

Retraction

Retracted: Design of Game Data Acquisition System Based on Artificial Intelligence Algorithm

Security and Communication Networks

Received 17 October 2023; Accepted 17 October 2023; Published 18 October 2023

Copyright © 2023 Security and Communication Networks. 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This article has been retracted by Hindawi following an investigation undertaken by the publisher [1].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) Peer-review manipulation

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] X. Li and H. Wang, "Design of Game Data Acquisition System Based on Artificial Intelligence Algorithm," *Security and Communication Networks*, vol. 2022, Article ID 1679092, 11 pages, 2022.

Research Article

Design of Game Data Acquisition System Based on Artificial Intelligence Algorithm

Xiangrui Li and Huayi Wang 

Department of Culture Technology, Jeonju University, Jeonju 55069, Jeollabuk-do, Republic of Korea

Correspondence should be addressed to Huayi Wang; wanghuayi2022@163.com

Received 25 April 2022; Revised 16 June 2022; Accepted 1 July 2022; Published 31 July 2022

Academic Editor: Panagiotis D. Diamantoulakis

Copyright © 2022 Xiangrui Li and Huayi Wang. 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.

Game data collection system is a tool used to collect the behavior data of users about the game. It can be used for data analysis of user behavior so that game manufacturers can keep abreast of market dynamics and popular trends, and they also can have a deeper understanding of the behavioral habits and psychology of player user groups. The defects of the current data acquisition system include that the data are not encrypted. The network transmission efficiency is relatively low. The acquisition speed is slow, and the settings cannot be dynamically changed. This paper proposes to study how to enhance the acquisition ability and improve the analysis efficiency in the design of data acquisition system for solving these problems. Therefore, on the basis of artificial intelligence algorithm, this paper designs a game data collection system by using artificial neural network algorithm, support vector algorithm, and cluster analysis algorithm, which solves the basic problem of slow data collection in current data collection and plays a role in improving the efficiency of network transmission. The experimental results in this paper show that when the number of data is more than 300, the time-consuming time reaches more than 68 ms. When the number of written data is more than 300, it takes more than 181 ms. When the number of deleted data is more than 300, it takes more than 236 ms. From the above data, it shows that the designed game data collection system is rapid and efficient.

1. Introduction

With the rapid development of mobile Internet technology and the popularity of smartphones in recent years, China's mobile game industry has also seen rapid development. The current data collection method is to implant relevant collection tools in the game to obtain real-time user game behavior data, to understand market dynamics and popular trends, so that the behavioral habits and psychology of the player user group can be grasped more deeply.

China's game industry has also developed rapidly. The game market is gradually transitioning from the incremental market to the stock market. How to snatch existing game users to increase game revenue has become the biggest challenge for game manufacturers. Therefore, based on artificial intelligence algorithm, this paper applies it in the research of game data collection system, which can greatly improve the optimization and operation efficiency of the

game, expanding the analysis of market channels, so that precise marketing can be carried out.

The game data collection system designed by artificial intelligence algorithm in this paper has the following major innovations:

- (1) It can not only collect general data that all mobile games need to use, but also collect custom data customized for business needs according to the configuration files issued by developers. Developers only need to add this tool to project dependencies through Gradle, and call the tool's initialization code in application to complete the tool integration.
- (2) Using the non-buried point solution can quickly and automatically obtain a large amount of inspiring user operation information, which is of great value in game applications dominated by user interaction. Through analyzing the user's behavior data, the click

probability of the user on the corresponding interface of the application can be quickly analyzed, so that the research and development engineers can make more in-depth optimization for some function points that users pay much attention to.

- (3) It also improves tool stability. The number of crashes brought to the game due to insufficient tool stability should be minimized to avoid additional problems for developers. Tool usage should be as simple and understandable as possible. The interface for users to call should be as simple as possible to reduce the learning cost for developers to integrate this tool.

2. Related Work

Artificial intelligence is a branch of computer science that attempts to understand the essence of intelligence and produce a new type of intelligent machine that can respond in a manner similar to human intelligence. Research in this field includes robotics, language recognition, image recognition, natural language processing, and expert systems, etc. Artificial intelligence has been used in many fields. Wang et al. proposed a new hybrid approach that used an ensemble data fluctuation network (DFN) and multiple artificial intelligence (AI) algorithms, which was called the DFN-AI model [1]. Majetta et al. described a method of generating large amounts of data and using it to find the relationship between a room controller and a certain room. Simulation scenarios with different room location, usage, and controller models could be defined and developed with it [2]. Wang has built a linguistics artificial intelligence teaching model with improved machine learning algorithms. The efficiency of the teaching process was improved according to the teaching needs of linguistics. A pedagogical evaluation was conducted, and an MCTS-based root cause analysis algorithm was also optimized [3]. Grath et al. intend to investigate the clinical utility of applying deep learning denoising algorithms to standard wide-field optical coherence tomography angiography (OCT-A) images [4]. Zakaria et al. discussed the optimization of hyperparameters for both models. Then sensitivity analysis and uncertainty analysis were performed. The model's ability to predict river levels with different lead times (1, 3, 6, 9, 12, and 24 hours) was then investigated [5]. Sebastianelli et al. aimed to describe a new tool to support agencies in implementing targeted responses, which was based on quantitative and multiscale elements to combat and prevent emergencies, such as the current COVID-19 pandemic [6]. Liu et al. presented a crowd-sourced inference method with variational tempering that obtains the ground truth. Both worker reliability and task difficulty level were taken into account, and local optima was ensured [7]. Mirotta et al. focused on interpreting fuel rod behavior during power pulses using an online fuel motion monitoring system called a hodoscope [8]. The study of Nowakowski et al. included a simulation of e-waste collection requests in Tokyo, Philadelphia, and Warsaw, which was the algorithm used to compare various city, street, and building layouts. The results showed that the best of the four algorithms was simulated annealing to facilitate mobile on-demand

collection of e-waste, and the worst was tabu search [9]. Barcelos et al. introduced a new current-based method to identify bearing damage, applying artificial intelligence algorithms. Experiments and field tests showed promising results, validating the method for bearing damage diagnosis [10]. The work of Prabakaran introduced a new current-based approach that applied artificial intelligence algorithms to identify bearing damage. Experiments and field tests presented promising results, validating this method for the diagnosis of bearing damage [11]. Mihai et al. in the paper aimed to develop three machine learning algorithms. It could significantly improve the drug discovery process, which was possible to combine computer scientists and drug development experts [12]. The landslide susceptibility maps produced by Chen et al.'s research could be used to manage landslide hazards and risks in counties, township, and other similar areas [13]. However, the above-mentioned field research on artificial intelligence algorithms only stays in the theoretical part, and the practicality is not strong.

3. Artificial Intelligence Algorithms

3.1. Artificial Neural Networks. Overview: the term "Artificial Neural Network" (ANN) is derived from biomedical neural networks. A neuron can establish connections with multiple surrounding neurons through dendrites and axons to receive, process, and transmit information [14]. The human body's complex nervous system is built on hundreds of millions of neurons. Therefore, building a neural network that mimics the biological nervous system can help understand and capture the information implicit in the data. A single neuron model is shown in Figure 1.

The forward transmission of neuron information can be represented by

$$y = f(w_i x_i + b). \quad (1)$$

In formula (2), X_i represents the input signal; w_i represents the weight; b represents the bias part; y represents the signal output. The commonly used activation function is the sigmoid function as

$$f(x) = \frac{1}{1 + \exp(x)}. \quad (2)$$

In traffic forecasting, traffic data are input into the forecasting model as a time series. First, the training of the model needs to be completed. Then the trained model is used to make predictions. Usually, the optimization objective can be set as the error:

$$E(x) = \frac{1}{2} \sum_{i=1}^N y_i(x_i) - y_i(x_i)^2. \quad (3)$$

In formula (3), $y_i(x_i)$ represents the output data; $y'_i(x_i)$ represents the target data; and N represents the number of data. Neural networks have often been used in combination with other optimization algorithms in traffic prediction in previous studies to obtain better prediction models [15].

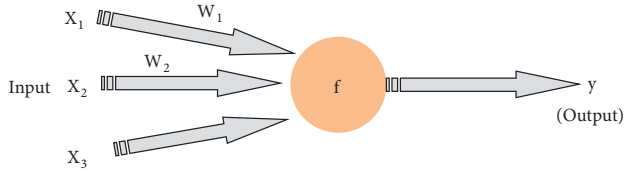


FIGURE 1: Single neuron structure.

3.2. Support Vector Machine Algorithm. Support vector machine (SVM) is a class of generalized linear classifiers that perform binary classification on data by supervised learning. It is a classifier with sparsity and robustness. Originally developed for solving linearly separable problems, support vector machine algorithms were developed on the basis of statistical theory and then gradually extended to the non-linear case.

The mapping from the input feature space to the k -dimensional space in nonlinear classification is as

$$x \in R^l \longrightarrow y \in R^k. \quad (4)$$

The classification can be done using

$$g(x) = w^T x + w_0 = \sum_{i=1}^{N_s} \lambda_i y_i x_i x + w_0. \quad (5)$$

In formula (5), N_s represents the number of support vectors. Typical choices of kernel functions are as

$$K(x, z) = (x^T z + 1)^q, q > 0, \quad (6)$$

$$K(x, z) = \exp\left(-\frac{x - z^2}{\sigma^2}\right), \quad (7)$$

$$K(x, z) = \tanh(\beta x^T z + \gamma). \quad (8)$$

After a suitable kernel function is selected, the mapping to a higher dimensional space is implicitly defined. The Wolfe double optimization task becomes

$$\max \left(\sum_i \lambda_i - \frac{1}{2} \sum_{i,j} \lambda \lambda y y K(x_i, x_j) \right) \sum_i \lambda_i y_i = 0. \quad (9)$$

The resulting linear classification is as follows:

$$\text{If } g(x) = \sum_{i=1}^{N_s} \lambda y K(x_i, x) + w_0 (< 0), x \in w_1 (w_2). \quad (10)$$

Figure 2 shows a nonlinear SVM architecture, where the number of nodes is determined by the number of support vectors.

The SVM algorithm has great advantages in dealing with problems with smaller sample sizes. In case of larger problem design or when dealing with multiclassification problems, it is difficult to implement due to the complex solving process and the large amount of computation [16].

3.3. Cluster Analysis Algorithms. Meaning: agglomeration refers to the division of a collection of physical or abstract

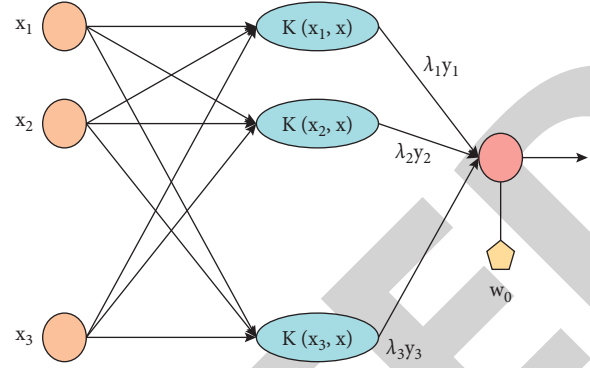


FIGURE 2: SVM architecture of kernel functions.

objects into multiple classes or groups, so that all objects belonging to the same class have a high degree of similarity, while objects in different classes are quite different [17]. The dissimilarity is calculated according to the attribute value of the described object, and the distance is the most commonly used measure. The class to be divided into clustering is unknown, which is different from classification, that is, clustering is an unsupervised observational learning. It is a data reduction technique that groups together variables or cases with similar data characteristics. It can be used in the development and research of data storage technology in the design and research of game data acquisition systems, and can construct Mysql database by using cluster analysis algorithm.

Suppose that $x = (x_1, \dots, x_p)$, $y = (y_1, \dots, y_p)$ are two points in space whose Minkowski distance is

$$d(x, y) = \left(\sum_i^p |x_i - y_i|^m \right)^{\frac{1}{m}}, \quad m > 0. \quad (11)$$

When $m = 1, 2, \infty$, three commonly used distances are obtained:

(1) When $m = 1$, it is the absolute value distance as

$$d(x, y) = \sum_i^p |x_i - y_i|. \quad (12)$$

(2) When $m = 2$, it is the Euclidean distance as

$$d(x, y) = \sqrt{\sum_i^p (x_i - y_i)^2}. \quad (13)$$

(3) When $m = \infty$, it is the Chebyshev distance as

$$d(x, y) = \max_i |x_i - y_i|. \quad (14)$$

Minkowski distance satisfies the following three properties as

$$d(x, y) = d(y, x), \quad (15)$$

$$d(x, y) \leq d(x, z) + d(z, y), \quad (16)$$

$$\text{If } x \neq y, d(x, y) > 0, d(x, x) = 0. \quad (17)$$

It should be noted that Minkowski distance is only limited to measure the similarity between numerical individuals and cannot be used to measure the similarity of attribute individuals. Observing the expression of Min's distance, it is easy to know that Min's distance is easily affected by larger data and ignores data with smaller values [18]. Min's distance often has a large error when there is data with a large value in a certain sample. In addition, Min's distance does not eliminate the effect of dimension. Since there is a linear correlation between the data, it will affect the distance between them. To eliminate this effect, the researchers propose the Mahalanobis distance based on the covariance between the data. Two vectors are randomly selected from the sample, and their Mahalanobis distance is as

$$d(x, y) = (x - y) \sum_{-1}^T (x - y)^T. \quad (18)$$

- (1) It is often necessary to discriminate the effect of a certain kind of clustering in practice. The standard function of clustering used is as follows.

Between-class dispersion sum of squares functions:

$$J = \sum_j^k (C_j - C)^T (C_j - C). \quad (19)$$

In formula (19), k represents the number of clusters; c_j represents the class center of a cluster; c represents the sample center.

- (2) Within-class dispersion sum of squares functions:

$$J = \sum_{j=1}^k \sum_{i=1}^{N_j} x_i^{(j)} - c_j^2. \quad (20)$$

In formula (20), $x_i^{(j)}$ represents an individual (sample data point) in cluster j ; c_j is the cluster center; k is the number of clusters set according to prior knowledge; N_j represents the sample capacity of cluster j ; $x_i^{(j)} - c_j$ represents the distance from $x_i^{(j)}$ to the class center c_j .

Therefore, it can be seen from the above method that the artificial neural network algorithm can play a role in helping it establish the framework of the data acquisition system in the framework of the data acquisition system. The support vector machine method can help the establishment of the reporting module in the data acquisition system. The cluster analysis algorithm can help the design of data storage services, and the data acquisition system designed by the above algorithm is of great significance at present.

4. Construction and Design of the Data Acquisition System

4.1. Acquisition Process. The entire data collection tool collection process starts after the data collection tool SDK is initialized when the app starts. When the app is installed for the first time, it obtains and reports the information required for app channel statistics. When it is started for the first time

today, it will report yesterday's PV data and startup times and other total data from yesterday. Otherwise, it will report the application profile data that needs to be counted for this startup, such as the startup time and the duration of the application used after the last startup. The collection of real-time data such as player behavior occurs after the application is started and when an interaction event occurs during the player's operation. After the original touch event is generated, the original event information needs to be matched with the information of the view layer. Then, the target data are obtained by step-by-step reflection according to the data path, and after finding the original interaction event of the data, the data are reported according to the reporting policy. The data analysis module processes the data and generates data files, and the interaction module grabs the data files and uploads them to the server. The overall workflow of the data acquisition tool is shown in Figure 3.

4.2. Outline Design of Technical Architecture. From the perspective of technical implementation, the mobile game application data collection tools are mainly divided into two parts: One part is data collection, and the other part is data reporting. Among them, the core part is data collection. The data collection and data reporting are connected to the SQLite database through the Handler message mechanism. This section will introduce the outline design of the technical architecture of the mobile game application data collection tool in two parts: the outline design of the technical architecture of data collection and the outline design of the technical architecture of data reporting.

4.3. Outline Design of Data Acquisition Technology Architecture. The overall technical architecture adopts a top-to-bottom transmission design. The mobile game application receives touch events from the user, and the system calls the callback method of the related event when the event occurs. Since the custom AOP code has been inserted into the header of the callback method by means of byte code instrumentation at compile time, the AOP code acts as a proxy to perform data collection and processing operations for related events [19]. After the data collection is completed, the collected relevant data are distributed through the handler message delivery mechanism to distribute the messages of the user operation events, and the messages are cached in the message queue. The outline design of the data acquisition technology architecture is shown in Figure 4.

4.4. Outline Design of Data Reporting Technical Architecture. The data reporting part defines the data reporting policy, which determines whether the current data needs to be reported. The data sources include the current in-memory cached data from the Handler message distribution center and the historical data stored in the SQLite database. The schematic design of the data reporting technology architecture is shown in Figure 5.

As can be seen from the figure, the data reporting technology includes the current in-memory cached data

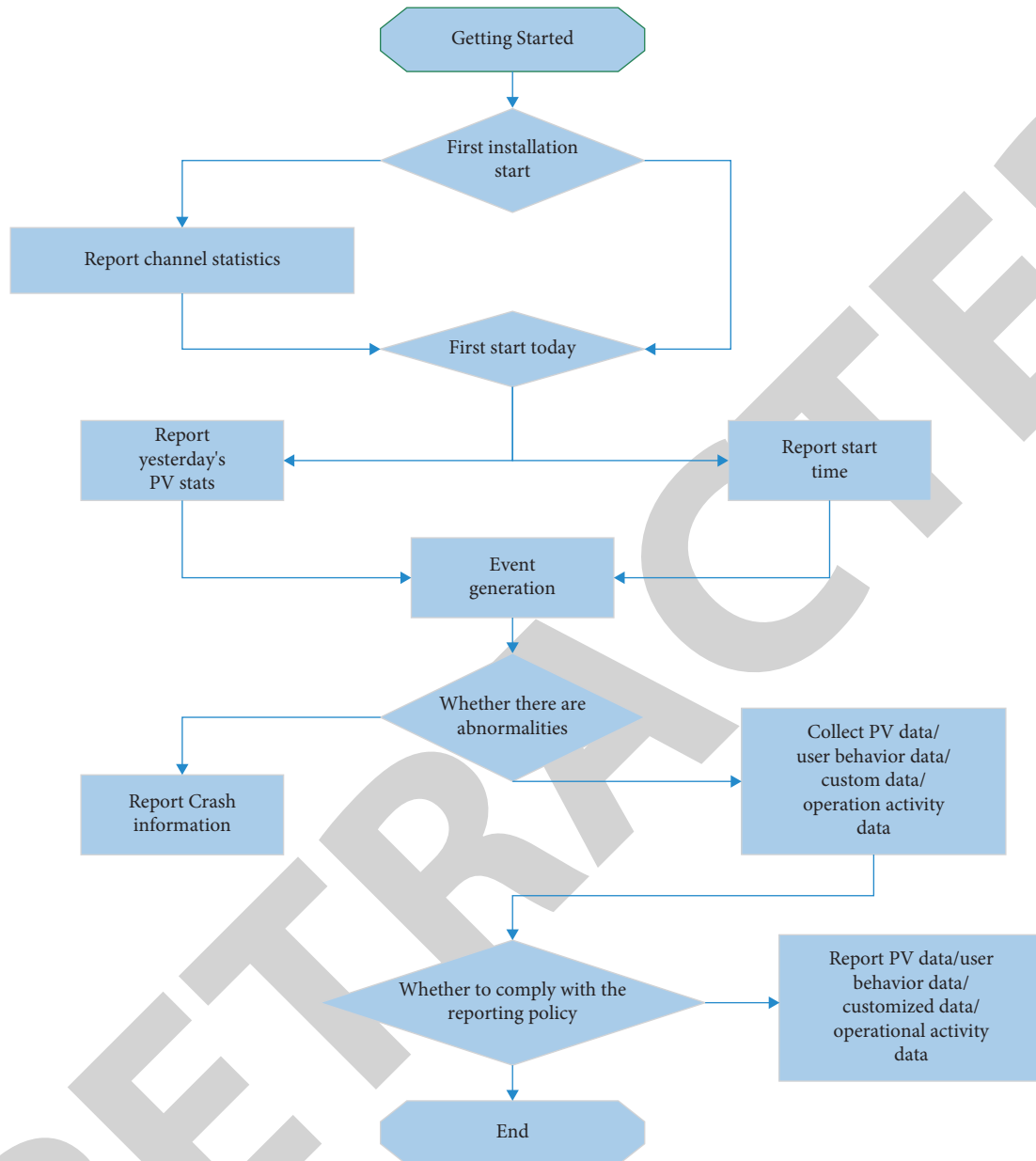


FIGURE 3: Workflow diagram.

from the Handler message distribution center and the historical data stored in the SQLite database, which are loaded into the data storage server through the Kafka cluster.

4.5. Outline Design of the General Data Acquisition Module. The general data collection module is responsible for collecting general data. This module will introduce the collection of general data from four parts: application overview, channel statistics, user equipment, and PV statistics.

4.5.1. Application Overview. The application profile mainly collects crash information, supplemented by application version, device identification information, usage time, and startup times. Crash information plays an essential role in

the development process. By collecting crash information, it can count the stack information of various exceptions that occur in user scenarios. Developers repair crashes in time to reduce the frequency of crashes, which can improve the stability of the application and improve the user experience. The model class design of the application profile is given in Table 1.

4.5.2. Channel Design. Channel statistics mainly count the channel from which the application currently used by the user is downloaded. The purpose of obtaining geographic location information is to better statistically analyze the geographic distribution of the user. To realize channel statistics, this paper designs a custom multichannel packaging tool. It writes the channel information in the generated

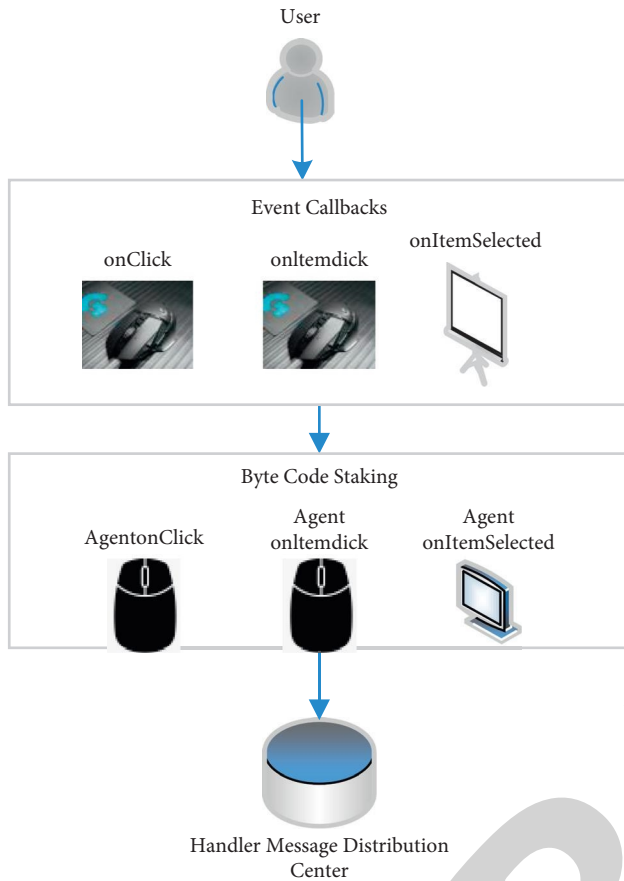


FIGURE 4: Data acquisition technology architecture diagram.

package through a custom multichannel packaging tool, so that the data collection SDK can obtain the channel information when the application is running. The model class design of channel statistics is given in Table 2.

4.5.3. User Equipment. User device statistics system version number, screen resolution, remaining memory, networking mode, IP address, and other device information, such as `ACTIVITY_SERVICE`, `CONNECTIVITY_SERVICE`, and `WIFI_SERVICE`, etc. By obtaining the object corresponding to the system service, the corresponding device information can be obtained. The model class design of the user equipment is given in Table 3.

4.5.4. PV Statistics. PV statistics are responsible for counting the number of clicks each person has on each page every day. The statistical information includes the name of the current page and the corresponding PV times. These data are of great significance for the statistics of users' preferences for APP usage. The model class definition of PV statistics is given in Table 4.

The statistical method of PV time mainly updates the PV times of the page in the corresponding data table by implanting the database update operation in the `onCreate` method of activity or fragment. It takes an event page as an example. If users open a certain event page frequently, it

means that the current activities in the online environment are very attractive to users, which has great guiding significance for future event organization and strategy formulation.

4.6. Data Storage Service. This paper uses Kafka as the message system for server-side data storage. Since Kafka is distributed, it can meet the requirements of high concurrency and high throughput brought by real-time data reporting by data collection tools. The processing flow of the data storage service data flow is shown in Figure 6.

As a message middleware, the Kafka cluster is not only responsible for processing the data reported from the client, storing the data in MySQL in turn, but also processing the data query from the data statistics analysis service and the PC front-end, and extracting the data from MySQL. As the amount of business data increases, the number of Kafka can be increased to horizontally expand Kafka and improve the throughput of the Kafka cluster. The specific storage of MySQL data is similar to the client's SQLite storage [20].

This section is the key chapter of this paper and is the realization part of the whole data acquisition tool. In this chapter, the design and implementation of general data collection module, custom data collection module, user behavior data collection module, byte code instrumentation module, data reporting module, and server-side data storage module of game application data collection tools are introduced in detail in combination with the artificial intelligence algorithm analysis and outline design in the Methods section. Through the detailed introduction in this chapter, all requirements of the game application data acquisition system have been completed.

5. Game Data Acquisition System Test and Results

This section tests and analyzes the mobile game data collection tool to verify whether the design and implementation of the tool meet the design requirements of the tool.

5.1. Test Environment. Based on the support vector machine algorithm in the artificial intelligence algorithm, combined with its characteristics in data analysis, the latest data on the distribution of Android platform versions released on the official website of Google Android developers show that Android 5.1, Android 6.0, Android 7.0, and Android 8.0 versions occupy 19.2%, 28.1%, 22.3%, and 21.7% of the market share, respectively, which occupies the top four in market share. Therefore, this article selects an Android mobile phone with the above four Android versions as the test mobile phone in this chapter. The selected mobile phone parameters are given in Table 5.

5.2. Test Plan. The testing part of this paper will start from two aspects, namely, functional testing and performance testing. The goal of functional testing is to verify whether each module of the data acquisition tool meets the basic data

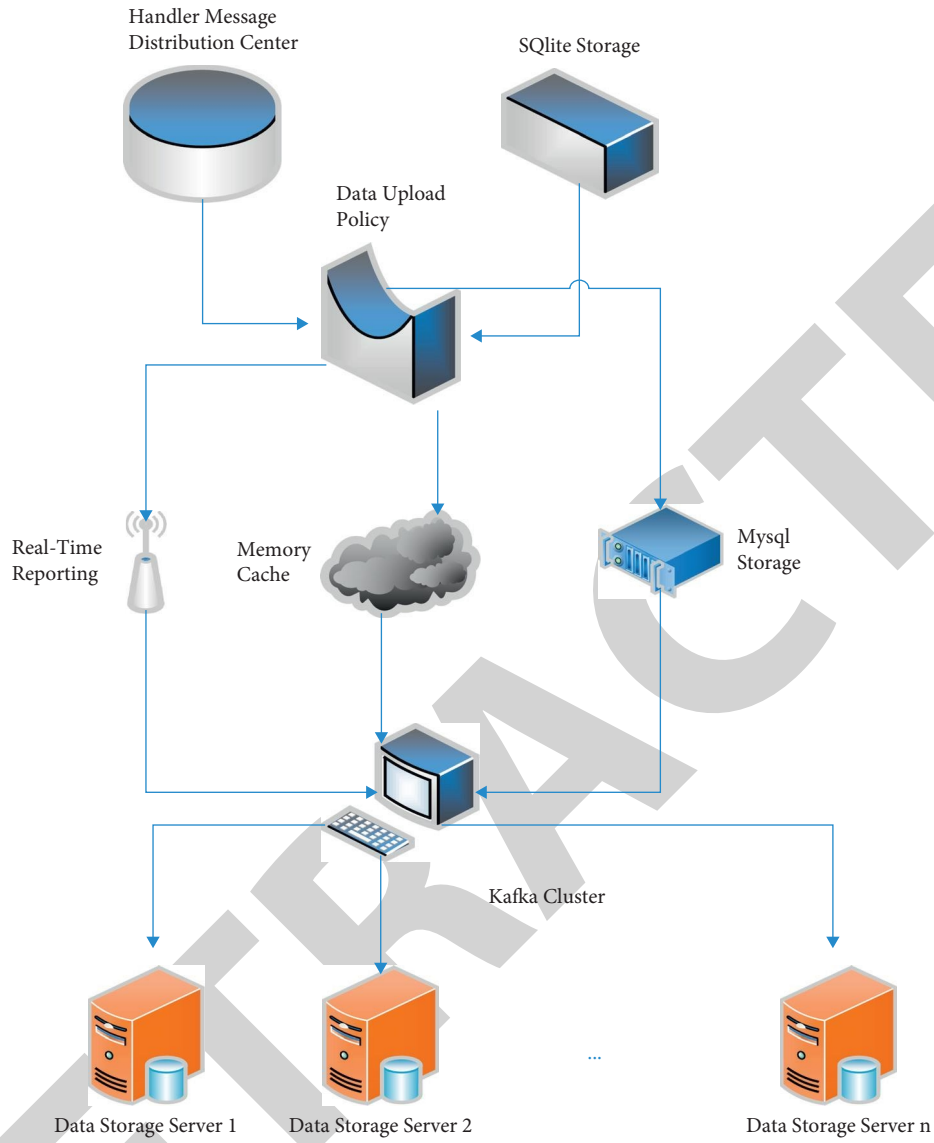


FIGURE 5: Overview of data reporting technology.

TABLE 1: Application overview model class.

Model type	Field name	Meaning
Application Data Block	appVersion	Application version
	crashInfo	Application crash information
	deviceTag	Device identification
	usedTime	Length of use
	launchCount	Number of starts

TABLE 2: Channel statistics model class.

Model type	Field name	Meaning
Channel Data Block	channelName	Channel name
	deviceName	Device name
	userPosition	User location

TABLE 3: User device model class.

Model type	Field name	Meaning
Device Data Block	deviceModel	Device model
	systemVersion	System version number
	Screen	Screen resolution
	iResolution	Remaining memory
	Memory remain	Networking method
	netMode	Network operator
	netOperator	IP address
	ipAdress	User location

TABLE 4: PV statistics model class.

Model type	Field name	Meaning
PVD Data Block	pageName	Page name
	deviceTag	User device ID
	pagePVCount	Page PV count.

TABLE 5: Test phone parameter information.

Device model	System version	Screen size (inch)	Screen resolution
OPPO R9	Android 5.1	5.5	1920 * 1080
Meizu MX6	Android 6.0	5.5	1920 * 1080
Huawei nova2 Plus	Android 7.0	5.5	1920 * 1080
Redmi K50 Pro	Android 8.0	6.7	1920 * 1080

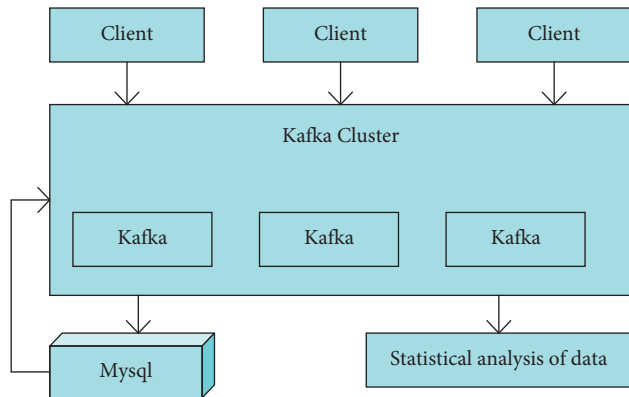


FIGURE 6: Processing flow of data storage service data flow.

acquisition functional requirements, including the general data acquisition module and the custom data acquisition module, to ensure the integrity of the tool function [21]. Since this article collects data from Android mobile games, this article takes the practice project chess and card game as an example and tests the functional requirements and performance requirements of the data collection tool according to the requirements put forward by the above evaluation in this article.

5.3. Functional Test of the General Data Acquisition Module.

The function test of the general data acquisition module will start from the four functions of application overview information, channel statistics information, user equipment information, and PV statistics information to test whether the data collected by each function is correct. The test details of the general data acquisition module are as follows:

5.3.1. Test Content. Collect all information including application profile information, channel statistics, user device information, and PV statistics, such as app version number, crash information, app download channel number, phone screen resolution, page clicks, etc.

5.3.2. Test Steps. Through the channel packaging tool, the information of three different channels, Test1, Test2, and Test3, is written to the generated Release APK file. Then it opens the game, collects the application overview, channel statistics, and user device information, and clicks the button to switch pages to collect PV statistics on different pages. Finally, by inserting the code that accesses the data out-of-bounds in the code, the crash information is counted when this code is executed.

5.3.3. Expected Results

- (1) During the running of the game, there is no game lag phenomenon
- (2) When the array out-of-bounds code is executed, the application flashes back, and the collected Crash information is also caused by the out-of-bounds array
- (3) The application profile data, channel statistics, user equipment data, and PV statistics are all the correct data collected

Test results are shown in Figure 7.

5.4. Function Test of the Custom Data Acquisition Module.

The function of the custom data acquisition module is mainly based on the needs of the game project itself, configure the target data file to collect, and collect the corresponding data. This section will take chess and card games as an example to test the collection of the following information: time when entering/exiting the game; the time when the purchase room card option in the mall interface is clicked; the specific room card amount data of this option; the push message of the operation activity is clicked; entering the game selection page, the specific game is clicked event. The test details of the custom data acquisition module are as follows:

5.4.1. Test Content. Collect the time of entering the game and exiting the game; the specific amount data when the recharge amount option is clicked; the news push of the operation activity is clicked; the event of the subgame is selected.

5.4.2. Test Steps. First, it configures a custom data collection configuration file on the server and delivers it to the client. Next, it opens the chess and card game, receives the configuration file, collects the time of entering the game, then enters the mall interface, and selects room cards of different numbers in turn. After collecting the amount information of the selected room card, it enters the game selection interface. When Doudizhu or Mahjong is selected, it determines which game is selected to capture the event. When there is a news push of operational activities, it collects the page address to which the clicked news push jumps. Finally, when the user exits the game, the time of exiting the game is collected.

5.4.3. Expected Results

- (1) During the running of the game, there is no game lag

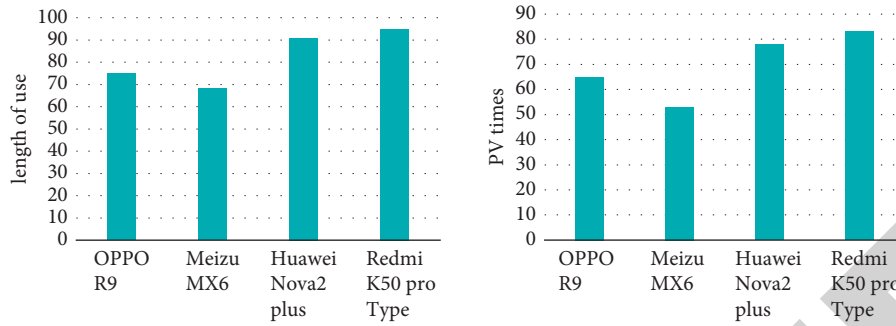


FIGURE 7: Test result graph.

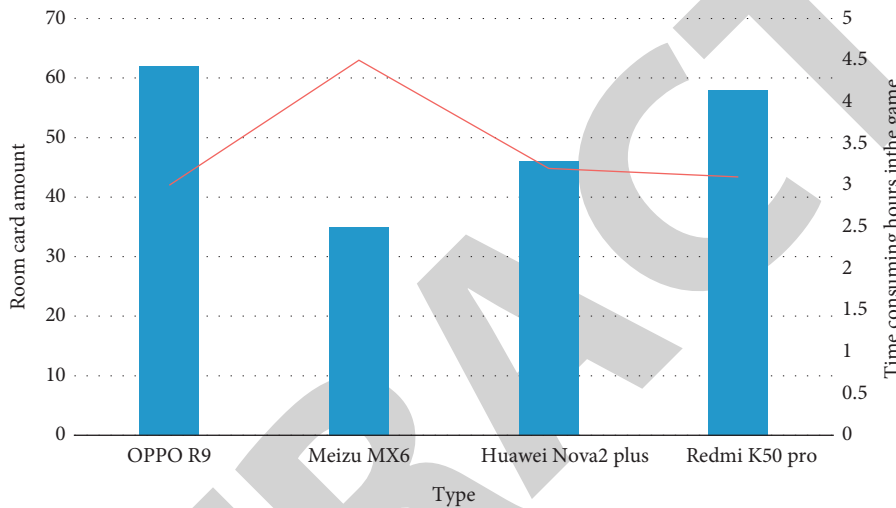


FIGURE 8: Functional test result graph.

- (2) The collected time, amount, url, and other data will be displayed in the console in the form of logs

5.4.4. *Test Results.* The result is shown in Figure 8.

5.5. *Data Collection Performance Test.* The data collection performance test mainly tests the time-consuming situation of collecting data, and the result of the data collection performance test is an important basis for judging whether the data collection tool meets the basic requirements of mobile game collection data [22]. This article will generate a large amount of user behavior data by frequently operating the game within a certain period of time. It judges whether the performance of the data collection tool meets the actual use requirements according to whether there is a freeze or crash. The data acquisition performance test results are shown in Figure 9.

From the table that when the number of continuously collected data is less than 1000, the time consumption is within the acceptable range within 3 seconds, and there is no stuck phenomenon or crash phenomenon.

5.6. *Data Read and Write Performance Test.* The data read and write performance test mainly tests the time-consuming situation of reading and writing data in the SQLite database,

to verify whether there is a serious performance problem when using the SQLite database to read and write data under the non-real-time reporting strategy. The test method is to modify the reporting policy to batch reporting and set the reporting threshold to 1000. It adds time-counting code to the methods of inserting data and reading data, respectively, and outputs time-consuming through the console.

5.7. *Test Results.* After the above functional tests of each module of the data acquisition tool, it shows the functions of each module, passing the test, and are in line with expectations. The functional test results show that the data acquisition tool can complete the functions of general data acquisition and custom data acquisition and has the function of local database storage. The evaluation of the data storage service based on the clustering algorithm is no different than the previous data storage service. Through functional testing, the mobile game data collection tool has met the expectations of functional design requirements and completed the basic functions of data collection and data reporting [23, 24]. The performance test of the system includes the data acquisition performance test and the data read and write performance test. If the test passes, it means that the test results of the system meet the expected values and perfectly fit the normal operation of the game application.

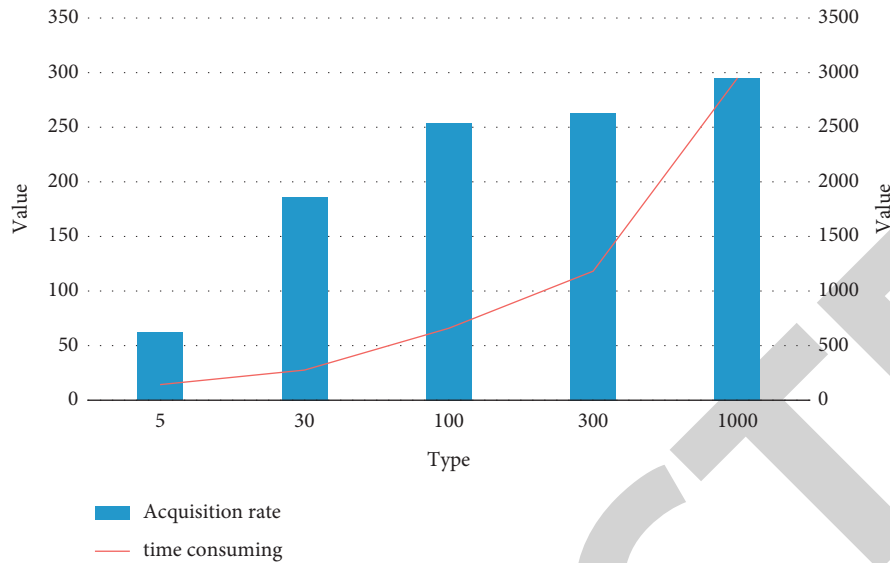


FIGURE 9: Relationship between time consumption, acquisition rate, and number of acquisitions.

6. Conclusions

This paper describes the design and research of a game data acquisition system based on artificial intelligence algorithms. The whole paper used artificial neural network algorithm in artificial intelligence algorithm and support vector machine algorithm to construct four modules of game data acquisition system: general data acquisition module, automatic data acquisition module, user behavior data acquisition module and data reporting module. This paper used the cluster analysis algorithm to conduct research and evaluation. It can be seen from the evaluation results that the mobile game application data collection tool functionally fulfills the functional requirements of general data collection, custom data collection, event data collection, data reporting, and back-end data storage. In addition, there is no stuttering phenomenon in the process of collecting data, which is in line with the performance requirements. It can be seen from the above that the game application data collection system provides basic data collection functions for individual developers and small and medium-sized enterprises, which helps users to collect data generated by users when operating games faster and more conveniently, to provide an important data basis for the subsequent improvement of product experience and optimization of product strategy. Mainly in the following aspects: only relying on client-side statistics cannot fully collect data, and some data still needs the cooperation of the server to complete. Taking the statistics of crash data as an example, the crash data obtained by the client-side statistics can only show the situation of crash occurring on this single mobile phone device and cannot obtain the total number of crash occurrences of the game application that day from a macro perspective. The ranking list of the total number of occurrences after aggregation of the same crash type, whether the number of crashes today has increased or decreased compared with the number of crashes yesterday, and so on.

Therefore, after the client collects the data and reports it to the server, the server needs to classify, aggregate, and count the data to maximize the utilization of the collected data.

Data Availability

The data underlying the results presented in the study are included within the manuscript.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] M. Wang, L. Zhao, R. Du et al., "A novel hybrid method of forecasting crude oil prices using complex network science and artificial intelligence algorithms," *Applied Energy*, vol. 220, pp. 480–495, 2018.
- [2] K. Majetta, C. Clauß, and C. Nytsch-Geusen, "Provision of data to use in artificial intelligence algorithms for single room heating," *Electronics*, vol. 10, no. 4, 523 pages, 2021.
- [3] L. Wang, "Role of machine learning and artificial intelligence algorithms for teaching reform of linguistics," *Journal of Intelligent and Fuzzy Systems*, vol. 40, no. 2, pp. 3251–3262, 2021.
- [4] O. M. Grath, M. W. Sarfraz, A. Gupta, Y. Yang, and T. Aslam, "Clinical utility of artificial intelligence algorithms to enhance wide-field optical coherence tomography angiography images," *Journal of Imaging*, vol. 7, no. 2, pp. 32–35, 2021.
- [5] M. Zakaria, M. A. Malek, M. Zolkepli, and A. N. Ahmed, "Application of artificial intelligence algorithms for hourly river level forecast: a case study of Muda River, Malaysia," *AEE - Alexandria Engineering Journal*, vol. 60, no. 4, pp. 4015–4028, 2021.
- [6] A. Sebastianelli, F. Mauro, G. D. Cosmo, F. Passarini, M. Carminati, and S. L. Ullo, "AIRSENSE-TO-ACT: a concept paper for COVID-19 countermeasures based on artificial intelligence algorithms and multi-source data processing,"

- International Journal of Geo-Information*, vol. 10, no. 1, pp. 34-35, 2021.
- [7] J. Liu, M. Li, W. C. Tang, and S. M. N. Islam, "A cyber physical system crowdsourcing inference method based on tempering: an advancement in artificial intelligence algorithms," *Wireless Communications and Mobile Computing*, vol. 2021, no. 1, Article ID 6618980, 2021.
- [8] S. Mirotta, P. Querre, J. Baccou, A. Gerbaud, and T. Gerbaud, "Use of Artificial Intelligence algorithms for hodoscope measurement interpretations," *Nuclear Instruments and Methods in Physics Research Section A Accelerators Spectrometers Detectors and Associated Equipment*, vol. 1010, no. 152, Article ID 165432, 2021.
- [9] P. Nowakowski, K. Szwarc, and U. Boryczka, "Vehicle route planning in e-waste mobile collection on demand supported by artificial intelligence algorithms," *Transportation Research Part D: Transport and Environment*, vol. 63, pp. 1-22, 2018.
- [10] A. Barcelos, F. M. Mazzoni, and A. Cardoso, "Bearing damage analysis with artificial intelligence algorithms," *Journal of Control, Automation and Electrical Systems*, vol. 33, no. 1, pp. 282-292, 2021.
- [11] A. Prabaharan, "Comparative study of artificial intelligence algorithms for autonomous vehicle," *International Journal of Science and Research*, vol. 9, no. 1, pp. 1579-1584, 2020.
- [12] D. P. Mihai, C. Trif, G. Stancov, D. Radulescu, and G. M. Nitulescu, "Artificial intelligence algorithms for discovering new active compounds targeting TRPA1 pain receptors," *Journal of AIDS*, vol. 1, no. 2, pp. 276-285, 2020.
- [13] W. Chen, Z. Sun, X. Zhao, X. Lei, A. Shirzadi, and H. Shahabi, "Performance evaluation and comparison of bivariate statistical-based artificial intelligence algorithms for spatial prediction of landslides," *ISPRS International Journal of Geo-Information*, vol. 9, no. 12, 696 pages, 2020.
- [14] X. Wang, G. Chen, H. Gong, and J. Jiang, "UAV swarm autonomous control based on Internet of Things and artificial intelligence algorithms," *Journal of Intelligent and Fuzzy Systems*, vol. 40, no. 3, pp. 7121-7133, 2020.
- [15] M. S. Jung, W. S. Dong, and D. W. Lim, "Development of a forward head posture warning system for PC users by using artificial intelligence algorithms," *Journal of Institute of Control*, vol. 26, no. 5, pp. 355-362, 2020.
- [16] R. Benotsmane, L. Dudás, and G. Kovács, "Survey on artificial intelligence algorithms used in industrial robotics," *Multi-disciplinárís Tudományok*, vol. 10, no. 4, pp. 194-205, 2020.
- [17] Y. Wang, R. Sun, W. Sun, and T. Yu, "Research on algorithms of fusion system of artificial intelligence university curriculum derivatives based on Internet big data," *Journal of Physics: Conference Series*, vol. 1952, no. 4, Article ID 042044, 2021.
- [18] F. Wang, L. P. Casalino, and D. Khullar, "Artificial intelligence algorithms for medical prediction should Be nonproprietary and readily available-reply," *JAMA Internal Medicine*, vol. 179, no. 5, pp. 731-732, 2019.
- [19] S. Almuhaideb, O. Alswailem, N. Alsubaie, I. Ferwana, and A. Alnajem, "Prediction of hospital no-show appointments through artificial intelligence algorithms," *Annals of Saudi Medicine*, vol. 39, no. 6, pp. 373-381, 2019.
- [20] T. J. Brinker, A. Hekler, A. Hauschild et al., "Comparing artificial intelligence algorithms to 157 German dermatologists: the melanoma classification benchmark," *European Journal of Cancer*, vol. 111, no. 2019, pp. 30-37, 2019.
- [21] T. Bao, L. Cheng, B. Chen, and T. Yu, "Software design of reactive power optimization analysis based on artificial intelligence algorithms," *Dianli Xitong Baohu yu Kongzhi/Power System Protection and Control*, vol. 46, no. 3, pp. 89-96, 2018.
- [22] X. Lin, C. Lin, X. Wang, J. Xue, and E. Zheng, "Precise positioning of agricultural vehicles under static conditions via artificial intelligence algorithms," *International Agricultural Engineering Journal*, vol. 27, no. 3, pp. 248-254, 2018.
- [23] R. Allen and D. Masters, "Artificial Intelligence: the right to protection from discrimination caused by algorithms, machine learning and automated decision-making," *ERA Forum*, vol. 20, no. 4, pp. 585-598, 2020.
- [24] A. S. Bakirov and I. E. Suleimenov, "ON the possibility of implementing artificial intelligence systems based on error-correcting code algorithms," *Journal of Theoretical and Applied Information Technology*, vol. 99, no. 1, pp. 83-84, 2021.