Can We Use Machine Learning To Predict Win/Loss Rates in Chess Using Meta-Game Data?

Nitish Joson Terance Joe Heston

Abstract

Artificial intelligence research in chess has traditionally centered on analyzing board positions and moves. While effective for engines, this perspective overlooks the wider context that shapes human play. Players face time pressure, emotional, and psychological influences beyond the board. This paper examines whether Machine Learning can predict win, loss, or draw outcomes using only meta-game data such as ELO differences, time management, and activity statistics. A Random Forest classifier trained on a large dataset achieved competitive accuracy, revealing correlations between contextual signals and results.

1. Introduction

Chess has long been a central domain for exploring the capabilities of artificial intelligence and machine learning. From early engines that relied on brute-force search to modern neural networks like AlphaZero, much of this research has concentrated on analyzing moves and board states. These approaches have proven highly successful for computers that aim to find the best move, but they overlook an equally important question: how can we model and predict the behavior of human players?

Unlike machines, humans do not play under ideal conditions. They are constrained by time pressure, skill differences, tournament standings, and psychological factors such as confidence or fatigue. These influences, while not visible on the board itself, shape the outcome of games in significant ways. Understanding them requires shifting focus from positions and moves to the broader context of play.

This paper investigates that shift by asking:*Can we use Machine Learning to predict win/loss rates in chess using meta-game data?* By examining features such as time usage, ELO differences, and other non-board variables, the goal is to evaluate whether contextual information about the game environment can meaningfully contribute to predicting outcomes. In doing so, this work contributes to a more human-centered understanding of chess performance, complementing the board-focused approaches that dominate current research.

2. Background

This paper contributes to the area of how technology can influence chess. To understand how computers have influenced chess, it helps to trace the game's history alongside technological change. Looking at the sequence of milestones makes it easier to compare how each stage reshaped the way chess was studied and played.

2.1 Before Chess Engines Era

Before chess engines, the main way of learning chess was through following the footsteps of masters. Books and annotated games communicated insights that masters had gained through experience. These players often spent hours analyzing even a single position, and their love for the game stemmed from the highly romanticized view of chess in that era. Still, without computers, players relied more on intuition than on calculation, since thorough analysis was time-consuming. For example, Capablanca emphasized principled play over calculating lines before play. Because of the limits of human analysis, certain openings, games, and positions were debated for decades with no clear resolution. Much of the literature was written by great masters, but even they were not immune to mistakes in their evaluations.

2.2 After Chess Engines Era

The arrival of chess engines completely changed how players study the game. Engines can calculate millions of moves every second, which means they can quickly point out the best options in almost any position. This saved players the long hours they used to spend working out moves on their own and made preparing openings and reviewing games much faster. Engines also settled arguments that had lasted for decades by showing which moves were truly strong and which older ideas were actually mistakes. As a result, the overall level of play improved, and players at all levels gained access to tools that once would have taken years of experience to develop. That being said, previous methods of following the footsteps of masters were still used by many players. Chess engines offered calculation, but did not offer any help with intuition.

2.3 Human-like Chess AI

The most recent step in this journey has been the rise of human-like chess AIs such as AlphaZero and Leela Chess Zero. These systems learn patterns and strategies by playing millions of games against themselves. As a result, they demonstrate a style that feels creative and intuitive, sometimes even surprising grandmasters with moves that seem unusual but later prove effective. Furthermore, an ongoing project called MIAIA chess seeks to achieve human play as closely as possible. This allows the computer to understand what moves seem appealing to humans and can show how to avoid common blunders and mistakes. These advances have brought engines closer to human-like thinking meaning that they can better help humans improve their chess, but they still differ in one key way: they do not share the same limitations that humans face. A machine never gets nervous before a critical move, runs low on energy after a long tournament, or gets demotivated after losing a game.

3. Related Works

As mentioned before, in machine learning studies of chess, most prior work has focused on board-centric features. Researchers have trained large neural networks on millions of games to classify positions as winning, drawing, or losing, often by approximating engine evaluations through supervised learning [6]. Reinforcement learning approaches, most notably AlphaZero and Leela Chess Zero, have shown that models can achieve superhuman strength through self-play, learning strategies without human heuristics [5, 7]. While these methods represent breakthroughs in AI, they remain tied to evaluating positions and move sequences on the board.

By contrast, much less research has explored meta-game data as a lens for prediction. Studies have shown that time pressure increases error rates and blunders, showing the role of temporal dynamics in human play [1]. Similarly, ELO rating differences have long been recognized as reliable predictors of performance [2]. Psychometric analyses have also linked expertise to differences in memory, pattern recognition, and decision-making speed [8]. More recent work has explored the role of emotions in chess problem solving [3] and the impact of competitive stress on cognitive performance, including factors such as momentum, fatigue, and psychological pressure [4].

This body of work suggests that while board-focused methods dominate AI research, meta-game features offer complementary insights into human performance. By extending models beyond the board to include these meta features, future research can develop more human-centered predictive frameworks that better reflect the realities of competitive play.

4. Dataset

The data used in this project is sourced from lichess.org open database, which provides chess game records in PGN (Portable Game Notation) format. PGN is a widely accepted standard for storing and analyzing chess games. Lichess collects PGN data from two main sources:

1. Games played directly on the Lichess platform

2. Games broadcasted by Lichess from major tournaments and events

For this project, I chose to use three datasets from the official broadcast games dataset (July - 2025, June - 2025, and May - 2025), which is released under the *Creative Commons Attribution-ShareAlike 4.0* license. Although this dataset is significantly smaller than the full archive of all user-played games on Lichess, it provides three key advantages:

- Game quality: Broadcast games typically feature strong players, including titled and professional competitors, which improves consistency and quality. More formal games also carry more weight to players compared to casual games played online.

- Annotations: These PGNs often include clock times and engine evaluations, whereas user-played games rarely do.

- Size: This project was conducted on a personal laptop with limited disk space and processing power. Handling the full Lichess archive would have been resource-intensive without providing proportionally valuable insights. The broadcast dataset is compact enough to be processed efficiently on local hardware while still offering high-quality data for analysis.

4.1 Definitions Related to the Dataset

This list gives a standard breakdown of terminology that will be referred to in the code and in the paper:

- Time Controls and Clocks: Refers to the total time each player has to complete the game (e.g., 3 + 0 means 3 minutes with no increment). Time usage per move is logged into digital formats like PGN, typically using time-stamp notation (e.g., [%clk 00:03:17]). The data is cumulative, and time spent per move is calculated by taking the difference between successive clock values.
- Player Ratings: Chess platforms assign ratings (e.g., 1000 or 2800) based on performance using ELO or Glicko rating systems. The higher the rating, the better the player. A player's rating difference with their opponent is a key contextual factor when evaluating other features.
- Game Phases: Chess is broadly divided into the opening, middlegame, and endgame. Each phase has its own principles, techniques, theory, and style of play.
- Behavioral Indicators from Time: From move-by-move data, several behavioral metrics can be computed, such as average time per move, standard deviation of time, and the number of peaks in time usage (moves where they spent a significant amount of time).

5. Methodology

This section includes the main features of the model and the architecture of the project in the form of the pipeline.

5.1 Features

The features I have used for this include time, elo, and some basic overall position data not specific to any player (this way we gain context into the type of game played without showing who is better). Specifically, I used the average time spent by each player (both as a ratio of starting time and also as a number), the standard deviation of time used, the elo of both the black and white players, the maximum time used by both players, The amount of moves the players peaked in time usage, number of captures total from both players (not separated by player, used to signify more of what phase ended on), how many moves the game lasted for, number of checks, number of promotions, and whether both players had decided to castle.

5.2 Pipeline

We begin by downloading the lichess.org database. The data will be in PGN format which is not ideal for the information we will be extracting. Thus, we run a script to turn the PGN file into a JSON file, with only the data which is necessary from the PGN data. We get rid of all move centric data, keeping the features aforementioned. Further, we get rid of invalid data that may be errors in the dataset. Still, this data is very broad, so it is then handed to another script called the Feature Extractor which takes the data and splits it for the analysis and the models. To see a visual representation of the process, refer to figure 1. Each JSON file the feature extractor creates is specifically tailored towards its goal.

The goal of each section is different. The analysis is meant to look at general statistics of the position and the dataset and see whether it can be used to predict the game's results. One part of the analysis looks at behavioral characteristics of time, and the other visualizes chess games in a graph. The goal for the models is to prove that AI and other models can use meta-game data to improve or predict human play. This data then goes to a Random Forest Classifier, which takes the features as input and outputs the likely result of the game. The models themselves take 20% of the data as training data and take 80% as testing data. Different variations of the models have been created to judge feature importance in the machine.

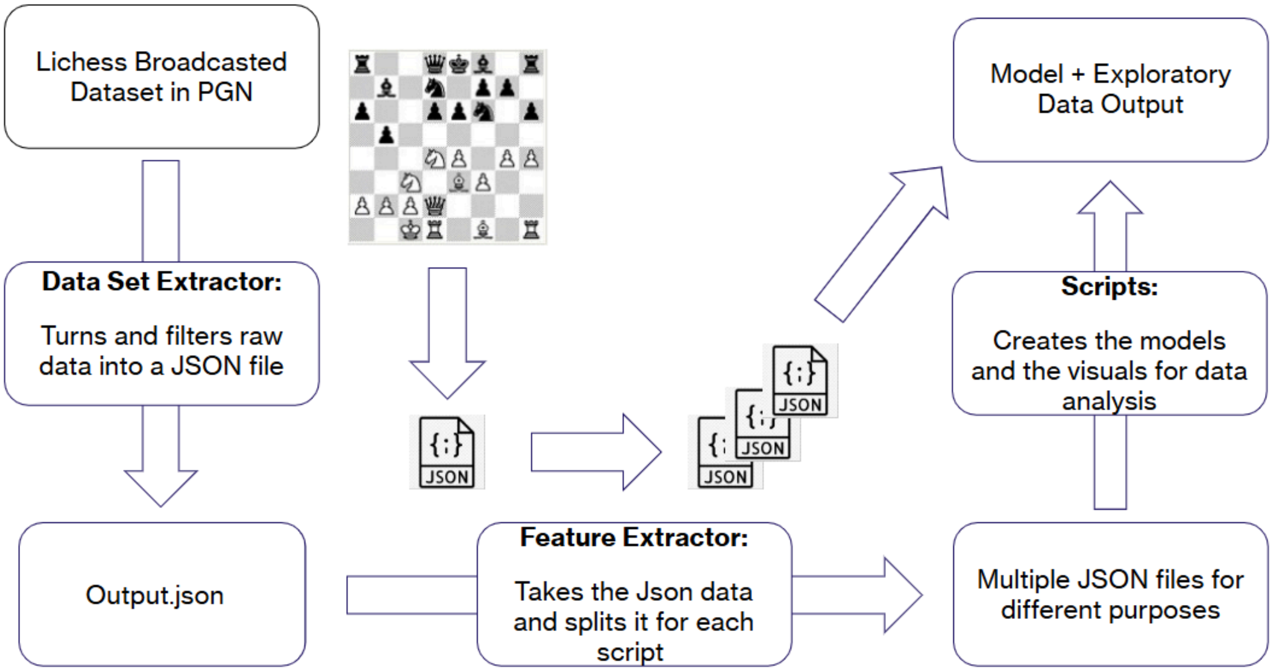

Lichess Broadcasted Dataset in PGN
Data Set Extractor:
Turns and filters raw data into a JSON file
Output.json
JSON
JSON
JSON
JSON
Feature Extractor:
Takes the Json data and splits it for each script
Multiple JSON files for different purposes
Scripts:
Creates the models and the visuals for data analysis
Model + Exploratory Data Output

Figure 1: Pipeline Diagram

6. Evaluation

This section shows the insights gained in exploratory data analysis and how it was used in the model. All implementation details and scripts used in this study are available at my github repository: AstroAtomic/Chess-Research-Paper. It is still necessary to download the datasets as Github doesn't allow large file uploads, see the Read Me for more details.

6.1 Exploratory Data Analysis

The first bit of analysis I did was checking the average time between winners and losers for the first 30 moves. Referring to Figure 2, the y-axis represents the percent amount of time that is used on average, and the x-axis is the move-number. We can see that during a chess game, the first few moves are typically quick between moves 1-5. Moves 5 to 25, typically where most middle games occur, show an increase in time-usage. Then as it approaches move 30 and beyond, the time usage shrinks. We can also see that on average, losers spend slightly more time than winners.

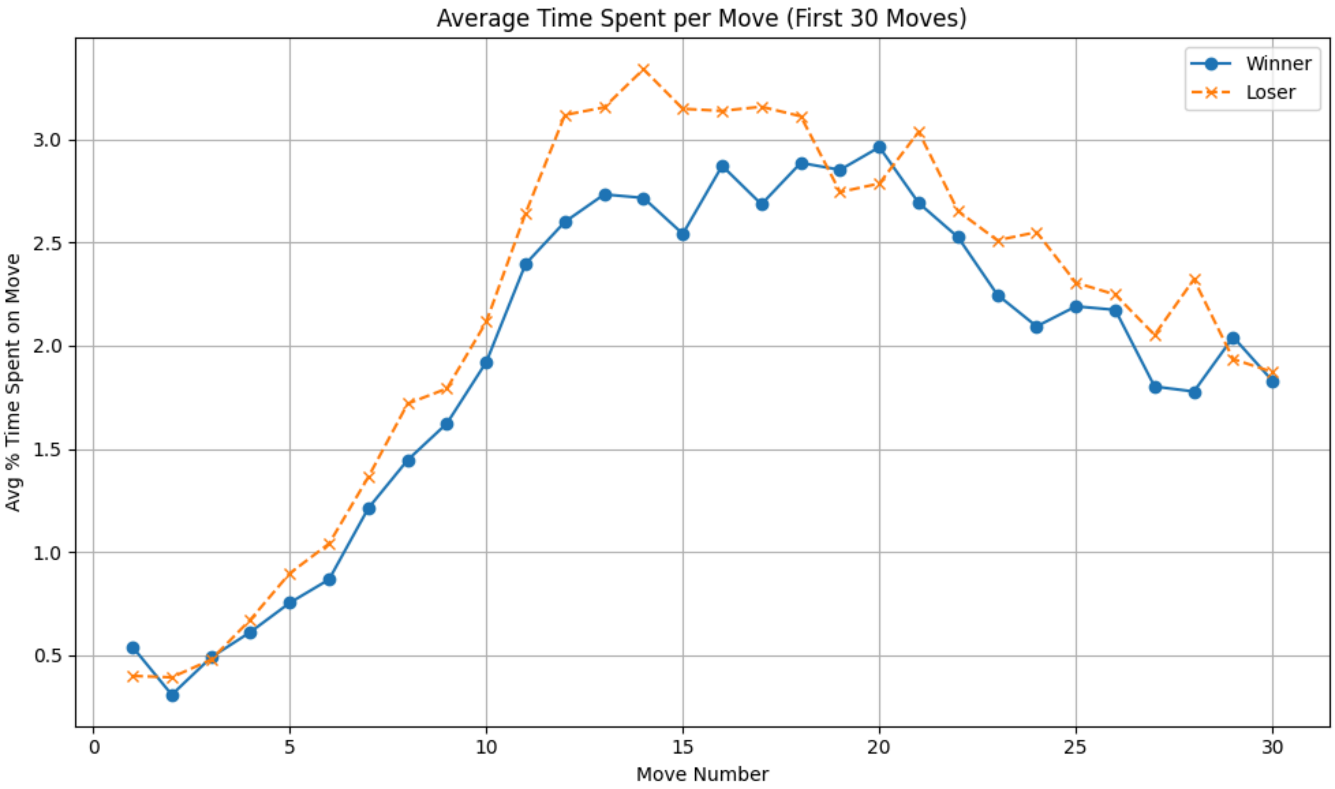


Figure 2: average time diagram between winners and losers per move

The next step in analysis that was done was to check visually if there are any correlation between the features that were chosen. Referring to the heat maps in Figures 3, 4, and 5, we can see that there are stark visual differences when white wins, draws, and losses based on the behavioral time data. The difference in color distribution between each diagram shows that there are differences in correlation depending on the result. One example is standard deviation and average of time (irrespective of color). When the result is a draw, the color shown between them is blue (meaning when one increases the other typically decreases). When the result is a win for white, the color is more neutral/white (no correlation), and when the result is a loss for white, it shows a more red color (meaning when one increases, the other increases). This can confirm that time meta data can be used to predict the outcome of games.

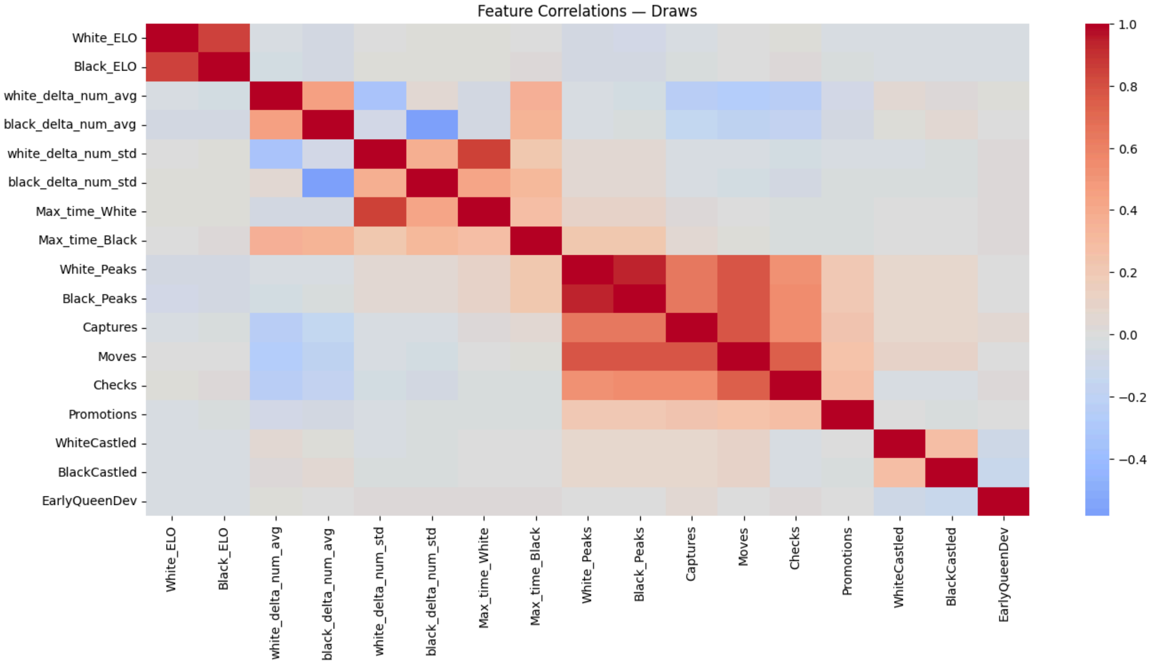


Figure 3: Feature Relationship Heatmap - White Draws

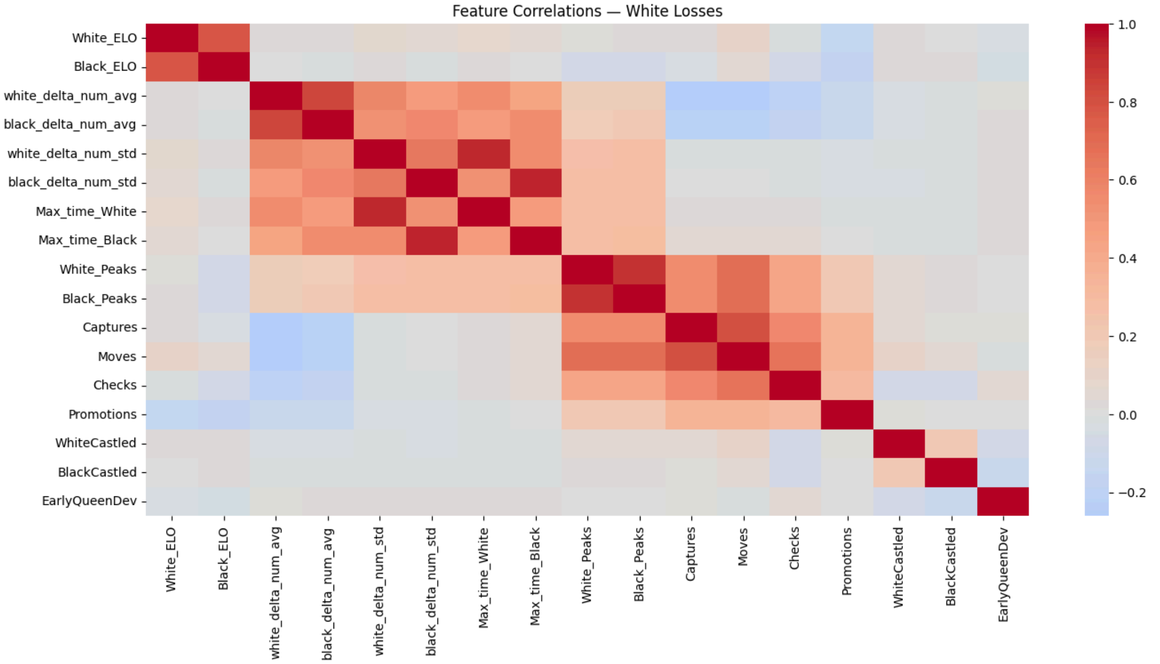


Figure 4: Feature Relationship Heatmap - White Losses

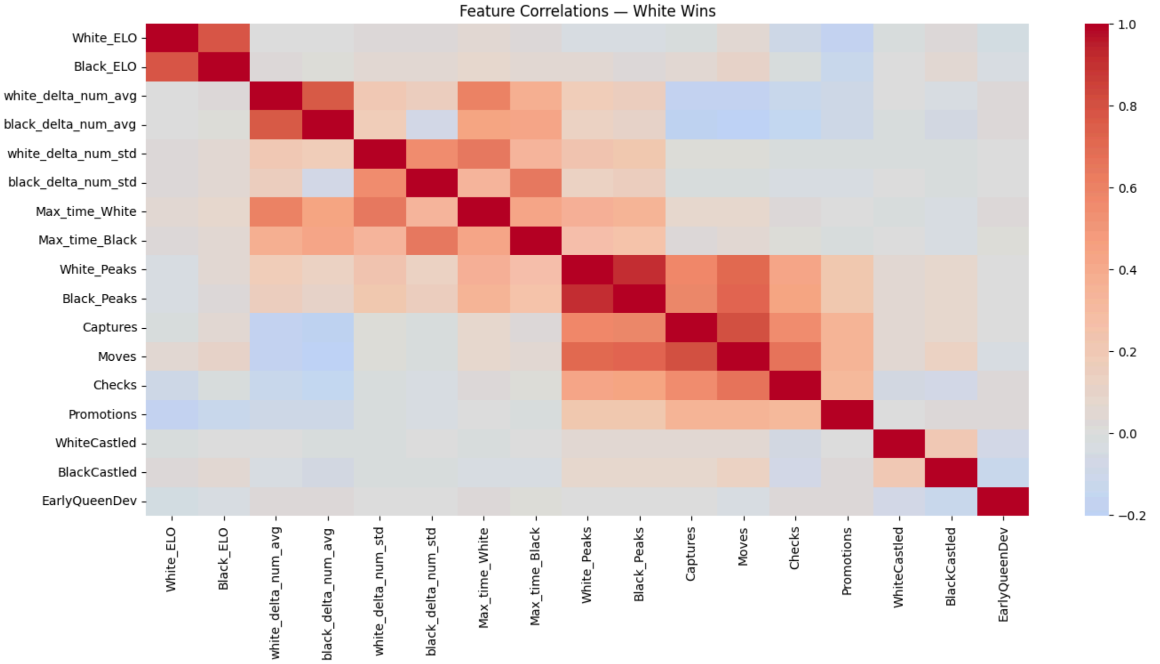
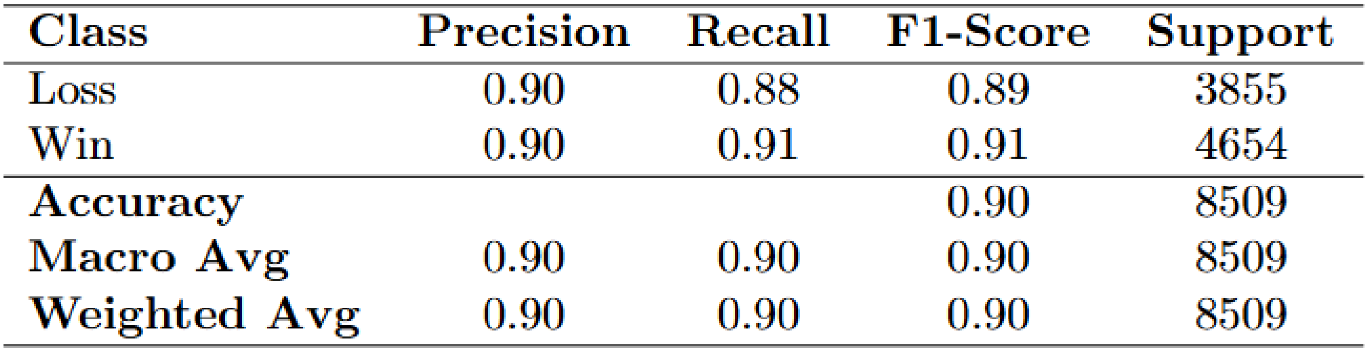


Figure 5: Feature Relationship Heatmap - White Wins

6.2 Model performance evaluation

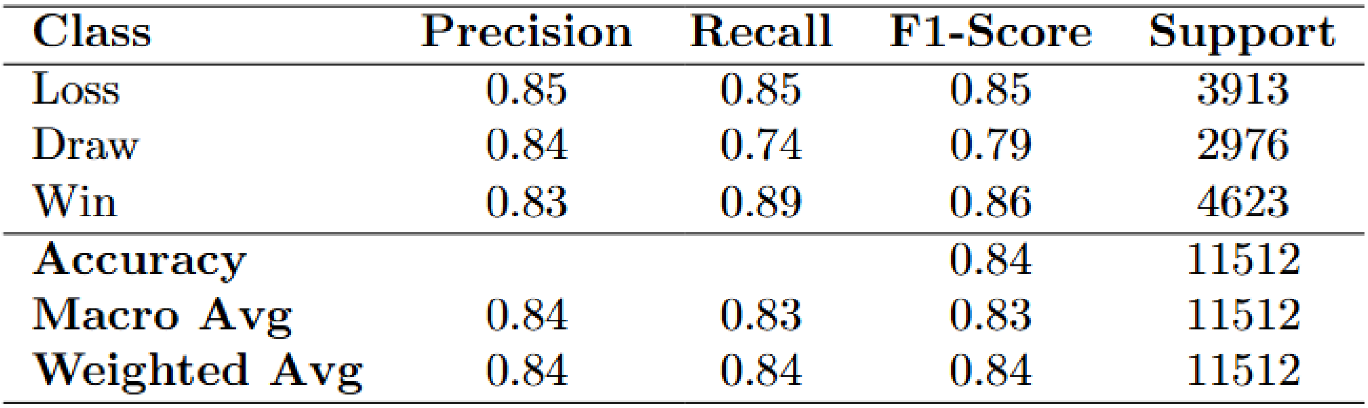
Now that we had sufficient evidence, it was time to build the model. For this, I chose a Random Forest algorithm. I developed two versions of the model: one that classifies wins, draws, and losses, and another that focuses only on wins and losses.

Starting with the win–loss version, White recorded a total of 23,255 wins and 19,286 losses. As shown in Table 1 and Table 2, this model achieved up to 90% accuracy. Other performance metrics such as the F1-score and recall were closely aligned, differing by only 1–2%.
The version that included wins, draws, and losses reached an overall accuracy of 84%. For this model, the F1-score and recall followed within a margin of 3–10%.



Class	Precision	Recall	F1-Score	Support
Loss	0.90	0.88	0.89	3855
Win	0.90	0.91	0.91	4654
Accuracy			0.90	8509
Macro Avg	0.90	0.90	0.90	8509
Weighted Avg	0.90	0.90	0.90	8509

Table 1: Classification report for chess outcomes (Win vs Loss)



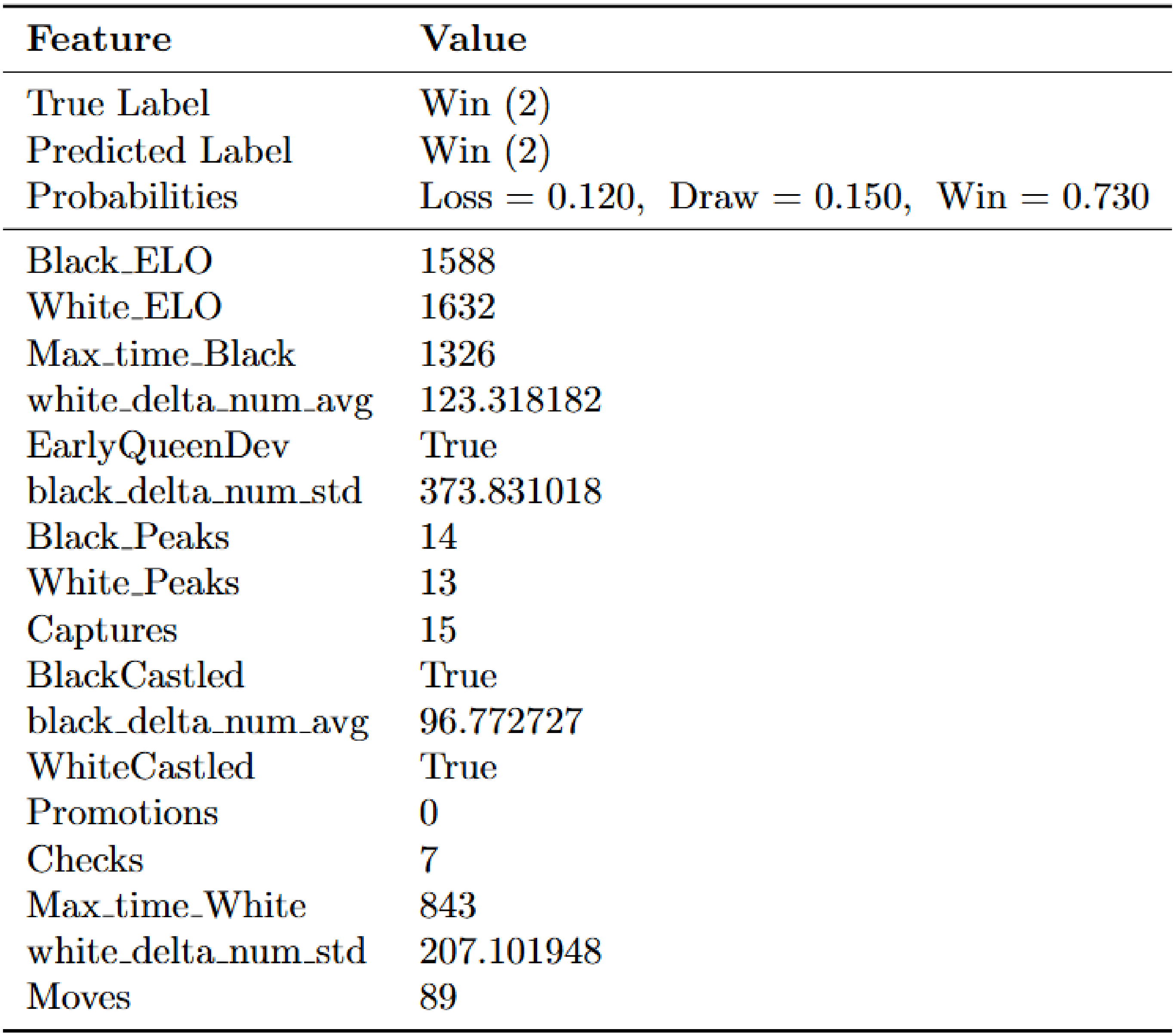
Class	Precision	Recall	F1-Score	Support
Loss	0.85	0.85	0.85	3913
Draw	0.84	0.74	0.79	2976
Win	0.83	0.89	0.86	4623
Accuracy			0.84	11512
Macro Avg	0.84	0.83	0.83	11512
Weighted Avg	0.84	0.84	0.84	11512

Table 2: Classification report for chess outcomes (Loss, Draw, Win). Overall accuracy = 0.84

6.3 Model Prediction Examples

Referring to Table 3, the model correctly predicted a game that ended in a win. The probability distribution strongly favored the Win outcome (0.73), with Loss and Draw receiving much lower confidence scores (0.12 and 0.15, respectively). Several features contributed to this accurate prediction: the White player held a higher ELO rating (1632 vs. 1588), both sides showed balanced tactical peaks, and the game included aggressive signals such as early queen development, multiple checks, and a high number of captures. It also shows that white has spent more time than black (the average) and has more consistent time usage than black (the standard deviation), suggesting white was more calm and calculative during the game. Together, these factors suggested a decisive result, and the model aligned with the true outcome.

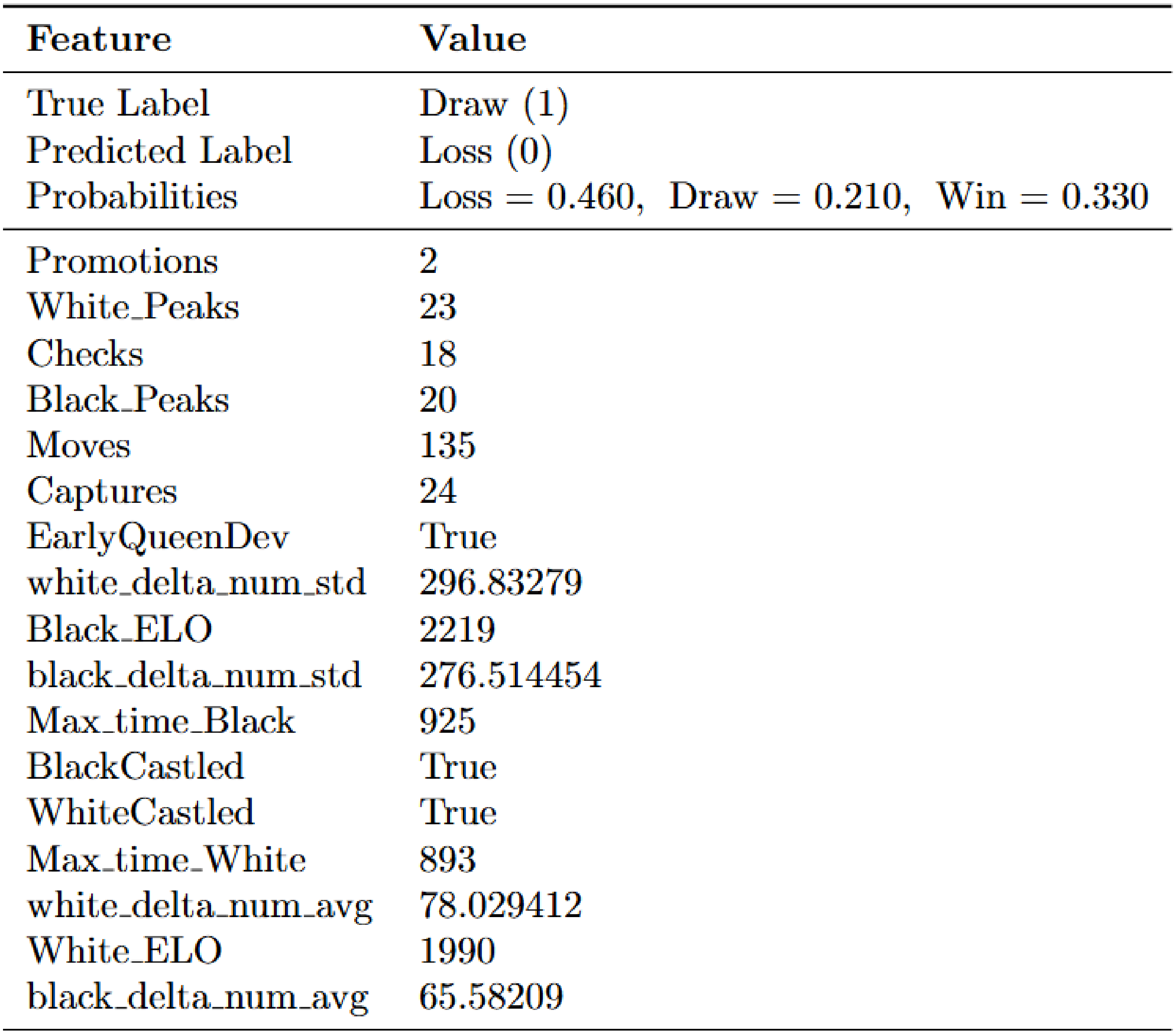
Table 3: Example (Correct Prediction) -- Index: 55526



Feature	Value
True Label	Win (2)
Predicted Label	Win (2)
Probabilities	Loss = 0.120, Draw = 0.150, Win = 0.730
Black_ELO	1588
White_ELO	1632
Max_time_Black	1326
white_delta_num_avg	123.318182
EarlyQueenDev	True
black_delta_num_std	373.831018
Black_Peaks	14
White_Peaks	13
Captures	15
BlackCastled	True
black_delta_num_avg	96.772727
WhiteCastled	True
Promotions	0
Checks	7
Max_time_White	843
white_delta_num_std	207.101948
Moves	89

In contrast, Table 4 shows an incorrectly classified example where the true result was a Draw, but the model predicted a Loss with the highest probability (0.46). Here, the game's long length (135 moves), high number of captures and checks, and even the occurrence of two promotions created the appearance of a decisive outcome. Additionally, the stronger rating of the Black player may have biased the model toward expecting a win or loss rather than a draw. This highlights a common limitation: the model tends to associate highly active games with big rating differences with decisive results, meaning that the model is lacking more contextual information.

Table 4: Example (Incorrect Prediction) -- Index: 6653



Feature	Value
True Label	Draw (1)
Predicted Label	Loss (0)
Probabilities	Loss = 0.460, Draw = 0.210, Win = 0.330
Promotions	2
White_Peaks	23
Checks	18
Black_Peaks	20
Moves	135
Captures	24
EarlyQueenDev	True
white_delta_num_std	296.83279
Black_ELO	2219
black_delta_num_std	276.514454
Max_time_Black	925
BlackCastled	True
WhiteCastled	True
Max_time_White	893
white_delta_num_avg	78.029412
White_ELO	1990
black_delta_num_avg	65.58209

7. Limitations and Future Work

One limitation of the current model is that it relies solely on in-game features such as move counts, timing variations, and tactical events (e.g., checks, captures, or promotions). While these indicators are valuable, they do not capture the broader context of the players themselves. Important metadata that could influence outcomes—such as psychological state, fatigue, or confidence—remains unavailable. For instance, whether a player was on a losing streak, their age, or even the psychological pressure of a tournament setting may significantly affect decision-making during a game. Without access to such contextual variables, the model still cannot yet truly predict human moves and game results.

Future work could integrate richer sources of metadata to provide a more holistic view of player performance. Historical performance trends (e.g., streaks of wins or losses), player demographics, or even real-time biometric data could improve the model's ability to predict human behavior. Additionally, incorporating natural language data from post-game commentary or player interviews might provide insight into psychological drivers behind decisions. Expanding the feature set beyond purely game mechanics could therefore make the model both more accurate and more interpretable in capturing the complex human dynamics of competitive chess. Perhaps if such projects were attempted, it could unlock a lot about human psychology in general.

8. Conclusion

This research shows that chess outcomes can be predicted using metagame features such as time usage, ELO differences, and activity statistics, though with limitations compared to board-based analysis. The model captured meaningful signals—especially in decisive games—but struggled with active draws, underscoring both the potential and constraints of metagame approaches. While time management and consistency provide a human-centered lens on play, the absence of richer metadata, such as tournament context or psychological factors, restricts predictive power. Expanding beyond the board could yield models that better reflect human competition, supporting training, preparation, and insights into player psychology.

References

[1] C. F. Chabris and E. Hearst, "Visualization, pattern recognition, and forward search: Effects of playing speed and sight of the position on grandmaster chess errors," Cogn. Sci., vol. 27, no. 4, pp. 637–648, 2003. Link: Visualization, pattern recognition, and forward search: Effects of playing speed and sight of the position on grandmaster chess errors.

[2] A. E. Elo, The Rating of Chessplayers, Past and Present. New York, NY, USA: Arco Publishing, 1978. Link: The Rating of Chess Players, Past and Present by Sam Sloan | Goodreads

[3] S. Guntz, J. L. Crowley, D. Vaufreydaz, R. Balzarini, and P. Dessus, "The role of emotion in problem solving: First results from observing chess," arXiv preprint arXiv:1810.11094, 2018. Link: The Role of Emotion in Problem Solving: First Results from Observing Chess

[4] I. Palacios-Huerta, "Cognitive performance in competitive environments: Evidence from a natural experiment," J. Public Econ., vol. 139, pp. 40–52, 2016. Link: Cognitive performance in competitive environments: Evidence from a natural experiment - ScienceDirect

[5] J. Schrittwieser et al., “Mastering Atari, Go, chess and shogi by planning with a learned model,” Nature, vol. 588, no. 7839, pp. 604–609, 2020. Link: Mastering Atari, Go, chess and shogi by planning with a learned model | Nature

[6] D. Silver et al., “Mastering chess and shogi by self-play with a general reinforcement learning algorithm,” arXiv preprint arXiv:1712.01815, 2017. Link: [1712.01815] Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm

[7] N. Tomašev, U. Paquet, D. Hassabis, and V. Kramnik, “Assessing game balance with AlphaZero: Exploring alternative rule sets in chess,” arXiv preprint arXiv:2009.04374, 2020. Link: Assessing Game Balance with AlphaZero: Exploring Alternative Rule Sets in Chess

[8] H. L. J. van der Maas and E.-J. Wagenmakers, “A psychometric analysis of chess expertise,” Amer. J. Psychol., vol. 118, no. 1, pp. 29–60, 2005. Link: A psychometric analysis of chess expertise - PubMed