Machine Learning to Study Patterns in Chess Games

Student Number: 690065435

Academic Year 2022/2023

Abstract

Abstract here.

I certify that all material in this report which is not my own work has been identified.

Signature: _____

1 Introduction

2 Project Specification

3 Design

3.1 Downloading the Data

Collecting the data correctly was paramount to the success of my project, and it was important to use a large sample size to ensure that our insights represent the general population of chess games. I used the Lichess Open Database [CITATION HERE] of standard rated games for my data source – they upload tens of millions of games every month in PGN format, and they are easily accessible to the public. I decided to focus on games in 2022, as this enables me to capture the latest trends in chess. I will only be analysing Rated Bullet, Rated Blitz, and Rated Rapid games – I will not be including Rated Classical games, as they have a significantly smaller sample size and the longer time controls result in a vastly different playstyle.

I started by downloading the PGN files for each month, which introduced difficulties with big data. Each month's file is around 30 GB to download and they are compressed, which means each month is around 210 GB when uncompressed, resulting in approximately 2.5 TB of data for the year. My laptop only had 1 TB of storage, and the sheer size of the data would make it unrealistic to process in a reasonable amount of time. Therefore, I aimed to collect a sample of approximately 5 million chess games from each month, so I took 6 million games from each month as this included Rated Classical games that I would later filter out.

3.2 Data Processing

PGN (Portable Game Notation) files are a standard format for recording chess games – each game has headers containing information like the white player, black player, their ELO ratings, and the result of the game. In addition, it stores a list of all the moves in each game. While PGN files are well-structured, they are not easy to process. The pgn2data Python library [CITATION HERE] provides a simple way to convert PGN files into CSV files, but it exported both the moves and the metadata of each game in separate CSV files. I did not need the moves, so it was using up unnecessary space and taking a long time to process each game. It was also designed for generic PGN files so it did not include additional useful metadata provided by Lichess such as the opening of the game.

Thus, I wrote my own Python script to convert each PGN file into a single CSV file containing only the metadata of each game. I used the python-chess library to parse the PGN files and iterate over each game. I skipped Rated Classical games and games that involved a non-human player. To optimise performance, I split each month's PGN into six files each containing 1 million games with pgn-extract (a PGN manipulator written in C) [CITATION HERE], and I used Python's multiprocessing library to process each month in parallel – this reduced the processing time for each month from 120 minutes to 20 minutes.

Figure 1: Data Pipeline Diagram for the Project
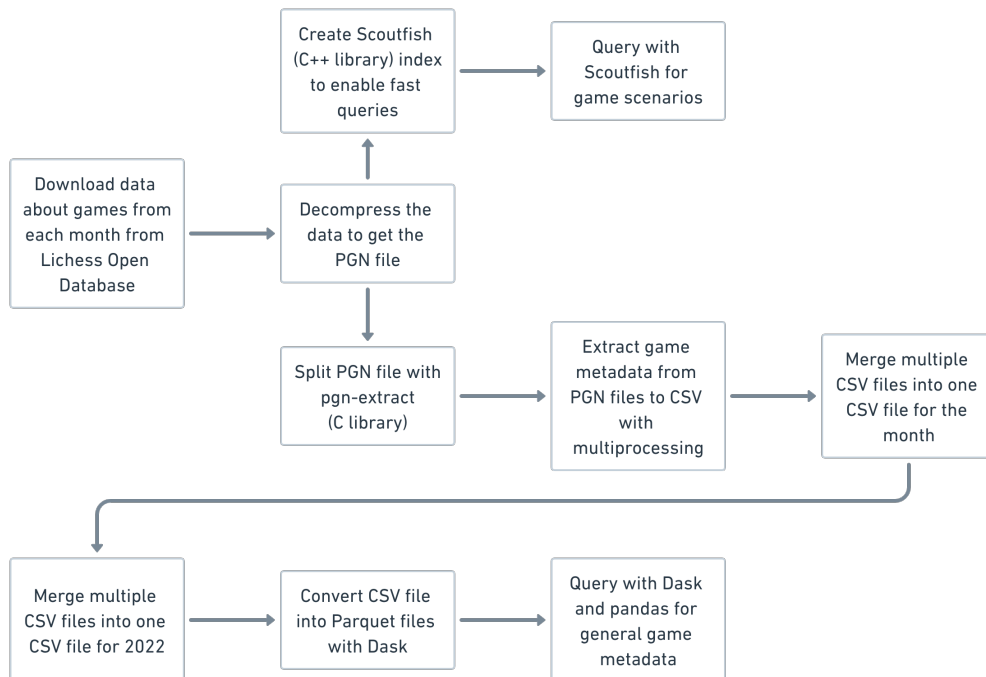
4 Development

5 Testing

6 Description of the Final Project

7 Evaluation of the Final Project

8 Critical Assessment of the Project

9 Conclusion

References