24787 ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING FOR ENGINEERING DESIGN
CLASSIFICATION OF MUSIC USING NEURAL NETWORKS

Anton Galkin **Pushkar Rege** **Ryan Pocratsky**

ABSTRACT

In this project we have attempted to classify 'mp3' files into different genres using a multilayer feed-forward artificial neural network. We extracted features from the 'mp3' files like tempo, the power of key frequencies of individual sections, and power of key frequencies between beats. These features were used as inputs to a multilayered perceptron network which would output the song's genre classification. Attempting to classify 12 genres resulted in poor accuracy. By reducing the scope to 3 genres, we were able to achieve 72% accuracy on the validation data set. An overview of the approach taken, suggestions for further improvements, and indications of further work are presented.

INTRODUCTION

Classifying music by genre manually is time-consuming and tedious. Current methods involve classifying a band or album as one genre rather than classifying individual songs. Individual classification allows for similar songs to be grouped into playlists independent of band or album information. Then, users can create playlists based on genre and be assured that only similar songs will be played.

The objective of this project is to create a neural network that can classify music by genre. A neural network was selected because of the number of training samples available and the number of features per sample. Since a neural network processes the inputs in parallel, it can classify a song quicker than other classification methods once the feature data is processed.

Each song has several features that could be extracted as inputs to the network. These features relate to the song tempo, amplitude, and frequency information in certain sections of the song. These features are extracted from each song and stored in a database by running the preprocessing MATLAB code. Utilizing the Netlab code from Aston University, a multilayer perceptron network was created, trained, and validated with the database.

Originally, 100 songs for each of 12 genres (1200 songs total) were collect to be processed and stored. However, the feature processing time limited us to processing 450 songs. The limited number of songs per genre was one reason for poor results when attempting to classify songs into 12 categories. Reducing the scope to focusing on three genres with 100 samples each provided better results. For a neural network with 558 inputs, 250 hidden nodes, k-fold validation number of 3, and 2000 training iterations, a 66.7% training accuracy and 72.6% validation accuracy was achieved.

RELATED WORK

Similar work has been done in the Media Lab at MIT for classification of Folk Music using Hidden Markov Models [1]. The best classification performance achieved was 77 percent on the validation data set. Tong Zhang, from HP Laboratories, also worked on music classification. He used a decision tree method to categorize songs based on key aspects and features within the song [2]. The categories used were not correlated to genres but rather singer gender and instruments played. His results indicate that is it possible to automatically group music based on key extracted features.

Another machine learning course at Carnegie Mellon University, 15781, has attempted music classification with neural networks and learning vector quantization [3]. The neural network used had 128 inputs, one hidden layer with 30 nodes, and an output node for each of the four genres analyzed. About 38,000 training input samples were extracted from 96 songs, 24 songs per genre. Additionally, the number of training iterations used was 200,000. The classification confidence for the neural network was about 55%. The learning vector quantization method resulted in a confidence of about 70%.

PLANNED USER INTERACTION

Once the network is developed to accurately classify songs, a program could be created for consumers to categorize their mp3 collection. The user interface would be similar to that shown in Figure 1.

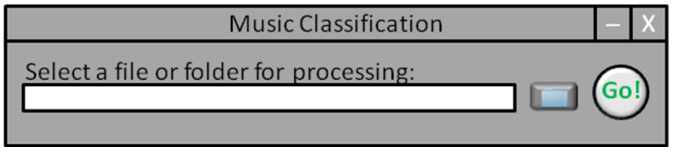

Figure 1: Envisioned Graphical User Interface

Future program design will utilize the trained networks and take the input of a directory or music file. It will automatically identify 'mp3' files in a directory and change the genre information of each song. Once the 'Go' button is pressed a window will pop-up to give an estimate of the time required to process the task. The pop-up window will also indicate when the task has completed.

TECHNICAL APPROACH

Before attempting to solve this problem using alternate classification methods like K-nearest neighbor and K-means

clustering, the data was checked for whether it was separable into the desired categories. These methods would allow for quick additions of new genres classifications because it only involves learning data related to that new genre. However, both methods failed to classify music. They resulted in classification accuracy of less than 10% which was worse than the accuracy of a random classification. A neural network was implemented because it performs sufficiently better on data that is not separable.

The accuracy of the neural network's categorization ability depends in large part on the features used to identify each song. If the features represent meaningful data, are representative of the song as a whole, and distinguish it from songs of another genre, then they should be useful to correctly predict the classification of the song.

Thus, feature selection was of paramount importance to the success of the neural network. The first and easiest option was taking the raw data and feeding it into the neural network as features. However, digitizing a song and turning it into a discretized waveform resulted in a huge amount of data. At the standard 44.1 KHz sampling rate, a 3-minute song has almost 8,000,000 floating point values per track, which is in excess of what a neural network can reasonably handle.

The preferred approach for such a large volume of data was collecting a set of meaningful statistics which summarized various aspects of the data. In the case of a song, it was determined that tempo and frequency data would be the key area of focus. These features for different instruments and vocal styles, which are characteristic of particular genres of music, would produce distinctive signatures in each genre classification.

Key frequencies analyzed were:

f_{key} = [50 100 250 500 1000 1500 2000 2500 3000 4000 6000 10000] (frequency in Hz)

They are more closely spaced at the low end of the spectrum for higher resolution and represent a wide range of instruments of various frequencies [4]. Typical frequency ranges of common instruments are shown in Table 1.

Using a fast Fourier transform, the waveform of each song was represented in a frequency space. The power was numerically calculated by dividing by the number of data points and taking twice the absolute value. The higher frequencies were truncated because they are outside the range of human hearing. This allowed the power of each key frequency to be recorded to discern various musical instruments or vocal styles characteristic of particular musical genres.

For this analysis, each song was broken down into 10 sections. The program created for feature processing can but used to vary this number. Increasing the number of sections analyzed offers better resolution into the characteristics of each genre but increases the number of input into the neural network. Additionally, comparing results using different numbers of sections requires significant computational time and memory to calculate and store the features from the lengthy pre-processing session for each option.

Table 1: Frequency ranges of common instruments [4]

INSTRUMENT	FREQUENCY (HZ)
ELECTRIC GUITAR (BODY)	240
ELECTRIC GUITAR (CLARITY)	2500
ACOUSTIC GUITAR (BOTTOM)	80
ACOUSTIC GUITAR (CLARITY)	2500
BASS GUITAR (ATTACK)	700-1000
BASS GUITAR (BOTTOM)	60-80
BASS GUITAR (STRING NOISE)	2500
PIANO (BASS)	80-120
PIANO (PRESENCE)	2500-5000
PIANO (CRISPNESS)	10000
PIANO (HONKY-TONK)	2500
PIANO (RESONANCE)	40-60
HORNS (FULLNESS)	120-240
HORNS (SHRILL)	2500-5000

Thus, 10 sections of a song were used because it gave a significantly smaller data set than the entire waveform while giving information on the changes that can occur throughout a song. The overall effect was detecting variations within a single song along the time scale – how the beginning may have different frequency distributions than the middle or end, echoing the tendency of some musical genres to have long intervals of comparative quiet or noise, or being able to isolate the characteristics of an instrument solo during part of a song.

Functions were also used from Professor Dan Ellis, at Columbia University, to estimate the overall tempo (beats per minute) of each song, as well as the timestamp of each beat. The frequency analysis performed on each section was then repeated, but limited to the time interval between two consecutive beats (typically on the order of one second) which occurred closest to the meeting point of two sections. This allowed the option to emphasize minute nuances of a song which occur on a small time scale, which might otherwise escape detection using the 10-section approach.

Overall, this gave a wealth of data which could appropriately characterize each song in its appropriate genre. Namely: tempo, power of key frequencies of individual sections, and power of key frequencies between beats. For the 10 sections of a song, the number of inputs was 558.

Based on these features as inputs to a multiple layer perceptron, the neural network was optimized to achieve the

maximum validation and training accuracy by altering the number of hidden nodes, k-fold validation number, and number of training iterations (epochs). Since the number of inputs was 558, the number of hidden nodes was varied from 25 to 300. Performing several initial tests, it was observed that epochs between 1500 and 3000 provided significant reduction in the error rate. Epochs greater than 3000 provided little benefit for the time cost associated with additional iterations. Therefore, the epochs were varied between 1500 and 3000. The k-fold validation number was varied from 3 to 4 because the neural network performed better with more training samples. A program was setup to vary these three parameters and test the accuracy of the created network to determine the best performing networks.

RESULTS

Based on the samples processed, the greatest training accuracy obtained for classifying 12 genres was only 25%, shown in Figure 2. The classification tended to favor the genres that more samples were processed from. These results are better than randomly selecting a classification. However, it cannot be used to classify music in a consumer application. Therefore, the classification scope was reduced to three genres: Classical, Rap, and Electronic.

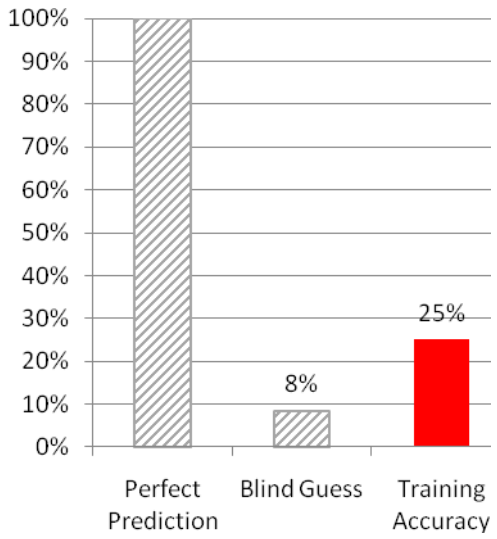

Figure 2: Best training accuracy results of a neural network trained to classify the song samples into 12 genres using 200 hidden nodes, 3-fold validation, and 1750 epochs

The best performing neural networks for classification between these genres are shown in Figure 2. Each network shown in Figure 3 utilized 3-fold validation to obtain these results.

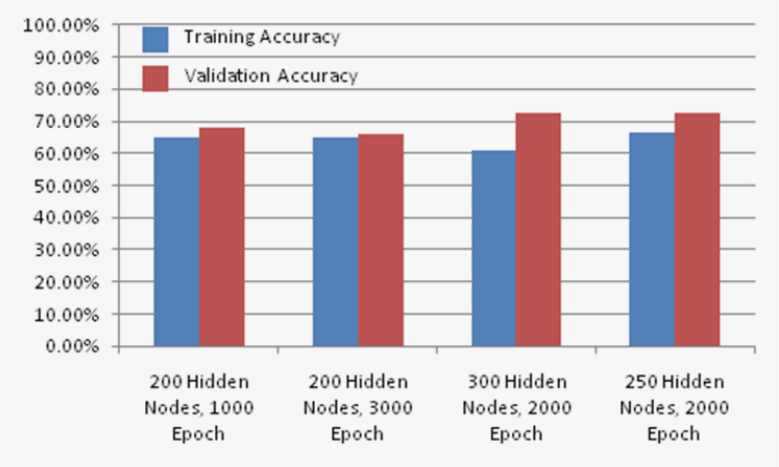

Figure 3: Highest performing neural network results for classification between classical, rap, and electronic.

The best performing network gave a training accuracy of 66.7% and validation accuracy of 72.6%. This network had 250 hidden nodes and 2000 training iterations.

DISCUSSION

The accuracy for classification of training data and validation data was compared. A few neural networks showed indications of over-fitting to the training data set. Additional song samples per genre could improve the accuracy of the validation set while reducing the likelihood of over-fitting. The neural networks with hidden nodes between 200 and 300 tended to perform the best. The accuracy of the best performing networks increased slightly as the epochs increased above 2000 but over-fitting was more likely.

The biggest challenge in this project was selection of the correct attributes to be given as inputs to the neural network. Accurately capturing the features that distinguish songs between highly overlapping genres requires a significant knowledge of music and signal processing. Other features such as emotion and the meaning of lyrics that help classify songs cannot be obtained from a simple analysis of the 'mp3' waveform. Also, compared to a 'wav' file the 'mp3' format is lossy. A significant amount of data is lost in the 'wav'-'mp3' conversion. This limited the features that could be extracted from the songs. However, 'mp3' files are the most popular music format.

Another major challenge was determining the genre of the songs which were used for training the neural network. Some genres were virtually indistinguishable. There was a significant overlap between songs that have the following genres: Rock, Pop, Alternative and Metal. Incorrect classification of training samples would lead to incorrect learning of the neural network. Hence, careful consideration was needed while attaching labels to different songs. A further break down of main genres is to sub-level style classifications may need to be performed to achieve greater accuracy. Another approach is to identify key songs that clearly define a genre classification. Then, use only these key songs to train the neural network. However, this

requires a subjective studying of music within a genre and personal definitions of each genre.

From the machine learning point of view, the biggest challenge was that the classification in the parametric space was not linear. This was verified by applying the 'K-nearest neighbor' and 'K-clustering' to the training dataset. A neural network was selected since they are effective when data is not linearly separable. However, the overlapping training data set could be too similar to train the neural network.

CONCLUSIONS

We were able to design a neural network that could classify a song as either classical, rap or electronic with an accuracy up to 72%. This accuracy is similar to the results obtain by others. With additional samples and identifying key training samples that define each genre, further improvement on the accuracy is possible.

FUTURE WORK

One of the main limitations of our project was the inability of the system to classify between overlapping genres. We believe that with better selection of features, better defined genre classifications, and a larger training set of songs that clearly define each genre we will be able to overcome these limitations. Obtaining the proper features to classify a song will also be useful for additional areas of study.

Once these features are identified, they can also be used along with user input to train another neural network to recognize personal preference. With personal preference identified, locating songs that a consumer will like with greater accuracy can improve online music stores suggestions, like those from Amazon or iTune, and improve the listening experience for online music players like Pandora.

Additionally, identifying those features and linking a metric of popularity to the training data set could give insight into what causes a song to be popular. This information is valuable for music label companies and band. Music labels could use it to recognize if a new band will become popular before signing them to a contract. Bands could also use the information to create songs that match the characteristics of popular songs. Being able to accurately classify music provides several other opportunities for understanding various aspects of music.

ACKNOWLEDGMENTS

We would like to thank Professor Levent Burak Kara and all his TAs for their help and support during this project. We would also like to thank Professor Dan Ellis, Associate Professor of Electrical Engineering at Columbia University for the code used to read an 'mp3' file in MATLAB and determine tempo information.

REFERENCES

[1] Chai, Wei and Vercoe, Barry "Folk Music Classification Using Hidden Markov Models," Media Laboratory, Massachusetts Institute of Technology, Cambridge, MA: http://alumni.media.mit.edu/~chaiwei/papers/chai_ICAI183.pdf

[2] Zhang, Tong "Semi-Automatic Approach for Music Classification," SPIE Conference on Internet Multimedia Management Systems IV, 10 September 2003, Orlando, Florida: http://www.hpl.hp.com/techreports/2003/HPL-2003-183.pdf

[3] Talupur, Muralidhar; Nath, Suman; Yan, Hong; "Classification of Music Genre" Project Report for 15781 Carnegie Mellon University, Electrical and Computer Engineering: http://www.cs.cmu.edu/~yh/files/GCfA.pdf

[4] Yates, Robert "1/3 Octave Frequency Chart" April 2011: www.reverse-engineering.info/Audio/bwl_eq_info.pdf