Combining Sources of Description for Approximating Music Similarity Ratings

Daniel Wolff and Tillman Weyde

City University London, Department of Computing
Northampton Square, London EC1V 0HB, UK
daniel.wolff.1@soi.city.ac.uk

Abstract. In this paper, we compare the effectiveness of basic acoustic features and genre annotations when adapting a music similarity model to user ratings. We use the Metric Learning to Rank algorithm to learn a Mahalanobis metric from comparative similarity ratings in in the MagnaTagATune database. Using common formats for feature data, our approach can easily be transferred to other existing databases. Our results show a notable correlation between songs' genres and associated similarity ratings, but learning on a combined feature set clearly outperforms either individual approach.

Keywords: Music Information Retrieval, Music Recommendation, Computational Modelling, Music Similarity, Music Perception

1 Introduction

Adapting music recommendation systems to the needs or preferences of users is a critical factor in the success of commercial music sites today. For businesses as well as for users, presenting relevant results to the latter promises to increase the - be it aesthetical, social or financial - revenue. Depending on the context and intention, different ways of determining relevance in music may be appropriate to fulfil the expectations of the above parties.

Our focus lies on the generating models of perceived or stated music similarity for acoustic recordings of music, which can be applied in music exploration or recommendations systems. To this end, we exploit user ratings from a human computation source, which yield relative similarity ratings about triples of songs. The raw data is approximated using binary rankings expressing "Songs {A, ...} are more similar to Song B than the Songs {C, ...}". Such rankings are used for constraining the optimisation of generalised metrics, defined on the vector space of features describing the music. There are several algorithms available for this task. In the present paper, we choose the MLR algorithm for its robust behaviour, and focus on the effects of using different sources of information, namely content-based features and genre annotations as well as different representations of these for training the metrics.

1.1 Related Work

The selection of features suitable for a specific task has been a field of active research, relevant to many disciplines within Music Information Retrieval. Properties of features and their selection routines, besides customisation to users, allow for a definition and selection of relevant information, the structuring of datasets and thus for specialised indexing methods to be used.

In 2001, Pickens categorised selection techniques for music information retrieval on symbolic data, focussing in the relation of musicological properties of the extraction routines and their implications for retrieval performance [1]. Novello et al. performed a study on music similarity perception [2]. Asking subjects to select pairs of best- and worst fitting songs out of triplets presented, they rendered a musical similarity space using multidimensional scaling. Their evaluation of the data gives important insights in the concordance within similarity ratings of musicians and non-musicians, and the correlation of these with musical genres.

In the evaluation and optimization chapter of his dissertation [3] Pampalk extensively evaluates the performance of 14 content-based features in a genre-classification task: The correlation of songs' genres and clusters inferred from a similarity defined by weighted feature influences are compared, using leave-one-out cross-validation. The tests were performed on several databases with sizes of 100 to 15335 western pop and classical music tracks. Moreover, combinations of the six best-performing features were evaluated using a combinatorial approach. Results showed that spectral features have a strong weight in best performing configurations, alongside with percussivity and fluctuation pattern features.

A set-based method for learning a feature weighting using an interactive playlist-based user survey has been presented by Allan et al. [4]. Users could specify their preferred similarity concepts using two example song sets, one for similar and one for dissimilar songs. Moreover, recommendations were improved using a feedback loop.

Barrington et al. [5] used timbral and harmonic features, as well as tags and information mined from the web for text-based audio retrieval. Different ways of combining these information sources – calibrated score averaging, RankBoost, and kernel combination support vector machines – were evaluated. As shown in [6], the kernel combination approach enabled a straightforward analysis of the different features' influences. McFee et al. [7] have designed an algorithm for learning a Mahalanobis metric to rankings, based on the Cutting-Plane Structural SVM training algorithm of [8]. They used it in a hybrid approach for parametrising a purely content-based music similarity metric using collaborative filtering data. Their content-based classifiers were successfully applied for music discovery in the so-called long tail, i.e. sets of sparsely annotated and barely documented music, e.g. new or less popular songs, where this method

enables improved recommendation.

Other algorithms for learning Mahalanobis metrics from comparative user ratings have been published: Schultz and Joachims [9] trained a weighted Euclidean distance metric using relative comparisons. In [10], Davis et al. formulate a metric learning problem similar to the above, as an LogDet-optimisation task. Their approach uses another arbitrarily predefined Mahalanobis metric for the regularisation target.

2 Music and Descriptions

The majority of the data used for the experiments in Section 4 is based on the MagnaTagATune dataset. TagATune is a web-based[1] game, collecting tags associated with certain songs in a human-computation manner. Furthermore, in a bonus mode of the game, user votes on perceptual outliers out of song triplets are collected: Users have to agree on a song out of three which is the least similar to the remaining songs. Documenting the application of this game on a song database from the Magnatune label, MagnaTagATune combines the audio content, derived features and tagging information of 25863 30-second audio clips into a publicly available dataset [11]. The data from the bonus mode contains 7650 individual votes on 533 triplets of clips. The clips C_i, $i \in \{1, \cdots 1019\}$ included in these triplets constitute the dataset used in our experiments.

2.1 Genre Annotations

We extend the information in this dataset by extracting the genre tags the Magnatune label assigned to the clips' corresponding albums for indexing and marketing purposes. This information is publicly available via their xml catalogue[2]. The catalogue contains ordered genre descriptions which exhibit a hierarchical character, which is ignored in this application. Each clip in our experiment subset is tagged with around 2-4 genre descriptions. Thereby, a vocabulary of 44 genre tags is established. The genre information for an individual clip C_i is now expressed using binary feature vectors $F_i^{genre} \in \{0,1\}^{44}$, each component corresponding to whether the clip is annotated with a particular genre description.

2.2 Content-based Features

The content-based features contained in the MagnaTagATune dataset have been created using the "The Echo Nest" API 1.0. The algorithms used in the API have been describend im [12]. We use the segment-based chroma and timbre information for each clip to generate a single feature vector describing the entire

[1] http://www.gwap.com/gwap/gamesPreview/tagatune/
[2] http://magnatune.com/info/api.html

clip. The features used here are intended to represent the rough harmonic and timbral content of each clip. This is achieved by separately clustering the chroma and timbre vectors into four clusters $t_i^j \in \mathbb{R}^{12}$, $c_i^j \in \mathbb{R}^{12}_{\geq 0}, j \in \{1, \cdots 4\}$ for each clip C_i, $i \in \{1, \cdots 1019\}$. As the temporal segments related to each of the chroma and timbre vectors are of different length, we use a weighted k-means variant, including the single feature vectors' corresponding segment lengths for determining the cluster centroids. Vectors only accounting for a short frame of time thus have less impact in determining one of the cluster centroids. For each cluster, we save the accumulated weight of the corresponding vectors in the scalars $\lambda(c_i^j), \lambda(t_i^j) \in [0, 1]$. The chroma centroids are then normalised using

$$\tilde{c}_i^j = \frac{c_i^j}{\max_k(c_i^j(k))}. \tag{1}$$

As the components of the timbre centroids feature strong outliers, these were clipped to the percentile $p_t^{0.85}$ corresponding to the interval $\left[0, p_t^{0.85}\right]$ including 85% of the absolute component values $|t_i^j|$ of all clusters j and clips C_i. Afterwards, the clipped values were shifted and scaled to fit the interval $[0, 1] \ni \tilde{t}_i^j$.

Finally, the above values are combined into feature vectors

$$F_i^{audio} = \left(\tilde{c}_i^1, \cdots \tilde{c}_i^4, \lambda(c_i^1) \cdots \lambda(c_i^4), \tilde{t}_i^1, \cdots \tilde{t}_i^4, \lambda(t_i^1) \cdots \lambda(t_i^4)\right)^T \in \mathbb{R}^{104}. \tag{2}$$

2.3 Combined Features

In order to combine the information from genre and audio features, both feature vectors are concatenated into the combined feature vector

$$F_i^{comb} = \left(F_i^{audio}(1), \cdots F_i^{audio}(104), F_i^{genre}(1), \cdots F_i^{genre}(44)\right)^T \in \mathbb{R}^{148}. \tag{3}$$

2.4 PCA, Reduced Features

For each of the single and combined features, a Principal Component Analysis (PCA) is performed. After sorting according to variance in the principal components, we reduce the dimensionality of the transformed features, keeping only the 20 components with greatest variance across the 1019 clip dataset. In this way we gain a set of three different features sharing a constant dimensionality.

After transformation into the principal component space, across the whole dataset, the individual feature components are shifted and scaled to fit the interval of $[0, 1]$. The impact of such normalising of the features was tested. we found that normalising the features after transformation generally improved the results of the following metric learning. In our experiments below, we call these PCA-reduced and normalised features $\tilde{F}_i^{genre}, \tilde{F}_i^{audio}, \tilde{F}_i^{comb} \in \mathbb{R}^{20}$.

2.5 Binary Rankings

As mentioned above the dataset contains a set of vote statistics for the outlying clip given three clips. We gather the binary rankings, used as ground truth in our experiments, by approximating these voting statistics: For each such triplet of songs, where possible, a "winning" outlier is determined, by following the majority of the votes. Triplets not featuring unequivocal voting results are dismissed.

From the remaining triplets, we generate 533 binary rankings. Each ranking is defined by two sets r_i^s and r_i^d, where r_i^s contains relatively more and r_i^d less similar clips to a given clip C_i. These rankings very roughly approximate the above user votings, and, when compared to other representations of the MagnaTagATune comparison data, show a greater applicability to the metric learning algorithm explained below. Only 12 sets r_i^d and r_i^a contain more than one clip. Thus, most of the rankings can be read as information about clips C_i, C_j, C_k, with $C_j, r_i^s = \{j\}$ being more similar to the query clip C_i than $C_k, r_i^d = \{k\}$.

Note that the comparison of binary rankings used in this method, e.g. in the training of the metric or evaluation of a metric's performance, is only based on the relative ranking positions of clips: The correctness of ranking is defined by evaluating the relative positions of results marked as more or less similar; a correct ranking positions the more similar clips before the less similar ones.

3 Metric Learning To Rank

McFee et al. developed an algorithm for learning a Mahalanobis distance from binary rankings [7]. The Mahalanobis metrics described here resemble a weighted Euclidean metric, but they also allow for a weighting according to rotation and translation of the vectors. Given two vectors $x, y \in \mathbb{R}^N$, the family of Mahalanobis metrics can be expressed as

$$d_W(x, y) = \sqrt{(x - y)^T W (x - y)}, \tag{4}$$

where $W \in \mathbb{R}^{N \times N}$ is a positive semidefinite matrix, parametrising the distance function. Technically, these distance functions also include pseudometrics, which allow for a zero distance between two non-identical vectors.

The distance function is optimised using an algorithm based on Structural SVM [13]. Using a constrained regularisation approach, the matrix W is determined by comparing possible correct and incorrect rankings and their corresponding parametrisation to W. A feature map ψ, combining feature data and rankings, is given by the matrix valued partial order feature, described in [14]. Used in the sense as below, it emphasizes directions in feature space which are correlated with correct rankings. Given a set of training query feature vectors $q \in X \subset \mathbb{R}^N$ and the associated training rankings y_q^*, the complete quadratic optimisation problem is given by

$$\begin{aligned}
\min_{W,\xi} \quad & tr(W^T W) + c\frac{1}{n}\sum_{q\in X}\xi_q, \\
\text{s.t.} \quad & \forall q \in X,\ \forall y \in Y \setminus \{y_q^*\} : \\
& \left\langle W, \psi(q, y_q^*)\right\rangle_F \geq \langle W, \psi(q,y)\rangle_F + \Delta(y_q^*, y) - \xi_q, \\
& W_{i,j} \geq 0, \xi_q \geq 0.
\end{aligned} \tag{5}$$

The ξ_q allow for some of the training constraints to be violated. Here, c determines the balance between the regularisation and ranking loss term. $\langle *, * \rangle_F$ denotes the Frobenius matrix product. The ranking-loss term $\Delta(y_q^*, y)$ assures the margin between the given training rankings y_q^* and incorrect rankings y. Common evaluation measures for information retrieval systems are used to determine the respective minimal margin sizes. We selected the AUC-related methods for our experiments, being more robust than nearest neighbour approaches considering the rankings carry very sparse information: The AUC curve compares the relation of true positives and false positives in the ranking calculated using the current training state of the metric. Most of the training rankings y_q^* feature just two defined clip positions which either are in correct or incorrect order. As the complete set of possible rankings Y to consider for each training ranking is too large, a cutting-plane approach (see [8]) is used to predict the most violated constraints. The MLR framework is available online[3].

4 Experiments

We applied the MLR algorithm as described in Section 3 for training a distance measure using the rankings from Section 2.5. The experiment described below is part of a series of general experiments on the feasibility of metric learning from user comparisons. Varying the feature types used for the songs' descriptions, the abilitiy of the learned metrics to reproduce the given rankings were compared. In the following experiments, we used the same constraint – regularisation tradeoff factor $c = 10000$ (see Section 3), which was determined to work well in previous experiments using the same ground truth with similar features.

For our experiments we use fivefold cross-validation. The following figures plot the mean performance over the five different partitions for training and test sets. For assessing training performance, we measure the percentage of rankings in the test sets to be correctly reproduced by a trained metric. Rankings are fulfilled, if all clips in r_i^s are ranked before any clip in r_i^d.

4.1 Content vs. Annotation

Figure 1 shows the metric learning success curves regarding three different feature types: content-based features only, genre features only and the combination

[3] http://cseweb.ucsd.edu/~bmcfee/code/mlr/

of these features. With a maximal accuracy of 81.8%, the combined approach has a performance strongly exceeding that of the isolated features. The strength of this effect is probably related to the capability of Mahalanobis metrics to model correlations between features of the different types.

Looking at the individual features, the final results of the content-data (73.37%) come very close to the slightly better performing genre features (73.74%). But when considering the smaller training sets, the genre features seem much more effective. The 68.66% baseline using an unweighted Euclidean distance for these features shows that the genre feature space has greater correlation to the users ratings than the content-based features, only allowing for a 61.7% baseline. Relative to the baselines, the performance gain using the MLR training is much greater on the content-based features.

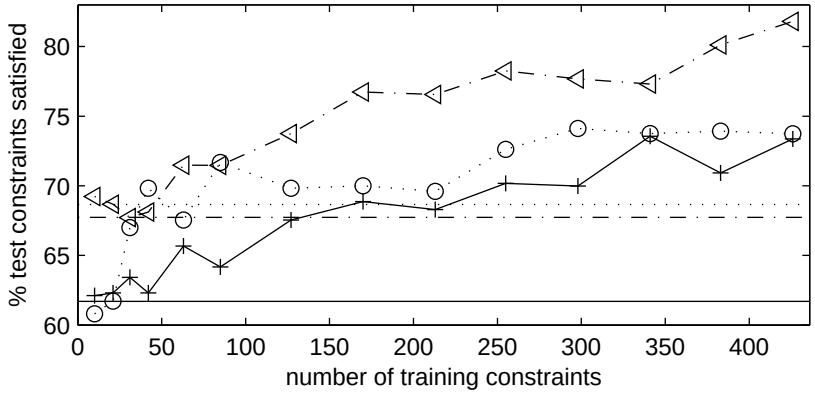

Fig. 1. Results for increasing training set size. Plotted are the mean percentages of fulfilled rankings in the test sets. Top to bottom: Combined features (line-dotted, ◁), genre features (dotted, ○), and content-based features (continuous line, +). The performances of the Euclidean metrics are represented by the straight lines at the bottom, line shapes represent feature types as above.

PCA experiments The above experiments may very well be influenced by the information density and especially dimensionality of the described feature representations. Therefore we conducted a second experiment, this time using the fixed-dimensional approximations based on Principal Component Analysis (PCA) of the above features. The following experiments were performed with $c = 1000$. Our early experiments underlined that this factor depends on the feature dimension.

When applying the same experiment as above on the PCA features, the resulting learning curves as pictured in Figure 2 closely resemble the situation

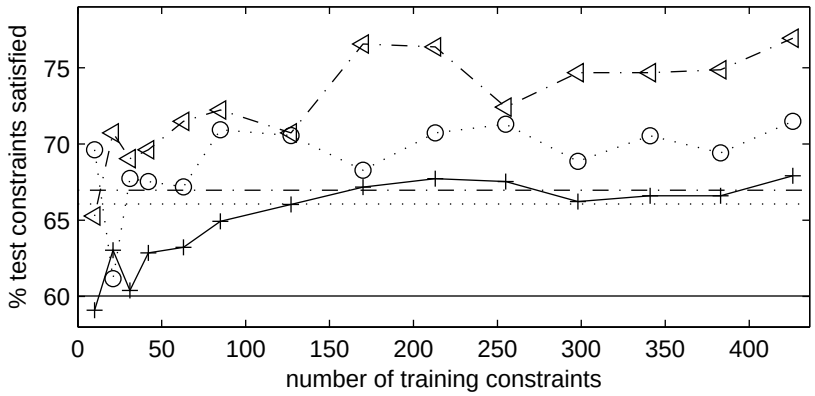

Fig. 2. Results for the PCA reduced features, increasing the training set size. Plotted are the mean percentages of fulfilled rankings. Combined features (line-dotted, ◁), genre features (dotted, ○), and content-based features (continuous line, +). The performances of the Euclidean metric are represented by the straight lines at the bottom.

without PCA (see Figure 1). The baseline Euclidean metric results for both single-medium feature types have dropped less than 2%, but the performances of the trained metrics drop by 5.4% (to 67.92%) for content-based and 2.3% (to 71.48%) for the genre features, showing a significantly lower performance after training.

Here, the metric based on the combined features achieves a performance of 76.9%, indicating an informational gain by combining the two feature types, instead of just adding dimensions to parametrise. Moreover, the combined features' baseline now exceeds the performance of both of the baselines related to the single features.

Generalisation When considering the performance on the training data, for the raw features, the metric based on genre features performs worse than the one based on content-based features. The content-based features seem less enabling the learning of a general perceptual trend, specifically fitting to the training rankings.

The results using PCA-reduced features show a strong decrease in the adaptation ability of the metrics. The genre- and content-based features now are almost even in performance on the training data, but in analogy to the above case, the metrics based on genre features perform better on the test sets. This also underlines a stronger correlation of the users ratings and our genre features.

	F^{audio}	F^{genre}	F^{comb}
Raw	100 %	92.32%	100%
PCA	85.60%	85.41%	91.28%

Table 1. Training success for different feature types. Noted are percentages of **training set rankings** correctly reproduced after training with full-size training sets. Statistics are shown for both raw and PCA-reduced feature versions.

5 Discussion

We have used metric learning for predicting users' music similarity ratings, comparing the influence of different types of descriptions of the music clips. In line with findings in [3] and others, our experiments show that the combination of content based and annotated features, where available, does improve the adaptability of resulting feature spaces. Here, the performance gains achievable with single feature types do not simply add up linearly when these features are combined. Generally we observe a strong influence of less data-specific parameters, e.g. dimensionality and number of training examples, on the optimisation process. Further research has to be done towards tuning the procedures for learning metrics to specific properties of the features at hand, like sparseness and dimensionality.

The features used in this study are basic in nature, and for the case of the genre features also very sparse. More elaborate feature extraction methods may very well improve the performance of the content-based features in particular. Moreover, the representation of the annotations does not accurately reflect the intention of general tag annotations: Usually, the positive information about assigned tags is more important than the information of missing ones. The applied metrics, as linear functions, can not reflect such a bias when using the proposed feature representation.

Further experiments are also planned with regards to the representation of the MagnaTagATune triplet comparison data via binary rankings, in order to better represent the individual user votes in the actual training data used for the algorithms.

References

1. Pickens, J. (2001) A survey of feature selection techniques for music information retrieval. *Proceedings of the 2nd International Symposium on Music Information Retrieval (ISMIR)*.
2. Novello, A., Mckinney, M. F., and Kohlrausch, A. (2006) Perceptual evaluation of music similarity. *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR)*.

3. Pampalk, E. (2006) *Computational Models of Music Similarity and their Application in Music Information Retrieval*. Ph.D. thesis, Vienna University of Technology, Vienna, Austria.
4. Allan, H., Müllensiefen, D., and Wiggins, G. (2007) Methodological considerations in studies of musical similarity. *8th International Conference on Music Information Retrieval*, pp. 473–478.
5. Turnbull, D. R., Barrington, L., Lanckriet, G., and Yazdani, M. (2009) Combining audio content and social context for semantic music discovery. *SIGIR '09: Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, New York, NY, USA, pp. 387–394, ACM.
6. Barrington, L., Yazdani, M., Turnbull, D., and Lanckriet, G. (2008) Combining feature kernels for semantic music retrieval. *Proceedings of the 9th International Conference on Music Information Retrieval (ISMIR)*, pp. 614–619.
7. Mcfee, B. and Lanckriet, G. (2010) Metric learning to rank. *Proceedings of the 27th annual International Conference on Machine Learning (ICML)*.
8. Joachims, T., Finley, T., and Yu, C.-N. J. (2009) Cutting-plane training of structural svms. *Machine Learning*, **77**, 27–59.
9. Schultz, M. and Joachims, T. (2003) Learning a distance metric from relative comparisons. *Advances in Neural Information Processing Systems (NIPS)*, MIT Press.
10. Davis, J. V., Kulis, B., Jain, P., Sra, S., and Dhillon, I. S. (2007) Information-theoretic metric learning. *Proceedings of the 24th international conference on Machine learning*, New York, NY, USA, pp. 209–216, ICML '07, ACM.
11. Law, E., West, K., Mandel, M., Bay, M., and Downie, J. S. (2009) Evaluation of algorithms using games: the case of music annotation. *Proceedings of the 10th International Conference on Music Information Retrieval (ISMIR)*, October, pp. 387–392.
12. Jehan, T. (2005) *Creating Music by Listening*. Ph.D. thesis, Massachusetts Institute of Technology, MA, USA.
13. Tsochantaridis, I., Joachims, T., Hofmann, T., and Altun, Y. (2005) Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research (JMLR)*, **6**, 1453–1484.
14. McFee, L., B.and Barrington and Lanckriet, G. (2010) Learning similarity from collaborative filters. *Proceedings of the International Society of Music Information Retrieval Conference*, pp. 345–350.