Connect4 AI: Evaluation Function Analysis

Hadi Bagdadi

February 25, 2025

1 Introduction

This document presents an analysis of the evaluation function used in the alphaBetaAI implementation for the Connect4 game. The evaluation function is a critical component of the minimax algorithm with alpha-beta pruning, as it determines how favorable a board state is for the player.

2 Mathematical Function for Evaluation

The evaluation function can be expressed mathematically as follows:

$$\text{Eval}(\textit{board}) = \begin{cases} \pm 10000, & \text{if terminal state (win/loss)} \\ \text{position\_score} + \text{pattern\_score} + \text{trap\_score}, & \text{otherwise} \end{cases} \tag{1}$$

Where each component is calculated as:

$$\text{position\_score} = (\textit{our\_weighted\_sum} - \textit{opponent\_weighted\_sum}) \times w_{position} \tag{2}$$

$$\text{pattern\_score} = \left(\sum \text{window\_evaluations}\right) \times w_{pattern} \tag{3}$$

$$\text{trap\_score} = \text{trap\_count} \times w_{trap} \tag{4}$$

The weights $w_{position}$, $w_{pattern}$, and w_{trap} vary based on the game phase:

Weight	Early Game	Mid Game	Late Game
$w_{position}$	1.2	1.0	0.8
$w_{pattern}$	1.0	1.5	2.0
w_{trap}	15	20	30

Table 1: Dynamic weights based on game phase

2.1 Position Weighting Matrix

The position weight matrix assigns higher values to strategically important positions:

$$\text{Position Weights} = \begin{bmatrix} 3 & 4 & 5 & 7 & 5 & 4 & 3 \\ 4 & 6 & 8 & 10 & 8 & 6 & 4 \\ 5 & 8 & 11 & 13 & 11 & 8 & 5 \\ 5 & 8 & 11 & 13 & 11 & 8 & 5 \\ 4 & 6 & 8 & 10 & 8 & 6 & 4 \\ 3 & 4 & 5 & 7 & 5 & 4 & 3 \end{bmatrix} \tag{5}$$

2.2 Window Evaluation Scoring

For each window of four consecutive cells (horizontal, vertical, or diagonal):

Window Pattern	**Score**
Player has 4 in a row	+100
Player has 3 in a row with an empty space	+15
Player has 2 in a row with two empty spaces	+5
Player has 1 with three empty spaces	+1
Opponent has 3 in a row with an empty space	-15
Opponent has 2 in a row with two empty spaces	-3

Table 2: Scoring for different window patterns

2.3 Trap Detection

A trap is detected when placing a piece in an empty cell would create multiple winning threats (two or more potential ways to form four in a row). Each trap detected adds to the trap score.

3 Worked Example

Consider the following mid-game board state:

$$\text{Board} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 0 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 & 0 & 0 \\ 2 & 1 & 2 & 2 & 1 & 0 & 0 \end{bmatrix} \tag{6}$$

Where:

- 1 represents the AI player (position)

- 2 represents the opponent
- 0 represents empty cells

Assuming we're in the mid-game phase (moves 10-20), we use these weights:

$$w_{win} = 10000 \tag{7}$$
$$w_{position} = 1.0 \tag{8}$$
$$w_{pattern} = 1.5 \tag{9}$$
$$w_{trap} = 20 \tag{10}$$

3.1 Position Score Calculation

$$\text{AI pieces at positions with weights} = 11 + 8 + 8 + 5 + 5 = 37 \tag{11}$$
$$\text{Opponent pieces at positions with weights} = 10 + 8 + 4 + 3 = 25 \tag{12}$$
$$\text{Position score} = (37 - 25) \times 1.0 = 12 \tag{13}$$

3.2 Pattern Score Calculation

Looking at all possible windows of 4 cells:

- Horizontal: One window with 2 AI pieces and 2 empty spaces (+5)
- Vertical: One window with 3 AI pieces and 1 empty space (+15)
- Diagonals: No significant patterns

$$\text{Pattern score before weighting} = 5 + 15 = 20 \tag{14}$$
$$\text{Weighted pattern score} = 20 \times 1.5 = 30 \tag{15}$$

3.3 Trap Detection

There are no trap setups (positions where placing a piece would create multiple winning threats)

$$\text{Trap score} = 0 \times 20 = 0 \tag{16}$$

3.4 Total Score

$$\text{Total evaluation} = 12 + 30 + 0 = 42 \tag{17}$$

This positive score indicates an advantage for the AI player.

4 Motivation for the Evaluation Function

The evaluation function is designed with several key strategic elements that make it effective for Connect4:

1. **Positional Control**: It values controlling the center and other strategically important positions on the board through a weighted position matrix. Center control gives more potential winning lines.

2. **Pattern Recognition**: It identifies threatening patterns that could lead to wins, prioritizing sequences with more pieces aligned. This rewards progress toward winning configurations.

3. **Trap Detection**: It rewards positions that create multiple winning threats, which are powerful in Connect4 as they often force a win regardless of opponent moves.

4. **Dynamic Phase Weighting**: The function adapts throughout the game, emphasizing position control early, pattern development in the mid-game, and trap detection in the late game.

This multi-faceted approach allows the AI to balance immediate tactical considerations with longer-term strategic planning, making it more effective against different opponents and in various game situations. By prioritizing different aspects of evaluation based on the game phase, the AI can adapt its strategy as the board fills up and different types of threats become more relevant.