A Novel Genre-Specific Feature Reduction Technique through Association Analysis

Adam Lefaivre
University of Lethbridge
Lethbridge, Canada
lefaivreadam@gmail.com

John Zhang
University of Lethbridge
Lethbridge, Canada
john.zhang@uleth.ca

***Abstract*—We consider the genre classification problem in Music Information Retrieval and report our initial investigation on reducing the number of features that are used in genre classification. Each music genre has its own characteristics, which distinguish it from other genres. We adapt association analysis to capture those characteristics using acoustic features, i.e., each genre's characteristics are represented by a set of features and their corresponding values. Our goal is to select the "most representative" features for each genre. Such features are unique in distinguishing a genre and therefore should be singled out. We propose two criteria for comparing and selecting those unique features of each genre. The details of our proposed approach are presented. The effectiveness of our approach is demonstrated and discussed through empirical experiments.**

I. Introduction

In recent decades, large volumes of music have been digitized and are now available in online collections and streaming services. However, managing them by hand is extremely time-consuming and error-prone. An emerging area, *Music Information Retrieval* (*MIR*) calls for the necessity of computational approaches to various core managing tasks involving music data, such as *instrument recognition*, *genre classification*, etc. [1].

Quite often many operations in MIR need to categorize music pieces before any further steps can be conducted. For instance, in *playlist recommendation*, we may need to know the genre information of a music piece before a recommendation can be made on it. While there are many criteria that people use to categorize music, such as mood, instrumentation, etc., genres are still one of the most widely used ways to classify music. If a genre classification method is effective enough, it will assist in music data management, e.g., archive querying. Musical genres, however, are notoriously subjective and ambiguous, with low human classification rates [1]. When applying computational classification techniques to music data, one faces an unavoidable challenge – the large number of acoustic features. In a classification process, not only does such a challenge entail a heavy computational cost, it also introduces further confusion and ambiguity, ultimately reducing classification accuracy.

In the sequel, we present our initial efforts to tackle feature reduction in genre classification. Through association analysis, we generate those ”characteristic” content-based acoustic features of individual genres, which are then further refined to select the "most" representative ones. We propose different selection criteria toward this end.

II. Related Works

Some initial attempts in music genre classification are carried out by Tzanetakis and Cook [2], who propose content-based genre classification and a "standard" set of features for doing it. They also provide their own data for genre classifications, which is now known as the GTZAN dataset (http://marsyas.info/downloads/datasets.html). Some recent discussions are given by Ajoodha et al. [3], where magnitude, tempo, and pitch-based features, and several off-the-shelf classifiers are used, and a relationship between the number of features and classification improvements is shown. Medhat et al. [4] classify the ballroom music dataset with eight (8) genres, resulting in the best accuracy of 92.12%.

Association analysis [5] is presented in an unrelated context to MIR, among others, by Liu et al. [6]. In MIR, It is proposed as a way to retrieve genre specific music files by Rompré et al. [7]. Shan et al. [8] attempt to classify musical styles by using association analysis to construct a classifier through melody mining. Zhang and Arjannikov [9] report an empirical study on different nested dichotomies (binary-based classifiers). They show that balanced nested dichotomies often perform better than unnested ones. However, many of the ensemble classifiers they tested do not perform better than the individual base classifiers. The work in [10] shows improvements in classification accuracy when using feature selection via the wrapper approach and classifier ensembles. Lefaivre and Zhang [11] use association analysis among feature-value pairs over individual genres to distinguish and select, for each genre, its characteristic feature-value pairs. This helps improve the classification accuracy. In [12], a similar approach based on association analysis is employed to conduct subgenre classifications, a more subtle and ambiguous task in MIR.

To the best of our knowledge, currently there are only a few works on feature reduction in music genre classification. In [13], *Gain Ratio* is proposed as a feature selection measure, which is used to select a splitting attribute that will separate the music pieces into different classes. The splitting criterion is employed to identify the class or the music genre. The experimental results indicate a satisfactory result for music genre classification. In another work [14], four diverse audio feature sets, i.e., dynamic, rhythm, spectral, and harmony, have been proposed to characterize the music contents precisely. From the features, five different statistical parameters are considered as representatives, including up to the 4-th order central moments of each feature, and covariance components. The score levels of all feature attributes are calculated and

ISSN 2305-7254

ordered. The high score feature attributes are only considered for genre classification. The approach proposed in this paper is different. We start from the characteristic features already obtained for individual genres and then examine and single out those truly representative ones.

III. Problem Statement and Approach

A. Problem Statement

It is known that in music repositories, the acoustic musical contents from genres contain rich information that makes them distinguishable from each other. Such information can be extracted and utilized. A problem faced by the MIR community is that there can be a huge number of features that could be used in classifying music. But using them without any discrimination not only incurs heavy computational cost but may also introduce ambiguity and confusion in the later classification process. In this work, we present a systematic approach to feature reduction in music genre classification. We start from characterizing each music genre by a set of acoustic features through association analysis and then we examine feature-value pairs across those genres in a combined manner. We propose different criteria to compare and choose which features are the most "representative" for individual genres.

The goal of our work in this paper is two-fold. (1) We expect to see an increase in genre classification accuracy in a "low-quality" music dataset if our proposed approach is applied to it; (2) For a "high-quality" music dataset, our approach may remove features but such a removal will not affect the final classification accuracy, i.e., our approach is stable.

Association analysis is proposed by Agrawal et al. [5]. In a problem domain, a set of data items that "frequently" occur together shows some statistical relationship. Those frequent items are put into *frequent itemsets*, e.g., a frequent 3-itemset means the three items in the set occur together frequently. The *support* of an itemset is the percentage of the co-occurrence of the items in it. Only the itemsets whose support exceeds a *minimum support*, m_s, are frequent. We adapt the *Apriori* [5] association algorithm in our work

B. Our Proposed Approach

For music genre classification, each piece in a music dataset is represented as a vector $P = \{p_1, p_2, \cdots, p_n\}$, where p_i is the value of the feature $f_i \in F$ and $F = \{f_1, f_2, \cdots, f_n\}$ is the acoustic feature set selected for classification. The values of those features are extracted using some software frameworks, e.g., *Marsyas* [15]. Given a music dataset, our proposed approach is to reduce the number of features needed in the genre classification process, towards the goal to lessen the computational cost and increase the classification accuracy.

We start out our approach by first making use of association analysis to find a list of characteristic feature-value pairs for an individual music genre. Algorithm A1 is fully presented in [12] and we adapt it in our proposed approach. m_s is the minimum support needed.

The superscripts tr and te correspond to training and testing. Given a dataset, suppose that it has n genres and each genre G has a set of pieces labelled G. A set GS is randomly chosen from it, to balance the number of music pieces in each genre. Since Apriori handles discrete values, we discretize acoustic features' real values, using a binning method, and then normalize them. During this process, we encode each value for each feature systematically. After encoding, each piece is represented as a set of feature-value pairs, called an *fv-set*. For each frequent fv-set returned by Apriori, we set the number of feature-value pairs in it, to be at least 2. For the genre G, after this step, we obtain its M sets of frequent fv-sets, denoted as GS_i^F, where $i = 1, \cdots$, $i = M$ ($M = 10$ in our experiments), from which we produce a more representative characteristic set, called GS^C. Intuitively speaking, such a characteristic set represents a genre from a viewpoint based on acoustic features and tells us why a genre is distinct from another computationally. We will further refine and distill these characteristic sets to find those truly representative features.

Algorithm A1: Characterizing music genres by feature-value pairs
1. Bin all music pieces using Equal Frequency Binning
2. For each genre G's dataset GS
3. Split GS into training set GS^{tr} and testing set GS^{te}
4. For each G's GS^{tr}
5. Randomly generate M subsets of it (denoted as GS_i)
6. For each genre G and for each of its GS_i
7. GS_i^F = Call *Apriori* to GS_i with m_s
8. For each genre G
9. We append fv-sets from GS_i^F to GS^C and remove any duplicates

Function δ-Removal: Feature Reduction using δ
1. $reduce\_char\_sets(GS_1^C, ..., GS_i^C, ..., GS_n^C, \delta)$:
2. For each feature $f_j \in F$
3. For each characteristic set GS_i^C
4. if $c_i^{f_j} / c_{G*}^{f_j} < \delta$
5. Remove f_j from all genres characteristic sets ($R_{\delta,1}$)
6. Or Remove f_j from GS_i^C $R_{\delta,2}$

In this initial work, we introduce two criteria in order to remove further non-essential and/or redundant features from the characteristic set of genres. They are described as follows.

The first one is called δ-removal. We introduce a parameter δ (between 0 and 1) which is used to reduce the characteristic sets of each genre even further. With this parameter we are able to identify features that do not have a significant impact across all genres. We determine the number of times a feature $f_j \in F$ occurs in a characteristic set of a particular genre GS_i^C by the notation $c_i^{f_j}$, and the number of times it occurs in the characteristic sets of all genres as $c_{G*}^{f_j}$, where $G*$ is the set of all genres. There are two variations of δ-removal. We denote the first variation as $R_{\delta,a}$, which specifies that if $c_i^{f_j} / c_{G*}^{f_j} < \delta$ then this feature is removed from all genres' characteristic sets. The second variation is denoted as $R_{\delta,i}$, which is to remove the feature from the characteristic set of the individual genre under examination. The first variation represents our view that if a feature appears only few times in any genre, we suspect that it will not contribute to classification of any genres and therefore should be removed from the characteristic sets of all the genres. On the other hand, the second variation represents our view that if a feature does not appear enough times in a particular genre, we suspect that it will not contribute to classifying music pieces of that genre and is removed from the characteristic set of that particular genre. Which one is better? Well it all depends the user's view of a feature that does not show up often in characteristic sets of genres and the datasets we are working on. These two criteria are summarized in Function $\delta - Removal$.

With the characteristic sets of individual genres ready, we can classify an unseen music piece for its genre from the testing music dataset, which is represented as a vector of feature-value pairs and scored against a pair of genres by comparing the differences between their respective characteristic sets. For a new music piece P from a subset of GS^{te} we maintain a score vector $(S_{G_1}, S_{G_2}, \cdots, S_{G_n})$, where S_{G_i} is the "score" of G_i for P. Algorithm A2 is adapted from [11], which is to classify a new music piece, after the refined characteristic sets have been obtained for individual genres.

Algorithm A2: Evaluating pairwise music genres by feature-value pairs

```
For a new music piece P (represented by feature-value pairs)
    For the characteristic sets of two genres G_i and G_j, GS_i^C and GS_j^C
        Call function reduce_char_sets
        Calculate the except difference of GS_i^C and GS_j^C using φ
            i.e., DC_ij = GS_i^C − GS_j^C and DC_ji = GS_j^C − GS_i^C.
        Score on P using DC_ij and DC_ji
            s_i = Counting(P, DC_ij)
            s_j = Counting(P, DC_ji)
            if s_i > s_j then
                S_GS_i += 1
            else
                S_GS_j += 1
    Set the genre of the highest score to be the one for P.
```

When classifying new pieces into their genres, there are musical and acoustic elements that are common to all genres, which cause confusions. We use a parameter ϕ to conduct a "fuzzy" check for whether an fv-set from GS_i^C appears in GS_j^C, and vice versa. In this sense ϕ is a strictness factor. For instance, if $\phi = 60\%$, then {b4, c3} matches with {a2, b4, c3} but not with {a2, b4, c2}. The except difference between the two characteristic sets, $GS_i^C - GS_j^C$, consists of those fv-sets that are present in GS_i^C but not in GS_j^C. We precalculate the except differences among all pairs of genres.

The procedure $Counting$ counts how many fv-sets in the except difference are in the subset of P's feature-value vector. We add 1 to to a genre's score, if the corresponding $Counting$ procedure returns a higher value.

IV. Empirical Experiments

A. Experiment Setup

We apply our approach on two popular datasets in the MIR literature, namely the *Latin Music Database* (D_{LMD}) [16], and the *GTZAN dataset* (D_{GTZ}). D_{GTZ} is one of the earlier datasets that is widely used in music genre classification. D_{GTZ} has a set of standard deviations extracted using JAudio [17]. Its features include 13 MFCCs, spectral centroid, spectral flux, zero crossings, strongest beat overall, beat sum overall, strength of strongest beat, strongest frequency via zero crossings, and method of moments. The sampling rate is 22050Hz, with a window size of 2048ms, and a hopsize of 1024ms; standard deviations and averages are calculated. D_{GTZ} has 100 music pieces per genre, 30 seconds each, with 10 genres, and a total of 76 features extracted. For the dataset D_{LMD}, its features are those found by Silla et al. [18], as accessed online (https://sites.google.com/site/carlossillajr/resources/the-latin-music-database-lmd.). These features are extracted using the MARSYAS [15] framework and include 5 MFCCs, spectral centroid, rolloff, and flux, zero crossings, low energy, relative amplitudes, beats per minute, maximum periods of pitch peak. Mean and variances are calculated for each. D_{LMD} has approximately 3000 music pieces per genre with 10 genres. We use only the middle 30 second segment of each music piece, with a total of 30 features extracted. Since D_{LMD}'s features are derived already, its labelling is done by experts, and there are more pieces, it is regarded as a higher-quality dataset. Though D_{GTZ} and D_{LMD} are both used widely in the MIR community, it is found in our experience the latter one is of higher quality than the former one.

Each dataset is preprocessed by selecting the same number of pieces per genre. In our experiments we only use one binning method to preprocess numeric features, which is equal frequency based binning (denoted as B_{ef}), it is selected due to its better overall performance in previous association analysis tasks [11], [12], [19]. We also fix ϕ to be 0.4, as we found in previous classification tasks that a stricter ϕ value yielded more characteristic representations of each genre [11], [12]. We use an 80%20% split for training and testing, with M=10.

B. Experiment Results

1) Removal method $R_{\delta,a}$: We experiment first with $R_{\delta,a}$ on D_{GTZ} and D_{LMD} on three different minimum supports (m_s), which are used in Algorithm A1 and specify why a feature is considered useful or not for a genre. The results are shown in Table I for dataset D_{GTZ}.

As can be seen from the table, we have used varying δ thresholds in our experiments. After removing features, we can see that the classification accuracies increase. We compare this increase to the first row of Table I ($\delta = 0$) which represents no feature reduction being done. The increase is slight, however, even with a larger number of features removed a similar accuracy is maintained. With some features removed we still see that the accuracy increases and this occurs with all three different supports.

On the other hand, the results from dataset D_{LMD}, shown in Table II, are more interesting. As mentioned before D_{LMD} is of higher-quality in terms of the genre classification task. This is confirmed in our experiments. As we increase δ, more features are removed. With more features removed, we see a decrease of classification accuracy and this happens across all the parameters we set up. One reason that we can explain this is that there are already a smaller number of features and after more descriptive features are removed, when we attempt to classify a new music piece, we just do not have enough features that enable a particular genre to stand out. Therefore the classification process is confused, resulting in lower classification accuracies.

We hope that our feature reduction method will remove features that are more correlated on average to other features, and are therefore less useful. To do this we examine the average of the absolute Pearson correlation coefficients [20] for each feature to every other feature. We take the ratio of the average absolute coefficients in the subset of features removed to all other features against the average of the absolute

TABLE I. RESULTS ON D_{GTZ} - $R_{\delta,a}$

$m_s = 6$				$m_s = 7$				$m_s = 8$			
δ	Avg. Accuracy	# of Features Removed	Corr. Analysis	δ	Avg. Accuracy	# of Features Removed	Corr. Analysis	δ	Avg. Accuracy	# of Features Removed	Corr. Analysis
0	64.5	0	-	0	60.8	0	-	0	62.3	0	-
0.0065	65.0	1	1.144	0.0025	60.4	1	0.813	0.0003	62.4	5	0.198
0.0075	64.4	2	0.979	0.0030	60.8	4	0.954	0.0004	62.8	6	0.283
0.0085	64.6	3	1.086	0.0035	61.5	6	1.047	0.0005	62.4	7	0.359
0.0095	64.0	5	1.204	0.0040	60.5	9	1.149	0.0006	63.8	8	0.477
0.0105	61.4	7	1.246	0.0045	58.5	16	1.160	0.0007	63.8	8	0.477

TABLE II. RESULTS ON D_{LMD} - $R_{\delta,a}$

$m_s = 2$				$m_s = 3$				$m_s = 4$			
δ	Avg. Accuracy	# of Features Removed	Corr. Analysis	δ	Avg. Accuracy	# of Features Removed	Corr. Analysis	δ	Avg. Accuracy	# of Features Removed	Corr. Analysis
0	71.8	0	-	0	70.0	0	-	0	68.3	0	-
0.003	71.3	1	1.026	0.0011	68.9	1	1.026	0.0005	67.2	1	1.026
0.004	70.6	4	1.094	0.0016	68.9	2	0.982	0.0010	67.1	2	0.982
0.005	69.2	7	1.074	0.0021	68.4	4	0.988	0.0015	66.5	5	1.028
0.006	67.4	10	1.138	0.0026	67.9	5	1.028	0.0020	64.2	8	1.069
0.007	67.4	10	1.138	0.0031	67.7	9	1.121	0.0025	61.4	11	1.131

coefficients for all features. This ratio converges to 1 with greater δ values. For this analysis, if the ratio is greater than 1 we remove more correlated features than the average. We find that the majority of experiments in Tables I and II provide the removal of more correlated features, and that across all experiments further removal of features using our method removes more correlated features. So, while the performances are not necessarily competitive with the absolute best classifications on these datasets we find that our feature reduction technique might be very useful for removing redundant features.

2) Removal method $R_{\delta,i}$: With $R_{\delta,i}$, if a feature is deemed below δ for a genre, we just remove it from the characteristic set of that genre. Now let us take a look at Tables III and IV, in which Max and Avg. # denote the maximum and average number of features removed, respectively. It is very surprising to see that after removing several features, the classification accuracy is similar and in some cases even better. We find the highest accuracy on D_{GTZ} across all experiments when $\delta = 0.020$. This threshold removes 29 features in one genre and removes an average of 9.2 features across all genres. An improvement in accuracy with this many features removed demonstrates the value of this approach. What is more interesting is that with D_{GTZ} we can even have a higher m_s (previously known to provide poor performance [11], [12], [19]) with an effective δ value yielding similar accuracies as a lower support without δ. This makes our feature reduction approach useful, because computational cost is saved by producing smaller but more effective characteristic sets. In Table IV we again see a steady decrease in performance with each increasing δ value, we believe this is because the feature space is already smaller, so any features that are removed will have a greater impact in the classification tasks. One impressive result in Table IV however, is that with the least strict δ we are actually able to achieve a higher accuracy than the original accuracy for two supports (2% and 3% for δ of 0.003 and 0.0012 respectively), and with other δ values we at least match the original accuracy, which confirms the effectiveness of our approach.

As before, we provide another correlation analysis on $R_{\delta,i}$. This is found in a similar way; we calculate the average correlation of the set of features removed across all genres. In general the average correlation of the features removed is greater than the average correlation of all the features. Meaning that we are removing more redundant features on average.

C. Discussions

Our feature reduction strategy is flexible enough to remove any number of features, and in our experiments we test δ at discrete thresholds of increasing strictness, starting with a small enough threshold to remove the first feature. In some other $R_{\delta,i}$ experiments we have found that, regardless of the δ value supplied by the user, a similar number of features are removed at various thresholds. This occurs when a certain set of features are not representative of any particular genre but the rest are. For all experiments we notice that features removed at a less strict threshold are also removed at a stricter threshold. For the genre-specific removal we are also removing features that represent the genres in a "biased" manner, since a feature j could have a very low $c_i^{f_j}$ count for genre G_i and a high $c_{i+1}^{f_j}$ count for genre G_{i+1} and still be removed. In this way the parameter may handle an issue such as model overfitting by removing features known to skew genre classification experiments. The effectiveness of this observation needs to be confirmed through more experiments.

Our proposed method of feature reduction provides similar accuracies when compared with the classification done with no features removed. For example, even after 10 features are removed for D_{LMD}, we are still able to achieve a similar performance. For the results on D_{GTZ}, an improvement in performance with a reduced number of features across all supports and in both removal experiments is noticed. This means that we are able to find, for a dataset that is not of high-quality, even more representative characteristic sets after feature reduction is performed. In the experiments conducted, we see that a lower m_s is able to provide a higher accuracy. We believe this is due to an accumulation of more descriptive fv-sets per genre. We also find that the δ parameter provides plenty of flexibility in determining the number of features removed.

Finally not only are we able to remove biased features per genre using the δ threshold, but we have shown that in most cases, when removing initial portions of the overall feature set we are also removing more correlated features. The

TABLE III. RESULTS ON D_{GTZ} - $R_{\delta,i}$

$m_s = 6$					$m_s = 7$					$m_s = 8$				
δ	Avg. Accuracy	Max #	Avg. #	Corr. Analysis	δ	Avg. Accuracy	Max #	Avg. #	Corr. Analysis	δ	Avg. Accuracy.	Max #	Avg. #	Corr. Analysis
0.008	64.5	1	0.2	0.978	0.0025	60.8	1	0.1	0.814	0.0002	62.3	3	0.5	0.198
0.009	63.1	3	0.5	1.152	0.0030	59.5	3	0.4	0.954	0.0004	62.3	3	0.6	0.283
0.010	62.8	3	0.7	1.204	0.0035	59.9	4	0.7	1.000	0.0006	60.6	4	0.8	0.477
0.011	62.1	6	1.4	1.276	0.0040	59.9	7	1.2	1.047	0.0008	61.6	6	1.1	0.594
0.012	62.1	8	2.1	1.250	0.0045	59.9	10	2.1	1.149	0.0010	61.6	7	1.2	0.594
0.016	62.6	22	5.3	1.199	0.0065	62.7	22	4.9	1.146	0.0015	63.4	12	2.6	0.840
0.020	65.6	29	9.2	1.231	0.0085	62.8	29	7.7	1.190	0.0020	64.0	23	4.2	0.975
0.024	63.9	42	14.2	1.234	0.0105	58.1	38	10.8	1.139	0.0025	64.1	25	5.2	0.992

TABLE IV. RESULTS ON D_{LMD} - $R_{\delta,i}$

$m_s = 2$					$m_s = 3$					$m_s = 4$				
δ	Avg. Accuracy	Max #	Avg. #	Corr. Analysis	δ	Avg. Accuracy	Max #	Avg. #	Corr. Analysis	δ	Avg. Accuracy.	Max #	Avg. #	Corr. Analysis
0.003	71.9	1	0.3	1.026	0.0012	70.7	1	0.2	1.026	0.0005	68.2	1	0.1	1.026
0.004	71.5	3	0.8	1.094	0.0017	70.0	2	0.5	0.982	0.0010	68.0	2	0.4	0.982
0.005	70.3	6	1.3	1.074	0.0022	70.0	4	0.9	0.986	0.0015	67.9	4	1.1	1.028
0.006	67.9	10	2.7	1.138	0.0027	69.0	4	1.2	1.028	0.0020	66.2	7	2.3	1.068
0.007	67.0	10	3.5	1.138	0.0032	67.6	9	2.4	1.138	0.0025	62.6	10	3.6	1.131
0.008	66.1	12	4.5	1.117	0.0052	63.3	14	4.9	1.089	0.0035	60.6	13	5.5	1.225
0.010	62.7	17	6.9	1.113	0.0062	60.9	16	6.3	1.097	0.0040	59.1	16	6.4	1.095
0.012	59.2	20	8.8	1.043	0.0072	60.5	18	7.2	1.116	0.0045	59.9	18	7.2	1.081

correlation value seems to increase for the first number of features removed for all experiments and in most cases, stays above 1.

V. FUTURE WORK

The characteristic feature sets of genres contain rich information that we can mine and refine. Currently, we are thinking of utilizing the variance of a feature in terms of the number of occurrences it appears in different genres. The aim is to remove a feature whose variance is too high relative to the characteristic sets of all genres. If the number of times a feature appears in different genres is drastic, we do not trust it any more, as the classification of a new music piece may be too dependent on this feature, and therefore confuse a classification algorithm. Another similar direction is to remove a feature whose variance is too low. If a feature appears almost the same number of times in all genres, we believe that its contribution in genre classification is not so important and can be neglected.

Furthermore, since we accumulate all of the characteristic sets of each genre, we can calculate how many times a feature is used in the classification of a new piece. This ascribes some importance to that feature. We can then use this to either reduce features or find the "importance" of any feature used.

We will do more experiments on other off-the-shelf datasets to examine the effectiveness of our approach. We also plan to create a dataset that initially contains all of the possible acoustic features, apply our approach to it, and see what features will be removed and what will happen eventually.

REFERENCES

[1] T. Li, M. Ogihara, and G. Tzanetakis, *Music Data Mining*. Belmont, CA, USA: CRC Press Wadsworth, 2011.

[2] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, 2002.

[3] R. Ajoodha, R. Klein, and B. Rosman, "Single-labelled music genre classification using content-based features," in *Proceedings Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference*, Nov 2015, pp. 66–71.

[4] F. Medhat, D. Chesmore, and J. Robinson, "Automatic classification of music genre using masked conditional neural networks," in *Proceedings IEEE International Conference on Data Mining*, Nov 2017, pp. 979–984.

[5] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," *SIGMOD Record*, vol. 22, no. 2, pp. 207–216, 1993.

[6] B. Liu, W. Hsu, and Y. Ma, "Integrating classification and association rule mining," in *Proceedings International Conference on Knowledge Discovery and Data Mining*, 1998, pp. 80–86.

[7] L. Rompré, I. Biskri, and J. Meunier, "Using association rules mining for retrieving genre-specific music files," in *Proceedings International Flairs Conference*, 2017, pp. 706–711.

[8] M.-K. Shan, F.-F. Kuo, and M.-F. Chen, "Music style mining and classification by melody," in *Proceedings International Conference on Multimedia and Expo*, vol. 1, 2002, pp. 97–100.

[9] T. Arjannikov and J. Zhang, "An empirical study on structured dichotomies in music genre classification," in *Proceedings International Conference on Machine Learning and Applications*, 2015, pp. 493–496.

[10] D. Chathuranga and L. Jayaratne, "Musical genre classification using ensemble of classifiers," in *Proceedings International Conference on Computational Intelligence, Modelling and Simulation*, Sept 2012, pp. 237–242.

[11] A. Lefaivre and J. Z. Zhang, "Music genre classification: Genre-specific characterization and pairwise evaluation," in *Proceedings of the Audio Mostly 2018 on Sound in Immersion and Emotion*, 2018, pp. 22:1–22:4.

[12] A. Lefaivre and J. Zhang, "Characterizing and classifying music subgenres," in *Proceedings of the 23rd Conference of Open Innovations Association FRUCT*, 2018.

[13] J. Mitra and D. Saha, "An efficient feature selection in classification of audio files," *Computer Science Information Technology*, vol. 4, 03 2014.

[14] B. K. Baniya, J. Lee, and Z. Li, "Audio feature reduction and analysis for automatic music genre classification," in *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2014, pp. 457–462.

[15] G. Tzanetakis. (2017, October) Music analysis, retrieval and synthesis for audio signals. [Online]. Available: http://marsyas.info/index.html

[16] C. N. Silla Jr., A. L. Koerich, and C. A. A. Kaestner, "The latin music database," in *Proceedings of the 9th International Conference on Music Information Retrieval*, Philadelphia, USA, 2008, pp. 451–456.

[17] D. McEnnis, I. Fujinaga, C. McKay, and P. Depalle, "Jaudio: A feature extraction library," in *Proceedings the International Society of Music Information Retrieval*, 2005, pp. 600–603.

[18] C. N. Silla, A. L. Koerich, and C. A. A. Kaestner, “Automatic music genre classification using ensemble of classifiers,” in *Proceedings IEEE International Conference on Systems, Man and Cybernetics*, 2007, pp. 1687–1692.

[19] A. Lefaivre and J. Z. Zhang, “Capturing genre characteristics of music through feature-value analysis,” in *IEEE 5th International Conference on Systems and Informatics*, 2018, pp. 1039–1044.

[20] W. Kirch, Ed., *Pearson’s Correlation Coefficient*. Springer Netherlands, 2008.