

# VR user experience prediction based on swarm intelligence optimization algorithm to optimize transformer model

Qianwen Jiang<sup>1</sup>, Jae Eun Yoon<sup>1,2,\*</sup>

<sup>1</sup>Graduate School of Techno Design, Kookmin University, Seoul, 02707, Korea

<sup>2</sup>dreamask@hanmail.net

\*corresponding author

**Abstract.** This article optimizes the transformer model based on swarm intelligence optimization algorithm to enhance the predictive ability of users' virtual reality (VR) experience. By analyzing the accuracy and loss value changes of the training set, we found that the model showed significant improvement during the training process. Specifically, the accuracy of the training set gradually increased from 62.74% to 86.94% and stabilized; At the same time, the loss value also decreased from 0.71 to 0.37, showing a good convergence trend. This indicates that the model effectively captures important features in the data during the learning process, thereby improving its predictive performance. Further analyzing the confusion matrix of the training set, we can see that 552 VR immersive predictions are correct, while 92 have errors. Among them, 55 instances that should have been predicted as Level 1 immersion were mistakenly classified as Level 2 immersion, while 37 instances that should have been Level 2 immersion were incorrectly predicted as Level 1. This reflects the confusion of the model in certain categories, despite achieving an overall accuracy of 85.69%. On the test set, 230 predicted results were correct, while 45 were not. Among them, 29 instances that should have been level 1 immersion were misclassified as level 2, and 16 instances that should have been level 2 immersion were misclassified as level 1, resulting in an accuracy rate of 83.63% on the test set. In addition, by outputting the ROC curve of the test set with an AUC value of 0.829, it further proves that the model has good classification performance. In summary, this study significantly improved the accuracy of transformer models in predicting VR experiences through swarm intelligence optimization algorithms. This achievement not only validates the effectiveness of swarm intelligence technology in complex data processing, but also provides new ideas and methods for future related fields. With the development of virtual reality technology, accurate prediction of user experience will help improve product design and service quality, thereby promoting the development of the entire industry.

**Keywords:** Swarm intelligence optimization algorithm, transformer, VR user experience.

## 1. Introduction

Virtual reality (VR) technology has developed rapidly in recent years and is widely used in many fields such as gaming, education, healthcare and training. Its research background dates back to the 1960s, when VR systems relied heavily on computer graphics and human-computer interaction technologies [1]. With the improvement of computing power and the advancement of sensor technology, modern VR devices are becoming more and more popular, and the user experience has also been significantly

improved. However, while VR technology has made significant progress in various fields, how to enhance user immersion and satisfaction is still an important research topic.

Machine learning (ML), as a powerful data analysis tool, plays an important role in predicting users' virtual reality experiences [2]. By analyzing large amounts of user data, machine learning is able to identify patterns of user behavior in a VR environment. These patterns include eye tracking, motion capture, and physiological signals to help researchers better understand how users react in different scenarios [3]. This data-driven approach offers new perspectives for optimizing virtual reality experiences, enabling designers to make more scientific decisions based on actual data.

Machine learning can also enable personalized recommendations, providing virtual reality content tailored to users based on their past behavior and preferences. For example, in the field of education, VR courses can be recommended according to students' learning progress and interests, thus improving learning results [4]. In addition, by combining physiological signals with machine learning algorithms, users' emotional changes in the VR environment can be monitored in real time, enabling the system to dynamically adjust the virtual environment to enhance immersion, and this emotion recognition capability provides strong support for enhancing the overall experience.

In short, with the development and popularization of virtual reality technology, the method combined with machine learning is becoming an important tool to improve user experience. In the future, we can foresee more intelligent and personalized virtual reality applications, which will greatly enrich the way people interact in the digital world, but also bring new opportunities and challenges for various industries. This article is based on swarm intelligence optimization algorithm to optimize the transformer model for predicting user VR experience.

## 2. Data set sources and data analysis

This article uses a private dataset for experimentation, which contains a lot of personal information and VR immersion experience of the participants. Each participant collected information such as age, gender, duration of VR headphones, wearing type, environment type, sound type, and VR immersion level. Select some data for display, and the results are shown in Table 1.

**Table 1.** Partial text data.

| Age | Gender | VRHeadset | Duration | Wear type | Environment type | Type of sound | ImmersionLevel |
|-----|--------|-----------|----------|-----------|------------------|---------------|----------------|
| 40  | 1      | 1         | 172      | 1         | 0                | 1             | 2              |
| 49  | 2      | 2         | 156      | 1         | 1                | 2             | 1              |
| 37  | 1      | 1         | 98       | 1         | 0                | 1             | 2              |
| 48  | 2      | 3         | 108      | 2         | 1.5              | 2             | 1              |
| 54  | 1      | 2         | 122      | 1         | 0                | 1             | 2              |
| 39  | 1      | 2         | 170      | 1         | 0                | 1             | 2              |
| 45  | 2      | 1         | 170      | 1         | 0                | 1             | 2              |

## 3. Method

### 3.1. The Arctic puffin optimization algorithm

The Arctic puffin optimization algorithm is an emerging swarm intelligence optimization algorithm inspired by the survival and foraging behavior of Arctic puffins in extreme environments. This algorithm simulates the collaboration and competition mechanisms of puffins in searching for food, forming an effective global optimization strategy [7]. The basic principle of PBOA is to view each individual as a possible solution, and gradually improve the quality of the solution through information sharing and communication between individuals. The algorithm mainly includes several key steps such as population initialization, fitness evaluation, individual updating, and selection.

In the specific implementation process, PBOA first generates a set of randomly initialized individuals, which represent potential solutions in the search space. Next, the algorithm will evaluate the fitness of

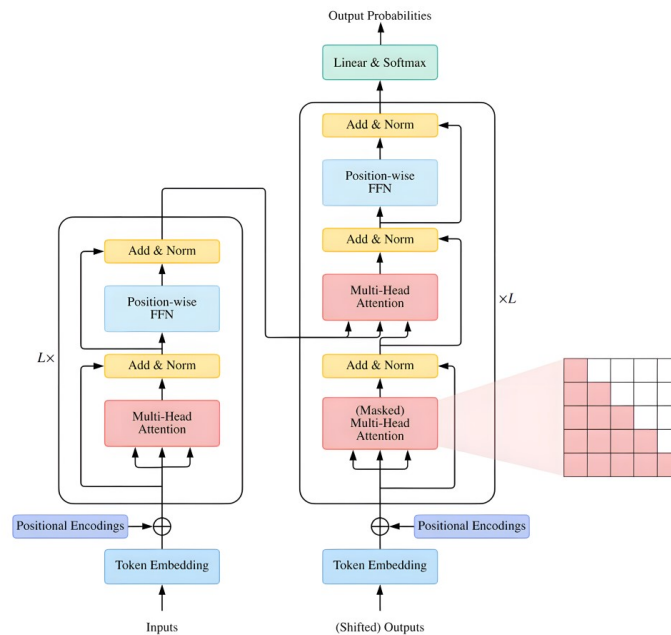
each individual, that is, their performance in the objective function. According to their fitness values, puffins will engage in "foraging" and "gathering", where individuals with higher fitness will attract other individuals to approach them, while also retaining a certain degree of randomness to avoid early convergence to local optima [8]. This process dynamically adjusts the interaction between puffins, allowing the entire population to explore effectively in the search space, thereby increasing the probability of finding the global optimal solution.

In addition, PBOA also introduces some adaptive mechanisms to enhance its performance in complex optimization problems [9]. For example, by setting dynamic adjustment parameters, the information transmission intensity and update strategy between individuals can be automatically changed based on the current search status, thus better balancing the relationship between exploration and development [10]. This makes the Arctic puffin optimization algorithm not only have strong global search capabilities, but also effectively utilize existing information for local refinement search.

### 3.2. Transformer

The Transformer algorithm is a deep learning model originally proposed by Vaswani et al. in 2017, mainly used for natural language processing (NLP) tasks [11]. Unlike traditional recurrent neural networks (RNNs) and long short-term memory networks (LSTMs), Transformers are entirely based on self attention mechanisms and can process input data in parallel, significantly improving training efficiency and effectiveness [12]. The core idea is to capture contextual information and generate richer representations by focusing on the relationships between various elements in the input sequence.

The basic structure of Transformer model includes two parts: encoder and decoder. The encoder consists of multiple stacked identical layers, each layer containing two main components: self attention mechanism and feedforward neural network [13]. The self attention mechanism allows the model to dynamically adjust its representation while considering the importance of other words in the input sequence when processing a certain word [14]. This process is achieved by computing the "Query", "Key", and "Value" vectors, ultimately generating a weighted contextual representation. The decoder is similar, but also adds attention to the previously generated output, allowing the model to utilize the generated information during generation. The network architecture of the transformer is shown in Figure 1.



**Figure 1.** The network architecture of the transformer.

In addition, Transformer also introduces positional encoding to address the issue of lacking sequence order information. Add positional encoding to each input vector to enable the model to distinguish words at different positions [15]. At the same time, in order to improve the training effectiveness of the model, Transformer adopts a multi head attention mechanism, which enhances the model's ability to capture information by parallel computing multiple self attention subspaces.

### 3.3. Transformer model optimized by swarm intelligence optimization algorithm

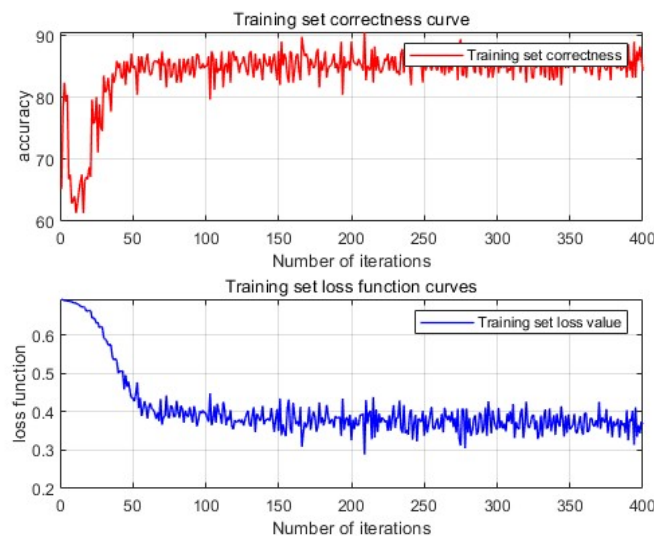
LSwarm Intelligence Optimization Algorithms are a type of algorithm that solves complex optimization problems by simulating group behavior in nature, including Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and others. The core idea of these algorithms is to find the optimal solution through mutual cooperation and competition among individuals [16]. When optimizing Transformer models, swarm intelligence algorithms can be used to adjust the hyperparameters, structural design, and learning strategies during the training process, thereby improving model performance.

In specific applications, swarm intelligence optimization algorithms first define a fitness function that is typically related to the performance of the Transformer model on specific tasks, such as accuracy or loss value [17]. Then, by initializing a set of "individuals" representing different combinations of hyperparameters such as learning rate, batch size, number of layers, etc., these individuals explore in the search space [18]. Taking particle swarm optimization as an example, each particle will update its historical best position and the best position of the entire population to move in the search space. As the number of iterations increases, the particles gradually converge to the optimal hyperparameter combination, effectively improving the performance of the Transformer model on specific tasks.

In addition, swarm intelligence optimization algorithms can also be combined with other technologies, such as transfer learning or data augmentation strategies in deep learning, to further enhance the effectiveness of Transformer models [19].

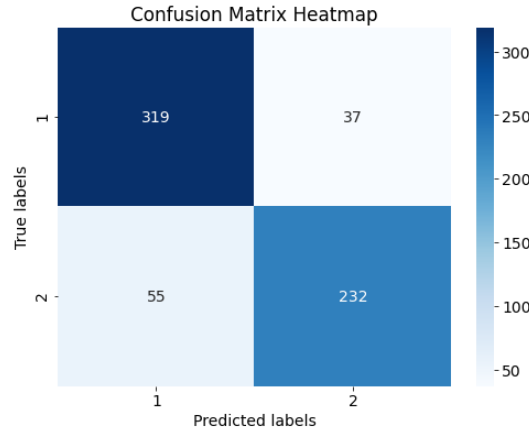
## 4. Result

In this experiment, Matlab R2022a was used for training. The training set and test set are randomly divided into two groups in a 7:3 ratio, with 25% of the data in the training set selected as the validation set. The maximum training time is set to 50, the GPU is set to 32GB, the maximum number of nodes in the tree is set to 5, and the maximum depth of the tree is set to 200. Output the accuracy and loss value variation curves of the training set, as shown in Figure 2. Output the confusion matrix of the training set, as shown in Figure 3, and output the confusion matrix of the test set, as shown in Figure 4.



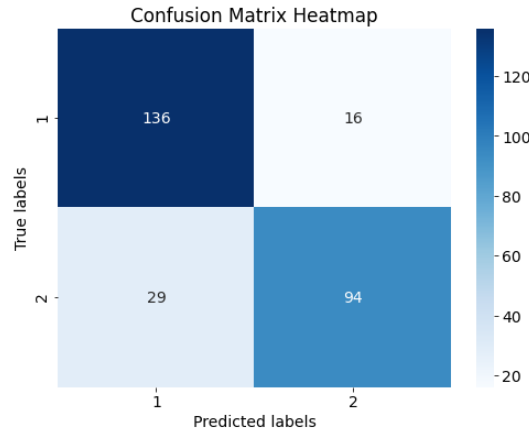
**Figure 2.** The accuracy and loss value variation curves.

From the changes in accuracy and loss, it can be seen that the accuracy of the training set gradually increased from 62.74% to 86.94% and stabilized, while the loss value gradually decreased from 0.71 to 0.37 and converged. It can be seen that the predictive performance of the model gradually improved in the training set.



**Figure 3.** Confusion matrix.

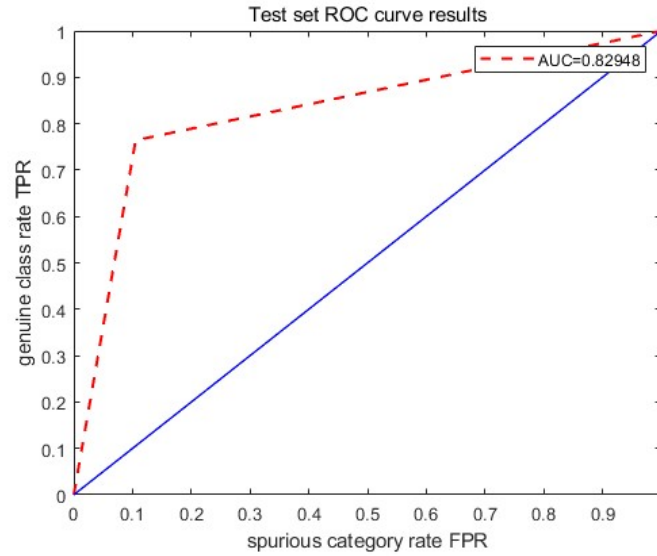
From the confusion matrix of the training set, it can be seen that 552 VR immersion predictions are correct and 92 are incorrect. Among them, 55 instances that were originally predicted as level 1 immersion were predicted as level 2 immersion, and 37 instances that should have been predicted as level 2 immersion were predicted as level 1 immersion. Overall, the accuracy of the training set is 85.69%.



**Figure 4.** Confusion matrix.

From the confusion matrix of the test set, it can be seen that 230 VR immersion predictions are correct and 45 are incorrect. Among them, 29 instances that were originally predicted as level 1 immersion were predicted as level 2 immersion, and 16 instances that should have been predicted as level 2 immersion were predicted as level 1 immersion. Overall, the accuracy of the test set is 83.63%.

Output the ROC curve of the test set, as shown in Figure 5, with an AUC value of 0.829.



**Figure 5.** The ROC curve of the test set.

## 5. Conclusion

This article optimizes the transformer model based on swarm intelligence optimization algorithm to predict users' virtual reality (VR) experience [20]. Through the analysis of the training and testing sets, we observed a significant improvement in the predictive performance of the model. During the training process, the accuracy of the training set gradually increased from 62.74% to 86.94%, and eventually stabilized, indicating that the model gradually mastered the data features through continuous learning and adjustment. Meanwhile, the loss value also decreased from 0.71 to 0.37, indicating that the model effectively reduced prediction errors during the optimization process. This process reflects the effectiveness of swarm intelligence optimization algorithms in processing complex data, providing a good foundation for subsequent research.

Further analysis of the confusion matrix in the training set revealed that 552 VR immersive experiences had correct prediction results, while 92 had errors. Among them, 55 instances that were originally predicted as level 1 immersion were mistakenly classified as level 2 immersion, while 37 instances that should have been predicted as level 2 immersion were mistakenly labeled as level 1. Although such classification errors exist, the overall accuracy of the training set reached 85.69%. In the test set, 230 VR immersive experiences were correctly predicted, while 45 showed deviations. Among them, 29 instances that should have been level 1 immersion were misclassified as level 2, and 16 instances that should have been level 2 immersion were marked as level 1. Nevertheless, the overall accuracy of the test set still reached 83.63%, and the AUC value under the ROC curve was 0.829, indicating that the model has strong predictive ability.

In summary, this study indicates that optimizing transformer models through swarm intelligence optimization algorithms can effectively improve the accuracy of predicting user VR experiences. This not only provides feasible methodologies for related fields, but also demonstrates the potential of artificial intelligence in user experience evaluation. With the development of technology, this data-driven approach is expected to play an important role in a wider range of application scenarios, providing support and guidance for the future development and application of VR technology. Therefore, in-depth exploration of the application of swarm intelligence algorithms in other fields will be of great significance and help promote the development of related technologies to a higher level.

## References

- [1] Gamage, Nisal Menuka, et al. "So predictable! continuous 3d hand trajectory prediction in virtual reality." The 34th Annual ACM Symposium on User Interface Software and Technology. 2021.

- [2] Fan, Ching-Ling, et al. "Fixation prediction for 360 video streaming in head-mounted virtual reality." Proceedings of the 27th workshop on network and operating systems support for digital audio and video. 2017.
- [3] David-John, Brendan, et al. "Towards gaze-based prediction of the intent to interact in virtual reality." ACM symposium on eye tracking research and applications. 2021.
- [4] Hu, Zhiming. "Gaze analysis and prediction in virtual reality." 2020 IEEE conference on virtual reality and 3D user interfaces abstracts and workshops (VRW). IEEE, 2020.
- [5] Petrangeli, Stefano, Gwendal Simon, and Viswanathan Swaminathan. "Trajectory-based viewport prediction for 360-degree virtual reality videos." 2018 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR). IEEE, 2018.
- [6] Singh, Avinash Kumar, et al. "Visual appearance modulates prediction error in virtual reality." IEEE Access 6 (2018): 24617-24624.
- [7] Gamage, Nisal Menuka, et al. "So predictable! continuous 3d hand trajectory prediction in virtual reality." The 34th Annual ACM Symposium on User Interface Software and Technology. 2021.
- [8] Wang, Jialin, et al. "Real-time prediction of simulator sickness in virtual reality games." IEEE Transactions on Games 15.2 (2022): 252-261.
- [9] Magalie, Ochs, Jain Sameer, and Blache Philippe. "Toward an automatic prediction of the sense of presence in virtual reality environment." Proceedings of the 6th international conference on human-agent interaction. 2018.
- [10] Jin, Weina, et al. "Automatic prediction of cybersickness for virtual reality games." 2018 IEEE Games, Entertainment, Media Conference (GEM). IEEE, 2018.
- [11] Feng, Xianglong, Yao Liu, and Sheng Wei. "LiveDeep: Online viewport prediction for live virtual reality streaming using lifelong deep learning." 2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR). IEEE, 2020.
- [12] \*\*, Weina, et al. "Automatic prediction of cybersickness for virtual reality games." 2018 IEEE Games, Entertainment, Media Conference (GEM). IEEE, 2018.
- [13] Wang, Jialin, et al. "Real-time prediction of simulator sickness in virtual reality games." IEEE Transactions on Games 15.2 (2022): 252-261.
- [14] Liu, \*\*aonan, and Yansha Deng. "Learning-based prediction, rendering and association optimization for MEC-enabled wireless virtual reality (VR) networks." IEEE transactions on wireless communications 20.10 (2021): 6356-6370.
- [15] Petrangeli, Stefano, Gwendal Simon, and Viswanathan Swaminathan. "Trajectory-based viewport prediction for 360-degree virtual reality videos." 2018 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR). IEEE, 2018.
- [16] David-John, Brendan, et al. "Towards gaze-based prediction of the intent to interact in virtual reality." ACM symposium on eye tracking research and applications. 2021.
- [17] Yan, Haodong, et al. "GazeMoDiff: Gaze-guided Diffusion Model for Stochastic Human Motion Prediction." arXiv preprint arXiv:2312.12090 (2023).
- [18] Anwar, Muhammad Shahid, et al. "Subjective QoE of 360-degree virtual reality videos and machine learning predictions." IEEE Access 8 (2020): 148084-148099.
- [19] Chen, Xiao-Lin, and Wen-Jun Hou. "Gaze-based interaction intention recognition in virtual reality." Electronics 11.10 (2022): 1647.
- [20] Sidenmark, Ludwig, Mathias N. Lystbæk, and Hans Gellersen. "Ge-simulator: An open-source tool for simulating real-time errors for hmd-based eye trackers." Proceedings of the 2023 Symposium on Eye Tracking Research and Applications. 2023.