UNVEILING MUSIC STRUCTURE VIA PLSA SIMILARITY FUSION

Jerónimo Arenas-García

Dept. of Signal Theory and Communications
Universidad Carlos III de Madrid
28911 Leganés, Spain

A. Meng, K. B. Petersen*, T. Lehn-Schiøler*
L. K. Hansen and J. Larsen

Dept. of Informatics and Mathematical Modelling
Technical University of Denmark
Denmark
*Currently at *Epital.dk*

ABSTRACT

Nowadays there is an increasing interest in developing methods for building music recommendation systems. In order to get a satisfactory performance from such a system, one needs to incorporate as much information about songs similarity as possible; however, how to do so is not obvious. In this paper, we build on the ideas of the Probabilistic Latent Semantic Analysis (PLSA) that has been successfully used in the document retrieval community. Under this probabilistic framework, any song will be projected into a relatively low dimensional space of "latent semantics", in such a way that that all observed similarities can be satisfactorily explained using the latent semantics. Additionally, this approach significantly simplifies the song retrieval phase, leading to a more practical system implementation. The suitability of the PLSA model for representing music structure is studied in a simplified scenario consisting of 10.000 songs and two similarity measures among them. The results suggest that the PLSA model is a useful framework to combine different sources of information, and provides a reasonable space for song representation.

1. INTRODUCTION

Given two songs, most people would agree that it is possible to tell if the two songs are similar or not. However, similarity between songs can be "defined" in many different ways: They may have the same beat, the same guitar sound, the same lead singer, etc. One may also extend the domain beyond the sound-based context, and state that two songs are similar if they were produced in the same year or if they are targeted to the same audience. In short, similarities among songs are many and varied.

When building music recommendation systems, one would like to integrate as much information about songs similarity as possible. This goal leads to a natural question: Given some song, does the combination of all possible similarities point to a (non-empty) set of songs? One could imagine that two different similarities are mutually excluding, meaning that they point, for any given query song, to disjoint sets of neighbors. If that were the case, then there would be little interest in combining all thinkable similarities for a single solution, since this solution would be an empty set. This question of agreement is important and touched upon in a small number of papers.

In [1], the authors study user consensus on a set of musical artists, but the variability of user evaluations is "casting doubt on the concept of a single ground truth". This somehow surprising conclusion may be a consequence of subjective user evaluations, or simply of undetected underlying user groups. This claim is supported by [2], in which the agreement of different similarities, including subjective, social and acoustic ones, is investigated. In this study, the authors find that there is an agreement between subjective and acoustic measures which is comparable to the internal agreement between subjective users.

The question of agreement between similarities is, to the best of our knowledge, not yet answered clearly by data. However, it seems obvious that, to build powerful music recommendation systems, one should try to integrate different sources of information or similarities between songs; how many such similarities one should take into consideration is unclear, but the need to fuse different sources of information seems evident.

Looking at the literature about music content-based search and retrieval systems, we can find many different solutions to how the information of the chosen features should be combined in order to build a space where similarity between songs can be measured. In [3], for instance, some low level features such as the loudness, pitch, brightness, bandwidth and harmonicity, are aggregated by the mean, variance and autocorrelation. In [4], the MFCCs are binned using a vector quantization tree in which the decision thresholds are set to maximize the mutual information between the inputs and the labels of a training set. In other approaches, such as [2, 5, 6], the data cloud of low level features is modeled using a probability distribution, typically estimated using a Gaussian mixture model (GMM). For many low-level features this is a sensible thing to do and well justified given the empirical distribution of the features. But as the feature set is expanded from, say MFCCs or zero crossing rates, to playlist co-occurrence, production year, or blog-gossip, it becomes increasingly unlikely that any practical family of distributions will suffice to model the observations, and thus to build a reasonable similarity space.

In this paper we propose a generalized framework for building music recommendation systems that are based on a combination of a number of, possibly redundant, sources of information regarding song similarity. Our approach makes use of the ideas of Probabilistic Latent Semantic Analysis (PLSA) [7, 8], which has been successfully applied in web document retrieval, including the possibility of combining heterogeneous similarity measures between documents, such as the appearance of common words or common links [8]. The basic idea is to project the songs into a space of relatively small dimension (the latent semantics) in such a way that all observed similarities can be satisfactorily explained using the latent semantics. In this way, the "overall" distance between two songs can be determined from the latent semantics only. As in

the web document retrieval case, we will see that the application of PLSA to build music models simplifies the implementation of music recommendation systems, significantly reducing the computational burden of the song retrieval phase.

This analogy between songs and documents can be regarded as a purely technical convenience, but might also start a new line of thinking in which songs aspects are interpreted as "words". In any case, if this is a fruitful analogy, future research could investigate music using the elaborated machinery already deployed for web-mining, and apply the suggested tool for boosting the performance of music recommendation systems.

The rest of the paper is organized as follows: Section 2 introduces the different levels of representation for music analysis that will be used throughout the paper, while Section 3 reviews the formulation for the Generalized version of PLSA that can be used for the design of music recommendation systems that simultaneously consider multiple measures of similarity between songs. Different algorithms can be used to adjust the parameters of the PLSA model; in this paper we consider Non-negative Matrix Factorization (NMF) algorithms as described in Section 4. In Section 5 we evaluate the possibilities of the approach by carrying out experiments in a simplified scenario, and in Section 6 we extract some conclusions about the work, and discuss lines for future research.

2. MUSIC REPRESENTATION LEVELS

In this section, we introduce some notation, and define the different levels for music representation that will be considered along the paper:

- *Songs*: This level corresponds to the pieces of music that are known by the system. The set of all songs will be denoted as $\{s_l\}_{l=1}^{L}$.
- *Similarities*: Each of the different criteria that we use to measure distances between songs. In this paper, we will consider that each similarity criterion is characterized by a set of clusters or groups (e.g., $c_j^{(k)}$ for the jth group associated to the kth similarity), and that each song s_l is defined by a certain distribution over the clusters of each similarity criterion, subject to restrictions:

$$P(c_j^{(k)}|s_l) > 0, \quad \forall j,k$$

$$\sum_{j=1}^{n_k} P(c_j^{(k)}|s_l) = 1, \quad \forall k$$

  where n_k is the number of groups along the kth similarity dimension.

  In a real situation, there are different ways in which we can estimate this similarity information. For instance, when the song recordings are available, we can directly extract "sound features" from the music waveform (e.g., zero crossing rate, MFCCs, spectrogram-based features, etc) and carry out a hard or soft clustering in the resulting feature space. There are also situations in which we have access to metadata information (e.g. music genre) that can be interpreted as the labels of a multi-class classification problem. In such cases, we can either use the class membership information provided by the metadata or, alternatively, the outputs of a classification system operating on the "sound features" to predict the class membership probabilities associated to each song. Other sources of information, such as web-based search or playlist order can also be exploited.
- *Latent Semantics*: This is the representation space where songs are projected to get a compact representation. As with songs, each semantic group, $z_i, i = 1, \dots N$, is represented by a certain distribution along each similarity dimension. These latent semantics are not known a priori, but have to be determined from the set of songs and their representations along the different similarity criteria.

The basic hypothesis we are accepting here is that it is possible to find latent semantics that are able to simultaneously explain all the available similarity information. In the next section we explain how such a model can be obtained.

3. GENERALIZED PLSA FOR MUSIC SIMILARITIES FUSION

Our model for modeling music structure is based on the Probabilistic Latent Semantics Analysis (PLSA) that has been successfully used in the analysis and retrieval of text documents [7]. The analogy is as follows: songs (documents) can belong to a set of hidden and unknown states or groups, $\{z_i\}_{i=1}^{N}$, i.e., the latent semantics. We assume soft membership, so that each song can be represented as distribution over the different hidden states, thus satisfying the constraint:

$$\sum_{i=1}^{N} P(z_i|s_l) = 1 \tag{1}$$

where $P(z_i|s_l)$ is the probability that song s_l belongs to the semantic group z_i.

Next, each (hidden) group of songs is characterized by some cluster distribution over each of the similarity dimensions we are considering, i.e.,

$$z_i : P(c_1^{(k)}|z_i), P(c_2^{(k)}|z_i), \dots, P(c_{n_k}^{(k)}|z_i)$$

Of course, each of these distributions have to be a real distribution, i.e.,

$$\sum_{j=1}^{N_k} P(c_j^{(k)}|z_i) = 1 \tag{2}$$

Now, we can express $P(c_j^{(k)}|s_l)$ through the expansion

$$P(c_j^{(k)}|s_l) = \sum_{i=1}^{N} P(c_j^{(k)}|z_i, s_l) P(z_i|s_l) \tag{3}$$

$$= \sum_{i=1}^{N} P(c_j^{(k)}|z_i) P(z_i|s_l) \tag{4}$$

where we are assuming that all the knowledge about the cluster distribution is propagated via the semantic groups.

As it is usual in the PLSA approach, we assume that $P(c_j^{(k)}|s_l)$ are unknown, but we have access to some estimations of these quantities that we will denote as $\tilde{P}(c_j^{(k)}|s_l)$. Then, for each similarity criterion, we would like to find the set of probabilities $P(c_j^{(k)}|z_i)$ and $P(z_i|s_l)$ that maximize the likelihood of our observations,

$$\prod_{j,l} P(c_j^{(k)}|s_l)^{\tilde{P}(c_j^{(k)}|s_l)}$$

Finally, taking logarithms, and introducing the decomposition model for $P(c_j^{(k)}|s_l)$ [Eq. (3)], we get the following set of log-likelihoods to be maximized:

$$L_k = \sum_{j,l} \tilde{P}(c_j^{(k)}|s_l) \log \left(\sum_{i=1}^{N} P(c_j^{(k)}|z_i) P(z_i|s_l) \right), \quad (5)$$

for $k = 1, \ldots, K$, K being the total number of available similarities.

Note that the different log-likelihoods for different similarities cannot be maximized independently since they are coupled through terms $P(z_i|s_l)$. As in [8], we propose to maximize the following combined log-likelihood function

$$L = \sum_{k=1}^{K} \alpha_k L_k \quad (6)$$

where α_k, satisfying $\sum_k \alpha_k = 1$, measures the importance assigned to the kth similarity. Note that, proceeding in this way, we can adjust models that are specially good at explaining different similarities (for instance, we can obtain a model which is specially good at explaining similarity in the co-play dimension, while still integrating some of the information in the other similarity dimensions). The maximization of this mixed log-likelihood w.r.t. $P(c_j^{(k)}|z_i)$ and $P(z_i|s_l)$ can be carried out using different methods, such as versions of the Expectation-Maximization algorithm, or the Non-negative Matrix Factorization (NMF) approach discussed in Section 4.

Song retrieval procedure

Once the PLSA model has been trained, we can use the latent semantics for song retrieval using very compact expressions. For instance, the probability that any song in the dataset should be recommended given some query song, s_q, can be calculated using

$$\begin{aligned} P(s|s_q) &= \sum_{i=1}^{N} P(s|z_i, s_q) P(z_i|s_q) \\ &= \sum_{i=1}^{N} P(s|z_i) P(z_i|s_q) \\ &= \sum_{i=1}^{N} \frac{P(z_i|s)P(s)}{P(z_i)} P(z_i|s_q) \end{aligned} \quad (7)$$

where we have used the assumption that song probability distribution propagates through the latent semantics in replacing $P(s|z_i, s_q)$ by $P(s|z_i)$, and where $P(s)$ is the a priori probability of each song, that can be estimated, e.g., using a measure of song popularity. Finally, the a priori probabilities assigned to each latent semantic can be precalculated using

$$P(z_i) = \sum_l P(z_i|s_l) P(s_l), \quad i = 1, \ldots, N \quad (8)$$

Note that the complexity in evaluating (7) grows linearly with the number of latent semantics. This is a very important advantage with respect to the case in which similarity clusters were considered directly. Effectively, if the expansion were made with respect to all clusters in all similarities, we would get

$$P(s|s_q) = \sum_{j_1 \cdots j_K} P(s|c_{j_1}^{(1)}, \ldots, c_{j_K}^{(K)}) P(c_{j_1}^{(1)}, \ldots, c_{j_K}^{(K)}|s_q) \quad (9)$$

In this sense, we can interpret the PLSA model as a bottleneck that is reducing the complexity of the problem from all possible combinations of clusters ($\prod_i n_{c_i}$) to just the number of hidden states (N). Nevertheless, maximization of combined likelihood (6) assures that the latent semantics retain as much information as possible about the different similarity dimensions that are taken into account.

Constrained song retrieval

The fact that PLSA is a probabilistic framework provides a lot of flexibility when carrying out search tasks. For instance, imagine that we want to constrain the search to one of the clusters. Then, we can refine the search as follows:

$$\begin{aligned} P(s|s_q, c_j^{(k)}) &= \sum_{i=1}^{N} P(s|z_i, s_q, c_j^{(k)}) P(z_i|s_q, c_j^{(k)}) \\ &= \sum_{i=1}^{N} P(s|z_i) P(z_i|s_q, c_j^{(k)}) \end{aligned} \quad (10)$$

where we have used the fact that cluster information also propagates through the hidden states. Note also that it is our assumption that

$$P(c|z, s) = \frac{P(z|c, s)P(c|s)}{P(z|s)} = P(c|z),$$

so that we have also

$$\begin{aligned} P(s|s_q, c_j^{(k)}) &= \sum_{i=1}^{N} \frac{P(z_i|s)P(s)P(c_j^{(k)}|z_i)P(z_i|s_q)}{P(z_i)P(c_j^{(k)}|s_q)} \\ &= \sum_{i=1}^{N} \frac{P(z_i|s)P(s)P(c_j^{(k)}|z_i)P(z_i|s_q)}{P(z_i)\sum_i P(c_j^{(k)}|z_i)P(z_i|s_q)} \end{aligned} \quad (11)$$

which depends just on the parameters of the PLSA model and the a priori distribution of songs and semantic groups.

4. NMF OPTIMIZATION OF THE PLSA MODEL

In [9] the authors showed the relation between Non-negative Matrix Factorization (NMF) using Kullback-Leibler divergence and PLSA. In this section, we propose a multiplicative NMF update scheme for determining the unknown parameters of the combined PLSA model. Instead of minimizing the log-likelihood cost function (6), we will solve the following NMF optimization problem

$$\min_{\mathbf{W}^{(k)}, \mathbf{H}} \quad \sum_{k=1}^{K} \alpha_k ||\tilde{\mathbf{P}}^{(k)} - \mathbf{W}^{(k)}\mathbf{H}||_F^2 \quad (12)$$

$$\text{s.t.} \qquad \mathbf{W}^{(k)} \geq \mathbf{0}, \mathbf{H} \geq \mathbf{0} \quad (13)$$

where $||\mathbf{A}||_F^2$ denotes the squared Frobenius norm of a matrix, hence $\sum_{i,j} \mathbf{A}_{i,j}^2$, and $\mathbf{A} \geq 0$ means that all elements in $\mathbf{A}$ are non-negative.

By proper normalization of $\mathbf{W}^{(k)}$ and $\mathbf{H}$ we ensure the validity of the following interpretation

$$\left(\mathbf{W}^{(k)}\mathbf{H}\right)_{j,l} = \sum_{i=1}^{N} P(c_j^{(k)}|z_i) P(z_i|s_l), \quad (14)$$

from which, $\mathbf{W}^{(k)}_{j,i} = P(c^{(k)}_j|z_i)$ and $\mathbf{H}_{i,l} = P(z_i|s_l)$.

One way of minimizing (12) is to use a multiplicative update method, see e.g. [10]. Assuming the algorithm has converged to some point within the feasible region where $\mathbf{W}^{(k)} > \mathbf{0}$ and $\mathbf{H} > \mathbf{0}$, it can be shown that this point is a stationary point, which may or may not be a local minimum (see [10] for a more complete discussion about algorithms for solving NMF types of problems).

The following pseudo-code provides a multiplicative update scheme for solving the NMF problem given in 12. It can be easily seen that, if matrices $\mathbf{W}^{(k)}$ and $\mathbf{H}$ are initialized to strictly positive values, then these matrices remains positive throughout the iterations, as a consequence of multiplicative update scheme.

1. Initialize $\mathbf{W}^{(k)}$ and $\mathbf{H}$.
2. Iterate:
   (a)
   $$\mathbf{W}^{(k)}_{j,i} = \frac{(\tilde{\mathbf{P}}^{(k)}\mathbf{H}^T)_{j,i}}{(\mathbf{W}^{(k)}\mathbf{H}\mathbf{H}^T)_{j,i} + 10^{-9}}\mathbf{W}^{(k)}_{j,i} \quad (15)$$
   for $k = 1, \ldots, K$.
   (b) Normalize $\mathbf{W}^{(k)}$ such that $\sum_j \mathbf{W}^{(k)}_{j,i} = 1$ for $i = 1, \ldots, N$ and $k = 1, \ldots, K$
   (c)
   $$\mathbf{H}_{i,l} = \frac{\sum_k \alpha_k \left(\mathbf{W}^{(k)}\tilde{\mathbf{P}}^{(k)}\right)_{i,l}}{\sum_k \alpha_k \left(\mathbf{W}^{(k)^T}\mathbf{W}^{(k)}\mathbf{H}\right)_{i,l} + 10^{-9}}\mathbf{H}_{i,l} \quad (16)$$
3. Repeat 2 until some convergence criteria is met.

5. EXPERIMENTS

5.1. Dataset description

To illustrate the suitability of the PLSA model we have used a data set which was downloaded from the free section of the Amazon music service[1]. This data set has been previously used in combination with a genre plug-in for Winamp (see [11]). The original data set consists of 12631 music snippets, most of them of length ~ 30 secs. distributed unevenly among 227 genres and sub-genres. The original taxonomy provided by Amazon had problems with overlapping genres, e.g., how can one differentiate between International/Rock and just Rock?. Since we are going to use the genre information as a source of similarity between songs, and in order to minimize confusion among genres, we decided to keep only the songs belonging to unambiguous first level genres. Constraining also the minimum snippet length to 10 secs. resulted in a total number of 9823 music snippets distributed among the following 12 genres : "Rock" (2446), "Blues" (644), "Classical (Instrumental)" (361), "Country" (733), "Dance & DJ" (1002), "Folk" (872), "Jazz" (1261), "New Age" (596), "Opera & Vocal" (287), "Pop" (1005), "Rhythm & Blues" (287) and "Rap & Hip-Hop" (329).

[1]Downloaded in August, 2005.

5.2. Song similarity extraction

In this experimental section we consider two different kinds of similarities that are estimated from the raw audio data, and combined using the PLSA model. Once a set of "sound features" are extracted, a first similarity measure makes use of the available genre information, while the second one is just based on the similarity among the extracted sound features.

5.2.1. "Sound feature" extraction

Here we make use of aggregated features as described in [12]. We extracted the first seven Mel Frequency Cepstral Coefficients (MFCC) on a 20msec. time-scale with 10msec. overlap. The first coefficient is discarded (to remove the influence of different recording volumes). Then, MFCCs are collected using a window size of one second, thus creating a six dimensional time series of 100 samples. For each such block, the time series is modelled using a multivariate autoregressive model (MAR) of lag three : $\mathbf{x}_n = \sum_{p=1}^{3} \mathbf{A}_p \mathbf{x}_{n-p} + \mathbf{e}_p$, where $\mathbf{x}_n$ is used to denote a vector of MFCC features inside the window. The values of the three matrices $\mathbf{A}_p$, together with the mean and covariance of the residuals, $\mathbf{e}_n$, are concenated into a single feature vector (MAR feature) of length 135, representing one second of the music snippet.

5.2.2. Learning similarity from "sound features"

In web documents we have direct access to $\tilde{P}(c^{(k)}_j, s)$, which are given either by the term frequency or by the hyperlink frequency. In other words, in document analysis we have direct access to the similarity information, while in music we mostly have access to sound features. From sound features, however, we can estimate different types of similarity information, possibly using some of the metadata associated with the songs. In this paper, we have used two different approaches, the first of them being a supervised method, and the second following an unsupervised approach:

Similarity $c^{(1)}$ (supervised): Having some kind of labels for the songs in our dataset, for instance, genre labels, we could straightforwardly use these as $\tilde{P}(c^{(k)}_j|s_l)$. In most databases music snippets typically belong to a single genre only, hence, $\tilde{P}(c^{(k)}_j|s_l)$ can be either 0 or 1. A more powerful approach is to train a classifier to predict the a posteriori probability of each of the classes, so that we acquire information about soft-membership to classes.

As a first similarity measure we have used the outputs of a neural network trained using the MAR features as inputs, and the genre-information as the desired labels. The neural network consists of a non-linear feature extraction phase, using the rKOPLS algorithm presented in [13], followed by a linear classifier. Though each MAR feature was assigned to just one genre, soft membership of the music snippets to the different genres was determined using late fusion. Hence, simply summing the outputs of all MAR features in one snippet and normalizing to get $\sum_j \tilde{P}(c^{(1)}_j|s) = 1$.

Similarity $c^{(2)}$ (unsupervised): We can straightforwardly derive valid similarity sources of information by carrying out a hard or soft clustering in the "sound features" space. In this paper, to obtain a second similarity dimension, MAR features were clustered using a K-means algorithm with cosine distance measure. The cosine distance was chosen to

provide improved robustness in high dimensions. Five fold cross-validation was used determine a reasonable number of clusters (70) for the dataset. Though the clustering uses hard assignment, soft membership was obtained using the same strategy that was explained for the supervised case.

We should mention here that other approaches for learning music similarity can also be used. An example could be in terms of co-occurrence in playlists, i.e., how often music piece A is played after music piece B and vice-versa.

5.3. Results and discussion

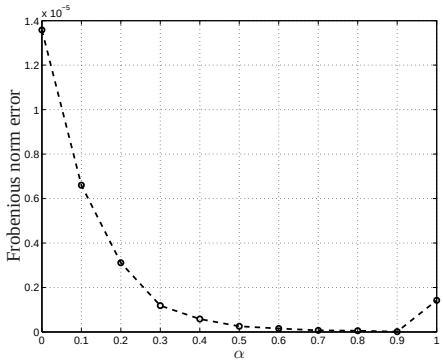
The modified NMF algorithm suggested in Section 4 was run with a varying dimension of the "latent semantics" ranging from 3 to 48. Considering also different values α between 0 (only similarity $c^{(2)}$ was used) and 1 (using only $c^{(1)}$ similarity measure), a dimension of 42 latent semantics was found from a reasonable compromise in the Frobenious norm error (i.e., a smaller number resulted in larger error, while a larger number of dimensions only provided very slight reductions of the error). A dimensionality of 42 is more or less in-between the 12 dimensions used in similarity $c^{(1)}$ and the 70 dimensions used for the similarity measure $c^{(2)}$. In any case, note that this number is much smaller than the number of possible combinations using one cluster from each similarity criterion, and thus the PLSA approach provides a much more compact and convenient representation for song recommendation than the direct use of (9). In each run of the NMF algorithm, the algorithm was stopped after 1000 iterations, which seemed reasonable when considering the error as a function of the iterations.

Figure 1(a) shows the Frobenius norm error calculated between the empirical distribution $\tilde{P}(c_j^{(1)}|s_l)$ and the model ($\mathbf{W}^{(1)}$ and $\mathbf{H}$) as a function of varying α. Figure (b) shows the corresponding log-likelihood, just to illustrate the good correspondence between both errors, and how the likelihood increases with decreasing Frobenious norm. With $\alpha = 0$, the latent space is estimated purely from the $c^{(2)}$ similarity measure, which explains the high approximation error to the real distribution. Conversely, when $\alpha = 1$ (this corresponds to considering only the $c^{(1)}$ similarity measure) a much better solution, with respect to similarity $c^{(1)}$, is obtained. It is interesting to notice (see also [8]) that the error is smaller the range $\alpha = 0.3$ to $\alpha = 0.9$ than for $\alpha = 1$. In other words, incorporating some information about the $c^{(2)}$ similarity, serves to improve the capabilities of the PLSA model to represent similarity in dimension $c^{(1)}$. Actually, the latent semantics model with $\alpha = 0.9$ has been found to increase the genre classification rate provided by the rKOPLS-based genre classifier[2].

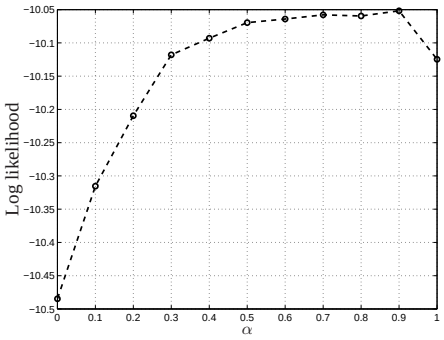
Figure 2(a) shows the Frobenious norm error calculated between the $c^{(2)}$ similarity measure and the corresponding model ($\mathbf{W}^{(2)}$ and $\mathbf{H}$). Looking at both Figs. 1 and 2, one can conclude that using only one of the similarities alone always results in a very small likelihood of the observations associated to the other similarity. However, there exists good compromise values of $\alpha \in [0.3, 0.8]$, given that all these values would provide improved performance on the first similarity dimension, while keeping a reasonably good representation for the second similarity.

Additionally, as we have already mentioned, the PLSA approach provides a much more compact representation space than the combined used of the clusters in similarity $c^{(1)}$ and $c^{(2)}$ and, therefore, a more convenient space for music recommendation.

[2]Classification rates for some of the most unlikely classes increased significantly, e.g., from 0% to 14% and 40% for the "Country" and "New age" genres, respectively



(a) Frobenious norm error on similarity $c^{(1)}$



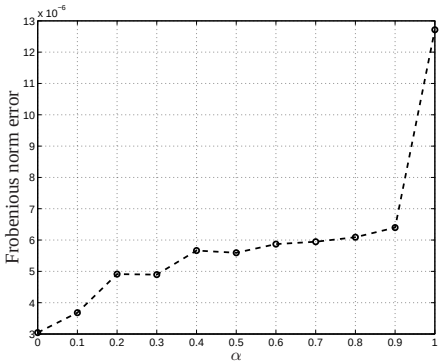
(b) Log-likelihood error on similarity $c^{(1)}$

Fig. 1. Figure (a) shows the Frobenius norm error between the empirical distribution ($\tilde{P}(c_j^{(1)}|s_l), \forall j, l$) and the model given by $\mathbf{W}^{(1)}$ and $\mathbf{H}$ as a function of α. Figure (b) shows the corresponding log-likelihood.

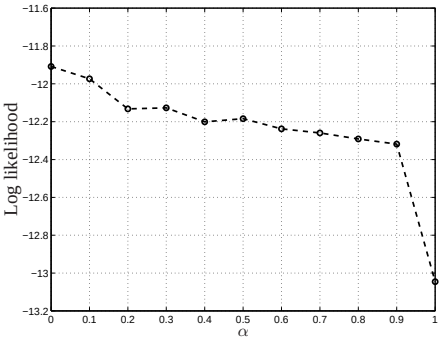
6. CONCLUSIONS

In this paper we have presented the extension of the PLSA framework for its application in music recommendation systems. Basically, the proposed PLSA model works by projecting the songs into a latent semantic space. This space is obtained by maximizing a combined log-likelihood which takes into account different sources of similarity between songs. By doing so, the latent semantics can satisfactorily explain all observed similarities and provide a very convenient representation for song retrieval, while keeping complexity under control. Preliminary experimental tests carried out combining two measures of similarly provided a better representation than when using just one of the similarity dimensions alone.

We think that the analogy between documents and songs is a fruitful one, and opens new lines for investigating music structure using the elaborated machinery already deployed for web-mining, and for improving the performance of music recommendation systems.



(a) Frobenious norm error on similarity $c^{(2)}$



(b) Log-likelihood error on similarity $c^{(2)}$

Fig. 2. Figure (a) shows the Frobenius norm error between the empirical distribution $(\tilde{P}(c_j^{(2)}|s_l), \forall j, l)$ and the model given by $\mathbf{W}^{(2)}$ and $\mathbf{H}$ as a function of α. Figure (b) shows the corresponding log-likelihood.

Acknowledgments

This work has been partly by Spanish Ministry of Education and Science grant CICYT TEC-2005-00992, by Madrid Community grant S-505/TIC/0223 and by the Danish Technical Research Council, through the framework project 'Intelligent Sound', `www.intelligentsound.org` (STVF No. 26-04-0092).

7. REFERENCES

[1] D. Ellis, B. Whitman, A. Berenzweigh, and S. Lawrence, "The quest for ground truth in musical artists similarity," in *Proc. of the Intl. Symp. on Music Information Retrieval*, 2002.

[2] A. Berenzweig, B. Logan, D. Ellis, and B. Whitman, "A large scale evaluation of acoustic and subjective music similarity measures," in *Proc. of the Intl. Symp. on Music Information Retrieval*, 2003.

[3] E. Wold, T. Blum, D. Keislar, and J. Wheaton, "Content-based classification, search, and retrieval of audio," *IEEE Multimedia*, vol. 3, pp. 27–36, 1996.

[4] J. Foote, "Content-based retrieval of music and audio," in *Multimedia Storage and Archiving Systems II, Proc. of SPIE*, vol. 3229, pp. 138–147, 1997.

[5] J.-J. Aucouturier and F. Pachet, "Music similarity measures: What's the use?," in *Proc. of the Intl. Symp. on Music Information Retrieval*, 2002.

[6] B. Logan and A. Solomon, "A music similarity function based on signal analysis," in IEEE Intl. Conf. on Multimedia & Expo, 2001.

[7] T. Hofmann, "Probabilistic Latent Semantic Analysis," in *Proc. 15th Conf. on Uncertainty in Artificial Intelligence*, pp. 289–296, 1999.

[8] D. Cohn and T. Hofmann, "The Missing Link – A Probabilistic Model of Document Content and Hypertext Connectivity," in *Neural Information Processing Systems 13*, 2001.

[9] E. Gaussier and C. Goutte, "Relation between PLSA and NMF and implications," in *SIGIR*, pp. 601–602, 2005.

[10] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca and R. J. Plemmons "Algorithms and Applications for Approximate Nonnegative Matrix Factorization," in *Computational Statistics and Data Analysis. Elsevier. To appear* , 2007.

[11] T. Lehn-Schiøler, J. Arenas-García, K. B. Petersen and L. K. Hansen "A Genre Classification Plug-in for Data Collection," in *International Symposium on Music Information Retrieval (ISMIR)*, 2006.

[12] A. Meng and P. Ahrendt and J. Larsen and L. K. Hansen "Temporal Feature Integration for Music Genre Classification," in *IEEE Transactions on Audio, Speech and Language Processing. To appear*, 2007.

[13] J. Arenas-García, K. B. Petersen and L. K. Hansen, "Sparse Kernel Orthonormalized PLS for feature extraction in large data sets,", in *Advances in Neural Information Processing Systems 19, MIT Press, Cambridge, MA*, 2007