A WEB-BASED APPROACH TO ASSESSING ARTIST SIMILARITY USING CO-OCCURRENCES

Markus Schedl, Peter Knees, Gerhard Widmer

Department of Computational Perception
Johannes Kepler University (JKU) Linz
Altenberger Straße 69
A-4040 Linz, Austria

and

Austrian Research Institute for Artificial Intelligence (ÖFAI)
Freyung 6/6
A-1010 Vienna, Austria

markus.schedl@jku.at

ABSTRACT

In this paper, we present a similarity measure for music artists based on search results of Google queries. Co-occurrences of artist names on web pages are analyzed to measure how often two artists are mentioned together on the same web page. We estimate conditional probabilities using the extracted page count. These conditional probabilities give a similarity measure which is evaluated using a data set containing 224 artists from 14 genres. For evaluation, we use two different methods, intra-/intergroup-similarities and k-Nearest Neighbors classification. Furthermore, a confidence filter and combinations of the results gained from three different query settings are tested. It is shown that these enhancements can raise the performance of our similarity measure. Comparing our results to those of similar approaches show that our approach, though being quite simple, performs well and can be used as a similarity measure that incorporates "social knowledge".

Keywords: artist similarity, genre classification, web mining, co-occurrence analysis

1. INTRODUCTION

Elaborating methods for defining similarities between pieces of music or artists is an important task in the field of music information retrieval (MIR). It allows to form clusters of songs or artists which are similar according to certain aspects. Furthermore, similarity measures can be used to classify unknown songs or artists, to generate playlists containing similarly sounding tracks, to recommend new titles to users (recommender systems), or to visualize music repositories [9, 13].

There exist a number of similarity measures which are based on low-level audio features, e.g. [10, 12, 7, 2]. Such low-level features are extracted directly from the audio signal and incorporate, for example, rhythmic or timbral properties. Although it has been shown that signal-based similarity measures perform well for various classification tasks, e.g. [3, 11], they also have certain disadvantages. The most apparent limitation is that signal-based measures obviously depend on the audio signal, e.g. in the form of digital music files. If no audio data is available, e.g. in a scenario where an individual wants to extend his/her collection with unknown music, signal-based measures cannot be applied. Moreover, feature extraction and calculation of similarities are usually very time-consuming tasks since they include computationally complex operations like the calculation of *Mel Frequency Cepstral Coefficients (MFCCs)* [7, 2] or *Gaussian Mixture Models (GMMs)* [2, 1].

An alternative approach, which is used here, is based on text mining in general, in particular on web mining. It comprises extracting and analyzing information from the Internet. The various sites of the World Wide Web reflect the opinions of a large number of different people, interest groups and companies. This is a kind of "social knowledge" that we wish to extract via text mining and use to assess the similarity of artists. Approaches like the one presented here offer the advantage of including cultural information and can be used independently of any audio signal. However, web-based text mining obviously depends on the existence of web pages dealing with the research topic. If such web pages cannot be made out, e.g. because the query for the search engine cannot be defined adequately or comprises ambiguous words, web-based approaches run into serious problems. For example, a search for music-related web pages offering information about artists like “Kiss”, “Bush”, or “Porn” will most probably reveal a huge number of sites not dealing with these artists. Regardless of this shortcoming, web-based music similarity measures in most cases produce valuable results.

The remainder of this paper is organized as follows. Related work is briefly summarized and compared to our work in Section 2. In Section 3, we present the used methods and different query settings. Section 4 describes the conducted experiments and evaluation results and compares them to similar approaches. Finally, in Section 5, we draw conclusions and point out possible future research directions.

2. RELATED WORK

As for related work, only few publications on the topics of web mining and co-occurrence analysis for MIR exist. First steps into this direction can be found in [8], where radio station playlists and compilation CD databases are used to find co-occurrences between titles and between artists. In [5, 14], user collections from “OpenNap”, a music sharing service, are analyzed to obtain a similarity measure based on community metadata. The artist co-occurrences extracted from these collections are evaluated by comparison with direct subjective similarity judgments obtained via a web-based survey. In contrast to this survey of non-professionals, in [4], expert opinions of professional editors taken from the “All Music Guide (AMG)”[1] are used to create a similarity network. To this end, the “similar artists” links of 400 artists are extracted from AMG. Furthermore, playlist co-occurrences from “The Art of the Mix”[2] are visualized by a network containing more than 48.000 artists. An approach that uses the same text mining technique as [14] on artist information retrieved from web pages is presented in [6]. This approach, like the one presented here, is based on results of queries to search engines. In [6], however, complete artist-related web pages, rather than only page counts, are retrieved and analyzed with respect to word occurrences. Thereafter, a common text mining technique, namely *term frequency · inverse document frequency (tf · idf)*, is used for weighting the extracted words. *Support Vector Machines (SVMs)* and *k-Nearest Neighbors (k-NN)* are used for classification and evaluation. We also use k-NN and will directly compare our results to those presented in [6]. Also investigating co-occurrences of artist names on web pages, [15] presents a similar approach to ours, but uses a slightly different similarity measure. Starting with a seed artist, the Amazon[3] web service “Listmania!” is used to obtain a list of potentially related artists. Based on this list, co-occurrences are derived by querying Google. Thereafter, the “relatedness” of each “Listmania!”-artist to the seed artist is calculated as the ratio between the combined page count and the minimum of the single page counts for both artists.

The approach to be presented in this paper differs from the above approaches in that we, unlike Zadel and Fujinaga in [15], calculate complete distance matrices. This offers additional information since we can also predict which artists are **not** similar. Such information is necessary, for example, when it comes to creating playlists that incorporate a broad variety of different music styles. Moreover, in [15], artists are extracted from “Listmania!”, which uses the database of the web shop Amazon. The number of artists in this database is obviously smaller than the number of artist-related web pages indexed by Google. For example, most local artists or artists without a record deal are not contained. Thus, the approach of [15] cannot be used for such artists.

Considering the normalization used in [15] (minimum of the single page counts for both artists), another advantage of our approach becomes apparent. Since we use an asymmetric distance matrix, cf. Section 3, our measure incorporates more information. The resulting positive impact will be shown in the experiments in Subsection 4.1.

A shortcoming of our co-occurrence approach is that creating a complete distance matrix has quadratic computational complexity in the number of artists. Despite this fact, our approach is quite fast for small- and medium-sized collections with some hundreds of artists since it is very simple and does not rely on extracting and weighting hundreds of thousands of words like the *tf · idf* approach of [6]. Moreover, using heuristics could reduce the computational complexity.

3. WEB MINING BY CO-OCCURRENCE ANALYSIS

Since our similarity measure is based on artist co-occurrences, we need to count how often artist names are mentioned together on the same web page. To obtain these page counts, the search engine Google was used. Google has been chosen for the experiments because it is the most popular search engine at the moment. Furthermore, investigations of different search engines showed that Google yields the best results for musical web crawling [6].

Given a list of artist names, we use Google to estimate the number of web pages containing each artist and each pair of artists. Since we are not interested in the content of the found web pages, but only in their number, the search is restricted to display only the top-ranked page. In fact, the

[1] *http://www.allmusic.com*
[2] *http://www.artofthemix.org*
[3] *http://www.amazon.com*

only information we use is the page count that is returned by Google. This raises performance and limits web traffic. The outcome of this procedure is a symmetric matrix C, where element c_{ij} gives the number of web pages containing the artist with index i together with the one indexed by j. The values of the diagonal elements c_{ii} show the total number of web pages containing artist i. Based on the page count matrix C, we then use relative frequencies to calculate a conditional probability matrix P as follows. Given two events a_i (artist with index i is mentioned on web page) and a_j (artist with index j is mentioned on web page), we estimate the conditional probability p_{ij}(the probability for artist j to be found on a web page that is known to contain artist i) as shown in Formula 1.

$$p\left(a_i \wedge a_j \mid a_i\right) = \frac{c_{ij}}{c_{ii}} \tag{1}$$

Obviously, P is not symmetric. Since we need a symmetric similarity function in order to use k-NN, we compute a symmetric equivalent P_s by simply calculating the arithmetical mean of p_{ij} and p_{ji} for every pair of artists i and j.

Addressing the problem of finding only music-related web pages, we used three different query settings.

- *"artist1" "artist2"* music
- *"artist1" "artist2"* music review
- allintitle: *"artist1" "artist2"*

The first one, in the following abbreviated as *M*, searches only for web pages containing the two artist names as exact phrases and the word "music". The second one, which has already been used in [14], restricts the search to pages containing the additional terms "music" and "review". This setting, abbreviated as *MR*, was used to compare our results to those of [6]. The third setting (*allintitle*) only takes into consideration web pages containing the two artists in their title. It is the most limiting setting, and the resulting page count matrices are quite sparse. However, our evaluation showed that this setting performs quite well on the k-NN classification task and can be used successfully in combination with *M* or *MR*.

4. EXPERIMENTS AND EVALUATION

We conducted our experiments on the data set already used in [6]. It comprises 14 quite general and well-known genres with 16 assigned artists each. A complete list can be found on the Internet[4]. Two different evaluation methods were used: ratios between intra- and intergroup-similarities and hold-out experiments using k-NN classification.

[4] *http://www.cp.jku.at/people/schedl/music/artist_list_224.pdf*

4.1. Intra-/Intergroup-Similarities

This evaluation method is used to estimate how well the given genres are distinguished by our similarity measure P. For each genre, the fraction between the average intragroup-probability and the average intergroup-probability is calculated. The higher this ratio, the better the differentiation of the respective genre. The average intragroup-probability for a genre g is the probability that two arbitrarily chosen artists a and b from genre g co-occur on a web page that is known to contain either artist a or b. The average intergroup-probability for a genre g is the probability that two arbitrarily chosen artists a (from genre g) and b (from any other genre) co-occur on a web page that is known to contain either artist a or b. Thus, the average intragroup-probability gives the probability that two artists from the same genre co-occur. The average intergroup-probability gives the probability that an artist from genre g co-occurs with an artist not from genre g.

Let A be the set of all artists and A_g the set of artists assigned to genre g. Formally, the average intra- and intergroup-probabilities are given by Equations 2 and 3, where $|\,A_g\,|$ is the cardinality of A_g and $A \setminus A_g$ is the set A without the elements contained in the set A_g.

$$intra_g = \frac{\sum_{a_1 \in A_g} \sum_{a_2 \in A_g}^{a_2 \neq a_1} p_{a_1 a_2}}{|\,A_g\,|^2 - |\,A_g\,|} \tag{2}$$

$$inter_g = \frac{\sum_{a_1 \in A_g} \sum_{a_2 \in A \setminus A_g} p_{a_1 a_2}}{|\,A \setminus A_g\,| \cdot |\,A_g\,|} \tag{3}$$

Obviously, the ratio $intra_g / inter_g$ should be at least greater than 1.0 if the similarity measure is to be of any use.

4.1.1. Results and Discussion

Table 1 shows the results of evaluating our co-occurrence approach with this first evaluation method. It can be seen that the *allintitle*-setting yields the best results as the average intergroup-similarities are very low. Hence, nearly no artists from different genres occur together in the title of the same web page. Especially for the genres "Jazz" and "Classical", the results are excellent. However, for "Alternative Rock/Indie" and "Electronica", the ratios are quite low. This can be explained by the low average intragroup-similarities for these genres. Thus, artists belonging to these genres are seldom mentioned together in titles. Analyzing the page count matrices revealed that the *allintitle*-setting yields good results if web pages containing artists from the same genre in their title are found. If not, the results are obviously quite bad. This observation motivated us to conduct experiments with confidence filters and combinations of the *allintitle*-setting with *M* and *MR*. These experiments are described in detail in the next section.

keywords	music, review			music			allintitle		
genre	**intra_avg**	**inter_avg**	**ratio**	**intra_avg**	**intra_avg**	**ratio**	**intra_avg**	**inter_avg**	**ratio**
Country	0.088	0.032	2.723	0.104	0.039	2.644	2.170e-3	3.092e-5	70.163
Folk	0.052	0.098	0.529	0.054	0.039	1.374	5.877e-4	2.453e-5	23.963
Jazz	0.094	0.038	2.460	0.132	0.039	3.399	5.052e-3	2.377e-5	212.534
Blues	0.132	0.026	5.085	0.106	0.024	4.483	2.058e-3	2.377e-5	58.496
RnB/Soul	0.068	0.032	2.148	0.078	0.044	1.780	9.400e-4	3.785e-5	24.839
Heavy Metal/Hard Rock	0.208	0.126	1.649	0.267	0.083	3.206	8.808e-4	5.708e-5	15.432
Alternative Rock/Indie	0.091	0.072	1.261	0.191	0.079	2.426	3.733e-4	8.822e-5	4.232
Punk	0.139	0.098	1.419	0.192	0.067	2.860	1.109e-3	2.300e-5	48.210
Rap/Hip-Hop	0.110	0.066	1.654	0.153	0.055	2.798	1.855e-3	7.092e-5	26.159
Electronica	0.074	0.042	1.774	0.134	0.047	2.872	4.494e-4	5.014e-5	8.962
Reggae	0.135	0.048	2.807	0.072	0.036	2.013	9.807e-4	2.847e-5	34.455
Rock 'n' Roll	0.075	0.041	1.817	0.086	0.045	1.899	1.556e-3	6.248e-5	24.907
Pop	0.134	0.066	2.040	0.178	0.072	2.470	1.501e-3	8.023e-5	18.819
Classical	0.312	0.010	31.733	0.201	0.011	18.177	1.154e-2	4.504e-6	2561.504
mean			4.221			3.743			223.762

Tab. 1. Results of the evaluation of intra-/intergroup-similarities using our co-occurence measure. On the left, the results for the queries using the additional keywords *+music+review* are illustrated. The middle columns show the results for the queries with additional *+music*. The rightmost columns show the results for the queries only taking into account web pages with artists in their title. For each genre, the **average intragroup-probability**, the **average intergroup-probability** and the **ratio** between these two probabilities is depicted. The higher the **ratio**, the better the differentiation of the respective genre.

Moreover, Table 1 shows that, aside from “Classical”, “Blues” is distinguished quite well. Also remarkable is the very bad result for “Folk” music in the *MR*-setting. This may be explained by intersections with other genres, e.g. “Country”.

The approach presented in [15] was tested on the list of artists already used in [6]. The results, which are visualized in Table 2, are slightly worse than the results using our approach on the same data set. An explanation for this is that we use an asymmetric similarity measure that, for each pair of artists (*artist1* and *artist2*), incorporates probability estimations for *artist1* being mentioned on web pages containing *artist2* and for *artist2* appearing on web pages of *artist1*. This additional information is lost when using the normalization method proposed in [15].

In Table 3, the evaluation results for the approach of [6], again using exactly the same list of artists, are depicted. To obtain them, the distances between the feature vectors gained from the $tf \cdot idf$ calculations are computed for every pair of artists. This gives a complete similarity matrix. Since most of the query settings used in [6] differ from ours, we can only compare the results of the *MR*-setting. Taking a closer look at the results shows that $tf \cdot idf$ performs better for eight genres, our approach performs better for six genres. However, the mean of the ratios is better for our approach because of the high value for the genre “Classical”. A possible explanation is that web pages concerning classical artists often also contain words which are used on pages of other genres' artists. In contrast, classical artist names seem to be mentioned only together with other artists belonging to the same genre, which is reflected by the very high ratios of our approach for this genre.

4.2. Classification with k-Nearest Neighbors

The second set of evaluation experiments was conducted to show how well our similarity measure works for classifying artists into genres. For this purpose, the widely used technique of k-Nearest Neighbors was chosen. This technique simply uses the k data items for prediction that have a minimal distance to the item that is to be classified. The most frequent class among these k data items is predicted for the unclassified data item. As for the partitioning of the complete data set into training set and test set, we used different settings, referred to as *tx*, where *x* is the number of data items from each genre that are assigned to the training set. In a *t15*-setting, for example, 15 artists from each genre are used for training and one remains for testing. For measuring the distances between two data items, we use the similarities given by the symmetric probability matrix P_s. We ran all experiments 1.000 times to minimize the influence of statistical outliers on the overall results. The *accuracy*, in the following used for measuring performance, is defined as the percentage of correctly classified data items over all classified data items in the test set. Since the usage of confidence filters may result in unclassified data items, we introduce the *prediction rate* which we define as the percentage of classified data items in the complete test set.

In a first test with setting *t8*, k-NN with $k = 9$ performed best, so we simply used 9-NN for classification in the subsequent experiments. It is not surprising that values around 8 perform best in a *t8*-setting, because in this case the number of data items from the training set that are used for predic-

keywords	music, review			music			allintitle		
genre	**intra_avg**	**inter_avg**	**ratio**	**intra_avg**	**intra_avg**	**ratio**	**intra_avg**	**inter_avg**	**ratio**
Country	0.136	0.050	2.725	0.150	0.058	2.591	2.988e-3	5.401e-5	55.328
Folk	0.080	0.159	0.502	0.082	0.058	1.340	1.115e-3	4.294e-5	25.962
Jazz	0.129	0.059	2.273	0.180	0.056	3.235	6.585e-3	3.842e-5	171.398
Blues	0.178	0.040	4.448	0.154	0.036	4.222	3.125e-3	5.572e-5	56.080
RnB/Soul	0.097	0.050	1.950	0.107	0.065	1.655	1.180e-3	5.719e-5	20.627
Heavy Metal/Hard Rock	0.295	0.185	1.592	0.379	0.122	3.112	1.517e-3	1.095e-4	13.857
Alternative Rock/Indie	0.141	0.116	1.209	0.286	0.118	2.430	7.118e-4	1.622e-4	4.389
Punk	0.201	0.140	1.429	0.272	0.097	2.796	1.591e-3	3.888e-5	40.909
Rap/Hip-Hop	0.164	0.097	1.683	0.223	0.080	2.774	3.256e-3	1.169e-4	27.850
Electronica	0.111	0.062	1.798	0.187	0.068	2.758	6.581e-4	9.009e-5	7.305
Reggae	0.216	0.086	2.513	0.111	0.058	1.934	1.622e-3	4.808e-5	33.745
Rock 'n' Roll	0.117	0.065	1.793	0.131	0.069	1.905	2.199e-3	9.718e-5	22.630
Pop	0.210	0.107	1.952	0.257	0.112	2.302	2.316e-3	1.387e-4	16.698
Classical	0.423	0.016	26.438	0.270	0.016	16.592	1.548e-2	7.556e-6	2048.574
mean			3.736			3.551			181.811

Tab. 2. Results of the evaluation based on intra-/intergroup-similarities using relatednesses according to [15].

keywords	music, review		
genre	**intra_avg**	**inter_avg**	**ratio**
Country	0.118	0.049	2.425
Folk	0.064	0.043	1.480
Jazz	0.131	0.048	2.722
Blues	0.134	0.047	2.875
RnB/Soul	0.109	0.060	1.812
Heavy Metal/Hard Rock	0.080	0.049	1.618
Alternative Rock/Indie	0.075	0.049	1.521
Punk	0.098	0.053	1.848
Rap/Hip-Hop	0.129	0.050	2.545
Electronica	0.077	0.039	1.985
Reggae	0.135	0.045	3.025
Rock 'n' Roll	0.105	0.050	2.099
Pop	0.081	0.052	1.577
Classical	0.230	0.025	9.164
mean			2.621

Tab. 3. Results of the evaluation based on intra-/intergroup-similarities using the $tf \cdot idf$ approach according to [6].

tion equals the number of data items chosen from each class to represent the class in the training set. The *t8*-setting without any confidence filter gives accuracies of about 69% for *M*, about 59% for *MR* and about 74% for *allintitle*. Using setting *t15*, these results can be improved for *M* (≈75% using 9-NN) and for *allintitle* (≈80% using 6-NN). For *MR*, no remarkable improvement could be achieved.

In the case that no confidence filter is used, like in the first tests described above, a random genre is predicted for the artist to be classified if his/her similarity to all artists in the training set is zero. Due to the sparseness of its similarity matrix, this problem mainly concerns the *allintitle*-measure. To overcome the problem and benefit from the good performance of the *allintitle*-measure but also address the sparseness of the respective similarity matrix, we tried out some confidence filters to combine the similarity measures that use the three different query settings. The basic idea is to use the *allintitle*-measure if the confidence in its results is high enough. If not, the *M*- or *MR*-measure is used to classify an unknown data item. We experimented with confidence filters using mathematical properties of the distances between the unclassified data item and its nearest neighbors. The best results, however, were achieved with a very simple approach based on counting the number of elements with a probability/similarity of zero in the set of the nearest neighbors. If this number exceeds a given threshold, the respective data item is not classified with the *allintitle*-measure, but the *M*- or *MR*-measure is used instead. Using this method, only artists that co-occur at least with some others in the title of some web pages are classified with *allintitle*. On the other hand, if not enough information for a certain artist is available in the *allintitle*-results, *MR* or *M* is used instead. These two measures usually give enough information for prediction. Indeed, their prediction

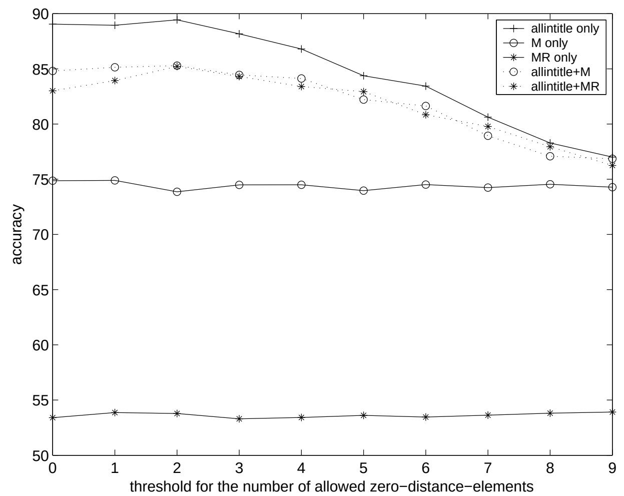

Fig. 1. Accuracies in percent for single and combined similarity measures using 9-NN *t15*-validation and the confidence filter. The combined results are depicted as dotted lines. It is remarkable that the high values for the *allintitle*-accuracies come along with up to 18% of unpredictable artists. All other measures (single and combined) leave no data items unpredicted.

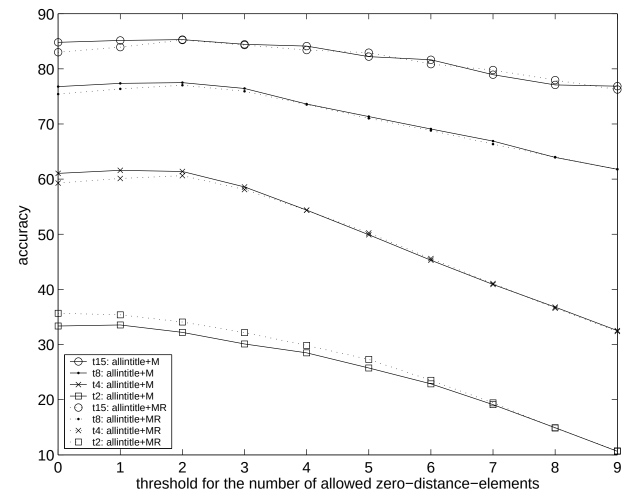

Fig. 2. Accuracies in percent for different combinations of the three settings *(allintitle*, *M*, *MR*) and different training set sizes. 9-NN classification was used.

rates equal 100% for the data set used for our evaluations. This is also manifested in Figure 1 which shows that the accuracies for *MR* and *M* are independent of the threshold for the confidence filter.

4.2.1. Results and Discussion

We already mentioned the classification accuracies of up to 80% for uncombined measures. Since we wanted to analyze to what extent the performance can be improved when using combinations, we conducted *t15*-validations using either a single measure or combinations of *allintitle* with *MR* and *M*. The results are shown in Figure 1. Along the abscissa, the influence of different thresholds for the confidence filter can be seen. The falling accuracies for *allintitle* with raising threshold values confirms our assumption that the performance of the *allintitle*-measure depends strongly on the availability of enough information. It is important to note that the uncombined *allintitle*-measure does not always make a prediction when using the confidence filter, also cf. Figure 3. Remarkable are the very high accuracies (fraction between correctly classified artists and classifiable artists) of up to 89,5% for *allintitle* with a threshold value of 2. However, in this setting, 14% of the artists cannot be classified. Taking a closer look at the *MR*- and *M*-settings shows that they reach accuracies of about 54% and 75% respectively and that these results are independent of the threshold for the confidence filter. In fact, *MR* and *M*, at least for the used data set, always provide enough information for prediction. Combining the measures by taking *allintitle* as primary one and, if no prediction with it is possible, *MR* or *M* as fallback also combines the advantages of high accuracies and high prediction rates. Indeed, using the combination *allintitle+M* gives accuracies of 85% at 100% prediction rate. Since the accuracies for *M* are much higher than for *MR*, the combination of *allintitle* with *M* yields better results than with *MR*. Compared to the k-NN results of [6], these accuracies are at least equal although the co-occurrence approach is much simpler than the $tf \cdot idf$ approach. However, the single *MR*-setting performs quite poorly with our approach. This can be explained by the fact that web pages containing music reviews seldom mention other artists, but usually compare new artists' albums to more recent ones by the same artist.

In addition, we were interested in the number of artists needed to define a genre adequately. For this reason, we ran some experiments using different training set sizes. In Figure 2, the results of these experiments for 9-NN classification using the combinations *allintitle+M* and *allintitle+MR* are depicted. It was observed that *t15* and *t8* again provide very high accuracies of up to 85% and 78% respectively. Examining the results of the *t4*- and *t2*-settings reveals much lower accuracies. These results are remarkably worse than those of [6] for the same settings (61% for *t4* with our approach using 9-NN vs. 76% with the $tf \cdot idf$ approach using 7-NN and the additional search keywords "music genre style", 35% for *t2* with our approach vs. 43% with the $tf \cdot idf$ approach using 7-NN and the same additional keywords). In these two settings, the additional information used by the $tf \cdot idf$ approach seems to be highly valuable. As a final remark on Figure 2, we want to point out that the prediction

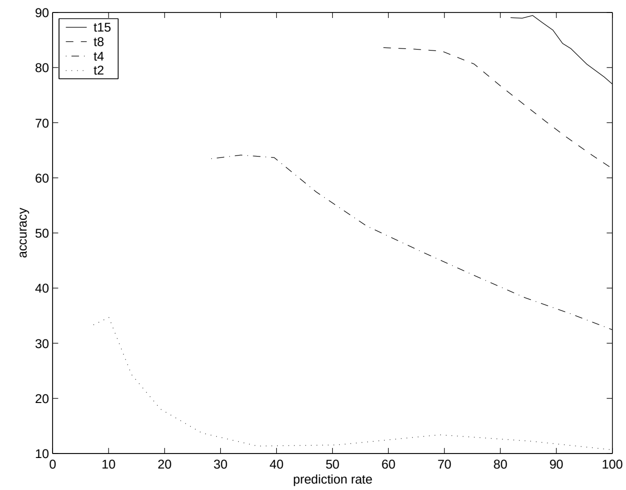

Fig. 3. Accuracy plotted against prediction rate for different training set sizes and 9-NN classification. Only the uncombined *allintitle*-setting was used for this plot.

correct genres \ predicted genres	Co	Folk	Jazz	Bl	RnB	HM	AR	Punk	Rap	El	Re	RnR	Pop	Cl
Co	86.5						7.5					6		
Folk		82.9				6.7			10.5					
Jazz			100											
Bl				100										
RnB					85.8							7.2	7.1	
HM	0.5					95.3		4.1					0.1	
AR				1.8	5.9	6.6	49.5	1.4	0.1	27.4	6.1		1.1	
Punk							6.5	88.1					5.4	
Rap							4.5		89.1				6.5	
El							1.1			94.1			4.8	
Re											100			
RnR												95	5	
Pop			1		0.1	5.2	1						87.2	5.5
Cl														100

Fig. 4. Confusion matrix for the averaged results of 1.000 runs using 9-NN *t15*-validation. The confidence filter was applied to the *allintitle*-setting. The values are the average accuracies in percent.

rate for all depicted experiments is 100%.

As already mentioned, the uncombined *allintitle*-setting using the confidence filter does not always yield a prediction. To analyze the trade-off between accuracy and prediction rate, we plotted these properties for the *allintitle*-setting in Figure 3. This figure shows that, in general, an increase in accuracy goes along with a decrease in prediction rate. However, an increase in prediction rate accompanied by a slight increase in accuracy which yields the maximum accuracy values can be seen at the beginning of each plot. The highest accuracies obtained for the different settings are 89% for *t15* (86% prediction rate), 84% for *t8* (59% prediction rate), 64% for *t4* (34% prediction rate), and 35% for *t2* (10% prediction rate). These maximum accuracy values are usually achieved with a threshold of 1 or 2 for the confidence filter. It seems that restricting the number of allowed zero-distance-elements in the set of the nearest neighbors to 0 is counterproductive since it decreases the prediction rate without increasing the accuracy.

Finally, to investigate which genres are likely to be confused with others, we calculated a confusion matrix, cf. Figure 4. It can be seen that the genres “Jazz”, “Blues”, “Reggae”, and “Classical” are perfectly distinguished. “Heavy Metal/Hard Rock”, “Electronica”, and “Rock ’n’ Roll” also show very high accuracies of about 95%. For “Country”, “Folk”, “RnB/Soul”, “Punk”, “Rap”, and “Pop” , accuracies between 83% and 89% are achieved. In comparison with the results of [6], where “Pop” achieved only 80%, we reach 88% for this genre. In contrast, our results for the genre “Alternative Rock/Indie” are very bad (≈50%). A more precise analysis reveals that this genre is often confused with “Electronica”, which may be explained by some artists producing music of different styles (over time), like “Depeche Mode” in “Alternative Rock/Indie” or “Moby” and “Massive Attack” in “Electronica”. “Depeche Mode”, for example, was a pioneer of “Synthesizer-Pop” in the 1980s.

5. CONCLUSIONS

In this paper, we presented an artist similarity measure based on co-occurrences of artist names on web pages. We used three different query settings (*M*, *MR*, and *allintitle*) to retrieve page counts from the search engine Google. Experiments showed that the *allintitle*-setting provides high accuracies with k-Nearest Neighbors classification. High prediction rates, however, are achieved with the *M*-setting. In order to exploit the advantages of both settings, the two measures were combined using a simple threshold-based confidence filter. We showed that this combination gives accuracies of up to 85% at 100% prediction rate (no unclassified artists). These results are at least equal to those presented in [6] when using a sufficient number of training samples from each genre. In [6], however, a much more complex approach, $tf \cdot idf$, is used. For scenarios with only very few artists available to define a genre, the $tf \cdot idf$ approach performs better due to its extensive use of additional information. In contrast, less information is used in the approach presented in [15]. Our approach differs from that of

Zadel and Fujinaga, among other things, in that they use a symmetric similarity measure and a different normalization method. As a result, their approach performs slightly worse than ours.

Further research may focus on the combination of web-based and signal-based data to raise the performance of similarity measures or to enrich signal-based approaches with cultural metadata from the Internet. Since the data set used for evaluation contains quite general genres and well-known artists, it would be interesting to test our approach on a more specific data set with a more fine-grained genre taxonomy. Finally, heuristics that reduce the computational complexity of our approach should be tested. This would enable us to process also large artist lists.

6. ACKNOWLEDGMENTS

This research is supported by the Austrian Fonds zur Förderung der Wissenschaftlichen Forschung (FWF) under project numbers L112-N04 and Y99-START, and by the EU 6th FP project SIMAC (project number 507142). The Austrian Research Institute for Artificial Intelligence is supported by the Austrian Federal Ministry for Education, Science, and Culture and by the Austrian Federal Ministry for Transport, Innovation, and Technology.

7. REFERENCES

[1] J.-J. Aucouturier and F. Pachet. Finding songs that sound the same. In *Proceedings of the IEEE Benelux Workshop on Model based Processing and Coding of Audio*. University of Leuven, Belgium, November 2002. Invited Talk.

[2] J.-J. Aucouturier and F. Pachet. Music Similarity Measures: What's the Use? In *Proceedings of the 3rd International Symposium on Music Information Retrieval (ISMIR'02)*, Paris, France, October 2002.

[3] A. Berenzweig, B. Logan, D. P. W. Ellis, and B. Whitman. A Large-Scale Evaluation of Acoustic and Subjective Music Similarity Measures. In *Proceedings of the 4th International Symposium on Music Information Retrieval (ISMIR'03)*, Washington, D.C., USA, 2003.

[4] P. Cano and M. Koppenberger. The Emergence of Complex Network Patterns in Music Artist Networks. In *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR'04)*, Barcelona, Spain, October 2004.

[5] D. P. W. Ellis, B. Withman, A. Berenzweig, and S. Lawrence. The Quest for Ground Truth in Musical Artist Similarity. In *Proceedings of the 3rd International Symposium on Music Information Retrieval (ISMIR'02)*, Paris, France, 2002.

[6] P. Knees, E. Pampalk, and G. Widmer. Artist Classification with Web-based Data. In *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR'04)*, pages 517–524, Barcelona, Spain, October 2004.

[7] B. Logan and A. Salomon. A Music Similarity Function Based on Signal Analysis. In *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME'01)*, Tokyo, Japan, 2001. Institute of Electrical and Electronics Engineers.

[8] F. Pachet, G. Westerman, and D. Laigre. Musical Data Mining for Electronic Music Distribution. In *Proceedings of the 1st WedelMusic Conference*, 2001.

[9] E. Pampalk. Islands of Music: Analysis, Organization, and Visualization of Music Archives. Master's thesis, Vienna University of Technology, Austria, December 2001.

[10] E. Pampalk, S. Dixon, and G. Widmer. Exploring Music Collections by Browsing Different Views. In *Proceedings of the 4th International Conference on Music Information Retrieval (ISMIR'03)*, Washington, D.C., USA, October 2003.

[11] E. Pampalk, S. Dixon, and G. Widmer. On the Evaluation of Perceptual Similarity Measures for Music. In *Proceedings of the 6th International Conference on Digital Audio Effects (DAFx'03)*, volume 8, London, UK, September 2003.

[12] E. Pampalk, A. Rauber, and D. Merkl. Content-based Organization and Visualization of Music Archives. In *Proceedings of the Association for Computing Machinery (ACM) Multimedia*, pages 570–579, Juan les Pins, France, December 2002. ACM.

[13] M. Schedl. An Explorative, Hierarchical User Interface to Structured Music Repositories. Master's thesis, Vienna University of Technology, Austria, December 2003.

[14] B. Whitman and S. Lawrence. Inferring Descriptions and Similarity for Music from Community Metadata. In *Proceedings of the 2002 International Computer Music Conference*, pages 591–598, Goeteborg, Sweden, September 2002.

[15] M. Zadel and I. Fujinaga. Web Services for Music Information Retrieval. In *Proceedings of the 5th International Symposium on Music Information Retrieval (ISMIR'04)*, Barcelona, Spain, October 2004.