2019 IEEE International Conference on Big Data (Big Data)

Collective Tensor Completion with Multiple Heterogeneous Side Information

Huiyuan Chen
Department of Computer and Data Sciences
Case Western Reserve University
hxc501@case.edu

Jing Li
Department of Computer and Data Sciences
Case Western Reserve University
jingli@cwru.edu

***Abstract*—Tensor completion has been successfully applied to many real-world applications. In a wide variety of situations, data utilized in many learning tasks are of high dimensions, usually extracted from multiple heterogeneous sources. Therefore, data can be represented by a primary tensor and multiple matrices generated from multi-view side information or metadata. Joint analysis of tensors and matrices has great potential to gain better understanding of the underlying relationships among these multiple heterogeneous sources. The existing tensor completion methods, which recover the missing elements of a partially known tensor with single view side information, can yield interpretable results for large-scale datasets. However, their limitations up to now are lack of modeling multi-view heterogeneous data and suitably learning the low-rank property of tensor.**

In this study, we fill this gap by developing a novel collective tensor completion method, which tightly fuses multi-view heterogeneous data sources. Our method exploits special common latent structures from the primary tensor and multiple side matrices through coupled tensor-matrix decomposition, in which the common latent structures can compactly represent all the data. In addition, rank estimation of a tensor is a challenging task due to its discrete nature. Instead of approximating the rank by widely used trace norm or nuclear norm, we directly utilize Schatten p-norm on the latent structures to better approximate the rank and to enhance its robustness to noise.

***Index Terms*—Tensor Completion; Heterogeneous Information; Multi-view Learning; Couple Tensor-matrix Factorization; Schatten p-norm**

I. INTRODUCTION

Tensors are multidimensional or N-way generalizations of matrices and have recently gained increasing attention because of their capabilities to express wealthy multi-modal or multi-aspect data [1], [2]. Tensor completion, which aims to recover the missing entries of partially observed tensors by exploring their intrinsic low-rank structures, has enjoyed a broad range of many real-world applications such as computer vision [3], [4], multivariate spatio-temporal analysis [5], [6] and recommender system [7], [5], [8], [9].

Heterogeneous side information: Side information has been proved to be very useful in improving the accuracy for tensor completion [10], [11], [12], [13], [5]. The basic principle is that by incorporating additional features/similarities of entities with a tensor, there can be meaningful correlations among them. These features/similarities, collected from heterogeneous side information, can thus be used to improve the quality of tensor completion. One popular model is the so-called CMTF, which jointly decomposes a tensor with one or more similarity matrices from a single view to improve performance [10]. While the simple matrix/tensor computations and strong mathematical theory behind those coupled tensor-matrix models make them appealing, these methods are inherently limited to incorporate single view of side information [1], [2].

However, data utilized in many emerging tasks are often heterogeneous, extracted from multiple views/sources [14], [15], [16], [17], [18]. For instance, a person can be identified by his/her face (image), voice (audio), or signature (text) with information from diverse sources. In many situations, data consist of a primary tensor and multiple matrices from multi-view side information or metadata. Thus, it is a natural question whether tensor completion models can be generalized to incorporate multi-view heterogeneous data sources.

Low-rank tensor: Low-rank is often a necessary condition to limit the degrees of freedom of high dimensional data in tensor completion [19]. However, the $\mathrm{rank}(\cdot)$ function is unfortunately not convex and it is NP-hard to calculate the rank of a tensor [1]. Arguably, the most widely used rank approximation method is extending matrix trace norm to the tensor case as a convex surrogate of rank minimization [20], [3], [4]. For example, Liu et al. first defined the tensor trace norm as a combination of trace norms of its unfolding matrices [3]. Unfortunately, recent studies show that the tensor trace norm is substantially suboptimal and is not a tight bound of the tensor rank [21], [22]. Several rank variants such as tensor-train rank [23] and tensor tubal rank [24] were also proposed for high-order tensors but often with heavy computational cost.

Contributions: Here we propose a novel method, named TenHet (**Ten**sor Completion with Multi-view **Het**erogeneous Information), to address two major challenges in tensor completion. First, we jointly study the tensor along with multi-view side information to improve our understanding of the underlying relationships among entities in the tensor. For instance, a personalized recommender system [7], [5] allows users to annotate movies with reviews, which forms a 3-way tensor $(user, movie, review)$ as shown in Figure 1.

In addition to the tensor, rich heterogeneous side information of users (e.g., user's historical behavior data, user's location-based social networks), movies (e.g., movie's categories, movie's content in Wikipedia), and reviews (e.g.,

978-1-7281-0858-2/19/$31.00 ©2019 IEEE

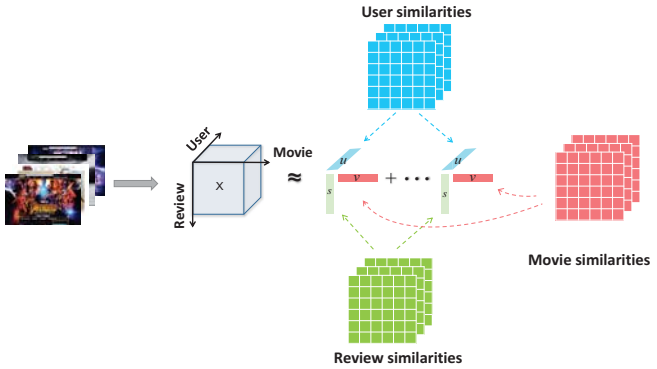


Fig. 1: Overview of TenHet that jointly utilizes the tensor $user \times movie \times review$ with multiple heterogeneous side information of users, movies and reviews, which are represented by matrices.

tags and geotags, memes and comments) are very valuable to improve the recommendation quality [5]. In this scenario, the tensor and multiple matrices are coupled in the "user", "movie" and "review" mode, respectively. Effective use of such rich information can thus allow us to better understand why users evaluate movies with the reviews they post.

Second, we provide a tighter approximation of the tensor rank, by applying Schatten p-norm on each latent matrix factor of a tensor. Our framework is an extension of the matrix completion problem [25], [26], [27]. The Schatten p-norm has been empirically shown to be superior to the trace norm and it requires much fewer observed elements to recover a matrix with a small p [28]. Nevertheless, the Schatten p-norm is non-convex and non-smooth when $0 < p < 1$. We therefore propose an iterative algorithm to solve a smoothed subproblem by approximating the Schatten p-norm at each iteration, which guarantees convergence. The experimental results on both synthetic and real-world datasets show the effectiveness of the proposed tensor completion model.

Our contributions are summarized as follows:

- We investigate the problem of tensor completion with multi-view heterogeneous side information. We formulate this problem as a coupled tensor-matrix optimization problem. The key idea of our formulation is to collectively leverage the primary tensor as well as multi-view side information to infer a latent low-rank representation for each mode of the tensor.
- We propose a tighter estimation of the tensor rank by applying Schatten p-norm on each latent matrix factor of a tensor. We further develop an iterative algorithm to solve a non-convex Schatten p-norm problem.
- We propose an effective algorithm (TenHet) for our optimization problem and further analyze its optimality, convergence, and complexity. The algorithm can be easily scaled up through parallel computing for large datasets.
- We perform extensive experiments on both synthetic and real-world datasets to validate the effectiveness of our proposed algorithm. The results demonstrate TenHet consistently outperforms several state-of-the-art methods.

The rest of the paper is organized as follows. Section II introduces some related work. Section III gives the tensor algebra and task description. Section IV introduces the collective tensor completion with multi-view side information. The learning algorithm and its theoretical analysis are discussed in Sections V. Section VI presents experimental results. Section VII concludes the paper.

II. RELATED WORK

Kolda et al. provided a comprehensive survey on different tensor decomposition methods [1]. Subsequently, tensor completion has garnered increasing attention in a wide range of applications including computer vision, spatio-temporal analysis, and recommender systems [3], [6], [8]. Similar to matrix completion [27], low rank is often a necessary assumption for high dimensional tensor data. However, computing the rank of a tensor is an NP-hard problem [29], [30].

To address this problem, one common approach is to assume that the CP rank of the target tensor is fixed [1], [2]. Nevertheless, it is quite challenging to manually select the rank of a tensor. Another popular approach is to apply the trace norm minimization as a convex surrogate for rank minimization [31], [3], [32], [1]. For example, Liu et al. extended matrix completion to the tensor case by treating the tensor norm as the combination of the trace norm of its unfolding matrices [3]. They further presented an efficient framework by applying the nuclear norm of factor matrices rather than unfolding matrices [4]. However, Cun et al. showed that those approaches may lead to sub-optimal solutions by using trace norm for tensors [22]. Several rank variants such as tensor-train rank [23] and tensor tubal rank [24] were also proposed for high-order tensors but often involving with heavy computation for obtaining these tensor ranks.

Recently, Schatten p-norm was suggested to replace the trace norm for matrix completion since it had empirically shown to be superior to the trace norm [25]. Moreover, Zhang et al. theoretically proved that Schatten p-norm with a small p required much fewer observed elements than the trace norm did [28]. In the case of tensor, Ryota et al. introduced structured Schatten p-norms on the unfolding matrices of a tensor to improve system's performance, but with heavy computational cost [26]. Our work here builds on this line of work but extends it by applying the Schatten p-norm on the latent factors of tensor, rather on the tensor itself.

On the other hand, side information has been proved to be very helpful in improving the accuracy for tensor completion [10], [11], [13], [5], [12], [33]. For example, Acar et al. jointly decomposed a tensor with one or more similarity matrices in one unified framework [10]. Narita et al. applied the graph Laplacians regularization on the factor matrices of a tensor to improve its accuracy [11]. Lamba et al. developed a kernelized probabilistic tensor completion model to effectively deal with the cold-start problem [13]. Zhou et al. proposed a Riemannian tensor model that integrated the tensor and

the side information by overcoming their inconsistency [12]. However, most existing tensor models can only incorporate single view of side information, which may lead to poor performance.

Multi-view learning significantly improved performance of many systems by providing compatible and complementary information from a diversity set of data sources [14], [18], [34]. Zhang et al. introduced a low-rank tensor model to explore the complementary information for multi-view subspace clustering [35]. Nevertheless, their goal was to stack multiple subspace representation matrices as a tensor structure and performed the task of clustering, which is distinct from the task of tensor completion in this work. Our proposed method seamlessly integrates the primary tensor with multi-view side information to infer a latent low-rank representation for each mode of the tensor, which can obtain more interpretable results.

III. BACKGROUND AND PROBLEM SETUP

Notations. Throughout the paper, we denote $\mathbf{X}$ as matrix. A tensor is denoted $\boldsymbol{\mathcal{X}}$. The main notations are listed in Table I. More tensor operators can be found in the literature [1].

A. Tensor Algebra

In this work, we follow the notations introduced by Kolda and Bader [1]. *Tensors* are multidimensional arrays that extend the concept of matrices. The *order* of a tensor is the number of its dimensions, also known as ways or modes. A *fiber* is a vector extracted from a tensor by fixing every index but one. A *slice* is a matrix extracted from a tensor by fixing all but two indices. Note that an N-way tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ reduces to a vector when $N = 1$, and a matrix when $N = 2$. The $(i_1, \ldots, i_N)$-th element of $\boldsymbol{\mathcal{X}}$ is denoted as $\boldsymbol{\mathcal{X}}_{i_1,\ldots,i_N}$.

Matricization, also known as unfolding or flattening, is the process of reordering the elements of a tensor into a matrix. The mode-n matricization of an N-way tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ is represented as $\mathbf{X}_{(n)} \in \mathbb{R}^{I_n \times I_1 \ldots I_{n-1} I_{n+1} \ldots I_N}$ and is arranging the mode-n fibers of the tensor as columns of the long matrix. We further give the definition of CANDECOMP/PARAFAC (CP) decomposition and coupled tensor-matrix factorizations as follows:

Definition 1. (CP Decomposition). Given an N-way tensor $\mathcal{X}$, its CP decomposition is an approximation of N factor matrices $\mathbf{U}_i \in \mathbb{R}^{I_i \times R}$, $i = 1, \ldots, N$, such that:

$$\mathcal{X} \approx [\![\mathbf{U}_1, \mathbf{U}_2, \cdots, \mathbf{U}_N]\!]$$

where $[\![\cdot]\!]$ denotes the Kruskal operator and R is a positive integer denoting an approximation of the rank of tensor $\mathcal{X}$.
Tensor Rank: The rank of a tensor is the smallest number of rank-one tensors, that generates the tensor as their sums, i.e., the smallest R that achieves exact CP decomposition.

Definition 2. (Coupled Tensor-Matrix). If a tensor shares one or more modes with other matrices or other tensors, then they can be coupled with one another [36]. For example, in a recommender system, a triple relationship $user \times movie \times review$ tensor and a $user \times user$ friendship matrix can be coupled since they share the $user$ mode.

TABLE I: Notation.

Symbol	Description
$\|\cdot\|_F$	Frobenius norm
$\|\cdot\|_{S_p}$	Schatten p-norm
$\|\cdot\|_{2,1}$	$l_{2,1}$ norm
$\odot$	Khatri-rao product
$\langle \mathbf{X}, \mathbf{Y} \rangle$	Inner product of two matrices
$\mathbf{X}_{(i)}$	Matricized tensor $\mathcal{X}$ on i-th mode

Definition 3. (Schatten p-norm). The Schatten p-norm ($0 < p < \infty$) of a matrix $\mathbf{U} \in \mathbb{R}^{m \times n}$ is defined as:

$$\|\mathbf{U}\|_{S_p} = \left(\sum_{i=1}^{\min(n,m)} \sigma_i^p \right)^{\frac{1}{p}} = \left(\mathrm{Tr}((\mathbf{U}^T\mathbf{U})^{\frac{p}{2}}) \right)^{\frac{1}{p}}$$

where σ_i is the i-th singular value of $\mathbf{U}$. A widely used Schatten norm is the Schatten 1-norm: $\|\mathbf{U}\|_{S_1} = \sum_{i=1}^{\min(n,m)} \sigma_i$, which is also called trace norm or nuclear norm.

B. Problem Definition

We are often interested in analyzing tensors when additional side information are also available from distinct sources. In the multi-view tensor completion settings, let $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ denote the input data tensor, in which only partial elements are observed. The order N represents the number of dimensions for tensor $\boldsymbol{\mathcal{X}}$. In addition, we have N sets of affinity/similarity matrices from multi-view heterogeneous side information $\{\mathbf{A}_i^{(1)}, \mathbf{A}_i^{(2)}, \cdots, \mathbf{A}_i^{(n_i)}\}_{i=1}^N$, where each set corresponds to one mode of tensor $\boldsymbol{\mathcal{X}}$ and n_i denotes the number of views for mode-i. Our goal is to recover the tensor $\boldsymbol{\mathcal{X}}$ with the guidance from those multi-view side information.

Taking a recommender system as an example (Figure 1), a primary $user \times movie \times review$ tensor represents the triple relationships among users, movies, and reviews. Moreover, several affinity/similarity matrices from multiple heterogeneous information can be constructed to describe the relationships among $user \times user$, $movie \times movie$, and $review \times review$, respectively. The goal is to improve the quality of tensor completion with the help of these multi-view side information.

IV. MULTI-VIEW TENSOR COMPLETION

A. Preliminaries

As an extension of the standard tensor factorization, Acar et al. first proposed a coupled tensor-matrix factorization model to jointly analysis the tensor and matrix [10]. The joint factorization of a third-order tensor $\boldsymbol{\mathcal{X}}$ with a matrix $\mathbf{Y}$ on its first mode can be written as:

$$\min_{\mathbf{A},\mathbf{B},\mathbf{C},\mathbf{V}} \quad \|\boldsymbol{\mathcal{X}} - [\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!]\|_F^2 + \|\mathbf{Y} - \mathbf{A}\mathbf{V}^T\|_F^2 \qquad (1)$$

where $[\![\mathbf{A}, \mathbf{B}, \mathbf{C}]\!]$ denotes the CP decomposition of the tensor $\boldsymbol{\mathcal{X}}$ [1]. The model captures the common underlying latent

structures (e.g., $\mathbf{A}$) from the tensor and matrix simultaneously, which obtains more accurate results than standard tensor factorization model.

Similarly, several variants of coupled tensor-matrix factorization models have been proposed to boost the system performance in many of applications [11], [5].

However, these methods have several limitations: i) Most of existing work can only incorporate single view side information. While in reality, data may have multiple representations (views). Ignoring those rich side information might lead to unsatisfied results. ii) They are highly sensitive to noisy input data. The input tensor $\boldsymbol{\mathcal{X}}$ might be corrupted by noise, which leads to misleading factor matrices $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$. iii) The low-rank structure of tensor is still unclear. The CP decomposition of tensor in Eq. (1) does not contain the low-rank information unless they can obtain the *smallest* number of rank one tensor decomposition, i.e., the CP rank, which is generally NP-hard to compute [1].

B. The Proposed Model

To address the above issues, we propose a collective tensor-matrix completion model that can incorporate multi-view heterogeneous side information with the tensor data. We present our model in terms of its robustness, generalization, and low-rank approximation in details.

1) ***Robust Tensor Completion****:* To be robust against noise, we assume that the observed tensor $\boldsymbol{\mathcal{X}}$ is not perfect but contains two components: a high-quality tensor $\boldsymbol{\mathcal{L}}$ and a noise tensor $\boldsymbol{\mathcal{E}}$. Here we adopt an additive decomposition, i.e., $\boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{L}} + \boldsymbol{\mathcal{E}}$. In addition, the high-quality tensor $\boldsymbol{\mathcal{L}}$ should better reveal the underlying low-rank structure of the tensor data while the noise tensor $\boldsymbol{\mathcal{E}}$ is expected to be sparse. Therefore, rather than performing the CP decomposition of original data tensor $\boldsymbol{\mathcal{X}}$, we decompose the tensor $\boldsymbol{\mathcal{L}}$ as:

$$\begin{aligned} \min_{\boldsymbol{\mathcal{L}},\mathcal{E},\mathbf{U}_i} \quad & \|\boldsymbol{\mathcal{L}} - [\![\mathbf{U}_1, \mathbf{U}_2, \cdots, \mathbf{U}_N]\!]\|_F^2 + \beta\|\boldsymbol{\mathcal{E}}\|_0 + \sum_{i=1}^{N} \lambda_i \cdot Reg(\mathbf{U}_i) \\ s.t., \quad & \boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{L}} + \boldsymbol{\mathcal{E}} \end{aligned} \tag{2}$$

where $\mathbf{U}_i \in \mathbb{R}^{I_i \times R}$ for $i = 1, \ldots, N$ are the factor matrices of the tensor $\boldsymbol{\mathcal{L}}$ and R denotes the dimensionality of $\mathbf{U}_i$. The parameter β controls the sparsity of the noise tensor $\boldsymbol{\mathcal{E}}$ by using l_0-norm. $Reg(\mathbf{U}_i)$ denotes the constraints on each mode of factor matrix $\mathbf{U}_i$, which are guided by prior knowledge from multi-view side information (see the definition in Eq. (3)) and λ_i represents the impact of side information for each mode.

2) ***Model Multi-view Side Information****:* As discussed before, each factor matrix $\mathbf{U}_i$ of $\boldsymbol{\mathcal{L}}$ is knowledgeable with the guidance of its side information. To incorporate those information, we adopt the idea of collective matrix factorization model, due to its flexibility to model complicated dependency structures in the multi-view learning [14], [37], [38]. The key idea is to collectively leverage the primary tensor as well as multi-view side information to infer a shared latent low-rank representation for each mode of the tensor. To be specific, given one factor matrix $\mathbf{U}_i$ in Eq. (2) and its multi-view side information $\{\mathbf{A}_i^{(1)}, \cdots, \mathbf{A}_i^{(n_i)}\}$, the co-training optimization function $Reg(\mathbf{U}_i)$ can be formulated as:

$$\min_{\mathbf{U}_i, \mathbf{G}_i^{(j)}} Reg(\mathbf{U}_i) = \sum_{j=1}^{n_i} (\|\mathbf{A}_i^{(j)} - \mathbf{G}_i^{(j)} \mathbf{G}_i^{(j)^T}\|_F^2 + \|\mathbf{G}_i^{(j)} \mathbf{S}_i^{(j)} - \mathbf{U}_i\|_F^2) \tag{3}$$

where $\mathbf{G}_i^{(j)} \in \mathbb{R}^{I_i \times R}$ is the low-rank representation for j-th view matrix $\mathbf{A}_i^{(j)}$, and $\mathbf{S}_i^{(j)} \in \mathbb{R}^{R \times R}$ can be regarded as a scale matrix for $\mathbf{G}_i^{(j)}$ since different views might not be comparable at the same scale when factorizating them together. We define the scale matrix as: $\mathbf{S}_i^{(j)} = \mathrm{diag}(\sum_\tau \mathbf{G}_i^{(j)}(\tau, 1), \sum_\tau \mathbf{G}_i^{(j)}(\tau, 2), \ldots, \sum_\tau \mathbf{G}_i^{(j)}(\tau, R))$.

The Eq. (3) has an intuitive interpretation: all low-rank factor matrices $\{\mathbf{G}_i^{(1)}, \cdots, \mathbf{G}_i^{(n_i)}\}$ extracted from multi-view side information $\{\mathbf{A}_i^{(1)}, \cdots, \mathbf{A}_i^{(n_i)}\}$ should be consistent with the factor matrix $\mathbf{U}_i$ from the tensor. For example, as shown in Figure 1, the factor matrices learned from multiple $user \times user$ matrices should be close to the factor matrix from the $user \times movie \times review$ tensor because they all represent the same set of users involving in the recommender system. Similarly, we can apply the constraints $Reg(\mathbf{U}_i)$ on all modes of the tensor (see Eq. (5)).

3) ***Tensor Rank Reduction****:* To preserve the low-rank structure, the tensor rank of $\boldsymbol{\mathcal{L}}$ should be considered. However, the tensor rank is not very well defined. As discussed earlier, computing the CP rank of a tensor is an NP-hard problem. Another popular tensor rank is the Tucker rank [1], which defines as $\mathrm{rank}(\boldsymbol{\mathcal{L}}) := (\mathrm{rank}(\mathbf{L}_{(1)}), \cdots, \mathrm{rank}(\mathbf{L}_{(N)}))$, where $\mathbf{L}_{(i)}$ is the mode-i matricization of $\boldsymbol{\mathcal{L}}$. By such rank reduction, the Tucker rank is then computable. For example, the Sum of Nuclear Norm (SNN) define the tensor rank as $\mathrm{rank}(\boldsymbol{\mathcal{L}}) = \sum_i \|\mathbf{L}_{(i)}\|_*$, which is the combination of the trace norm of each matricization of the tensor [3]. Some variants of tensor rank are also proposed for high-order tensors [21], [39]. However, those methods are not efficient for large-scale data, i.e., heavy computation of singular value decomposition for the huge unfolded matrix at each iteration.

To tackle this issue, we turn our attention to the factor matrices of the tensor $\boldsymbol{\mathcal{L}}$. Motivated by the fact that mode-i matricization $\mathbf{L}_{(i)}$ can be represented by its factor matrices, i.e., $\mathbf{L}_{(i)} = \mathbf{U}_i(\mathbf{U}_N \odot \cdots \mathbf{U}_{i+1} \odot \mathbf{U}_{i-1} \odot \cdots \mathbf{U}_1)^T$, where $\odot$ is the Khatri-rao product [1]. By using the fact that $\mathrm{rank}(\mathbf{AB}) \leq \min(\mathrm{rank}(\mathbf{A}), \mathrm{rank}(\mathbf{B}))$ for any two matrices $\mathbf{A}$ and $\mathbf{B}$, we then have a nice upper bound: $\mathrm{rank}(\mathbf{L}_{(i)}) \leq \mathrm{rank}(\mathbf{U}_i)$. Based on the above observations, we can reduce the Tucker rank as: $\mathrm{rank}(\boldsymbol{\mathcal{L}}) := (\mathrm{rank}(\mathbf{U}_1), \cdots, \mathrm{rank}(\mathbf{U}_N))$. Instead of directly computing the rank for the matrix $\mathbf{L}_{(i)}$, we can then approximate the rank for a much smaller matrix $\mathbf{U}_i$. In this work, we provide a novel tensor rank approximation as the combination of Schatten p-norm on each latent factor matrix:

$$\text{rank}(\boldsymbol{\mathcal{L}}) \leftarrow \sum_{i=1}^{N} \|\mathbf{U}_i\|_{S_p} \tag{4}$$

where $\|\cdot\|_{S_p}$ is the matrix Schatten p-norm.

Remark: The Eq. (4) has following benefits: (i) Clearly, the size of $\mathbf{U}_i \in \mathbb{R}^{I_i \times R}$ is far less than mode-i matricization $\mathbf{L}_{(i)} \in \mathbb{R}^{I_i \times \Pi_{j \neq i} I_j}$, which can be computed efficiently and scalable to large dataset. (ii) The Schatten p-norm has been empirically shown to be superior to the trace norm since it requires much fewer observed elements to recover a tensor [26]. (iii) In fact, the trace norm of factor matrices (e.g., SNN) is a special case of Schatten p-norm when $p = 1$. Moreover, when $p \to 0$, $\text{rank}(\mathbf{U}_i) = \|\mathbf{U}_i\|_{S_p}$ is exactly the matrix rank [25], the rank of tensor becomes the sum of the ranks of all factor matrices, i.e., $\text{rank}(\boldsymbol{\mathcal{L}}) \leftarrow \sum_{i=1}^{N} \text{rank}(\mathbf{U}_i)$.

*4) **The Overall TenHet Model**:* By combining Eqs. (2)-(4), we then formulate our model as:

$$\begin{aligned} \min \mathcal{J} = &\|\boldsymbol{\mathcal{L}} - [\![\mathbf{U}_1, \mathbf{U}_2, \cdots, \mathbf{U}_N]\!]\|_F^2 + \beta\|\boldsymbol{\mathcal{E}}\|_{2,1} + \gamma \cdot \sum_{i=1}^{N} \|\mathbf{U}_i\|_{S_p}^p \\ &+ \sum_{i=1}^{N} \lambda_i \cdot \sum_{j=1}^{n_i} (\|\mathbf{A}_i^{(j)} - \mathbf{G}_i^{(j)} \mathbf{G}_i^{(j)^T}\|_F^2 + \|\mathbf{G}_i^{(j)} \mathbf{S}_i^{(j)} - \mathbf{U}_i\|_F^2) \\ &s.t. \quad \boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{L}} + \boldsymbol{\mathcal{E}} \end{aligned} \tag{5}$$

where λ_i is a parameter representing the impact of side information on factor matrix $\mathbf{U}_i$, and γ controls the influence of Schatten p-norm for $\mathbf{U}_i$, $i = 1, \ldots, N$. Moreover, the l_0-norm on tensor $\boldsymbol{\mathcal{E}}$ is nonconvex and challenging to solve, we replace l_0-norm with $l_{2,1}$-norm to characterize the sparsity of noise tensor since the $l_{2,1}$-norm is more robust to noise [40]. Moreover, instead of Schatten p-norm, the p-th power of Schatten p-norm, i.e., $\|\cdot\|_{S_p}^p$, is computed for its easy implementation.

V. OPTIMIZATION ALGORITHM

A. Learning Algorithm

We solve the optimization problem in Eq. (5) by using the alternating direction method of multipliers (ADMM) [41], which is the most widely used solver for tensor completion problems. The objective function in Eq. (5) is not joint convex w.r.t. $\mathbf{U}_i$ for $1 \leq i \leq N$, and it involves a fourth-order term w.r.t. $\mathbf{G}_i^{(j)}$. To address these challenges, we use a variable substitution technique by setting $\mathbf{M}_i = \mathbf{U}_i$ and $\mathbf{Q}_i^{(j)} = \mathbf{G}_i^{(j)}$, and obtain the equivalent form of Eq. (5) as:

$$\begin{aligned} \min \mathcal{J}_1 = &\|\boldsymbol{\mathcal{L}} - [\![\mathbf{U}_1, \mathbf{U}_2, \cdots, \mathbf{U}_N]\!]\|_F^2 + \beta\|\boldsymbol{\mathcal{E}}\|_{2,1} + \gamma \cdot \sum_{i=1}^{N} \|\mathbf{M}_i\|_{S_p}^p \\ &+ \sum_{i=1}^{N} \lambda_i \cdot \sum_{j=1}^{n_i} (\|\mathbf{A}_i^{(j)} - \mathbf{Q}_i^{(j)} \mathbf{G}_i^{(j)^T}\|_F^2 + \|\mathbf{G}_i^{(j)} \mathbf{S}_i^{(j)} - \mathbf{U}_i\|_F^2) \\ &s.t. \quad \boldsymbol{\mathcal{X}} = \boldsymbol{\mathcal{L}} + \boldsymbol{\mathcal{E}} \\ &\quad\ \mathbf{M}_i = \mathbf{U}_i (i = 1, \ldots, N),\ \mathbf{Q}_i^{(j)} = \mathbf{G}_i^{(j)} (j = 1, \ldots, n_i) \end{aligned} \tag{6}$$

The partial augmented Lagrangian function of Eq. (6) is:

$$\begin{aligned} \mathcal{J} = &\mathcal{J}_1 + \sum_{i=1}^{N} \left(\langle \mathbf{Y}_i, \mathbf{M}_i - \mathbf{U}_i \rangle + \frac{\mu}{2} \|\mathbf{M}_i - \mathbf{U}_i\|_F^2 \right) \\ &+ \sum_{i=1}^{N} \sum_{j=1}^{n_i} \left(\langle \mathbf{Z}_i^{(j)}, \mathbf{Q}_i^{(j)} - \mathbf{G}_i^{(j)} \rangle + \frac{\mu}{2} \|\mathbf{Q}_i^{(j)} - \mathbf{G}_i^{(j)}\|_F^2 \right) \\ &+ \langle \boldsymbol{\mathcal{T}}, \boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{L}} - \boldsymbol{\mathcal{E}} \rangle + \frac{\mu}{2} \|\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{L}} - \boldsymbol{\mathcal{E}}\|_F^2 \end{aligned} \tag{7}$$

where $\mathbf{Y}_i$, $\mathbf{Z}_i^{(j)}$, and $\boldsymbol{\mathcal{T}}$ are the Lagrange multipliers and μ is the penalty parameters. $\langle \cdot, \cdot \rangle$ denotes the matrix/tensor inner product. We then successively update each variable until convergence.

Updating $\mathbf{M}_i$: The objective function involving $\mathbf{M}_i$ is:

$$\min \quad \mathcal{J}(\mathbf{M}_i) = \gamma \|\mathbf{M}_i\|_{S_p}^p + \frac{\mu}{2} \|\mathbf{M}_i - \mathbf{U}_i + \mathbf{Y}_i/\mu\|_F^2 \tag{8}$$

By setting the derivative of $\mathcal{J}(\mathbf{M}_i)$ to zero, we have:

$$\mathbf{M}_i(\gamma \mathbf{D}_i + \mu \mathbf{I}) - \mu \mathbf{U}_i + \mathbf{Y}_i = 0$$

Using the above equation, we obtain:

$$\mathbf{M}_i = (\mu \mathbf{U}_i - \mathbf{Y}_i)(\gamma \mathbf{D}_i + \mu \mathbf{I})^{-1} \tag{9}$$

where $\mathbf{D}_i = p(\mathbf{M}_i^T \mathbf{M}_i)^{\frac{p-2}{2}}$. Note that Eq. (9) is not a closed-form solution since $\mathbf{D}_i$ is dependent on $\mathbf{M}_i$. Nevertheless, if $\mathbf{D}_i$ is known, then the matrix $\mathbf{M}_i$ can be computed by Eq. (9). Inspired by this observation, we use an alternative updating algorithm to solve $\mathbf{M}_i$. Function solverM() is described in Algorithm 1. At each iteration, $\mathbf{M}_i$ is updated with current $\mathbf{D}_i$, and then $\mathbf{D}_i$ is updated with the current $\mathbf{M}_i$. Theorem 1 shows that algorithm solverM() will converge when $0 < p \leq 2$, which covers the range we are interested.

Theorem 1. *When $0 < p \leq 2$, the solverM() in Algorithm 1 will monotonically decrease the objective function in Eq. (8) at each iteration until convergence.*

The proof details can be found in Appendix.

Algorithm 1: solverM()

Input: $\mathbf{M}_i$, $\mathbf{U}_i$, $\mathbf{Y}_i$, μ, γ and $p \in (0, 2]$
Output: $\mathbf{M}_i^*$: the solution for Eq. (10) .

Initialize $\mathbf{M}_i^{(0)} = \mathbf{M}_i$, $\mathbf{D}_i^{(0)} = p(\mathbf{M}_i^T \mathbf{M}_i)^{\frac{p-2}{2}}$, $t = 0$;
repeat
  Update $\mathbf{M}_i^{(t+1)} \leftarrow (\mu \mathbf{U}_i - \mathbf{Y}_i)(\gamma \mathbf{D}_i^{(t)} + \mu \mathbf{I})^{-1}$;
  Update $\mathbf{D}_i^{(t+1)} \leftarrow p(\mathbf{M}_i^{(t+1)^T} \mathbf{M}_i^{(t+1)})^{\frac{p-2}{2}}$;
  $t \leftarrow t + 1$;
until *Convergence*
return $\mathbf{M}_i^* = \mathbf{M}_i^{(t+1)}$

Updating $\mathbf{U}_i$: To compute $\mathbf{U}_i$, we can minimize the following objective function:

$$\begin{aligned} \min \mathcal{J}(\mathbf{U}_i) = &\|\mathbf{U}_i \mathbf{B}_i^T - \mathbf{L}_{(i)}\|_F^2 + \lambda_i \sum_{j=1}^{n_i} \|\mathbf{G}_i^{(j)} \mathbf{S}_i^{(j)} - \mathbf{U}_i\|_F^2 \\ &+ \frac{\mu}{2} \|\mathbf{U}_i - \mathbf{M}_i - \mathbf{Y}_i/\mu\|_F^2 \end{aligned}$$

where $\mathbf{B}_i = (\mathbf{U}_N \odot \cdots \odot \mathbf{U}_{i+1} \odot \mathbf{U}_{i-1} \odot \cdots \odot \mathbf{U}_1)$. $\mathbf{L}_{(i)}$ denotes the mode-i matricization of $\boldsymbol{\mathcal{L}}$. $\mathbf{U}_i$ can then be updated by:

$$\begin{aligned}\mathbf{U}_i =&(2\mathbf{L}_{(i)}\mathbf{B}_i + 2\lambda_i \sum_{j=1}^{n_i} \mathbf{G}_i^{(j)}\mathbf{S}_i^{(j)} + \mu\mathbf{M}_i + \mathbf{Y}_i) \\ &(2\mathbf{B}_i^T\mathbf{B}_i + (2\lambda_i n_i + \mu)\mathbf{I})^{-1}\end{aligned} \tag{10}$$

Updating $\mathbf{G}_i^{(j)}$: The solution for $\mathbf{G}_i^{(j)}$ can be obtained by optimizing $\mathcal{J}(\mathbf{G}_i^{(j)})$ and its solution is:

$$\begin{aligned}\mathbf{G}_i^{(j)} =&(2(\mathbf{A}_i^{(j)})^T\mathbf{Q}_i^{(j)} + 2\mathbf{U}_i(\mathbf{S}_i^{(j)})^T + \mu\mathbf{Q}_i^{(j)} + \mathbf{Z}_i^{(j)}) \\ &(2(\mathbf{Q}_i^{(j)})^T\mathbf{Q}_i^{(j)} + 2\mathbf{S}_i^{(j)}(\mathbf{S}_i^{(j)})^T + \mu\mathbf{I})^{-1}\end{aligned} \tag{11}$$

Similarly, we can solve the auxiliary variable $\mathbf{Q}_i^{(j)}$ as:

$$\mathbf{Q}_i^{(j)} =(2\mathbf{A}_i^{(j)}\mathbf{G}_i^{(j)} + \mu\mathbf{G}_i^{(j)} - \mathbf{Z}_i^{(j)})(2(\mathbf{G}_i^{(j)})^T\mathbf{G}_i^{(j)} + \mu\mathbf{I})^{-1} \tag{12}$$

Updating $\boldsymbol{\mathcal{L}}$: The optimal solution for $\boldsymbol{\mathcal{L}}$ is:

$$\boldsymbol{\mathcal{L}} = (2 \cdot [\![\mathbf{U}_1, \cdots, \mathbf{U}_N]\!] + \mu\boldsymbol{\mathcal{X}} - \mu\boldsymbol{\mathcal{E}} + \boldsymbol{\mathcal{T}})/(\mu + 2) \tag{13}$$

Updating $\boldsymbol{\mathcal{E}}$: The noise tensor $\boldsymbol{\mathcal{E}}$ is obtained by solving:

$$\min_{\boldsymbol{\mathcal{E}}} \quad \beta\|\boldsymbol{\mathcal{E}}\|_{2,1} + \frac{\mu}{2}\|\boldsymbol{\mathcal{E}} - \boldsymbol{\mathcal{X}} + \boldsymbol{\mathcal{L}} - \boldsymbol{\mathcal{T}}/\mu\|_F^2$$

Since the $l_{2,1}$-norm of tensor $\boldsymbol{\mathcal{E}}$ can be defined on its mode-N matricization, i.e., $\|\boldsymbol{\mathcal{E}}\|_{2,1} = \|\mathbf{E}_{(N)}\|_{2,1}$, above objective function can be transformed into the matrix form:

$$\min_{\boldsymbol{\mathcal{E}}} \quad \beta\|\mathbf{E}_{(N)}\|_{2,1} + \frac{\mu}{2}\|\mathbf{E}_{(N)} - \mathbf{F}\|_F^2$$

where $\mathbf{F} = \mathbf{X}_{(N)} - \mathbf{L}_{(N)} + \mathbf{T}_{(N)}/\mu$. According to [40], The close-solution for $\mathbf{E}_{(N)}$ is:

$$\mathbf{E}_{(N)}(:,i) = \begin{cases} \frac{\|\mathbf{F}(:,i)\|_2 - \beta/\mu}{\|\mathbf{F}(:,i)\|_2}\mathbf{F}(:,i), & \text{if } \|\mathbf{F}(:,i)\|_2 > \beta/\mu \\ 0, & \text{otherwise} \end{cases} \tag{14}$$

where $\mathbf{E}_{(N)}(:,i)$ is the i-th column of the matrix $\mathbf{E}_{(N)}$. After computing $\mathbf{E}_{(N)}$, we then transform it back to tensor $\boldsymbol{\mathcal{E}}$.

Updating $\mathbf{Y}_i$, $\mathbf{Z}_i^{(j)}$ and $\boldsymbol{\mathcal{T}}$: The Lagrange multipliers are updated by using the gradient ascent as:

$$\begin{aligned}\mathbf{Y}_i &\leftarrow \mathbf{Y}_i + \mu(\mathbf{M}_i - \mathbf{U}_i) \\ \mathbf{Z}_i^{(j)} &\leftarrow \mathbf{Z}_i^{(j)} + \mu(\mathbf{Q}_i^{(j)} - \mathbf{G}_i^{(j)}) \\ \boldsymbol{\mathcal{T}} &\leftarrow \boldsymbol{\mathcal{T}} + \mu(\boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{L}} - \boldsymbol{\mathcal{E}})\end{aligned} \tag{15}$$

The overall procedure of TenHet with initializations of all variables is summarized in Algorithm 2.

Algorithm 2: TenHet

Input: $\boldsymbol{\mathcal{X}}$, R, $\{\mathbf{A}_i^{(1)}, \cdots, \mathbf{A}_i^{(n_i)}\}_{i=1}^N$, β, γ, $\{\lambda_i\}$, p, tol.
Initialize $\mathbf{U}_i^{(0)}$, $(\mathbf{G}_i^{(j)})^{(0)}$ and $\boldsymbol{\mathcal{E}}^{(0)}$ randomly.
Set $\boldsymbol{\mathcal{L}}^{(0)} = \boldsymbol{\mathcal{X}}$, $\mathbf{M}_i^{(0)} = \mathbf{U}_i^{(0)}$, $(\mathbf{Q}_i^{(j)})^{(0)} = (\mathbf{G}_i^{(j)})^{(0)}$.
$\mathbf{Y}_i^{(0)} = \mathbf{0}$, $(\mathbf{Z}_i^{(j)})^{(0)} = \mathbf{0}$, $\boldsymbol{\mathcal{T}}^{(0)} = \mathbf{0}$, $\mu = 10^{-6}$.
repeat
  for $i \leftarrow 1$ ***to*** N **do**
    Update $\mathbf{M}_i^{(t+1)}$ by solverM() in Algorithm 1
    Update $\mathbf{U}_i^{(t+1)}$ by Eq. (10)
    Update $\mathbf{Y}_i^{(t+1)}$ by Eq. (15)
    /* Update each mode of tensor */
    for $j \leftarrow 1$ ***to*** n_i **do**
      Update $(\mathbf{G}_i^{(j)})^{(t+1)}$ and $(\mathbf{Q}_i^{(j)})^{(t+1)}$ by Eq. (11) and Eq. (12)
      Update $(\mathbf{Z}_i^{(j)})^{(t+1)}$ in Eq. (15)
    end
  end
  Update $\boldsymbol{\mathcal{L}}^{(t+1)}$ and $\boldsymbol{\mathcal{E}}^{(t+1)}$ by Eq. (13) and Eq. (14)
  Update $\boldsymbol{\mathcal{T}}^{(t+1)}$ by Eq. (15)
until $\frac{\|\boldsymbol{\mathcal{L}}^{(t+1)} - \boldsymbol{\mathcal{L}}^{(t)}\|_F}{\|\boldsymbol{\mathcal{L}}^{(t)}\|_F} \leq tol$
return $\boldsymbol{\mathcal{L}}$

B. Convergence and Complexity Analysis

There is no theoretical guarantee for the global convergence of ADMM for the non-convex problems or convex problems with multiple block variables. Here, we show the convergence property of ADMM under mild conditions in Theorem 2. The proof sketch of Theorem 2 is similar to [42] and it is omitted here due to page limitation.

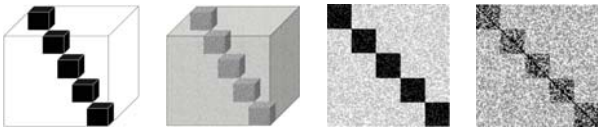
Theorem 2. *(Weak Convergence Condition) Let* $\boldsymbol{\mathcal{W}} := (\mathbf{M}_i, \mathbf{U}_i, \mathbf{G}_i^{(j)}, \mathbf{Q}_i^{(j)}, \boldsymbol{\mathcal{L}}, \boldsymbol{\mathcal{E}}, \mathbf{Y}_i, \mathbf{Z}_i^{(j)}, \boldsymbol{\mathcal{T}})$ *and* $\{\boldsymbol{\mathcal{W}}^{(k)}\}_{k=1}^{\infty}$ *be the sequence generated by Algorithm 2. Assume that* $\{\boldsymbol{\mathcal{W}}\}_{k=1}^{\infty}$ *is bounded and* $\boldsymbol{\mathcal{W}}^{(k+1)} - \boldsymbol{\mathcal{W}}^{(k)} \to 0$. *Then any accumulation point of* $\{\boldsymbol{\mathcal{W}}^{(k)}\}_{k=1}^{\infty}$ *satisfies the KKT condition of objective function in Eq. (6)*

The time complexity is mainly dominated in the steps of updating $\mathbf{M}_i$ and $\mathbf{U}_i$. According to solverM(), updating $\mathbf{M}_i$ requires $\mathcal{O}(T(I_i^2R + R^3))$, where T is the number of iterations for solverM() to converge. Moreover, the complexity of updating $\mathbf{U}_i$ is $\mathcal{O}(R\sum_{n=1}^{N}\Pi_{m\neq i}^{N} I_m + R\Pi_{m=1}^{N} I_m + n_i I_i R^2)$. In practice, $R \ll I_i$, the order of tensor N and the number of views n_i can be regarded as small constants. Therefore, the actual time complexity can be denoted as $\mathcal{O}(R\sum_{n=1}^{N}\Pi_{m\neq i}^{N} I_m + R\Pi_{m=1}^{N} I_m + TI_i^2R)$.

VI. EXPERIMENTS

In this section, we evaluate TenHet on both synthetic and benchmark datasets and compare it against the following baselines:

- TRPCA: a robust tensor principal component analysis model [19].
- FaLRTC: which estimates the low rank structure by imposing the trace norm on its unfolding matrices [3].
- TNCP: which imposes the trace norm on each factor matrix [4].
- TFAI: which recovers the tensor by incorporating a single view of auxiliary information [11].
- CMTF: which constructs common latent factors shared by a tensor and single view of side information by using coupled matrix-tensor factorization [10].
- CGSI: which exploits the side information to improve the accuracy of Riemannian tensor completion [12].
- TenHetOne: a degraded version of TenHet with single view side information.



(a) (b) (c) (d)

Fig. 2: (a) True clusters in tensor, (b) Tensor with noise, (c) Side similarity matrix with low noise and (d) with high noise. All with five balanced disjoint clusters.

A. Synthetic Dataset

The synthetic dataset we used is similar to the dataset in the task of matrix completion with multi-view side information [43]. In this scenario, the underlying structures for both tensor and multi-view side information are consistent with each other. More specifically, the true cluster structure is first embedded into the primary tensor, and noise is then added such that the true structures should not be easily obtained by mining the tensor alone. We first generate a noisy block-wise diagonal 3-way tensor $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{r \times r \times r}$ with t disjoint balanced clusters on the diagonal. The entries on the block-wise diagonal of $\boldsymbol{\mathcal{X}}$ are randomly sampled from a uniform distribution within $[1, 2]$ and the off-diagonal are within $[0, 1]$. Figure 2a shows an example of a block-wise diagonal tensor with five balanced disjoint true clusters and Figure 2b is the input tensor $\boldsymbol{\mathcal{X}}$ with noise. For multi-view side information, we generate multiple similarity matrices for *each mode* of the tensor by first incorporating the cluster information of the tensor (e.g., clusters are also on the diagonal) and then adding different levels of noise [43]. Different similarity matrices have different amounts of noise. To be specific, we first generate a true block-wise diagonal similarity matrix S^i_{ture} for mode-i of the tensor such that the block-wise diagonal elements are all one and off-diagonal are all zero. We then generate n_i similarity matrices for mode-i by adding different level of noise. Figures 2c and 2d show two examples of similarity matrices with low and high level of noise, but maintaining the clusters the same as the tensor.

Implementation Details: We set the size of tensor $r = 500$, the number of clusters $t = 50$ and the number of views $n_i = 3$ for each mode of the tensor. In the experiments, we randomly remove elements from $\boldsymbol{\mathcal{X}}$ with various missing ratios ranging from 20% to 80%. All the algorithms will try to recover the whole tensor. To evaluate the quality of different methods, we adopt the relative square error (RSE) as the evaluation metric, which is defined as RSE $:= \|\boldsymbol{\mathcal{L}} - \boldsymbol{\mathcal{X}}\|_F / \|\boldsymbol{\mathcal{X}}\|_F$ [4], [3].

Note that the models TRPCA, FaLRTC, TNCP do not consider any side information. TFAI, CMTF, CGSI and TenHetOne can only incorporate single view side information. For these methods, the side matrices with the least noise are used for each mode. For all methods, the dimensionality of all factor matrices is set to $R = 40$. For comparison methods, their parameters are tuned for optimal performance. For TenHetOne and TenHet, we fix the regularization parameters as $\beta = 10^{-3}$, $\gamma = 0.01$, $\lambda_i = 0.1$, and $p = 0.5$ (Schatten p-norm). The impact of parameters will be discussed later. For each method, the experiments are repeated ten times independently and the average results are reported.

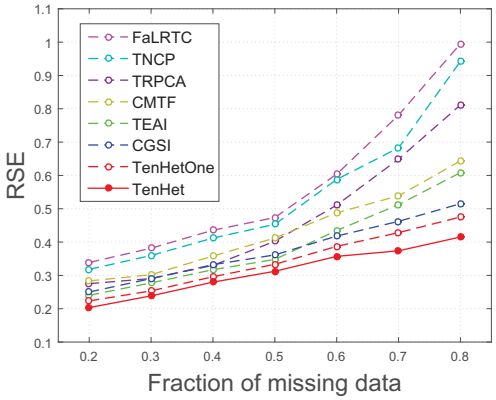
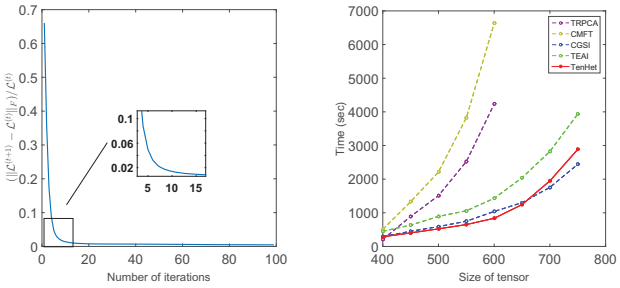


Fig. 3: RSE of different tensor completion methods.

Experimental Results: Figure 3 shows the average RSE of different methods with various missing ratios. Clearly, TenHet achieves the best performance across a wide range of missing ratios. We also have the following observations. First, most of tensor completion models perform well when the fraction of missing data is less than 40%. With a larger fraction of missing data, the models that consider side information (i.e., CMTF, TFAI, CGSI, TenHetOne and TenHet) generally perform better than the rest approaches. Even with a significant of missing data in the tensor (up to 80%), with the guidance of side information, the intrinsic tensor structures are still preserved by these approaches. Second, TenHet outperforms TenHetOne, indicating the contributions of adding more side information. Third, in terms of low rank constraints, TNCP performs better than FaLRTC, indicating the superiority of trace norm on the factor matrices rather on the unfolded matrices of the tensor. Having this analogy, our TenHet, imposing the Schatten p-norm on factor matrices of the tensor, can thus capture more accurate intrinsic structures in the data. Also, TenHet is robust to noise due to its decomposition of a high-quality tensor data.

We further study the convergence behavior of TenHet on the synthetic data with 50% missing. Instead of checking the overall value of the objective function in Eq. (6), it is more interesting to see how well the tensor $\boldsymbol{\mathcal{L}}$ is recovered [4]. Figure 4(a) shows the relative change of $\boldsymbol{\mathcal{L}}^{(t)}$ between two consecutive steps with respect to the number of iterations. It clearly shows that the error decreases rapidly during initial iterations. Usually, less than 30 iterations are sufficient for convergence.

Scalability: The scalability of TenHet is also studied. We vary the size of the tensor r from 400 to 750, and set the number of clusters $t = 0.1r$. The experiments are performed on a 2.40GHz machine over ten independent runs and the performance is measured in running time (seconds). The



(a) (b)

Fig. 4: (a) Convergence of TenHet. (b) Running time of different tensor-based models.

running time of FaLRTC and TNCP is not included here because of their low prediction accuracy. Figure 4(b) reports the running time on different tensor completion models. TenHet and CGSI achieve similar performance while both of them are much faster than the other approaches. At the same time, as shown earlier, by incorporating more side information, TenHet can obtain more accurate results than CGSI. Also note that TenHet is more efficient (and more accurate) than the other two approaches that can incorporate side information, namely TFAI and CMTF.

B. Real Benchmark Datasets

We further evaluate the proposed method on two real-world datasets: Last.FM[1] and DrugBank[2]. For recommendation dataset Last.fm, similar to [9], [44], we can obtain a $user \times item \times tag$ tensor, in which $\boldsymbol{\mathcal{X}}_{ijk} = 1$ means that user i has tagged an item j with the tag k; $\boldsymbol{\mathcal{X}}_{ijk} = 0$, otherwise. Totally, we have $219,702$ observed elements involving $2,917$ users, $1,853$ items, and $2,045$ tags. For side information, two affinity matrices for users are computed. One is from anonymized user's social network, i.e., user-user interaction matrix, the other is from the user-item rating matrix, i.e., each user has an item vector profiles, an affinity matrix can be measured by the self-tune Gaussian method [45]. In Last.fm dataset, the items are artists, two affinity matrices can be also constructed from side sources. One is from the user vector profiles from the rating matrix and the other is from items' semantic context (e.g., artists' context) in the Wikipedia. The similarity between two web documents can be then measured by using the text mining package gensim[3]. For tags, only one affinity matrix is computed by their semantic similarity [44].

For DrugBank dataset, the objective is to identify unknown drug-target-disease relationships in computational drug discovery. We first download the drug-disease associations from the literature [46]. For each drug-disease pair, their target proteins can be collected from the public DrugBank database. To obtain a dense tensor, we only focus on those targets that interact with at least two drugs. Finally, we can obtain a $drug \times target \times disease$ tensor with size of $593 \times 501 \times 313$ and $20,778$ known triple relationships of drug-target-disease. For side information, we collect several well-studied similarity matrices for drugs, targets, and diseases from literatures [46], [15], [43], [47], respectively. Specifically, we have two drug-drug similarities based on drug chemical structures and side-effects; two target-target similarities based on their protein sequences and Gene Ontology (GO) annotations; two disease-disease similarities based on disease phenotypes and Human Phenotype Ontology (HPO). For tensor completion methods that can only incorporate a single view side information, we choose the similarities defined based on drug chemical structures, target protein sequences, and disease phenotypes since all of which have long been considered valuable domain knowledge in drug discovery. The statistics about Last.FM and DrugBank dataset are shown in Table II[4].

TABLE II: Basic dataset statistics.

Dataset	# Users	# Items	# Tag	# Triples
Last.FM	2,917	1,853	2,045	219,702
DrugBank	593	501	313	20,778

We adopt a widely used evaluation protocol for tag recommendation [9], [44]. Briefly, for each user i, we randomly select one post from the dataset, i.e., a user-item pair (i, j) that user i has provided tags for item j. We then remove all triples (i, j, k) from the observed data. The post (i, j) will form a test set S_{test}. The remaining observations are the training set $S_{train} = S / S_{test}$. We then predict a ranked top-n list for each of the removed post in S_{test}. The evaluation metric used in this work is the F1-measure for the top-n list, where $n = 5, 10, 15$, and the definition of F1-measure score can be found in [9].

Table III shows the F1-measure scores of all the methods. TenHet consistently outperforms the competing methods on most of the experiments. For instance, TenHet performs better than TRPCA and CGSI with an average improvement of 9.9% and 7.2% in terms of F1-measure scores, respectively. The trend is very similar to the results on the synthetic data. Approaches with side information perform better than approaches without side information, and TenHet with multi-view information performs better than approaches with single side information. An intuitive explanation is that models without or with single view side information can be sensitive to the noise or bias contained in the input tensor. Utilizing multi-view data can help improve the performance by exploiting the compatible and complementary information across multi-view data sources.

Parameter studies: There are three kinds of regularization parameters (β, γ and $\{\lambda_i\}_{i=1}^N$) and one pre-defined parameter p (Schatten p-norm) in the proposed model. For third-order tensor with size $n_1 \times n_2 \times n_3$, since we matricize the noise tensor in the third mode in Eq. (14), we can empirically set $\beta = 1/\sqrt{\max(n_1, n_2)n_3}$ as suggested by [19]. We next study

[1]https://grouplens.org/datasets/hetrec-2011/
[2]https://www.drugbank.ca/
[3]https://radimrehurek.com/gensim/

[4]Table II regards the triple $(drug, target, disease)$ as $(user, item, tag)$.

TABLE III: Experimental results on real benchmark datasets.

Method	Top-5		Top-10		Top-15	
	Last.FM	DrugBank	Last.FM	DrugBank	Last.FM	DrugBank
TRPCA	0.453	0.634	0.407	0.415	0.319	0.421
FaLRTC	0.439	0.512	0.387	0.367	0.297	0.384
TNCP	0.443	0.514	0.395	0.386	0.302	0.397
TEAI	0.461	0.597	0.412	0.416	0.326	0.439
CMTF	0.453	0.657	0.408	0.411	0.321	0.434
CGSI	0.466	0.660	0.411	0.419	0.334	0.432
TenHetOne	0.476	0.669	0.414	0.501	0.323	0.441
TenHet	**0.513**	**0.673**	**0.424**	**0.514**	**0.336**	**0.448**

the influence the rest parameters.

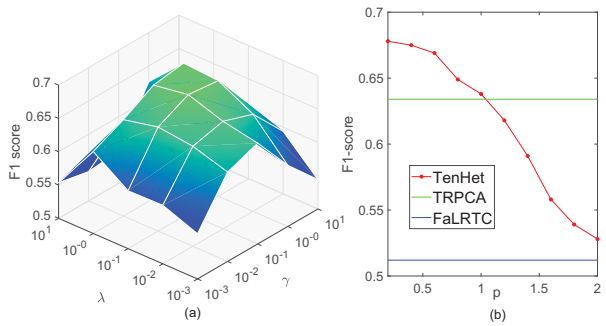


Fig. 5: Parameter sensitivity of our model.

1) *The impact of γ and λ_i:* Recall that γ controls the influence of Schatten p-norm and λ_i controls the contributions of side information for mode-i of the tensor. If the side information for mode-i is noisy, then a relative small λ_i is preferred. In this study, we simply set $\lambda_1 = \lambda_2 = \lambda_3 = \lambda$. We first set $p = 0.5$ and vary values of λ and γ from $\{0.001, 0.01, 0.1, 1, 10\}$. We then estimate the F1 score for the Top-5 performance on DrugBank dataset. As shown from Figure 5(a), our method is relatively stable, and slightly better when λ_i and γ are within $\{0.01, 0.1, 1\}$.

2) *The impact of Schatten p-norm:* Parameter p reflects the impact of tensor rank reduction. To better study its influence, we set all $\lambda_i = 0$ (i.e., exclude all multi-view side information), then compare it with two tensor completion models: TRPCA and FaLRTC. The p varies from 0.2 to 2 and γ is simply set to 0.1. Figure 5(b) shows TenHet has better performance with smaller p, consistent with the fact that Schatten p-norm can approximate the rank function when $p \to 0$. Although Schatten p-norm is not convex and is not smooth when $0 < p < 1$, it still constantly performances better than TRPCA. When $1 \leq p \leq 2$, the performance of TenHet is better than FaLRTC, but not as good as TRPCA.

VII. CONCLUSION

In this paper, we have proposed a general framework to perform tensor completion with multiple heterogeneous side data sources. The proposed method, TenHet, integrates the tensor with multi-view side information simultaneously, which is able to find accurate and interpretable low-rank structures in the data. By formulating multi-view tensor completion as an optimization problem, we propose an effective algorithm based on ADMM to solve the optimization problem, and prove its optimality, correctness and scalability. Extensive experiments on synthetic and real-world datasets demonstrate the effectiveness of our approach.

Acknowledgments: This work has been supported in part by NSF CCF1815139 and by an allocation of computing time from the Ohio Supercomputer Center.

REFERENCES

[1] T. G. Kolda and B. W. Bader, "Tensor decompositions and applications," *SIAM review*, vol. 51, no. 3, pp. 455–500, 2009.

[2] E. E. Papalexakis, C. Faloutsos, and N. D. Sidiropoulos, "Tensors for data mining and data fusion: Models, applications, and scalable algorithms," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 8, no. 2, p. 16, 2017.

[3] J. Liu, P. Musialski, P. Wonka, and J. Ye, "Tensor completion for estimating missing values in visual data," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 208–220, 2013.

[4] Y. Liu, F. Shang, L. Jiao, J. Cheng, and H. Cheng, "Trace norm regularized candecomp/parafac decomposition with missing data," *IEEE transactions on cybernetics*, vol. 45, no. 11, pp. 2437–2448, 2015.

[5] P. Bhargava, T. Phan, J. Zhou, and J. Lee, "Who, what, when, and where: Multi-dimensional collaborative recommendations using tensor factorization on sparse user-generated data," in *WWW*, 2015.

[6] K. Takeuchi, H. Kashima, and N. Ueda, "Autoregressive tensor factorization for spatio-temporal predictions," in *ICDM*, 2017.

[7] H. Yin, H. Chen, X. Sun, H. Wang, Y. Wang, and Q. V. H. Nguyen, "Sptf: A scalable probabilistic tensor factorization model for semantic-aware behavior prediction," in *ICDM*, 2017.

[8] H. Chen and J. Li, "Adversarial tensor factorization for context-aware recommendation," in *RecSys*, 2019.

[9] S. Rendle, L. Balby Marinho, A. Nanopoulos, and L. Schmidt-Thieme, "Learning optimal ranking with tensor factorization for tag recommendation," in *KDD*, 2009.

[10] E. Acar, T. G. Kolda, and D. M. Dunlavy, "All-at-once optimization for coupled matrix and tensor factorizations," *arXiv preprint arXiv:1105.3422*, 2011.

[11] A. Narita, K. Hayashi, R. Tomioka, and H. Kashima, "Tensor factorization using auxiliary information," *Data Mining and Knowledge Discovery*, vol. 25, 2012.

[12] T. Zhou, H. Qian, Z. Shen, C. Zhang, and C. Xu, "Tensor completion with side information: A riemannian manifold approach." in *IJCAI*, 2017.

[13] H. Lamba, V. Nagarajan, K. Shin, and N. Shajarisales, "Incorporating side information in tensor completion," in *WWW*, 2016.

[14] J. Liu, C. Wang, J. Gao, and J. Han, "Multi-view clustering via joint nonnegative matrix factorization," in *SDM*, 2013.

[15] H. Chen and J. Li, "A flexible and robust multi-source learning algorithm for drug repositioning," in *BCB*, 2017.

[16] L. He, C.-T. Lu, Y. Chen, J. Zhang, L. Shen, S. Y. Philip, and F. Wang, "A self-organizing tensor architecture for multi-view clustering," in *ICDM*, 2018.

[17] H. Chen and J. Li, "Drugcom: Synergistic discovery of drug combinations using tensor decomposition," in *ICDM*, 2018.
[18] C. Chen, H. Tong, L. Xie, L. Ying, and Q. He, "Fascinate: Fast cross-layer dependency inference on multi-layered networks," in *KDD*, 2016.
[19] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, and S. Yan, "Tensor robust principal component analysis: Exact recovery of corrupted low-rank tensors via convex optimization," in *CVPR*, 2016.
[20] Q. Shi, H. Lu, and Y.-m. Cheung, "Tensor rank estimation and completion via cp-based nuclear norm," in *CIKM*, 2017.
[21] B. Romera-Paredes and M. Pontil, "A new convex relaxation for tensor completion," in *NeurIPS*, 2013.
[22] C. Mu, B. Huang, J. Wright, and D. Goldfarb, "Square deal: Lower bounds and improved relaxations for tensor recovery," in *ICML*, 2014.
[23] I. V. Oseledets, "Tensor-train decomposition," *SIAM Journal on Scientific Computing*, vol. 33, no. 5, pp. 2295–2317, 2011.
[24] Z. Zhang and S. Aeron, "Exact tensor completion using t-svd," *IEEE Transactions on Signal Processing*, vol. 65, no. 6, pp. 1511–1526, 2016.
[25] F. Nie, H. Huang, and C. H. Ding, "Low-rank matrix recovery via efficient schatten p-norm minimization." in *AAAI*, 2012.
[26] R. Tomioka and T. Suzuki, "Convex tensor decomposition via structured schatten norm regularization," in *NeurIPS*, 2013.
[27] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational mathematics*, vol. 9, no. 6, p. 717, 2009.
[28] M. Zhang, Z.-H. Huang, and Y. Zhang, "Restricted p-isometry properties of nonconvex matrix recovery," *IEEE Transactions on Information Theory*, vol. 59, no. 7, pp. 4316–4323, 2013.
[29] C. J. Hillar and L.-H. Lim, "Most tensor problems are np-hard," *Journal of the ACM (JACM)*, vol. 60, no. 6, p. 45, 2013.
[30] V. Gupta, T. Koren, and Y. Singer, "Shampoo: Preconditioned stochastic tensor optimization," in *ICML*, 2018.
[31] S. Gandy, B. Recht, and I. Yamada, "Tensor completion and low-n-rank tensor recovery via convex optimization," *Inverse Problems*, vol. 27, no. 2, p. 025010, 2011.
[32] M. Signoretto, L. De Lathauwer, and J. A. Suykens, "Nuclear norms for tensors and their use for convex multilinear estimation," *Linear Algebra and Its Applications*, 2010.
[33] H. Chen and J. Li, "Modeling relational drug-target-disease interactions via tensor factorization with multiple web sources," in *WWW*, 2019.
[34] H. Gao, F. Nie, X. Li, and H. Huang, "Multi-view subspace clustering," in *ICCV*, 2015.
[35] C. Zhang, H. Fu, S. Liu, G. Liu, and X. Cao, "Low-rank tensor constrained multiview subspace clustering," in *ICCV*, 2015.
[36] M. R. de Araujo, P. M. P. Ribeiro, and C. Faloutsos, "Tensorcast: Forecasting with context using coupled tensors (best paper award)," in *ICDM*, 2017.
[37] A. P. Singh and G. J. Gordon, "Relational learning via collective matrix factorization," in *KDD*, 2008.
[38] H. Chen and J. Li, "Learning multiple similarities of users and items in recommender systems," in *ICDM*, 2017.
[39] Q. Yao, J. T.-Y. Kwok, and B. Han, "Efficient nonconvex regularized tensor completion with structure-aware proximal iterations," in *ICML*, 2019.
[40] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 171–184, 2012.
[41] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.
[42] D. Goldfarb and Z. Qin, "Robust low-rank tensor recovery: Models and algorithms," *SIAM Journal on Matrix Analysis and Applications*, vol. 35, no. 1, pp. 225–253, 2014.
[43] X. Zheng, H. Ding, H. Mamitsuka, and S. Zhu, "Collaborative matrix factorization with multiple similarities for predicting drug-target interactions," in *KDD*, 2013.
[44] P. Symeonidis, "Clusthosvd: Item recommendation by combining semantically enhanced tag clustering with tensor hosvd," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 9, pp. 1240–1251, 2016.
[45] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *NeurIPS*, 2005.
[46] A. Gottlieb, G. Y. Stein, E. Ruppin, and R. Sharan, "Predict: a method for inferring novel drug indications with application to personalized medicine," *Molecular systems biology*, vol. 7, no. 1, p. 496, 2011.
[47] M. P. Menden, D. Wang, M. J. Mason, B. Szalai, K. C. Bulusu, Y. Guan, T. Yu, J. Kang, M. Jeon, R. Wolfinger *et al.*, "Community assessment to advance computational prediction of cancer drug combinations in a pharmacogenomic screen," *Nature communications*, vol. 10, no. 1, p. 2674, 2019.

APPENDIX

we first introduce a matrix inequality in Lemma 1.

Lemma 1. *For any two positive definite matrices $\mathbf{A}_1, \mathbf{A}_2 \in \mathbb{R}^{n\times n}$, the following inequality holds when $0 < p \leq 2$ [25]:*

$$Tr(\mathbf{A}_1^{\frac{p}{2}}) - \frac{p}{2}Tr(\mathbf{A}_1\mathbf{A}_2^{\frac{p-2}{2}}) \leq Tr(\mathbf{A}_2) - \frac{p}{2}Tr(\mathbf{A}_2\mathbf{A}_2^{\frac{p-2}{2}})$$

we next prove Theorem 1 in details.

proof of Theorem 1

Proof. It can be easily verified that Eq. (9) is the solution to the following equivalent problem:

$$\min_{\mathbf{M}_i} \quad \frac{\mu}{2\gamma}\|\mathbf{M}_i - \mathbf{O}\|_F^2 + \frac{1}{2}Tr(\mathbf{M}_i^T\mathbf{M}_i\mathbf{D}_i) \tag{16}$$

where $\mathbf{O} = \mathbf{U}_i - \mathbf{Y}_i/\mu$. Thus in the t iteration:

$$\mathbf{M}_i^{(t+1)} = \arg\min_{\mathbf{M}_i} \frac{\mu}{2\gamma}\|\mathbf{M}_i - \mathbf{O}\|_F^2 + \frac{1}{2}Tr(\mathbf{M}_i^T\mathbf{M}_i\mathbf{D}_i^{(t)})$$

which indicates that:

$$\begin{aligned}&\frac{\mu}{2\gamma}\|\mathbf{M}_i^{(t+1)} - \mathbf{O}\|_F^2 + \frac{1}{2}Tr((\mathbf{M}_i^{(t+1)})^T\mathbf{M}_i^{(t+1)}\mathbf{D}_i^{(t)})\\ &\leq \frac{\mu}{2\gamma}\|\mathbf{M}_i^{(t)} - \mathbf{O}\|_F^2 + \frac{1}{2}Tr((\mathbf{M}_i^{(t)})^T\mathbf{M}_i^{(t)}\mathbf{D}_i^{(t)})\end{aligned}$$

By substituting $\mathbf{D}_i^{(t)} = p[(\mathbf{M}_i^{(t)})^T\mathbf{M}_i^{(t)}]^{\frac{p-2}{2}}$, we have:

$$\begin{aligned}&\frac{\mu}{2\gamma}\|\mathbf{M}_i^{(t+1)} - \mathbf{O}\|_F^2 + \frac{p}{2}Tr((\mathbf{M}_i^{(t+1)})^T\mathbf{M}_i^{(t+1)}[(\mathbf{M}_i^{(t)})^T\mathbf{M}_i^{(t)}]^{\frac{p-2}{2}})\\ &\leq \frac{\mu}{2\gamma}\|\mathbf{M}_i^{(t)} - \mathbf{O}\|_F^2 + \frac{p}{2}Tr((\mathbf{M}_i^{(t)})^T\mathbf{M}_i^{(t)}[(\mathbf{M}_i^{(t)})^T\mathbf{M}_i^{(t)}]^{\frac{p-2}{2}})\end{aligned} \tag{17}$$

On the other hand, according to Lemma 1, we have:

$$\begin{aligned}&Tr([(\mathbf{M}_i^{(t+1)})^T\mathbf{M}_i^{(t+1)}]^{\frac{p}{2}}) - \frac{p}{2}Tr((\mathbf{M}_i^{(t+1)})^T\mathbf{M}_i^{(t+1)}[(\mathbf{M}_i^{(t)})^T\mathbf{M}_i^{(t)}]^{\frac{p-2}{2}})\\ &\leq Tr([(\mathbf{M}_i^{(t)})^T\mathbf{M}_i^{(t)}]^{\frac{p}{2}}) - \frac{p}{2}Tr((\mathbf{M}_i^{(t)})^T\mathbf{M}_i^{(t)}[(\mathbf{M}_i^{(t)})^T\mathbf{M}_i^{(t)}]^{\frac{p-2}{2}})\end{aligned} \tag{18}$$

Combining the inequalities (17) and (18), we have:

$$\begin{aligned}&Tr([(\mathbf{M}_i^{(t+1)})^T\mathbf{M}_i^{(t+1)}]^{\frac{p}{2}}) + \frac{\mu}{2\gamma}\|\mathbf{M}_i^{(t+1)} - \mathbf{O}\|_F^2\\ \leq &Tr([(\mathbf{M}_i^{(t)})^T\mathbf{M}_i^{(t)}]^{\frac{p}{2}}) + \frac{\mu}{2\gamma}\|\mathbf{M}_i^{(t)} - \mathbf{O}\|_F^2\end{aligned}$$

Multiplying both sides of the above inequality with γ, we have $\mathcal{J}(\mathbf{M}_i^{(t+1)}) \leq \mathcal{J}(\mathbf{M}_i^{(t)})$, where $\mathcal{J}(\mathbf{M}_i)$ is defined by Eq. (8). □