Intelligent Classification of Electronic Music

Siva Sankalp P., Trinayan Baruah, Shlesh Tiwari, Sankar Ganesh S.
School of Electronics Engineering (SENSE)
VIT University, Vellore, India

Abstract— **Electronic Dance Music (EDM) is a class of music produced primarily using electronic instruments and computers comprising several sub-genres. Disc jockeys (DJs) mix EDM songs from similar sub-genres for live performances in concerts and festivals. Given a large database of electronic music, if a system can classify songs into their respective sub-genres, it can greatly help producers and DJs while making mixes. We describe an approach to classify four major sub-genres of EDM using artificial neural networks. We extracted several rhythmic and statistical features from a database of 400 songs, evenly distributed among the four sub-genres, and used these features to train a Back Propagation Network (BPN) and a Probabilistic Neural Network (PNN). In addition to classification, we also employed the RELIEFF feature selection algorithm to reduce the number of features and improve our results. Comparing the performances of the two networks on the basis of their correct classifications, we observed that BPN combined with the features proposed yielded a better result of 94.16%.**

Keywords- Electronic Dance Music; neural networks; genre classification, Probabilistic Neural Network; Back Propagation Network

I. INTRODUCTION

In the world of music today, Electronic Dance Music - colloquially called EDM - has been constantly gaining prominence over the past decade due to the growing number of producers and ever-evolving Digital Audio Workstations (DAWs), the software widely used by music producers to record, edit, and produce music. Considering this, it is no surprise that there is an increase in the number of Electronica DJ performances all over the world. EDM originated as an underground culture, and was not quite heard of until recently. Following this increased interest in EDM, several sub-genres have evolved over times that include Dubstep, Techno, Electro-house, Trance, etc. Before a live performance, a DJ compiles a playlist of song which involves mixing two or more songs. The DJ must make sure that the songs go well together while mixing which can be ensured by selecting songs from the same or a matching sub-genre. The above mentioned task of creating a playlist with similar songs can be simplified with the help of artificial neural networks. This is the primary goal of our work.

The next section gives an overview of related work on genre classification in electronic music based on audio content analysis. The remainder of the paper deals with sub-genre description and song collection, feature extraction, and classification. We explore the use of artificial neural networks for the classification of electronic dance music. This is followed by comparison and evaluation of the results. We finally conclude with a brief discussion of applications and future work.

II. RELATED WORK

In the relatively new field of Music Information Retrieval, various systems have been developed that classify music according to a certain type of similarity. A majority of research work done in this field focuses on classification of broader genres [1, 2, 3], but not the multitude of sub-genres contained within them. This number is large when it comes to the domain of electronic music, and this calls for a necessary sub-division within the genre. Initially, most of the genre classification approaches relied on metadata [4, 5], and not on the actual audio. As a consequence, complete musical records were required to produce results which limited the applicability of the approach. This called for an improved method which would take into account the actual content of the song (content-based analysis).

There is a considerable amount of literature available on content-based analysis of music, and this has invited attention in the form of various content-based researches on genres like rock, pop, jazz, etc. Electronic music, however, has not been the host to comparable research. One of the most notable works on electronic music classification can be seen in Diakopoulos et al [6] where the authors have used spectral, loudness, and temporal features to classify electronica. In another work [7], timbre and rhythmic similarities are investigated to find similar segments in electronic dance music. Kirss [8] presents a comparison of machine learning algorithms in classifying electronic music.

III. SONG COLLECTION AND GENRE DESCRIPTION

For our purpose, we selected four sub-genres based on varied rhythmic characteristics. We collected 100 songs from each of the following genres- Trance, Electro-house, Dubstep, and Techno making it a total of 400 songs. The actual genre in which these songs belong to was decided on the basis of self-perception and also their classification in online libraries like last.fm[1] and beatport[2]. The size of the dataset was judged according to previous works on music classification. From each of these songs a standard audio clipping tool was used to extract segments of around 22 seconds which gives the most representative part of the song. Apart from the extraction, the

[1] Available. [Online]: http://www.last.fm/

[2] Available. [Online]: http://www.beatport.com/

978-1-4799-1812-6/14/$31.00 ©2014 IEEE

songs were converted to WAV format which is the standard format for reading files in MATLAB. Our dataset consisted of songs from at least 40 different contributing artists, diversifying the range over which classification could be effectively done. The selection criteria included general aural response along with distinguishing rhythmic properties.

Our selection of songs for this paper includes four of the most prominently followed and performed sub-genres in Electronic Dance Music [9]: Dubstep, Trance, Electro-house, and Techno. The following gives a brief overview into each of the selected sub-genres.

A. Dubstep

Dubstep makes use of overwhelming bass lines and reverberant drum patterns, clipped samples, and occasional vocals. The tempo is nearly always in the range of 138-142 beats per minute, with a clap or snare inserted every third beat in a bar. Notable artists include *Skrillex*, *Skream*.

B. Trance

Trance is characterized by a tempo of between 130 and 160 bpm, featuring repeating melodic synthesizer phrases, and a musical form that builds up and down throughout the track. It often features crescendos and breakdowns, and sometimes vocals. Notable artists include *Armin van Buuren*, *Ferry Corsten*.

C. Electro-house

Electro-house typically uses a “four to the floor” beat, often a low bass, and the use of recognizable instrumentation like piano chords and soulful vocals. Tempo usually ranges from 115-125 bpm. Notable artists include *Zedd*, *Kaskade*.

D. Techno

Techno is mainly an instrumental genre, relying heavily on syncopation, and utilize the hard drum beats of hip-hop music. They make use of multi-layered beats by using multiple drum machines and lay tracks over different tracks. Notable artists include *Juan Atkins*, *Luke Slater*.

IV. Feature Extraction

Feature selection is a very important task in all kinds of classification problems. After an extensive literature survey on the types of features used in Music Classification problems [2, 9, 10], we decided to use Rhythmic features for our test. The features that we used broadly fall into two categories: Statistical Spectrum Descriptors and Rhythm Histogram features.

Statistical Spectrum Descriptors describe the content according to the occurrence of beats or other rhythmic variation of energy on a specific critical band. They were computed on 24 critical bands [11] and some of the features extracted were mean, median, variance, kurtosis, skewness, minimum and maximum value. This resulted in a 128-dimensional feature vector.

The Rhythm Histogram features describe the general rhythmic content in an audio [11]. The histogram actually contains 60 bins which reflect the modulation frequency between 0 and 10 Hz. This set is calculated by taking the median of the histogram at a regular interval. The overall length of this feature vector is 60. To extract these features we used the RPextract toolbox[3] in MATLAB. The final feature vector was a 228 length vector. Since a 228 feature space is very large and a considerable amount of training time as well as hidden nodes are required to learn the pattern properly we used the RELIEFF algorithm [12] to select the most prominent features for genre recognition. Using this algorithm, only the 50 most significant features were chosen for our purpose.

V. Neural Networks

Artificial Neural networks are one of the most widely used pattern recognition tools in machine learning. The basic mechanism of working of an Artificial Neural Network is the interconnection of a system of “neurons” which can compute values from inputs. The main purpose of training a neural network is to get the right set of weights between the nodes so that we can get the correct nonlinear function mapping between the input and output. Gradient descent is commonly used to compute the error for each node after which the weights to be adjusted by small amounts in order to better produce the desired output for each input. Once the neural network is trained we can test it by performing feed forward computation on a test set.

Our ANN consists of 228 input nodes (50 for networks using RELIEFF based features), 15 hidden nodes which were found after experimenting and 4 output nodes. We use two algorithms for our purpose. The first algorithm is the Back Propagation Network [13, 14] and the other is the Probabilistic Neural Network [15].

A. Back propagation algorithm

The Back Propagation algorithm trains a given feed-forward multilayer neural network (fig. 1) for a given set of input patterns with known classifications. For each given input pattern, it examines the output response. It then compares this output with the known or desired response to compute the error value. Based on the error, the connection weights are adjusted in a way that reduces the error. This process is repeated until the error is sufficiently small.

Step 0: Initialize small random values to the weights.

Step 1: Feed-forward:

a) Initialize each input neuron (X_i) with the input features.

b) Calculate the weighted sum at each hidden neuron (Z_j) as z_in_j.

c) Apply the sigmoidal function on this sum at each hidden neuron and compute the output signal,

$$z_j = f(z\_in_j),$$

$$\text{where,} \quad f(x) = \frac{1}{1 + e^{-x}}.$$

[3] Available. [Online]: http://www.ifs.tuwien.ac.at/mir/downloads.html

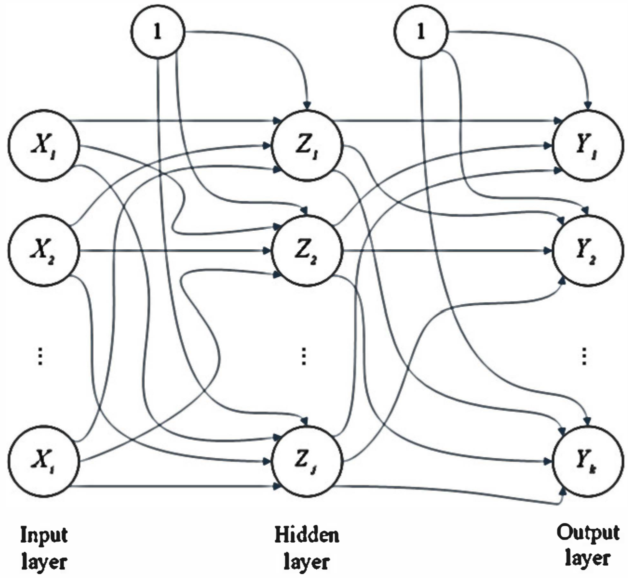

Figure 1. Back propagation neural network architecture.

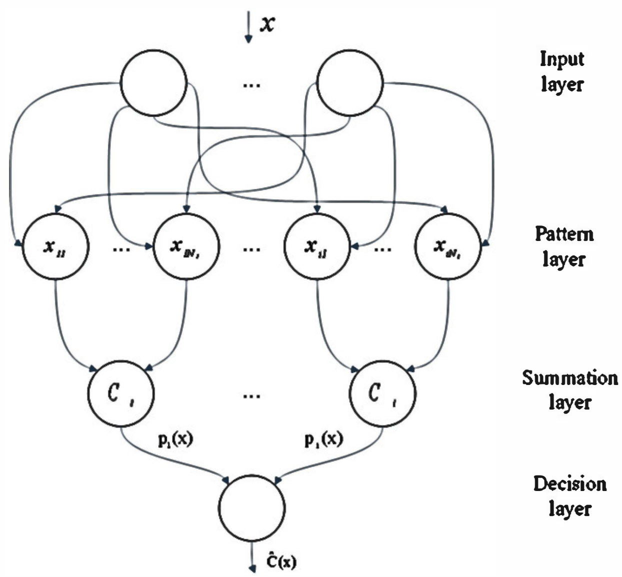

Figure 2. Probabilistic neural network architecture.

d) Now, at each output neuron (Y_k), compute the weighted sum (y_in_k) and apply the sigmoidal function to arrive at the final output,

$$y_k = f(y\_in_k).$$

Step 2: Back propagation of error

a) Computer error information term (δ_k) at each output neuron (Y_k) corresponding to the target value (t_k) and send it to the hidden layer.

$$\delta_k = (t_k - y_k) f'(y\_in_k),$$

where, $f'(x) = f(x)\{1 - f(x)\}$.

b) Each hidden neuron, Z_k, sums its weighted delta inputs, multiplies by the derivative of sigmoidal function to calculate its error information term, δ_j,

$$\delta_k = \delta\_in_j \, f'(y\_in_k).$$

c) Weight updating:

$$w_{hidden}(new) = w_{hidden}(old) + \alpha \delta_k z_j,$$

$$w_{input}\,(new) = w_{input}(old) + \alpha \delta_j x_i.$$

Step 3: Repeat steps 1 and 2 until a sufficiently small error is reached.

B. *Probabilistic Neural Network*

The Probabilistic Neural Network is based on the theory of Bayesian classification and the estimation of probability density function (PDF). A PNN consists of several sub-networks, each of which estimates a pdf for each of the classes using Parzen windows method. The PNN architecture consists of four layers: input layer, pattern layer, summation layer, and decision layer. No. of neurons in the pattern layer correspond to the no. of training examples, and no. of neurons in summation layer correspond to no. of classes.

While performing classification, the input layer neurons do not perform any computation and simply forward the input to the pattern layer. Each neuron in the pattern layer, x_{ij}, then computes its output

$$\varphi_{ij}(x) = \frac{1}{(2\pi)^{d/2}\sigma^d} \exp\left[-\frac{(x - x_{ij})^T (x - x_{ij})}{2\sigma^2}\right]$$

where σ is the smoothing parameter, d is the dimension of the pattern vector; x and x_{ij} represent the input pattern vector and the neuron vector respectively. Each neuron in the summation layer then computes the maximum likelihood of pattern x being classified into class C_i by summing and averaging the output of all neurons belonging to that class (fig. 2).

$$p_i(x) = \frac{1}{(2\pi)^{d/2}\sigma^d} \frac{1}{N_i} \sum_{j=1}^{N_i} \exp\left[-\frac{(x - x_{ij})^T (x - x_{ij})}{2\sigma^2}\right]$$

where N_i is the total number of samples in class C_i.If the *a priori* probabilities for each class are the same, and the cost associated with misclassification are equal, the decision layer makes decision in accordance with Bayes decision rule based on the outputs of all summation units.

$$\hat{C}(x) = \arg\max\{p_i(x)\}; i = 1, 2, \ldots, m$$

where, $\hat{C}(x)$ is the estimated class and m is the no. of classes.

C. Classification

The two algorithms and two sets of features were used to train four different classifiers. One set of features includes all 228 features, and the other includes the features obtained by running the RELIEFF algorithm and selecting the most prominent features; we used 50 most significant for our purpose. We trained the classifiers using 280 songs, which included 70 from each sub-genre. Once the training was complete, we tested the classifiers using the remaining 120 songs, which consisted of 30 from each sub-genre. We used a spread of 15 in PNN, as we achieved optimized results at that value.

VI. Results

The first classifier – BPN with all 228 features – had an accuracy of 94.16% with just 7 songs incorrectly classified. BPN with RELIEFF features, the second classifier, had a comparatively low accuracy of 76.67%. PNN with all features had an accuracy of 78.33%, and PNN with RELIEFF features performed almost as good with an accuracy of 76.67%.

TABLE I. Confusion Matrix - BPN with all features

	Dubstep	Trance	Techno	E-House
Dubstep	28	0	0	2
Trance	0	28	0	2
Techno	0	0	29	1
E-House	0	2	0	28

TABLE II. Confusion Matrix - BPN with RELIEFF features

	Dubstep	Trance	Techno	E-House
Dubstep	28	1	0	1
Trance	5	11	4	10
Techno	0	1	28	1
E-House	0	1	4	23

TABLE III. Confusion Matrix - PNN with all features

	Dubstep	Trance	Techno	E-House
Dubstep	27	3	0	0
Trance	2	20	0	8
Techno	0	3	24	3
E-House	2	5	0	23

TABLE IV. Confusion Matrix - PNN with RELIEFF features

	Dubstep	Trance	Techno	E-House
Dubstep	29	1	0	0
Trance	2	17	5	6
Techno	0	3	26	1
E-House	2	6	2	20

Based on the above results we observe that the first classifier produces a better classification in this case as the error is only 5.84% compared to an average error of 22.78% in the other three cases. Also, by removing features using feature selection algorithms, we observe that the accuracy reduced. But it is always a better option to reduce computation load since the systems will be faster. The reduced accuracy with RELIEFF algorithms can be attributed to the fact that Trance and Electro-house sub-genres consist of quite similar features, thus requiring more than 50 features to provide effective discrimination between the two.

VII. Applications

As music distribution and consumption have evolved digitally and are increasingly gaining popularity, it has led to sales of millions of records, making it seemingly difficult to organize and manage databases. This was true of major genres in the past, but now sub-genres are evolving at a high pace, especially in EDM, which calls for the need to include the organization and management of music tracks on the sub-genre level as well. Most DJs in EDM tend to produce and mix songs only in a particular genre or a group of similar genres. It is, therefore, essential for them to have a database of songs in their genre of interest. The technique we proposed in this paper can be used to categorize databases into its respective genres and simplify this tedious task for producers and DJs.

Although we have demonstrated our approach using electronic music, this technique can be extended to wider genres of music in general such as jazz, pop, rock, RnB, classical, reggae, etc. by training the system with suitable songs.

VIII. Conclusion and Future Work

Both the artificial neural network algorithms performed significantly well in classifying the songs as per their sub-genres. The complete feature set performed comparatively better in both the algorithms when compared to the reduced feature set. This result is attributed to the fact that the distinction between trance and electro-house is quite unclear, and it takes more than 50 features to distinguish between them, making the complete feature vector seem as a better option. The trade-off is with computational load and accuracy. The RELIEF algorithm provides a significant decrease in the computational time, but results in lower accuracy. Hence, for sub-genres with well-defined distinctions, RELIEF algorithm seems more attractive. Using the proposed features on four sub-genres, BPN produced an accuracy of 94.16% with the complete feature set.

A fruitful approach to future research involves adding more sub-genres and an increased dataset for a wider process of training and classification. Additionally, different sets of features may be selected as input vectors in order to compare and contrast the results with each of them. As the number of sub-genres and input vectors increase, BPN may no longer prove to be the optimal solution. Also, systems can be developed which help DJs to not only find songs from similar genre but also songs which can be mixed based on a similarity feature so that the overall mixing process produces a great sound. Further, unsupervised training algorithms can be utilized to see how well those approaches work. Moreover, this technique can be used to develop systems which recommend music in genres not just limited to EDM. Software can also be designed that can automatically extract the most representative part of the song, which can be combined with the recognition system to make automatic music genre classification easier and convenient. Lastly, extraction of more convenient features can make it possible to implement artist recognition algorithms so that the digital library is better organized than just being organized by genre.

ACKNOWLEDGMENT

We are deeply indebted to Dr. Arulmozhivarman for his valuable feedback on this paper. We would also like to thank Prof. Mohanaprasad for his support and guidance during the course of this paper.

REFERENCES

[1] G. Tzanetakis and P. Cook, “Musical Genre Classification of Audio Signals,” IEEE Trans. Speech Audio Processing, vol. 10, no. 5, pp. 293-302, July 2002.

[2] T. Li and G. Tzanetakis, “Factors in Automatic Musical Genre Classification of Audio Signals,” in Proc. Workshop Applications of SignalProcessing to Audio and Acoustics (WASPAA), New Paltz, NY, Oct. 19-22, 2003.

[3] S. Clark, D. Park, and A. Guerard, "Music Genre Classification Using Machine Learning Techniques,” Sep. 2012.

[4] M. Levy and M. Sandler, “A semantic space for music derived from social tags,” in Proc. ISMIR 2007, pp. 411–416, 2007.

[5] P. Lamere: “Social Tagging And Music Information Retrieval,” Journal of New Music Research, Vol. 37, No. 2, 2008.

[6] D. Diakopolus, O. Vallis, J. Hochenbaum, J. Murphy, and A. Kapur,“21st Century Electronica: MIR Techniques for Classification and Performance,” in Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR 2009), Kobe, Japan, 2009.

[7] B. Rocha, N. Bogaards, & A. Honingh, "Segmentation and Timbre Similarity in Electronic Dance Music," Proceedings of the SMC, 2013.

[8] P. Kirss,“Audio Based Genre Classification of Electronic Music”. Master’s thesis, University of Jyväskylä, June 2007.

[9] G. Peeters, “A large set of audio features for sound description (similarity and classification) in the CUIDADO project,” Tech. Rep., IRCAM, 2004.

[10] Michael A. Casey, Remco Veltkamp, Masataka Goto et. al., “Content-Based Music Information Retrieval: Current Directions and Future Challenges,” in Proceedings of the IEEE, Vol. 96, Issue 4, 2008, pp. 668-696.

[11] Thomas Lidy, Andreas Rauber, “Evaluation of Feature Extractors and Psycho-acoustic Transformations for Music Genre Classification,” Proceedings of the Sixth International Conference on Music Information Retrieval (ISMIR 2005), pp. 34-41, London, UK, September 11-15, 2005.

[12] K. Kira and L. Rendell, “The Feature Selection Problem: Traditional Methods and a New Algorithm,” Proc. AAAI’92, San Jose, CA, July 1992, pp. 129-134.

[13] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, “Learning representations by back-propagating errors,” Nature, Vol. 323, 1986, pp. 533–536.

[14] Laurene V. Fausett , “Backpropagation Neural Net,” in Fundamentals of Neural Networks, 1st ed. Pearson Education, pp. 289-300.

[15] Donald F. Specht, “Probabilistic Neural Networks,” Neural Networks, Vol. 3, pp. 109–118, 1990.