

Retraction

Retracted: The Construction of a Smart Education System in Colleges and Universities Based on Cloud-Side Collaborative Computing Task Scheduling Algorithm

Journal of Sensors

Received 19 December 2023; Accepted 19 December 2023; Published 20 December 2023

Copyright © 2023 Journal of Sensors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

- (1) Discrepancies in scope
- (2) Discrepancies in the description of the research reported
- (3) Discrepancies between the availability of data and the research described
- (4) Inappropriate citations
- (5) Incoherent, meaningless and/or irrelevant content included in the article
- (6) Manipulated or compromised peer review

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

- [1] L. Yu, "The Construction of a Smart Education System in Colleges and Universities Based on Cloud-Side Collaborative Computing Task Scheduling Algorithm," *Journal of Sensors*, vol. 2022, Article ID 8949974, 12 pages, 2022.

Research Article

The Construction of a Smart Education System in Colleges and Universities Based on Cloud-Side Collaborative Computing Task Scheduling Algorithm

Lingyun Yu 

Academic Affairs Office, Xiamen University of Technology, Xiamen 361024, China

Correspondence should be addressed to Lingyun Yu; yulingyun@xmut.edu.cn

Received 1 December 2021; Revised 2 January 2022; Accepted 15 January 2022; Published 10 February 2022

Academic Editor: Wei Zhang

Copyright © 2022 Lingyun Yu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to improve the resource scheduling efficiency of the smart education system in colleges and universities and improve the stability and security of educational resource processing, this paper combines cloud-side collaborative computing technology to improve the cloud computing task scheduling algorithm and uses the improved algorithm for task scheduling. Moreover, this paper combines the global optimization and local optimization processes to make the final optimal solution performance better and further reduce the task scheduling execution time in the cluster, thereby improving the computing efficiency of cloud computing. In addition, this paper combines the improved algorithm to construct the system structure framework and verifies the system performance through experiments. The experimental research results show that the smart education system in colleges and universities based on the cloud-side collaborative computing task scheduling algorithm constructed in this paper is very good in all aspects and has a greater advantage in processing educational resources.

1. Introduction

Today, when Internet information technology is becoming more and more mature, the concept of teaching interaction is becoming more and more abundant, which also makes the choice of learning methods and content more diversified. At present, it is more and more common to use portable mobile devices to realize the teaching interaction of courses. In evaluation theory and interactive teaching theory, both students and teachers are the subjects of evaluation. Moreover, students' evaluation of their own learning achievements is mainly through mutual evaluation and self-evaluation, and students can also evaluate teachers' teaching behavior [1]. Such technical means can not only help students form better learning habits and strong learning interests but also enable teachers to more intuitively understand the learning situation of each student. In particular, this method is even more important when most teachers are still subjectively evaluating the learning behavior of students, making it possible to evaluate what was previously considered impossible [2]. Cloud storage [3] is a new network stor-

age technology extended and developed on the basis of cloud computing. Under the premise of high-speed network and Web 2.0 technology, it uses cluster technology, distributed file system technology, network technology, data compression technology, and other related technologies to connect a large number of hardware storage devices of various types, scattered in various places, and work together, and provides data storage, access, and other functions by forming an expandable storage resource pool.

This article combines the cloud-side collaborative computing task scheduling algorithm to construct a smart education system in colleges and universities, analyzes the process of smart education in colleges and universities, and further improves the effect of smart education on this basis.

2. Related Work

The rise of the concept of cloud computing drives the development of cloud storage technology. The cloud storage service models provided by cloud storage service providers are mainly divided into two types [4]: one is large-capacity file

sharing, which focuses on the storage and sharing of large files by users; the other one is cloud synchronous storage, which is the synchronous storage of files. However, due to the chaos in the cloud storage market and because there are so many cloud storage products, it is difficult to define the two storage modes [5]. Dropbox is an online file storage service developed by Dropbox. It synchronizes files through cloud computing and other advanced technologies in a network environment. Users can store and share various files and folders and support different operating systems and multiple smart clients [6]. Google Drive is an online cloud storage service launched by Google. It can not only provide free capacity space but also pay for expansion; at the same time, Google Docs is built-in, and users can collaborate with others in real time; in addition, Google will also provide third-party provider API interface, allowing people to save content from other programs to Google Drive [7]. OneDrive (formerly known as Sky Drive) is a cloud storage service launched by Microsoft, capable of file storage, file management, permission control, online office, and other services [8]. It can access the research status of informatization and intelligence in the field of education at home and abroad through multiple clients. Literature [9] mainly studies how to achieve system integration and resource sharing and improve hardware utilization, but it lacks research on massive data processing. Literature [10] proposes to solve the problem of information islands of enterprises with a hybrid architecture model of the two, but there is no introduction to resource integration and on how to improve resource utilization. Literature [11] proposed a cloud-based virtual desktop system to study the use of virtual classrooms in the education field. Literature [12] proposes to integrate knowledge organization and knowledge management and treat knowledge as a service (KAAS), demonstrating how this concept can be used in British universities. Literature [13] mainly studies the sharing and utilization of resources, but it does not solve the problem of unbalanced utilization of resources. Literature [14] proposes to model the complex causal relationship in the system by constructing a system model and verify the result-factor relationship existing in the system model through model analysis. Literature [15] proposes a new teaching system that integrates Web Services of courseware resources through the SOAP protocol. Literature [16] summarizes the transformation from traditional education informatization to education services under the cloud computing environment and the computing architecture of cloud education.

For the problems of mutual independence and closure of education systems, most of the researches are carried out on the integration of heterogeneous system resources of SOA, and the operability of interconnection between systems is also studied through a large number of aspects such as integration, redevelopment, and adaptation. Literature [17] describes the architecture and basic characteristics of modern distance education based on cloud computing, analyzes the changes that distance education cloud brings to education and teaching methods, and looks forward to its deployment. Literature [18] summarizes the fundamentals of education cloud advanced features such as efficient facility organization, educational resource sharing, reduction of

R&D investment, and management and operation cost savings. Computer science disciplines are more concerned with issues such as the infrastructure and implementation technology of educational cloud services. Literature [19] proposed a cloud architecture model based on the shared content object reference model, which breaks through the solidified learning methods of traditional learning systems and realizes the unified storage and flexible distribution of educational resources such as courses and learning materials by constructing content access middleware. Literature [20] describes the transformation from traditional E-learning to cloud services and cloud service architecture and discusses the commercialization model of education cloud services and the potential risks of their promotion and operation.

3. Educational Resource Cloud Task Scheduling

The task scheduling process in the cloud computing system involves the user level, task scheduling mechanism, virtual machine level, and data center level. The user submits the task, and the service agent connecting the user and the virtual machine will assign the task to the virtual machine to execute the task according to the scheduling policy provided by the task scheduling mechanism. The process is shown in Figure 1.

Task scheduling is to map tasks submitted by users with virtual machine resources in the data center, and the mapping process is implemented through a certain algorithm. The cloud computing task scheduling process is based on the process of performing tasks on virtual machines. It can be divided into two levels of task scheduling: the first level scheduling is mainly the scheduling between the user and the virtual machine, and the second level scheduling is mainly the scheduling between the host and the virtual machine. Figure 1 shows the first-level scheduling process.

The difference between the task scheduling process in cloud computing and the task scheduling process in other systems is caused by the particularity of the cloud computing platform. First of all, the network environment in which cloud computing is located determines the diversity of resources available in the task scheduling process. Second, the distributed nature of cloud computing determines the larger scale of task scheduling. Finally, the diversity of cloud computing use groups determines the difference. The task scheduling must comply with different constraints.

In the process of task scheduling, cloud computing needs to consider various requirements put forward by users, such as how to reduce costs, improve resource utilization, and improve work efficiency. At present, the optimization goals that are more considered in the process of cloud computing task scheduling are as follows:

- (1) *Time*: when taking time as the optimization goal. The time here can be considered from many aspects, such as the completion time of the task. Average completion time: both users and cloud computing service providers hope to complete their work in a relatively small amount of time or to complete more work in the same time, so time is an important optimization goal in the task scheduling process.

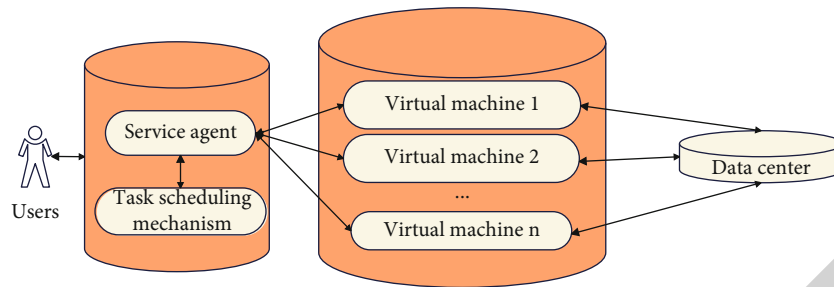


FIGURE 1: Flow chart of user acquisition service in cloud computing.

- (2) *Resource utilization*: resource utilization refers to the rational allocation of physical resources in the process of task scheduling to maximize the use of virtual machines in the system. The improvement of resource utilization is also a way to improve work efficiency, so resource utilization is a very important optimization goal in the task scheduling process.
- (3) *Cost*: cost refers to the amount of expenditure required for the user to submit the job task for processing on the virtual machine. The cost issue is the common concern of users and suppliers, so the cost issue is worth studying.
- (4) *System load*: system load rationalization can ensure that the system virtual machines are allocated reasonably, that is, to ensure the task completion progress, but also to improve the work efficiency of the virtual machines, which depends on the pros and cons of the scheduling algorithm used by the system. Therefore, the system load is one of the optimization goals, and its status is self-evident.

The chaotic cat group algorithm used in the scheduling model of the chaotic cat group algorithm proposed in this paper is an improvement based on the cat group algorithm. Among them, at the beginning of the implementation of the chaotic cat group algorithm, the cat group needs to be initialized. According to the grouping rate MR (mixture ratio), the cat group is divided into the search mode and the tracking mode. The search mode can be regarded as a local optimization, and the tracking mode can be regarded as a global optimization. After two optimization processes, the optimal solution is finally obtained. Cats will have different activity states in the two modes. In the search mode, the speed and position of the cat are updated through a series of operations according to the mode state of the cat at this time. According to the fitness function developed in this paper, the fitness value of the cat at the current position is calculated, and the optimal value is retained. In the tracking mode, update the speed component and position component of the cat and check whether the value exceeds the set maximum and minimum range. For those within the range, no processing is performed, the individual values outside the range are set to the corresponding boundary values through mapping transformation, and the individual fitness value after the

position change is calculated. More detailed operation steps of the two modes will be given below. The algorithm compares the cat population fitness values in the two modes and selects the largest fitness value in the population and keeps it as the optimal solution for this iteration. After that, the algorithm compares the experimental results according to the end conditions set in the algorithm in advance. If the requirements are met, the program will end, and if the requirements are not met, we continue to the next step. In the next step, the cat in the search mode and the cat in the tracking mode will exchange information, and the cat's position information will be chaotically mapped. The chaotic mapping is mainly to improve the diversity of the population and avoid the algorithm "premature." The population after chaos mapping has returned to the mode division, and the next generation of population optimization will be started from then on. The specific solution steps of the cat group algorithm are simplified to the following steps:

- (1) The algorithm initializes the cat group, that is, initializes the position t and velocity component of the cat group, the population size is N , the maximum number of iterations K_{max} , the grouping rate (mixture ratio MR), and so on. The range of activities of the cat group is $[a, b]$. The size of the grouping rate MR determines the number of cats in the tracking mode. According to the actual situation, most cats are in a lazy state, most cats are in search mode, and a few are in tracking mode, so the value of MR is generally set to be small
- (2) The algorithm calculates the fitness value of all cats in the population and records the maximum fitness value
- (3) The algorithm randomly groups the cat group according to the MR grouping rate: it is judged whether the cat is in the tracking mode or the searching mode in the next step
- (4) Tracking mode
- (5) Search mode
- (6) The algorithm calculates the fitness values of the individuals in the two modes and selects the individual with the largest current optimal fitness

- (7) The algorithm judges whether the termination condition is met at the moment. If it is satisfied, the algorithm outputs the optimal solution and ends the procedure: otherwise, use the chaotic map to update the position of the cat and repeat the processes (3)-(6) for optimization and iterative processing. The general algorithm has two termination conditions: the number of iterations reaches the upper limit and the optimal solution meets the requirements. This article uses the maximum number of iterations to be executed

The flowchart of the chaotic cat swarm algorithm is shown in Figure 2 [21].

The cloud computing task scheduling process can be divided into two levels of task scheduling: the first level is to complete the scheduling between the user and the virtual machine and the second level is to complete the scheduling between the host and the virtual machine. In the task scheduling process of cloud computing, the task division between users and virtual machines needs to be considered, and the scheduling between hosts and virtual machines needs to consider the mapping relationship between them. The task scheduling studied in this paper belongs to the first-level scheduling. In this task scheduling process, the large task submitted by the user is divided into n smaller subtasks. Then, using task scheduling strategies and following the principles of load balancing, optimal span, QoS, etc., these subtasks are reasonably allocated to the corresponding m computing resource nodes for execution ($n > m$). The scheduling model is shown in Figure 3.

As shown in Figure 3, the task scheduling process involves users, tasks, schedulers, and virtual machines. The user submits the task, and the system divides the large-scale task into multiple subtasks and distributes the tasks to the designated virtual machines for execution according to the scheduling strategy in the scheduler. The scheduling mechanism mentioned here refers to different scheduling algorithms proposed by users according to their own needs.

3.1. Task Execution Time Function. From the task scheduling model established above, in the task scheduling process, n tasks submitted by users are allocated to m virtual machines, and the number of tasks n is much larger than the number of virtual machines m . The virtual machine list is defined as $VMlist = \{VM1, vW2, VM3, VM4, \dots, vwm\}$, where m is the number of virtual machines. The list of all submitted task length sets is defined as $TaskLengthlist = \{TaskLength1, TaskLength2, TaskLength3, TaskLength4, \dots, TaskLengthn\}$, where $TaskLength$ is the length corresponding to each task. The execution time of a task submitted by the user on the virtual machine is set to $Exetime$, and the formula is as follows:

$$Exetime = \frac{TaskLength}{VMdeal}, \quad (1)$$

Among them, $VMdeal$ is the computing power of each virtual machine, and its calculation formula is shown in the following formula:

$$dealE_j = proNum_j * MIPS_j + BW_j. \quad (2)$$

Among them, $proNum_j$ represents the number of CPUs of each virtual machine, $MIPS_j$ represents the instructions processed by the CPU per second, and BW is the virtual machine network bandwidth. In the actual task scheduling process, due to the randomness of task allocation, each task may be allocated to more than one virtual machine. Therefore, in the calculation of the virtual machine time, it is not possible to simply add up. Now, we give the total time $_{ij}$ when all tasks are executed on the virtual machine, and the formula is shown as follows:

$$Totaltime_{ij} = \sum_{j=1}^n \left(\text{avg} \left(\sum_{i=1}^m Exetime_i \right) \right). \quad (3)$$

It means that the running time of a task on multiple virtual machines is averaged first, and then, the execution time of all tasks is summed.

3.2. Load Balancing Function. The load of a virtual machine is defined as follows: in a task allocation scheme X , the load $rTime_j$ of a virtual machine is the task execution time allocated to all tasks on the j -th virtual machine. The formula is shown as follows:

$$rTime_j = \frac{VMload_j}{dealE_j}. \quad (4)$$

Among them, $VMload$ is the sum of all task instruction lengths on the virtual machine, and its formula is shown as follows:

$$VMload_j = \sum_{j=1}^n TaskLength_j. \quad (5)$$

Therefore, the load value of all virtual machines is as follows:

$$VMsload = \sum_{j=1}^m VMload_j. \quad (6)$$

According to the computing power $dealE_j$ of each virtual machine mentioned above, the maximum computing power of the system is shown as follows:

$$dealE_{sum} = \sum_{j=1}^m dealE_j. \quad (7)$$

The average calculation time of all virtual machines in the system is defined as $rTime_{avg}$, and its formula is shown as follows:

$$rTime_{arg} = \frac{VMsload}{dealE_{sum}}. \quad (8)$$

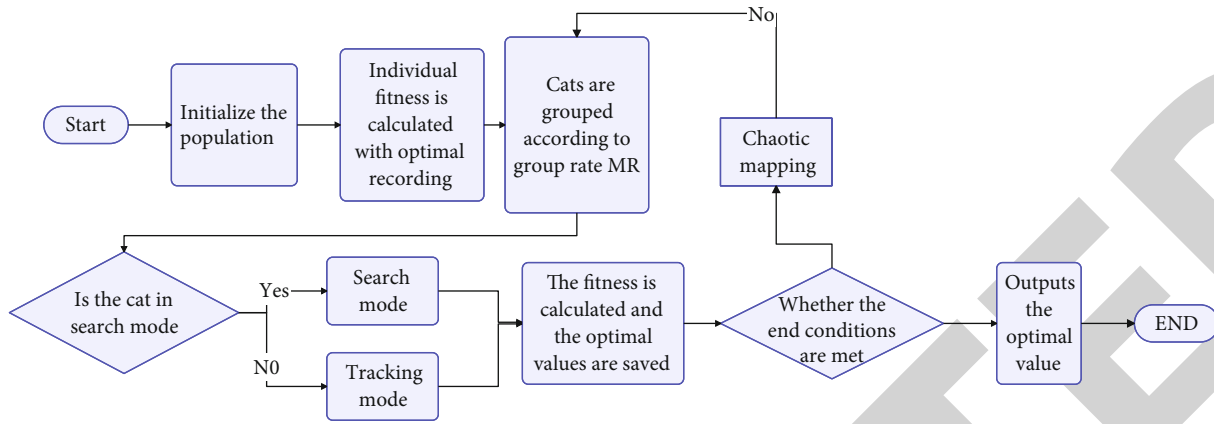


FIGURE 2: Algorithm flow chart.

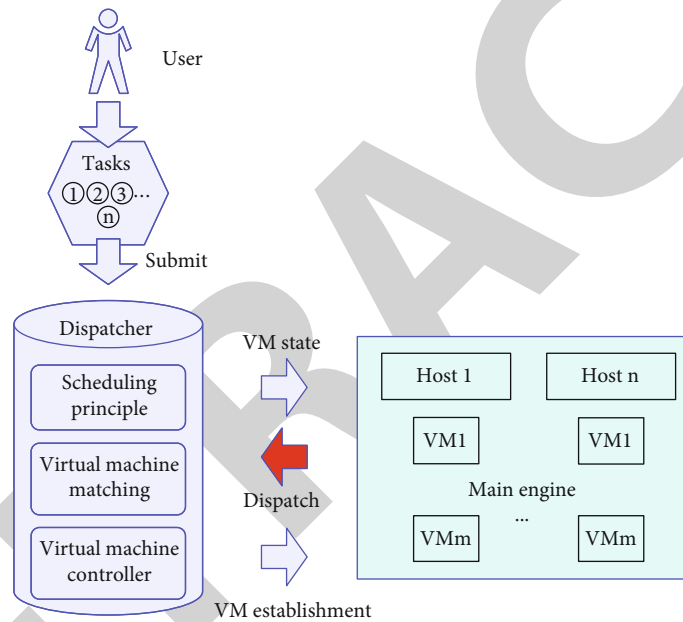


FIGURE 3: Scheduling model diagram.

First, the system load balance degree is given, and its function is shown as follows:

$$\text{Loadsys} = \sqrt{\frac{1}{m} \sum_{j=1}^m (r\text{Time}_j - r\text{Time}_{\text{arg}})^2} \quad (9)$$

Loadsys is not only the standard deviation of the system load; it also represents the load balance of the system. In order to make the system more balanced, Loadsys should be made as small as possible.

In the process of multiobjective task scheduling in this paper, two objectives are selected as the optimization objects, and the objective function is formulated according to the optimization objectives to achieve the optimization effect. The designed objective function is mainly used to calculate the size of the individual’s fitness value. The size of the fit-

ness value is also called the individual’s fitness value. Generally, the fitness value is nonnegative, and the size of the fitness value reflects the individual’s fitness. The greater the fitness value, the greater the individual’s fitness. Based on the principle of survival of the fittest in the iterative process of populations, people prefer to choose individuals with greater fitness to retain, so the final design of the fitness function also shows that the fitness value is proportional to individual superiority. However, for the design of the objective function, because the objectives to be optimized are diverse, and the degree to be achieved is also different, the positive and negative results are not required. For example, in order to obtain a smaller value, the objective function is to be smaller, and then, the fitness value of the individual will be larger.

In this paper, the chaotic cat swarm algorithm is used to optimize the multiobjective function, and the function selects execution time and load balance as the optimization objective. When the population is iteratively selected, fitness

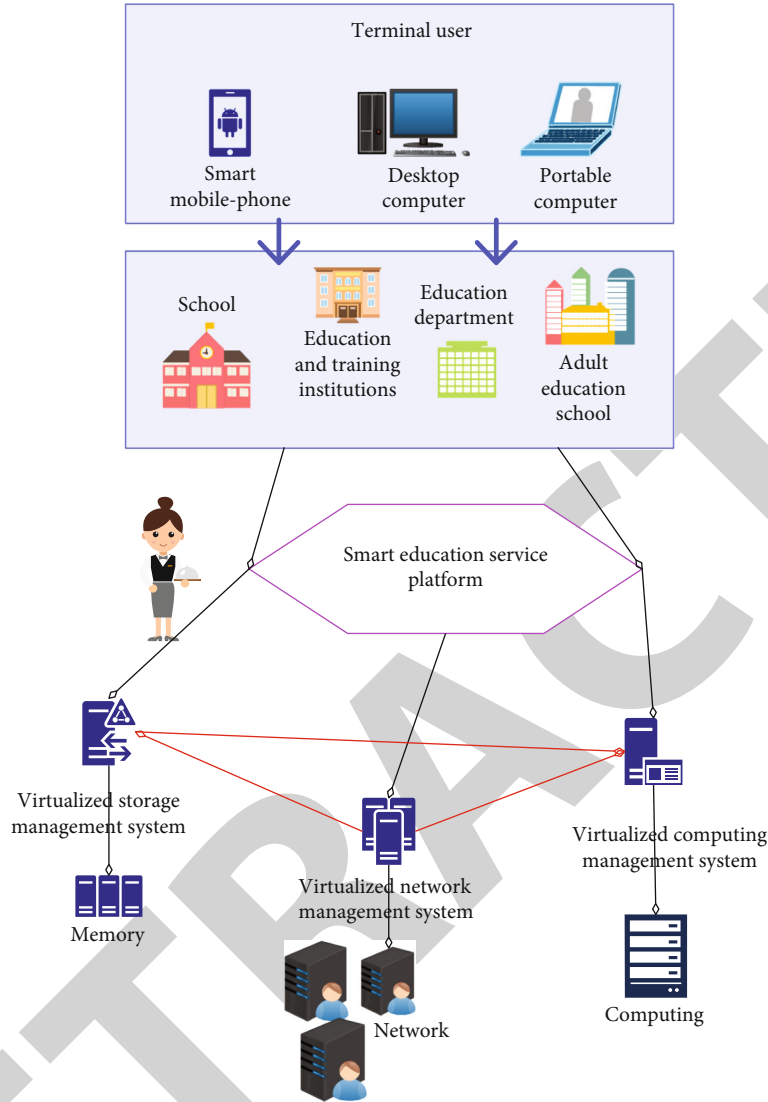


FIGURE 4: Smart education cloud service computing model diagram.

is used as the selection condition. Individuals with greater fitness are better, and the fitness function is set according to user needs. According to the fitness principle, it is set as the position of the i -th cat after t iterations, and then, the fitness function is designed to calculate its value. The fitness function is shown as follows:

$$F_1(x'_1) = \frac{1}{\log_2 \text{Excetime}(x'_1)}, \quad (10)$$

$$F_2(x'_1) = \frac{1}{\log_2 \text{VMload}(x'_1)}. \quad (11)$$

Among them, formula (10) is the fitness function of the execution time of the individuals in the population, and formula (11) is the fitness function of the load calculation of the individuals in the population.

In order to integrate the two functions with large differences, the value of each function takes the logarithm and then takes the reciprocal and finally adds up to obtain the objective function as shown in (12).

In the iterative process of the chaotic cat group, the cat group is divided into two working modes according to the grouping rate MR. The tracking mode is analyzed here. In the chaotic cat group tracking mode, the change of the position information of the cat group is relatively simpler than the change in the search mode, but its search range becomes larger as a result, so this mode corresponds to the global optimization process of the algorithm. Due to the large search range, when updating the cat group information, it is necessary to consider whether the updated cat group has run out of the prescribed range of activities. Once the range is exceeded, the information of the exceeding part needs to be changed, and the way to change is to map the position of the exceeding part. The specific steps of the tracking mode are as follows:

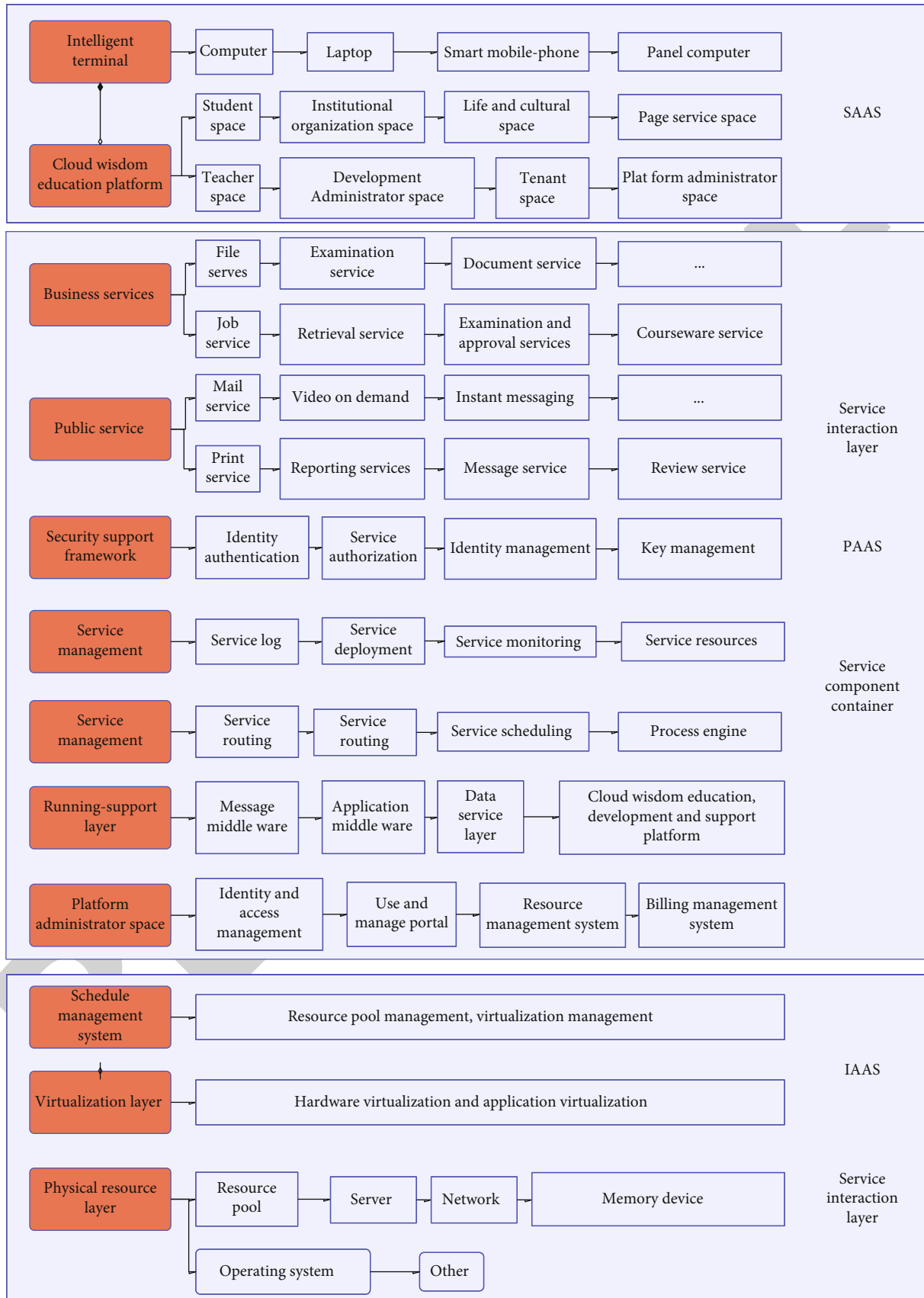


FIGURE 5: Smart education cloud service system structure diagram.

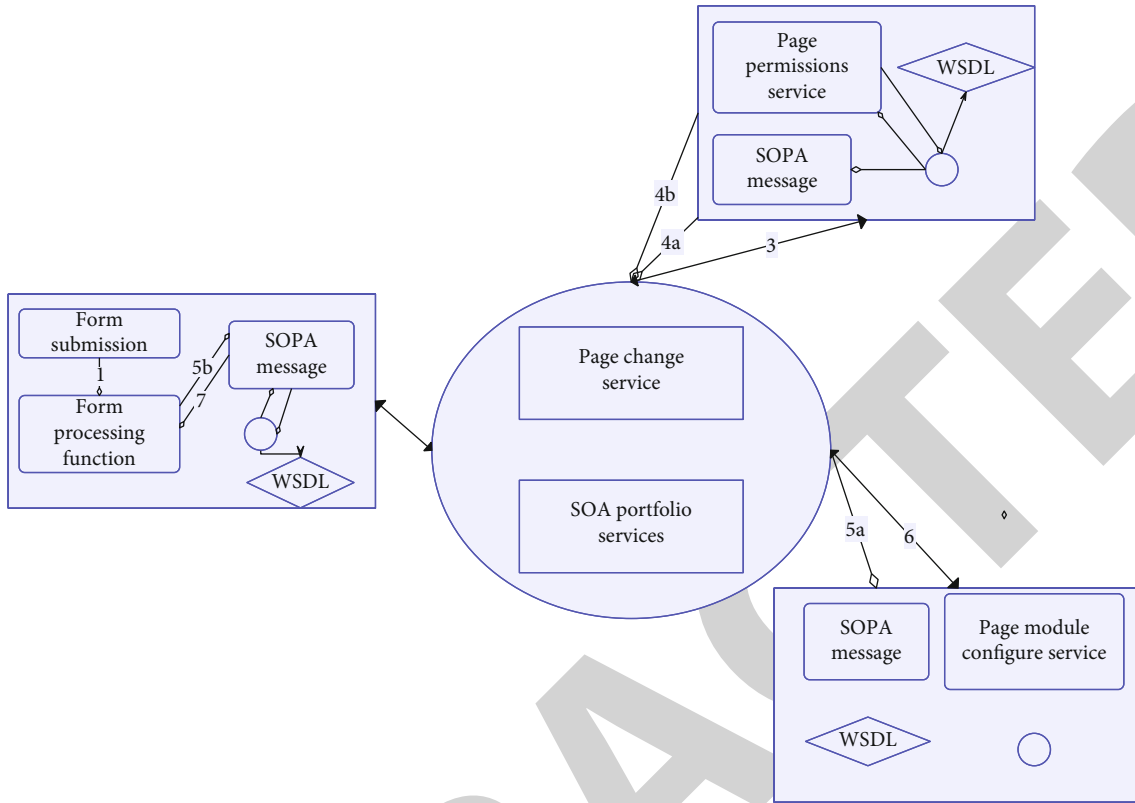


FIGURE 6: Work flow chart of page service.

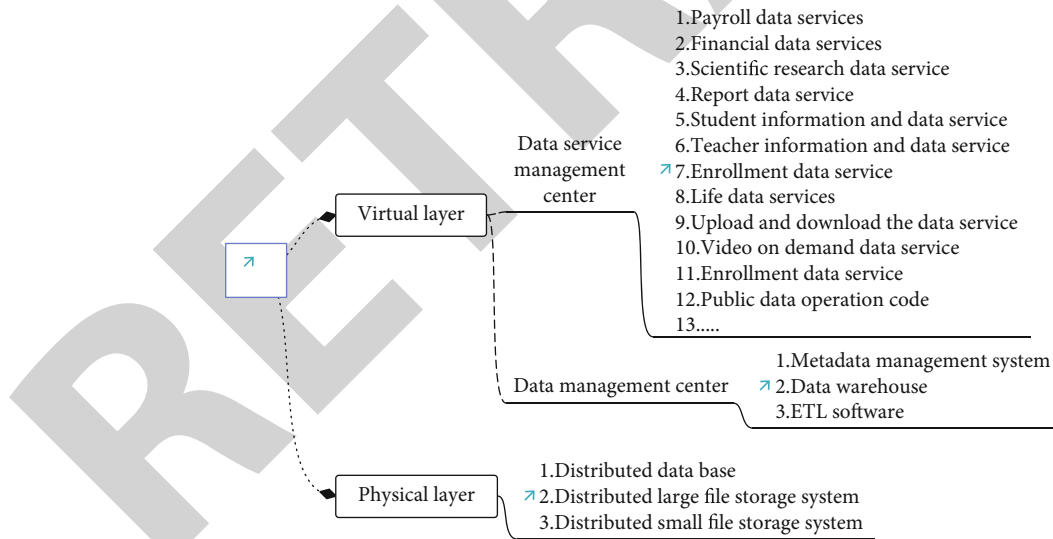


FIGURE 7: Data resource structure diagram.

(1) The algorithm uses formulas (12) and (13) to update the speed variable of each cat and compare it with the set speed boundary value. If it exceeds the range, it will be set to its neighboring boundary value

$$V_{k,d,t+1} = V_{k,d,t} + r1 \times c1 \times (X_{best,d,t} + X_{k,d,t}), \quad d = 1, 2, \dots, M, \quad (12)$$

$$V_{k,d,k+1} = \begin{cases} V_{max,d,t}, & V_{k,d,t} > V_{max,d,t}, \\ V_{min,d,t}, & V_{k,d,t} < V_{min,d,t}. \end{cases} \quad (13)$$

(2) The algorithm uses formulas (14) and (15) to update the position variable of each cat and compare it with

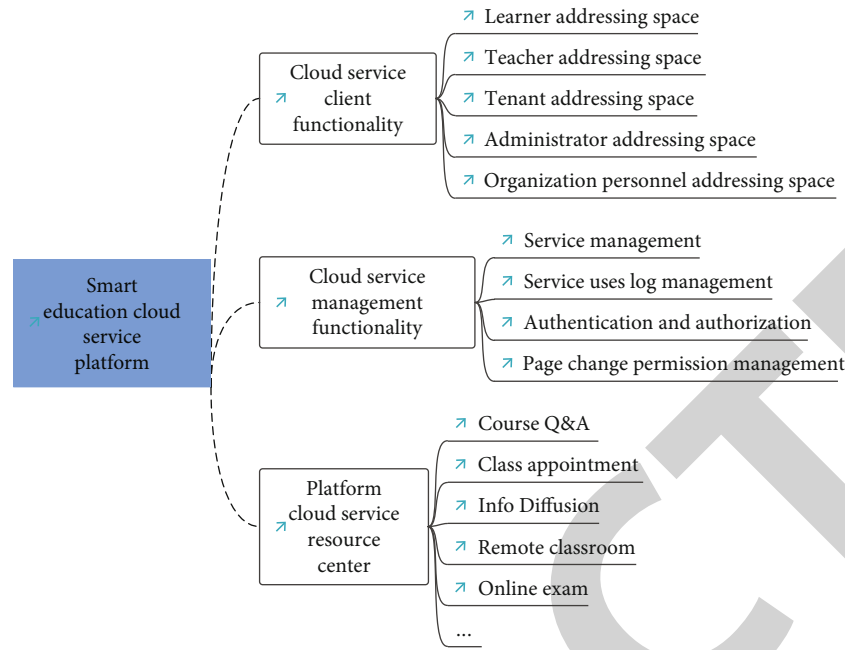


FIGURE 8: Data resource structure diagram.

SOA construction layer	Service management middleware				Virtualization resources	Physical resource
Service interface	User management	Task management	Resource management	Security administration	Calculation resource pool	Server cluster
Service registration	Account management	Image deployment and management	Load balancing	Identity authentication	Storage resource pool	Memory device
Service search	User environment configuration	Task scheduling	Fault detect	Access authorization	Network resource pool	Network device
Service access	User interaction management	Task execution	Recovery from failure	Comprehensive protection	Data resource pool	Data base
Service workflow	Use billing	Life management	Monitor statistics	Audit evaluation	Software resource pool	Utility software

FIGURE 9: SOA-based cloud service architecture.

the set position boundary value. If it exceeds the range, it will be set to its neighboring boundary value

$$X_{k,d,t+1} = X_{k,d,t-1} + V_{k,d,t} \quad (14)$$

$$X_{k,d,k+1} = \begin{cases} X_{\max,d,t}, & X_{k,d,t} > X_{\max,d,t} \\ X_{\min,d,t}, & X_{k,d,t} < X_{\min,d,t} \end{cases} \quad (15)$$

Among them, $x_{\text{best},d,i}$ is the optimal value of the fitness function of the k -th cat, $r1$ is a constant, and $c1$ is a variable, usually in the range of $[0,1]$.

The chaos-improved cat swarm algorithm is mainly used to improve global optimization. First, we used seven different chaotic maps to improve the model steps of the algorithm. Through research and analysis, we concluded that

logistic and sinograms are considered to be the best. Finally, we compare the algorithm before and after the improvement through experiments to prove the superiority of the improved algorithm. Therefore, the mapping used in this paper is a logistic chaotic mapping.

The addition of the logistic chaotic map enables the entire population to achieve a balance between global search ability and local search ability and has a good advantage in resource allocation task scheduling. The logistic chaotic map is used in the chaotic cat swarm algorithm, and its expression is shown as follows:

$$x(n+1) = u \times x(n) \times (1 - x(n)). \quad (16)$$

Among them, n is the number of iterations, u is the chaos parameter, $u \in [0, 4]$, $x \in [0, 1]$. In the classic logistic

TABLE 1: Evaluation of cloud task scheduling effect.

No.	Task scheduling
1	84.7
2	92.1
3	86.4
4	90.0
5	92.6
6	88.6
7	93.0
8	89.7
9	91.1
10	89.7
11	94.0
12	93.7
13	93.3
14	90.7
15	89.9
16	90.0
17	86.4
18	86.5
19	89.5
20	91.5
21	86.6
22	85.9
23	92.8
24	86.7
25	88.3
26	85.9
27	91.1
28	85.8
29	92.2
30	92.6
31	92.9
32	86.6
33	92.2
34	93.6

mapping, u is a variable. It can be seen from equation (16) that u affects the state of the logistic mapping. The value of u is also affected by xn . When u increases, the value range of xn also increases, and the distribution is more uniform. When the value of u increases to 4, the logistic mapping also tends to be surjective, and the xn sequence distribution is the most uniform at this time. This is further confirmed by calculations. After calculation, it is found that when $u = 4$, $x \in [0.1]$, and $x \notin \{0.25, 0.50, 0.75\}$, the chaotic state of the system is safe. The conventional variable is $Cx(n)$. After that, we use formulas (17) and (18) to map and transform chaotic variables. The formula is as follows:

$$x(n) = \frac{[Cx(n) - a]}{(b - a)}, \quad n = 1, 2, \dots, N, \quad (17)$$

TABLE 2: Evaluation of the processing effect of cloud computing education resources.

No.	Resource handling
1	92.6
2	83.6
3	83.9
4	92.6
5	82.1
6	82.0
7	85.8
8	86.1
9	86.0
10	91.5
11	84.1
12	92.7
13	91.6
14	94.6
15	88.5
16	91.5
17	94.6
18	81.4
19	87.3
20	93.2
21	93.0
22	91.3
23	84.4
24	85.4
25	93.0
26	85.5
27	94.2
28	85.5
29	93.1
30	89.7
31	93.5
32	93.7
33	82.7
34	92.0

$$Cx(n) = a + x(n) \times (b - a), \quad n = 1, 2, \dots, N. \quad (18)$$

Among them, a and b are the range of activity of the cat group, and $Cx(n)$ is the position of the cat group.

4. Construction of Smart Education System in Colleges and Universities Based on Cloud Computing Task Scheduling Algorithm

A low-cost, multitenant-oriented, and scalable digital education cloud service architecture needs to be established. The layers of the architecture are interrelated and independent of each other, and the upper layer has implementation dependence characteristics on the lower layer, but the result

TABLE 3: Evaluation of system education effect.

No.	Teaching effect
1	90.7
2	86.6
3	85.3
4	84.2
5	80.2
6	90.4
7	82.5
8	87.1
9	81.8
10	81.3
11	82.8
12	90.3
13	84.0
14	82.4
15	91.5
16	80.1
17	83.7
18	89.8
19	84.7
20	90.2
21	81.6
22	82.0
23	83.8
24	87.5
25	84.7
26	90.7
27	82.0
28	84.4
29	83.3
30	85.0
31	85.7
32	83.6
33	89.6
34	83.7

of the lower layer needs to be delivered to the user by the upper layer, as shown in Figure 4.

From the division of the cloud computing service level and scope, the architecture of the high-performance smart education cloud service platform is divided into three layers. It consists of an education cloud infrastructure layer, an education cloud service platform layer, and an education software and system application layer as shown in Figure 5.

Education cloud service users find the option to change the service in the page service. The page change service here is the change of the public service on the specific page. For example, ordinary students want to add a comment module under the remote course. The specific page service workflow is shown in Figure 6.

The data center of the smart education cloud service platform solves the problems of low utilization rate of

high-quality education resources and low data processing efficiency by building data services. By building a cloud storage environment, the problem of scattered, unshared, and low storage capacity of educational resources is solved. The data center includes a physical layer and a virtual layer, and its data resource structure diagram is shown in Figure 7.

Figure 8 shows the overall service level structure of the smart education cloud service platform.

From the perspective of system layering, the SOA-based cloud service architecture can be divided into the physical resource layer, virtualized resource pool layer, service middleware layer, and SOA building layer. Its architecture model is shown in Figure 9.

This paper calculates the performance of the smart education system of colleges and universities based on the cloud-side collaborative computing task scheduling algorithm proposed in this paper and separately counts cloud task scheduling, education resource processing, and education system effect evaluation. The results are shown in Tables 1–3.

From the artistic research, we can see that the smart education system of colleges and universities based on the cloud-side collaborative computing task scheduling algorithm constructed in this paper is very good in all aspects and has a greater advantage in processing educational resources.

5. Conclusion

Using the distributed storage technology of cloud storage, educational resources are divided into multiple blocks and stored on nodes in the cloud, and cloud storage technology is used to improve backup management and capacity management, which greatly guarantees the security of educational data. Therefore, according to the many bottlenecks faced by the current education system, cloud storage technology is used to establish a smart education cloud storage system, and cloud computing technology is used to centralize and virtualize the management of various types of hardware infrastructure equipment for education. Moreover, through the efficient storage and high sharing of various educational resources and data, it breaks the isolated educational system construction model; guarantees the security, flexibility, stability, and reliability of educational data resources; and avoids multilevel repetitive construction. This paper combines the cloud-side collaborative computing task scheduling algorithm to construct a smart education system in colleges and universities, analyzes the process of smart education in colleges and universities, and further enhances the effect of smart education on this basis. The smart education system of colleges and universities based on cloud-side collaborative computing task scheduling algorithm constructed in this paper is very good in all aspects and has a greater advantage in processing educational resources.

Data Availability

The labeled dataset used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares no competing interests.

Acknowledgments

This study is sponsored by the Xiamen University of Technology.

References

- [1] B. S. Abdelshaheed, "Using flipped learning model in teaching English language among female English majors in Majmaah University," *English Language Teaching*, vol. 10, no. 11, pp. 96–110, 2017.
- [2] S. F. Alfalah, J. F. Falah, T. Alfalah, M. Elfalah, N. Muhaidat, and O. Falah, "A comparative study between a virtual reality heart anatomy system and traditional medical teaching modalities," *Virtual Reality*, vol. 23, no. 3, pp. 229–234, 2019.
- [3] V. Andrunyk, T. Shestakevych, and V. Pasichnyk, "The technology of augmented and virtual reality in teaching children with ASD," *Econtechmod: Scientific Journal*, vol. 7, no. 4, pp. 59–64, 2018.
- [4] T. A. Ashraf, "Teaching English as a foreign language in Saudi Arabia: struggles and strategies," *International Journal of English Language Education*, vol. 6, no. 1, pp. 133–154, 2017.
- [5] V. L. Dayarathna, S. Karam, R. Jaradat et al., "Assessment of the efficacy and effectiveness of virtual reality teaching module: a gender-based comparison," *International Journal of Engineering Education*, vol. 36, no. 6, pp. 1938–1955, 2020.
- [6] A. Gupta, "Principles and practices of teaching English language learners," *International Education Studies*, vol. 12, no. 7, pp. 49–57, 2019.
- [7] M. S. Hadi, "The use of song in teaching English for junior high school student," *English Language in Focus (ELIF)*, vol. 1, no. 2, pp. 107–112, 2019.
- [8] O. Hernandez-Pozas and H. Carreon-Flores, "Teaching international business using virtual reality," *Journal of Teaching in International Business*, vol. 30, no. 2, pp. 196–212, 2019.
- [9] K. Kim, C. Jin, and S. Lee, "The development of interactive application prototype for adaptive instruction," *Korean Association for Educational Information and Media*, vol. 23, no. 1, pp. 139–168, 2017.
- [10] F. Kong, "Application of artificial intelligence in modern art teaching," *International Journal of Emerging Technologies in Learning (IJET)*, vol. 15, no. 13, pp. 238–251, 2020.
- [11] A. Mahboob, "Beyond global Englishes: teaching English as a dynamic language," *RELC Journal*, vol. 49, no. 1, pp. 36–57, 2018.
- [12] R. Mayne and H. Green, "Virtual reality for teaching and learning in crime scene investigation," *Science & Justice*, vol. 60, no. 5, pp. 466–472, 2020.
- [13] K. E. McCool, S. A. Bissett, T. L. Hill, L. A. Degernes, and E. C. Hawkins, "Evaluation of a human virtual-reality endoscopy trainer for teaching early endoscopy skills to veterinarians," *Journal of Veterinary Medical Education*, vol. 47, no. 1, pp. 106–116, 2020.
- [14] M. Pantic, R. Zwitterloot, and R. J. Grootjans, "Teaching introductory artificial intelligence using a simple agent framework," *IEEE Transactions on Education*, vol. 48, no. 3, pp. 382–390, 2005.
- [15] M. Reymus, A. Liebermann, and C. Diegritz, "Virtual reality: an effective tool for teaching root canal anatomy to undergraduate dental students—a preliminary study," *International Endodontic Journal*, vol. 53, no. 11, pp. 1581–1587, 2020.
- [16] H. Sundari, "Classroom interaction in teaching English as foreign language at lower secondary schools in Indonesia," *Advances in language and Literary Studies*, vol. 8, no. 6, pp. 147–154, 2017.
- [17] M. Taubert, L. Webber, T. Hamilton, M. Carr, and M. Harvey, "Virtual reality videos used in undergraduate palliative and oncology medical teaching: results of a pilot study," *BMJ Supportive & Palliative Care*, vol. 9, no. 3, pp. 281–285, 2019.
- [18] C. Yang, S. Huan, and Y. Yang, "A practical teaching mode for colleges supported by artificial intelligence," *International Journal of Emerging Technologies in Learning (IJET)*, vol. 15, no. 17, pp. 195–206, 2020.
- [19] S. C. H. Yang, W. K. Vong, R. B. Sojitra, T. Folke, and P. Shafto, "Mitigating belief projection in explainable artificial intelligence via Bayesian teaching," *Scientific Reports*, vol. 11, no. 1, pp. 1–17, 2021.
- [20] O. Zawacki-Richter, V. I. Marín, M. Bond, and F. Gouverneur, "Systematic review of research on artificial intelligence applications in higher education—where are the educators?," *International Journal of Educational Technology in Higher Education*, vol. 16, no. 1, pp. 1–27, 2019.
- [21] S. Zou, "Designing and practice of a college English teaching platform based on artificial intelligence," *Journal of Computational and Theoretical Nanoscience*, vol. 14, no. 1, pp. 104–108, 2017.