Adapting Similarity on the MagnaTagATune Database: Effects of Model and Feature Choices

Daniel Wolff
MIRG, School of Informatics
City University London
London EC1V 0HB
daniel.wolff.1@soi.city.ac.uk

Tillman Weyde
MIRG, School of Informatics
City University London
London EC1V 0HB
t.e.weyde@city.ac.uk

ABSTRACT

Predicting user's tastes on music has become crucial for a competitive music recommendation systems, and perceived similarity plays an influential role in this. MIR currently turns towards making recommendation systems adaptive to user preferences and context. Here, we consider the particular task of adapting music similarity measures to user voting data. This work builds on and responds to previous publications based on the MagnaTagATune dataset. We have reproduced the similarity dataset presented by Stober and Nürnberger at AMR 2011 to enable a comparison of approaches. On this dataset, we compare their two-level approach, defining similarity measures on individual facets and combining them in a linear model, to the Metric Learning to Rank (MLR) algorithm. MLR adapts a similarity measure that operates directly on low-level features to the user data. We compare the different algorithms, features and parameter spaces with regards to minimising constraint violations. Furthermore, the effectiveness of the MLR algorithm in generalising to unknown data is evaluated on this dataset. We also explore the effects of feature choice. Here, we find that the binary genre data shows little correlation with the similarity data, but combined with audio features it clearly improves generalisation.

Categories and Subject Descriptors

H.5.5 [**Information Interfaces and Representation**]: Sound and Music Computing—*Modeling*; I.5.3 [**Pattern Recognition**]: Clustering—*Similarity Measures*

Keywords

Similarity Adaptation; Music Information Retrieval; MagnaTagATune

1. INTRODUCTION

Today we are able to digitally store and almost instantly retrieve records of music with very little physical effort of the user. Now, a major problem problem which arises the retrieval, recommendation, or discovering new music. Large music databases, both online and individually owned, demand elaborate techniques to automatically compare, classify and index music.

The appropriateness of specific algorithms, particularly the features used for representing and comparing music, depends not only on the acoustic content of music, but also on the user's perception and the context of the user and the music. Thus, increasing efforts are being made to replace established rigid music classification and recommendation models with approaches that can be adapted to users and contexts.

In line with this development, numerous user data sets on music retrieval behaviour are being made available. To acquire such data has been very costly in the past. The data ranges from media playlists, over search result click-troughs to the kind of data which more recently has been collected using GWAPs (games with a purpose). Such data can be used to further improve recommendations in online advertising, search machines or help training other machine learning models. For computational musicologists, personal ratings or usage data concerning music allow for the development of new, automatically adapted models of music perception or the analysis of cultural characteristics in the use of music.

We are interested in models reproducing users' similarity votes for music clips, which can be used in any of the above applications. To this end we trained similarity measures to the feature and similarity data present in the MagnaTagATune database, testing three different feature sets. In this paper, we compare our approach **WOLF11**([7]) to the one presented by Stober and Nürnberger at AMR 2011, **STOB11** ([6]), using the same dataset.

This paper juxtaposes the algorithms on each of their stages: The next section summarises the basic assumptions and published results for the approaches to be compared. In Section 2, we compare the features used by WOLF11 and STOB11. Although both use the features provided in the MagnaTagATune dataset, there are differences in processing and selection. This section also covers the processing of the similarity votings in MagnaTagATune, which we adopted from STOB11 for comparability. Section 3 introduces and discusses the differences in both modelling approaches, which are central in this comparison. Section 4 compares the approaches' success in reproducing user ratings. In three separate experiments we compare general results in WOLF11 and STOB11, different feature sets in WOLF11, and generalisation capabilities of both approaches. This leads us to some general conclusions and directions for future research noted in Section 5.

WWW 2012 Companion,, April 16–20, 2012, Lyon, France.
ACM 978-1-4503-1230-1/12/04.

1.1 Previous Work

For a literature review of learning similarity measures in general, we refer to [6] and [8]. The similarity part of the MagnaTagATune dataset that we work with has only recently been analysed. It was created in the following way: In the regular TagATune music game (with a purpose), the similarity data we work with comes from the bonus mode, which encourages the team to collaborate in voting on the "odd one out" of a triplet of songs. This yields relative information on the rated similarity of the presented clips, which has been collected in the MagnaTagATune dataset by Law et al.[2] together with the 29 seconds audio clips themselves, content-based features and other information.

In WOLF11, we used the dataset for adapting a music similarity metric to the included similarity constraints. Via the *Metric Learning to Rank* algorithm by McFee et al.[4], we compared the effectiveness of content-based and metadata-based features on adapting a Mahalanobis distance function to the similarity votings. Our results show that a good part of the information contained in the similarity votings can be learned. The adaptation also improves the model performance on unknown test sets, showing an ability to generalise from few training data. We furthermore showed that, although genre- and content-based features have been adapted to the same level of constraint satisfaction, genre features show a greater ability to generalise over unknown data.

At the same time, in STOB11, Stober et al. compared different methods for linear- and quadratic optimisation of a weighting of similarity measures. Their experiments, based on the same dataset, test the ability of the algorithms to fulfil all of the user similarity constraints, using increasing portions of the dataset for training. Moreover they compare the time efficiency in learning. Interested in a user-adaptable parameter space, only weightings of predefined feature similarity measures are adapted. They analyse eight different learning variants on two different subsets of the similarity constraints, the smaller of which is specially designed to be solvable by all of the approaches.

Whilst both approaches address the problem of inconsistencies in the MagnaTagATune similarity dataset, the data are filtered differently, rendering the results of STOB11 and WOLF11 difficult to relate. Thus, in Section 4 of this paper, we apply our features and learning paradigm presented in WOLF11 to the similarity data and within the evaluation framework described in STOB11.

2. PREPROCESSING MAGNATAGATUNE

The similarity dataset spans over a subset of 1019 clips in the dataset. In order to define similarity measures on the clips, features are extracted for representing and comparing them. The MagnaTagATune dataset comes with a set of audio features extracted using the Echo Nest API. Moreover, clips are annotated with 188 unique tags from the standard mode of the TagATune Game. Finally, genre tag annotation for the clips has been made available from the catalogue of the Magnatune label.

2.1 Audio Feature Data

The Echo Nest features contain a large set of content-based features, spanning from low to upper-mid level complexity. In WOLF11, our features were deliberately designed to contain only straightforwardly extractable data. Thus only the **chroma** and **timbre** features are used. As both of these features are delivered on a segment-level, the information is aggregated per type to 4 representative feature vectors $\in \mathbb{R}^{12}$. These vectors are the centroids of clusters determined using a weighted k-means variant. The weight of each of the **4 centroids** is saved as a scalar. Cluster centroids are then clipped and normalised to $[0, 1]$. For a more detailed description of our feature processing, we refer to [7].

In STOB11, the mean and standard deviation of the chroma and timbre features, each $\in \mathbb{R}^{12}$ are used. Moreover, following Donaldson et al. [1], global analysis results for each clip, including **key**, **mode**, **loudness**, **tempo**, **time signature**, **energy** and the recent Echo Nest API **danceability** features are used to represent the content of the clips. Note that these 11 distinct features are compared separately, as described in Section 3.

2.2 Text-based Feature Data

Social tagging data has been proven to enable music recommendation both in research and in commercial application. In line with findings of Novello et al.[5], we assume a strong connection between the way we describe music and the way we compare it. The Magnatune label, marketing the songs used in the TagATune game, provides 1-3 genre terms for the songs in our dataset, using a vocabulary of **44 genres**. For each clip, we represent these annotations as 44 binary values.

STOB11 uses the tag annotations provided in the MagnaTagATune dataset. As the tags assigned to the clips during the TagATune game are distributed rather sparsely, they combine several tags on the basis of singular/plural forms, spelling correction and semantic similarity. After filtering unused tags, they arrive at a vocabulary of **99 tags**, each represented by a single binary value.

2.3 Similarity Data

In the bonus mode of TagATune, two players try to agree on one outlier in a presented triplet of audio clips. The MagnaTagATune dataset contains 7650 votings in 346 triplets, referring to 1019 clips. The information of each voting, e.g. C_k being the outlier in (C_i, C_j, C_k) can be used to derive two relative similarity constraints:

$$\begin{aligned} & (C_i, C_j) \overset{\text{sim}}{>} (C_j, C_k) \\ \wedge \quad & (C_i, C_j) \overset{\text{sim}}{>} (C_i, C_k), \end{aligned} \tag{1}$$

where the relation $\overset{\text{sim}}{>}$ means "more similar than". The resulting relations can be represented as edges in a directed graph of pairs of clips, as presented by McFee et al.[3]. After removing contradicting edges, Stober et al. select 674 relations as similarity constraints using a randomised approach. This selection then forms their ***all constraints* set**. Based

Table 1: Comparison of feature dimensions (Feat.) representing the clips, and the number of parameters (#P) for the similarity measures in WOLF11 and STOB11.

Features	Feat. WOLF11	Feat. STOB11	#P WOLF11 (MLR)	#P WOLF11 (DMLR)	#P STOB11
chroma	52	24	$52 \cdot 148$	52	2
timbre	52	24	$52 \cdot 148$	52	2
tags	44	99	$44 \cdot 148$	44	99
global features	/	7	/	/	7
$\sum$	148	154	21904	148	110

on their code we have randomly selected 10 of such sets, which are used for the comparison in Section 4.

3. MODELLING MUSIC SIMILARITY

The approaches compared in this paper use different architectures for defining a similarity function on the clips' features, thereby offering different means for adaptation and later interpretation of the results.

In WOLF11 we model similarity on clips using Mahalanobis distances, representing pseudometrics for measuring vector distances. In order to compare two clips C_i and C_j, we concatenate each clip's features into a single feature vector x_i, and x_j, respectively. On these vectors, the Mahalanobis distance is defined as

$$d_W(C_i, C_j) = \sqrt{(x_i - x_j)^T W (x_i - x_j)}, \qquad (2)$$

where $x_i, x_j \in \mathbb{R}^N$ and $\mathbf{W} \in \mathbb{R}^{N \times N}$ represents the positive semi-definite transformation defining the metric. The actual dimensions of feature vectors and the resulting parameters in W can be read in Table 1. If W is **diagonal**, the resulting distance measure describes a weighted Euclidean distance metric.

We train our model using Metric Learning to Rank (MLR) algorithm by McFee et al.[4]. This is a structural SVM based quadratic optimisation algorithm, which we use with a linear kernel function. Their MLR toolkit allows for a diagonal restriction of W, and results for this mode are labeled as **DMLR**.

In STOB11, a similarity model is parametrised by weighting a sum of individual, predefined distance measures based on the individual feature types, which they term *facets* and *facet distances*, respectively. The **facet weight vector w** is defined as follows:

$$\begin{aligned} d_w(C_i, C_j) &= \sum_{i=1}^{l} w_i \delta_{f_i}(x_i - x_j), \quad \text{where} \qquad (3)\\ w_i &\geq 0 \\ \sum_{i=1}^{l} w_i &= l. \end{aligned}$$

Here, δ_{f_i} represent semi-metrics only operating on the subspace of the specified feature type f_i. In their experiments, there are $l = \mathbf{110}$ **facets** (see Table 1) and thus $w \in \mathbb{R}^{110}$. For determining w, a gradient descent approach, a maximum margin approach and several quadratic programming approaches with different slack formulations allowing violated constraints are compared. We will compare our results to those of two selected candidates from STOB11: The quadratic optimisation minimising slack in the quadratic part of the problem **QPMIN** (referenced as QPmin(ξ^2) in [6]), and the SVM-based **LIBLINEAR**. LIBLINEAR was selected for its superior performance. As the non-negativity in Equation 3 was violated in this case, QPMIN has been selected for further comparison for its high average performance across the QP methods tested.

3.1 Model Comparison

The modelling of similarity in STOB11 allows for feature-specific specially engineered similarity functions. Parametrisation is only applied at the second level, the weighted addition of the individual feature outputs. Although allowing for less freedom in the automatic adaptation of the similarity measure, given a user-understandable set of features and conventional similarity functions on these, this model allows for a straightforward display and goal-directed manual manipulation of the weightings by users.

Our model in WOLF11 applies the parametrisation at a more technical level, linearly weighting and relating different entries across all features. The number of parameters in this model is very high (approximately the dimension of features times the parameter count in STOB11). The resulting Mahalanobis matrix W can be compared to other such matrices by comparing the implied Gaussian distributions. Moreover, the parametrised pseudometric allows for a standard geometric interpretation of the results.

4. EXPERIMENTS

In their experiments, Stober et al.[6] include the training sets in their test sets, focusing on the general ability of the algorithms to adapt to the complete data set. The figures below show the average performance over **10** randomly extracted ***all constraints*** **sets** using the methods from STOB11 as pointed out in Section 2.3. For each of these *all constraints* sets, subsets of training data, increasing in size, are randomly selected. For each subset size, **5 training sets** are selected randomly. The different algorithms and feature configurations are evaluated on all data within the respective *all constraints* set, including the training data. The specific training performance is also plotted.

4.1 QPMIN, LIBLINEAR, MLR and DMLR

In [8], we have evaluated, amongst others, **MLR** variants on MagnaTagATune. In Figure 1, the performances of MLR and DMLR, using the combined content-based and tag-based features, are compared to QPMIN and LIBLINEAR in STOB11. Considering the number of constraints which can be trained, MLR shows the best performance by **learning all of the constraints** in the set. This is probably related to the greater flexibility which comes with the

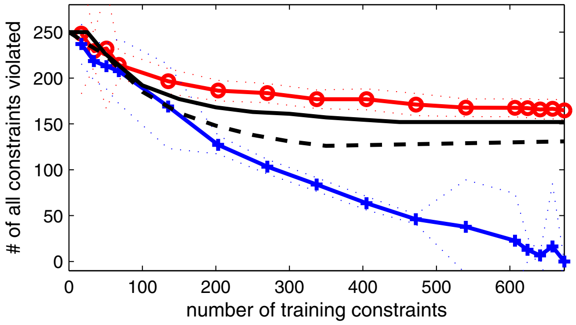

Figure 1: Learning results for increasing training set size. The plots reflect the number of violated constraints in the *all constraints* set. Dotted curves display the standard deviation across training set variations. Top to bottom: DMLR(O), QPMIN(-), LIBLINEAR(--), and MLR(+).

200 times higher number of parameters in W than in w (see Table 1). The dotted MLR variance curve shows that, until every constraint is trained, the general performance can vary strongly depending on the sampling of the training set. DMLR performs similar to, but slightly worse than QPMIN, violating at least 152 constraints. This was expected as the weighted Euclidean distance is similar to the weighting used in STOB11. Moreover, the number of parameters only differ by 38. The results of LIBLINEAR show a maximum performance of 130 violated constraints. This comes with slight violations of the non-negativity constraints in Equation 3.

4.2 Influence of Feature Type

In [7], we measured the influence of the different feature types described in Section 2 on the adaptation performance of MLR. Here, we want to extend these using test results from the application on the *all constraints* data.

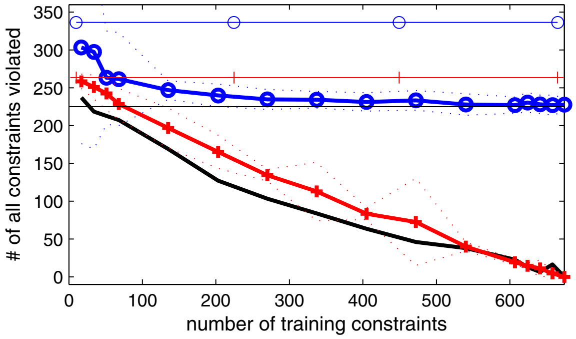

Figure 2: Violated constraints on the *all constraints* set. Baselines are plotted in style of features. Top to bottom: Genre(O), audio(+) and combined features(-).

In Figure 2, the results for using **fully parametrised MLR** with tag-only, genre-only and combined features are compared. Not surprisingly, the combined features show superior performance. In contrast to the results in [7], with the *all constraints* data the **genre features do not correlate** with the user votings. This is both reflected in the baseline as well as in the relatively poor adaptation behaviour, leaving 120 constraints violated.

In [7] we have shown that the audio features, having a greater dimensionality, allow for a better adaptation despite a low baseline at the start of the process. In this experiment, the **audio features** allow for a **complete satisfaction** of the constraints.

4.3 Generalisation

The ability to generalise over unknown data determines the success of trained music similarity measures in the application to music recommendation. In Figure 3, the average errors on the unknown parts of the test set, regarding the training subsets used in Figure 1 are plotted. For small training sets, the genre features perform similar to the audio features. As the number of known constraints rises over 180, the genre features enabling only slightly greater fulfilment of unknown similarity constraints.

The particular comparison of the audio-only and combined features is interesting as well: As Figure 3 shows, the audio features, though showing almost equal performance in Figure 3, leave a **large amount of violations** in the unknown parts of the training set. Thus, the generalisation on these features is rather low, but the **data allows for a specific learning** of training set feature vectors' similarity values, leading to the final fulfilment of all constraints. The binary nature of the genre features renders some of the clips inseparable when only relying on this information. Moreover, their lower dimensionality allows for fewer parameters to be optimised. However, when combined with the audio features, they do not only improve learning on smaller training sets somewhat, but they also improve generalisation on the unknown test set markedly.

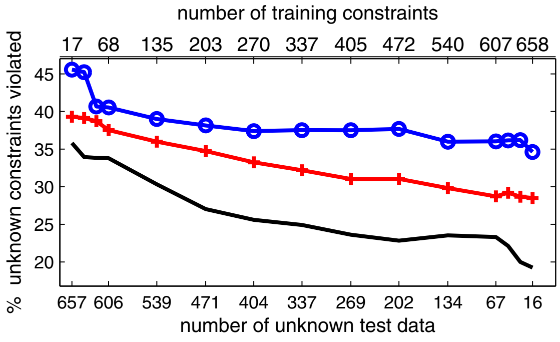

Figure 3: Learning results for increasing training set size. The plots reflect the percentage of violated unknown test constraints. Values for training sizes 0 (see baselines Fig. 2) and 674 (no unknown data left) have been omitted. Features: Genre(O), audio(+) and combined(-).

5. DISCUSSION

In the previous sections, we have compared two different approaches for modelling music similarity, and adapting the models to user votings. The presented experiments show that the two-level approach of linear weighting complex features in STOB11 performs slightly better than our approach when restricted to low-level features. However, both approaches fail to learn about 24% of the constraints and the differences may be due to the applied adaptation models. The full matrix MLR model has shown better performance, fulfilling all constraints after adaptation. Using the full feature set, even the generalisation error drops to 20%. Although all of the constraints can be learned even with only content-based features, the generalisation is clearly improved by including the genre features, indicating that they provide additional information no present or easily accessible in the audio data.

We feel that both approaches presented here are very much worth pursuing: using powerful complex learning models to adapt to different user groups and providing users with interfaces to enable personal adjustments. This raises the new question of how they could be combined. The better learning performance of the full Mahalanobis matrix comes at the price of additional complexity, which makes it hardly suitable for presentation to users, even experts, for manual fine-tuning. To enable the use of the more powerful models and to enable user interaction, new approaches are needed. Possible directions for future work are the user interaction on the level of pre-learned complex models using low-level features, the reduction of model parameters using clustering, or the pre-learning of complex features that better correlate with similarity data.

6. ACKNOWLEDGEMENTS

We thank Sebastian Stober for providing the code to generate the dataset used in this paper and Brian McFee for providing the MLR code.

7. REFERENCES

[1] J. Donaldson and P. Lamere. Using visualizations for music discovery. Tutorial at the 10th Int. Conf. on Music Information Retrieval (ISMIR), 2009.

[2] E. Law and L. V. Ahn. Input-agreement: A new mechanism for collecting data using human computation games. In *Proc. of CHI*. ACM Press, 2009.

[3] B. McFee and G. Lanckriet. Heterogeneous embedding for subjective artist similarity. In *Proc. International Symposium on Music Information Retrieval (ISMIR)*, 2009.

[4] B. Mcfee and G. Lanckriet. Metric learning to rank. In *Proceedings of the 27th annual International Conference on Machine Learning (ICML)*, 2010.

[5] A. Novello, M. F. Mckinney, and A. Kohlrausch. Perceptual evaluation of music similarity. In *Proceedings of the 7th International Conference on Music Information Retrieval (ISMIR)*, 2006.

[6] S. Stober and A. Nürnberger. An experimental comparison of similarity adaptation approaches. In *Proc. of 9th International Workshop on Adaptive Multimedia Retrieval (AMR)*, Barcelona, Spain, Jul 2011. To appear.

[7] D. Wolff and T. Weyde. Combining sources of description for approximating music similarity ratings. In *Proc. of 9th International Workshop on Adaptive Multimedia Retrieval (AMR)*, Barcelona, Spain, Jul. To appear.

[8] D. Wolff and T. Weyde. Adapting metrics for music similarity using comparative judgements. In *Proc. International Symposium on Music Information Retrieval (ISMIR)*, 2011.