Nilesh Bhoi (nnbhoi) Niranjan Balasubramani (nibalasu) Rasi (rasi01) Sanchit Relan (srelan)

In this project we have combined a custom neural network with a custom minimax algorithm to play Gomoku on a 15x15 board. The neural network learns game states and adapts from game scenarios, enhancing strategy along the way. If neural network gives invalid action, we use the minimax algorithm with a custom evaluation function to strategically minimize losses and maximize gains to produce optimal moves. This dual approach not only makes the AI highly skilled in Gomoku but also makes it adaptable to various playing styles, displaying a state-of-the-art AI and game theory.

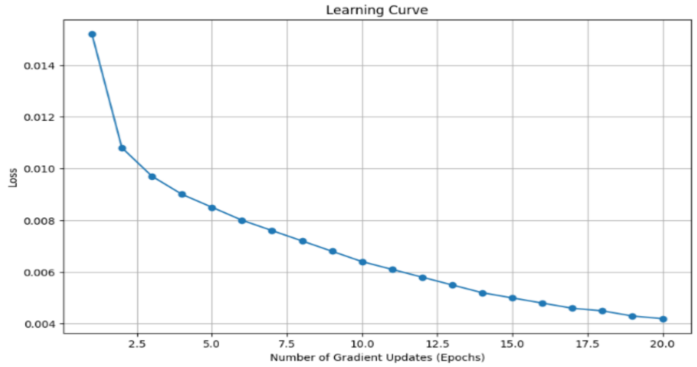
Key Features of the Custom Minimax [1]:

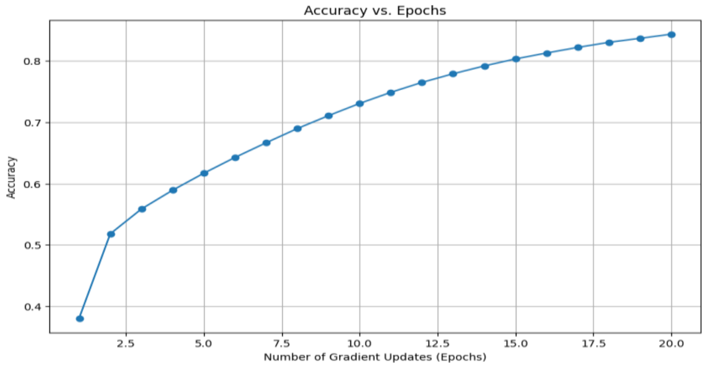
The custom heuristic evaluation function dynamically allocates the score and weighs the board state. It scans the board in various directions (horizontal, vertical, diagonal, and anti-diagonal) and aggregates scores based on consecutive actions by the player on the board in continuous direction and blocks the opponent two moves before it is about to win. Scores have more weight when the game is close to completion.

The MIN player chooses an aggressive strategy and gets assigned higher scores for potential winning sequences (get_line_score() function), promoting a more proactive and offensive gameplay. Due to the weighted evaluation favoring the MIN player, the MAX player often resorts to a defensive approach, focusing on blocking rather than creating its own threats. The aggressive approach led to early victories for the MIN player, with many games concluding rapidly in favor of the Submission. The MAX player's reactive behavior and defensive strategy, primarily focused on countering the threats posed by the MIN player, often led to missed opportunities for offense. Due to the chosen strategy, the MIN player wins early for most games but also loses early sometimes. In a few cases, the performance of the MIN player outweighs that of the MAX player and is quick to finish a game.

Key Features of the Custom Neural Network [2]:

Input Data: In the project 2880 input and output game states are chosen from the Gomo-cup competition,2022.
Neural Network Architecture: We have used a sequential convolutional neural network (CNN) with six convolutional layers. The architecture first increases and then decreases the depth of the network, starting with 64, escalating to 256 and going back to 64(64->128->256->128->64), Activation function used in this layer is RELU. The final layer is a single 1x1 convolutional layer which is then followed up by a reshape layer with sigmoid activation function, for binary classification of the predicted state. In the project, we have used the Adam optimizer in our neural network, which produces better results than SGD in our scenario. Also, **epoch size** of **20** was optimal in this scenario.
Training Results: As mentioned below in the figure, we are getting an **accuracy** of around **84%** and **loss** of **0.042**.





Results:

On average, we are **winning 75%** of the games and above with a **deviation of** $\pm 5\%$ consistently.

Discussion:

One of the approaches we followed in our custom neural network was to reduce the number of layers and increase the number of epochs. This resulted in a slow increase in accuracy as the epochs increased. The learning curve was slower and took more compute time to train. The trained model was also overfitting the data, resulting in less wins compared to our current approach.
The other approach that we tried involved achieving a balance between - accessing the model and custom minmax function to calculate the action manually. When the depth limit is reached, the model predicts the utility to arrive at optimal action. However, it introduced a significant overhead while calculating utilities of all children at each step. In future scope, this problem can be addressed by performing n-fold cross-validation.

Bibliography:

[1] Allis, Louis Victor, Hendrik Jacob Herik, and Matty PH Huntjens. Go-moku and threat-space search. Maastricht, The Netherlands: University of Limburg, Department of Computer Science, 1993.

[2] Fu, Xiang. "GomokuPro: An Implementation of Enhanced Machine Learning Algorithm Utilizing Convolutional Neural Network in Gomoku Strategy and Predictions Model." *2022 7th International Conference on Intelligent Computing and Signal Processing (ICSP)*. IEEE, 2022.