Constrained Coupled Matrix-Tensor Factorization and its Application in Pattern and Topic Detection

Sanaz Bahargam
Boston University
bahargam@bu.edu

Evangelos E. Papalexakis
University of California Riverside
epapalex@cs.ucr.edu

***Abstract*—Traditionally, time-evolving topic discovery approaches have focused on the temporal evolution of the topic itself. However, especially in settings where content is contributed by a community or a crowd, an orthogonal notion of time is the one that pertains to the level of expertise of the content creator: the more experienced the creator, the more advanced the topic will be.**

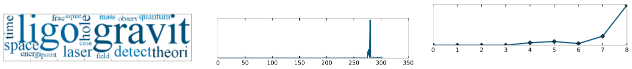
In this paper, we propose a novel time-evolving topic discovery method which, in addition to the extracted topics, is able to identify the evolution of that topic over time, as well as the level of difficulty of that topic, as it is inferred by the level of expertise of its main contributors. Our method is based on a novel formulation of Constrained Coupled Matrix-Tensor Factorization, which adopts constraints that are well motivated for, and, as we demonstrate, are necessary for high-quality topic discovery.

***Index Terms*—Topic Discovery, Time-evolving, Tensors, Coupled Matrix-Tensor Factorization, Constrained Factorization**

I. Introduction

Traditionally, topic modeling and discovery methods have focused on extracting high quality, interpretable topics that aim to succinctly represent the inherent latent structure within a corpus. Recently, there has been significant interest in studying the evolution of topics over time, and this has found particular applications in [3].

To the best of our knowledge, the state-of-the-art in time-evolving topic extraction has focused on a notion of "time" that pertains to the particular moment that a topic emerged and how it evolved throughout its history within a corpus. However, when we are dealing with topic extraction from community and crowd-based platforms, such as `Stack Exchange`, an additional notion of "time" arises. This notion of time is related to the evolution of the user who contributes the content: a relatively new user is more likely to contribute "entry-level" content, whereas an experienced user who has already contributed a significant amount of posts, is more likely to create content that is more advanced. Previous work on topic detection has overlooked this notion of time, which relates to user maturity and experience, and which, as we showcase in this paper, can provide valuable insights on how advanced a particular topic is. In addition to being able to tease out latent concepts of varying levels, these insights are also useful in bootstrapping automated curriculum design approaches [1] which require a set of concepts to be taught in a curriculum, as well as prerequisite relations for those concepts, which can be given via the user maturity dimension in our topic discovery.



(a) Words (b) Time in Weeks (c) "Level of difficulty"

Fig. 1: An example of a topic discussed by advances users at a specific time. This pattern indicates topics discussed in response to an external event. The peak of the "time" mode corresponds to February of 2016 when detection of gravitational waves was announced by Ligo lab; furthermore, "gravitational waves" is an advanced Physics topic, and our method correctly infers its level of difficulty.

In this paper, we introduce a time-evolving topic discovery method, based on Constrained Coupled Matrix-Tensor Factorization, which effectively models time and user maturity/experience towards *extracting interpretable topics, their temporal evolution, as well as their level of difficulty*. Figure 1 shows a representative such topic detected by our algorithm. The topic corresponds to "Gravitational Waves Detection by Ligo Lab"; an advanced Physics topic, which is indicated by the "level of difficulty" aspect of our results, and the topic made its appearance in February of 2016 (as indicated by the "time" aspect), which was the date it was announced. For the longer version of this paper with a more thorough and more experiments, see [2].

II. Preliminary Definitions

Vectors are denoted by boldface lower case letters, e.g $\mathbf{a}$. Matrices are denoted by boldface Capital letters, e.g $\mathbf{A}$. Tensors are denoted by Calligraphic letters, e.g $\mathcal{A}$. An entry of a vector $\mathbf{a}$, a matrix $\mathbf{A}$, or a tensor $\mathcal{A}$ is denoted by a_i, a_{ij}, a_{ijk}. Let $\mathcal{T}_{\mathbf{A}}$ be the matricization of $\mathcal{T}$in the first mode. The Kronecker product of two matrices is denoted by $\mathbf{A} \otimes \mathbf{B}$. The n mode product is denoted by $\times_n$. The outer product of two vectors is denoted by $\circ$. $\parallel \mathbf{A} \parallel_F$ denotes the Frobenius norm of matrix $\mathbf{A}$. Moore-Penrose Pseudoinverse of $\mathbf{A}$ is denoted by $\mathbf{A}^\dagger$.

III. Proposed Model

In a wide variety of applications, we have data that form a tensor and have side information or metadata that may form matrices or other tensors. For instance, suppose we have a (word, time, post number) tensor that indicates how many times a word was used in a specific time and specific post numbers. Usually, question answering platforms also have some metadata on the questions/answers, for instance, tags of the questions, that can form a (words, tags) matrix. Thus we have a third-order tensor, $\mathcal{T} \in \mathbb{R}^{\mathbf{I}\times\mathbf{J}\times\mathbf{K}}$, and a matrix $\mathbf{Y} \in \mathbb{R}^{\mathbf{I}\times\mathbf{F}}$, coupled in the first mode of each and there is a one-to-one correspondence of elements in the first mode of the tensor and the matrix ("word" mode in our case). The coupled-matrix and tensor factorization (CMTF) algorithms jointly factorize multiple data sets in the form of higher-order tensors and matrices by extracting a common latent structure from the shared mode. The existing work on coupled-matrix tensor factorization only considers non-negativity constraints, e.g. $\mathbf{A} \geq 0$. Non-negativity is an important feature of latent factors since many real-world tensors have non-negative values and hidden components have a physical meaning only when non-negative.

Although non-negativity improves interpretability, in many applications it is not enough to make sense of the data. When the goal of factorization is to find the latent topics within the tensor and the matrix, we would like to find as many non-overlapping structures as possible. Non-overlapping latent components directly imply that the latent topics are concise and hence interpretable. We can control the amount of overlap in latent components by imposing orthogonality constraint on each latent component. This means for the first mode, we would like the columns of the latent component $\mathbf{A}$ to be orthogonal, $\forall i,j \quad \mathbf{A}_i^T\mathbf{A}_j \leq \epsilon_{\mathbf{A}} \quad i \neq j$. If $\epsilon_{\mathbf{A}}$ is set to 0, this implies latent components should be completely orthogonal, while values greater than 0 means some overlap is allowed. Furthermore, in practice we desire the factors to be sparse as well. Sparsity constraints improve parsimony and offer a simpler and hence more interpretable model. We enforce the sparsity constraint by imposing constraint on ℓ_1 norm of each column in factor matrices and on the core tensor. Enforcing sparsity on each column of the factor matrices means sparsity is imposed uniformly on each latent component for each mode. Sparsity becomes specially favorable when it is imposed on the core tensor; meaning only a few latent components interact with each other. This removes redundancy and achieves compact sparse representations of the core and hence the core tensor will be easily interpretable.

To the best of our knowledge, we are the first to introduce the constraint coupled-matrix tensor factorization problem with non-negativity, sparsity, and orthogonality constraints. Our intuition and constraints are captured in a formal definition as follows.

Problem 1 (): Given a tensor $\mathcal{T} \in \mathbb{R}^{\mathbf{I}\times\mathbf{J}\times\mathbf{K}}$, auxiliary matrix $\mathbf{Y} \in \mathbb{R}^{\mathbf{I}\times\mathbf{F}}$, and number of factors for each component $\mathbf{R_1}$, $\mathbf{R_2}$, $\mathbf{R_3}$, find the components $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$, and tensor $\mathcal{G}$ such that

$$\min \parallel \mathcal{T} - \mathcal{G}_{\times_3}\mathbf{C}_{\times_3}\mathbf{B}_{\times_2}\mathbf{A}_{\times_1} \parallel_F^2 + \parallel \mathbf{Y} - \mathbf{A}\mathbf{D}^T \parallel_F^2,$$

$$\text{Subject to: For each factor } F \in \{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$$

$$F \geq 0 \text{ and } \forall i \parallel F_i \parallel_1 \leq \epsilon_{F1}, \text{ and } \forall i,j \quad F_i^T F_j \leq \epsilon_{F2} \quad i \neq j$$

$$\text{For core tensor } \mathcal{G}, \mathcal{G} \geq 0, \parallel \mathcal{G} \parallel_1 \leq \epsilon_{\mathcal{G}},$$

For the sake of interpretation, it is enough for the core to be sparse, having a few non-zero elements. Lifting orthogonality constraint from the core tensor means we allow interaction between the same factors, but we only allow a few factors to interact with each other.

IV. Proposed Algorithm

We propose an Alternating Least Squares (ALS) algorithm which converges to a locally optimal solution [7]. Using the ALS method, we solve for each factor at a time while fixing all other factors. If we seek to estimate $\mathbf{A}$, it turns out that we need to concatenate the two pieces of the data $\mathcal{T}$and $\mathbf{Y}$, whose rows refer to matrix $\mathbf{A}$, that is the matricized tensor $\mathcal{T}_{\mathbf{A}}$ and matrix $\mathbf{Y}$, and we can then solve for $\mathbf{A}$ as

$$\mathbf{A} = \begin{bmatrix} \mathcal{T}_{\mathbf{A}} \\ \mathbf{Y} \end{bmatrix}^T \left(\begin{bmatrix} \mathcal{G}_{\mathbf{A}}(\mathbf{B} \otimes \mathbf{C}) \\ \mathbf{D} \end{bmatrix}^\dagger \right)^T \tag{1}$$

Algorithm 1 shows our ALS algorithm to solve the constrained coupled-matrix tensor factorization, **ConCMTF–ALS**. These constraints include non-negativity, sparsity and orthogonality imposed by $\mathbf{A} \geq 0$, $\forall i \parallel \mathbf{A}_i \parallel_1 \leq \epsilon_{\mathbf{A}}$ and $\forall i,j \quad \mathbf{A}_i^T\mathbf{A}_j \leq \epsilon_{\mathbf{A}} \quad i \neq j$ respectively. Rather than alternating to solve each factor completely, we solve for each column of each factor independently. This is possible since the columns of each factor are independent and the constraints we consider can be specified for each column as well. A column in the factor of the first mode, $\mathbf{A}$, indicates a group of words and a column in $\mathbf{B}$indicates specific weeks in the lifetime of the forum. It is important to note the effect of specifying sparsity constraints on the columns rather than the whole matrix. This means *sparsity will be spread uniformly across the whole matrix.* It is worth mentioning that our algorithm can allow any convex constraints to be placed for each factor.

Algorithm 1 The Alternating Least Squares for Constrained Coupled Matrix-Tensor Factorization **ConCMTF–ALS**

```
Input: The tensor $\mathcal{T} \in \mathbb{R}^{\mathbf{I}\times\mathbf{J}\times\mathbf{K}}$ and auxiliary matrix $\mathbf{Y} \in \mathbb{R}^{\mathbf{I}\times\mathbf{F}}$
Output: Coupled Decompositions $\mathbf{A} \in \mathbb{R}^{\mathbf{I}\times\mathbf{R_1}}, \mathbf{B} \in \mathbb{R}^{\mathbf{J}\times\mathbf{R_2}}, \mathbf{C} \in \mathbb{R}^{\mathbf{K}\times\mathbf{R_3}}, \mathbf{D} \in \mathbb{R}^{\mathbf{F}\times\mathbf{R_1}}$
Initialize $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \mathcal{G}$ to non-negative random values
while convergence criterion is not met do
    $\mathbf{A} \leftarrow \underset{\mathbf{A}}{\operatorname{argmin}} ||[\mathcal{T}_{\mathbf{A}} \quad \mathbf{Y}] - \mathbf{A}[\mathcal{G}_{\mathbf{A}}(\mathbf{C} \otimes \mathbf{B})^T \quad \mathbf{D}^T]||_{Fro}$
        Subject to: $\mathbf{A} \geq 0$ and $\forall i \parallel \mathbf{A}_i \parallel_1 \leq \epsilon_{\mathbf{A}}$
        and $\forall i,j \quad \mathbf{A}_i^T\mathbf{A}_j \leq \epsilon_{\mathbf{A}} \quad i \neq j$
    Normalize the columns of $\mathbf{A}$
    Similar updates for $\mathbf{B}$ and $\mathbf{C}$ (omitted for brevity).
    $\mathbf{D} \leftarrow \underset{\mathbf{D}}{\operatorname{argmin}} ||\mathbf{Y} - \mathbf{A}\mathbf{D}^T||_{Fro}$
        Subject to: $\mathbf{D} \geq 0$ and $\forall i \parallel \mathbf{D}_i \parallel_1 \leq \epsilon_{\mathbf{C}}$
        and $\forall i,j \quad \mathbf{D}_i^T\mathbf{D}_j \leq \epsilon_{\mathbf{C}} \quad i \neq j$
    Normalize the columns of $\mathbf{D}$
    $\mathcal{G} \leftarrow \underset{\mathcal{G}}{\operatorname{argmin}} ||vec(\mathcal{T}) - (\mathbf{C} \otimes \mathbf{B} \otimes \mathbf{A})vec(\mathcal{G})||_{Fro}$
        Subject to: $\mathcal{G} \geq 0$ and $\parallel \mathcal{G} \parallel_1 \leq \epsilon_{\mathcal{G}}$
return $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$, $\mathcal{G}$
```

Another advantage of our algorithm is that it can be easily used for PARAFAC decompositions instead of Tucker3 with minimal changes. To achieve this, instead of initializing core to random values in Line 1, we set the core tensor to a super diagonal tensor. In addition, there is no need to estimate core tensor in each iteration and hence Line 12 and 13 can be removed from the algorithm.

V. Results

For our experiment, we focus on question and answers related to the field of physics and python programming in `Stack Exchange`.

Data

`Stack Exchange` is a question answering website created in 2008. answers on a wide range of topics. `Stack Exchange` allows each question to be annotated with one or more terms (tags) indicating the subject matter of the question. We used the latest Physics and programming Data Dump [1] in **`Stack Exchange`**. We only consider the questions which have at least one tag (almost 30 000 questions). From the physics forum data, we created a tensor (multi-way array) $\mathcal{T}$ with three modes (word, time, post number) of size $1351 \times 304 \times 9$. When a user $\mathbf{u}$ uses word $\mathbf{w}$ at week $\mathbf{t}$ in his p^{th} post, we will increase $\mathcal{T}(\mathbf{w}, \mathbf{t}, \log(p))$. Thus, the (i, j, k) value of Tensor $\mathcal{T}$ indicates how many times word i was used at week j in $\log(k^{th})$ posts of all users. Note that post number is relative to each users' sign up date. Hence, if a user signs up and writes a question/answer, her post number is 1.

In our application, beside the words, post numbers and time stamps, we also have the tags associated with each question by the users. We can use question tags as a word-tag matrix indicating how many times each word has been used for a specific tag. We denote this matrix by $\mathbf{Y}$(words and tags) of size 1351×527. with tag j.

We used the questions in the programming Stack Exchange forum which had the "python" tag and we created a tensor with three modes of size $432 \times 411 \times 50$. Similar to Physics data, we created an auxiliary word-tag matrix of size 432×30.

Experimental Evaluation In this part, we evaluate our algorithms under CP/PARAFAC and Tucker3 decomposition models for CMTF. **Our dataset and our code are freely available for download**[2] . We compare our results to non-negative PARAFAC decomposition [4] and sparse non-negative Tucker3 [6]. We refer to them as $\mathbf{PARAFAC{-}NS}$ and $\mathbf{TUCKER3{-}NS}$ respectively. To decide the right number of latent factors (F) to be extracted in each algorithm, we used AutoTen [5] which allows us to find more structured latent embeddings in the data.

$\mathbf{PARAFAC{-}NS}$ vs. $\mathbf{ConCMTF{-}ALS}$ with PARAFAC:

Figure 2 shows two components selected from obtained components using $\mathbf{PARAFAC{-}NS}$ algorithm on Physics dataset. We observe that in these two decompositions, there are overlaps in the set of words found by $\mathbf{PARAFAC{-}NS}$ as well as overlap in time and post number modes. In fact, post numbers have identical trends and the words gravity, time, light, speed, wave, particle, and energy are among frequent words in both components. Moreover, the set of words in both components include a (relatively) large number of words and the word factors are very dense. If the goal of factorization is to find latent structure and patterns in the data, these two components are very similar and hence give us the same structure and little information about the data.

We also used our algorithm, $\mathbf{ConCMTF{-}ALS}$, assuming a CP/PARAFAC decomposition. For this decomposition, we only imposed non-negativity and orthogonality constraint on components $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$ with $\epsilon_{\mathbf{A}}$ = 0.05, $\epsilon_{\mathbf{B}}$ = 0.6, $\epsilon_{\mathbf{C}}$ = 0.2, and $\epsilon_{\mathbf{D}}$ = 0.2. The intuition behind this is that we would like to find components which are distinct in their set of words and the level of maturity (post number values). However, we allow decompositions to have overlap in the time mode as we seek patterns in any period of forums lifetime.

Figure 3 illustrates the components produced by $\mathbf{ConCMTF{-}ALS}$ on Physics dataset. As shown in the figure, the set of words in each component are sparse and they do not share many words as it was in the case of $\mathbf{PARAFAC{-}NS}$ components. The post numbers of each component are non-overlapping as well. The first word component depicts the words "mass", "wave", "equation", "velocity", "particle" which were used in very low post number (i.e. by new users). These are in fact basic topics in physics. The second component covers topics related to harmonic motion and waves topics. Compared to the first component these words appear in larger post numbers, i.e. they are posted by more advanced users. The last component included the words related to "Toroidal inductors and transformers" which appeared in large post number and by very advanced users.

Figure 1 is an example of a component which only appeared in a specific time period and moreover in specific post numbers. This pattern indicates words discussed in response to an external event and the peak in time mode corresponds to Feb, 2016. This is the time that the detection of gravitational waves was announced by Ligo lab.

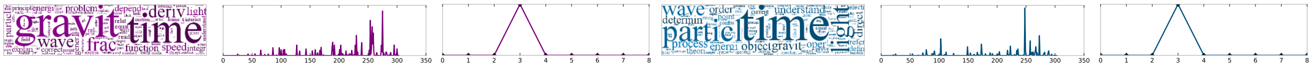
Figure 4 shows two components extracted by our algorithm. The set of words in each component are sparse and they do not share many words and each component shows semantically coherent topics. The first topic includes words related to multiprocessing with a presence across various post numbers. This reveals that such a topic is of interest regardless of the expertise of users. The second topic includes topics related to web crawling. The associated post number reveals that this topic is mainly of interest to new users with lower experience.

VI. User Study

To evaluate the quality of the topics found by our algorithm, we conducted a user study with two goals: 1) evaluate the cohesion of each learning unit, and 2) evaluate the ordering of the units. In the following sub-sections, we present the details of our conducted user-study and the results of our study. We asked the following question to our volunteers: Count the number of odd words in each topic.

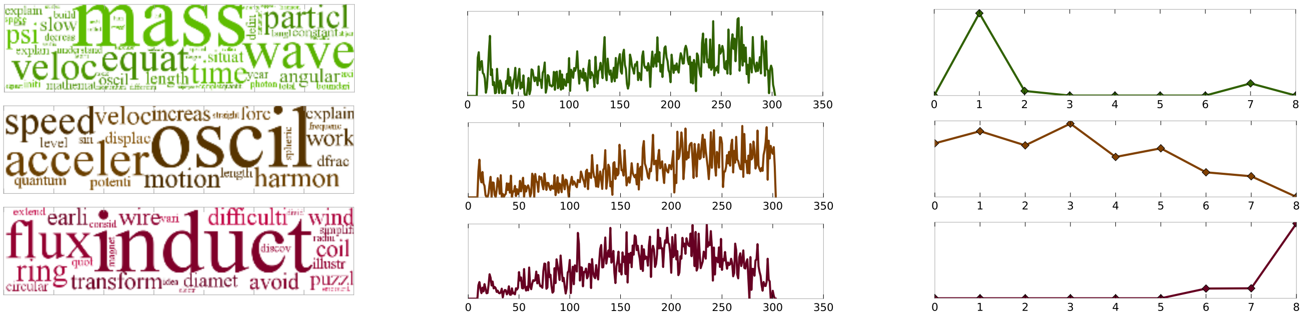
[1] //archive.org/details/stackexchange

[2] https://github.com/sanazb/Constrained_CMTF



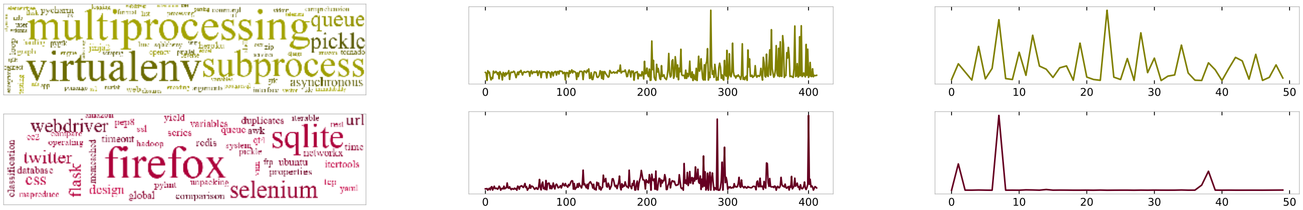
(a) Words (b) Time in Weeks (c) Log Post Number (d) Words (e) Time in Weeks (f) Log Post Number

Fig. 2: An example of two components extracted by **PARAFAC−NS** algorithm on Physics dataset. The two components are similar in word, time and post number modes. The words gravity, time, light, speed, wave, particle, and energy are frequent in both components.



Words Time in Weeks Log Post Number

Fig. 3: An example of four components extracted by **ConCMTF−ALS** algorithm on physics dataset. All components have distinct set of words and distinct post numbers.



Words Time in Weeks Post Number

Fig. 4: An example of four components extracted by **ConCMTF−ALS** algorithm on programming dataset.

a	Min	Max	Median	Mean	# Concepts
Unit 1	1	4	2	2.3	11
Unit 2	1	4	2	2.1	11
Unit 3	0	1	0	0.3	12
Unit 4	1	4	2	2.5	12
Unit 5	1	5	1.5	2	13

TABLE I: Survey results for Q1 (number of odd words in each unit)

Table I summarizes the results of our survey including the min, max, mean, and the median of values that our participants reported as the number of odd words in each topic (unit). For all units, the number of odd words is very low, demonstrating good cohesion in each set of words. It is also important to evaluate the inter-judge agreement in a survey like ours. Due to the nature of the ratings, an appropriate way of analyzing the agreement is by using Krippendorffs α statistical measure, which is applicable to the current scenario of judges assigning a value to a specific variable. The overall agreement measured by Krippendorffs α for our ten judges turns out to be 0.32. This indicates that there is a fair but imperfect agreement.

Applicability to Curriculum Design Our proposed topic discovery has implications to curriculum design since it is able to identify topics along with their level of difficulty; those levels of difficulty are key in determining prerequisite and co-requisite relations between concepts in the syllabus. Here, we demonstrate this applicability of our topic discovery to automated curriculum design, along the lines of the recently proposed work of [1]. In order to achieve this, we order the topics based on their relevant difficulty. What follows is the curriculum we obtained from the online discussion after removing all non-physics terms.

Flow, Mass, Work, Density, Motion, Speed, Velocity, Displacement, Acceleration, Momentum, Gravity, Force, Waves, Electromagnetic, Radioactivity, Quantum, Particles

This curriculum is consistent with the majority of curricula taught in basic physics courses in online/traditional classrooms.

VII. Conclusion and Future Work

We proposed a time-evolving topic discovery method, powered by a novel constrained Coupled Matrix-Tensor Factorization model. Our approach identifies the level of difficulty of extracted topics, and through qualitative and quantitative experimentation, we demonstrate that it produces high-quality interpretable time-evolving topics.

Acknowledgments

Research was supported by an Adobe Data Science Research Faculty Award.

References

[1] R. Agrawal, B. Golshan, and E. Papalexakis. Toward data-driven design of educational courses: A feasibility study. In *JEDM-Journal of Educational Data Mining*, 2016.

[2] S. Bahargam and E. Papalexakis. A constrained coupled matrix-tensor factorization for learning time-evolving and emerging topics. *arXiv preprint arXiv:1807.00122*, 2018.

[3] G. MILLER. Data from a century of cinema reveals how movies have evolved. *www.wired.com*, 2014.

[4] M. Mørup, L. Hansen, and S. Arnfred. Sparse higher order non-negative matrix factorization. *Neural Computation*, 2006.

[5] E. E. Papalexakis. Automatic unsupervised tensor mining with quality assessment. In *SDM*, 2016.

[6] E. E. Papalexakis, C. Faloutsos, and N. D. Sidiropoulos. Parcube: Sparse parallelizable tensor decompositions. In *ECML PKDD*. Springer, 2012.

[7] G. Tomasi and R. Bro. A comparison of algorithms for fitting the parafac model. *Computational Statistics & Data Analysis*, 2006.