{X}})$, where $G(\boldsymbol{\mathcal{X}})$ is defined as:

$$G(\boldsymbol{\mathcal{X}}) = \begin{cases} \boldsymbol{\mathcal{X}}_{ijk}, \text{if } \boldsymbol{\mathcal{X}}_{ijk} > \epsilon, \\ 0, \text{otherwise}, \end{cases} \tag{6}$$

and ϵ is a constant that defines the minimum entry in the tensor $\boldsymbol{\mathcal{Z}}$. ϵ can be empirically chosen to control the sparseness of the auxiliary tensors (we use $\epsilon = 0$ in this work). Note that the clamping function $G(\cdot)$ can also work with vectors and matrices. Then we compute $\boldsymbol{\mathcal{Z}}_B$ that captures the unique variance in $\boldsymbol{\mathcal{X}}_B$ from $\boldsymbol{\mathcal{X}}_A$ and $\boldsymbol{\mathcal{Z}}_A$ that captures the unique variance in $\boldsymbol{\mathcal{X}}_A$ from $\boldsymbol{\mathcal{X}}_B$ as:

$$\boldsymbol{\mathcal{Z}}_A = G(\boldsymbol{\mathcal{X}}_A - \boldsymbol{\mathcal{X}}_B), \tag{7}$$

and

$$\boldsymbol{\mathcal{Z}}_B = G(\boldsymbol{\mathcal{X}}_B - \boldsymbol{\mathcal{X}}_A). \tag{8}$$

We also use the weight vectors $\mathbf{w}_B \in \mathbb{R}_+^R$ and $\mathbf{w}_A \in \mathbb{R}_+^R$ to capture the discriminative coefficient of each corresponding component. Our intuition is that while factorizing the original tensors into its latent patterns (mobility in our case), we want to find a discriminative score for each pattern that corresponds to its unique contribution in each tensor. With the notations presented, we formally derive the minimization problem as:

$$J_3 = J_0' + J_{R3}, \tag{9}$$

where J_0' differs from J_0 in that it does not require the manual split of common and discriminative parts in the factor matrix, and J_{R3} is a function to factorize the auxiliary tensors, defined as:

$$J_{R3} = \alpha \left\| \mathcal{Z}_B - [\![\mathbf{w}_B; \mathbf{U}_B^{(L)}, \mathbf{U}_B^{(T)}, \mathbf{U}_B^{(V)}]\!] \right\|^2 + \beta \left\| \mathcal{Z}_A - [\![\mathbf{w}_A; \mathbf{U}_A^{(L)}, \mathbf{U}_A^{(T)}, \mathbf{U}_A^{(V)}]\!] \right\|^2. \quad (10)$$

To solve Eq. 9 we use the block coordinate descent method. At each iteration, we only update one factor matrix while fixing the others. Therefore, a factor matrix $\mathbf{U}_q^{(m)}$ where $q \in \{A, B\}$ and $m \in \{L, V, T\}$ in Eq. 9 can be obtained through the following minimization problem:

$$\min \frac{1}{n_q} \left\| \boldsymbol{\mathcal{X}}_{q(m)} - \mathbf{U}_q^{(m)} (\mathbf{U}_q^{(m'')} \odot \mathbf{U}_q^{(m')})^T \right\|^2 + \alpha \left\| \boldsymbol{\mathcal{Z}}_{q(m)} - \mathbf{U}_q^{(m)} \Lambda_{\mathbf{w}_q} (\mathbf{U}_q^{(m'')} \odot \mathbf{U}_q^{(m')})^T \right\|^2, \quad (11)$$

where $\odot$ denotes the Khatri-Rao product,$\Lambda_{\mathbf{w}_q}$ is a diagonal matrix with $\mathbf{w}_q$ as its diagonal entries while m' and m'' are used to index factor matrices other than $\mathbf{U}_q^{(m)}$. Similarly, we update each factor matrix alternatively until convergence. The optimization of the factor matrices is a nonnegative least squares problem in the context of NMF and several methods have been extensively studied, including the multiplicative updating (MU) method, the hierarchical alternating least squares method, the active-set methods, and the alternating proximal gradient method (APG) [23]. While all of these algorithms do not require searching for the step size at each iteration, APG-based methods have shown the superiority in the convergence speed and quality. In fact, APG can be regarded as a special case of the block coordinate descent method (BCD) with two blocks [24]. In this paper, we adopt the same framework as in [24] to solve our problem.

The BCD method solves the following generic optimization problem:

$$\min_x F(x_1, \ldots, x_n) = f(x_1, \ldots, x_n), \quad (12)$$

where variable x is partitioned into n blocks x_1,..., x_n and $f(\cdot)$ is differentiable and convex for each i while keeping the others fixed. Let $\bigtriangledown f_i(x_i)$ is the block-partial gradient of f at x_i. Then, for any x_i^1 and x_i^2, the following property holds:

$$\left\| \bigtriangledown f_i(x_i^1) - \bigtriangledown f_i(x_i^2) \right\|_F \leqslant \mathcal{L} \left\| x_i^1 - x_i^2 \right\|_F, \quad (13)$$

for a suitably chosen $\mathcal{L}$ as the Lipschitz constant. Applying the acceleration technique [23, 5], we keep two sequences x_i^k and z^k alternatively updated during each iteration k:

$$x_i^k = \arg\min_{x_i} \left(\langle \bigtriangledown_i f(x_i^{k-1}), x_i - x_i^{k-1} \rangle + \frac{\mathcal{L}_i^k}{2} \left\| x_i - x_i^{k-1} \right\|_F \right), \quad (14)$$

and

$$z^k = x_i^k + \frac{\alpha_k - 1}{\alpha_{k+1}} (x_i^k - x_i^{k-1}), \quad (15)$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product and $\frac{\alpha_k - 1}{\alpha_{k+1}}$ is the extrapolation weight. The update step size α_k is chosen the same as in [5]:

$$\alpha_{k+1} = \frac{1 + \sqrt{4\alpha_k^2 + 1}}{2}. \quad (16)$$

Eq. 14 has the following closed form for nonnegative factorization:

Algorithm 1: `PairFac` algorithm for discovering the shared and discriminative subspace from tensor pairs.

Input : original tensors $\boldsymbol{\mathcal{X}}_B$ and $\boldsymbol{\mathcal{X}}_A$, and $\mathbf{R}$.
Output: $\{w_q\}$, $\{U_q^{(m)}\}$ for $q \in \{A, B\}$ and $m \in \{L, V, T\}$

Compute $\boldsymbol{\mathcal{Z}}_B$ and $\boldsymbol{\mathcal{Z}}_A$ by Eq. 7 and Eq. 8;
Randomly initialize $\mathbf{U}_{q,0}^{(m)}$, $\forall q$ and $\forall m$;
Set $z_0^{\mathbf{U}_q^{(m)}} = \mathbf{U}_{q,0}^{(m)}$, $\forall q$ and $\forall m$;
Set $z_0^{\mathbf{w}_q} = \mathbf{w}_{q,0} = [\frac{\mathbf{1}}{\mathbf{R}}]$, $\forall q$;
Set $\alpha_1 = 1$ and $k = 1$;
while *not converged* **do**
  $\mathbf{U}_{q,k}^{(m)} = G\left(z_{k-1}^{\mathbf{U}_q^{(m)}} - \frac{1}{\mathcal{L}_k^{\mathbf{U}_q^{(m)}}} \frac{\partial f}{\partial \mathbf{U}_q^{(m)}} \right)$, $\forall q$ and $\forall m$;
  $\mathbf{w}_{q,k} = G\left(z_{k-1}^{\mathbf{w}_q} - \frac{1}{\mathcal{L}_k^{\mathbf{w}_q}} \frac{\partial f}{\partial \mathbf{w}_q} \right)$, $\forall q$;
  $\alpha_{k+1} = \frac{1 + \sqrt{4\alpha_k^2 + 1}}{2}$;
  $z_k^{\mathbf{U}_q^{(m)}} = \mathbf{U}_{q,k}^{(m)} + \frac{\alpha_k - 1}{\alpha_{k+1}} (\mathbf{U}_{q,k}^{(m)} - \mathbf{U}_{q,k-1}^{(m)})$, $\forall q$ and $\forall m$;
  $z_k^{\mathbf{w}_q} = \mathbf{w}_{q,k} + \frac{\alpha_k - 1}{\alpha_{k+1}} (\mathbf{w}_{q,k} - \mathbf{w}_{q,k-1})$, $\forall q$;
  $k = k + 1$;
end

$$x_i^k = G\left(z_i^{k-1} - \frac{1}{\mathcal{L}_i^k} \bigtriangledown f_i(x_i^{k-1}) \right). \quad (17)$$

In this paper, we extend the above for `PairFac`. We can think of each block x_i as each factor matrix $\mathbf{U}_q^{(m)}$. Then, the gradient of Eq. 11 with respect to $\mathbf{U}_q^{(m)}$ can be computed as:

$$\frac{\partial f}{\partial \mathbf{U}_q^{(m)}} = \frac{1}{n_q} \mathbf{U}_q^{(m)} \mathbf{F}_q^{(m)T} \mathbf{F}_q^{(m)} - \boldsymbol{\mathcal{X}}_{q(m)} \mathbf{F}_q^{(m)} + \alpha (\mathbf{w}_q \mathbf{U}_q^{(m)} \mathbf{F}_q^{(m)T} \mathbf{F}_q^{(m)} - \boldsymbol{\mathcal{Z}}_{q(m)} \mathbf{F}_q^{(m)}), \quad (18)$$

where

$$\mathbf{F}_q^{(m)} = \mathbf{U}_q^{(m')} \odot \mathbf{U}_q^{(m'')}, \quad (19)$$

and

$$\mathcal{L}^{\mathbf{U}_q^{(m)}} = \left\| \mathbf{F}_q^{(m)T} \mathbf{F}_q^{(m)} \right\|^2 \quad (20)$$

is the Lipschitz constant with respect to q and m. The gradient of Eq. 11 with respect to $\mathbf{w}_q$ can be written as:

$$\frac{\partial f}{\partial \mathbf{w}_q} = \mathbf{w}_q \mathbf{F}_{w_q}{}^T \mathbf{F}_{w_q} - \boldsymbol{\mathcal{Z}}_{q(m)} \mathbf{F}_{w_q}, \quad (21)$$

where

$$\mathbf{F}^{\mathbf{w}_q} = \mathbf{U}_q^{(m)} \odot \mathbf{U}_q^{(m')} \odot \mathbf{U}_q^{(m'')}, \quad (22)$$

and

$$\mathcal{L}^{\mathbf{w}_q} = \left\| \mathbf{F}^{\mathbf{w}_q T} \mathbf{F}^{\mathbf{w}_q} \right\|^2 \quad (23)$$

is the Lipschitz constant with respect to q. Our complete `PairFac` algorithm that uses the above updating rules for solving Eq. 9 is presented in Algorithm 1.

5. EVALUATION

In this section, we use a synthetic dataset to evaluate the performance of our proposed method. As discussed in section 4, there are three existing models that we adopt for

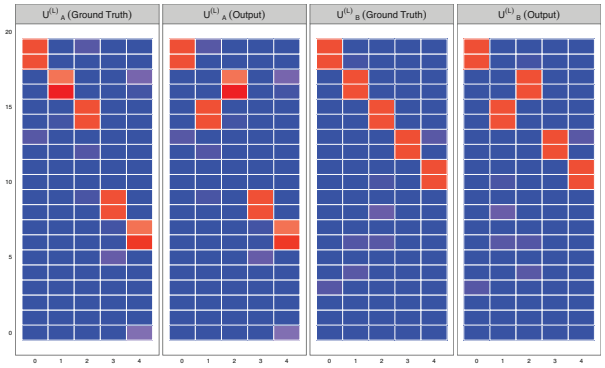

Figure 2: **Illustration of the output from our approach.** We reorder the components of each output factor matrix by its associated weight in increasing order from left to right. The weight vector $\mathbf{w}_A = [.001, .023, .005, .485, .485]$ and the weight vector $\mathbf{w}_B = [.001, .022, .004, .487, .487]$.

comparisons, including CDNTF[14], and our extension of RSJNMF[8] to RSJNTF, and our extension of SDCDNMF[11] to SDCDNTF.

5.1 Synthetic Data Setup

We generated two three-way tensors $\boldsymbol{\mathcal{X}}_B \in \mathbb{R}^{I\times J\times K}$ and $\boldsymbol{\mathcal{X}}_A \in \mathbb{R}^{I\times J\times K}$ according to the equation $\boldsymbol{\mathcal{X}}_B = \sum_{r=1}^{R} \mathbf{U}^{(L)}_{B,r} \circ \mathbf{U}^{(T)}_{B,r} \circ \mathbf{U}^{(V)}_{B,r}$ and $\boldsymbol{\mathcal{X}}_A = \sum_{r=1}^{R} \mathbf{U}^{(L)}_{A,r} \circ \mathbf{U}^{(T)}_{A,r} \circ \mathbf{U}^{(V)}_{A,r}$, where $\boldsymbol{\mathcal{X}}_B$ and $\boldsymbol{\mathcal{X}}_A$ shared the first $\mathbf{K}$ components in the first factor matrix and have exactly the same columns in the second and third factor matrices. Our generation rules of the synthetic dataset follow the idea of [11] with an additional set of parameters so that the process can be adapted for the synthetic dataset with more flexibility in the dimensional settings. The shared parts in the first factor matrix are generated as:

$$\mathbf{U}^{(L)}_{C,r} = \begin{cases} 1, s\times r \leqslant r < s\times(r+1), \\ 0, \text{otherwise}, \end{cases}$$

, where $s = \frac{I}{\mathbf{R}+(\mathbf{R}-\mathbf{K})}$. We generate the discriminative parts in the first factor matrix as:

$$\mathbf{U}^{(L)}_{D:B,r} = \begin{cases} 1, s\times\mathbf{K} + s\times r \leqslant r < s\times\mathbf{K} + s\times(r+1), \\ 0, \text{otherwise}, \end{cases}$$

and

$$\mathbf{U}^{(L)}_{D:A,r} = \begin{cases} 1, s\times\mathbf{R} + s\times r \leqslant r < s\times\mathbf{R} + s\times(r+1), \\ 0, \text{otherwise}. \end{cases}$$

In addition, each row of $\mathbf{U}^{(T)}$ and $\mathbf{U}^{(V)}$ is set to be a unit vector with only one non-zero entry at a randomly selected dimension. We further add sparse Gaussian noise $\mathcal{N}(0,\sigma^2)$ with different amounts of variance to 20% of the entries in $\mathbf{U}^{(L)}_B$ and $\mathbf{U}^{(L)}_A$.

5.2 Results

5.2.1 *Algorithm Output Illustration*

We first provide the illustration of the output from our approach with the synthetic dataset generated by setting $I = J = K = 20$, $\mathbf{R} = 5$, $\mathbf{K} = 3$, $\sigma^2 = 0.1$ and $\alpha = \beta = 1e-4$. Fig. 2 shows an example of the factor matrices obtained from our method in comparison with the ground truth factor matrices. Each column of the output factor matrices is associated with a discriminative score, where a higher score represents a greater discriminative power of this component in comparison with the corresponding factor matrix in the second tensor. We observe that our method nicely segments each output factor into two parts based on the learned weights. The weights of the common components are almost zero while the discriminative components equally shared the discriminative power.

5.2.2 *Performance Evaluation:*

We include three baselines and one modification of our method for comparative studies:

- CDNTF [14] takes an input $\mathbf{K}$ and splits the factor matrix into $\mathbf{K}$ common components and $(\mathbf{R}-\mathbf{K})$ discriminative components by solving Eq. 1 with multiplicative updating rules.
- RSJNTF is our tensor extension of RSJNMF[8]. It also requires the input $\mathbf{K}$ and shares the similar framework with CDNTF with additional mutually orthogonal constraints on the common and discriminative components. We develop multiplicative updating rules to solve Eq. 2.
- SDCDNTF is our tensor extension of SDCDNMF [11], which also requires the input $\mathbf{K}$. It is under the same framework with RSJNTF with relaxed constraints on the shared components. We extend the block coordinate descent framework to SDCDNTF to solve Eq. 4.
- `PairFac` does not require the specification of $\mathbf{K}$. Instead, it learns two weight vectors that represent the discriminative scores for each of its components. We use a matrix-wise alternating proximal descent framework to solve Eq. 9.
- `PairFacC` solves the same optimization problem as in Eq. 9, however it adopts a column-wise alternating proximal descent framework.

Performance Metrics To quantitatively evaluate the performance of our proposed approach in comparison with existing literature, we use three measures, namely, (a) the relative reconstruction error, (b) the quality of the recovered discriminative components and (c) the quality of the recovered the recovered common components. To measure the quality of the reconstruction, we compute the relative reconstruction error as:

$$\frac{1}{2}\left(\frac{\left\|\boldsymbol{\mathcal{X}}_B - [\![\mathbf{U}^{(L)}_B, \mathbf{U}^{(T)}_B, \mathbf{U}^{(V)}_B]\!]\right\|^2}{||\boldsymbol{\mathcal{X}}_B||^2} + \frac{\left\|\boldsymbol{\mathcal{X}}_A - [\![\mathbf{U}^{(L)}_A, \mathbf{U}^{(T)}_A, \mathbf{U}^{(V)}_A]\!]\right\|^2}{||\boldsymbol{\mathcal{X}}_A||^2}\right).$$

The quality of the recovered discriminative part of factor matrix is computed as the similarity between the output factor matrix and the ground truth factor matrix, which follows $sim_D(\mathbf{U},\bar{\mathbf{U}}) = \frac{1}{\mathbf{R}-\mathbf{K}}\sum_{r>\mathbf{K}}^{\mathbf{R}} \cos(\mathbf{U}_r, \bar{\mathbf{U}}_r) = \frac{\mathbf{U}_r\cdot\bar{\mathbf{U}}_r}{\|\mathbf{U}_r\|\|\bar{\mathbf{U}}_r\|}$, where $\mathbf{U}_r$ is the r-th discriminative component in the ground truth and $\bar{\mathbf{U}}_r$ is the output r-th discriminative component. Because there is an ambiguity in the column orderings [1], we try out all possible permutations of $\mathbf{R}-\mathbf{K}$ components and we compute the maximum similarity. Similarly, we compute the maximum similarity score on the common components as: $sim_C(\mathbf{U},\bar{\mathbf{U}}) = \frac{1}{\mathbf{R}}\sum_{r\leqslant\mathbf{K}}^{\mathbf{R}} \cos(\mathbf{U}_r, \bar{\mathbf{U}}_r) = \frac{\mathbf{U}_r\cdot\bar{\mathbf{U}}_r}{\|\mathbf{U}_r\|\|\bar{\mathbf{U}}_r\|}$.

Experiment Setup: Following the setup introduced in section 5.1, we generate another synthetic dataset by setting $I = 100$, $J = 10$, $K = 20$, $\sigma^2 = 0.5$, $\mathbf{R} = 10$, and $\mathbf{K} = 5$. For SDCDNTF, we experiment with α and β in $[10^{-5}, 10^{-4}, 10^{-3}, .01, .1, .3]$. For RSJNTF, following [8], we

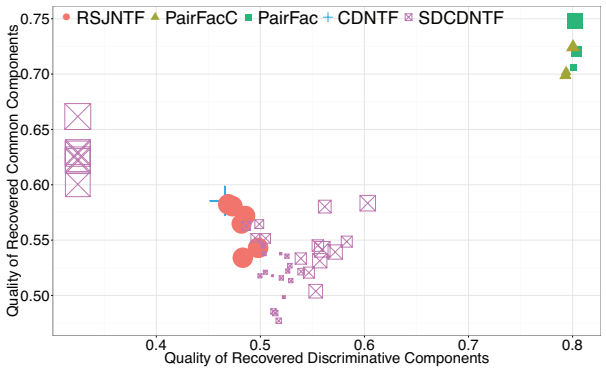

Figure 3: **Comparison of our approach with existing methods.** Each point represents the average score of 30 runs. The size of points represents the reconstruction error.

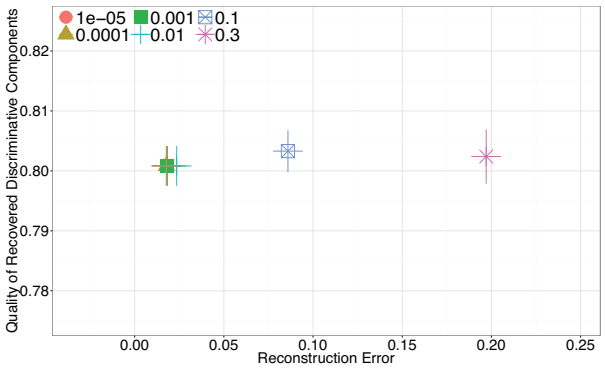

Figure 4: **The effect of α on the reconstruction error and the recover quality.** As α increases, the reconstruction error increases as well, while the recover quality on the discriminative components increases up to a certain point after which it deteriorates.

set a super parameter α from the same range. Finally, we set $\alpha = \beta$ from the same range for `PairFac` and `PairFacC`. We plot the average reconstruction error versus average similarity score on the discriminative components as well as on the common components from 30 runs for each set of parameters.

Fig. 3 presents the comparison of the various methods. Our approach has comparable reconstruction quality with that of SDCDNTF. However, it has superior performance in terms of both the quality of recovered discriminative and common components.

5.2.3 *Parameter Sensitivity*

In our approach, parameters α and β control the weight placed on identifying the discriminative components, with the trade-off of potentially jeopardizing the reconstruction quality.

We vary the parameter α in `PairFac` as defined in Section 5.2.2. As we increase α, we place more weights on learning the discriminative components. Fig. 4 shows that with the increase of α, although the quality of the recovered discriminative increases, the reconstruction error increases as well. As `PairFac` learns the discriminative weights we label the components with their discriminative power so that we can distinguish the common components and the discriminative components. During this process, we need to identify a cut-off point for the (ranked) weights. The components that have discriminative power higher than this cutoff would be regarded as unique patterns to each tensor data.

There can be different approaches to segment a one-dimensional vector, most of which rely on how well the two sets of data points can be separated. Fig. 5 (a) shows the distribution of the weight vectors in different evaluation environment. As α becomes smaller, there is a more clear separation in the bimodal distribution. To check the degree of separation, we computed *Ashman's D*, which is commonly used to quantify the separation from a mixture of two normal distributions as [3]:

$$D = 2^{\frac{1}{2}} \frac{|\mu_1 - \mu_2|}{\sqrt{\sigma_1^2 + \sigma_2^2}}.$$

From Fig. 5 (b) we can observe that as α becomes larger, D initially increases and reaches its peak before it starts decreasing. This is likely because for a small α the weight term in the regularization does not make a substantial impact on learning the unique patterns. However, when α becomes very large, the two distributions are more difficult to separate.

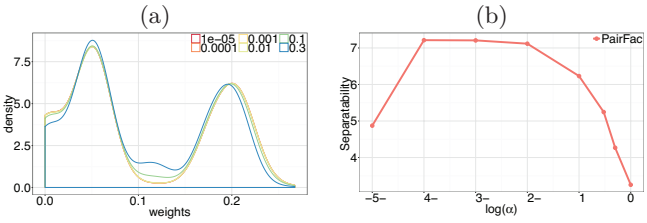

Figure 5: **(a) Weights distribution (b) α vs D (Separability).** (a) shows that as α chooses a smaller value, the distribution of the weights are more close to a bimodal distribution; (b) shows that as α increases, the bimodal distribution are more separable, while the separability decreases after a certain point.

6. CASE STUDIES

In this section, we illustrate the application of our method in a real case study. In particular, we use `PairFac` to analyze the effects of the Paris terrorist attacks in the surrounding urban space. We collect Twitter check-ins and traffic sensor data in Paris and apply our approach to the tensors constructed based on the two datasets respectively.

6.1 Dataset

Table 1 summarizes the data sources we used for our case study. The first dataset is the geo-tagged tweets from Paris collected through Twitter API between the period of Oct 6th, 2015 and Nov 20, 2015. The region is defined by a rectangle boundary [1] that covers the Paris area. 75,982 geo-located tweets were extracted during the period covered. The second dataset includes approximately 2.5 million records of traffic sensor data [20]. It provides the hourly occupancy rate of 2,889 road segments in the area of Paris and covers the same period as above. Our third dataset is from Foursquare collected by Yang et al. [25] and it contains 86,033 check-ins from 15,375 POIs in the area of Paris area between April 2012 and September 2013.

[1]N 48° 54′ 32.6118″, E 2° 24′ 33.7104″, N 48° 48′ 56.361″, E 2° 14′ 36.7794″.

Data Source	Type of Data	Dimensions extracted	Volume of raw data extracted
Twitter	Geo-tagged Tweets	Location, Time	75,982 tweets
Service des Déplacements	Traffic Sensor databases	Location, Time	2,492,640 hourly occupancy rate from 2,885 road sensors
Foursquare	Checkins and POI database	Location, Time, Activity	86,033 checkins with 15,375 POI information

Table 2: Data fusion from heterogeneous data sources in Paris.

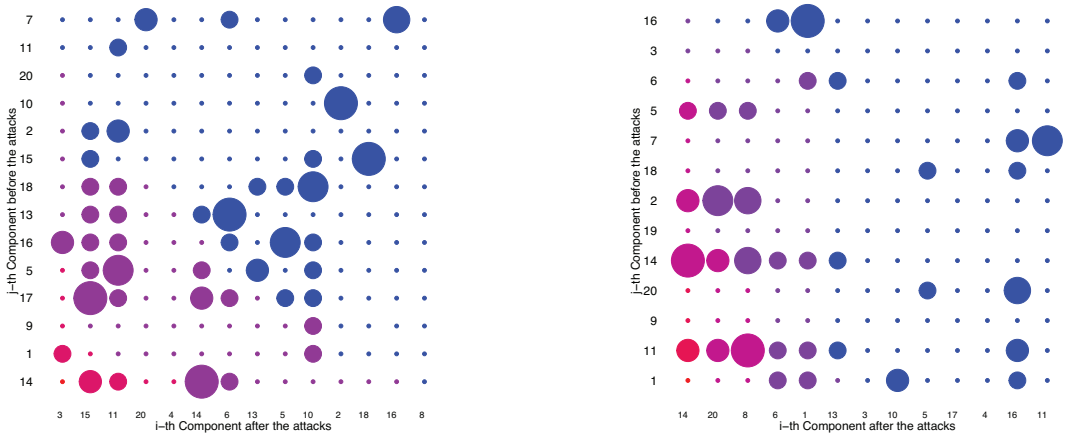

Figure 6: **Pair-wise Cosine similarity of the components from traffic sensor data (left) and from Twitter check-ins (right).** We re-order the components from each tensor by its associated weight in an increasing order so that the bottom-left places the component with the highest discriminative weight. The size of the point represents the cosine similarity between each pair of components. The color is filled as by the joint discriminative score as computed by $\mathbf{w}_{Ai} * \mathbf{w}_{Bj}$, where points with high joint discriminative scores are colored in red and ones with low scores are colored in blue.

6.2 Experiment Setup

We construct three-mode tensors, where the three dimensions are location, time, and venue type, respectively. For the location dimension, we overlay the city of Paris with a grid including 646 square cells, where each cell's side is 1000 meters. For the temporal dimension, we segment a week into $24 \times 7 = 168$ hourly intervals. Finally, for the venue dimension, we extract the nine primary categories in the Foursquare venue hierarchy that includes *Professional & Other Places* (POP), *Travel & Transport* (TT), *Food* (F), *Outdoors & Recreation* (OR), *Nightlife Spot* (NS), *Shop & Service* (SS), *Residence* (R), *Arts & Entertainment* (AE), and *College & University* (CU). For the data tensor of geo-tagged Tweets, we first construct a matrix LT, where LT_{ij} is the number of geo-located tweets that fall in i-th grid cell at the j-th hour in the week. Similarly, we construct the LT matrix based on the traffic sensor data, where LT_{ij} is the average occupancy rate in i-th grid cell at the j-th hour in the week. Then, we construct a matrix FTV, where FTV_{ijk} is the probability of Foursquare check-ins in k-th venue category that falls in i-th grid cell at the j-th hour in the week. Thus, for each cell at a given hour in the week, we know from the matrix FTV the probability distribution of activities over the nine categories. Finally, the entries in the data tensor are computed as:

$$\mathcal{X}_{ijk} = \frac{LT_{ij} \times FTV_{ijk}}{\sum_{ijk} \mathcal{X}_{ijk}}, \tag{24}$$

for both $\boldsymbol{\mathcal{X}}_B$ and $\boldsymbol{\mathcal{X}}_A$. $\boldsymbol{\mathcal{X}}_B$ contains the normalized aggregated values over four weeks between Oct 16th, 2015 (Friday) and Nov 12th, 2015, and $\boldsymbol{\mathcal{X}}_A$ is constructed based on the normalized values in the following week, between Nov 13th, 2015 (Friday) and Nov 20th, 2015.

In our experiments we set $\alpha = \beta = 10^{-5}$ and include spatial similarity regularization to enforce the contagious areas to share the same components. Finally $\mathbf{R} = 20$ for both datasets.

6.3 Results

The output of `PairFac` is the mobility patterns as components learned from the tensor factorization along with their discriminative weights. To interpret these patterns in a comparative analysis, we first align similar components in two periods together. The similarity between j-th component *before* the attacks and i-th component *after* the attacks is computed as:

$$cosine(\mathbf{U}^{(L)}_{B_j}, \mathbf{U}^{(L)}_{A_i}) \cdot cosine(\mathbf{U}^{(T)}_{B_j}, \mathbf{U}^{(T)}_{A_i}) \cdot cosine(\mathbf{U}^{(V)}_{B_j}, \mathbf{U}^{(V)}_{A_i}).$$

For each component pair of (j,i), we also compute the joint discriminative score as $\mathbf{w}_{Ai} \cdot \mathbf{w}_{Bj}$.

Fig. 6 lists all the pair-wise components from each dataset. We re-order the components in each *axis* so that the one with the highest discriminative score falls onto the bottom-left in each figure. The size of each point represents the similarity score of the components, and the color represents the joint discriminative score. As we observe from Fig. 6, the similar yet discriminative components nicely locate in the bottom-left region, while the similar ones - but with low discriminative scores - are pinpointed on the top-right corner in each figure. Due space limitations, we only show a subset of the top-ranked common and discriminative components from each dataset.

Nightlife: Fig. 7 shows the component with the largest joint discriminative score among all. It corresponds to the 14-th component before the attacks and 14-th component after the attacks (the point at the position of (14,14) in the left part of Fig. 6). This turns out to be the one that represents the night life activities in the neighborhoods near to the attack sites. We can see from Fig. 7 that before the attacks more traffic is observed during the late night on Fridays and

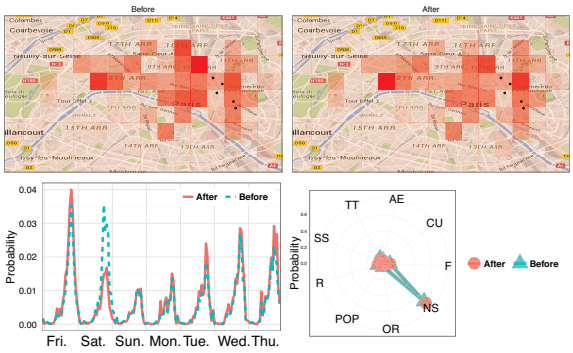

Figure 7: **Nightlife Spot pattern from the traffic sensor data (NS).** $14-th$ component before the attacks and $14-th$ component after the attacks. Two maps show the probability distribution of traffic in different areas of Paris before (left) and after (right), where dark red stands for higher probability. The bottom-left figure shows the distribution of traffic over the week (24×7), where blue dotted line represents the distribution before the attacks and solid red line for the one after. The radar chart on the bottom-right shows the traffic in areas with different distributions over the venue categories (defined in section 6.2) from Foursquare, where blue dots represents the distribution before the attacks and red dots as the one after.

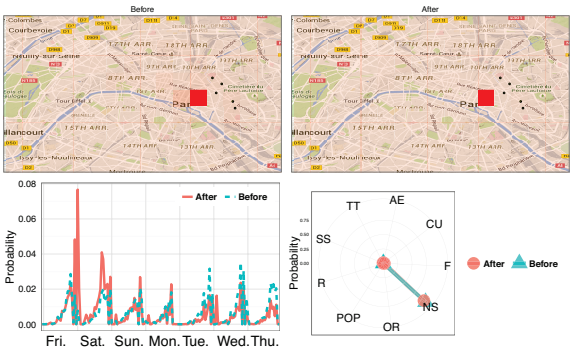

Figure 8: **Nightlife Spot (NS) pattern from Twitter check-ins.** $14-th$ component before the attacks and $14-th$ component after the attacks. We observe that on the following day of the attacks (Saturday), there is a spike of activities in the areas that are populated with nightlife venues, while the activities in this region decrease to a much smaller volume afterwards.

Saturdays. The volume of traffic decreased on the Saturday and Sunday after the attacks. This might be due to the restrictions of movements on the vehicles in some parts of the city. The Twitter check-ins (as pointed at (14,14) location in the right part of Fig. 6), are depicted in Fig. 8 and show a larger volume of activities on Friday midnight. In retrospect, this is expected since this is the time that the attacks happened and hence, there might be a spike of Twitter activities. What is interesting and contrasting to the pattern from the traffic data is that we observe a much higher probability of activities on the second day of the attacks, while much less over the next week. This combined could explain the situation of the streets in the regions close to the attack sites; they are fueled with fewer than normal vehicles, but angered citizens are in the area to commemorate the victims from the attacks or to just *protest* against violence towards the civilians expressing their opinions on Twitter.

Shop & Services: The pair of components from the traffic data with the least discriminative score are the patterns for *shops*. As shown in Fig. 9 (corresponds to point at the position of (18,15) in Fig. 6 on the left), while there is a slight decrease of the traffic on Saturday and Sunday, the patterns before and after are almost the same otherwise. This might be because the primary driving region of this pattern falls in the southeast part of the city, which is relatively further away from the attack sites.

Transportation: The mobility pattern that represents the way people move around the city through public transportation is presented in Fig. 10 (point (11, 7) in Fig. 6 on the right). We observe a spike of activities on Friday from Twitter data. This is possibly due to the international friendly match between France and Germany in Stade de France in North Paris, which is accessible via public transport. The distribution of the locations for this component is more concentrated in areas where Paris Metro stations are located. This is also reflected in the activities distribution (bottom-right in Fig. 10), where the activity is more focused on Travel

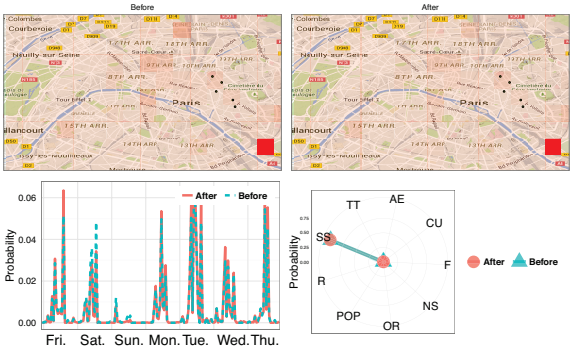

Figure 9: **Shop & Services (SS) pattern from traffic data.** The pattern is almost identical before and after the attacks.

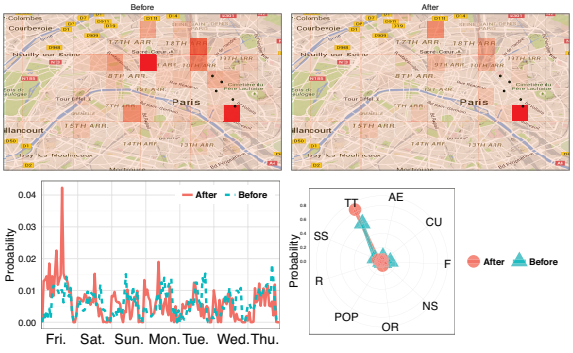

Figure 10: **Pattern of Travel & Transportation (TT) from Twitter data.** It stays relatively the same while being more concentrated towards the use of public transportation.

& Transportation venue types. This observation might be explained by the fact that public transportation stations function as hubs of the city to commute to different areas.

7. CONCLUSION

In this work, we provide a novel method to quantitatively evaluate the impact of a disastrous outbreak on the human mobility in the city. We propose a novel analytic approach

that aims to automatically discover the impact of an exogenous event on multiple aspects of human activities in the urban environment. We apply our model to traffic sensor data and Twitter check-ins data in Paris surrounding the period of Terrorist attacks in Nov 2015. We find that the mobility pattern that represents the city's nightlife is the one with the most changes across the time while the pattern for public transportation stays relatively stable.

Finally, we acknowledge the limitations of this study. Currently, the discriminative score captures both the changes in the volume of a certain mobility pattern as well as in the distribution of this mobility pattern in either location, time, or venue. This helps to ensure that the factorization can reveal the patterns that are mostly close to the ones underline. However, it suffers the problem of being less interpretable. In our future work, we plan to add regularizations to the current framework so that the discovery of the changes can be more targeted in particular dimensions. Besides, in this work, we only consider the immediate changes in one week in the mobility patterns, using the terrorist attacks in Paris as a case study. In the future, we plan to explore the possibility of characterizing a risk scenario based on the results of various case studies. We hope through this, the analysis could shed predictability insights for what the city might go through in the wake of disastrous events.

Acknowledgement

This work is part of the research supported from NSF #1423697, #1634944, the CRDF at the University of Pittsburgh, and the ARO Young Investigator Award W911NF-15-1-0599 (67192-NS-YIP). Any opinions, findings, and conclusions or recommendations expressed in this material do not necessarily reflect the views of the funding sources.

References

[1] Evrim Acar et al. "Scalable Tensor Factorizations with Missing Data." In: SIAM 2011.

[2] Harshavardhan Achrekar et al. "Predicting flu trends using twitter data". In: *INFOCOM WKSHPS 2011*.

[3] Keith M Ashman, Christina M Bird, and Steven E Zepf. "Detecting bimodality in astronomical datasets". In: *arXiv preprint astro-ph/9408030* (1994).

[4] James P Bagrow, Dashun Wang, and Albert-Laszlo Barabasi. "Collective response of human populations to large-scale emergencies". In: *PloS one* (2011).

[5] Amir Beck and Marc Teboulle. "A fast iterative shrinkage-thresholding algorithm for linear inverse problems". In: *SIAM journal on imaging sciences* (2009).

[6] Nicholas Diakopoulos, Mor Naaman, and Funda Kivran-Swaine. "Diamonds in the rough: Social media visual analytics for journalistic inquiry". In: *VAST 2010*.

[7] Sunil Kumar Gupta et al. "Nonnegative shared subspace learning and its application to social media retrieval". In: *ACM SIGKDD 2010*.

[8] Sunil Kumar Gupta et al. "Regularized nonnegative shared subspace learning". In: *Data mining and knowledge discovery* (2013).

[9] R.A. Harshman. "Foundations of the PARAFAC procedure: Models and conditions for an" explanatory" multimodal factor analysis". In: (1970).

[10] Andreas Kaltenbrunner et al. "Urban cycles and mobility patterns: Exploring and predicting trends in a bicycle-based public transport system". In: *Pervasive and Mobile Computing* (2010).

[11] Hannah Kim et al. "Simultaneous Discovery of Common and Discriminative Topics via Joint Nonnegative Matrix Factorization". In: *ACM SIGKDD 2015*.

[12] Tamara Gibson Kolda. *Multilinear operators for higher-order decompositions*.

[13] Yu-Ru Lin and Drew Margolin. "The ripple of fear, sympathy and solidarity during the Boston bombings". In: *EPJ Data Science* (2014).

[14] Wei Liu et al. "Mining labelled tensors by discovering both their common and discriminative subspaces". In: SIAM 2013.

[15] Michael Mathioudakis and Nick Koudas. "Twittermonitor: trend detection over the twitter stream". In: *ACM SIGMOD 2010*.

[16] Sharad Mehrotra et al. "Technological Challenges in Emergency Response". In: *IEEE Intelligent Systems* (2013).

[17] William E Schlenger et al. "Psychological reactions to terrorist attacks: findings from the National Study of Americans' Reactions to September 11". In: *Jama* (2002).

[18] Xuan Song et al. "Intelligent system for human behavior analysis and reasoning following large-scale disasters". In: *IEEE Intelligent Systems* (2013).

[19] Andranik Tumasjan et al. "Predicting elections with twitter: What 140 characters reveal about political sentiment." In: *ICWSM* (2010).

[20] Direction de la Voirie et des déplacements - Service des Déplacements. *Données trafic issues des capteurs permanents*. http://opendata.paris.fr/explore/dataset/comptages-routiers-permanents/.

[21] Qi Wang and John E Taylor. "Quantifying human mobility perturbation and resilience in Hurricane Sandy". In: *PLoS one* (2014).

[22] Xiaofeng Wang, Matthew S Gerber, and Donald E Brown. "Automatic crime prediction using events extracted from twitter posts". In: *SBP*. 2012.

[23] Yangyang Xu. "Alternating proximal gradient method for nonnegative matrix factorization". In: *arXiv preprint arXiv:1112.5407v1* (2011).

[24] Yangyang Xu and Wotao Yin. "A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion". In: *SIAM Journal on imaging sciences* (2013).

[25] Dingqi Yang, Daqing Zhang, and Bingqing Qu. "Participatory Cultural Mapping Based on Collective Behavior Data in Location-Based Social Networks". In: *ACM TIST* (2016).

[26] Fuzheng Zhang et al. "Sensing the pulse of urban refueling behavior: A perspective from taxi mobility". In: *ACM TIST* (2015).