Coupled Graph Tensor Factorization

Ahmed S. Zamzam, Vassilis N. Ioannidis, and Nicholas D. Sidiropoulos

***Abstract*—Factorization of a single matrix or tensor has been used widely to reveal interpretable factors or predict missing data. However, in many cases side information may be available, such as social network activities and user demographic data together with Netflix data. In these situations, coupled matrix tensor factorization (CMTF) can be employed to account for additional sources of information. When the side information comes in the form of item-correlation matrices of certain modes, existing CMTF algorithms do not apply. Instead, a novel approach to model the correlation matrices is proposed here, using symmetric nonnegative matrix factorization. The multiple sources of information are fused by fitting outer-product models for the tensor and the correlation matrices in a coupled manner. The proposed model has the potential to overcome practical challenges, such as missing slabs from the tensor and/or missing rows/columns from the correlation matrices.**

***Index Terms*—Tensor factorization, matrix factorization, parafac model, missing entries**

I. Introduction

A wide variety of data is nowadays analyzed using matrix and tensor factorization techniques. Examples range from social networks, recommender systems, biomedical applications, computer vision and communication networks. The datasets in these applications contain a large number of intentionally or unintentionally missing entries, as a result of limitations in the data collection / aggregation process, privacy concerns, unwillingness to share information and other factors. The available data is often organized in matrix or tensor form, and imputation of missing entries is a task of paramount importance that arises naturally in these settings. For example, online retailers cannot acquire ratings for a product from every possible customer, hence rating prediction is of interest for marketing purposes. Moreover it is common that side information is available, and analyzing data from multiple sources jointly can enable more accurate predictions.

In social networks and recommender systems, the objective is to unveil hidden links between the data in order to recommend new friends or items. Item or subject correlation matrices are often available (or can be estimated), and these correlations reflect an underlying graph structure [1]. Additionally, higher order tensors capture the multi-relational nature of the data, and hence, are suitable for modeling these datasets. One can exploit the "regularity" present among data in order to recover the missing entries. This property is modeled by assuming a low rank representation of the tensor [2].

A.S. Zamzam, V.N. Ioannidis, and N.D. Sidiropoulos are with the Department of Electrical and Computer Engineering, Univ. of Minnesota, Minneapolis, MN 55455, USA. Emails: {AhmedZ, ioann006, nikos}@umn.edu.
A.S. Zamzam and N.D. Sidiropoulos were partially supported by NSF CIF-1525194, ECCS-1231504.

In this paper, a novel algorithm that performs coupled factorization of multi-way tensors and associated item correlation matrices is proposed in order to impute missing entries in both the matrices and the tensors. The general problem formulation allows to address the realistic but challenging scenario where imperfect and/or incomplete side information is available. Also, the proposed model can handle the so called *cold start* problem where an entire slab is missing from the tensor.

Throughout this paper, lower and upper boldface letters are used to denote vectors and matrices, respectively. The tensors are denoted by underlined upper case boldface symbols. For any general matrix $\mathbf{X}$, $\mathbf{X^T}$, $\mathbf{X^{-1}}$, $\mathrm{Tr}(\mathbf{X})$, and $\mathrm{diag}(\mathbf{X})$ denote the transpose, the inverse, the trace, and the diagonal of $\mathbf{X}$, respectively. The Khatri-Rao and Hadamard products of two matrices $\mathbf{X}$ and $\mathbf{Y}$ are denoted by $\mathbf{X} \odot \mathbf{Y}$ and $\mathbf{X} * \mathbf{Y}$, respectively. The operator $\mathrm{vec}(\cdot)$ denotes the vectorization of its matrix argument.

This paper is organized as follows. Section II introduces the problem formulation studied in this paper. In section III, related work from the literature is reviewed. Section IV introduces the proposed idea and summarizes the developed algorithm, and Section V demonstrates the effectiveness of the proposed algorithm. Section VI summarizes our conclusions.

II. Problem Formulation

Consider a tensor $\underline{\boldsymbol{X}}$ of order N and size $I_1 \times I_2 \times \cdots \times I_N$ for integers $I_1, I_2, \cdots, I_N$. An entry of $\underline{\boldsymbol{X}}$ is denoted by $\underline{\boldsymbol{X}}(i_1, i_2, \cdots, i_N)$, where each index i_k refers to the kth-mode of the tensor. The mode-n matricization of a tensor $\underline{\boldsymbol{X}} \in \mathbb{R}^{I_1, I_2, \cdots, I_N}$ is denoted by $\underline{\boldsymbol{X}}_{(n)}$ and arranges the mode-n one-dimensional fibers to be the columns of the resulting matrix; see [3] for details. In addition, consider a set of correlation (similarity) matrices $\{\mathbf{G}_m\}_{m=1}^{M}$, where $M \leq N$. The (i,j)-th entry of $\mathbf{G}_m$ reflects the similarity between the i-th and j-th data items corresponding to the m-th mode of the tensor. The problem of finding the low-rank factors of the tensor shared with the correlation matrices that approximate the given entries of the data is considered. The set of known entries in the tensor $\underline{\boldsymbol{X}}$ and the correlation matrix $\boldsymbol{G}_m$ are specified by $\underline{\mathbf{W}} \in \{0,1\}^{I_1 \times I_2 \times \cdots \times I_N}$ and $W_m \in \{0,1\}^{I_m \times I_m}$, respectively.

Without loss of generality, in this paper we focus on the 3-way tensor case $\underline{\mathcal{X}} \in \mathbb{R}_+^{I \times J \times K}$. Our goal is to recover the missing values in the tensor and the similarity matrices. This can be viewed as trying to estimate missing values based on available data and assumed relations between the modes. In the matrix case, for example, consider the movie rating data where each user has watched only few movies and some movies will be recommended to him/her based on earlier ratings. In addition, the recommendation system can benefit from available data about user interaction in the social network. So, filling the incomplete tensor and similarity matrices based on the available data is the goal of this work.

III. Related Work

Coupled matrix tensor factorization (CMTF) has been widely studied in the literature [1], [4]–[6] where the matrices are assumed to share some factors with the tensor. However, the particular case of using correlation matrices as side information has not been considered. For instance, consider the tensor $\underline{\boldsymbol{X}}$ which can be factored using low-rank PARAFAC model [7] that can be written in vectorized form as follows

$$\text{vec}(\underline{\boldsymbol{X}}) \cong (\boldsymbol{C} \odot \boldsymbol{B} \odot \boldsymbol{A})\, \mathbf{1} \tag{1}$$

where $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ are the factors of the tensor with rank R, and $\mathbf{1}$ is the vector of all ones with appropriate dimension. In addition, consider another matrix $\boldsymbol{Z}$ which provides additional information for one of the modes. Assuming the matrix contains the items of the first mode of the tensor, the low-rank matrix model can be written as

$$\boldsymbol{Z} \cong \boldsymbol{A}\boldsymbol{D}^T \tag{2}$$

where $\boldsymbol{A}$ and $\boldsymbol{D}$ are the low-rank factors of the matrix.
Consequently, the latent factors can be found by solving the following optimization problem

$$\begin{aligned} \min_{\boldsymbol{A},\boldsymbol{B},\boldsymbol{C},\boldsymbol{D}} \quad & \| \text{vec}(\underline{\boldsymbol{X}}) - (\boldsymbol{C} \odot \boldsymbol{B} \odot \boldsymbol{A})\mathbf{1} \|_F^2 \\ & + \| \boldsymbol{Z} - \boldsymbol{A}\boldsymbol{D}^T \|_F^2 \,. \end{aligned} \tag{3}$$

It is clear that this optimization problem is not convex. It is easy to see that it subsumes the tensor factorization problem, which is NP-hard in general. Therefore, alternating least squares (ALS) can be employed to extract the factors. In addition, accelerated solutions for this problem were proposed in [4] which can make the iterative method much faster by exploiting the sparsity of the data. The case of missing values in both the matrices and the tensor was handled in [1], [4].
In some cases the side information matrices represent similarity between the items of one mode of the tensor. Similarity matrices have been used before coupled with tensor factorization [2]. Assuming that the underlying low-rank factors follow a simplified distribution allows for incorporation of the correlation information in a Bayesian framework. For example, if the low-rank factors are assumed to be drawn from a Gaussian distribution, then the *Maximum Likelihood Estimator* of the latent factor given the correlation matrices of low-rank factors can be found by solving the following optimization problem

$$\begin{aligned} \min_{\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}} \quad & \| \text{vec}(\underline{\boldsymbol{X}}) - (\boldsymbol{C} \odot \boldsymbol{B} \odot \boldsymbol{A})\mathbf{1} \|_F^2 + \\ & \text{Tr}(\boldsymbol{A}{\boldsymbol{G}_1}^{-1}\boldsymbol{A}^T) + \text{Tr}(\boldsymbol{B}{\boldsymbol{G}_2}^{-1}\boldsymbol{B}^T) + \text{Tr}(\boldsymbol{C}{\boldsymbol{G}_3}^{-1}\boldsymbol{C}^T). \end{aligned} \tag{4}$$

where $\boldsymbol{G}_1$, $\boldsymbol{G}_2$ and $\boldsymbol{G}_3$ are the correlation matrices between the items in mode-1, mode-2 and mode-3, respectively. The drawback of this approach is that it assumes that the full similarity matrices are available (so that the inverses can be computed) which is not the case in many applications. For example, in movie rating data, not all users will provide their social network data from which one can infer their similarity. Also, assuming a specific (perhaps over-simplified) distribution (e.g., Gaussian or Laplacian) limits the applicability of the model since the latent factors do not follow these models in real situations. In many scenarios, interpretability can be restored if additional conditions are imposed, such as non-negativity.
The fact that correlation matrices are symmetric positive semidefinite necessitates a corresponding parametrization. Also, a non-negativity assumption on the latent factor is often plausible, and it promotes uniqueness and therefore interpretability of the factors. Hence, the Symmetric Nonnegative Matrix Factorization (SymNMF) was introduced as a suitable model for this type of matrices [8], [9]. The factorization of the graph similarity matrices using the SymNMF model was shown to be successful in terms of clustering the items [9]. Also, the interpretability of the revealed factors was demonstrated in [8]. Hence, the following optimization problem is solved to obtain the nonnegative low-rank factors of the correlation matrix $\boldsymbol{G}_{\boldsymbol{A}}$

$$\min_{\boldsymbol{A} \geq \mathbf{0}} \quad \| \boldsymbol{G}_{\boldsymbol{A}} - \boldsymbol{A}\boldsymbol{A}^T \|_F^2 \,. \tag{5}$$

Yet, this model was not yet been used when the similarity matrix is coupled with another dataset, such as a tensor.
For tensors, [10] proposed augmenting the least-squares model-fitting cost with a penalty term that takes into account $S(i, j) \geq 0$, a measure of similarity between customer i and customer j. Albeit interesting, this approach does not capture negative correlations/(dis-)similarities.

IV. Proposed formulation

The correlation matrices are thought of as describing underlying graphs, which justifies the name of the proposed algorithm: *Coupled Graph Tensor Factorization* (CGTF). Due to the efficacy of the nonnegative matrix factorization in providing identifiable factors and serving as a soft clustering tool, symmetric nonnegative factorization [9], [11] will be considered to model the graphs. In addition, the proposed model assumes the tensor factors to be scaled versions of the graph factors. Based on this idea, the tensor and the correlation matrices can be modeled as follows

$$\underline{\mathbf{X}} \cong [[\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]] \tag{6}$$

$$\boldsymbol{G}_A \cong \boldsymbol{A}\boldsymbol{D}_A\boldsymbol{A}^T \tag{7}$$

$$\boldsymbol{G}_B \cong \boldsymbol{B}\boldsymbol{D}_B\boldsymbol{B}^T \tag{8}$$

$$\boldsymbol{G}_C \cong \boldsymbol{C}\boldsymbol{D}_C\boldsymbol{C}^T \tag{9}$$

where $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ represent the low-rank factors corresponding to three modes of the tensor, and $[[\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}]]$ is the outer product of these matrices resulting in a tensor. The matrices $\boldsymbol{D}_A$, $\boldsymbol{D}_B$, and $\boldsymbol{D}_C$ represent nonnegative diagonal matrices that are included to account for the scaling difference between the factors of the tensor and the correlation matrices. Adding the diagonal loading matrices endows the model with the ability to adjust the relative weighting differences between the tensor and the side information matrices. This is motivated by the nature of various side information matrices, e.g., social networks and cosine similarity matrices. In order to reveal the latent factors shared between the tensor and side information, the diagonal scaling matrices are used. In order to find the latent factors that approximate the tensor and the correlation matrices in the given elements, the following optimization problem has to be solved.

$$\begin{aligned}\min_{\boldsymbol{A},\boldsymbol{B},\boldsymbol{C},\boldsymbol{D}_A,\boldsymbol{D}_B,\boldsymbol{D}_C\geq 0} \quad & \|\underline{\boldsymbol{W}} * (\underline{\boldsymbol{X}} - [[\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}]])\|_F^2 \\ & +\mu_1\|W_A * (\boldsymbol{G}_A - \boldsymbol{A}\boldsymbol{D}_A\boldsymbol{A}^T)\|_F^2 \\ & +\mu_2\|W_B * (\boldsymbol{G}_B - \boldsymbol{B}\boldsymbol{D}_B\boldsymbol{B}^T)\|_F^2 \\ & +\mu_3\|W_C * (\boldsymbol{G}_C - \boldsymbol{C}\boldsymbol{D}_C\boldsymbol{C}^T)\|_F^2\end{aligned} \tag{10}$$

Clearly, the optimization problem in (10) is not convex, and is in fact NP-hard. Aiming for a tractable and scalable approximation, we propose to use the projected gradient method to estimate the latent factors $\boldsymbol{A}$, $\boldsymbol{B}$ and $\boldsymbol{C}$ and the diagonal loadings $\boldsymbol{D}_A$, $\boldsymbol{D}_B$ and $\boldsymbol{D}_C$. The algorithm is compactly described in the following table where $\boldsymbol{k} = \operatorname{vec}(\boldsymbol{K})$, $\boldsymbol{d}_a$=diag$(\boldsymbol{D}_A)$, $f(\boldsymbol{x})$ is the objective function in (10), $iter$ is the maximum number of iterations, ϵ is the threshold used as a stopping criterion and R is the rank of the $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ factors.

Algorithm 1 Coupled Graph Tensor Factorization

Require: $\underline{\boldsymbol{W}}$, $\underline{\boldsymbol{X}}$, $W_{\boldsymbol{A}}$, $\boldsymbol{G}_A$, W_B, $\boldsymbol{G}_B$, W_C, $\boldsymbol{G}_C$, μ_1, μ_2, μ_3, Θ, ϵ, R

Initialize:
$\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ using (5)
$\boldsymbol{x}^{(0)} \leftarrow [\boldsymbol{a}^T \;\; \boldsymbol{b}^T \;\; \boldsymbol{c}^T \;\; \boldsymbol{d}_a^T \;\; \boldsymbol{d}_b^T \;\; \boldsymbol{d}_c^T]^T$

for $r = 1, \ldots, \Theta$ **do**
  $\boldsymbol{x}^{(r)} \leftarrow [\boldsymbol{x}^{(r-1)} - s_r \nabla f(\boldsymbol{x}^{(r-1)})]^+$
  $d \leftarrow f(\boldsymbol{x}^{(r-1)}) - f(\boldsymbol{x}^r)$
  if $d \leq \epsilon$ **then**
    break
  end if
end for
Return: $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}_A$, $\boldsymbol{D}_B$, $\boldsymbol{D}_C$

The step size s_r can be selected using backtracking (Armijo) rule [12]. CGTF stops either if the difference of two consecutive evaluations of the objective function is less than ϵ or the maximum number of iterations Θ is reached.

Remark 1: Define $\underline{\boldsymbol{Y}}$ to be the fitted tensor using the estimated factors which is given by

$$\underline{\boldsymbol{Y}} = [[\boldsymbol{A},\boldsymbol{B},\boldsymbol{C}]]. \tag{11}$$

Therefore, the gradients of cost function in (10) with respect to the variables $\boldsymbol{A}$ and $\boldsymbol{D}_A$ are given by

$$\begin{aligned}\frac{\partial f}{\partial \boldsymbol{a}} = & \, 2\operatorname{vec}\Bigg(\Big[\underline{\boldsymbol{W}}_{(1)} * (\underline{\boldsymbol{Y}}_{(1)} - \underline{\boldsymbol{X}}_{(1)})\Big]^T (\boldsymbol{B}\odot\boldsymbol{C}) \\ & +2\mu_1\Big[\boldsymbol{W}_A * (\boldsymbol{A}\boldsymbol{D}_A\boldsymbol{A}^T - \boldsymbol{G}_A)\Big]\boldsymbol{A}\boldsymbol{D}_A\Bigg),\end{aligned} \tag{12}$$

$$\frac{\partial f}{\partial \boldsymbol{d}_A} = 2\Big((\boldsymbol{A}\odot\boldsymbol{A})^T \operatorname{vec}\big(\boldsymbol{W}_A * (\boldsymbol{A}\boldsymbol{D}_A\boldsymbol{A} - \boldsymbol{G}_A)\big)\Big). \tag{13}$$

In practice, the correlation matrices or graphs have missing entries which generaly complicates solving (5). Furthermore, entries may be systematically (as opposed to randomly) missing. For example, in social networks, some people prefer not to reveal their connections at all. In these scenarios, the steps summarized in the following algorithm are followed in order to initialize the latent factors $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$ while the diagonal loadings are initialized by ones.

Algorithm 2 Solving (5) using SymNMF algorithm [8]

Require: $W_{\boldsymbol{A}}$, $\boldsymbol{G}_A$, ϵ, R

Construct $\tilde{\boldsymbol{G}}_A$ by removing the missing items from the graph
$\tilde{\boldsymbol{G}}_A = \boldsymbol{U}_G\boldsymbol{\Lambda}_G\boldsymbol{U}_G \qquad \Longleftarrow$ Reduced EVD
$\boldsymbol{B} \leftarrow \boldsymbol{U}_G\boldsymbol{\Lambda}_G^{1/2}$
$\boldsymbol{Q} \leftarrow \boldsymbol{I}$
repeat
  $\boldsymbol{T} \leftarrow \max(\boldsymbol{0}, \boldsymbol{B}\boldsymbol{Q})$
  $\boldsymbol{T}^T\boldsymbol{B} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V} \qquad \Longleftarrow$ SVD
  $\boldsymbol{Q} \leftarrow \boldsymbol{V}\boldsymbol{U}^T$
until $|\operatorname{Tr}(\boldsymbol{T}^T(\boldsymbol{T} - \boldsymbol{B}\boldsymbol{Q}))| > \epsilon$
$\tilde{\boldsymbol{A}} \leftarrow \boldsymbol{T}$
Set the low-rank representation of the missing item to be the average of the known items
Return: $\boldsymbol{A}$

V. Numerical Tests

A. Synthetic Data

A synthetic tensor $\underline{\boldsymbol{X}}$ of size $10 \times 25 \times 25$ and rank 6 was generated according to the PARAFAC model. The true factors $\boldsymbol{A}, \boldsymbol{B}$ and $\boldsymbol{C}$ were drawn from Gaussian i.i.d distribution with mean 0 and variance 4. In addition, three correlation matrices $\boldsymbol{C}_A, \boldsymbol{C}_B$ and $\boldsymbol{C}_C$ were generated to represent the correlation between the points in the low dimension representation. The entries of $\underline{\boldsymbol{X}}$ were corrupted with random i.i.d Gaussian noise. One entire slab of the second mode of the tensor is removed and reserved to evaluate the prediction performance. In addition, the correlation information of thirty percent of the items in each mode was removed and reserved. The values of the regularization parameters were set to 1. The proposed algorithm was used to recover the missing correlation information and the missing slab from $\underline{\boldsymbol{X}}$. Fig. 1 depicts the normalized mean squared error *NMSE* which is calculated as $\sum_{i_3=1}^{I_3} \|\hat{\underline{\boldsymbol{X}}}(:,:,i_3) - \underline{\boldsymbol{X}}(:,:,i_3)\|_F^2 / \sum_{i_3=1}^{I_3} \|\underline{\boldsymbol{X}}(:,:,i_3)\|_F^2$ against the signal to noise ratio of the tensor (SNR).

B. Real Data

The dataset used to assess performance of the proposed algorithm was first presented in [13]. It comprises a three-way tensor indicating the frequency of a certain user conducting a specific activity at a certain location. It contains information about 164 users, 168 locations and 5 activities. A binary tensor $\underline{\boldsymbol{X}}$ is constructed to represent the links between the users, the locations and the activities. In other words, $\underline{\boldsymbol{X}}_{(i,j,k)}$ equals 1 if user i visited location j and performed activity k;

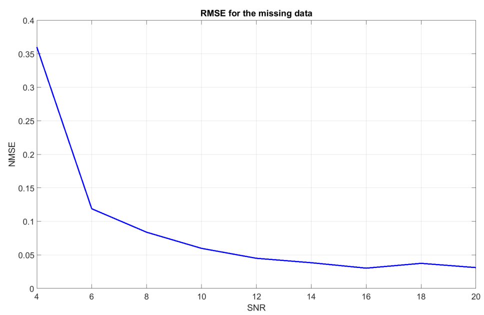

Fig. 1: NMSE of the missing entries versus the SNR.

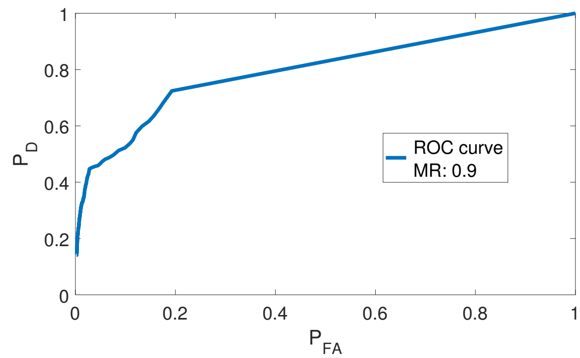

Fig. 2: ROC curve with Missing Ratio (MR) 0.9.

otherwise, it is 0. Additionally, similarity matrices between the users and the activities are provided. The similarity value between two locations is defined by the inner product between the corresponding feature vectors. The dataset is missing social network information for 28 users, and feature vectors for 32 locations. In the experiments, the number of the columns in $\mathbf{A}, \mathbf{B}$ and $\mathbf{C}$ is set to 4. The factors are initialized by the Symmetric Nonnegative Matrix Factorization (SymNMF) factors of the corresponding similarity matrices. Since symmetric NMF is unique under certain conditions, the initialization is likely to be a very good one [11]. Also, the values of the regularization parameters are set to 0.01.

We evaluate the performance of our novel algorithm by drawing the receiver operating characteristic (ROC) curve in Fig. 2 with 90% of the tensor data missing. We define the probability of detection P_D as the ratio between the links from the initial tensor that were discovered over the number of the actual links. The probability of false alarm P_{FA} is calculated as the number of links that appear only in the estimated tensor divided by the number of the zeros in $\underline{\boldsymbol{X}}$. In addition, the NMSE is presented for different percentages of missing data comparing with other models. The Nonnegative Tensor Factorization (NTF) model (implemented in [14]) was used to factor the tensor data only (without the side information), as well as the CANDECOMP/PARAFAC Weighted OPTimization (CP_WOPT) algorithm [6]. For the Coupled Matrix Tensor Factorization (CMTF) algorithm, we used the side information as matrices which share only one factor with the tensor while the other factor is not shared with the tensor. All the algorithms were initialized using our proposed initialization which enhances the performance of all of them. The NMSE of the estimated tensor $\underline{\hat{\boldsymbol{X}}}$ is compared across the various models and algorithms. Table I shows that the proposed algorithm outperforms other models used to factor the tensor. This can be attributed to the fact that our algorithm benefits the most from the available side information.

Missing	NTF	CP_WOPT	CMTF	CGTF
40%	2.0815	0.9517	0.975	**0.6151**
50%	–	0.9574	1.001	**0.7188**
70%	–	0.9825	1.002	**0.7848**

TABLE I: NMSE for different ratios of missing data.[1]

VI. Conclusions and Future Work

To judiciously incorporate prior information represented by a graph, this paper introduced a novel factorization model, CGTF, to impute missing entries of the tensors and the graphs. To the best of our knowledge, it is the only model that considers symmetric nonnegative factorization of matrices sharing latent factors with a tensor. Also, the proposed method is the only one among the ones considered here that can overcome the so-called *cold-start* problem, where the tensor has missing slabs or the similarity matrices are not complete. The novel algorithm makes accurate prediction of the missing values and can be used in many real world settings, especially in recommendation systems. Our future research agenda includes refinement of the optimization algorithm, replace LS with K-L divergence (10) for integer valued data, and use of Kernel matrices instead of correlation matrices which capture more general relations.

References

[1] B. Ermiş, E. Acar, and A. T. Cemgil, “Link prediction in heterogeneous data via generalized coupled tensor factorization,” *Data Mining and Knowledge Discovery*, vol. 29, no. 1, pp. 203–236, 2015.

[2] J. A. Bazerque, G. Mateos, and G. Giannakis, “Rank regularization and bayesian inference for tensor completion and extrapolation,” *IEEE Trans. Signal Process.*, vol. 61, no. 22, pp. 5689–5703, 2013.

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

[4] E. E. Papalexakis, C. Faloutsos, T. M. Mitchell, P. P. Talukdar, N. D. Sidiropoulos, and B. Murphy, “Turbo-SMT: Accelerating Coupled Sparse Matrix-Tensor Factorizations by 200x.” in *SDM*. SIAM, 2014, pp. 118–126.

[5] 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.

[6] E. Acar, D. M. Dunlavy, T. G. Kolda, and M. Mørup, “Scalable Tensor Factorizations with Missing Data.” in *SDM*. SIAM, 2010, pp. 701–712.

[7] R. A. Harshman, “Foundations of the parafac procedure: Models and conditions for an" explanatory" multi-modal factor analysis,” 1970.

[8] K. Huang, N. D. Sidiropoulos, and A. Swami, “Non-negative matrix factorization revisited: Uniqueness and algorithm for symmetric decomposition,” *IEEE Transactions on Signal Processing*, vol. 62, no. 1, pp. 211–224, 2014.

[9] D. Kuang, H. Park, and C. H. Ding, “Symmetric Nonnegative Matrix Factorization for Graph Clustering.” in *SDM*, vol. 12. SIAM, 2012, pp. 106–117.

[1] The NTF model does not provide meaningful results for high percentages of missing values.

[10] A. Narita, K. Hayashi, R. Tomioka, and H. Kashima, "Tensor factorization using auxiliary information," *Data Mining and Knowledge Discovery*, vol. 25, no. 2, pp. 298–324, 2012.
[11] K. Huang and N. Sidiropoulos, "Putting nonnegative matrix factorization to the test: a tutorial derivation of pertinent Cramer-Rao bounds and performance benchmarking," *IEEE Signal Processing Magazine*, vol. 31, no. 3, pp. 76–86, April 2014.
[12] D. Bertsekas, "Nonlinear programming," 1999.
[13] V. W. Zheng, B. Cao, Y. Zheng, X. Xie, and Q. Yang, "Collaborative filtering meets mobile recommendation: A user-centered approach," in *AAAI*, 2010.
[14] C. A. Andersson and R. Bro, "The N-way toolbox for MATLAB," *Chemometrics and Intelligent Laboratory Systems*, vol. 52, no. 1, pp. 1–4, 2000.