Architecture Design of Teenager Mental Health Evaluation System Based on High Concurrency and High Availability

Yongxiong Zhang\(^1\)*, Xiangzheng Huang\(^1\) and Liangming Wang\(^2\)

\(^1\) Department of Economics and Trade, GuangZhou College of Technology and Business, Guangzhou, China
\(^2\) School of Software Engineering, South China University of Technology, Guangzhou, China
Email: csyxzhang@qq.com

Abstract. In order to solve the problem of high concurrency and stability of real-time collection and analysis of teenager mental health data, this paper proposes a high concurrency and high availability system architecture based on Docker container, choreography and deployment technology and artificial fish swarm algorithm. To solve the problem of low efficiency of Kubernetes default scheduling algorithm, this paper designs a parallel scheduling algorithm using artificial fish swarm algorithm. The experimental results show that the parallel scheduling algorithm is effective. For the complexity of container redundant operation, this paper designs a microservice rank scheme. Combined with AFS-FCM algorithm, the microservice rank of the system platform is determined, so that the redundant equipment can be automatically optimized for high availability.

1. Introduction
As the improvement of the material generating power, the quality of people’s life is also improved. However, with the material satisfaction, people’s psychological problems become more and more prominent. According to the Statistics of WHO [1], mental health problem become the fourth largest disease in the world. More than 300 million people around the world suffer from depression, a mental health disorder. People’s physical and mental health is greatly damaged. Mental illness is the main cause of suicide. “For decades, teenagers have been completely missing from national health plans,” said the WHO assistant director general. If we can find out the teenager with mental health problems such as self-injury tendency, suicide or depression in time, it will play a positive and important role in solving the mental health problems of teenager.

The research on “mental health status of teenagers” mainly focuses on the mental health status of teenagers, analyzes the causes and studies the corresponding countermeasures, etc. For example, American scholars Greenspoon and Saklofske [2] proposed the Dual-Factor Model of Mental Health (DFM), then Suldo [3] and others improved DFM. Thalji [4] integrated the subjective well-being index to break through the condition that mental health degree was evaluated according to a single standard of psychopathology. These results left important clues for future research. But there are few researches on how to combine the application of new technologies such as big data and cloud computing to analyze the mental health status of teenagers in real time, especially on how to effectively evaluate the mental health of teenagers. Therefore, it is urgent to integrate relevant new technologies to build an effective system to make up for the lack of current research.
Since the data of teenager mental health need to be collected and analyzed in real time, the amount of concurrent access is high, it requires the system to have the characteristics of high concurrency and high availability. High concurrency can be achieved by dynamic load balancing, container cluster elastic scaling and container resource scheduling. High availability can be achieved by appropriate redundancy policies. In this paper, Docker technology is used in combination with artificial fish swarm algorithm (AFSA) to design an architecture of teenager mental health evaluation system with high concurrency and high availability, which provides a reliable technical architecture scheme for the subsequent research on teenager mental health.

The rest of this paper is organized as follows. Section II introduces the Docker container and choreography deployment technology and artificial fish swarm algorithm. The third section introduces the design of high concurrent system architecture. Then, high availability system architecture design was introduced in section IV. Finally, the conclusion is presented and future work is discussed.

2. Descriptive Analysis for Background Theory

2.1. Docker Container and Choreography Deployment Technology

Early application development generally uses the application architecture mode of single architecture. However, with the increase of data requests and the complexity of functions, the cost of machine resources and maintenance time is very high. In recent years, microservice architecture has emerged in the software architecture pattern, and image deployment based on Docker container is the mainstream application and deployment of microservice architecture.

Docker is an open source project based on Go language, using C/S architecture. As the core background process, Docker Daemon is responsible for converting the requests of Docker Client into system calls, so as to complete the management and use of containers and images. The Docker container shares the operating system kernel with the host, so it consumes less resources. Compared with traditional virtual machines, Docker has the advantages of high speed of start, small occupation of hard disk space, better performance and large system support [5].

For Docker container, because Swarm is prone to be affected performance due to resource shortage caused by scheduling policy problems, Kubernetes is adopted as the choreography deployment technology in this paper. Kubernetes consists of multiple nodes and Master nodes responsible for node management and resource scheduling management. Kubernetes can manage the container cluster efficiently, and also realize the functions of customization. However, its Service policy may affect the concurrency performance of the system due to the imbalance of container instances. The default scheduling algorithm of Kubernetes can achieve the optimal scheduling of a single container, but the scheduling efficiency is not high in the application of multiple containers [6, 7].

2.2. Artificial Fish Swarm Algorithm

The artificial fish swarm algorithm [8] is studied by Dr. Li Xiaolei on the behavior of fish swarm, which has the advantages of small sensitivity to initial value, good global convergence and strong robustness.

Assume that the current state of an artificial fish is $S$, $\text{Vision}$ is its field of vision, and $S_v$ is the position of the viewpoint at a certain time. If $S_v$ is better than $S$, it will move one step towards the direction of $S_v$, that is, it will reach state $S_{\text{next}}$. Otherwise, it will continue to patrol other positions in $\text{Vision}$. Formula is as follows:

$$S_v = S + \text{Vision} \cdot \text{Rand}$$  \hspace{1cm} (1)

$$S_{\text{next}} = S + \frac{S_v - S}{||S_v - S||} \cdot \text{Step} \cdot \text{Rand}$$  \hspace{1cm} (2)

where Rand is a random number from 0 to 1, and Step is the move Step.

There are four basic behaviors for artificial fish in artificial fish swarm algorithm.
2.2.1. AF-Prey. Assuming that the current state is $S_i$, randomly select a state $S_j$ within the range of its perception. If the food concentration is $Y_i < Y_j$, move further in the direction of $S_j$; otherwise repeatedly try to set the number times. If the forward condition is still not met, move one step randomly.

2.2.2. AF-Swarm. Assuming the current state is $S_i$, the number of artificial fish in field ($d < \text{Vision}$) is N, and the center position is $S_c$. If $Y_c / N > \delta Y_i$, i.e. the center of artificial fish has a high food concentration and is not too crowded, then move one step in the direction of $S_c$, otherwise, the foraging behavior will be performed.

2.2.3. AF-Follow. Assuming that the current state is $S_i$, the number of artificial fish in field ($d < \text{Vision}$) is N, and the state of $Y_j$ with the largest food concentration is $S_j$, if $Y_j / N > \delta Y_i$, that is, the state of $S_j$ has a higher food concentration and is not too crowded, then move one step towards the direction of $S_j$, otherwise, the foraging behavior will be performed.

2.2.4. Random Act. Within the range of Vision, the artificial fish randomly select a state, and then move in that direction.

In order to achieve a higher food concentration, the four behavior of the artificial fish can be transformed according to the corresponding conditions, and then the optimal behavior can be evaluated.

3. Architecture Design of High Concurrency System

3.1. Analysis of Kubernetes Container Default Scheduling

Kubernetes [9] implements the scheduling process by default using the serial scheduler of GenericScheduler. All containers to be scheduled form a pod queue, and all optional nodes form a node queue. According to the pre-selected policy, the optimal node is screened.

Kubernetes scheduling process: the scheduler takes out the Pod object, selects the optimal available node, binds the Pod object to the node, and stores the result in the database of etcd. According to the pre-selected rule, the nodes that do not meet the conditions on the available physical machine nodes are filtered. Then, the nodes that meet the primary selection are scored, and the optimal node is selected for node scheduling. The start of the container requires the node Kubelet to periodically monitor the database of etcd to check if the container has an image. If not, you need to go to the repository registry to pull the image. Finally, once the container is created successfully, it can be used to provide external services.

The default scheduler of Kubernetes is a serial scheduling model. It is inefficient and cannot maximize the utilization of physical machine resources.

3.2. Parallel Scheduling Analysis of Artificial Fish Swarm Algorithm

Because default scheduling algorithm of Kubernetes is inefficient and cannot take into account the global problem, this section designs a parallel scheduling scheme combined with artificial fish swarm algorithm. This scheme can take the node as the location of the artificial fish, and each artificial fish as the task scheduler needs to give the scheduling scheme of the container to be scheduled to the node. The process of the algorithm is as follows:

**Step 1:** parameter initialization, including the initial position of artificial fish, host node data, artificial fish visual field, step size, attempt times and crowding factor, etc.
Step 2: according to the food concentration, the scheduling scheme of the artificial fish for the container is recorded. The resource data of the host node is updated, and the scheduling of the next container is continued until all the containers are scheduled. Then the feasible solution of the scheduling task obtained by the artificial fish is recorded.

Step 3: select another artificial fish in the fish group. If existing the artificial fish that has not been selected before, then operation 2 will be performed; otherwise, the iteration will end.

3.3. Parallel Scheduling Test of Artificial Fish Swarm Algorithm
The simulation uses the data of 8 host nodes and 8 containers to be scheduled. The number of artificial fish iterated is 4. Kubernetes’ default algorithm stops iterating in the 5th iteration, while the artificial fish swarm algorithm stops in the 31st iteration. Although the time complexity of artificial fish swarm algorithm is not as good as kubernetes’ default algorithm, it is easier for containers of the same web service to be dispersed to different host nodes, and the utilization of node resources is more balanced, thus improving the scheduling efficiency of containers.

3.4. High Concurrency Design of System Architecture
In this paper, Kubernetes and Docker container technologies are used to realize the design of the high concurrency system architecture, which is mainly composed of load balancing module, cluster module and performance monitoring module. The high concurrency system architecture is shown in figure 1.

![Figure 1. High concurrency system architecture.](image)

The load balancing module is responsible for forwarding high concurrent request to the cluster. Cluster module includes container cluster, host cluster and parallel scheduling module of artificial fish swarm algorithm. Container cluster realizes web service with Docker technology. The host cluster is the running carrier of the container cluster. The parallel scheduling module realizes the scheduling task of the container. Kubernetes manages the cluster module. The performance monitoring module realizes the real-time monitoring of the performance indicators of the cluster module and automatically performs the optimal adjustment.

4. Architecture Design of High Availability System

4.1. Clustering is an Important Technology in Data Analysis
In this paper, artificial fish swarm algorithm and fuzzy C-mean (FCM) algorithm [10] are combined to determine the rank of microservices on the system platform.

Set the microservice object $S = \{s_1, s_2, s_3, \ldots, s_n\}$ as the whole of the cluster analysis object, and define $u_{ij}$ as the membership degree of Object $s_j$ belonging to class $j(1 \leq j \leq c)$. Fuzzy clustering of the object $S$ is represented by fuzzy matrix $U = (u_{ij})$. $u_{ij}$ represents the element in row $i$ and column...
j of the matrix, indicating that the element i belongs to the membership degree of class j. The fuzzy partition space of S is as follows:

\[ M_k = \left\{ \left. U \right| u_{ik} \in [0,1], \forall i, k; \sum_{i=1}^{n} u_{ik} = 1; 0 < \sum_{k=1}^{c} u_{ik} < n, \forall i \right\} \]  

(3)

The objective function of fuzzy C-means algorithm is as follows:

\[ J_{FCM} = \sum_{k=1}^{c} \sum_{i=1}^{n} u_{ik}^m \| v_i - s_k \|^2 \]  

(4)

In equation (2), m means the fuzzy index, \( m > 1 \), normally \( m = 2 \), \( v_i \) means the center of class i.

\[ u_{ik} = \sum_{j=1}^{c} \left( \frac{v_i - s_j}{\| v_i - s_j \|} \right)^{\left( \frac{2}{m-1} \right)} \]  

(5)

\[ V = \frac{\sum_{k=1}^{c} u_{ik}^m s_k}{\sum_{k=1}^{c} u_{ik}^m} \]  

(6)

Since the fuzzy c-means algorithm is particularly sensitive to the initial value, if the center of clustering is not selected properly during initialization, it will fall into the problem of local optimization and not global optimization. However, the artificial fish swarm algorithm is insensitive to the initial value, and has the advantages of global optimization and parallel rapidity. Hence, in order to overcome the sensitivity of fuzzy C-means algorithm to initial value, this paper combines artificial fish swarm algorithm with fuzzy C-means algorithm.

The specific process of artificial fish swarm fuzzy clustering algorithm (AFS-FCM algorithm):

**Step 1:** initialize the parameters of the fish swarm algorithm, including the number of individuals in the artificial fish swarm, the state of randomly generating the best artificial fish, the initial position, field of vision, step size, number of iterations, crowding factor and the threshold value of swallowing behavior of each artificial fish.

**Step 2:** calculate the fuzzy matrix U according to formula (5);

**Step 3:** calculate each cluster center V according to formula (6);

**Step 4:** calculate the fitness value of each artificial fish according to formula (4), perform the behavior of artificial fish, and update the position status of each artificial fish;

**Step 5:** update the status of the optimal artificial fish;

**Step 6:** display the result if the stop condition is met, otherwise go to step 2.

4.2. Container Deployment Strategy of Microservice Rank

In order to ensure the high availability of the system platform, this paper adopts the scheme of multi-container instance redundant backup deployment for the system microservices. Using the AFS-FCM algorithm, the redundancy set of the container is \( Rd = \{ d_1, d_2, d_3, \ldots, d_q \} \), \( h_i \in Rd, i < n \) where \( d_1 < d_2 < d_3 < \cdots < d_q \), \( d_q \) has the highest redundancy, the lower corner mark q is consistent with the number of microservice ranks. For the set of microservice rank, there is unique element and its corresponding in the set of redundancy, according to the corresponding relationship f between higher level of microservice and higher degree of redundancy. It is recorded as \( f : \text{Rank} \to Rd \). Therefore, the sequence of microservice redundancy can be obtained as \( H = \{ h_1, h_2, h_3, \ldots, h_n \} \), where n represents the number of the microservice, \( h_i \in Rd, i < n \). Finally, the Docker container is opened or stopped according to the redundancy sequence.
4.3. High Availability Design of System Architecture

In the design of microservice architecture, the redundancy backup technology is generally used to build a highly available cluster, and the horizontal extension technology is used to expand the cluster. But the redundancy backup and horizontal expansion need to consume more machine and time cost. Because Docker container can realize resource isolation with the help of the virtualization of operating system and greatly improve resource utilization, this paper uses Docker container to achieve redundant backup and horizontal expansion requirements. So that the high availability of the system can be guaranteed.

Due to the large number of microservices in teenager mental health evaluation system, different business logic and diversity, the operation will be very complicated in the process of carrying out redundant backup of microservices to guarantee the high availability of the system. Therefore, this paper proposes a microservice rank scheme to solve the above problems. Because of the uncertainty of the evaluation standard of the microservice rank and the fuzziness of the microservice rank of the system platform itself, AFS-FCM algorithm was used in this paper to determine the microservice rank of system platform. Then the redundant devices were adjusted on and off according to the grade to ensure the high availability of the system under the condition of reasonable resource utilization. The high availability system architecture is shown in figure 2.

The redundancy of system platform microservice in container cluster depends on its own rank of microservice. The resource monitoring module is responsible for monitoring the resource usage of the container cluster, controlling the automatic on-off operation of redundant containers on the service nodes, and feeding back to the rank judgement module to update the microservice rank, thus completing the automatic high availability tuning process. Rank judgement module uses AFS-FCM algorithm to divide rank. It has the advantages of global optimization and parallel rapidity, and can ensure the stability and robustness of the system.

5. Conclusion

In this paper, Docker container, scheduling and deployment technology and artificial fish swarm algorithm are used to design a scheme aiming at high concurrency and high availability of teenager mental health evaluation system platform. In the high concurrency scheme, the artificial fish swarm algorithm is used to improve the default algorithm of Kubernetes. Even though the time complexity is increased, the utilization of node resources is more balanced and the scheduling efficiency of containers is improved. In the high availability scheme, in order to reduce the complexity of redundant operations, the microservice rank method is proposed, and the artificial fish swarm fuzzy clustering algorithm is used to determine the microservice rank of system platform, and then the redundant
equipment is adjusted on and off according to the rank to ensure the high availability of the system under the condition of reasonable resource utilization.

In the next research stage, we will continue to study the improved artificial fish swarm algorithm in the scheduling container to reduce the time complexity while ensuring high scheduling efficiency. At the same time, we will continue to optimize the AFS-FCM algorithm to improve the efficiency and accuracy of microservice rank clustering. Finally, high concurrency and high availability schemes will be applied to the platform of teenager mental health evaluation system to improve the reliability of the system.

Acknowledgments
This research was financially supported by the Characteristic Innovation Project (Natural Science) of Department of Education of Guangdong Province in 2017 (Subject No.: 2017KTSCX202); Project of philosophy and Social Sciences Planning of Guangzhou in 2020 (Subject No.: 2020GZGJ295).

References
[1] The Statistics of WHO https://www.who.int/.
[2] Greenspoon P J and Saklofske D H 2001 Toward an integration subjective well-being and psychopathology Social Indicators Research 54 (1) 81-108.
[3] Suldo S M, Shaffer E J and Riley K N 2008 A social-cognitive-behavioral model of academic predictors of teenagers’ life satisfaction School Psychology Quarterly 23 (1) 56-69.
[4] Thalji A L 2012 A Dual Factor Model of Mental Health in High School Students: Group Characteristics and Social Functioning (University of South Florida).
[5] Joy A M 2015 Performance comparison between Linux containers and virtual machines Proceedings of the 2015 International Conference on Advances in Computer Engineering and Application (ICVCEA)(IEEE) pp 342-346.
[6] Felter W, Ferreira A, Rajamony R and Rubio J 2015 An updated performance comparison of virtual machine and linux containers Proceedings of the 2015 IEEE International Symposium on Performance Analysis Systems and Software (ISPASS) pp 171-172.
[7] Adufu T, Choi J and Kim Y 2015 Is container-based technology a winner for high performance scientific application? Proceedings of the 2015 17th Asia-Pacific Network Operations and Management Symposium (APNOMS) (IEEE) pp 507-510.
[8] Chao Z, Feng-Ming Z, Fei L and Hu-Sheng W 2014 Improved artificial fish swarm algorithm 2014 9th IEEE Conference on Industrial Electronics and Applications (IEEE) pp 748-753.
[9] Kubernetes https://kubernetes.io/.
[10] Al-Ayyoub M, Al-Andolita M, Jararweha Y, Smadia M and Guptab B 2019 Improving fuzzy C-mean-based community detection in social networks using dynamic parallelism Computers & Electrical Engineering 74 533-546.