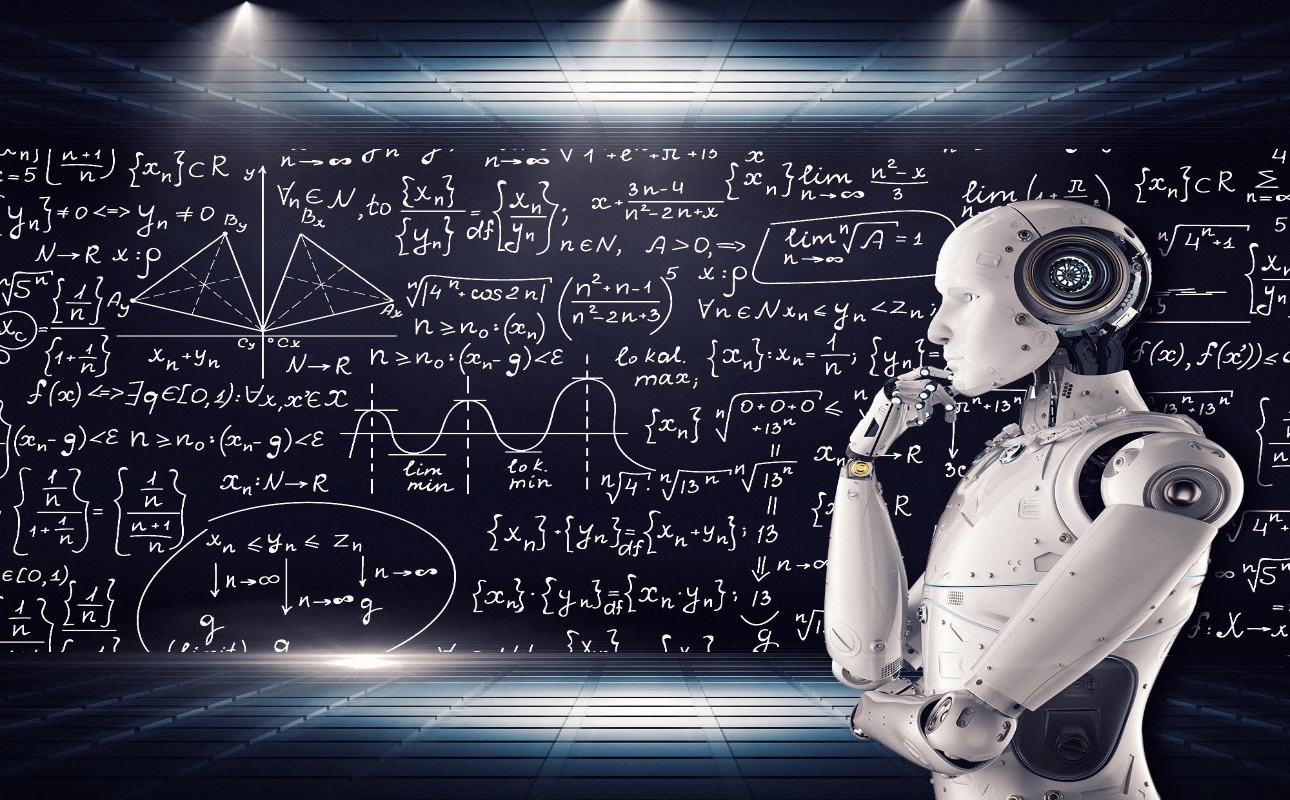
## P1#yIS1



AAI-511: Predicting Music Composers, with MIDI Files Dataset

July 2023

Team 8: Christopher Teli, Adam Graves, Ikenna Opurum

Github Link to Model: <https://github.com/cteliStolenFocus/aai-511-team-8/tree/main>

Table of Contents

[P12#yIS1 0](#_Toc140678807)

[Overview 1](#_Toc140678808)

[Goals/Strategize 1](#_Toc140678809)

[Design 2](#_Toc140678810)

[**- 1.0 Importing libraries** 2](#_Toc140678811)

[**- 1.1 Loading the Dataset** 2](#_Toc140678812)

[**- 1.2 Initial data summary check** 2](#_Toc140678813)

[**- 1.3 Optimize the dataframe** 2](#_Toc140678814)

[**- 1.4 Data cleaning and normalization** 3](#_Toc140678815)

# Overview

Using a dataset of multiple composer will contain MIDI files and sheet music of compositions from well-known classical composers like Bach, Beethoven, Chopin, Mozart, Schubert, etc. The dataset should be labeled with the name of the composer for each score.

# Goals/Strategize

The primary objective of this project is to develop a deep learning model that can predict the composer of a given musical score accurately.

The project aims to accomplish this objective by using two deep learning techniques: Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN).

The strategy includes:

* Feature engineering of the dataset
* In turn this will determine the final dataset for extraction and loading into the dataframe
* Building a suitable model
* With some research and development we will identify the best structure for the LSTM model, the best Optimizer, and how best to evaluate

# Design

1. **Data Importing and Pre-processing**

The process of the Extract, Transform, Load (ETL) Step one is to create a pickle file that will be used to load into the dataframe. This approach effectively distributes the workload of model building and feature extraction. In case the model performs poorly with the current data, a team member can execute feature extraction, generate a dataframe, and save it as a pickle file. This binary file can then be shared with the rest of the team, allowing them to utilize the preprocessed data without duplicating efforts.

Fields of the dataset:

columns**=**["Composer","Times", "Pitch", "Note\_Density", "Volume",

"Rhythmic\_Complexity", "Tempo"

## **- 1.0 Importing libraries**

Importing the necessary libraries to read the dataset:

**import** os

**import** glob

**import** pretty\_midi

**import** numpy **as** np

**import** matplotlib.pyplot **as** plt

**import** pandas **as** pd

## **- 1.1 Loading the Dataset**

Load the pickle file

* 'team8\_composer\_dataset.pkl'

## **- 1.2 Initial data summary check**

Review data field attributes and data types, ensure proper import and verify the field types

## **- 1.3 Optimize the dataframe**

Normalization of the data by removing unnecessary columns and replacing missing values

## **- 1.4 Data cleaning and normalization**

Normalization of the data by removing unnecessary columns and replacing missing values

**Summary**:

* The code defines a function called calculate\_features(midi\_file) that loads a MIDI file, extracts various musical features like pitch, note density, volume, rhythmic complexity, and tempo. The extracted features are returned as numpy arrays.
* The code defines another function called process\_composer\_data() that iterates over directories containing MIDI files for different composers. It uses the calculate\_features() function to extract features from each MIDI file and appends the data into a pandas DataFrame.
* The DataFrame is then saved to a pickle file for future use. If the pickle file already exists, the code loads the data from the pickle file instead of reprocessing the MIDI files.
* The code checks if the pickle file exists before processing the data. If the pickle file exists, it loads the data from the file, and if not, it calls the process\_composer\_data() function to create the dataset.

1. **Building the LSTM Model**