Baroque to Modern

An ML Approach to Predict Historical Periods for Classical Piano Music

IRON HACK

Mark Morcos

Introduction

What it's about
Using machine learning to predict the historical period of classical piano performances (from baroque to modern).

Why this topic
Combining classical music with data analysis and machine learning.

Project goal
Support educators and students in analysing and understanding music from different eras.

Dataset

- 1276 piano performances
- ~200 hours of recordings
- High-precision MIDI files
- Metadata with composition info

Field	Description
composer	Composer of the piece, standardised
title	Title of the piece, not standardised
split	Suggested train/validation/test split
year	Year of performance
midi_filename	MIDI filename
audio_filename	WAV filename
duration	Duration in seconds

Curtis Hawthorne, Andriy Stasyuk, Adam Roberts, Ian Simon, Cheng-Zhi Anna Huang, Sander Dieleman, Erich Elsen, Jesse Engel, and Douglas Eck. "Enabling Factorized Piano Music Modeling and Generation with the MAESTRO Dataset." In International Conference on Learning Representations, 2019.

Feature Engineering

Features were extracted from MIDI information:

- Pitch data: Note frequency.
- Rhythm data: Tempo and duration of notes.
- Dynamics data: Intensity of notes.

Feature Engineering

Pitch: Histogram, range
Capture tonal structure and range of pieces, differentiating musical styles.

Rhythm: Tempo, average note duration
Distinguishing between fast-paced Baroque pieces and slower Romantic compositions.

Dynamics: Average velocity, velocity range
Reflects expressiveness and emotional intensity, key traits of different periods.

Feature Engineering

Challenges
- Missing historical periods for the target column.

Approach
- Web scraping LastFM and AllMusic (respecting robots.txt).
- Retrieved and sanitised data to standardise categories.
- Spotify's audio features endpoint decommissioned.

Outcome
- Dataset size reduced by 50% after target population.
- Romantic period bias due to popular composers like Chopin, Schumann, and Liszt.

Model Training

<u>K-Nearest Neighbours:</u>
Linear search for the best value of K using cross-validation.

<u>Random Forest:</u>
Optimised number of trees using Grid Search.

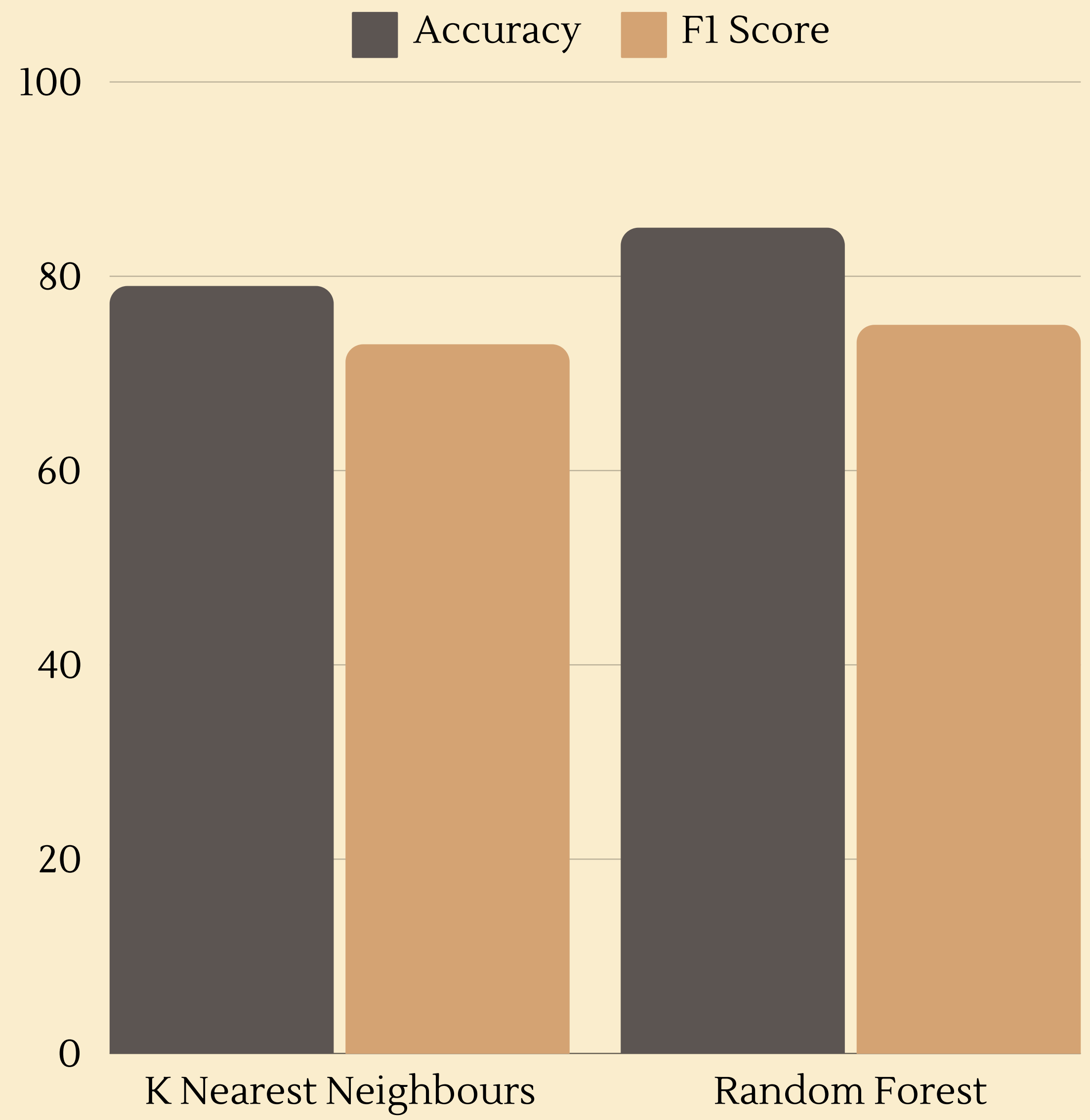
Both models saved using the <u>joblib</u> library for later use.

Results

<u>K-Nearest Neighbours:</u> Best accuracy achieved with K=77.

<u>Random Forest:</u> Best accuracy achieved with 300 trees, no depth limit and 5 as min split.

Random Forest outperformed KNN in terms of both accuracy (85% vs. 79%) and F1 score (75% vs. 73%).



Conclusion

- Achieved relatively high accuracy for predicting historical periods.
- Random Forest outperformed KNN, making it the final model choice.
- Chosen features were useful to classify pieces into their respective era.

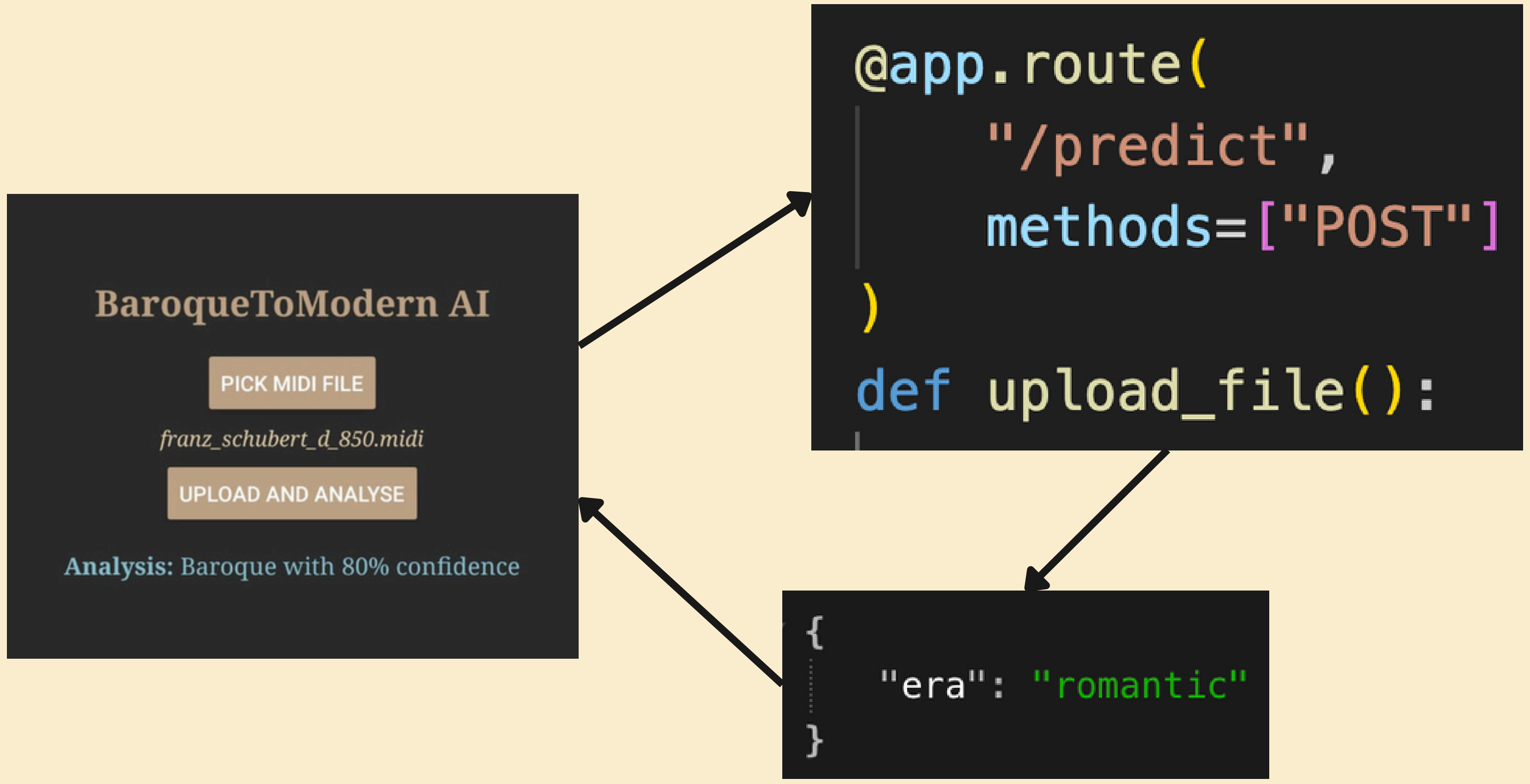
Proof of Concept

Python Server

- POST /predict endpoint:
  - Input: MIDI file
  - Output: Historical era

Mobile App

- Single screen to pick, upload and analyse a MIDI file
- Fetch JSON response from Python server and parse it



Future Work

Key, chord and cadence analysis as extra features

Test advanced ML models like Neural Networks.

Expand dataset with more compositions and eras.

Thank You