A Python-Based Data-driven Decision Engine Model for Chess Openings

[1]Eun June Lee, [2]Beom Jun Kim
[1,2]Department of Computer Science, Gaon Highschool, Gyeonggi-do, Republic of Korea

[1]Corresponding Author: eunjune1208@gmail.com

Keywords: Pseudo AI, Data-driven decision engine, Python, Chess game model

Abstract

This paper presents a pseudo AI model, which is a data-driven decision-making engine, for chess openings. In this paper, pseudo AI utilizes the data of chess openings to respond to the moves made by human players. This engine aims to improve understanding of the AI's learning and cognitive flow, and it is expected that this engine can be used in educational environments. This model, which does not involve data learning, can provide intuitive insights into AI algorithm for learners in educational settings.

1. Introduction

These days, as interest in Artificial Intelligence (AI) increases, some chess games have begun to use AI models deviating from traditional chess games. For example, the leading global platform of chess, chess.com, has introduced various AI players into their platform. This paper designed the data-driven decision engine, called pseudo AI, aimed at enhancing AI learning systems and contributing to the development of chess AI technology. Pseudo AI is a decision model that directly uses data itself, without training or learning the data. This model is more familiar to humans, because of its similarity to human decision making that depends on intuition and experience, and is easier to explain the AI's decision making process. This can facilitate an intuitive understanding of AI technology and learning systems, and can be useful as an educational resource.

2. Theoretical Background

Pseudo AI refers to a decision making system that is not AI, but similar to AI. It directly uses existing data as itself, without learning or performing any adaptive functions. It can be misjudged as the outdated form of modern AI. It has, however, both advantages and limitations when compared to modern AI systems.

The AI is superior to the pseudo AI system in terms of performance. The AI system generates functions, patterns, and algorithms based on the given data. As a result, it is more accurate, more rational, and faster than pseudo AI systems.

However, the pseudo AI is more suitable when imitating human algorithms, which operate unconsciously, based on experience. Also, the pseudo AI can be used as an educational tool at the learning site related to

AI or human cognitive style.

3. Algorithm implementation

3.1. Development of chess game algorithm using python

In this paper, a chess algorithm, that excludes promotion, castling, and *en passant* is implemented. Not including these special rules, every modern chess rules are developed, including pawn's double-step move, etc. The chess pieces are represented by texts, such as BR1, WP3, etc.

In the chess program, the pieces moves are implemented like moves in official chess. Player can input a piece type, the target coordinates. The coordinates, entered into the system, compared with the current position of the selected piece. In chess program, the pieces moves are implemented like moves in official chess. Player can write a piece type, the coordinates to move to. The coordinates, written on system, compared with the coordinates that the piece exists.

The Pawn can move forward one square, or make a double-step move, capture the front-diagonal piece. The Rook can move in horizontal or vertical directions, without restriction of the number of squares. The Bishop can move in diagonal directions over any number of squares, without limitation. The Knight can move in an L-shape: two square in one direction (horizontal or vertical), and one square perpendicular. The King can move in all 8 directions, but only one square. The Queen moves the same way that the King does, but without the limitation on the number of squares.

Every piece, except the Knight, cannot move if another piece is blocking the path. Only the Knight can jump over the other pieces that belong to the same team as the Knight.

The full source code of the chess game is available in the GitHub repository. [1]

3.2. Development of pseudo AI using python

This pseudo AI system uses the Python library, pandas, for data processing. The chess game program, described in Section 3.1, saves the data of the initial 10 moves into a specific path using pandas library. The pseudo AI utilizes this data to respond human players' moves in the game. It compares the players' moves with the saved data, then finds the corresponding moves.

```
df1 = pd.read_csv("C:/Users/dataframe/df1.csv")
for i in range(10):
    place1.append(df1.iat[1, i+1])
    type1.append(df1.iat[0, i+1])
```

Figure.1 Loading saved data from a specific path.

If there exist the same data in two or more dataframes, the priority is set based on the order of dataframes.

The full source code of the chess game with pseudo AI is available in the GitHub repository. [2]

4. Conclusion

The board represented in the game is as follows:

BR1	BN1	BB1	BK1	BQ1	BB2	BN2	BR2
BP1	BP2	BP3	BP4	BP5	BP6	BP7	BP8
NON	NON	NON	NON	NON	NON	NON	NON
NON	NON	NON	NON	NON	NON	NON	NON
NON	NON	NON	NON	NON	NON	NON	NON
NON	NON	NON	NON	NON	NON	NON	NON
WP1	WP2	WP3	WP4	WP5	WP6	WP7	WP8
WR1	WN1	WB1	WQ1	WK1	WB2	WN2	WR2

Figure.2 Game board

The squares without any pieces are displayed as “NON”. We verified that the chess game performs all intended motions, and pseudo AI responds to the player’s moves.

It is expected that this pseudo AI can be used for an educational environment where someone should learn the basic algorithm of chess AI. The chess game algorithm is so complicated that it is appropriate for verifying the advantages of pseudo AI in an educational environment. The complicated algorithm is effective for validating the accuracy and performance of the system. Thus, it is expected that it is adequate for AI’s basic algorithm educating.

However, these are estimations of the results of this research, and the empirical validation was not performed. Therefore, the further research on the educational effectiveness of this pseudo AI should be conducted in the future.

■ Reference

[1] E. J. Lee, “Python chess program,” (GitHub repository), https://github.com/LeeEunJune/python-chess-program, Accessed: Jul. 11. 2025.

[2] E. J. Lee, “Python chess educating statistical AI,” (GitHub repository), https://github.com/LeeEunJune/python-chess-educating-statistical-AI-, Accessed: Jul. 11. 2025.