Replication of "Quantifying the Complexity and Similarity of Chess Openings using Online Chess Community Data"

Lennart Gießing

February 13, 2025

1 Introduction

The goal of this study was to replicate the methodology used in the paper by Giordano De Marzo and Vito D. P. Servedio, titled *"Quantifying the Complexity and Similarity of Chess Openings using Online Chess Community Data"* [1]. The original study applied complex network theory to analyze chess openings by constructing a bipartite network of players and openings. Through the projection of this network, they identified clusters of similar openings and used these findings to predict which openings a player might adopt in the future. This report outlines the methodology used to replicate their work, compares the obtained results to those in the original study, and highlights key differences in approach.

2 Data Preprocessing

The dataset used in this study was obtained from the Lichess online database, which provides extensive metadata on chess games. The preprocessing phase involved multiple steps to clean, filter, and structure the data before analysis.

The first step was extracting the dataset. PGN files were downloaded for the period spanning October 2015 to December 2016. These files were compressed using Zstandard format and required decompression for processing. Each PGN file contained metadata on games played during the respective month, including player names, results, Elo ratings, timestamps, and opening codes (ECO). This process was not designed very efficiently as it took around 40 hours of computing time. The extracted metadata was compiled into a structured dataset, resulting in a large initial collection of raw games.

To ensure consistency in playing conditions, only Blitz games were retained. The dataset originally contained multiple formats, including Bullet and Classical games, but these were excluded.

Filtering based on player Elo ratings was applied. Games were retained only if both players had an Elo rating above 2000. Additionally, games where the Elo difference between opponents exceeded 50 points were removed.

A further filtering criterion was applied to ensure that only active players were included. Players were required to have played at least 100 games as White and 100 games as Black. For the dataset covering July to September 2016, a slightly more relaxed threshold of 50 games per color was used. Players who did not meet these criteria were excluded from the dataset.

To differentiate between openings played by White and Black, two additional columns were created. The WECO (White ECO) column assigned an ECO code to the opening played by the White player, while the BECO (Black ECO) column assigned an ECO code to the opening played by the Black player. This allowed for a distinction between openings based on the player's color, which was necessary for the subsequent network analysis.

Filtering Step	**My Dataset**	**Paper's Dataset**
Total Games Processed	89,656,443	Not Reported
Games Retained After Filtering	487,752	472,183
Minimum Player Elo	2000	2000
Maximum Elo Difference Allowed	50	50
Minimum Games per Player	100 as White, 100 as Black	100 as White, 100 as Black
Unique Players After Filtering	2,573	2,513
Unique Openings (ECO Codes)	984	982

Table 1: Comparison of Data Preprocessing for Oct15-Sep16: My Implementation vs. the Paper

2.1 Validating Links

To transform the player-opening bipartite network into a meaningful opening-opening network, I implemented the bipartite configuration model (BiCM). This process ensures that only statistically significant relationships between openings are retained.

The preprocessing begins by extracting unique players and openings from the dataset. A bipartite adjacency matrix M of dimensions $P \times O$ (where P represents players and O represents openings) is created. Each row corresponds to a player, while each column represents an opening. If a player has played a specific opening at least once, the corresponding entry in the matrix is set to 1.

To create a unipartite projection, I employed the BiCM projection method using the Poisson model. This method preserves the degree distribution of the bipartite network while eliminating spurious links. A p-value of 0.01 was used. The resulting projected network provides an opening-opening adjacency matrix , where an edge exists between two openings if they are statistically likely to be played by the same set of players.

2.2 Alternative Link Validation Method

As an alternative approach to validate the relatedness between chess openings, a randomization-based method using the Bipartite Configuration Model (BiCM) was considered. This method aimed to determine whether observed connections between openings were statistically significant by comparing them to a set of randomized networks.

The process started by computing the BiCM probability matrix, which captures the likelihood of a player playing a given opening while preserving the degree sequence of both players and openings. This probability matrix was then used to generate 300 synthetic bipartite networks where each link was assigned based on the computed probabilities.

For each of these randomized bipartite networks, a unipartite projection onto the opening space was performed, resulting in 300 projected matrices. By analyzing the distribution of link weights across these networks, a statistical significance threshold was established. Specifically, a 99th percentile threshold was applied, meaning that only links in the empirical opening-opening network that appeared more frequently than in 99% of the random projections were retained.

This validation process ensured that the final opening-opening network only contained connections that were significantly stronger than would be expected by chance. However, the results were very bad. Only very few links (256) remained.

3 Graph Construction and Clustering

After obtaining the validated adjacency matrix, I constructed a graph representation of the chess openings network. The graph was generated using the NetworkX library, where each node represents an opening and edges indicate statistically significant relationships between openings.

To ensure a meaningful network structure, isolated nodes (nodes with no connections(48)) and pairs of nodes (nodes that only connect to one other node(20)) were removed.

The remaining network was then converted into an *iGraph* structure to facilitate clustering analysis. A Leiden clustering algorithm was applied with a resolution parameter of 0.8 to detect communities within the chess openings network. Each node was assigned a cluster label corresponding to its detected community.

For visualization, I used the *spring layout* algorithm to generate node positions in a two-dimensional space. The network was plotted using *Plotly*, where nodes were color-coded based on their cluster assignments. Additionally, each cluster was labeled with its representative chess opening family, such as *Queen's Gambit*, *Sicilian Defense*, or *Ruy-Lopez*. These were found by creating a list of all openings that belong to a cluster. These were then transferred to ChatGpt in order to find out to which family of openings a cluster belongs. But it was not totally clear to which group each cluster actually belongs. A expert in chess openings would be needed to decide that.

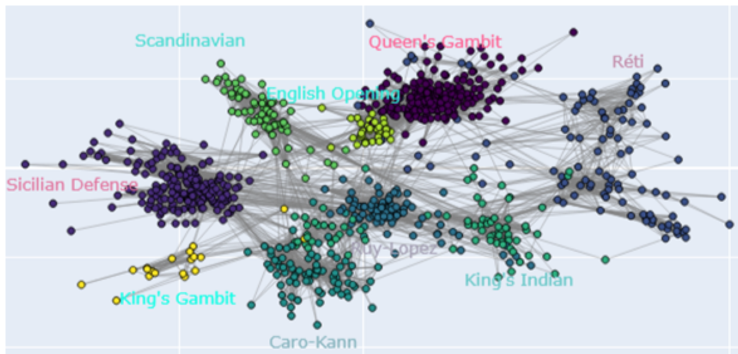

Figure 1: Replicated Chess Opening Network Visualization

4 Forecasting Future Opening Choices

One of the core objectives of the original study was to predict which openings a player would adopt in the future based on their past opening repertoire and the validated adjacency matrix. In order to conduct this forecasting analysis, the dataset was divided into two periods: July–September 2016 for training data and October–December 2016 for testing. Only players who appeared in both periods (10714) were retained, ensuring that predictions were made for individuals whose historical data was available. Also players with lower Elo rates were included. The adjacency matrix of openings was used to propagate past opening choices, under the assumption that if a player had played an opening, they were more likely to play openings closely connected to it.

The forecasting method was based on computing the density matrix ρ_{pa}, which measures the fraction of openings related to a new opening a that a player p had already played. A density threshold of 0.2 was applied, following the methodology described in the paper. And then changed later on to 0.03 due to a better F1 Score. If the density score for a player and an opening exceeded this threshold, the model predicted that the player would adopt this new opening in the test period.

5 Evaluation of Forecasting Results

To assess the performance of the forecasting model, I computed precision, recall, and F1-score. These metrics measure the effectiveness of predicting which chess openings a player will adopt in a future period based on historical data and the adjacency matrix of openings.

Precision is the fraction of correctly predicted openings among all predicted openings. A higher precision indicates fewer false positives.

Recall is the fraction of correctly predicted openings among all actual openings played. A higher recall indicates fewer false negatives.

F1-score is the harmonic mean of precision and recall, providing a balanced measure of prediction performance.

The table below compares the forecasting performance of different approaches:

Method	Precision	Recall	F1 Score
Paper results	0.10	0.47	0.16
My Basic Adjacency Model	0.0573	0.5500	0.1038
My Density Threshold (0.03)	0.0575	0.4675	0.1025

Table 2: Forecasting Performance Metrics

5.1 Analysis of Results

The results indicate that the basic adjacency model and the density threshold-based approach achieve recall values close to those reported in the paper but suffer from significantly lower precision. This suggests that the method used in this study tends to predict a large number of openings, many of which are false positives.

Compared to the results in the original paper, the forecasting method implemented here struggles to balance precision and recall. The lower F1-score indicates that while many openings are predicted, a significant portion of them are incorrect, leading to a high false positive rate.

To align more closely with the results from the paper, further refinements need to be explored.

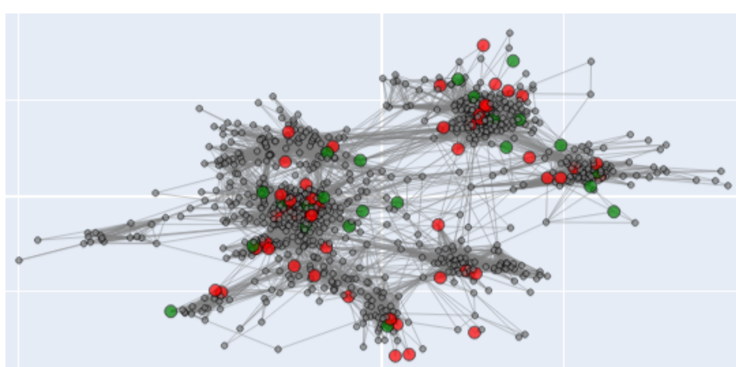

Figure 2: Random player and the openings he played from July-September (green) and from October-December (red)

6 Comparison to the Original Paper

While this replication followed the core methodology of De Marzo and Servedio, several differences exist. One key difference is the method used for opening-opening relatedness. The original paper applied a fully BiCM-filtered projection, while this study offers additionally a threshold-based filtering approach. Also, the forecasting method in the original paper involved a more refined transition probability function, rather than directly applying adjacency-based propagation. The original study also reported a better F1-score of 0.16, slightly higher than the 0.1038 achieved in this study, suggesting that further refinements to the methodology could improve accuracy.

Another important aspect of the original paper was the analysis of the *fitness of players* and the *complexity of openings*. The study applied the Economic Fitness and Complexity (EFC) algorithm to measure how difficult different openings are to play and to quantify a player's skill in choosing and using openings effectively. Their findings demonstrated that higher-rated players tend to play a more diverse and complex set of openings, while lower-rated players focus on simpler ones. They also showed that Magnus Carlsen's opening repertoire consisted mostly of high-complexity openings. This approach provided a **data-driven way** to assess both the difficulty of openings and the expertise of players.

Unfortunately, I was unable to fully replicate this part of the analysis in my study.

7 Conclusion

This study successfully replicated key aspects of the original paper, including the construction of the bipartite network, the projection into an opening-opening relatedness graph, the application of the Leiden clustering algorithm, and the forecasting of future opening choices. While the overall structure of the replication was aligned with the original study, methodological differences led to deviations in predictive performance.

References

[1] De Marzo, G., Servedio, V. D. P. (2023). "Quantifying the Complexity and Similarity of Chess Openings using Online Chess Community Data." *Scientific Reports.*