Progress Report

Daniel Cloutier Sagar Patel

October 22, 2020

1 Problem Statement

In this era, there is a massive amount of data collected in a large variety of different topics, of which lots of information can be extracted. The issue here comes with the fact that we cannot simply extract the information by visual inspection. There is currently too much data.

One such domain that contains a large amount of data is the game of chess. Many players like to study chess games to try and improve their play, but it's very hard to find games that are relevant or interesting to learn from amidst the plethora of games to choose from. Although there are many famous known games such as the Opera game of Paul Morphy vs Duke Karl or game 6 from the 1972 world championship match between Robert Fischer vs Boris Spassky [4], these are usually kept as beginner studies. Also, the fact that they are already well known doesn't necessarily help us discover anything new.

On top of these numerous known games, surely there are many more games that you could learn from. In fact, the website lichess.org logged 68,027,862 games at all levels of play in September of 2020 alone [5]. Some of these could be interesting to study, irrespective of the players ratings not being at the level of Grandmaster. All of this to say, there must be support from algorithms or machine learning techniques to be able to group similar games of interest together, or show games that lie far outside the norm, which could be interesting studies.

2 Problem Analysis

In order to tackle this problem, we must try and quantify what makes a game more or less similar to others and perhaps what other parameters might be used to determine whether a game would be worth studying or not. Luckily for us, all moves in a chess game can be represented in a general format called *Portable Game Notation* (PGN), which displays moves in a human readable ASCII format using *Standard Algebraic Notation* (SAN). This PGN contains information such as the date, time control, player names, player ratings as well as of course, the moves played during the game in SAN. SAN is used to describe which piece is moved ([Q]ueen, [R]ook, [K]ing, k[N]ight, [B]ishop, [P]awn/nothing) and to which square it is moved, using a system of a-h for horizontal position and 1-8 for vertical position, using the White player as the point of reference. We can also represent games using Forsyth-Edwards Notation (FEN) which describes a specific position in a chess game in one line of ASCII text [2].

For example, we could define a total "distance" of a game as some sum of the differences between two moves, defining the distance as some function of piece values, distance moved on the board and whether a piece was capture or not.

We must also take into consideration that the data we will be using will likely be the lichess games database, which contains games at all levels of play. This means some games will be of very poor quality while some will be of extremely high quality. In fact, some Grandmaster level players play chess online, or even stream their own games. This means that our data should be sorted, or at least quantified on more than one axis to more aptly determine whether a game that is an outlier is significantly more interesting than others, or is just a very poor quality game played by beginners.

Finally, we must also look at the clock times. Oftentimes, chess games are played with different time controls, varying anywhere between 1 minute for all your moves, up to the official International Chess Federation (FIDE - Fédération Internationale des Échecs) time control of "90 minutes for the first 40 moves of the game, followed by 30 minutes for the rest of the game with an additional 30 seconds added per move starting move 1." [3]. This also can affect the quality of games, as a game that lasts a mere 2 minutes will

likely be of far lesser quality than a game that lasted multiple hours.

3 Literature Review

Based on this problem analysis we were able to find related literature that addressed some of these problems. Namely, we found a distance function that classifies a move as a 9-dimensional vector [1]. Before we go into this however, they also define weights to each chess piece. Namely:

	K	Q	R	B	N	P	NULL
Weight	12	9	5	3	2	1	0

With the dimensions of the vector being:

1. x coordinate of the piece before move using SAN
2. y coordinate of the piece before move using SAN
3. x coordinate of the piece after move using SAN
4. y coordinate of the piece after move using SAN
5. x coordinate of the piece captured using SAN
6. y coordinate of the piece captured using SAN
7. weight of the piece before move
8. weight of the piece after move
9. weight of the piece captured

Unsupervised:

They used only 5000 GM games, this is good but we can do better. Lichess database has 1.5 Billion games available to us. Although they are at faster speeds. Standard FIDE clocks are 90+30, 30 minutes added at move 40. The slowest standard lichess game is 30+20. However, we can definitely filter our games because we have access to these.

The distance function. This is fundamentally good, but evaluating chess games on only one dimension to try and group or find outliers is not that telling. Why? Well:

1. Chess Openings (ECO Codes): Games with similar ECO codes will by default be similar. For example, ECO codes B20-B99 are all different variations of which the first moves are 1. e4 c5. This is relevant because if, for example, a game with opening B20 (Sicilian Defense) somehow has a similar distance than multiple games with code C53 (Giuoco Piano; 1. e4 e5 2. Nf3 Nc6 3. Bc4 Bc5 4. c3), that game could turn out to be interesting.

2. Player rating (ELO): With our data, games will be spread out over multiple different skill levels. Because of this, it is important to sort our games by skill level

3. Game Quality: It is possible to be able to sort our games by the "quality" of the game; that is we would define a function that looks at a move and determines whether it is good or bad. Either we can make our own, or we can use something existing. One thing to note is that we're changing the piece values so it is possible to at least attempt to make our own evaluation function if we want to.

4. Game Clock: Some games also have not only the total time for the game, but also have the amount of time a player took per move (starting April 2017 on Lichess). This information can also be used to further classify games. For example, you may not want to look at games in Bullet format (1 minute total to make all moves) if you're trying to improve at a slower time format. But if you want to improve in that format specifically, this might actually be something of value.

4 Subproblems, Statement and Analysis

Subproblem: Not all games have position evaluations, so we will possibly have to do our own position evaluations if we want to have that as a parameter for the unsupervised algorithm. That or we can scrap that idea as it isn't really necessary considering player ELO ratings will be already a good indicator of the quality of games.

5 Algorithmic Sketch, Illustration of Solution

Not sure what we would do here, but we can probably draw something pretty quick in paint (or figure out how to do graphs in LaTeX) as our sketch. It probably isn't that complex... Probably a famous last words moment though.

References

[1] J. A. Brown et al. "A Machine Learning Tool for Supporting Advanced Knowledge Discovery from Chess Game Data". In: *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*. 2017, pp. 649–654. DOI: 10.1109/ICMLA.2017.00-87.

[2] Steven J. Edwards. *Standard: Portable Game Notation Specification and Implementation Guide*. URL: https://ia802908.us.archive.org/26/items/pgn-standard-1994-03-12/PGN_standard_1994-03-12.txt.

[3] International Chess Federation. *FIDE Handbook*. URL: handbook.fide.com.

[4] Timothy Glenn Forney. *Best Chess Games of All Time*. URL: https://www.chessgames.com/perl/chesscollection?cid=1001601.

[5] *lichess.org game database*. URL: https://database.lichess.org/. (accessed: 22.10.2020).