



HAL
open science

Tensor Methods for Multi-Aspect Trajectory Data Mining

Eleftherios Kofidis

► **To cite this version:**

| Eleftherios Kofidis. Tensor Methods for Multi-Aspect Trajectory Data Mining. 2020. hal-02639989

HAL Id: hal-02639989

<https://hal.science/hal-02639989>

Preprint submitted on 28 May 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Tensor Methods for Multi-Aspect Trajectory Data Mining

Eleftherios Kofidis

Abstract

Tensor models have been well established as a natural and powerful way of representing systems and data that involve multiple aspects/dimensions. Assisted by their unique ability to unveil latent information through tensor decomposition methods, they have proved successful in numerous applications. It is thus of no surprise that they have also been employed for trajectory data mining applications, especially those that enrich mobility data with additional, generally heterogeneous aspects. This article aims at providing a brief overview of the related literature.

Index Terms

Block-Term Decomposition (BTD), Canonical Polyadic Decomposition (CPD), data mining, matrix, rank, tensor, trajectory, Tucker decomposition (TD)

I. INTRODUCTION

MASSIVE spatial trajectory data, representing the mobility of a diversity of moving objects, such as people, vehicles, and animals, have become available with the advent of location-acquisition and mobile computing technologies [1]. This has fed a number of related data mining problems, such as trajectory pattern mining and classification, with applications found in urban computing, disaster prevention and management, and aircraft and vessel trip planning, among others. Enriching raw trajectories of (x, y, t) points with additional and heterogeneous dimensions (such as weather conditions, points of interest (POIs), social interactions, etc.) gives rise to the so-called *multi-aspect* trajectory data.

The multi-aspect character of these data renders *tensor* models and methods [2] well-suited to their representation and analysis. Tensors, i.e., multi-dimensional arrays, are a natural way of representing systems and data involving (explicitly or implicitly) multiple dimensions (or “ways”/“modes”). For example, a 3-way location tensor may show the set of (x, y, t) points of a raw trajectory. Refining the time description to week day and hour of day would increase the order of the tensor to 4. Additional dimensions can be included (provided they have some correlation with the rest) towards a higher-order tensor with a higher expressive power. Alternatively, contextual data can be considered (as separate matrices or tensors) through their *coupling* (fusion) with the tensor. Besides their being a natural representation model, able to capture multiple dependencies and relations, tensors are useful for their richness of decomposition models and methods, which allow to unveil latent information in the data not recoverable otherwise [2]. They have thus been successfully applied in several data mining tasks, including those of urban computing [3], in the form of a number of possible decomposition models adaptable to the specifics of each application [4].

For trajectory data mining, tensors can serve as alternative (for example, to graphs or matrices) models¹ [1] and hence allow the exploitation of the rich arsenal of tensor decomposition methods to accomplish tasks like pattern extraction [3], location or trajectory prediction [6], and completion of missing data [7]. Non-static scenarios, such as with the formation and deformation of groups of moving objects, can be coped with incremental and online tensor decomposition [8], [9]. This article aims at providing a brief overview of the related literature, including works from other areas (e.g., computer vision) that are deemed pertinent.

The rest of this article is organized as follows. The notation to be used is explained in the following subsection. Knowledge about tensors and their decomposition needed for the purposes of this article is outlined in Section II. Section III reviews the literature of tensor-based trajectory data mining. Related work on other kinds of data that could however be helpful in this context is briefly cited in Section IV. Some required math is deferred to the appendix.

A. Notation

Bold lower- and upper-case letters will be used to denote vectors and matrices, respectively. Tensors will be denoted by bold upper-case calligraphic letters. The mode- n unfolding of a tensor \mathcal{X} will be denoted by $\mathbf{X}_{(n)}$. \times_n will denote the mode- n product. The superscripts T and \dagger will denote transposition and pseudo-inversion, respectively. The outer, Kronecker, Khatri-Rao and column-wise Khatri-Rao products will be respectively denoted by \circ , \otimes , \odot , and \odot_c . $\|\cdot\|_{\text{F}}$ stands for the Frobenius norm. \mathbb{R} is the field of real numbers.

E. Kofidis is with the Department of Statistics and Insurance Science, University of Piraeus, 185 34 Piraeus, Greece and the Computer Technology Institute & Press “Diophantus” (CTI), Greece. E-mail: kofidis@unipi.gr (<https://www.unipi.gr/unipi/en/kofidis.html>).

¹For example, the development of a non-parametric, non-sequential (all locations are considered at once) model for human trajectory synthesis that is proposed in [5] with the aid of adversarial networks, relies on a 4th-order tensor (space, time and duration at stay points) to represent the trajectory data.

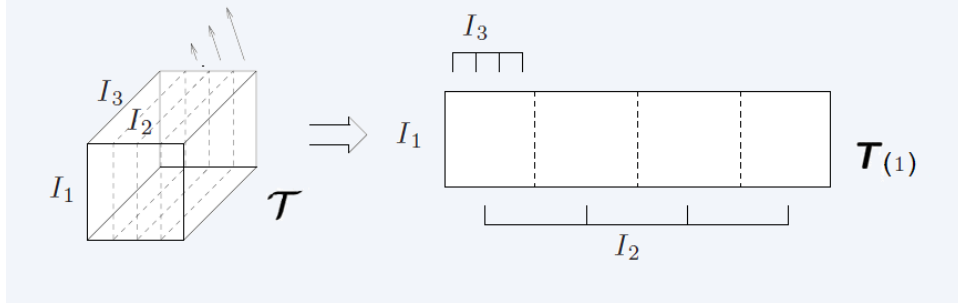


Fig. 1. Mode-1 unfolding of a 3-way tensor.

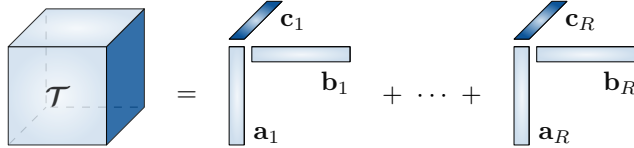


Fig. 2. Canonical polyadic decomposition of a 3rd-order tensor.

II. TENSORS: A BRIEF INTRODUCTION

Tensors are arrays of multiple dimensions² generalizing matrices. The number of dimensions (or ways or modes³) is called the *order* of the tensor. Thus, a matrix is a 2nd-order or 2-way tensor while a vector is a tensor of order one (and a scalar is a tensor of order zero).

Although higher-order tensors are a subject of *multi-linear algebra*, in practice they can be seen and processed with the aid of matrix algebra via their matricization or *unfolding*. Consider, for simplicity and without loss of generality, 3-way tensors, and let $\mathcal{T} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$. There are three unfoldings of \mathcal{T} , one for each of its three modes. The mode-1 unfolding is the $I_1 \times I_2 I_3$ matrix $\mathbf{T}_{(1)}$ built as shown in Fig. 1. Thus, modes 2 and 3 are combined (concatenated) and the index for mode 3 runs faster than for mode 2. Mode-2, $\mathbf{T}_{(2)} \in \mathbb{R}^{I_2 \times I_3 I_1}$, and mode-3, $\mathbf{T}_{(3)} \in \mathbb{R}^{I_3 \times I_1 I_2}$, unfoldings result in an analogous manner, with the index of the second of the concatenated modes running faster. A tensor can also be viewed as consisting of slices (or slabs). Thus, as shown in Fig. 1, $\mathbf{T}_{(1)}$ results from the concatenation of the *lateral* slices of the tensor. $\mathbf{T}_{(2)}$ and $\mathbf{T}_{(3)}$ result by concatenating its *frontal* and *horizontal* slices, respectively,

Although, as seen above, a tensor is simply the higher-order analogue of a matrix and it can be represented (and processed) via its matrix unfoldings, there are a few fundamental differences with matrices, which make tensors more useful and interesting. First, in contrast to the matrix case, where the rank is well-defined and relatively easy (in polynomial time) to be computed, the notion of *tensor rank* is not always well-defined and its computation is far from being easy (an NP-hard problem [2]). The rank of a matrix is equal to its column rank (the number of linearly independent columns) and its row rank (the number of linearly independent rows). It can also be defined as the minimum number of rank-1 (=outer product of two vectors) matrices that give the matrix in a linear combination. For a higher (than 2) order tensor, the ranks of its column (mode-1) space, row (mode-2) space, tube (mode-3) space, etc. do not necessarily coincide with each other nor with the minimum number of rank-1 tensors summing up to the given tensor. It is in the latter way that the tensor rank is defined. Thus, in analogy with a rank-1 matrix, a rank-1 tensor (say of order 3) is the outer product of three vectors: $\mathcal{T} = \mathbf{a} \circ \mathbf{b} \circ \mathbf{c}$. What does this represent? Seeing the tensor as a stacking of its frontal slices, a rank-1 tensor has frontal slices that are rank-1 matrices that are scalar multiples of one another. If the tensor is expressed as the sum of R such tensors and R is minimal (=tensor rank), the so-called canonical polyadic decomposition (CPD) [2] results:

$$\mathcal{T} = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r \triangleq [\mathbf{A}, \mathbf{B}, \mathbf{C}] \quad (1)$$

with $\mathbf{A} \triangleq [\mathbf{a}_1 \ \mathbf{a}_2 \ \cdots \ \mathbf{a}_R] \in \mathbb{R}^{I_1 \times R}$ and $\mathbf{B} \in \mathbb{R}^{I_2 \times R}$ and $\mathbf{C} \in \mathbb{R}^{I_3 \times R}$ similarly defined. CPD (less frequently referred to as PARAFAC or CANDECOMP or Kruskal decomposition) is the most widely known and used tensor decomposition model, due to its conceptual simplicity. Each of the R rank-1 terms can be seen as representing one of the R components (or clusters) forming the tensor. A schematic diagram of CPD is given in Fig. 2.

²In a manner analogous to the way matrices (2-way arrays) represent linear operators, tensors represent *multi-linear* operators. A tensor will, however, be simply defined here as an array of numbers.

³These two terms do not in general have exactly the same meaning but they are commonly used interchangeably in the tensor data mining literature.

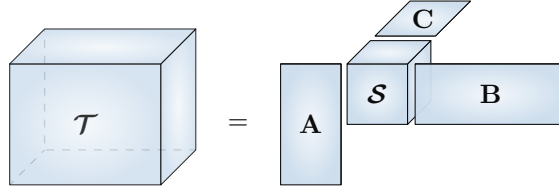


Fig. 3. Tucker decomposition (TD) of a 3rd-order tensor.

Recall the slice-based interpretation of the rank-1 tensor described previously. Adding R such terms as above readily leads to the following expression for the i_3 th frontal slice of \mathcal{T} :⁴

$$\mathcal{T}(:, :, i_3) = \mathbf{a}_1 \cdot [\mathbf{c}_1]_{i_3} \cdot \mathbf{b}_1 + \cdots + \mathbf{a}_R \cdot [\mathbf{c}_R]_{i_3} \cdot \mathbf{b}_R = \mathbf{A} \cdot \text{diag}(\mathbf{C}(i_3, :)) \cdot \mathbf{B}^T, \quad i_3 = 1, 2, \dots, I_3 \quad (2)$$

This viewpoint of CPD reveals another very important difference with matrices. Instead of decomposing (factorizing) a single matrix, a number of matrices are *jointly* decomposed, with the term *jointly* here referring to the common factors \mathbf{A} and \mathbf{B} . The $I_3 > 1$ factorizations in (2) can thus be expected to allow for the identification of $\mathbf{A}, \mathbf{B}, \mathbf{C}$. Moreover this is possible under *mild* conditions, which contrasts the inherent non-uniqueness of unconstrained (or mildly constrained) matrix decompositions.⁵ The most well-known example of *essential*⁶ CPD uniqueness condition is the (sufficient) Kruskal's condition [2], which states that, if $k(\mathbf{A}) + k(\mathbf{B}) + k(\mathbf{C}) \geq 2R + 2$, then R is the tensor rank and $\mathbf{A}, \mathbf{B}, \mathbf{C}$ are essentially unique. In this condition, $k(\mathbf{M})$ stands for the Kruskal rank of a matrix \mathbf{M} , that is, the maximum number k such that *any* k columns of \mathbf{M} are linearly independent. Obviously $k(\mathbf{M}) \leq \text{rank}(\mathbf{M})$, with the equality holding for generic matrices.

It is thus of no surprise that the CPD model has been by far the most popular in the applications of tensor methods: it is conceptually simple and allows the unique identification of its factors, which is what is mostly needed in multi-way data analysis applications. Moreover, as also mentioned in the following, a rich palette of algorithms is available for its computation. However, CPD is also very *constrained* in its structure. Namely, each component in one of the modes only interacts with the corresponding components in the other modes. No other interaction is allowed, which renders the CPD model inadequate to capture more complex multi-dimensional relations as it is required in some contexts [10]. A much less constrained decomposition model, featuring full interaction among all the components, is offered by the so-called Tucker Decomposition (TD) [2]:

$$\mathcal{T} = \mathcal{S} \times_1 \mathbf{A} \times_2 \mathbf{B} \times_3 \mathbf{C}, \quad (3)$$

where $\mathbf{A}, \mathbf{B}, \mathbf{C}$ may now have different numbers of columns, say R_1, R_2, R_3 , \mathcal{S} is an $R_1 \times R_2 \times R_3$ tensor (called *core*), and the mode- n product $\mathcal{S} \times_n \mathbf{U}$ yields a tensor with mode- n unfolding $\mathbf{U}\mathbf{S}_{(n)}$. TD is schematically given in Fig. 3. The most popular method of computing the TD, higher-order singular value decomposition (HOSVD), relies on the SVDs of the tensor unfoldings and offers an (approximate) alternative to the matrix SVD tool for estimating the modal ranks, $\text{rank}(\mathbf{T}_{(n)})$, and subspaces of a tensor. Possibly with truncation per mode, this is a quasi-optimal (in the least squares sense) rank- (R_1, R_2, R_3) approximation. Observe that the CPD is a special case of the TD, corresponding to a diagonal core tensor.

Though a powerful representation model, the TD suffers from lack of uniqueness, contrasting the mild uniqueness conditions for CPD. A non-unique decomposition can be useless in practice since it may not allow the extraction of interpretable information. An intermediate, in terms of degrees of freedom and uniqueness guarantees, alternative is given by the so-called block-term decomposition (BTD) [11]. In its most widely used version, known as rank- $(L_r, L_r, 1)$ BTD, it has the form

$$\mathcal{T} = \sum_{r=1}^R \mathbf{E}_r \circ \mathbf{c}_r, \quad (4)$$

where the $I_1 \times I_2$ matrix \mathbf{E}_r is of rank L_r and can hence be factorized as $\mathbf{A}_r \mathbf{B}_r^T$ with $\mathbf{A} \in \mathbb{R}^{I_1 \times L_r}$ and $\mathbf{B} \in \mathbb{R}^{I_2 \times L_r}$. Fig. 4 depicts BTD schematically. A comparison with (1) shows that rank- $(L_r, L_r, 1)$ BTD generalizes CPD by allowing the first two modes to have a richer than rank-1 structure per term. Note that CPD can be otherwise seen as rank- $(1, 1, 1)$ BTD. Lying between the two extremes, namely CPD and TD, BTD enjoys a combination of their advantages. As expected, however, its uniqueness is less well guaranteed than in CPD. Thus, a (sufficient) condition for (essentially) identifying \mathbf{c}_r 's and $\mathbf{A}_r, \mathbf{B}_r$, subject to an $L_r \times L_r$ matrix uncertainty per block term, is that $\mathbf{A} \triangleq [\mathbf{A}_1 \ \mathbf{A}_2 \ \cdots \ \mathbf{A}_R]$ and $\mathbf{B} \triangleq [\mathbf{B}_1 \ \mathbf{B}_2 \ \cdots \ \mathbf{B}_R]$ be of full column rank, and $\mathbf{C} \triangleq [\mathbf{c}_1 \ \mathbf{c}_2 \ \cdots \ \mathbf{c}_R]$ have no collinear columns [11]. That the factors $\mathbf{A}_r, \mathbf{B}_r$ can only be identified with a matrix uncertainty should be expected from the intrinsic non-uniqueness of the factorization of the matrix \mathbf{E}_r . Nevertheless, it is very commonly the case in practice that only their product, \mathbf{E}_r , needs to be uniquely identified (e.g., [10]).

⁴Analogous expressions can be written for the other modes, due to the symmetry of the CPD model.

⁵Unless strong constraints (such as orthogonality, sparsity, statistical independence, etc.) are imposed on a matrix decomposition $\mathbf{T} = \mathbf{A}\mathbf{B}^T$, it is impossible to uniquely determine its factors since $\tilde{\mathbf{A}} \triangleq \mathbf{A}\mathbf{Q}$ and $\tilde{\mathbf{B}} \triangleq \mathbf{B}\mathbf{Q}^{-T}$ will give the same result for *any* non-singular matrix \mathbf{Q} .

⁶Clearly, the CPD factors can only be determined subject to a (column) permutation and scaling ambiguity.

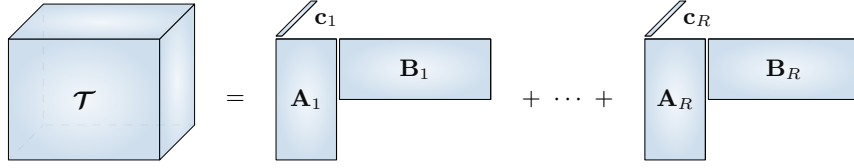


Fig. 4. Rank- $(L_r, L_r, 1)$ block-term decomposition (BTD) of a 3rd-order tensor.

Subject to the uniqueness condition and provided there is no modeling error, the BTD of a tensor can be algebraically computed with the aid of a generalized eigenvalue decomposition (GEVD) [11] or a simultaneous matrix diagonalization (SMD) method [12]. In practice, however, a best BTD approximation is sought for, which can be stated as follows, on the common assumption of Gaussian i.i.d noise:

$$\min_{\mathbf{A}, \mathbf{B}, \mathbf{C}} \left\| \mathcal{T} - \sum_{r=1}^R (\mathbf{A}_r \mathbf{B}_r^T) \circ \mathbf{c}_r \right\|_{\mathbb{F}}$$

This problem can be solved via an alternating least squares (ALS) method, which alternately solves for each one of the factors considering the others fixed at their present values [13]. To this end, the three mode unfoldings of the BTD are employed:

$$\mathbf{T}_{(1)}^T = (\mathbf{B} \odot \mathbf{C}) \mathbf{A}^T \quad (5)$$

$$\mathbf{T}_{(2)}^T = (\mathbf{C} \odot \mathbf{A}) \mathbf{B}^T \quad (6)$$

$$\mathbf{T}_{(3)}^T = \underbrace{\left[(\mathbf{A}_1 \odot_c \mathbf{B}_1) \mathbf{1}_{L_1} \quad \cdots \quad (\mathbf{A}_R \odot_c \mathbf{B}_R) \mathbf{1}_{L_R} \right]}_{\mathbf{S}} \mathbf{C}^T \quad (7)$$

Note that the above holds for CPD as well, with the Khatri-Rao products being defined columnwise and $\mathbf{S} \triangleq \mathbf{A} \odot_c \mathbf{B}$. All-at-once optimization schemes, for example employing nonlinear least squares [14], can be also used. It should be noted that, in view of the non convex nature of the problem, local minima might be reached depending on the initialization. In practice, multiple random (and GEVD or SMD-based, whenever possible) initialization is employed to ensure global optimization.

In addition to constraints on the structure of the decomposition, giving rise to the decomposition models described previously as well as a variety of others, other constraints, that can be dictated by the application context, can be incorporated. A notable example, particularly relevant in mobility data applications, is the constraint of *non-negativity*. A non-negative tensor may have to be factorized into factors that should also be non-negative in order for them to be meaningful. Algorithms for non-negative tensor decomposition are available, often inspired from non-negative matrix factorization. Nevertheless, constraints generally affect the uniqueness conditions as well as the rank of the tensor itself.

What is, however, much more challenging (in tensor decomposition in general but more specifically in BTD) is to efficiently estimate the model structure (ranks R and L_r for rank- $(L_r, L_r, 1)$ BTD) for a given tensor, particularly if it is a big one. Research in this subject is under-way and promising results for the BTD case were recently reported in [15].

III. TENSOR-BASED TRAJECTORY DATA MINING

Capitalizing on the expressive power of tensor decomposition models, a number of works on trajectory data mining rely on tensor methods for addressing the problem of scarcity of measurements. When considering a big urban network, for example, there may be no (or very few) data associated with some of the road segments. Higher-order tensor *completion* [7] proves a quite good solution to this kind of problems, able to deal even with missing data percentages of the order of 90 %. The reason behind this success is the high ability of the tensor models to capture underlying multi-dimensional relations, which, when translated to a low-rank tensor decomposition, can successfully effect the missing data imputation. TD is proposed in [16] for imputing missing traffic data, outperforming statistical/matrix-based techniques. The tensor modes are link, week, day, hour. The *tensorization* (folding) of the temporal dimension here is typical of the way implicit multi-dimensionality is translated into tensor modes. Similar ideas, again based on TD for tensor completion, are applied in [17] for traffic prediction. Notably, the tensor model therein is dynamic (a tensor sequence) and so is the completion approach proposed. Advantages over matrix-based techniques which do not fully exploit the inherent multi-dimensionality of the data are demonstrated. The imputation of missing speed measurements is achieved in [18] with the aid of an orthogonal TD of the (time of day, sensor, day) speed tensor, whose low multilinear rank is ensured via iterative soft thresholding that exploits the core tensor compressibility.

Estimation of travel time per road segment in *real* time is addressed in [19], again through a tensor representation (collecting GPS data) and its decomposition coupled with context information (including historical trajectory, POI, correlation with neighboring segments and times, etc.) formulated in matrices. Frobenius norm (Tikhonov) regularization is used to avoid overfitting, a common approach in tensor decomposition computation. Coupling with context matrices aids in solving the data scarcity problem. A very similar tensor-based (BTD) context-aware approach is followed in [20] to dynamically provide

personalized travel time estimation from a citywide perspective, using sparse and large-scale GPS trajectories. Speed and hence travel time is predicted in [21] with the aid of deep learning applied in the result of the speed tensor (non-negative CPD) factorization. The latter is (sparsely) built out of the speed data for (location, time) pairs, at three levels of temporal granularity. A non-negative CPD is also used in [22] to help extract activity patterns and hence classify the movement data into movement types such as home-to-work and work-to-home. A mobility graph (containing the flow among regions) is first constructed and then re-formulated into a tensor, whose t th frontal slice is the adjacency matrix of the flow graph at time t . The fusion (through coupled decomposition) of more than one transport modes is mentioned among the future research directions. The extraction of activity patterns in disaster evacuation scenarios with the aid of tensor CPD is proposed in [23]. The tensor contains the frequency of each person status (e.g., stay at home, move, safe, etc.) at (location, time) points. The high sparsity of the tensor is again addressed via its decomposition. Fusing (coupling) the person status with the movement direction tensors is also proposed. The same approach is followed in [24], namely the data sparsity problem is addressed via a context-aware CPD of the location-time tensor, to aid in the reconstruction of hidden trajectories. An analogous approach is taken in [25] for estimating the bus speeds. The TD of the traffic tensor, which contains the traffic volume per sensor, week, day and time of the day, helps coping with the data sparsity problem in [26]. The aim of the study is to predict travel time based on sparse traffic measurements and it achieves that with the aid of k -NN clustering on the completed traffic tensor. Clustering of the CPD components of a (location, time, user) tensor (possibly with additional contextual information) is employed in [27] to extract patterns of loan/return in bike sharing. Reliance on tensor completion for complete trajectory reconstruction from sparse mobile phone data is also found in [28], where the location tensor, containing the sub-trajectory per week and day of week, is CPDed. A tensor representation of the (x, y, t) data for pedestrian mobility is adopted and its sparsity is efficiently taken into account in [29] for the discovery of group patterns. In an effort to estimate the duration of refueling events, [30] also employs coupled matrix and tensor factorization (coupled regularized HOSVD) to solve the sparsity issue along time and gas station modes. Context information (such as POIs, price, brand, weather, etc.) is shown to considerably aid the task. Determining the real-time status of gas stations is mentioned among the plans for future work. In an Internet of Things (IoT) context, [31] benefits from the power of tensor completion to interpolate the missing spatio-temporal observations from a city-wide sensor network.

To perform network-wide prediction of speed and at multiple horizons, tensor representation is employed in [32]. Prediction is based on finding a common latent subspace between current-past and future states, using multi-way partial least squares (PLS). CPD, TD, and BTD are considered for modeling the tensor, with insignificant differences in the prediction performance attained here. An analogous problem is addressed in [33], where non-negative tensor decomposition is employed to produce features that are then clustered (K -means) for predicting traffic behavior. In [34], the tensor model is only used to contain the (x, y, t) data before it is condensed along time to a matrix for the purpose of predicting future location based on an exponentially fading window of recent locations. [35] encodes past trajectories and the scene context into a multi-way tensor that feeds a CNN aiming at predicting (other) drivers' trajectories.

In [36], a single transit trip is represented by a multivariate tuple (card ID, passenger type, time, boarding stop/station, alighting stop/station). Probabilistic latent semantic analysis (PLSA) is adopted, with the probabilities contained in a probability tensor and the probabilistic profile of the card ID pattern, passenger pattern, etc. being computed from the core tensor of its TD. The expectation maximization (EM) algorithm is employed for estimating the parameters of the PLSA model. A 4-way tensor, with modes for origin, destination, time, and day category (weekday or weekend), is built in [37] with the corresponding transition probabilities. Its high sparsity is again dealt with via its factorization. Notably, the large tensor size is addressed with the aid of a divide-and-conquer method resembling ParCube in [3]. Prediction of mobility is performed in [38] with the aid of decomposing the probability transition tensor corresponding to a multi-variate (i.e., with multiple attributes), multi-order, multi-modal and multi-user Markov model. In [39], a context-aware TD of the trajectory tensor (origin, destination, time) similar to that of [19] is proposed for understanding and predicting urban dynamics. A Bayesian approach is taken to solving the regularized decomposition. Similar contents, with a so-called multivariate multi-step transition tensor (M^2T^2), are found in [40]. The power of tensor-based completion/inference is demonstrated in [41], where location privacy is shown to be vulnerable to attacks utilizing EM-based tensor factorization even in the realistic scenario where very little training is available and some locations are even missing. [42] leverages the potential of factorization methods in user recommendation applications to develop a method of predicting *semantic* locations based on a Markov model which is built upon the movement history. Personalized predictions are achieved by including a user-specific mode to the matrix factorization model, giving rise to a tensor CPD.

With the aid of the CPD of a (location, vessel, time) tensor encoding the vessels' trajectories that are deduced from Automatic Identification System (AIS) data, [43] discovers mobility patterns and addresses the data sparsity issue. [44] goes one step further, by discovering mobility patterns via co-clustering. The latter is implemented (as in [45]) with the aid of the non-negative CPD of the AIS tensor, properly regularized to ensure sparse latent factors as required by the clustering problem. More information can be added to these trajectory points, including vessel identity, course/speed, ship type, etc.

IV. RELATED WORK

Though strictly speaking not relevant to trajectory data mining, there are some more works that deserve to be considered for their relevance with tensor-based methods and their close relation or analogy with the application of interest here. These

include applications in computer vision, where motion trajectories in videos are analyzed. [46] assesses tensor decomposition models for representing and effecting queries of multiple-object trajectories in video indexing and retrieval. In [47], the problem of matching trajectories over video frames for multi-target tracking is addressed with the aid of rank-1 tensor approximation. Human motion (trajectories of body joints), for ergonomic applications such as design of car seats, are analyzed with the aid of tensor modeling in [48]. Detecting *semantic* events in audio-visual multimedia is also relevant here, particularly when leveraging the multi-modal aspects of the problem [49].

Other related works include [50], [51] for context-aware (including geographic data) collaborative tensor-based filtering, and [52], which addresses the rareness of abnormal events in surveillance applications through factorizing an (object, time, location) tensor containing the frequencies of events. In [53], the aim is to leverage non-negative CPD to detect groups of players characterized by similar features (i.e., actions they perform during the game) and strategies, as well as their temporal trajectories, i.e., their evolution. This human behavioral pattern mining as applied in online gaming bears a close resemblance to unsupervised source localization and tracking in brain imaging [10].

APPENDIX

Similarly with vectors, one can define the inner product of two tensors of equal order and dimensions:

$$\langle \mathcal{S}, \mathcal{T} \rangle \triangleq \sum_{i_1, i_2, \dots, i_N} \mathcal{S}_{i_1, i_2, \dots, i_N} \mathcal{T}_{i_1, i_2, \dots, i_N},$$

whereby the Frobenius norm results as

$$\|\mathcal{T}\|_F \triangleq \sqrt{\langle \mathcal{T}, \mathcal{T} \rangle}$$

The outer product of N vectors $\mathbf{u}^{(n)} \in \mathbf{R}^{I_n \times 1}$, $n = 1, 2, \dots, N$,

$$\mathcal{T} = \mathbf{u}^{(1)} \circ \mathbf{u}^{(2)} \circ \dots \circ \mathbf{u}^{(N)},$$

is a rank-1 N th-order $I_1 \times I_2 \times \dots \times I_N$ tensor with its (i_1, i_2, \dots, i_N) entry being given by the product of the corresponding vector entries: $\mathcal{T}_{i_1, i_2, \dots, i_N} = u_{i_1}^{(1)} u_{i_2}^{(2)} \dots u_{i_N}^{(N)}$. The outer product of M tensors can be defined in an analogous manner. For example, the outer product of a $I_1 \times I_2$ matrix \mathbf{E} and a $I_3 \times 1$ vector \mathbf{c} is the $I_1 \times I_2 \times I_3$ tensor with (i_1, i_2, i_3) entry given by $E_{i_1, i_2} c_{i_3}$.

The Kronecker product of two matrices $\mathbf{A} \in \mathbf{R}^{M \times N}$ and $\mathbf{B} \in \mathbf{R}^{P \times Q}$ is the $MP \times NQ$ matrix

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{1,1} \mathbf{B} & a_{1,2} \mathbf{B} & \dots & a_{1,N} \mathbf{B} \\ a_{2,1} \mathbf{B} & a_{2,2} \mathbf{B} & \dots & a_{2,N} \mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ a_{M,1} \mathbf{B} & a_{M,2} \mathbf{B} & \dots & a_{M,N} \mathbf{B} \end{bmatrix}$$

The (partition-wise) Khatri-Rao product of two partitioned matrices, $\mathbf{A} = [\mathbf{A}_1 \ \dots \ \mathbf{A}_R]$ and $\mathbf{B} = [\mathbf{B}_1 \ \dots \ \mathbf{B}_R]$, with equal number of blocks, is defined as the matrix whose blocks are the Kronecker products of the corresponding \mathbf{A} and \mathbf{B} blocks:

$$\mathbf{A} \odot \mathbf{B} = [\mathbf{A}_1 \otimes \mathbf{B}_1 \ \dots \ \mathbf{A}_R \otimes \mathbf{B}_R]$$

If the partitioning is in columns (and hence the two matrices have the same number of columns), the column-wise Khatri-Rao product results:

$$\mathbf{A} \odot_c \mathbf{B} = [\mathbf{a}_1 \otimes \mathbf{b}_1 \ \dots \ \mathbf{a}_R \otimes \mathbf{b}_R]$$

ACKNOWLEDGMENT

This paper is supported by the MASTER project that has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement N°777695. The work reflects only the author's view and the EU Agency is not responsible for any use that may be made of the information it contains.

This work was done during the author's secondment at the Department of Computer Science, Dalhousie University, Halifax, NS, Canada. The hospitality of the Institute for Big Data Analytics is gratefully acknowledged. Special thanks are due to Prof. S. Matwin, Prof. L. Torgo, Prof. E. Milios, Dr. A. Soares, and Mr. M. Etemad. Last but not least, the kind hospitality and friendship of Dr. M. Gillis is gratefully acknowledged.

REFERENCES

- [1] Y. Zheng, "Trajectory data mining: An overview," *ACM Trans. Intelligent Systems and Technology*, vol. 6, no. 3, pp. 29:1–29:41, May 2015.
- [2] N. D. Sidiropoulos *et al.*, "Tensor decomposition for signal processing and machine learning," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3551–3582, Jul. 2017.
- [3] E. E. Papalexakis, C. Faloutsos, and N. D. Sidiropoulos, "Tensors for data mining and data fusion: Models, applications, and scalable algorithms," *ACM Trans. Intelligent Systems and Technology*, vol. 8, no. 2, Oct. 2016.
- [4] E. E. Papalexakis and C. Faloutsos, "Unsupervised tensor mining for big data practitioners," *Big Data*, vol. 4, no. 3, pp. 179–191, 2016.
- [5] K. Ouyang *et al.*, "A non-parametric generative model for human trajectories," in *Proc. IJCAI-18*, Stockholm, Sweden, Jul. 2018.
- [6] H. Georgiou *et al.*, "Moving objects analytics: Survey on future location & trajectory prediction methods," *arXiv:1807.04639v1 [cs.LG]*, Jul. 2018. [Online]. Available: <https://arxiv.org/abs/1807.04639>
- [7] Q. Song *et al.*, "Tensor completion algorithms in big data analytics," *ACM Trans. Knowledge Discovery from Data*, vol. 13, no. 1, pp. 6:1–6:48, Jan. 2019.
- [8] L. Kuang *et al.*, "A tensor-based approach for big data representation and dimensionality reduction," *IEEE Trans. Emerg. Topics Comput.*, vol. 2, no. 3, pp. 280–291, Jun. 2014.
- [9] H. Fanaee-T and J. Gama, "Multi-aspect-streaming tensor analysis," *Knowledge-Based Systems*, vol. 89, pp. 332–345, 2015.
- [10] K. Chatzichristos *et al.*, "Blind fMRI source unmixing via higher-order tensor decompositions," *J. Neurosci. Methods*, vol. 315, pp. 17–47, Mar. 2019.
- [11] L. De Lathauwer, "Decompositions of a higher-order tensor in block terms — Part II: Definitions and uniqueness," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 3, pp. 1033–1066, 2008.
- [12] —, "Blind separation of exponential polynomials and the decomposition of a tensor in rank- $(L_r, L_r, 1)$ terms," *SIAM J. Matrix Anal. Appl.*, vol. 32, no. 4, pp. 1451–1474, 2011.
- [13] L. De Lathauwer and D. Nion, "Decompositions of a higher-order tensor in block terms — Part III: Alternating least squares algorithms," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 3, pp. 1067–1083, 2008.
- [14] L. Sorber, M. Van Barel, and L. De Lathauwer, "Optimization-based algorithms for tensor decompositions: Canonical polyadic decomposition, decomposition in rank- $(L_r, L_r, 1)$ terms, and a new generalization," *SIAM J. Optim.*, vol. 23, no. 2, pp. 695–720, 2013.
- [15] A. A. Rontogiannis, E. Kofidis, and P. V. Giampouras, "Block-term tensor decomposition: Model selection and computation," *arXiv:2002.09759v1 [cs.NA]*, Feb. 2020, <https://arxiv.org/abs/2002.09759>.
- [16] H. Tan *et al.*, "A tensor-based method for missing traffic data completion," *Transportation Research Part C*, vol. 28, pp. 15–27, 2013.
- [17] —, "Short-term traffic prediction based on dynamic tensor completion," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 8, pp. 2123–2133, Aug. 2016.
- [18] J. H. de M. Goulart, A. Y. Kibangou, and G. Favier, "Traffic data imputation via tensor completion based on soft thresholding of Tucker core," *Transportation Research Part C*, vol. 85, pp. 348–362, 2017.
- [19] Y. Wang, Y. Zheng, and Y. Xue, "Travel time estimation of a path using sparse trajectories," in *Proc. KDD-2014*, New York City, NY, Aug. 2014.
- [20] K. Tang, S. Chen, and A. J. Khattak, "Personalized travel time estimation for urban road networks: A tensor-based context-aware approach," *Expert Systems With Applications*, vol. 103, p. 118–132, 2018.
- [21] Y. Shen *et al.*, "TCL: Tensor-CNN-LSTM for travel time prediction with sparse trajectory data," in *Proc. DASFAA-2019*, Chiang Mai, Thailand, Apr. 2019.
- [22] K. S. Naveh and J. Kim, "Urban trajectory analytics: Day-of-week movement pattern mining using tensor factorization," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 7, pp. 2540–2549, Jul. 2019.
- [23] Y. Kawai, Y. Ishikawa, and K. Sugiura, "Analysis of evacuation trajectory data using tensor decomposition," *Journal of Disaster Research*, vol. 14, no. 3, pp. 521–530, 2019.
- [24] N. Yang and P. S. Yu, "Efficient hidden trajectory reconstruction from sparse data," in *Proc. CIKM-16*, Indianapolis, IN, Oct. 2016.
- [25] F. Yang *et al.*, "Measuring temporal and spatial travel efficiency for transit route system using low-frequency bus automatic vehicle location data," *Advances in Mechanical Engineering*, vol. 10, no. 10, pp. 1–11, 2018.
- [26] J. Zhao *et al.*, "Highway travel time prediction using sparse tensor completion tactics and k -nearest neighbor pattern matching method," *Journal of Advanced Transportation*, vol. 2018, 2018, Article ID 5721058.
- [27] Y. Yan *et al.*, "Visual analytics of bike-sharing data based on tensor factorization," *J. Vis.*, vol. 21, pp. 495–509, 2018.
- [28] G. Chen *et al.*, "Complete trajectory reconstruction from sparse mobile phone data," *EPJ Data Science*, vol. 8, no. 30, 2019.
- [29] A. Sawas *et al.*, "Tensor methods for group pattern discovery of pedestrian trajectories," in *Proc. MDM-2018*, Aalborg, Denmark, Jun. 2018.
- [30] F. Zhang *et al.*, "Sensing the pulse of urban refueling behavior: A perspective from taxi mobility," *ACM Trans. Intelligent Systems and Technology*, vol. 9, no. 4, pp. 24:1–24:24, Dec. 2013.
- [31] P. Xu *et al.*, "Interpolating the missing values for multi-dimensional spatial-temporal sensor data: A tensor SVD approach," in *Proc. MobiQuitous-2017*, Melbourne, Australia, Nov. 2017.
- [32] J. Dauwels *et al.*, "Predicting traffic speed in urban transportation subnetworks for multiple horizons," in *Proc. ICARCV-2014*, Marina Bay Sands, Singapore, Dec. 2014.
- [33] Y. Han and F. Moutarde, "Analysis of large-scale traffic dynamics in an urban transportation network using non-negative tensor factorization," *Int. J. Intell. Transp. Syst. Res.*, vol. 14, no. 1, pp. 36–49, 2014.
- [34] J. N. S. H. V. Kamiesty, "Forecasting trajectory data: A study by experimentation," MSc thesis, Department of Communication Systems, Faculty of Computing, Blekinge Institute of Technology, Blekinge, Sweden, 2017.
- [35] T. Zhao *et al.*, "Multi-agent tensor fusion for contextual trajectory prediction," in *Proc. CVPR-2019*, Long Beach, CA, Jun. 2019.
- [36] L. Sun and K. W. Axhausen, "Understanding urban mobility patterns with a probabilistic tensor factorization framework," *Transportation Research Part B*, vol. 91, pp. 511–524, 2016.
- [37] L. Zhang *et al.*, "Fast time-aware sparse trajectories prediction with tensor factorization," *IEICE Trans. Inf. & Syst.*, vol. E101–D, no. 7, pp. 1959–1962, Jul. 2018.
- [38] J. Ding *et al.*, "Multi-user multivariate multi-order Markov based multi-modal user mobility pattern prediction," *IEEE Internet Things J.*, 2020, to appear.
- [39] J. Wang *et al.*, "Understanding urban dynamics via context-aware tensor factorization with neighboring regularization," *IEEE Trans. Knowl. Data Eng.*, 2020, to appear.
- [40] P. Wang *et al.*, " M^2T^2 : The multivariate multi-step transition tensor for user mobility pattern prediction," *IEEE Trans. Netw. Sci. Eng.*, 2020, to appear.
- [41] T. Murakami, "Expectation-maximization tensor factorization for practical location privacy attacks," *Proceedings on Privacy Enhancing Technologies*, vol. 4, pp. 58–75, 2017.
- [42] A. Karatzoglou, S. C. Lamp, and M. Beigl, "Matrix factorization on semantic trajectories for predicting future semantic locations," in *Proc. WiMob-2017*, Rome, Italy, Oct. 2017.
- [43] C. Liu and X. Chen, "Vessel track recovery with incomplete AIS data using tensor CANDECOM/PARAFAC decomposition," *The Journal of Navigation*, vol. 67, no. 1, pp. 83–99, 2013.
- [44] J. Wang *et al.*, "Vessel spatio-temporal knowledge discovery with AIS trajectories using co-clustering," *The Journal of Navigation*, vol. 70, no. 6, pp. 1383–1400, 2017.
- [45] E. E. Papalexakis, N. D. Sidiropoulos, and R. Bro, "From k -means to higher-way co-clustering: Multilinear decomposition with sparse latent factors," *IEEE Trans. Signal Process.*, vol. 61, no. 2, pp. 493–506, Jan. 2013.

- [46] X. Ma *et al.*, “Tensor-based multiple object trajectory indexing and retrieval,” in *Proc. ICME-2006*, Toronto, Canada, Jul. 2006.
- [47] X. Shi *et al.*, “Multi-target tracking by rank-1 tensor approximation,” in *Proc. CVPR-2013*, Portland, OR, Jun. 2013.
- [48] H. I. Masoud, Y. Zerehsaz, and J. J. Jin, “Analysis of human motion variation patterns using UMPCA,” *Applied Ergonomics*, vol. 59, pp. 401–409, 2017.
- [49] M. Chen *et al.*, “Semantic event detection via multimodal data mining,” *IEEE Signal Process. Mag.*, pp. 38–46, Mar. 2006.
- [50] L. Yao *et al.*, “Context-aware point-of-interest recommendation using tensor factorization with social regularization,” in *Proc. SIGIR-2015*, Santiago, Chile, Aug. 2015.
- [51] V. W. Zheng *et al.*, “Towards mobile intelligence: Learning from GPS history data for collaborative recommendation,” *Artificial Intelligence*, vol. 184–185, pp. 17–37, 2012.
- [52] A. J. Shah, “Abnormal behavior detection using tensor factorization,” MASC thesis, École de Technologie Supérieure, Université du Québec, Montreal, Canada, 2016.
- [53] A. Sapienza, A. Bessi, and E. Ferrara, “Non-negative tensor factorization for human behavioral pattern mining in online games,” *Information*, vol. 9, no. 66, pp. 1–18, 2018.