Louay Farah

Genetic Algorithms for Music Generation - Report

18 July 2023

Abstract

This report presents the work conducted for the music generation project using a Genetic Algorithm (GA). The aim of the project was to generate melodies and chord accompaniments that fit a specific chord progression. The implementation used the music21 library in Python to create MIDI files based on user input for the scale tonic and mode. The GA followed a standard procedure, including initialization, evaluation, selection (using a roulette wheel in my case), crossover, mutation, replacement, and termination. The generated melodies were then used to build a score with melody and accompaniment parts. This report provides a comprehensive overview of the method, implementation details, results, and evaluation of the project.

Introduction

Music generation using AI algorithms has gained significant attention in recent years. Genetic Algorithms, inspired by natural evolution, have shown promise in generating musical compositions. This project aimed to employ a Genetic Algorithm to generate melodies and chord accompaniments that harmonize with a specific chord progression. The project used the music21 library, which provides powerful tools for music analysis, manipulation, and composition.

Method

The Genetic Algorithm approach was chosen due to its ability to iteratively evolve a population of melodies toward the desired fitness criteria. The project focused on generating melodies that fit well with the 1 5 6 4 chord progression. I represented the melodies as sequences of notes with unified durations, with a length of 200 notes. I executed the GA for a population size of 100 melodies over 1000 generations. The selection process used the roulette wheel method, where the fittest melodies were selected as parents for the next generation.

Implementation

The implementation of the project involved Python programming language and the music21 library. The music21 library provided essential functionality for working with musical elements such as scales, chords, and pitches. The code started by taking user input for the scale tonic and mode, which determined the musical context for the generated melodies. The scale, chords, and pitches were created based on this input. The GA was then implemented with functions for initialization, evaluation, selection with a roulette wheel, crossover, mutation with a 0.1 mutation rate, and replacement. I evaluated the melodies based on their fitness to the 1 5 6 4 chord progression. The fittest melodies were selected as parents for the next generation, and crossover and mutation operations were applied to create new variations. The new melodies replaced some of the least fit melodies in the population. This process was iterated for 1000 generations. Finally, the best melody got from the GA was used to build a score with melody and accompaniment parts.

Results and Evaluation

I evaluated the generated MIDI files based on their fitness to the 1 5 6 4 chord progression. The evaluation focused on assessing how well the melodies aligned with the chord progression. The fitness function assigned a score to each melody, with higher scores showing better fitness. The results showcased the algorithm's ability to produce melodies that harmonized with the chord progression. The evaluation process revealed the GA was able to generate melodies that exhibited a strong alignment with the chord progression. However, further evaluation and analysis are needed to assess the overall quality and diversity of the generated melodies.

Conclusion

In conclusion, this project successfully applied a Genetic Algorithm to generate melodies and chord accompaniments that fit a specific chord progression. The implementation used the music21 library and followed the standard steps of a GA, including initialization, evaluation, selection, crossover, mutation, replacement, and termination. The generated melodies showed an understanding of the chord progression and exhibited harmonious characteristics. The project's results provide a foundation for further exploration and refinement of the GA-based music generation approach. Future work may involve fine-tuning the algorithm parameters, enhancing the diversity of generated melodies, and incorporating additional fitness criteria to create more expressive and diverse musical compositions.