Classification of music genres using machine learning
Deekshith H A
Department of Computer Science and Engineering
GITAM School of Technology, Bengaluru campus, Karnataka, India

Nithin Singh P
Department of Computer Science and Engineering
GITAM School of Technology, Bengaluru campus, Karnataka, India

Kamala L Assistant professor
Department of Computer Science and Engineering
GITAM School of Technology, Bengaluru campus, Karnataka, India

Deshik S Shinde
Department of Computer Science and Engineering
GITAM School of Technology, Bengaluru campus, Karnataka, India

Varun A S
Department of Computer Science and Engineering
GITAM School of Technology, Bengaluru campus, Karnataka, India

ABSTRACT
Music has always been one of the top online as well as offline usages. But not all people want to hear the same kind of rhythmic songs or genre in music. In the current world, there is a problem of automatically differentiating songs based on the user’s interest. There are a few softwares that classify the music genres from the data available in the details of the song, but it doesn’t go
through the internal rhythmic structure to classify it whether it is a rock, hip pop, etc. based on the genre. Other available softwares scans the description of the song given and differentiates it based on the readily available data. In order to overcome this complexity, we will classify the music through the help of machine learning technique and use several algorithms to classify it. In order to do that, we will classify the data into training data as well as test data. Using this we are going to classify the music genres and provide the easy way for the user to get the songs in their area of interest.

INTRODUCTION

With online music libraries that are readily available and easily accessible music material, people are finding difficult to classify the songs in their playlist based on their individual interests. One can classify and arrange the songs based on their genre. This Genre includes harmonic content, unique instrumentation and rhythm which may differ from one song to another. Nowadays, we can see this type of classification of songs based on the genre in few Audio streaming applications like Saavn, Wynk, Spotify and iTunes. These types of applications are widely used because of their unique classification feature.

Our model is designed based on the machine learning algorithm which uses convolutionary neural networks which will take audio files as an input, then identifies and divide the songs based on the genre of the songs. And also uses the same training process involved in training audio signal of MEL spectrogram. Our model uses time and frequency domain of audio signal feature for effective classification of songs which is taken from ML models like Logistic Regression, Random Forests and Vector Support Machines.

Based on audio dataset, the models are evaluated. With the help of above proposed models we are trying to add-on some extra features into our model, so
that we can build good system for songs classification. Our paper work includes the following.

- Existing system
- Problem Statement
- Proposed system

LITERATURE REVIEW

Discover new features for classifying music

This study suggests the automated grouping of music aims to organize the songs into different categories by using genre as classifier. This paper presents the work on testing the system for automatic tagging the songs based on music. These features try to distinguish between different albums and also uses music lyrics for classification. The quality of this type of system is evaluated based on the binary decision tree boosting algorithm. The effectiveness of the task in hand is explored in terms of very common features of the MEL Frequency Cepstral Coefficients. It can be seen that this type of systems provides unique features that will benefit for summarizing the many different aspects of songs.

Automatic music classification and summarization.

Automated classification and description of music is very helpful for indexing the music for faster retrieval and distribution of music over online. Extracting the popular themes from unstructured music data is very difficult. This work presents the usage of specific algorithms like SVM which split the music into pure and vocal based on the resulted attributes and also another algorithm is used for structuring, identifying and generating the music content automatically based on the knowledge of clustering concepts. The SVM method
is very efficient in music classification compared to Euclidean method. Accuracy of the music is checked by using listening methods on both the pure and vocal music list.

**A coherent architecture for the interpretation of natural language: Deep neural networks with multitasking learning.**

We define a unified convolutionary neural network framework that produces a set of language processed predictions given as sentence: speech tag, fragments, semance positions and semance-related words. With the help of strength-sharing and multitask learning methods all these tasks are trained together on the entire network. Apart from language model learned from unlabelled text, all tasks use labelled data and reflect a new type of semi-supervised learning. It will also show the improved widespread of multitask learning and semi-supervised learning.

**Music genre classification with machine learning techniques**

The goal of this study is to use machine-learning techniques to predict the genres of the songs. For this function, the extraction of features is done using signal processing techniques, and then machine learning algorithms are applied with these features to make a multi-class music classification.

**Genre classification of audio content using various classifiers and set of features**

For classifying the genre of musical piece, it evaluates the output of various classifiers on multiple audio feature sets. We also test performance of sets of features collected through tools to minimize dimensionality for each classifier. Ultimately, we are working on moving up the accuracy of classification by combining various classifiers. The accuracy of test genre
classification is approximately 80% plus or minus 4.2 percent on 10 genre sets of 1000 pieces of music using a collection of different classifiers. This result is higher than 71.1 plus or minus 7.3 percent which is the highest on this data collection. We often gain classification accuracy of 80 percent by using reduction in dimensionality or by putting together various classifiers. It is seen that best set of features is based on classifier used.

**The grouping of audio signals by musical genre**

Musical genres are vitally important categories for characterizing pieces of music produced by humans. The musical genre is categorized by similar features its members share. Usually these features are in relation to the Instrumentation of the song, rhythmic form and harmonic content. Hierarchies of genres are generally used to structurize the huge music collection available on the net. Annotation of musical genres is usually done manually. Automatic classification of music genres in this process will assist or substitute the human consumer, which will be a beneficial addition to music recovery systems. In addition, automatic classification of musical genre provides a basis for the creation and evaluation of functionality for any form of content-based analysis of musical signals. Here we discuss the automatic audio signals classification into a music genre hierarchy. In particular, it proposes 3 functionality sets to represent texture, rhythmic structure and pitch of music. Training classifiers for numerical pattern recognition using actual-world music samples can determine the efficiency and relative significance of the features proposed. It defines both the entire file and the classification schemes dependent on real-time frames. Categorisation of 61 per cent for ten music-based genres is accomplished using the proposed feature sets. This is comparable with the results published for classification of the human musical genre.
METHODOLOGY

SVM:

The support-vector machine builds hyperplane or collection of hyperplanes in an immense- or limitless-dimensional space that is used for classification, reconstruction or other tasks such as outlier’s recognition. Instinctively, the hyperplane with the greatest distance to the nearest training-data point in either class achieves a strong separation, as normally the greater the difference, the less the classifier's generalization error.

Random Forest:

Random forests are a learning method for grouping, regression and building a number of decision trees for training the model and which generates individual trees of different classes of classification and regression methods. Random forests make the clear decisions by using decision trees to over fit the training data collections. It follows 10 steps,

Step 1-- Information Pre-treatment
Step 2-- Separating Training and Testing Datasets
Step 3-- Transforming the Data
Step 4-- Building the CNN
Step 5-- Running Predictions on the Test Set
Step 6-- Checking the Confusion Matrix
Step 7-- Making a Single Prediction
Step 8-- Improving the Model Accuracy
Step 9-- Adding Dropout Regularization to Fight Over-Fitting
Step 10-- Hyper parameter Tuning

**Modules Description**

1. Data Collection
2. Image Obtainment and Pre-Treatment
3. Data Planning and Development of a model
4. Training Model
5. Testing and evaluating model

**Data Collection**

Collect the low-resolution animal image dataset from trustable source.

**Image Obtainment and Pre-Treatment**

We are getting the data from an online source in this section. We collect the data in the form of different images and then we maximize the size of the image for further use. In our model we use an algorithm for image size modification based on its requirements. And this process of image size maximizing or scaling also affect the resolution of an image which will also improve the performance of the model.

All computers can understand data in the form of binary digits only. The images that we see don’t understand by the computer. So, we have to translate the images to binary values. The images are changed to black and white and then each image is assigned with different binary values based on each pixel of that particular image. All these assigned values are stored in a sequence in an array and these values are easily accessed for performing calculations.
Data Planning and development of a model

Most of the time, first people split their collected data into sample and validate datasets. This sample dataset is used for building the model effectively and this set is also used to train the model. Then the model is built and well trained. Then the model is tested with validate dataset actual values to verify the authenticity of the model. When planning the data for the training and test phases also include the same process.

Training model

After model building, model training process involves providing learning data from real world examples as training data. After the model is trained will get partial values, these values are the output of training model and further comparing it with actual values for its accuracy.

Testing and evaluation of the model

Evaluation Pattern. Model Assessment is an important part of the cycle of design development. It helps find the best model representing our results, and how well the model chosen will perform in the future. ... Both approaches use a test range (not used by the participant) to evaluate participant output to prevent over fitting.

Evaluation Steps:

• Embellish the need for early test results. Raise test data issue as early as possible, as early as the planning phase of the test.

• In-depth surveys during design tests. Analysing future test data will occur early in the design process of the test.

• Build data for study.
• Run checks.

• Data save.

FLOW CHART
SYSTEM DESIGN
HARDWARE REQUIREMENTS

- Device: 2.4 GHz Pentium IV.
- Hard drive: 500 GB.
- Four GB of Ram

desktop or laptop machine with the above configurations is required or of the higher configuration

SOFTWARES NEEDED

- OS : Windows XP / 7 /above
- Programming Language : Python
- Software : Anaconda
- IDE : Jupyter Notebook

Anaconda

Anaconda Distribution includes conda and navigator Anaconda, as well as Python and hundreds of science packages. You installed all of these too when you installed Anaconda. You should use both the conda and Navigator to see the products and environments are right for you to navigate. You can also move between them, and you can see the research you are doing with one in the other.

Jupyter Notebook

Jupyter Notebook is an innovative programming environment which allows us to create notebook documents such as: - Live codes, Interactive widgets, Plots, Equations, Images.
This type of documents is compatible to be shared via email, Dropbox, GitHub version control systems.

**Python programming**

Python is a comprehensible, functional programming language, which is purely object oriented and supports high level data structures. The pleasant syntax and efficient typing combined with interpreted structure makes Python a programming language which is widely used for scripting and application development, where it can run ion many platforms.

Python is a high-quality, interpreted, collaborative, and object-oriented scripting language. It's designed to make Python highly legible. This also uses keywords in English where punctuation is used like other languages and has less syntactic structures than other languages.
RESULT

A result is the final outcome of the qualitatively or quantitatively represented acts or events. Analysis of performance is an organizational analysis, is a collection of simple quantitative relation between the quantities of results.
Predictions and Evaluation of RF Model

```
In [16]: predictions_rf = rf.predict(X_test)

In [17]: acc_rf = accuracy_score(y_test, y_test, predictions_rf)

In [18]: print(f'Overall accuracy of the model using test set is : {acc_rf*100}%')

Overall accuracy of the model using test set is : 78.99999999999999%

In [20]: print(classification_report(y_test, predictions_rf))

            precision    recall  f1-score   support

           0       0.21      0.61      0.31         6
           1       1.00      1.00      1.00       12
           2       0.00      0.33      0.01         4
           3       0.00      0.33      0.01         4
           4       0.00      0.33      0.01         4
           5       0.00      0.33      0.01         4
           6       0.00      0.33      0.01         4
           7       0.00      0.40      0.02         2
           8       0.00      0.33      0.01         4
           9       0.00      0.33      0.01         4

   micro avg       0.19      0.79      0.23     100
   macro avg       0.12      0.79      0.23     100
```
CONCLUSION

This program falls under nowadays sophisticated methodology of machine learning. RF, SVM is better suited for numerical processing especially in classification. In conclusion, as described through the literature review, we conclude that only a marginal progress is achieved in developing a predictive model for classification of Music Genres, and hence the need for combinational and more complex models to improve the accuracy of classification of the music. We conclude that the experimental outcome is more reliable than what we get from the existing method.

REFERENCE

1. Gerald Penn, Abdel-rahman Mohamed, Li Deng, Hui Jiang, Dong Yu and Ossama Abdel-Hamid. August 2014. Convolutionary spoken word comprehension networks. IEEE / ACM Audio, voice and language management teams 22(10):1533-1545

2. Donald Geman and Yali Amit. 1997. Form quantization, decision trees identification.

3. Breiman. 1996. Predictors. 24(2):123–140. Machine learning. With Leo Breiman. 2001. Random decision forests. 45(1):5–32 Machine learning.

4. Vladimir Vapnik and Corinna Cortes. In 1995. Support vector networks. 20(3):273–297 Machine learning.

5. Paul Mermelstein, along with Steven B Davis. 1990. Comparison of parametric representations for identification of monosyllabic phrases in continuously spoken sentences. In Readings on understanding of expressions, Elsevier, pages 65–74.

6. Dan Ellis. 2007. Analysis and synthesis of the Chroma functions. Laboratory tools for Speech Recognition and Organization and Audio-LabROSA.