Smart adjustment of action for an AI-driven entity

Nejneriu Mihai
"Gheorghe Asachi" Technical Universityof Iasi
mihai.nejneriu@student.tuiasi.ro
Scientific coordinator: Assist.Prof.PhD.Eng.Tiberius Dumitriu

Abstract: - *Artificial intelligence (AI) can be used to create a software agent able to autonomously solve a specific issue by reacting to external stimuli. Such a model can be equipped with self-learning capacity, which will allow it to quickly adapt to different environmental conditions and new stimuli, thus being able to coordinate an entity whose movements become autonomous.*

This study is focused on the development of an artificial intelligence model within a video game, designed to demonstrate how AI can adjust dynamically enemy actions to human player movements, enhancing the tactical challenge. This capacity to adjust an entity's movements as a response to human action can be used in many other fields, such as drone flying, autonomous cars, augmented reality, unknown environment exploration, etc. The AI-driven adaptive enemy contributes to the broader exploration of AI applications in various virtual environments, highlighting the versatility and potential impact of such predictive models.

Keywords- Artificial Intelligence, AI-driven , Artificial Neural Network.

1.Introduction

This study deals with creation of a smart computer system applicable not only in video games but also in various real-world scenarios. It is designed to illustrate how this system can adapt to human actions, whether in gaming or other fields like drone navigation [1], autonomous driving [2,3], or exploration of unfamiliar environments [4]. By developing AI-driven entities capable of adjusting their behavior based on human input, the potential of predictive models beyond the gaming context was explored, emphasizing their versatility and practical utility.

The core of this research lies in understanding how these AI-driven entities can track and respond to moving targets influenced by human decisions. The primary aim is to coordinate these entities to interact effectively with human operators enhancing decision-making processes and performance in diverse settings. The ultimate goal is to endow AI systems with the ability to swiftly adapt to changing environments and accurately predict human behavior, thereby optimizing outcomes and minimizing human error. To achieve these objectives, the model undergoes rigorous training in two stages: initially learning to approach the target based on predetermined criteria, and then refining its understanding through real-time interactions with human operators. Once trained, the AI system can be deployed across various domains, from assisting in complex navigation tasks to optimizing decision-making processes in dynamic environments. Furthermore, the adaptability of the model allows for customization to suit specific application requirements, broadening its range of utility and potential impact.

By combining empirical research with theoretical frameworks, this study propose an intelligent system capable of enhancing human performance and decision-making across diverse domains.

The algorithms used for predicting movement are based mainly on extracting data from the game regarding user actions to acquire a desirable level of efficiency. This can be a real challenge, given that technology lacks comprehension of human decisions. However, by connecting all applications to one commune AI-model, it is possible to obtain enough data to make the model capable of predicting the majority of humans. The trained model can then be stored locally on most devices and used in a signal-free environment, and adjusted accordingly.

The first algorithm taken in consideration was Q-learning, but it needs a lot of memory to be able to store every possible case. In a video game environment, this can be possible, but the game serves just as a mean to illustrate the taken decisions and to retrieve human related data. If used just in this scope, it cannot be extended to more relevant territories. Q-learning also needs to simulate the human behavior to be able to train properly, which is the first problem encountered. To adapt to user actions a video game was used as a base.

A (A-star algorithm)* is another popular algorithm incorporated in many projects, precisely due to its adaptability [5,6]. By modifying the heuristic function, it can be used in different fields. The main problem is that A-star is simply a decision-making algorithm, and not one that can adjust based on the situation, thus it is necessary to be improved.

The architecture of the application is presented in section 2 while in section 3 some results are explained. Finally, the conclusions and the references are showed.

2. APPLICATION'S ARCHITECTURE

The game itself is designed with multiple levels, each serving a specific purpose in training the artificial intelligence (AI). The initial level serves as a reconnaissance stage where the client communicates vital information to the server. This information includes the player's position, the type of map in play, and a comprehensive list of data related to enemies, encompassing their objective and positions on the map.

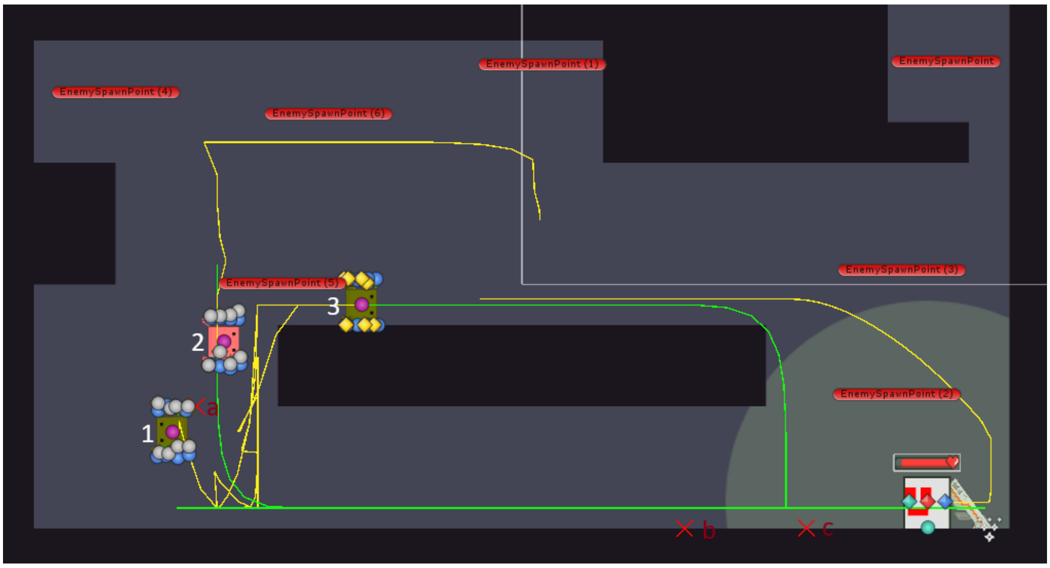

Figure 1. Assigned target for enemies in first level.

The neural network is hosted on the server, primarily to facilitate the reception of data from numerous clients, thus expediting the training process. For every connected client, the server spawns a new thread, enabling seamless communication between the server and the client at various stages. The client transmits data to the server, which in turn utilizes this data as input to generate an output, subsequently sending it back to the client. Further, the client assesses a score for the specific output, notifies the server regarding this assessment, and the server then updates its data to enhance score generation.

To simulate real-world scenarios, enemies employ the A-star algorithm to navigate the game environment. In this first level, the target for the enemies is set to the player's position, enabling the AI to adapt dynamically to the user's actions. Once this data is transmitted to the server, an additional step involves sending a random position within the A-star-generated path as a new target, where enemy with index 1 has as target the marker with index “a” (Figure 1). Once each enemy is assigned a target, it becomes the destination for the A* algorithm. A* is utilized to determine the shortest path from the enemy (acting as the source) to the designated target. In this scenario, the source is the enemy itself, and the target is the one mentioned earlier. The algorithm calculates the shortest path by assessing the distance from the current position to the destination, employing a heuristic function; in our case, the Manhattan distance (eq. 1) is used to estimate the remaining distance to the target. During each iteration, the algorithm selects the optimal move and proceeds to adjacent spaces based on adding the best remaining movement to the target.

In plane, the Manhattan distance between two points $P_1(X_1, Y_1)$ and $P_2(X_2, Y_2)$ is calculated as:

$$\mathrm{DM} = |X_1 - X_2| + |Y_1 - Y_2| \tag{1}$$

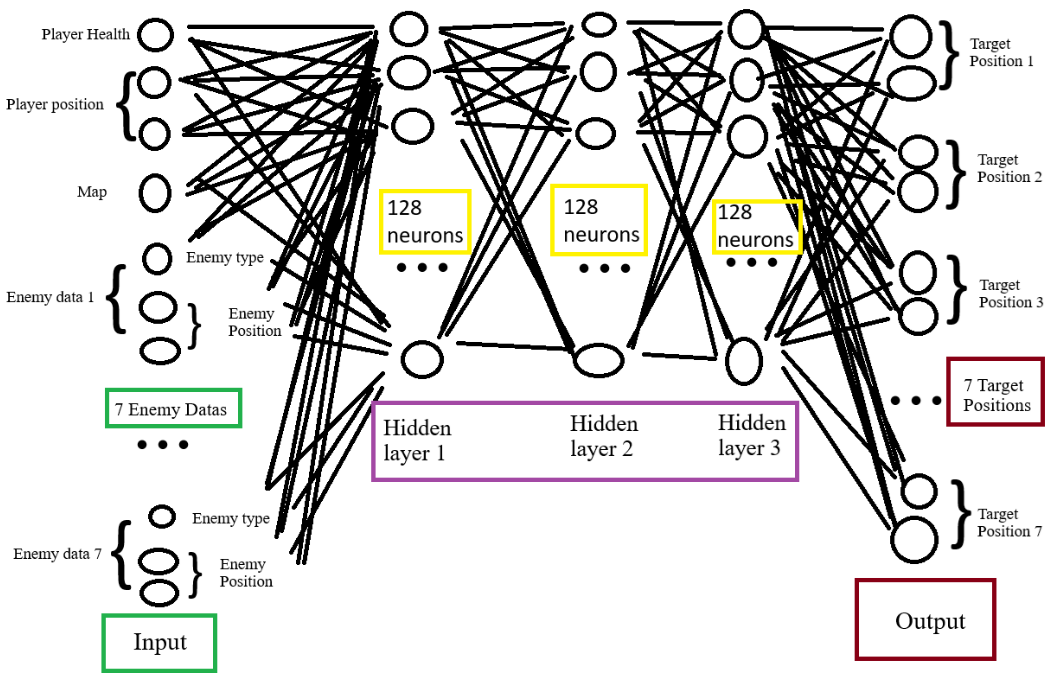

Figure 2. Neural Network Structure

Upon completing the initial phase, the training process will proceed to subsequent stages where the direct transmission of the target to the server ceases. Instead, the received data will serve as input for a neural network. Utilizing a straightforward mechanism, the neural network will then generate a new target (as illustrated in Figure 2), which subsequently becomes the destination point for the A* algorithm.

To simplify the process, the score was determined solely by computing the distance between the target and the player, with the aim of bringing the target closer to the player. A negative value will result in distance minimization. Additionally, the current target will be rewarded if the player experiences a decrease in health following execution, by adding points to the distance.

With this updated scoring system, the neural network employs backward propagation to adjust the weights of its nodes, facilitating ongoing learning and refinement of decision-making processes.

It is important to note that the scoring system is flexible, allowing modifications to achieve different outcomes. This adaptability adds a layer of customization to the training process, enabling developers to fine-tune the AI based on specific criteria or objectives. Overall, this multi-level training approach, coupled with dynamic target setting and a robust scoring system, contributes to the evolution and enhancement of the AI model's capabilities in the gaming environment.

3. RESULTS

Repeated training of the neural network during the initial level provides invaluable insights into the AI's learning progression. By subjecting the network to multiple training sessions, we can closely monitor its development and gauge its ability to adapt and improve over time. This iterative approach allows to observe subtle nuances in the AI's decision-making process and its capacity to learn from previous experiences.

In the image provided (Table 1), the left side displays the score attained with the target received from the client, while the right column shows the score computed after generating a new target. The scoring mechanism prioritizes movement towards the player, with the score subsequently negated to minimize distance. Results are generated every 20 iterations to showcase the progress, which is evident in the gradual improvement observed over time. It's important to note that the initial score calculation might yield poor results due to the AI's adaptation to unfamiliar moves. The score is negative, so a smaller number after "-" means a better score.

Despite the inherent challenge of data scarcity beyond the initial stage, visible advances in AI performance serve as an indicator of its potential for further improvement. Tracking the AI's evolution through successive training sessions not only provides valuable feedback on its learning trajectory but also gives information on future optimization strategies. By identifying patterns of improvement and areas requiring additional focus, we can iteratively refine the training process to maximize the AI's efficacy in real-world applications.

Table 1. Data extracted from the AI-driven system in the first level

Initial score	Final score
1. Initial score: -400.187016091088	-44.12525348891838
	-73.18319821255513
	-247.62019685350884
	-237.97669459732987
	-292.4595552841468
	-327.0785806258294
2. Initial score: -253.4931572998001	-10.729445070779402
	-24.989144083615102
	-160.23946064360783
	-106.36377719017331
	-162.67785670403552
	-184.01125359291413

3. Initial score: -470.1503306979869	-71.36430515085632
	-284.8910011997594
	-247.1126815952552
	-234.69875552978274
	-238.2060624174647
	-246.50962664401942
4. Initial score: -320.3744121877985	-77.94966749614211
	-189.10210098412426
	-140.45077424877323
	-186.89989169360422
	-229.44419262373907
	-135.61499675767365
5. Initial score: -163.6191583854462	-468.37964066360723
	-337.61234750450467
	-321.47572975504966
	-292.4984443527943
	-251.29971790675745
	-172.0274838888791
6. Initial score: -168.4513006226815	-505.37461356896404
	-280.6636029468519
	-266.5987020528848
	-218.94167734744752
	-154.6591479215267
	-154.24267272793065
7. Initial score: -61.48511533455621	-415.91972241618976
	-159.04536678899674
	-91.53211129306351
	-79.73537477005789
	-76.95765018708133
	-84.54794538997655
8. Initial score: - 47.2421246104286200	-420.1802634955971
	-158.19214774164928
	-82.08740710035472
	-93.7289089220464
	-109.56549370292262
	-106.9195285497361
9. Initial score: -114.07474932634523	-223.6965606808398
	-90.09051042030718
	-46.99960901041584
	-62.72929300860113
	-81.04128568490127
	-78.70484256750882
10. Initial score: -72.98616503150811	-43.919705085866354
	-30.151664767105004
	-38.548252569335105
	-54.66113185150512
	-59.54318600231079
	-67.43669271316038

4. CONCLUSIONS

This study demonstrates AI's ability to adapt enemy actions, processing data related to human movement. Its broader applications include drone navigation and autonomous driving, showcasing AI's versatility in various virtual environments.

By initially pretraining the model using A*, a renowned algorithm for pathfinding, we establish a solid foundation rooted in efficient navigation. This pretraining phase enables the model to grasp fundamental concepts of optimal path determination. Subsequently, by fine-tuning the model using a score-based feed-forward neural network, a layer of adaptability and refinement was introduced. This neural network leverages real-time data and feedback to dynamically adjust the model's decision-making process. Through this iterative approach, the model not only maintains its proficiency in pathfinding, but also evolves to adapt better to nuanced scenarios. This leads to promising outcomes characterized by improved efficiency and adaptability when navigating in complex environments.

The results of this study highlight the significant progress made in the development and application of artificial intelligence models within video games. By demonstrating the adaptability of AI systems to human player movements and the iterative evolution achieved through training, valuable insights into the potential of predictive modeling techniques have been gained.

5. REFERENCES

[1] D. Jano and S. Arogeti, *Drone's Attitude Estimation in Corridor-Like Environments*, 2019 European Conference on Mobile Robots (ECMR), Prague, Czech Republic, 2019, pp. 1-6, doi: 10.1109/ECMR.2019.8870961.

[2] Jingda Wu , Zhiyu Huang , Wenhui Huang and Chen Lv, *Prioritized Experience-Based Reinforcement Learning With Human Guidance for Autonomous Driving*, IEEE Transactions On Neural Networks And Learning Systems, Vol. 35, No. 1, January 2024, Page(S): 855 - 869, DOI: 10.1109/TNNLS.2022.3177685

[3] F. Vanderhaegen, *Heuristic-based method for conflict discovery of shared control between humans and autonomous systems - A driving automation case study*, Robotics and Autonomous Systems (vol. 146) 2021, DOI:10.1016/j.robot.2021.103867

[4] Max Schwarz, Marius Beul, David Droeschel, Sebastian Schüller, Arul Selvam Periyasamy, Christian Lenz, Michael Schreiber and Sven Behnke, *Supervised Autonomy for Exploration and Mobile Manipulation in Rough Terrain with a Centaur-Like Robot*, TECHNOLOGY REPORT, Front. Robot. AI, 13 October 2016 Sec. Humanoid Robotics, Volume 3, 2016, DOI: doi.org/10.3389/frobt.2016.00057

[5] Zailong Zhang, *A-star algorithm for expanding the number of search directions in path panning*,(School of computer science Nanjing University of Posts and Telecommunications),2021

[6] G. T. T. Bernardo et al., *A-Star Based Algorithm Applied to Target Search and Rescue by a UAV Swarm*, 2022 Latin American Robotics Symposium (LARS), 2022 Brazilian Symposium on Robotics (SBR), and 2022 Workshop on Robotics in Education (WRE), São Bernardo do Campo, Brazil, 2022, pp. 49-54, DOI: 10.1109/LARS/SBR/WRE56824.2022.9996054.