Initial Plan: Chess AI That Explains Its Decisions

Author: Ned Kingdon

Supervisor: Kirill Sidorov

➔ Project Description

Most existing chess engines provide optimal move suggestions based on numerical evaluations of the board but lack human-friendly explanations for their decision-making process. This lack of interpretability creates a barrier for players who wish to understand why a move is chosen, rather than simply accepting it as optimal. While tools like DecodeChess and Chess.com attempt to provide explanations, gaps remain in the simulatability and transparency of these algorithms. The insights gained from explainable AI (XAI) in chess could also contribute to broader applications of AI interpretability.

This project explores how techniques from XAI can improve chess education. By doing so, it seeks to bridge the gap between AI-driven analysis and human chess understanding, making the engine more accessible and educational, particularly for beginner players. Unlike top engines like Stockfish, which prioritize deep search capabilities, this AI will focus on explainability over raw strength, making it ideal for learning fundamental concepts.

The project will implement and evaluate several explainability approaches, refining the final solution based on testing and user feedback. The core techniques to be explored include:

1. Component-Based Evaluation – Breaking down the evaluation function into distinct components (e.g., king safety, material, positional play) to provide users with a clear understanding of how moves are assessed.
2. Multiple AI Personalities – Designing different evaluation styles (aggressive, defensive, positional) and comparing their move choices, explaining where and why they disagree.
3. Tactical Pattern Recognition – Identifying key chess motifs like pins, skewers, and mobility to offer insights into positional strengths and weaknesses.
4. Search Tree Examination – Analyzing how the AI evaluates different move sequences to highlight key decision-making patterns.

By evaluating these approaches through systematic testing, the project aims to identify an optimal balance of interpretability and accuracy, resulting in a chess AI that enhances player learning while maintaining strong decision-making capabilities

➔ Aims and Objectives

Aims

To build a competent chess engine that presents understandable reasoning for its decisions to help beginner chess players understand why certain moves are better than others.

Objectives

1. Chess Engine Development

- Implement a basic chess engine capable of selecting strong moves using a search algorithm.
- Conduct research on search algorithms (e.g., Minimax, Negamax) and finalize an approach in consultation with my supervisor.
- Implement the chosen search algorithm with Alpha-Beta pruning for efficiency.
- Utilize Python-Chess to manage board representation, move legality, and chess rules.

2. Explainability Techniques & Evaluation

- Review existing explainability tools (e.g., DecodeChess, Chess.com's analysis) to identify their limitations in algorithmic transparency and effectiveness.
- Implement component-based evaluation, breaking down move assessments into at least four fundamental factors (e.g., material balance, king safety, mobility, and pawn structure).
- Develop at least three distinct AI personalities (e.g., aggressive, defensive, positional) with differing evaluation heuristics.
- Provide explanations for disagreements between personalities to illustrate diverse strategic perspectives.
- Experiment with search-tree analysis to improve transparency in move selection.

3. Testing, Evaluation, and Quantifying Interpretability

- Design experiments to assess the effectiveness of explainability techniques.
- Research and define metrics for measuring interpretability, such as:

    - User comprehension tests (e.g., asking players to explain a move before and after seeing the AI's explanation).
    - Comparisons of AI-generated explanations with human reasoning.
- Conduct test matches between AI versions and analyze their performance.
- Gather feedback from beginner chess players to determine how useful and understandable the explanations are.

4. Refinement, Transparency, and Broader Implications

- Use testing insights to refine the AI, improving both move accuracy and clarity of explanations.
- Ensure that the methodology and evaluation framework are well-documented and reproducible.
- Explore whether findings in chess AI interpretability can generalize to other AI-driven decision systems.
- Implement the most effective explainability techniques and integrate a pre-existing chess Graphical User Interface (GUI) (e.g., Chess Partner) to display explanations.
- Ensure compatibility with the Universal Chess Interface (UCI) and consider additional visualization tools like Matplotlib and Pandas.

➔ Feasibility

Chess programming is a well-established field with extensive free resources available online, such as the chess programming wiki. Given these resources, I am confident in my ability to develop a functional chess engine. However, due to the experimental nature of this project, achieving a perfect solution may not be feasible.

A key challenge lies in evaluating the effectiveness of AI-generated commentary, as interpretability is inherently subjective. To address this, I will explore multiple explainability techniques, allowing flexibility in my approach. If a particular method proves computationally intensive, I can prioritize an alternative that is less resource-demanding while still achieving the project's goals.

All third-party software used will be open-source, ensuring transparency and reproducibility. Additionally, as this project involves user testing, I will apply for ethics approval to ensure compliance with research guidelines.

Risks	Mitigation
Ethical approval delayed or denied	Apply for approval at the early stage of the project
Due to hardware constraints, the AI's search depth may be significantly limited compared to engines like Stockfish, potentially affecting move accuracy	Seta search depth limit appropriate for beginner play. Balancing performance and explainability
Evaluating whether explanations effectively help users understand chess strategy may be subjective and difficult to quantify	Define clear evaluation metrics, such as user comprehension tests (e.g., asking players to explain a move before and after seeing the AI's explanation).
Given the 12-week timeline, attempting to implement too many features could lead to delays and an incomplete project.	Focus on core objectives first, then add additional features only if time permits

➔ Work Plan

Week 1	27/01/2025
ID	
0: Conduct background reading into chess AI and explainable AI 1: Meet with supervisor to discuss the initial report 2: Draft initial report	
Milestone: Initial plan report submitted	
Week 2	03/02/2025
ID	Tasks
0: Install Python-Chess, Stockfish, and other necessary libraries 1: Conduct initial research on chess AI fundamentals, focusing on: Minimax algorithm, evaluation functions and explainability in ai (XAI) 2: Meet with supervisor to discuss the search algorithm choice.	
Milestone: Setup the project environment and refine research direction	
Week 3	10/02/2025
ID	Tasks
0: Implement a basic chess engine using Python-Chess 1: Continue reading into search algorithms 2: Implement a pre-existing GUI to visualise the chess board	
Milestone: Create a working chess engine that can play basic legal moves and begin implementing search algorithm.	
Week 4	17/02/2025
ID	Tasks
0: Implement the decided searching algorithm, utilize chess programming wiki for guidance 1: Test against stockfish with low depth searches to check correctness 2: Continue reading into explainable AI	
Milestone: The AI can evaluate and play using the search algorithm and a basic evaluation function	
Week 5	24/02/2025
ID	Tasks

0: Research into evaluation function and continue reading into XAI 1: implement a dynamic hand crafted evaluation function, where the components of it can be explained to the user	
Milestone: The AI can break down the evaluation function into at least 4 basic components	
Week 6	03/03/2025
ID	Tasks
0: Continue reading into XAI 1: Implement multiple personalities like aggressive, passive etc. 2: verify the playstiles by conducting test matches against one another	
Milestone: AI has at least three distinct playing styles	
Week 7	10/03/2025
ID	Tasks
0: experiment and implement useful explanations of the personalities disagreements 1: Begin integrating GUI elements to display personality decisions	
Milestone: AI can explain its decision with regards to how the different personalities disagree with each other	
Week 8	17/03/2025
Reading week	
Week 9	24/03/2025
ID	Tasks
0: Begin implementing more explainability techniques such as search tree rationalization 1: also consider any new techniques that result from research	
Milestone: Have an alternative method to explain the AI's moves	
Week 10	31/03/2025
ID	Tasks
0: Research metrics for measuring explainability 1:Conduct test matches between AI versions 2: Test user comprehension of explanations	

Milestone: Identify the best explainability method	
Week 11	07/04/2025
ID	Tasks
0: Final refinements to AI logic and UI improvements 1: Write a draft for the beginning of the report, in particular the introduction, background and the methods section.	
Milestone: A complete and working solution to the problem with improvements based on testing	
Week 12	05/04/2025
ID	Tasks
0: Write the remaining categories in the report: Results, Discussion, Conclusion 1: Format and finalise the report 2: Submit code, report and any other necessary documentation	
Milestone: Submit project	