International Journal of
Applied Mathematics,
Electronics and Computers
ISSN: 2147-8228 www.atscience.org/IJAMEC
Advanced Technology and Science
Original Research Paper

A Survey of Tensor Factorization Frameworks on Audio Modelling#

Ünsal Gökdağ *[1]

Accepted 15th August 2014

Abstract: This survey is about Tensor Factorization methods for audio modeling, which focuses on probabilistic latent tensor factorization and generalized coupled tensor factorization by expectation maximization method while using several linear and nonlinear distance measure methods.

Keywords: *Tensor Factorization, Probabilistic Latent Tensor Factorization*

1. Introduction

With the advancements in the computational power, it is possible to have huge amounts of data for analyzing. To extract useful information from that data, one should make use of effective and efficient methodologies. The matrix factorization had major impact on clustering, non-negative matrix factorization, latent semantic indexing, collaborative indexing and many other methods when considered as matrix factorization problems. The advantage of matrix factorization lies within its most basic structure, matrices, which allows great parallelism in terms of computation. Also, matrix computations are well understood algorithms which are stable, efficient and sufficient. [1]

Tensor Factorization(TF) is used on multi-way analysis to identify and extract hidden structure inside the given data. Since there are numerous ways to factorize a matrix, several different models are emerged. For example, canonical decomposition(CP), non-negative matrix factorization(NMF), NMFD, NMF2D, and SF-SSNTF are common tensor factorization models. Also, TF is an excellent candidate for those domains which has the ability to estimate missing values while estimating the parameters of the given model. To do this, first the point estimation of the values has to calculated. However in this method, the model used for TF becomes very important because the correlation between different dimensions is given by the model itself.

For hierarchical modeling of an audio, Probabilistic Latent Tensor Factorization (PLTF)[2] makes a good choice. In detail, PLTF combines two different approaches, probabilistic graphical models and tensor decomposition of multiway data. Many audio and music processing models can be described by TF and tensor marginalization as the side of TF part of the algorithm. Also, probabilistic graphical models are used as a standard for describing the interaction between random variablesin statistical machine learning. In the article, tensor model is given by a factor graph, where tensors corresponds as factor nodes and indices are vertexes. After the model is given, the inference algorithm for TF can be derived from the factor graph. The authors also noted only the mathematical operations used in TF are analogous to the factor model in terms of inference algorithm of probabilistic graphical model.

Another article[3] uses Generalized Coupled Tensor Factorization for estimation of both TF model and missing values. In detail, the model incorporates different kinds of musical information while estimating the missing parts of the audio by taking an approximate musical score which can be from another musical piece and the spectrum of isolated piano sounds.

2. Definition

2.1. Tensor Factorization

We define a K dimensional multiway array $\Lambda \in \chi^{I_1 \times I_2 \times \ldots \times I_K}$ as a tensor. Here, I_k are finite index sets, where i_k is the corresponding index. We denote an element of the tensor $\Lambda(i_1, i_2, \ldots, i_K) \in \chi$ as $\Lambda^{i_1, i_2, \ldots, i_K}$. Also, for a set of indices $W = \{i_1, i_2, \ldots, i_K\}$ we use the notation $\Lambda(W)$ to denote an element of the $\Lambda^{i_1, i_2, \ldots, i_K}$. For the purpose of explaining tensor factorization we define $Z_{1:N} = \{Z_\alpha\}$ for $\alpha = 1 \ldots N$, sharing a set of indices W. Each tensor Z_α has an index set w_α such that $\bigcup_{\alpha=1}^{N} \overline{w}_\alpha = W$. then, w_α is a particular configuration of the indices in Z_α while $\overline{w}_\alpha$ denotes the compliment of the configuration w_α which is $\overline{w}_\alpha = X(w)/w_\alpha$.

One of the basic operations on tensors are tensor marginalization. A tensor marginalization operation is summing up the elements of a tensor over a given index set. For example, given two tensors Λ and $\hat{X}$ with index sets w and w_0, we can write $\hat{X}(w_0) = \sum_{\overline{w}_0} \Lambda(w)$.As an easy example, matrix multiplication operation is just a simple tensor marginalization operation where $\hat{X}(i,j) = \sum_k Z_1(i,k) Z_2(k,j)$. By defining a tensor $X = Z_1(i,k) Z_2(k,j)$ and summing up over the indices k, we can calculate the matrix multiplication. Formally speaking, define $W = \{i,j,k\}$, $w_0 = \{i,j\}$, $w_1 = \{i,k\}$ and $w_2 = \{k,j\}$. By definition, we can write $\overline{w}_0 = \{k\}$ therefore $\hat{X}(w_0) = \sum_{\overline{w}_0} Z_1(w_1) Z_2(w_2)$.

A tensor factorization (TF) model is the product of a set of tensors Z_α for $\alpha = 1 \ldots N$ which defined on the corresponding index set w_α marginalized over the set of indices w_0. Given a TF model, the latent TF problem is to estimate the set of latent tensors $Z_{1:N}$

$$minimize D(X||\hat{X}) s.t. \hat{X}(w_0) = \sum_{\overline{w}_0} \prod_{\alpha} Z_\alpha(w_\alpha) \quad (1)$$

where X is the observed tensor and $\hat{X}$ is the 'prediction' tensor.

[1] *IDEA Teknoloji Çözümleri, Sun plaza BBDO Blok Dereboyu Cd. Bilim Sk. No:5, 34398, Maslak /İstanbul / Turkey*
** Corresponding Author:email:unsal.gokdag@gmail.com*
 # This paper has been presented at the International Conference on Advanced Technology&Sciences (ICAT'14) held in Antalya (Turkey), August 12-15, 2014.

The function $D(\cdot \,||\, \cdot) \geq 0$ is the $distance$ function between tensors X and $\hat{X}$. Common distance functions are *Euclidian* (EU), *Kullback-Leibler*[4] (KL) and *Itakura-Saito*[5] (IS). β*-divergence* generalizes those distances over one distance function which will be explained in detail.

Probabilistic Latent Tensor Factorization(PLTF)

As detailed in the article[6], (PLTF) is a framework for calculating the KL and EU cost functions via full Bayesian inference for any tensor factorization model. The article exploited the duality between exponential families and Bregman divergences which allowed them to conceptualize the TF problem into an inference problem of a probabilistic graphical model with Gaussian or Poisson components which reduces to a parameter estimation problem.

Authors also used *message passing* to further reduce the model into a series of primitive matrix operations. To achieve this, they introduced a notation which resembles *undirected probabilistic graphical models* for TF. This notation makes the model useful for many application domains like audio processing, network analysis, collaborative filtering[7] or vision. All of them needs to be designed specifically for the application domain.

In the acticle[1], PLTF generative model is defined as follows:

$$\Lambda(w) = \prod_{\alpha}^{N} Z_{\alpha}(w_{\alpha})\text{intensity} \tag{2}$$

$$S(w) \sim PO(S; X(w))\text{KLcost} \tag{3}$$

$$X(w_0) = \sum_{w_0} S(w)\text{observation} \tag{4}$$

$$\hat{X}(w_0) = \sum_{\overline{w}_0} \Lambda(w)\text{parameter} \tag{5}$$

$$M(w_0) = \begin{cases} 1 & X(w_0)\ is\ missing \\ 0 & otherwise \end{cases} \tag{6}$$

In the model, $\Lambda(w)$ is referred as *latent intensity field*. Due to the reproductivity property of the Poisson distribution, both the observation $X(w_0)$ and the cost $S(w)$ have same type of distribution. so the problem of minimization the distance between observation and estimation becomes maximization of $log(p(X|Z_{1:n}))$. The missing values in the observation model can be handled by the likelihood maximization:

$$p(X, S|Z) = \prod_{w} (p(X(w_0)|S(w))p(S(w)|\Lambda(w)))^{M(w_0)} \tag{7}$$

2.2. Models

For better understanding, only the models from[2] will be used. All of the models have same variable convention, f is frequency, t is time frame, i is template index, v is local frequency of spectral template, τ is local time frame of spectral template, ι is instrument index, p is harmonic index, r is note label, c is channel index.

For all the factors, D is dictionary, E is excitation, G is gain, H is a filter, N is a harmonic dictionary and W is harmonic weight. Ξ, Ξ_1 and Ξ_2 are intermediate constant factors which computed only once.

2.2.1. NMF

NMF is the first TF model[8] for audio processing which opened

a new area for audio processing. Given a proper chosen model order, factors D and E becomes correlated with spectral templates and music score.

$$W = \{f, t, i\}$$

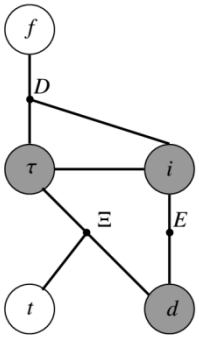
$w_0 = \{f, t\}$

$\overline{w}_0 = \{i\}$

$Z = \{f, i\}, \{i, t\}$

2.2.2. NMFD

The NMF model assumes that coefficients of spectral template components in each frequency bin are same, which makes the model unrealistic. One improvement over the NMF model is



capturing temporal variations by deconvolution which is introduced by Smaragadis [9] with the name of *Non-negative Matrix Factor Deconvolution*(NMFD).

$W = \{f, t, \tau, i, d\}$

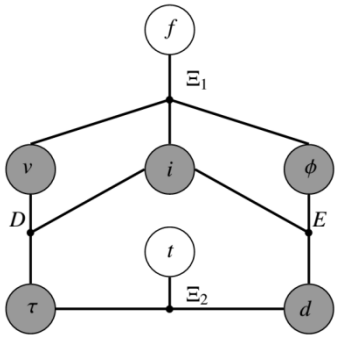
$w_0 = \{f, t\}$

$\overline{w}_0 = \{\tau, i, d\}$

$Z = \{f, \tau, i\}, \{d, i\}, \{d, t, \tau\}$

2.2.3. NMF2D

The NMFD model is further improved by Schmidt and Morup[10] by *Non-negative Matrix Factor 2D Deconvolution*. In addition to NMFD, NMF2D also captures pitch changes by using low-frequency spectrogram. The advantage of the model is that on log-frequency index, modulations correspond to shifts.



$W = \{f, t, v, \tau, i, \phi, d\}$

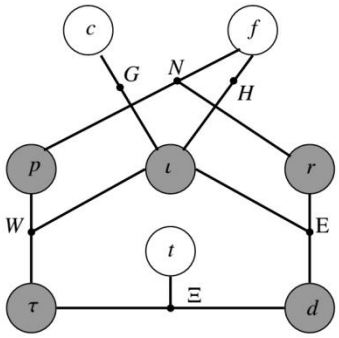
$w_0 = \{f, t\}$

$\overline{w}_0 = \{v, \tau, i, \phi, d\}$

$Z = \{v, \tau, i\}, \{\phi, d, i\}, \{v, f, \phi\}, \{d, t, \tau\}$

2.2.4. SF-SSNTF

Source-Filter Sinusoidal Shifted Nonnegative Tensor Factorization(SF-SSNTF) is another model[11]. The model resembles physically inspired source filter models of audio production in spectral domain by multiplying harmonic excitation withspectral envelope of a body response filter.



$W = \{c, t, f, \iota, p, r, \tau, d\}$

$w_0 = \{c, t, f\}$

$\overline{w}_0 = \{\iota, p, r, \tau, d\}$

$Z = \{c, \iota\}, \{f, \iota\}, \{f, p, r\}, \{p, \iota, \tau\}, \{r, \iota, d\}, \{d, t, \tau\}$

2.3. Generalization of Distance function: β-divergence

The performance of a TF or PLTF model directly depends on the distance function. β-Divergence allows the unification of the distance functions over one generalized function which allows calculation of different distance functions in the same algorithm.

$$d_\beta(x,y) = \begin{cases} \frac{1}{\beta(\beta-1)}(x^\beta + (\beta-1)y^\beta - \beta x y^{\beta-1}) & \beta \geq 2 \\ x(\log x - \log y) + (y - x) & \beta = 1 \\ x/y - \log(x/y) - 1 & \beta = 0 \end{cases} \quad (8)$$

β-divergence function becomes $D_{IS}(x||y)$ for $\beta = 0$, $D_{KL}(x||y)$ for $\beta = 1$ and $D_{EU}(x||y)$ for $\beta = 2$.

2.4. Inference and Update Rules

Inference is the estimation of the latent factors Z_α in an iterative fashion. The estimation operation is done via fixed point update where at each iteration Z_α by fixing other factors $Z_{\overline{\alpha}}$ where $Z_{\overline{\alpha}} = Z/Z_\alpha$

$$Z_\alpha \leftarrow Z_\alpha \circ \frac{\Delta_\alpha(M \circ X \circ \hat{X}^{\beta-2})}{\Delta_\alpha(M \circ \hat{X}^{\beta-1})} \quad (9)$$

where $\circ$ is the Hadamard product and M is mask array which consists of 0-1 entries depending on the actual values is missing or not. In the iteration, the function Δ_α is defined as

$$\Delta_\alpha(O)(w_\alpha) \equiv \sum_{\overline{w}_\alpha} \left(O(w_0) \prod_{\overline{\alpha} \neq \alpha} Z_{\overline{\alpha}}(w_{\overline{\alpha}}) \right) \quad (10)$$

The update operation of Z_α needs two separate computation of Δ_α with parameters $M \circ X \circ \hat{X}^{\beta-2}$ and $M \circ \hat{X}^{\beta-1}$. This operation can be summarized as taking marginal sum of a tensor which is one of basic operations in this framework. By plugging β values of $\{0,1,2\}$ into the equation above, we get individual update rules for each distance function.

Table 1: Update rules for different β values

β	Distance Function	Update Rule
0	Itakura-Saito	$Z_\alpha \leftarrow Z_\alpha \circ \frac{\Delta_\alpha(M \circ X/\hat{X}^2)}{\Delta_\alpha(M/\hat{X})}$
1	Kullback-Leibler	$Z_\alpha \leftarrow Z_\alpha \circ \frac{\Delta_\alpha(M \circ X/\hat{X})}{\Delta_\alpha(M)}$
2	Euclidian	$Z_\alpha \leftarrow Z_\alpha \circ \frac{\Delta_\alpha(M \circ X)}{\Delta_\alpha(M \circ \hat{X})}$

Example: Update Equations of NMFD Model

As explained above, update rules for each factor Ξ can be obtained by taking it from the model and contracting the resulting tensor on latent indices. This method can be shown gracefully on the graphical model.

$$\hat{X}(f,t) = \sum_{\tau,i,d} D(f,\tau,i) E(i,d) \Xi(d,t,\tau) \quad (11)$$

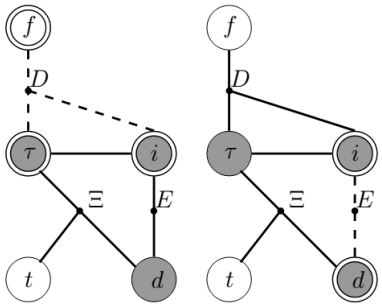
Because $\Xi(d,t,\tau)$ is constant, it can be factored out:

$$\hat{X}(f,t) = \sum_{\tau} \Xi(d,t,\tau) \sum_{\tau,i,d} D(f,\tau,i) E(i,d) \quad (12)$$

Because of the factor Ξ is constant, it will cancel out in the update Equation. After that, the update Equations are straightforward:

$$\Delta_D(O)(f,\tau,i) = \sum_t O(f,t) E(i,d) \quad (13)$$

n each iteration, function $O = \Delta_\alpha(M \circ X \circ \hat{X}^{\beta-2})$ and $O = \Delta_\alpha(M \circ \hat{X}^{\beta-1})$ needs to be calculated repeatedly where $\alpha \in \{D, E\}$



Graphical model of Δ_D and Δ_E calculation. Double circled indices are not summed over, therefore leaving the operation as a tensor marginalization where dotted lines are excluded from the model

3. Methods

3.1. Creating Generalized TF Framework

MUR's are easy to derive on hand but works very good on computer software if given with exact calculations. However, the need to derive each factor equation for every single topology is cumbersome and needs to be automated. The first step on my research is to create a generalized TF framework to make TF operation automated. Suppose we want to factor $X(w_0)$ over factors $Z_\alpha(w_\alpha)$. To do this, we first defined our variables carefully: Indices W, observation indices W_0, latent indices lat, factor indices f, size of the W as s, factors Z and approximation $\hat{X}$. After the definition of the variables, associated Z and Λ are created with the choice of fixing some of the Z which is vital for source separation process. After the variable initialization, $\Lambda(w) = \prod_\alpha^N Z_\alpha(w_\alpha)$ which is an N dimensional tensor is calculated. Keep that in mind that one of the primary operations on the computation framework is hadamard product so we basically reshape all factors $Z_\alpha(\alpha)$ to $Z_\alpha(W)$ while making the sizes of the dimensions $\hat{\alpha} \neq \alpha$ as 1 to maintain number of elements. After the resizing, our computation framework calculates $\Lambda = Z_1(W) \circ Z_2(W) \circ \ldots Z_N(W)$. Calculation of $\hat{X}$ is marginalizing Λ over latent indices therefore reducing its dimension to w_0. For numerical stability, we define ϵ which is smallest number in floating-point representation and make $\{\forall p \in w_0 | \hat{X}(p) < \epsilon \rightarrow \hat{X}(p) = \epsilon\}$. After calculation of Λ, we calculate $(M \circ X \circ \hat{X}^{\beta-2})$ and $(M \circ \hat{X}^{\beta-1})$ as parameters of two separate Δ_α function. after the operation we again increase the dimension of both n and d and for all factor not i, we dot product them to calculate delta functions. after that we marginalize n and d on not alpha indices and squeeze dimensions of them. after that operation we divide n by d hence calculation of Z_α

4. Conclusion

Tensor factorization methods proved to be viable choice on audio modeling. Preliminary works on performance assessment test showed 50% true positive audio transcription rate on several sample music pieces. Algorithm performance staggers on estimation of number of iterations to be performed on the intensity factor Λ. First approach is continuing iteration until the difference between iterations are negligible but with this approach, number of iterations cannot be estimated. Second approach is making predefined number of iterations over the intensity values which could lead to premature convergence on the tensors. On further works, first approach will be used.

Further works on the field includes a working algorithm on the audio transcription using sample piano notes and sample music songs consisting of only piano notes. Prediction performance can be increased by applying a hidden markov model over the obtained results from the TF framework. Another improvement could be score guided audio transcription using generalized coupled tensor factorization which uses musical information such as chromatic scales and tempo information to assist the algorithm about estimating correct notes on the time domain.

References

[1] Y. K. Yılmaz and A. T. Cemgil, "Algorithms for probabilistic latent tensor factorization", Signal Processing, vol. 92, no. 8, pp. 1853 – 1863, 2011.

[2] T. Cemgil, U. Simsekli, and Y. C. Subakan, "Probabilistic latent tensor factorization framework for audio modeling," in Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics WASPAA '11, 2011, pp. 137–140.

[3] U. Simsekli, A. T. Cemgil, and Y. K. Yilmaz, "Score guided audio restoration via generalised coupled tensor factorisation," in ICASSP, 2012, accepted.

[4] S. Kullback and R. A. Leibler, "On information and sufficiency," Ann. Math. Statistics, vol. 22, pp. 79–86, 1951

[5] S. Saito and F. Itakura, "Frequency spectrum deviation between speakers," Speech Communication, vol. 2, no. 2-3, pp. 149–152, 1983.

[6] K. Yilmaz and A. T. Cemgil, "Probabilistic latent tensor factorisation," in Proc. of International Conference on Latent Variable analysis and Signal Separation, vol. 6365, 2010, pp. 346–353.

[7] L. Xiong, X. Chen, T.-K. Huang, J. Schneider, and J. G. Carbonell, "Temporal collaborative filtering with bayesian probabilistic tensor factorization," in Proceedings of SIAM Data Mining, 2010.

[8] P. Smaragdis and J. Brown, "Non-negative matrix factorization for polyphonic music transcription," in Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on., oct. 2003, pp. 177 – 180.

[9] P. Smaragdis, "Non-negative matrix factor deconvolution; extraction of multiple sound sources from monophonic inputs," in Independent Component Analysis and Blind Signal Separation, ser. Lecture Notes in Computer Science, C. Puntonet and A. Prieto, Eds. Springer Berlin Heidelberg, 2004, vol. 3195, pp. 494 – 499.

[10] M. N. Schmidt and M. Mørup, "Nonnegative matrix factor 2-d deconvolution for blind single channel source separation," in Proceedings of the 6th international conference on Independent Component Analysis and Blind Signal Separation, ser. ICA'06. Berlin, Heidelberg: SpringerVerlag, 2006, pp. 700–707.

[11] Klapuri, T. Virtanen, and T. Heittola, "Sound source separation in monaural music signals using excitation-filter model and em algorithm," in Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on, march 2010, pp. 5510 –5513.Allahverdi N. Some Applications of Fuzzy Logic in Medical Area, Proceedings on the 3rd International Conference on Application of Information and Communication Technologies (AICT2009), Published by IEEE, 14-16 October 2009, Azerbaijan, Baku.