Factorization of Multiple Tensors for Supervised Feature Extraction

Wei Liu

Advanced Analytics Institute, Faculty of Engineering and Information Technology,
University of Technology Sydney, Sydney, Australia
Wei.Liu@uts.edu.au

Abstract. Tensors are effective representations for complex and time-varying networks. The factorization of a tensor provides a high-quality low-rank compact basis for each dimension of the tensor, which facilitates the interpretation of important structures of the represented data. Many existing tensor factorization (TF) methods assume there is one tensor that needs to be decomposed to low-rank factors. However in practice, data are usually generated from different time periods or by different class labels, which are represented by a sequence of multiple tensors associated with different labels. When one needs to analyse and compare multiple tensors, existing TF methods are unsuitable for discovering all potentially useful patterns, as they usually fail to discover either common or unique factors among the tensors: *1)* if each tensor is factorized separately, the factor matrices will fail to explicitly capture the common information shared by different tensors, and *2)* if tensors are concatenated together to form a larger "overall" tensor and then factorize this concatenated tensor, the intrinsic unique subspaces that are specific to each tensor will be lost. The cause of such an issue is mainly from the fact that existing tensor factorization methods handle data observations in an *unsupervised* way, considering only features but not labels of the data. To tackle this problem, we design a novel probabilistic tensor factorization model that takes both features and class labels of tensors into account, and produces informative *common and unique factors of all tensors* simultaneously. Experiment results on feature extraction in classification problems demonstrate the effectiveness of the factors discovered by our method.

Keywords: Feature Extraction, Tensor Factorization, Supervised Learning

1 Introduction

In this paper we study the problem of probabilistically factorizing a sequence of multiple tensors for feature extraction from multi-mode tensor data.. Various types of tensor factorization methods have been proposed in the literature, including Tucker decomposition [10], CP [3] (also known as PARAFAC), non-negative tensor factorization [11], and probabilistic tensor factorization [12]. These methods and their later variants can be considered as higher-order generalizations of matrix factorizations. Most of these existing methods are restricted to decomposing a single instance of a tensor object in an unsupervised manner. This raises the question of what strategy should be used when dealing with multiple tensor objects which are associated with class labels. Given a

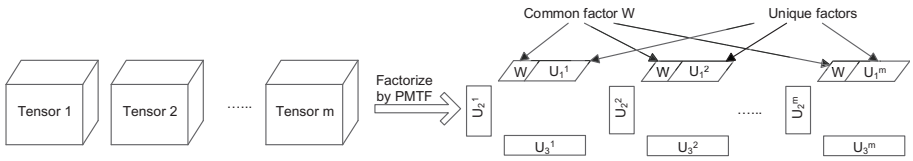

Fig. 1: Factorize all the tensors by our method PMTF, where matrix factors discover both common and unique patterns

tensor of M modes, existing TF methods decompose the tensor into M low-rank matrix factors, each of which explains a compact basis of each mode of the tensor. Two common approaches for using TF to factorize a sequence of m tensors are: (*option 1*) decompose each tensor separately – this approach generates a low-rank factor matrix for each mode of each tensor, and does not necessarily identify a potentially important "common factor" matrix that these tensors may share; or (*option 2*) concatenate all tensors along a certain mode to form one big tensor and then decompose it – in contrast to the first option, this strategy may discard possible "unique factor" matrices in the concatenated dimension, and only produces factors that are a consensus of the original tensors. Although it is possible to treat these tensors as a data stream and use sliding windows to analyse them incrementally [8,9], the actual decomposition on each element within each window is still limited to the above two options.

In this research we propose a novel strategy for probabilistically analysing multiple tensors, and introduce the concepts of *common factors* and *unique factors* along each mode of all tensors. As demonstrated in Fig. 1, the common space (denoted by **W**) along the first dimension of all tensors occupies a fraction of the factor matrix, while the remaining fraction is preserved for each tensor independently so that any unique patterns that are discriminative to each tensor are also preserved. We will show that when the tensors are associated with class labels, the factorization of both common and unique factors is especially beneficial for feature extraction (dimension reduction) of classification tasks. The strategy of decomposing common and unique factors provides a flexible choice on the sizes of common and unique spaces, such that the preceding two options become special cases of our proposed approach. When the common space is empty (i.e., when the size of **W** in Fig. 1 is zero) we obtain *option 1*, and when it is set to the full size of the factor matrix (instead of a fraction) we obtain *option 2*.

In summary, we make the following contributions in this paper:

1. We introduce the concepts of common factors and unique factors in decomposing a sequence of tensors, and formulate the problem of approximating low-rank representations of tensors as simultaneously optimizing the approximation of both common and unique factors;
2. We propose a PMTF (probabilistic multiple tensor factorization) model, which incorporates both the common and unique factor matrices inherently in the factorization process;
3. We perform empirical evaluations of PMTF on feature extraction for graph classification, which demonstrates the power and effectiveness of our method.

2 Related Work

Factorization methods for tensors that are essentially higher order generalizations of those for matrices have been studied, such as the probabilistic tensor factorization (PTF) method [7,12]. As a multi-dimensional generalization of matrix factorization, PTF is more attractive than matrix factorization not only because it considers more dimensions of information, but also because it usually allows for a unique decomposition of a data set into factors under mild conditions, which are usually satisfied by real data [6]. The field of multi-task learning [13] is also related to our research, however there is no existing work in the multi-task learning domain that studied the problem for tensor factorizations. Coupled tensor and/or matrix factorization methods [5] are also closely related to this research. However, no existing coupled factorization methods address the problem of discovering both shared and unique factors simultaneously. It has also been proposed to perform coupled tensor factorizations [1,14] by using generalised learning models. However, these papers only consider the case when the decomposed factors are all the same in the shared mode, and did not address how to discover discriminant factors from the shared identical mode between coupled matrices or tensors.

Different from all the above literature, in this research we propose the first method that simultaneously decomposes tensors into both common and unique factors, incorporating their class labels (or data generation sources) in the factorization processes, which significantly improve the effectiveness of the extracted features.

3 Tensor Factorization

Tensors are multidimensional (aka multi-mode) arrays. We denote tensors with 3 or more modes by calligraphic font (e.g., $\mathcal{X}$), denote matrices (tensors with 2 modes) by boldface uppercase letters (e.g., $\mathbf{U}$), and denote vectors (tensors with 1 mode) by boldface lower letters (e.g., $\mathbf{u}$). In the following, we first briefly introduce preliminaries of PTF, and then elaborate the proposed PMTF model. We give the definition of a standard tensor factorization as follows:

3.1 Probabilistic Tensor Factorization

Given a M-mode tensor $\mathcal{X} \in \mathbb{R}^{n_1 \times \ldots \times n_M}$ and the desired low rank r, probabilistic tensor factorization (PTF) method decompose $\mathcal{X}$ into M matrix factors $\mathbf{U}_d \in \mathbb{R}^{n_d \times r}, (d = 1, 2, 3, ..., M)$, such that $\mathcal{X} \approx \sum_{j=1}^{r} \mathbf{u}_1^j \otimes \mathbf{u}_2^j .... \otimes \mathbf{u}_M^j$, where $\mathbf{u}_d^j$ represents the jth column of $\mathbf{U}_d$, and $\otimes$ represents outer products. Taking 3-mode tensor as an example, the element-wise expression of the decomposition can be written as $\mathcal{X}_{i,j,k} \approx \sum_{d=1}^{r} (\mathbf{U}_1)_{r,i} (\mathbf{U}_2)_{r,j} (\mathbf{U}_3)_{r,k} \equiv < \mathbf{u}_1^i, \mathbf{u}_2^j, \mathbf{u}_3^k >$. For ease of interpretations, we use $\mathbf{U}_1$, $\mathbf{U}_2$,... $\mathbf{U}_M$, or $\mathbf{U}_d|_{d=1}^{M}$ (or simply $\mathbb{U}$) to represent the operation $\sum_{j=1}^{r} \mathbf{u}_1^j \otimes \mathbf{u}_2^j .... \otimes \mathbf{u}_M^j$ in the rest of the paper. Moreover, we will use the example of 3-mode tensor to represent the more generic cases of M modes.

Standard PTF method [12] assumes Gaussian distributions on the likelihood of tensor observations given matrix factors:

$$p(\mathcal{X}|\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}_3, \sigma^2) = \prod_{i=1}^{n_1} \prod_{j=1}^{n_2} \prod_{k=1}^{n_3} \left[\mathcal{N}(\mathcal{X}_{ijk} | < \mathbf{u}_1^i, \mathbf{u}_2^j, \mathbf{u}_3^k >, \sigma^2) \right]^{I_{i,j,k}}, \quad (1)$$

where $\mathcal{X}$ is a three-mode tensor, $\mathbf{U}_1, \mathbf{U}_2, \mathbf{U}_3$ are respectively the tensors factor matrices in each mode, the inner product of column vectors $< \mathbf{u}_1^i, \mathbf{u}_2^j, \mathbf{u}_3^k >$ is the mean of the Gaussian distribution which has variance σ^2, and the binary indicator $I_{i,j,k}$ equals 1 if value $\mathcal{X}_{i,j,k}$ is observed and equals 0 otherwise.

3.2 Common and Unique Subspaces

Without loss of generality, we define and solve the problem of learning common and unique factors from multiple tensors by using the scenario of two tensors (e.g., $m = 2$ in the example of Figure 1). This scenario corresponds to the case of binary classes, where each tensor contains instances from a class label. We omit the lengthy derivations for $m > 2$ scenarios due to their close theoretical similarity to the $m = 2$ scenario. Assume it is the first mode of tensors that we want to derive both common and unique factors, the probabilistic learning problem can be defined as follows:

Definition 1. *(Probabilistic multiple tensor factorization (PMTF)): Given two M-mode tensors $\mathcal{X}_1$ and $\mathcal{X}_2$, PMTF probabilistically decomposes each of them as the product of $M + 1$ factor matrices so that "$\mathcal{X}_1 \approx [\boldsymbol{W}|\boldsymbol{V}] \otimes \boldsymbol{U}_2 \otimes \boldsymbol{U}_3 ... \otimes \boldsymbol{U}_M$" and "$\mathcal{X}_2 \approx [\boldsymbol{W}|\boldsymbol{S}] \otimes \boldsymbol{T}_2 \otimes \boldsymbol{T}_3 ... \otimes \boldsymbol{T}_M$" hold simultaneously.*

In this definition, we use "[**W**|**V**]" and "[**W**|**S**]" to represent the first matrix factor of each tensor, where **W** is the common factor, and **V** and **S** are respectively the unique factors of the two tensors. Using the previous illustration in Figure 1, **V** and **S** are equivalent to $\mathbf{U}_1^1$ and $\mathbf{U}_1^2$. Since the common factor is located in the first dimension of the tensors, **W** is of size $n_1 \times s$, while **V** and **S** are of size $n_1 \times (r - s)$, where s is the desired cardinality of the common factor matrix ($0 \leq s \leq r$). The concatenation [**W**|**V**] is of size $n_1 \times r$, aligning with $\mathbf{U}_2$ and $\mathbf{U}_3$ which are of size $n_2 \times r$ and $n_3 \times r$ respectively. By using 3-mode tensors as an example, a graphical illustration representing the model of PMTF is shown in Fig. 3.

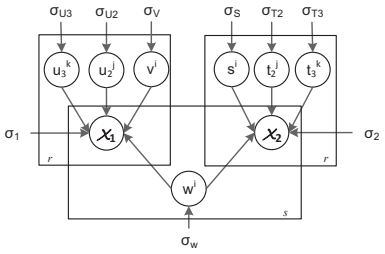

Fig. 3: Graphical representations on using PMTF to factorize two tensors $\mathcal{X}_1$ and $\mathcal{X}_2$. PMTF unveils both the common factor matrix **W** , and unique factors **V** and **S** (shown by their column vectors w^i, v^i, and s^i respectively).

The conditional probability of tensor data observations is modelled from:

$$p(\mathcal{X}_1|\mathbf{W},\mathbf{V},\mathbf{U}_2,\mathbf{U}_3,\sigma_1^2) = \prod_{i,j,k} \left[\mathcal{N}((\mathcal{X}_1)_{ijk}| < \mathbf{w}^i, (\mathbf{u}_2^j)_w, (\mathbf{u}_3^k)_w >, \sigma_1^2)\right]^{(I_1)_{i,j,k}} \times \prod_{i,j,k} \left[\mathcal{N}((\mathcal{X}_1)_{ijk}| < \mathbf{v}^i, (\mathbf{u}_2^j)_v, (\mathbf{u}_3^k)_v >, \sigma_1^2)\right]^{(I_1)_{i,j,k}}, \tag{2}$$

and

$$p(\mathcal{X}_2|\mathbf{W},\mathbf{S},\mathbf{K}_2,\mathbf{K}_3,\sigma_2^2) = \prod_{i,j,k}\left[\mathcal{N}((\mathcal{X}_2)_{ijk}|<\mathbf{w}^i,(\mathbf{t}_2^j)_w,(\mathbf{t}_3^k)_w>,\sigma_2^2)\right]^{(I_2)_{i,j,k}} \times \prod_{i,j,k}\left[\mathcal{N}((\mathcal{X}_2)_{ijk}|<\mathbf{s}^i,(\mathbf{t}_2^j)_s,(\mathbf{t}_3^k)_s>,\sigma_2^2)\right]^{(I_2)_{i,j,k}}, \tag{3}$$

where $(I_1)_{i,j,k}$ and $(I_2)_{i,j,k}$ contain binary indicators that respectively represent whether the entries at $\{i,j,k\}$ position of $\mathcal{X}_1$ and $\mathcal{X}_2$ are observed. Factor matrices of both tensors are modelled by Gaussian priors:

$$p(\mathbf{W}|\sigma_\mathbf{W}) = \prod_{i=1}^{n_1}\mathcal{N}(\mathbf{w}^i|0,\sigma_\mathbf{W}\mathbf{I}_w), \quad p(\mathbf{V}|\sigma_\mathbf{V}) = \prod_{i=1}^{n_1}\mathcal{N}(\mathbf{v}^i|0,\sigma_\mathbf{V}\mathbf{I}_v),$$

$$p(\mathbf{S}|\sigma_\mathbf{S}) = \prod_{i=1}^{n_1}\mathcal{N}(\mathbf{s}^i|0,\sigma_\mathbf{S}\mathbf{I}_s), \quad p(\mathbf{U}_d|\sigma_{\mathbf{U}_d}) = \prod_{i=1}^{n_d}\mathcal{N}(\mathbf{u}_d^i|0,\sigma_{\mathbf{U}_d}\mathbf{I}), \quad p(\mathbf{T}_d|\sigma_{\mathbf{T}_d}) = \prod_{i=1}^{n_d}\mathcal{N}(\mathbf{t}_d^i|0,\sigma_{\mathbf{T}_d}\mathbf{I}),$$

where d = 2 and 3, $\mathbf{I}_w$, $\mathbf{I}_v$, $\mathbf{I}_s$, and $\mathbf{I}$ are respectively identity matrices of size s by s, $r-s$ by $r-s$, $r-s$ by $r-s$, and r by r. The log-posterior probability of the factor matrices is then:

$$\begin{aligned}&\ln p(\mathbf{W},\mathbf{V},\mathbf{S},\mathbf{U}_d,\mathbf{T}_d|\mathcal{X}_1,\mathcal{X}_2,\Theta)\\ =&\ln p(\mathcal{X}_1|\mathbf{W},\mathbf{V},\mathbf{U}_2,\mathbf{U}_3,\sigma_1^2) + \ln p(\mathcal{X}_2|\mathbf{W},\mathbf{S},\mathbf{K}_2,\mathbf{K}_3,\sigma_2^2) + \ln p(\mathbf{W}|\sigma_\mathbf{W}) + \ln p(\mathbf{V}|\sigma_\mathbf{V})\\ &+\ln p(\mathbf{S}|\sigma_\mathbf{S}) + \ln p(\mathbf{U}_d|\sigma_{\mathbf{U}_d}) + \ln p(\mathbf{T}_d|\sigma_{\mathbf{T}_d}) + C'\end{aligned}$$

where $\Theta = \{\sigma_1,\sigma_2,\sigma_W,\sigma_V,\sigma_S,\sigma_{U_d},\sigma_{T_d}\}$, d = 2 and 3, C' is a constant that is not dependent on any of the parameters. By making use of the probability density function of Gaussian distribution, maximizing the above function is equivalent to minimizing the following sum of squared error:

$$\begin{aligned}\min &\sum_{i,j,k}(I_1)_{ijk}\left(((\mathcal{X}_1)_{ijk} - <\mathbf{w}^i,\mathbf{u}_2^j,\mathbf{u}_3^k>)^2 + ((\mathcal{X}_1)_{ijk} - <\mathbf{v}^i,\mathbf{u}_2^j,\mathbf{u}_3^k>)^2\right)\\ &+\sum_{i,j,k}(I_2)_{ijk}\left(((\mathcal{X}_2)_{ijk} - <\mathbf{w}^i,\mathbf{u}_2^j,\mathbf{u}_3^k>)^2 + ((\mathcal{X}_2)_{ijk} - <\mathbf{s}^i,\mathbf{u}_2^j,\mathbf{u}_3^k>)^2\right)\\ &+\sum_i\frac{\lambda_\mathbf{W}\|\mathbf{w}^i\|_2^2}{2} + \sum_i\frac{\lambda_\mathbf{V}\|\mathbf{v}^i\|_2^2}{2} + \sum_i\frac{\lambda_\mathbf{S}\|\mathbf{s}^i\|_2^2}{2} + \sum_j\frac{\lambda_{\mathbf{U}_2}\|\mathbf{u}_2^j\|_2^2}{2} + \sum_k\frac{\lambda_{\mathbf{U}_3}\|\mathbf{u}_3^k\|_2^2}{2}\\ &+\sum_j\frac{\lambda_{\mathbf{T}_2}\|\mathbf{t}_2^j\|_2^2}{2} + \sum_k\frac{\lambda_{\mathbf{T}_3}\|\mathbf{t}_3^k\|_2^2}{2}\end{aligned} \tag{4}$$

where $\lambda_\mathbf{W} = \sigma_\mathbf{W}/\sigma_1$, $\lambda_\mathbf{V} = \sigma_\mathbf{V}/\sigma_1$, $\lambda_\mathbf{S} = \sigma_\mathbf{S}/\sigma_2$, $\lambda_{\mathbf{U}_d} = \sigma_{\mathbf{U}_d}/\sigma_1$, $\lambda_{\mathbf{T}_d} = \sigma_{\mathbf{T}_d}/\sigma_2$ (d=2,3). The objective function in Eq. 4 is convex with respect to each matrix factor and can be minimized by gradient descent or block coordinate decent algorithms, which both iteratively update one parameter at a time. In our experiments, we alternate between optimizing the hyperparameters and updating the columns of matrix factors with the hyperparameters fixed.

3.3 Applying PMTF for Supervised Feature Extraction

Given a set of graphs, each of which is associated with a class label, the graph classification task is to predict the class of a new graph. Similar to general factorization-based

Table 1: Statistics of chemical compound data sets

Name	#graphs	Descriptions	Name	#graphs	Descriptions
AID83	27784	Breast Cancer	AID81	40700	Colon Cancer
AID123	40152	Leukemia	AID1	40460	Lung Cancer
AID33	40209	Melanoma	AID47	40447	Nerve Cancer
AID109	40691	Ovarian Cancer	AID41	27585	Prostate Cancer
AID145	40164	Renal Cancer	AID1481	217968	ATPase Inhibition
AID1416	217968	PERK Inhibition	AID1446	217968	Janus Kinase

dimension reduction methods such as PCA (where each Principle Component is used as a new feature), we apply our factorization method PMTF as a feature extraction method for classification problems, by making use of *the features (i.e., column vectors) defined by the new low-dimension feature spaces* $\boldsymbol{W}$, $\boldsymbol{V}$ *and* $\boldsymbol{S}$, in comparison to the *features defined by column vectors of matrix factors from standard factorization methods*. The performance of the features selected by our method for graph classification is evaluated in the next section.

4 Experiments and Analysis

We implement PMTF and PTF in Matlab by using the Tensor Toolbox [2]. This toolbox also contains an implementation of CP tensor decomposition, which we use in the evaluation. All experiment results presented in this section are from 5-fold cross validation with 10 repeated runs.

4.1 Data Sets

We apply PMTF to graph feature extraction and classification problem on chemical compound data sets, where each chemical compound is treated as a graph. We use bioassays of anti-cancer activity and kinase inhibition (AID)[1]: the task is to predict whether a compound is positive or negative in anti-cancer activities or in kinase inhibition activities. Details of these chemical compound data sets are reported in Table 1.

4.2 Feature Extraction for Graph Classification

In each data set we construct two tensors, one for each class, where all unique types of atoms found in a data set are converted to the labels of vertices, and the lengths of bonds between atoms are weights of the edges. So an entry of a tensor is a count, which tells that for a certain compound, how many edges (bonds) connect certain types of atoms and have certain edge weights (lengths).

We compare the accuracy of classification on data points projected into the new low-dimension feature space produced by PTF, PMTF and CP decomposition, where we vary the settings of low ranks (r) from 5 to 20. In PTF, all training data instances

[1] http://pubchem.ncbi.nlm.nih.gov

Table 2: Comparisons of different methods in their effectiveness of feature extraction for graph classification using logistic regression.

Data sets	AUC from Logistic Regression					AUC from SVMs with Quadratic Kernels				
	CP	PTF	GTF	PMTF	Best *s*	CP	PTF	GTF	PMTF	Best *s*
AID83	0.615	0.568	0.511	**0.727**	17	0.535	0.582	0.586	**0.610**	18
AID81	0.608	0.757	0.739	**0.762**	16	0.642	0.642	0.699	**0.738**	11
AID123	0.686	0.761	0.695	**0.778**	12	0.604	0.601	0.637	**0.646**	7
AID1	0.782	0.806	**0.886**	0.884	9	0.798	0.738	0.725	**0.810**	7
AID33	0.675	0.595	0.711	**0.806**	11	0.711	0.765	0.762	**0.822**	15
AID47	0.622	0.604	0.801	**0.829**	10	**0.792**	0.774	0.752	0.791	9
AID109	0.797	0.628	0.741	**0.799**	14	0.711	0.761	0.757	**0.808**	13
AID41	0.680	0.554	0.728	**0.741**	7	0.692	0.696	0.722	**0.800**	16
AID145	0.733	0.674	0.762	**0.891**	6	0.807	**0.893**	0.799	0.891	11
AID1481	0.603	0.590	0.660	**0.799**	9	0.720	0.696	0.705	**0.780**	18
AID1416	0.674	0.646	0.721	**0.807**	13	0.780	0.652	0.697	**0.797**	9
AID1446	0.865	0.749	0.758	**0.901**	5	0.889	0.808	0.869	**0.904**	6
Frd. test	✓ 0.006	✓ 0.018	✓ 0.051	Base	–	✓ 0.045	✓ 0.036	✓ 0.002	Base	–

are factorized together, so it only discovers common factors of both classes. In CP, tensors belonging to differents classes are factorized separately, hence it only find unique factors of the classes. We also include the GTF (Generalised Coupled Tensor factorization) method [14] in our evaluations, which is built on generalised linear models and produces common factors on the common mode of tensors.

To test the distinctness of the new data points from different classes under the new low-dimension feature space, we use two types of classifiers to learn from the new data points, a linear classifier – logistic regression, and a non-linear classifier – support vector machines (SVMs) with quadratic kernels (i.e., the kernel between two *vectorized* data samples x_i and x_j is: $k(x_i, x_j) = (1 + x_i^T x_j)^2$).

The Friedman test is reported as one of the most appropriate methods for validating multiple classifiers among multiple data sets [4]. To confirm the significance of the superiority of PMTF, we perform Friedman tests on the sequences of AUC values across all data sets, where *p*–values that are lower than 0.05 reject the hypothesis with 95% confidence that the classifiers in the comparison are not statistically different. In Tables 2 we report the performance of the classifiers on different factorization methods when the rank is 20. In each data set the AUC value of the best performing method is put in boldface font. To show the diversity of the data sets, we also present the best s values which are optimized from the training set of cross validation. From the low *p*–values shown in the bottom of Tables 2, it is easy to see that the low-rank spaces produced by PMTF are significantly better than the other corresponding methods in distinguishing the two class labels on each data set.

5 Conclusions and Future Work

In this research we focus on the problem of probabilistically factorizing a sequence of labelled tensors in order to improve tensor feature extraction for supervised learning.

We formulate this problem into the task of discovering common and unique factors from multiple tensors. The proposed PMTF model is a generic tensor factorization method that can potentially be applied to many practical problems. We have applied PMTF to the problem of feature extraction (dimension reduction) for graph classification. Empirical results demonstrate the superiority of the factors discovered by PMTF over other existing methods. We note that besides graphs, our method can also be applied to any other data represented in multi-mode forms (such as images and videos).

In future, we plan to investigate the use of PMTF in collaborative filtering problems, where different tensors represent different domains and the common/unique factors learned by PMTF can be helpful for building cross-domain recommendation systems. Besides, we also plan to apply the method of PMTF to other domains where tensor representation are used, such as text mining and information retrieval.

References

1. Acar, E., Kolda, T.G., Dunlavy, D.M.: All-at-once optimization for coupled matrix and tensor factorizations. MLG workshop (2011)
2. Bader, B., Kolda, T.: Efficient Matlab computations with sparse and factored tensors. SIAM Journal on Scientific Computing 30(1), 205–231 (2007)
3. Carroll, J., Chang, J.: Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckart-Young" decomposition. Psychometrika 35(3), 283–319 (1970)
4. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research 7, 1–30 (2006)
5. Ermiş, B., Acar, E., Cemgil, A.T.: Link prediction in heterogeneous data via generalized coupled tensor factorization. Data Mining and Knowledge Discovery 29(1), 203–236 (2015)
6. Heiler, M., Schnörr, C.: Controlling sparseness in non-negative tensor factorization. In: Proceedings of the 9th European Conference on Computer Vision (ECCV). pp. 56–67 (2006)
7. Shashua, A., Hazan, T.: Non-negative tensor factorization with applications to statistics and computer vision. In: Proceedings of the 22th International Conference on Machine Learning (ICML). pp. 792–799 (2005)
8. Sun, J., Papadimitriou, S., Yu, P.: Window-based tensor analysis on high-dimensional and multi-aspect streams. In: Proceedings of IEEE International Conference on Data Mining (ICDM). pp. 1076–1080 (2006)
9. Sun, J., Tao, D., Faloutsos, C.: Beyond streams and graphs: dynamic tensor analysis. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD). pp. 374–383 (2006)
10. Tucker, L.: Some mathematical notes on three-mode factor analysis. Psychometrika 31(3), 279–311 (1966)
11. Welling, M., Weber, M.: Positive tensor factorization. Pattern Recognition Letters 22(12), 1255–1261 (2001)
12. Xiong, L., Chen, X., Huang, T., Schneider, J., Carbonell, J.: Temporal collaborative filtering with bayesian probabilistic tensor factorization. In: Proceedings of SIAM Conference on Data Mining (SDM) (2010)
13. Xu, J., Tan, P.N., Luo, L.: Orion: Online regularized multi-task regression and its application to ensemble forecasting. In: Proceedings of the 2014 IEEE International Conference on Data Mining (ICDM). pp. 1061–1066 (2014)
14. Yılmaz, Y., Cemgil, A., Simsekli, U.: Generalised coupled tensor factorisation. In: Advances in Neural Information Processing Systems (NIPS) (2011)