Building Tag-Aware groups for Music High-Order Ranking and Topic Discovery

Dimitrios Rafailidis
Department of Informatics, Aristotle University, Greece

Alexandros Nanopoulos
Institute of Computer Science, University of Hildesheim, Germany

Yannis Manolopoulos
Department of Informatics, Aristotle University, Greece

ABSTRACT

In popular music information retrieval systems, users have the opportunity to tag musical objects. Through their tags, users express their personal preferences. Such preferences provide valuable insights about the formulation of user groups/communities. Here, we focus on the analysis of social tagging data to reveal coherent groups characterized by their users, tags and music objects, e.g. songs and artists. This analysis allows for the expression of discovered groups in a multi-aspect way. For each group we can reveal the most prominent users, tags, and music objects using a generalization of the popular web-ranking concept into the social data domain. We experiment with real data and our results show that each Tag-Aware group corresponds to a specific music topic and additionally, a three way ranking analysis is performed inside each group. We believe that building Tag-Aware groups is crucial in offering ways to add structure in the unstructured nature of tags.

Keywords: Data Retrieval, Knowledge Classification, Multimedia, Topic Discovery, High Order Ranking.

INTRODUCTION

Social tagging is the process of adding metadata by users in the form of keywords to annotate information items. In the case of music, the annotated items can be songs, artists, albums and playlists. Social tags are widely used, as high volume sources of descriptive metadata for music. Tags give the opportunity to a user to express his opinion, aiming to capture his personal view of resources he is interested in, including information about genre, mood, instrumentation and quality. A question is arising, why using tags is beneficial for music information. The paramount motivations are, firstly building playlists by tagging songs and secondly, summarizing and categorizing a user profile using recommendation systems based on tags. This way, social tags have become important for Music Information Retrieval. In particular, tags can provide an insight to user behavior and language usage, e.g. how different is "rap" from "hip-hop". Additional benefits are grouping music items based on tags, finding social groups with shared

interests, e.g. people that tag the same items or use the same tags, and generating user profiles from tagging behavior e.g. tag clouds based on tags applied by the user or representing his taste. Therefore, many music discovery and recommendation systems support the social tagging of music. According to Lamere (2008), social tags are used to help searching for items, exploring for new items, finding similar items, and finding other listeners with similar interests.

To assist in exploring and suggesting tags, some systems cluster similar tags together. These tag clusters can assist users in either applying tags or in identifying alternatives that may be useful in their search. Tag clustering has become an important issue for Music Information Retrieval systems. "Tag Radio" and "Tag Cloud" comprise two of the most popular applications of tag clustering. A music search engine can support a user-end tagging or labeling of artists, albums, and songs to create a site-wide folksonomy of music. Users can browse via tags, but the important benefit is the tag radio, permitting users to play music that has been tagged a certain way. The tagging process can be performed by genre "garage rock", mood "chill", artist characteristic "baritone", or any other form of user-defined classification "seen live". Listeners can search for artists or songs that have been tagged with a particular tag, or they can tune into "Tag Radio" where they listen to music that has been tagged with a particular tag.

On the other hand, it has been proved that sometimes tag clustering is not strong enough by itself to satisfy the listeners' requirements. Often users need to obtain results in a ranked list, where the position of each result should be taken into account. A question arising is why ranking is important in Music Information Retrieval. In the more general field of Information Retrieval, ranking of query/search answers has become mandatory for internet searches. When the answers of a query or search are varying in quality and are large in numbers, it is necessary to rank/order these answers based on some criteria. From a users' viewpoint, ranking is extremely useful especially when associated with the retrieval of a few (top-k) answers. This way, ranking has also become an important issue in music search engines, a basic application of Music Information Retrieval on the internet.

A premature form of ranking in music search engines can be found. For example, people often run upon tag clouds, while visiting music search engines, e.g. Last.fm. A particular song, artist or user can be described by a tag cloud. For example, if a tag cloud describes a certain artist, by clicking on a particular tag in the tag cloud, all artists that have been frequently tagged with that tag will be displayed. In particular, we note that larger fonts in such displays indicate most popular tags, like in Figure 1, where the font size is scaled by the tag's weight (popularity or rank).

Figure 1. A tag cloud from Last.fm, describing the artist "The Beatles"

Tag clouds consist of collective information from users and provide a visual illustration for these tags. They effectively provide a good overview about a certain music topic like "The Beatles" in Figure 1, and describe it in instant. However, although tag clouds virtually hide the notions of users and music objects (such as songs, artists, albums etc), these two dimensions, e.g. users and music objects, are visually missing from a (regular) tag cloud and simultaneously the respective ranking is also omitted. Therefore, from a social music tagging system there is a demand for a more powerful grouping and representation of information based on these three entities (i.e., tags, users, music objects), in which also a ranked list should be provided. Instead of just using tags, here we propose an enriched representation of the three entities by embedding them into a group called *Tag-Aware* group, in which the results are provided in ranked lists and each group refers to a certain music topic.

Next, an example of a Tag-Aware group referring to Beatles and similar artists is provided, whereas further discussion about this group follows in the results' subsection. Figure 2 presents four clouds, i.e. user, tag, song and artist clouds, quite opposite of Figure 1, where only one tag cloud appears. Evidently, a Tag-Aware group provides extra information about users and specifies correlated music objects. As social music tag systems are built upon humans, the extra information obtained from a Tag-Aware group becomes important. Having in mind a cloud as that of Figure 2, in Last.fm these users are called *neighbors*. While each user needs to discover users with similar preferences, the system should provide a convincing explanation to explain why they are correlated and which their respective links are. For example, if user "Vincent" prefers songs "You Never Give Me Your Money" and "Sun King", he would discover that the users "The Quarrymen" and "Xenob" have similar taste, since they tagged these songs as "Classic Pop" and "Rock and Roll" (note that user nicknames are not drawn from our dataset but have been randomly selected for the sake of the example). This way "Vincent" would understand why users in this group are relevant to his preferences. At the same time, a Tag-Aware group provides additional information about similar artists, while representing songs as links between them, since the linkage between the artist and the song cloud is based on the fact that all of them correspond to a specific music topic. Parallel, a user can observe which of these items are more relevant to his preferences, since each cloud contains results in a ranked form.

Figure 2. A Tag-Aware group based on artist "The Beatles" and similar artists.
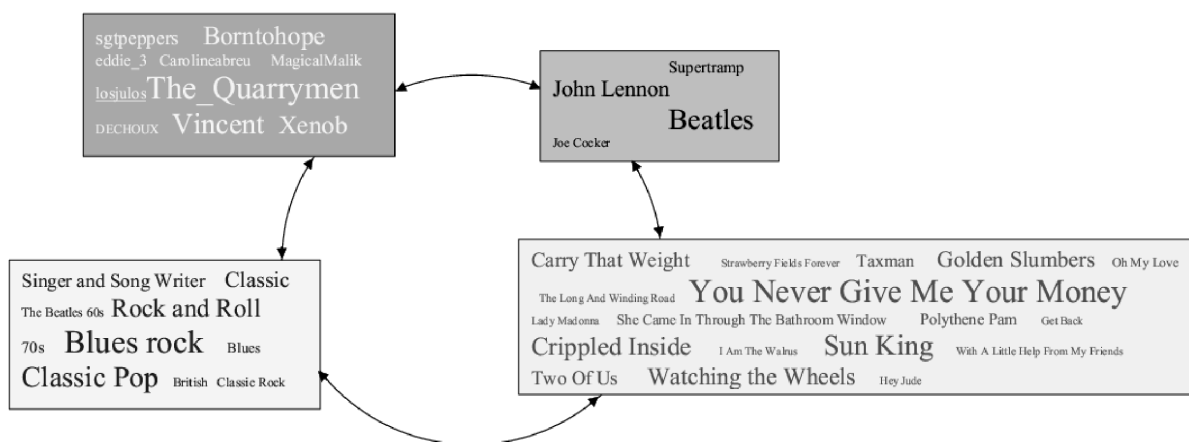
The Tag-Aware group production is based on the three dimensions: users, tags and music objects. To model these dimensions the proposed method uses multidimensional arrays, called *tensors*. An M-way or M-order tensor is an element of the tensor product of M vectors, each one with its own coordinate system. A first-order tensor is a vector, a second-order tensor is a matrix, whereas tensors of order three or higher are called higher-order tensors. This way, all dimensions are taken into account in the Tag-Aware group's generation process. A multilinear Algebra method called *Parallel Factors Decomposition* (PARAFAC) is applied to this tensor. An overview for tensor decompositions and applications appears at Kolda & Bader (2008).

In mathematics, multilinear algebra extends the methods of linear algebra. More specifically, just as linear algebra is built on the concept of *vectors* and develops the theory of vector spaces, multilinear algebra builds on the concept of tensors and develops the theory of *tensor spaces*. A vector space is a mathematical structure formed by a collection of vectors, which may be scaled and added. In particular, a tensor space assigns a tensor to each point of a Euclidean space. While applying PARAFAC to a tensor, a ranking based on the three considered dimensions is achieved. The proposed approach produces ranked lists, similar to the HITS algorithm developed by Kleinberg (1999), a link analysis algorithm for Web pages. It determines two values for a page: its authority that estimates the value of the page content, and its hub value that estimates the value of its links to other pages.

In this paper we propose an algorithm for building Tag-Aware groups, which aims at: (a) high order ranking, and (b) topic discovery. Our method develops the tensor model with data triplets of users, songs and tags gathered from Last.fm. On this real dataset we apply PARAFAC to factorize the tensor model and to generate *factors*, a kernel step of our algorithm, where each factor represents a Tag-Aware group (i.e. a meaningful division of the data). By summarizing these factors, the initial tensor can be constructed. In particular, the resulting Tag-Aware groups refer to a certain music topic, such as an artist, a mood or a genre description, while performing ranking to the three resulted lists (tags, users and music objects lists). Using this real dataset, experimentally we show the effectiveness of the proposed approach, since the two goals are achieved. More precisely, our contribution provides ways to:

- demonstrate the rich representation provided by Tag-Aware groups in an effective way,
- produce Tag-Aware groups, each of them corresponding to a specific music topic, such as an artist, a mood or a genre description, while the tag clouds are those which describe the respective topic,
- link tags, songs, artists and users inside each group and offer a transparent explanation for their relevance degree to explore similar preferences, since all clouds are correlated to the respective music topic, and
- face the vocabulary problem, a main challenge for social tags in Music Information Retrieval, by grouping synonymous tags together in music concepts.

The rest of this paper is organized as follows. The next section describes related work for implemented algorithms based on tags and how building Tag-Aware groups differs. In the following sections Tag-Aware groups are defined, and the problem description is provided. Next, a section introduces a methodology for producing Tag-Aware groups, whereas its next section reports the experimental results. The last section concludes this paper and proposes future work.

RELATED WORK

Symeonidis et al. (2008) introduce a method to generate personalized recommendations of musical items. They construct a three order tensor, to model altogether users, tags and musical items. Consequently, they proceed to unfold the tensor, where three new matrices are built. They apply SVD (Singular Value Decomposition) in each matrix and then the resulting tensor is built. In this resulting tensor, the latent components that govern the associations among the triplets user-tag-items can be discovered. Thus, the musical items can be recommended according to the captured associations. That is, given a user and a tag, the purpose is to predict whether and how much the user is likely to label with this tag a specific music item. In particular, considering a specific triplet in form (user, tag, music item), the likelihood that user1 will tag with tag1 the music item1 equals to the value in the reconstructed tensor at the respective coordinates (user1,tag1,music item1). In our approach we built a three order tensor to model users, tags and items, in an analogous manner to the initial step of the previous described method. Next, instead of reconstructing the tensor and discovering the respective associations, as the previous method do, we decompose the tensor into a number of factors, where each factor corresponds to a meaningful division of the data and describes a certain music topic. This way, for each factor we afford ranked lists of each entity (user,tag,item), where the tags are used to describe the topic of the respective factor. We should mention that the goal of our implementation is not to recommend musical items, as Symeonidis et al. (2008) do, but to provide a tag-aware group corresponding to a certain musical topic, consisting of three ranked lists, as HITS algorithm does, developed by Kleinberg (1999), where a search term in web can result to a ranked list of web sites as answers.

Karydis et al. (2009) focus on similarity-based clustering of tagged items. They try to provide an establishment of user profiles and to discover topics. They model the items, the tags on them and the users who assigned the tags in a multigraph structure. To discover clusters of similar items, they extend spectral clustering, an approach successfully used for the clustering of complex data, into a method that captures multiple values of similarity between any two items. Their scope is to provide a new method being superior to conventional spectral clustering that ignores the existence of multiple values of similarity among the items. Their method can be applicable to a dataset consisting of tags, temperature, pressure values or location names. In contrast to our method, their scope is to focus on spectral clustering. Instead, given a set of triplets {user,tag,music item}, our scope is to discover music topics and to provide ranking lists for each set of users, tags, songs and artists.

Wu et al (2006) also propose to mine latent topics based on the co-occurrence of triplets in the form {user, web page, tag}. Their algorithm input is the triplets and the output is a set of ranked lists. Each derived list refers to a certain topic of a web page. In contrast to our proposed algorithm, which is based on the PARAFAC algorithm, their method is based upon a statistical model for co-occurrence data, proposed by Hofmann & Puzicha (1998). Their method could not be applied in our music dataset, consisting of triplets {user, song, tag}, because our dataset produces a sparse model. Therefore, a sparse tensor model is required as input of our algorithm. To apply a method having as input the sparse tensor model, the proper choice is PARAFAC instead of the method described in Hofmann & Puzicha (1998). Further details about the sparse tensor can be found in subsection Sparse Tensor Model.

Pampalk (2001) demonstrates Islands of music, a visual representation for clustering listeners by tags. In particular, to derive such a Representation: (a) Last.fm listeners are randomly sampled, each one represented by a tag cloud, (b) 2000-dimensions tags are reduced to 120 dimensions using SVD, (c) k-means clustering is used to extract 400 prototypical listeners, and

(d) self organizing maps are embedded to create the Islands. The position of each Island depends on the relations between tags, whereas a path is provided to travel from an Island to another. In our approach, Tag-Aware grouping contains information about similar tags and artists as Islands do; moreover, it adds links inside the group with certain songs, tags and users based on their correlation to the music topic. This enhancement is achieved due to the fact that Tag-Aware grouping pays attention on the three dimensions: users, tags and music objects, while performing ranking in each dimension.

Levy & Sandler (2007) notice that tags yield better results than metadata or genres, because they are highly effective in capturing music similarity. In particular, they succeed in creating a browse-by-mood interface for a psychologically-motivated two dimension subspace representing musical emotion. In an analogous manner, we accomplish to produce a Tag-Aware group that refers to a mood description. Additionally, except for topic discovery, such as a mood description, we also perform a three way ranking.

Levy & Sandler (2008) describe a method with latent semantic models to reduce the disadvantages of the vocabulary problem (to be described in the next section). The main algorithm is to project a tag-query into a latent semantic space and discover the closest tags based on semantics. LSA (Latent Semantic Analysis) by Deerwester et al. (1990) assumes that there is some "latent" structure that relates the items and defines a semantic space. Items that are close together in this space have tags with similar meanings, even if they do not share any actual tags. Previously, some strategies to address many of these shortcomings were described by Guy & Tonkin (2006). Building Tag-Aware groups acts in a very similar way, since each group corresponds to a specific music topic or latent space. Moreover, each time other features are consider than tags, more precisely, users, songs and artists.

On the other hand, to accomplish Tag-Aware grouping we apply Parallel Factors decomposition (PARAFAC), which is a powerful tool for real-world applications. We note that Kolda et al. (2005) have studied PARAFAC related work for web analysis. They succeed to use multi way data representations and tensor decompositions for web search and related tasks. They model a sparse tensor (sparse because the vast majority of elements are 0s in the tensor) with three dimensions: the first two dimensions represent the web pages, whereas the third one adds the anchor text. Subsequently, a three way Parallel Factors (PARAFAC) decomposition of the web graph is applied to produce groups, with URLs associated with prominent topics. In our proposed method, after constructing the tensor by modeling the triplets {user, song, tag}, we apply PARAFAC decomposition of the tensor to group tags, music items and users in music topics.

PROBLEM DESCRIPTION

In this section we define the Tag-Aware grouping and provide the problem description. A Tag-Aware group is defined to a group of three ranked lists: user, tag and song list. Each group corresponds to a specific discovered topic, and its description is provided by the respective tag cloud. More precisely, the two goals are: (a) to perform ranking to three dimensions in each group, and in parallel (b) to discover a music topic, which corresponds to each group's topic.

A Tag-aware group is a visual representation using three dimensions: users, tags and music objects. In particular, in our method we consider songs and artists as music objects. This way, a Tag-Aware group consists of four clouds for users, tags, songs and artists. They have the ability to correspond to many categories of music topics. In general, collaborative or personal playlists (tag radio), tag groups, and muti-tag searching based on certain preferences comprise

motivations for users to tag, whereas often tagging frequencies are arranged by category. Tags can be organized into 9 basic categories: genre, locale, mood, opinion, instrumentation, style, misc (e.g. "name of composers"), personal (e.g. "seen live") and organization (e.g. "check out"). Obtained from Lamere 2008, Table 1 shows that the dominant category for Last.fm is genre. Analogously, as shown later, Tag-Aware groups may refer to these categories of music topics. Apparently, the created groups may belong to more than one category. For instance, we show that the generated group based on "The Beatles" belongs to category misc as a composer based group and to category genre as Classic pop based group. Despite the embedding of more than one category into a group, the proposed method of Tag-Aware grouping proved to be very effective.

Table 1. Tagging frequencies by category.

Tag Type	Frequency	Examples
Genre	68%	heavy metal, punk
Locale	12%	French, Seattle, NYC
Mood	5%	chill, party
Opinion	4%	love, favorite
Instrumentation	4%	piano, female vocal
Style	3%	political, humor
Misc	3%	Coldplay, composers
Personal	1	seen live, I own it
Organizational	1	check out

Next, aiming at ranking and topic discovery, we would like to introduce some examples to demonstrate the necessary logical steps, starting at one dimension and ending at three dimensions. In case of one dimension, eigenclusters can be produced, since SVD can be applied to e.g. a document matrix, each of them usually corresponding to a topic. In analogous manner, in case of two dimensions, the scores in each eigenvector perform the respective ranking and a topic can be discovered, while applying LSA to a two-dimensional matrix. In our method we face the case of three dimensions. In particular, we perform decomposition to N factors of the three-dimensional tensor and the produced eigenvectors provide ranking in each dimension (in our case the 3 dimensions are: user, tag and song). Each eigenvector-factor should refer to a topic.

Implemented algorithms based on tags find social groups with shared interests by generating user profiles from tagging behavior. For example, Last.fm groups locate people with common interests not necessarily related to music, such as "ear-chopped-off painter" fans, "car enthusiasts", and people that clearly like meticulous descriptions of their behavior. It is possible to listen to a "group or tag radio", but unfortunately sometimes this is a disappointing experience. This is due to the fact that the playlist was generated by averaging the tastes of the group members. For instance, by considering the average listener's behavior, it is observed that members of the "saxophonists" group listen to far more "Sonny Rollins" than others. Although, they quite often listen to "The Beatles", this is less important as everyone does so often. The average tastes are considered because the number of co-occurrences between two tags (uses of the tags in the same item) is calculated, such as only a high co-occurrence value can suggest a strong similarity between tags. Our method of Tag-Aware grouping ignores the average tastes,

but instead implements a method that resembles to the weighting scheme of term frequency - inverse document frequency tf ·idf (to be mentioned later). In particular, we avoid using the often applied tags in the groups' generation process. Furthermore, we calculate each item's weight in each cloud, scaled to the analogous font size to represent its strength inside the group.

A main challenge for social tags in music information retrieval is the vocabulary problem, which actually has many aspects, provided below:

- Synonyms of tags are often met, e.g. "female vocalists", "female voices" or "female singers".
- Ambiguity and polysemy are problems for tags as well, e.g. "love", could mean music a user loves or music about love.
- The vocabulary problem augments with spelling errors, multi-lingual tags or by using wrong words e.g. "r&b", "r and b".
- Meanwhile some tags may be too subtle for current systems to distinguish, e.g. "power metal" versus "speed metal".

A general definition of the vocabulary problem has been described by Furnas et al. (1987), whereas detailed information can be found in Lamere's survey (2008). The final outcome in the literature is to implement tag grouping to confront all these issues. In this direction, we show experimentally that tags with the same meaning are included to the same tag cloud and we identify their linkage through Tag-Aware grouping. Furthermore, negative tags like abuse tags or tags with no music perceptual description, e.g. "like that song" or "Lazy Eye", may reduce the effectiveness of social tags methods. Since, in general, these tags occur less frequently, in a preprocessing step we remove the negative tags before the process of Tag-Aware grouping starts.

PROPOSED APPROACH

In this section we describe the main algorithm for Tag-Aware grouping, which consists of 3 steps. The first step is to create the sparse tensor model, the second is to apply PARAFAC onto the sparse tensor to produce Tag-Aware-groups (factors), and, finally, the third one is to generate user, tag, song and artist clouds for each group.

Sparse Tensor Model

As mentioned, our initial step is to create the sparse tensor model based on triplets of the form {user, song, tag}. A tensor is a multidimensional array. More formally, an M-way or M-order tensor is an element of the tensor product of M vectors, each one with its own coordinate system. A first-order tensor is a vector, a second-order tensor is a matrix, whereas tensors of order three or higher are called higher-order tensors. In our approach, a three dimensional sparse tensor model is created, where the three dimensions represent users, songs and tags, respectively. The 1s in a tensor stand for the existence of a triplet and 0s for the absence. Our tensor is sparse since the vast majority of elements are 0s. Figure 3 (spaces denote 0s) shows an example modeling the following 7 triplets:

- {user1, song1, alternative rock}, {user2, song1, alternative rock}
- {user1, song1, 90s}, {user1, song3, 90s}
- {user4, song2, seen live}
- {user1, song2, rock and roll}, {user3, song2, rock and roll}

This example set consists of 5 distinct users, 4 distinct songs and 5 distinct tags. In Figure 3, 5 arrays are illustrated because of the 5 distinct tags, and in each array there are 5 elements in the

x-axis because of the 5 distinct users (u1, u2, u3, u4, u5) and 4 elements in the y-axis because of the 4 distinct songs (s1, s2, s3, s4). For example u1 and s1 note user 1 and song 1 respectively, and the upper left 1 inside the bottom array in Figure 3, labeled as alternative rock, notes the triplet {user1, song1, alternative rock}.

Figure 3. A sparse tensor of 7 triplets {user, song, tags}.
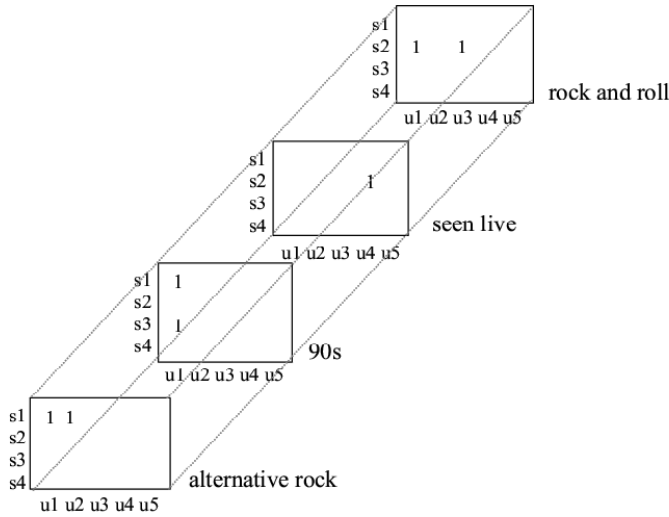
Matrices or tensors are denoted by boldface capital letters, e.g. $\mathbf{A}$, whereas vectors are denoted by boldface lowercase letters, e.g. $\mathbf{a}$. The element (i,j,k) of a third-order tensor $\mathbf{A}$ is denoted by $\mathbf{A}_{ijk}$. The *n*-th element in a sequence is denoted by a superscript in parentheses, e.g. $\mathbf{a}^n$ denotes the *n*-th matrix in a sequence, which is equal to the vector with the coordinates of the *n*-th dimension of tensor $\mathbf{A}$. Assuming these notations, let $\mathbf{X}$ denote the adjacency tensor of all triplets, defined as:

$$\mathbf{X}_{ijk} = \begin{cases} 1 & \text{if user } i \text{ tagged song } j \text{ with tag } k \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Before proceeding to the PARAFAC step, it is important to introduce two terms: "outer product" and "tensor rank". Given a vector $\mathbf{k}=(k_1,k_2,\ldots,k_m)$ with *m* elements and a vector $\mathbf{b}=(b_1,b_2,\ldots,b_m)$ with *n* elements, their outer product $\mathbf{k} \circ \mathbf{b}$, denoted by symbol $\circ$, is defined as the $m \times n$ matrix $\mathbf{A}$ obtained by multiplying each element of $\mathbf{k}$ by each element of $\mathbf{b}$:

$$\mathbf{k} \circ \mathbf{b} = \mathbf{A} = \begin{pmatrix} k_1 b_1 & \cdots & k_1 b_n \\ \vdots & \ddots & \vdots \\ k_m b_1 & \cdots & k_m b_n \end{pmatrix} \quad (2)$$

In general, the outer product for any two tensors $\mathbf{Q}$ and $\mathbf{W}$, a result tensor $\mathbf{R}$, each of whose indices corresponds to an index of $\mathbf{Q}$ or an index of $\mathbf{W}$ and each of whose components, is the

product of the component of $\mathbf{Q}$ and the component of $\mathbf{W}$ with identical values of the corresponding indices.

The term "rank" of a tensor $\mathbf{X}$ is defined as the smallest number of rank-one tensors that generate $\mathbf{X}$ as their sum. An M-way tensor is rank-one if it can be written as the outer product of M vectors based on Equation (2), i.e.,

$$\mathbf{X} = \mathbf{a}^{(1)} \circ \mathbf{a}^{(2)} \circ \ldots \circ \mathbf{a}^{(M)} \quad (3)$$

This means that each tensor element is the product of the corresponding vector elements. For example, if our tensor is a third-order rank-one tensor then it would equal to:

$$\mathbf{X} = \mathbf{u} \circ \mathbf{s} \circ \mathbf{t} \quad (4)$$

while the following relations must hold:

$$\mathbf{u} = \sum_{i=1}^{N} \mathbf{u}^{(i)} \quad (5)$$

$$\mathbf{s} = \sum_{i=1}^{N} \mathbf{s}^{(i)} \quad (6)$$

$$\mathbf{t} = \sum_{i=1}^{N} \mathbf{t}^{(i)} \quad (7)$$

where vectors $\mathbf{u}$, $\mathbf{s}$, $\mathbf{t}$ represent the set of users, songs and tags, respectively. The size of vector $\mathbf{u}$ equals the number of distinct users, the size of vector $\mathbf{s}$ equals the number of distinct songs, whereas $\mathbf{t}$ the number of distinct tags. As expected, our modeled third-order tensor is a tensor of more than one rank. To express our tensor $\mathbf{X}$ as a sum of small number of rank-one tensors, we have to apply PARAFAC.

Parallel Factors Decomposition - PARAFAC

The second step of our method, i.e. the Parallel Factors decomposition, comprises the kernel of the proposed approach. PARAFAC is applied to the created sparse tensor to produce N factors corresponding to the number of distinct Tag-Aware groups. PARAFAC can be considered as a higher order generalization of the matrix singular value decomposition (SVD) and principal component analysis (PCA). Its goal is to factorize a tensor into a sum of component rank-one tensors. Given our third-order modeled tensor $\mathbf{X}$, PARAFAC scope is to express it in the following form:

$$\mathbf{X} \approx \sum_{i=1}^{N} \mathbf{w}^{(i)} \mathbf{u}^{(i)} \circ \mathbf{s}^{(i)} \circ \mathbf{t}^{(i)} \quad (8)$$

where each vector is normalized to length one using the weights of vector $\mathbf{w}$.

PARAFAC implementations may vary. In our case, we embed the Alternative Least Squares (ALS) method. ALS method computes an estimate of the best rank-N PARAFAC model of a tensor **X** using an alternating least-squares algorithm (where N is the number of factors). The approximation $\approx$ in Equation (8) refers to the fact that there is a residual error between the estimated tensor and the real tensor. The main scope of the ALS algorithm is to minimize this error by repeating the procedure until some convergence criterion is satisfied. For more details readers should refer to Harshman (1970), Faber et al. (2003) or Tomasi & Bro (2005). The ALS procedure for an M-way tensor assumes that the number of factors N must be specified a priori. There is no straightforward algorithm to determine an optimal N value; in fact the problem is NP-hard according to Haastad (1990). The number of factors is estimated empirically based on the level of distinctiveness between the groups and the quality of each group. Also, the degree as to how effectively the tags describe the songs in each group is considered.

PARAFAC offers a three way decomposition that yields users, songs and tags. Figure 4 presents a N-component PARAFAC model, where the created sparse tensor **X** is expressed as the sum of N factors (groups), where $\mathbf{u}^i$, $\mathbf{s}^i$ and $\mathbf{t}^i$ are the i-th components in each factor. PARAFAC delivers a score for all items (users, tags and songs) in each dimension for every factor. Thus, for each factor a ranked list for each dimension is provided, having the respective items in descending order according to their scores. This way, we obtain a three way ranking. We should also mention that using ranking in a three dimensional way, becomes a very important issue, since we filter out inappropriate items (users, tags, songs and parallel artists), while keeping only the best of them based on their score (note that artists' score is calculated according to their respective songs). Thus, the remaining items which are faced as possible noise do not belong to the corresponding factors (Tag-Aware groups) and the proposed method becomes more effective.

Figure 4. Illustration of the PARAFAC model.
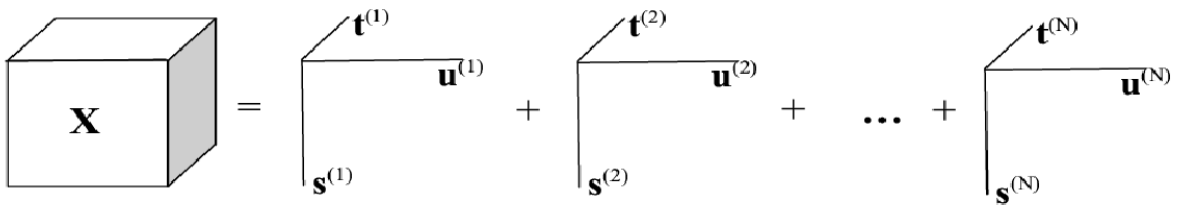
Generating Clouds

After modeling the triplets to the sparse tensor and applying PARAFAC, we retrieve the ranked lists based on the calculated scores. As mentioned, the effectiveness of the proposed approach will decrease if all the items of each ranked list for every factor are taken into account. Thus, we consider the best 20 items from the lists, where items could be users, tags or songs. We produce user, tag, song and artist clouds for each of N factors e.g. an artist cloud inside a group is a cloud with the respective artists, where the strength of a certain artist into the group is being represented with the analogous font size. In particular, the ranking position of an item depends

upon the font size in the cloud. To generate a proper visualization of each cloud, we need to take into account two crucial factors: (a) the respective position of each item (user, tag, song) based on the calculated score acquired by PARAFAC in each ranked list, and (b) the problem of the dominant tags and songs. To consider these two factors, it is necessary to map weights to each item, which represents the third step of the algorithm. Therefore, before proceeding to the third step, it is first necessary to introduce the problem with the dominant tags and songs.

According to Lamere (2008) users apply some tags more often than others and, thus, a few tags and songs may dominate. This becomes a problem for Tag-Aware grouping, since we aim at descriptive factors with representative tags and songs. There is no music perceptual meaning to take into account tags and songs that are often applied from users. To overcome this problem, first, we compute the number of tag and song occurrences in the resulting factors. Then, we calculate the weights of tags and songs by inverting their number of occurrences in each factor. Thus, each item in a cloud is scaled to the analogous font size. The motivation for the proposed weighting scheme, as a high order ranking method, comes from the fact that it effectively acts similarly to $tf \cdot idf$ in the popular web ranking method HITS algorithm (Kleinberg 1999), trying to evaluate how important a link is to a ranked list in a search query on the internet.

Finally, we must mention that a tag, a user or a song cloud is generated by applying our algorithm to the gathered triplets. We consider a threshold for the number of occurrences of a certain user, tag or song into the resulting factors. If a user's, tag's or song's occurrences exceed this threshold, we remove it from the cloud. This way, as depicted in Figure 2 (or later in the following figures), it is possible to retrieve different number of tags or songs for each cloud into the corresponding factors. As there is a lack of artists in the gathered triplets, the artist cloud is provided in a different way. Based on the fact that a song is performed by a certain artist, we add the artist to the artist cloud, with the font size equal to font size of the respective song. As observed, sometimes in a group there exists more than one song performed from a certain artist. In this case, we keep the song having the best rank, the larger font size in song cloud and we estimate the artist's font size in the cloud according to this song.

EXPERIMENTS

In the first subsection we describe the method to retrieve the data (triplets), and a toolbox to work with sparse tensors and PARAFAC. Then, we identify the problem of the dominant tags and songs in our dataset. In the second subsection we present the experimental results.

Data Description

To evaluate our method we have chosen a real data set mined from the social music website Last.fm, using the Audioscrobbler web services during July 2008. The extracted data are triplets of the form *{user, song, tag}*. The extraction process starts with the randomly selected user " Metalidol" (note that user nicknames are replaced with distinct ids to respect privacy). The procedure continues by propagating the crawling to his neighbors in a recursive way until 733 distinct users are found. Finally, we collect 70301 *{user, song, tag}* triplets with 985 songs and 3360 unique tags. As our initial choice for the user is a "metal" listener, the vast majority of the

gathered songs belong to genre "pop", "rock" or "metal". We also mention that *negative* tags (previous defined) are removed to avoid potential problems.

Next, we run the MA Tensor Toolbox, a toolbox for sparse tensors and PARAFAC, implemented by Brett Bader and Tamara Kolda. The toolbox provides a way to store the values of each triplet to the sparse tensor and to apply the PARAFAC modified method ALS, by calling the parafac_als() function from the toolbox. As expected, few tags and songs dominate into the extracted triplets. This observation becomes apparent in Figure 5, where the frequencies of distinct songs and tags are presented. The most frequent applied tag in our triplets is "rock", having 1486 occurrences in the 70301 triplets.

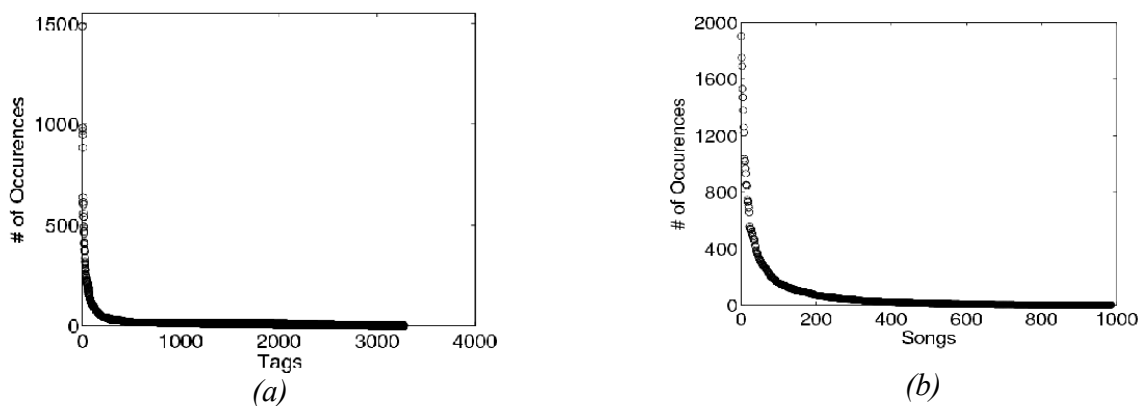*Figure 5. Frequency of distinct tags (a) and songs (b) into the crawled triplets*

(a)

(b)

Table 2. Top-10 tags and their frequencies in the collection of 70301 extracted triplets.

Tag	Freq
Rock	1486
Alternative	981
Alternative rock	966
Classic rock	949
Metal	833
Indie	637
Hard Rock	613
Progressive Rock	609
Punk	602
Electronic	558

Based on Liekens analysis, it seems that there is a spectrum of musical genres in the population of Last.fm users. The spectrum ranges from "indie" over "alternative" to "rock" and "metal", and then to "hip-hop" and "electronic" music with a sparse gap back to "indie". Tags like "rock", "alternative", "pop", "indie" are often selected by users. In our dataset, the most frequent tagged song is "Hysteria" from the album "Hysteria" by "Def Leppard" from genre "rock" with 1901 occurrences, while the second most frequent tagged song is "Bliss" from the album "Rainbow" by Maria Carey from genre "pop". To overcome the dominant tags and songs

problem, after producing the factors by PARAFAC, it is necessary to apply our implemented *tf·idf* method as described in the previous section.

Results

As mentioned in a previous section, to start PARAFAC the number of factors has to be a priori specified. According to the experimental framework of Kolda et al. (2005), the optimal number of factors achieves the best division of the dataset. Since our gathered dataset consists of music items, users and tags, the best division should be considered when each factor refers to a certain music topic. The respective tag cloud provides the description of each music topic. In particular, to ensure that each factor is consistent, irrelevant tags should not be grouped together by the proposed method. Therefore, the exploration concerns the tag clouds as the proper guide, while tuning to different number of factors R in range [1…40]. It was concluded that the optimal number of factors equals to 20. Each factor is a Tag-Aware group with the following components: user, tag, song and artist clouds. Each weight is computed as described in a previous section, and then the font size is scaled linearly by each component's weight in the Tag-Aware group.

To ensure the stability of the discovered Tag-Aware groups, in our experiments we remove or add triplets to the initial dataset. In particular, from Figure 6 it is obvious that even if the dataset size is reduced from 100% to 50%, the tag cloud of factor 9 still describes the respective music topic ("Beatles" in our case). We must mention that the tag clouds are used as a means to prove stability, because they do describe the discovered topic of each factor. To conclude, the proposed method is stable, since the eigenvectors-factors are not modified, when the dataset size is either reduced or increased. Moreover, the high-ranked tags are kept, whereas the new added tags describe appropriately the respective topic equally well. In addition to this experiment, we performed the "stability" experiments for the remaining 19 factors, where we remarked a similar performance.

Figure 6. Tag cloud transformation of factor 9 (Beatles), while reducing the dataset size to: (a) 100% (b) 90% (c) 80% (d) 70% (e) 60% (f) 50%.
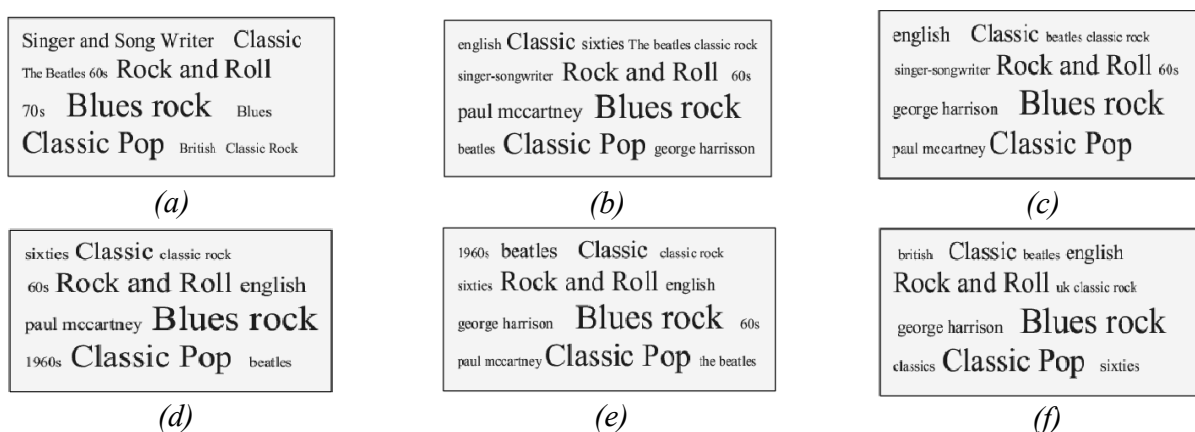
Since Tag-Aware groups analysis performs a high order ranking and a topic discovery, our evaluation can only be performed by examining: (a) if the respective ranked tag list (tag cloud)

properly describes the certain music topic, and (b) if the coherence of each group is maintained, an evaluation method similar to the evaluation method of the TOPHITS algorithm (Kolda et al. 2005).

In the sequel, we present three factors Tag-Aware groups generated by the proposed method. These factors represent three different types of Tag-Aware groups. Figure 2 shows factor 9, a Tag-Aware group which refers to the music topic of "The Beatles". It also contains their similar artists, the relevant songs, along with the users of similar preferences. We note that in this group "John Lennon" is the second dominant artist, not only as a member of "The Beatles", but as a solo artist as well. Also, "Supertramp" is correlated to "The Beatles" as an English progressive rock and pop band. "Joe Cocker" is also connected to "The Beatles", as he is an English rock/blues singer who became popular in the 1960s, and is most known for his gritty voice and his cover versions of popular songs, particularly those of "The Beatles". Their correlation is noted by tags like "classic pop", "blues rock", "british" etc. On one hand, we achieved to discover a Tag-Aware group and provide a rich representation at a glance. On the other hand, the production of such a group is a common task to solve for a system based on artist similarity and can be accomplished using other methods. More information can be found in MIREX (Music Information Retrieval Evaluation eXchange), in *Audio Artist Identification* task or in Downie (2008); however, the relevant links, users, songs and tags will be missing and moreover the respective three way ranking will also be omitted.

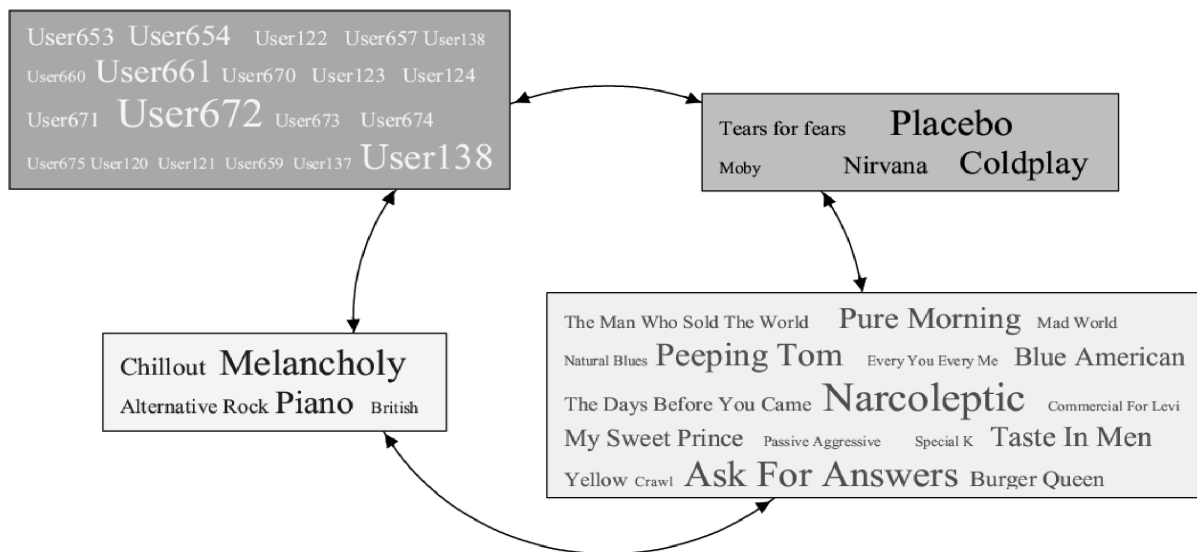*Figure 7. Factor 3, a Tag-Aware group based on artists and songs with "relaxed" music.*

Figure 7 depicts a Tag-Aware group which corresponds to a different type of music topic, such as a mood description, "relaxed" music in particular. In general, the songs of "Placebo" are relaxed music. However, a few songs of "Nirvana", like "The Man Who Sold the World", could also belong to this Tag-Aware group. This is a very difficult task to be automatically identified by a system based on genre or artist similarity, although the particular song can be evaluated as a correct result according to users' preferences for social tags, as users tend to tag the songs of "Placebo" and the corresponding song of "Nirvana" as "chillout" music. For example, if the proposed method was embedded into a recommendation system to perform ranking and topic

searching, then a user could pose a "chill out" music request (see Chen & Chen 2005 for more information on recommendation systems). Then the list retrieved from the system would consist of songs, the vast majority of which, would be songs by "Placebo" and "Coldplay", i.e. the dominant or the higher ranked artists inside the Tag-Aware group, whereas a few songs by "Nirvana" and "Moby" lower ranked artists would be added, in particular "The Man Who Sold the World" and "Natural Blues" respectively. This outcome could be also reached by a select mood system, as Levy & Sandler (2007) claim. We note the absence of a system providing simultaneously artist similarity, like the previous factor 9 and mood – selection like the current factor 3. Additionally, it would be very difficult for a mood selection system to create a recommendation list comprising of "Placebo" and "Nirvana". Also, while associating the two artists with the help of the song "The Man Who Sold the World", we can provide a transparent explanation to the recommendation system, since their connection is based on the certain music topic, "chillout" music. The appropriate tags establish the connections between components inside this Tag-Aware group. At the same time, a user could discover other users having similar preferences in "relaxed" music, and tune in their "radio" or explore their preferences as well. Moreover they could be based their search or explore on the most prominent, since the resulting items are in a ranked form.

Figure 8 presents factor 19, a Tag-Aware group consisting of "rock" and "metal". Correlated artists and certain songs from these two genres are included. A difficult issue comes up; an association for "rock" and "metal" artists is discovered based on user tags. It is apparent that there is a thin line between these two genres, and, thus, while observing this group we cross from one side to the opposite. In particular, genre overlapping in music is based on the genre folksonomy, described in more details by Lamere (2008). Therefore, this group illustrates a way to locate and explain the overlapping between these two genres.

Figure 8. Factor 19, a Tag-Aware group based on a mixture of "rock" and "metal".
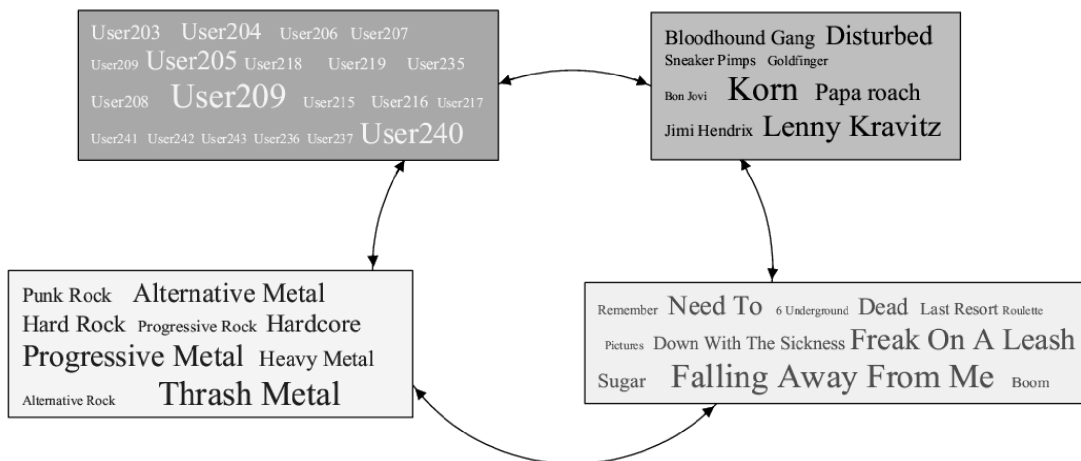
The three Tag-Aware groups or factors discussed in Figures 2, 7 and 8 are very representative results of the proposed approach applied on our dataset. Another example of group is the "latin" one, which contains users, tags, songs and artists referring to genres "flamenco", "bolero", "bugalu", "cha cha" and "cumbia". Finally, another example is the "melodic rock" group produced by "piano rock" and "symphonic rock" songs and artists, including the relevant users.

This group lies at a lower level in hierarchy of genres, under the genre "rock". More information about the hierarchy of genres can be found in Lamere's survey (2008).

We observed that our implementation is capable of dividing the initial unorganized triplets to music topics, by producing 20 concrete Tag-Aware groups. To ensure that effective results are derived, we show that each group provides a good overview and description about a certain music topic. Because the topics of each group belong to music concepts, to prove how each music item of the group is correlated to the respective music topic, we provide explanations for each resulting group based each time on a musical point of view.

CONCLUSION

We have presented an algorithm to implement Tag-Aware grouping by using multilinear algebra. The proposed approach aims at achieving ranking for the three basic elements from a social music tagging system: users, tags and music objects, and additionally each Tag-Aware group correspond to a music topic. We showed experimentally through real data that our method can be very effective, since we achieve a proper division of the data according to the discovered topics. At the same time, the included tags in each group can be very descriptive for the respective songs and artists, based on user preferences.

Three different representative examples for our method have been described. First, a Tag-Aware group which refers to a certain artist has been built; second, a Tag-Aware group relied on mood description, and, third, a Tag-Aware group with correlated-overlapping genres has been constructed. This way, Tag-Aware grouping can be used to build music similarity models. Also, the proposed method creates a multi-way and rich visualization to explore the complex music space and find new, interesting and relevant music. Finally, Tag-Aware grouping can generate transparent and explainable recommendations for their relevance, since the explanation relies on the fact that all clouds are referred to a specific music topic and help users to discover the most prominent users with similar preferences, since ranked lists are provided. This becomes important since social music tagging systems refer to humans.

An important topic for future work is the lack of transparency on the relationships between the Tag-Aware groups. These relationships could be calculated based on several criteria, e.g. artist, song or/and tag similarity between two different Tag-Aware groups. This can lead to a visual representation of the considered groups via a graph, where the connecting edges consist of three dimensions: weighted tags, artists and songs. Apparently, there is always the challenge that the generated groups have different music conceptual meaning and overwhelm the construction of such a graph. Finally, it would be possible to gather triplets from datasets having a priori specified the number of groups. This would offer a direction to our method, such as to measure the precision of our algorithm in a more formal way.

Last, PARAFAC algorithm in sparse tensors causes memory overflows during the tensor factorization process, while using a large dataset. An attempt to solve this 'blowup problem' can be found in Kolda & Sun (2008). Although, they succeeded to reduce the memory consumption, PARAFAC efficiency is questionable with respect to scalable datasets. If a more dramatic reduction is achieved, it would be feasible to apply the proposed algorithm to larger dataset.

REFERENCES

Audioscrobbler, Web Services for Last.fm, described at http://ws.audioscrobbler.com

Chen H.C. & Chen A.L.P. (2005). A Music Recommendation System Based on Music and User Grouping. *Journal of Intelligent Information Systems: Intelligent Multimedia Applications*, *24*(2), 113-132.

Deerwester S.C., Dumais S.T., Landauer T.K., Furnas G.W. & Harshman R.A (1990). Indexing by Latent Semantic Analysis. *Journal of the American Society of Information Science, 41*(6), 391-407.

Downie J.S. (2008). The Music Information Retrieval Evaluation Exchange (2005-2007): a Window into Music Retrieval Research. *Acoustical Science and Technology, 29*(4), 247-255.

Faber N.K.M., Bro R. & Hopke P.K. (2003). Recent Developments in CANDECOMP/PARAFAC Algorithms: a Critical Review. *Chemometrics and Intelligent Laboratory Systems, 65*(1), 191-137.

Furnas G.W., Landauer T.K., Gomez L.M. & Dumais S.T. (1987). The Vocabulary Problem in Human-system Communication. *Communications of the ACM, 30*(11), 964-971.

Guy M. & Tonkin E. (2006). Tidying up Tags. *D-Lib Magazine*, online article: www.dlib.org/dlib/january06/guy/01guy.html.

Harshman R.A. (1970). Foundations of the PARAFAC Procedure: Models and Conditions for an 'Explanatory' Multi-modal Factor Analysis. *UCLA working papers in phonetics, 16*, 1-84.

Haastad J. (1990). Tensor Rank is NP-complete. *Journal of Algorithms*, *11*, 644-654.

Hofmann T. &. Puzicha J. (1998). Statistical models for co-occurrence data. Technical report, A.I.Memo 1635, MIT.

Karydis I., Nanopoulos A., Gabriel H. & Spiliopoulou M. (2009). Tag-aware spectral clustering of Music Items. In *International Society for Music Information Retrieval Conference: Volume 10*, (pp. 159-164). Kobe, Japan.

Kleinberg J. (1999). Authoritative Sources in a Hyperlinked Environment. *Journal of the ACM, 46*(5), 604–632.

Kolda T. G. & Bader B.W. (2009). Tensor Decompositions and Applications. *SIAM Review, 51*(3), 455-500.

Kolda T.G., Bader B.W. & Kenny J.P. (2005). Higher-Order Web Link Analysis Using Multi-linear Algebra. In *IEEE International Conference on Data Mining: Volume 5*, (pp.242-249). Houston, TX.

Kolda T.G. & Sun J. (2008). Scalable Tensor Decompositions for Multi-aspect Data Mining. In *IEEE International Conference on Data Mining: Volume 8*, (pp.363-372). Pisa, Italy.

Lamere P. (2008). Social Tagging and Music Information Retrieval. *Journal of New Music Research, 37*(2), 101-114.

Last.fm, a social music platform, in http://www.last.fm

Levy M. & Sandler M. (2007). A Semantic Space for Music Derived from Social Tags. In *International Conference on Music Information Retrieval: Volume 8*, (pp 411-416). Vienna, Austria.

Levy M. & Sandler M. (2008). Learning Latent Semantic Models for Music from Social Tags. *Journal of New Music Research, 37*(2), 137-150.

Liekens A., Data mining music profiles, described at http://anthony.liekens.net/index.php/ Computers/DataMining, 28 March –2 April, 2007

Tensor MA toolbox by Bader B. & Kolda T., described at http://csmr.ca.sandia.gov/~tgkolda/ TensorToolbox/

Pampalk E. (2001). Islands of Music: Analysis, Organization and Visualizations of Music Archives, Master Thesis, Vienna University of Technology, Austria.

Symeonidis P., Ruxanda M., Nanopoulos A. & Manolopoulos Y. (2008). Ternary Semantic Analysis of Social Tags for Personalized Music Recommendation. In *International Conference on Music Information Retrieval: Volume 9,* (pp.219-224). Philadelphia, PA.

Tomasi G. & Bro R. (2005). PARAFAC and Missing Values. *Chemometrics and Intelligent Laboratory Systems, 75*(2), 163-180.

Wu, X., Zhang, L. & Yu Y. (2006). Exploring social annotations for the semantic web. *In Proceedings of the international Conference on World Wide Web: Volume 15*, (pp. 417-426), New York, NY.