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 [5], 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 [6]. 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 [3].

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." [4]. 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 [2]. 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

From here, they define a *similarity distance* function, $dist(f, g)$ between two moves f and g, described as the Euclidean distance between them in this 9-dimensional space [2]. Afterwards, we consider that the distance between two games is defined as:

$$dist(F, G) = \sum_{i=1}^{M} dist(F[i], G[i]) \tag{1}$$

where $F[i]$ and $G[i]$ are the i-th moves in games F and G, and M is the maximum number of moves between F and G. If one game has less moves than the other, a set of null moves is defined, and are appended to the shorter game so as to not lose generality [2]. Finally, after applying this to our games, we can find games that are similar and group them, as well as finding outlying games that could be interesting to look at.

There is, however, a setback. Namely, they used 500 games from 10 different Grandmaster level players for a total of 5000 games. We will have a completely different set of data, of which we can't confidently say they are all of high quality. On top of this, instead of grouping our games only along the axis of distance, we can also group them on different axes, seeing as we have a lot of extra data that the authors of this paper perhaps didn't have. There are multiple subproblems that we will have to address based on the context of our data.

4 Subproblems, Statement and Analysis

Right from the beginning of this project, we wanted to extract meaningful information from chess data in addition to implementing the research provided. Luckily we have access to over 1.5 billion records of chess games played by people all around the world [6]. We believe there is valuable information to learn if we can extract it effectively. We decided to look deeper into the point values assigned to each chess piece and the circumstances can affect these values.

During a chess match, each player can be assigned scores that are evaluated by various positional features - most importantly the number of pieces on each side and further the positions, centralization, and mobility of each piece. By adjusting the individual weight of specific pieces on the board it allows us to emphasize the difference in the importance of these pieces. The evaluation of the pieces can be changed due to many parameters such as pawns near the edges are worth less than those near the center, pawns close to promotion are worth far more, pieces controlling the center are worth more than average and trapped pieces are worthless, etc. In fact, a popular chess engine Stockfish changes piece values depending on the current state of the board [1]. Selecting effective weights for pieces allows us to estimate the player's positional advantage. For example, suppose we see a randomly chosen position in which White has a pawn advantage of 2 points (based on the previously displayed table on piece values). We could possibly assert a probability of close to 80% that the game would end in a win for White.

What is the correct weight for chess pieces? Unfortunately, there is no right answer. However, there are various algorithms that attempt to cover different strategies and positions. We intend to use these weighted values to find advantages that were not discovered by the researchers using other simple methods. More specifically, there is some research that was done on giving pieces specific values using regression analysis, using a base value of 100 for a pawn and adjusting the other pieces accordingly [8]. This can not only help with perhaps changing the similarity distance, but also can give us more pertinent information about which player has the advantage

Next, there is what is called ECO Codes, or Encyclopedia of Chess Openings. This classifies many different chess "openings", or starting moves into different classifications. Similar starting moves having similar classification codes [7]. This is important, especially in the context of distance between games, as similar chess openings will likely lead to an intrinsically similar similarity distance as defined in the previous section. Adding this as a dimension to our analysis can help look at games that have a similarity distance close to many games, but which started out very differently.

Third, we have some more context specific issues relating to our dataset. More notably, the fact that our games will not be strictly Grandmaster games. This means that in order to implement what was done in the paper "A Machine Learning System for Supporting Advanced Knowledge Discovery from Chess Game Data" with the similarity distance as defined above, we will likely have to add a dimension of player strength to avoid having low quality beginner games being flooded into the same category as higher quality top level games. This, or we can simply cut out players beneath a certain rating threshold, such as 2400 ELO, which is the FIDE definition of an International Master; only one level behind that of Grandmaster.

Finally, we have time controls. Recorded Grandmaster games are typically all played with the FIDE standard time control, however the games we will be using are all online games with a wide variety of time controls ranging from 1 minute bullet chess all the way up to 30 minute classical time controls. This drastic change in the allotted time to think about what moves to play will naturally change not only the way players make moves, but also the quality of their moves. This is something we will have to consider when creating our implementation.

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] URL: https://github.com/official-stockfish/Stockfish/blob/master/src/types.h.

[2] 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.

[3] 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.

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

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

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

[7] A Matanovic. *Encyclopedia of Chess Openings (five volumes)*. Belgrad: Chess Informant, 1974-1979.

[8] Vladimir Medvedev. *Point Value by Regression Analysis*. URL: https://www.chessprogramming.org/Point_Value_by_Regression_Analysis.