

# ***AI CONTROLLED PONG GAME USING NEAT PYTHON***

## GROUP MEMBERS

Muhammad Salar (21K-4619)

Muhammad Hamza (21K-4579)

Bilal Shakeel (21K-4874)

# INTRODUCTION

Pong, the quintessential arcade game, has been a cornerstone in the gaming industry. This project introduces a novel twist by integrating an AI opponent powered by the NEAT algorithm, capable of competing against human players and other AI entities with remarkable efficacy.

# PROBLEM DEFINITION

The static nature of traditional Pong AI creates a monotonous experience for players. This project addresses the need for a dynamic opponent that can adapt its strategies to the evolving gameplay, thereby enhancing player engagement.



## PROPOSED METHODOLOGY

The project employs Python and Pygame to simulate the Pong environment, establishing the foundational rules, paddle mechanics, and ball dynamics. The NEAT algorithm is at the heart of the AI development, enabling the evolution of neural networks that act as the AI players. These networks undergo training through interactive gameplay, with their efficacy gauged by their competitive prowess

# SYSTEM ARCHITECTURE

The system is structured around the PongGame class, which orchestrates the game's logic and the AI's developmental cycle. Functions like `test_ai`, `train_ai`, `move_ai_paddles`, and `calculate_fitness` are integral to the AI's learning mechanism. The system also incorporates a checkpoint management system to preserve the evolutionary progress of the AI

# DISTINCTIVE FEATURES

- **Dynamic Adaptation:** The AI dynamically adjusts its strategies in response to the ball's movement, ensuring a fluid and responsive gameplay.
- **Continuous Learning:** The AI's ability to learn from each session guarantees a consistently escalating challenge for players



# TECHNOLOGICAL STACK

- **Python:** The primary language for crafting the game and AI logic.
- **Pygame:** The framework for rendering the 2D game environment and handling user interactions.
- **NEAT-Python:** The chosen library for implementing the NEAT algorithm.
- **Development Environment:** The use of an IDE such as Visual Studio Code streamlined the development process.





## CONCLUSION

The project culminates in the creation of an AI-driven Pong game that elevates the player's experience by introducing a real-time learning and adapting AI opponent. This advancement significantly enriches the Pong gameplay, marking a substantial contribution to the domains of game development and artificial intelligence.

# PROJECT SCREENSHOTS

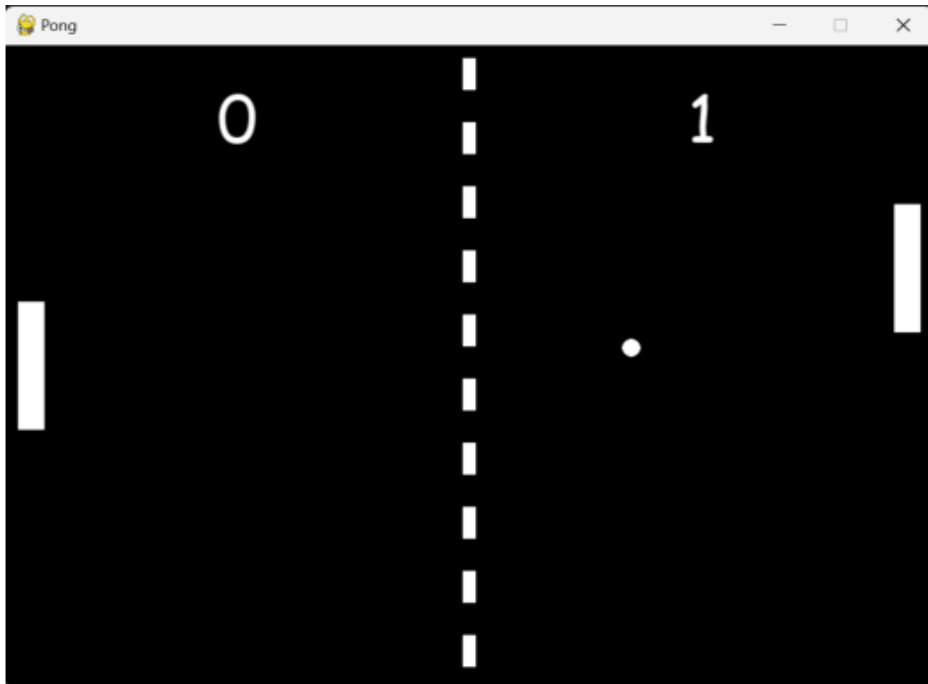


Figure 1: Game Window

```
***** Running generation 1 *****  
  
0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50 52 54 56 58 60 62 64 66 68 70 72 74 76 78 80 82 84 86 88  
90 92 94 96 98 Population's average fitness: -295.81818 stdev: 552.06318  
Best fitness: -0.69325 - size: (6, 22) - species 1 - id 47  
Average adjusted fitness: 0.873  
Mean genetic distance 2.303, standard deviation 0.249  
Population of 50 members in 1 species:  
  ID  age  size  fitness  adj fit  stag  
  --- --  ---  ---      ---      ---  
   1   1   50   -0.7    0.873    0  
Total extinctions: 0  
Generation time: 31.075 sec (34.989 average)  
Saving checkpoint to checkpoints/neat-checkpoint-1
```

Figure 2: Training Model



# **FUTURE DIRECTIONS**

Prospective developments include the introduction of multiplayer capabilities, the refinement of AI learning algorithms, and the incorporation of intricate game elements to further enhance the gaming experience.

# REFERENCES

A comprehensive list of references is included, citing the sources of information, libraries, and tools used throughout the project's development.

- **Python Documentation:**

- <https://docs.python.org/3/>

- **Pygame Documentation:**

- <https://www.pygame.org/docs/>

- **NEAT-Python Documentation:**

- <https://neatpython.readthedocs.io/en/latest/index.html#>

- **Efficient Evolution of Neural Network Topologies:**

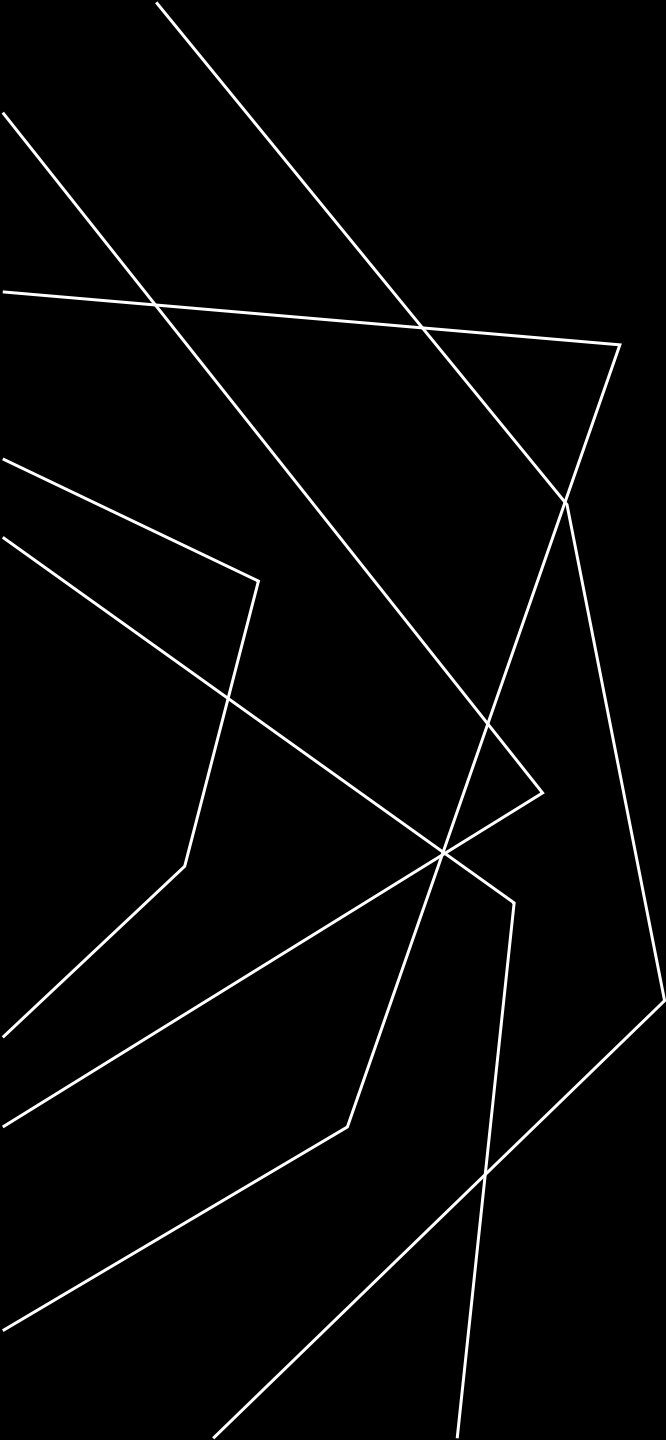
- <https://nn.cs.utexas.edu/downloads/papers/stanley.cec02.pdf>

- **Pong Game Documentation:**

- <https://pysdl2.readthedocs.io/en/latest/tutorial/pong.html#>

- **Visual Studio Code Documentation:**

- <https://code.visualstudio.com/docs>



THANK YOU