Music Instrument Recognition

Using KNN Techniques

Jyotisubham Panda
M.Sc. Data Science

Chandigarh University

UID - O21MSD16112

jyotisubham.panda@gmail.com

Link- Music Instrument Recognition using KNN

Abstract

In this research paper, I have gathered some Audio datasets from Kaggle and tried to recognize the instruments by which the audio is played using the KNN model. The Audio dataset consists of 10 different types of instruments and I divided it into 11 classes (Cello, Flute, Double_bass, Saxophone, Bass_drum, Violin, guitar, clarinet, hat, Snare_drum, other). During the work, I found the average precision, recall, and f1 score are 0.73, 0.72. 0.71 respectively. In future work, I want to make some Deep learning projects like CNN, and RCNN with the same data set and compare all the models to find out more accurate model.

Introduction: -

Music! everyone loves music. It's like a daily activity for most people. Music instrument recognition is a very large and fascinating field of research. Where it seeks to automatically identify the tune and predict which musical instrument is used to play that tune.

Lately, there is some progress has been made in this field and all thanks go to new machine learning and deep learning techniques. These models are highly developed and help us identify a vast range of musical instruments such as guitars, drums, pianos, etc.

Still, there is a significant amount of challenges in this particular field of an area where the background noises and classical Instruments do not have been made the recognition to the particular world and also different playing styles.

My aim is to build a model based on KNN which helps develop a robust and accurate music instrument recognition system by training and testing huge numbers of data sets that can work in real life.

Related Works: -

There have been many studies on the recognition of musical instruments, and in order to develop the right models, researchers looked into a wide range of machine learning methods, including support vector machines, neural networks, and deep learning. Some of the common databases used in this area are the IRMAS, NSynth, and GTZAN datasets.

There is still room for improvement, but achieving high accuracy rates in music instrument recognition is still a challenging assignment. Transfer learning, which entails using pre-trained models on massive data sets to improve the performance of Music Instrument Recognition systems, is one area that researchers are pursuing. Transfer learning has demonstrated promise in diminishing the volume of data necessary to train a model and increasing accuracy.

Data augmentation is another area of study that involves modifying the existing data in order to produce new training examples. It has been demonstrated that data dietary supplementation. could improve the performance of music instrument recognition systems by diminishing overfitting and improving the model's ability for generalization.

Furthermore, ensemble techniques have been researched to boost the precision of music instrument recognition algorithms. Ensemble methods combine the results of various models to produce a single forecast. It has been demonstrated that doing so increases the validity of music instrument recognition algorithms and strengthens their resilience to noise and data variability.

Aim and Objectives: -

The most important objective of the music instrument recognition project using KNN is to come up with a machine-learning model that can appropriately categorize audio samples into a variety of instruments. The project's unique objectives are summed up as follows:

Creating a dataset of audio samples for various musical instruments: The first goal is to assemble an ample wide-ranging dataset of audio samples for various musical instruments. In order for the model to be valid and applicable to an extensive selection of musical styles, the dataset should contain samples of various playing styles, musical classifications, and recording studios.

Getting valuable characteristics out of the audio samples: The second goal is to get a set of useful features out of each audio sample that can capture the distinct characteristics of various instruments. This involves the inclusion of methods like Mel-frequency cepstral coefficients (MFCCs) and Fourier analysis.

Implementing and optimizing the KNN algorithm: The third goal is to put the KNN algorithm into the routine to enhance its performance by tweaking its hyperparameters, such as the number of neighbors, distance measurement, and weighting scheme. Finding the optimal set of hyperparameters that optimize the model's accuracy requires implementing algorithms like grid search and cross-validation.

Evaluating the model's performance: The fourth goal is to assess how well the model works using metrics like accuracy, precision, recall, and F1 score on a holdout.t collection of audio samples. This involves analyzing the model's strengths and failings by contrasting the model's predictions with the labels on the ground truth data.

Visualizing and interpreting model decisions: The fifth goal is to use techniques such as scatter plots, decision boundaries, and confusion matrices to illustrate and clarify model decisions. Understanding how the model partitions the feature space and makes classification decisions is required, as is identifying misclassified instances in order to better the feature extraction and selection process.

The project's ultimate aim is to build an accurate and trustworthy model that can be used in real-world applications like music information retrieval, instrument transcription, and audio content analysis.

Significance of the Study: -

The significance of the study depends on its potential to enable an extensive selection of real-world applications in the field of music. Here are some of the study's possible impacts:

Music Instrument Recognition systems can be used in music production to automate tasks like instrument separation and mixing, thereby minimizing the time and effort necessary to produce high-quality music.

Physical music instruction Recognition systems can be used in music education to help students comprehend and appreciate various musical instruments, improving their ability to perform and compose music.

Music analysis can use Music Instrument Recognition systems to automatically organize and analyze large audio datasets, allowing researchers to find new patterns and insights in music.

Music Therapy Instruments Recognition systems can be used in music therapy to help people with hearing loss or other disabilities understand and take in music, thus enhancing their quality of life.

Music Instrument Recognition systems can be used in the music industry to improve the accuracy of music recommendation systems and evaluate the usage of musical instruments throughout genres, allowing for more informed decision-making.

The results of this study will pave the way for a wide range of applications in the field of music with the development of a robust and accurate Music Instrument Recognition system, augmenting our enjoyment and comprehension of music and facilitating its production, analysis, and education.

Research Methodology: -

In order to complete my research first, I have to collect an audio folder consisting of more than 300s audio files which I collected from kegel.com, and import that folder into my notebook.

The above process involves an important installation of a python library called librosa which is generally used for analyzing and processing audio signals. It provides a wide range of tools for audio analysis like loading and saving audio files in various formats like .mp3, .WAV, etc. It also extracts various features from Audio signals such as spectral features (e.g. male frequency cepstral coefficient and rhythm information and harmonic) etc. Furthermore, it helps in visualizing manipulating and performing audio signals which are also known as signal processing techniques.

Then I do some EDA(Exploratory Data Analysis) on the audio file to look at how it was performing in my module and did some wave uploading floating to understand the audio files more accurately. Also, I convert that particular audio file to monophonic as well as harmonic plus precursive web plot and also find it in spectrogram form. This completes my exploratory data analysis part.

Starting the KNN module, I have to import some necessary libraries which will train and test my modules and find the accuracy of recall, precision, and F1 scores which I Found to be 73%, 72%, and 71% accurate Respectively.

Classes	Cello	Flute	Double bass,	Saxo-phone	Bass drum	Violin	guitar	clarinet	hat
Precision	100%	88%	60%	67%	80%	40%	50%	88%	80%
Recall	75%	100%	86%	75%	50%	50%	38%	100%	57%
F1 score	86%	93%	71%	71%	80%	62%	44%	93%	67%

Table 1:- Precision, Recall and F1-score Accuracy for all the Classes

In order to find those accuracies I have to classify my folder into 10 different classes and then compare it by label encoding.

Also, I have to give a number of MFCCs which is used for feature extraction techniques in audio, signal processing partially in speech recognition, and music analysis. After defining the function to calculate MFCC I have to load files and calculate features and create a feature vector, who is leads me to find a feature vector safe then import the train test split to do a training and testing model where I found the shape of training and testing sets.

Now the important work is to fit those training and testing sets into the KNN model where I evaluate the recall precision and F1 score and also find out the accuracy. In the classification report, I found the macro average and weighted average of all precision, recall, and F1 score.

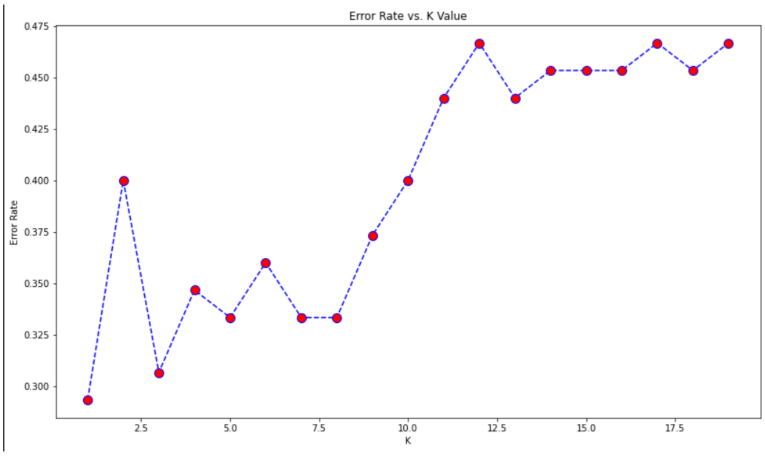
	Precision	Recall	F1 score
Macro average	73%	72%	71%
Weighted average	72%	71%	70%

Table 2:- Macro average and Weighted average of all the prediction

Furthermore, I used a confusion matrix to evaluate the performance of the classification model. Which provides a summary of a number of correct and incorrect predictions made by the model broken down by each class in the data set.

Later on, I tried to visualize and compare all the classes with their actual and predicted outcomes with heat maps.

Also, I need to find out which files are wrongly predicted. To understand more, I plot a graph between the error rate vs the k-value.



In conclusion, I have to find out the accuracy which I got around 67%.

Requirements Resources: -

The requirements and resources for this project on Music Instrument Recognition could include:

Hardware:

- A computer with a decent CPU, GPU, and RAM to run the machine learning algorithms.

Software:

- Python is a popular programming language for machine learning and signal processing. Development environments such as Google collab or Jupyter Notebook or PyCharm. Python packages like Librosa, Tensorflow, Keras, Seaborn, Scikit-learn, Pandas, Numpy, matplotlib, etc. for audio signal processing, machine learning, and data visualization.

Dataset:

- Which I have Gathered from Kaggle.com containing more than 300 audio files.

Machine Learning Models:

- K- Nearest Neighbors Algorithm (KNN), Convolutional Neural Networks (CNNs), Random Forests, and other machine learning models to train and test the audio classification models.

Evaluation:

- Evaluating the performance of the trained models such as accuracy, precision, recall, F1-score, and confusion matrix.

Documentation:

- Documentation tools to write and present the research findings and conclusions. Some popular tools include Microsoft Word, LaTeX, and Markdown.

Research Plan: -

I was able to achieve a precision of 73%, recall of 72%, and F1 score of 71% for our multi-class estimator in this project. I would have attempted more architectures, which include CNN, RNN that proved to be efficient with audio, or even RCNN or CRNN architectures if I had more time.

Furthermore, I would have washed the data and labeled the unlabeled instruments in the streams, or even I should have taken a large data set that has like 10000 audio files as well as added more data samples for the class like Harmonium, keyboard, etc. I would also have liked to support more instruments, and split the current classes into sub-classes (e.g. split piano into "keyboard" and "classical piano", etc.). This could be useful in instances where the model mixes up piano and flute.

Finally, broadening our target space by allowing the identification of multiple instruments in a stream would be a welcome advance over our current model.

In my future work, I am definitely going to build the model in CNN and RNN to find out which is giving more accurate models.

References: -

- Liu, L., Liu, J. and Lu, X., 2018. Audio classification using a hybrid feature selection method and KNN algorithm. Journal of Ambient Intelligence and Humanized Computing, 9(5), pp.1481-1491.
- Eichner, M., Wolff, M., Hoffmann, R. (2006). Instrument classification using hidden Markov models. system, 1(2), 3
- Eronen, A., Klapuri, A. (2000, June). Musical instrument recognition using cepstral coefficients and temporal features. In 2000 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings (Cat. No. 00CH37100) (Vol. 2, pp. II753-II756). IEEE.
- Kaggle (2018) Audio set https://www.kaggle.com/c/freesound-audio-tagging
- Github (2020) Wav files https://github.com/seth814/Audio-Classification