Hindawi
Mobile Information Systems
Volume 2022, Article ID 7145588, 11 pages
https://doi.org/10.1155/2022/7145588

Research Article

Application of Artificial Intelligence in Digital Games Based on Mathematical Statistics

Lian Xue

School of Computer and Computing Science, Zhejiang University City College, Hangzhou, 310015 Zhejiang, China

Correspondence should be addressed to Lian Xue; xuel@zucc.edu.cn

Received 8 March 2022; Revised 13 April 2022; Accepted 22 April 2022; Published 5 May 2022

Academic Editor: Chia-Huei Wu

Digital games are various games designed and developed with digital technology and implemented on digital equipment. With the active development of the modern game industry, related technologies such as real-time graphics rendering, realistic interaction, and artificial intelligence of games are also constantly improving. Among them, the artificial intelligence technology of games is limited by the development of theoretical artificial intelligence and the calculation time of real-time systems, and the development lags behind graphics and interactive technology. In order to solve these problems, this paper proposes the application of artificial intelligence in digital games based on mathematical statistics methods, aimed at studying and improving the intelligence level of digital games. The method of this paper is to study the mathematical statistics method and the basic principle of Sarsa learning algorithm and then propose the technology of artificial intelligence applied to digital games. The role of these methods is to study different types of mathematical statistics, to study the behavioral tree of artificial intelligence in digital games and the development prospects of the game, and to update the value of artificial intelligence in each iteration according to the Sarsa algorithm formula. This paper proposes a decision-making system to improve the intelligence level of digital games by analyzing the research status of digital games and the mathematical statistics of basketball digital games. In the basketball game experiment, the difference between the operation data of the decision-making system proposed in this paper and the operation data of human players in five game items reaches 0.51, 0.83, 0.58, 0.49, and 0.78, respectively.

1. Introduction

Artificial intelligence has been used in computer games since the very beginning and has been an integral part of gaming, unchanged from the days of Pong and Pac-Man until recently. This paper studies the application of mathematical statistics in classification and correlation analysis. It studies the principles of group customer valuation, construction of relevant indicators, indicator screening methods and weights, customer stratification, and customer value evaluation system, as well as customer stratification service standards and occupancy rates based on this model. In the past ten years, the computing speed of computer hardware has been significantly improved, which has driven the innovation of related computing technologies and led to the rapid development of the game industry. Game sales will increase year by year, many companies have invested in the game industry, and some big-budget games cost hundreds of millions of dollars to develop.

In recent years, research on artificial intelligence in games has gradually increased. However, while most research on artificial intelligence in games is limited to the development and technical aspects of game engines, the development of technical aspects also affects visual effects and through the pursuit of "real" visual effects. Artificial intelligence can respond to technology, improve advanced technology, and bring new inspiration and creativity to research at the technical level. This topic combines technology and art and comprehensively studies in-game artificial intelligence technology from multiple perspectives, as well as the multilevel impact of artificial intelligence technology on game intelligence.

The innovation of this paper is to propose a technology based on Sarsa algorithm for behavior tree optimization of digital games, which improve the way of creating artificial intelligence behavior trees in digital games, and provide a behavior tree workflow that can use the algorithm. This paper summarizes the respective characteristics of digital games and artificial intelligence technology according to the development status of digital games and artificial intelligence and demonstrates the technical feasibility of artificial intelligence technology to participate in digital game creation. This paper uses artificial intelligence technology to find a creative path for digital games to meet human needs and discusses the transformation of digital game creation concepts and the in-depth exploration of specific creative ideas in which artificial intelligence technology participates. It proposes the concept of interactive artificial intelligence on the basis of artificial intelligence games and discusses the development process of artificial intelligence technology participating in the creation of digital games.

2. Related Work

The development of digital games is changing with each passing day, and the research on them is also increasing. How to enhance the application of artificial intelligence in games is a hot research area in academia. Zhao proposed the definite (limit) value calculation formula of RM and DFM test values. He established a mathematical statistical method of residual static torque control by defining relevant parameters, limiting the corresponding torque limit value, the ratio of the measured value to the true value, and the confidence coefficient related to the confidence degree and confidence probability, which improved the residual torque control [1]. The aim of Mbbchir was to develop a statistical model based on patient parameters to predict the length of stay in the intensive care unit after cardiac surgery. The model was used to classify patients into short-term or long-term hospitalizations, and the measurements and main results of the experiment showed that EuroSCORE could be used to differentiate and predict patients for various surgical procedures in the ICU [2]. Alekseev studied the degree of multiple dependencies of wavelet transform coefficients obtained from the parameters of network packets, calculated standard deviations on this basis, and clustered the obtained coefficients to identify anomalies of the investigated network. He proposed to use the mathematical tool of wavelet transform to obtain clusters of coefficients to detect characteristics in major network traffic [3]. Hassabis et al. believe that a better understanding of biological brains could play a crucial role in building intelligent machines. They investigate the historical interaction between the fields of artificial intelligence and neuroscience and highlight current advances in artificial intelligence that may be key to advancing future research in both fields [4]. Makridakis surveys the current state of AI applications in healthcare and discusses its future. He studied the application of artificial intelligence in the treatment of stroke disease in detail, mainly analyzing the three main areas of early detection and diagnosis of patients, treatment, and outcome prediction and prognosis evaluation [5]. Liu et al. attempted to conduct a comprehensive review of artificial intelligence algorithms in rotating machinery fault diagnosis from both theoretical background and industrial application [6]. Sam calls the pattern of recipients and donors responding with an intelligent strategy a “numbers game.” She conducted in-depth interviews and observations on the daily life of councillors in a city, and based on existing data, she studied the preeminent position of outcome-based aid in the digital game [7].

3. Mathematical and Statistical Methods of Artificial Intelligence

3.1. Mathematical Statistical Methods. The basic methods of mathematical statistics include stochastic simulation, parameter estimation (Bayesian statistics), hypothesis testing, and statistical decision theory [8]. Mathematical statistics can usually be divided into the following categories: The first category is sampling survey and experimental design. Discussions are primarily concerned with theoretical and methodological issues in the collection of observational and experimental data, as well as statistical inference. The second type of task is mainly to explain the principles and methods of statistical inference. The third category includes product sampling inspection, statistical quality control, and reliability statistics. Mathematical statistics according to the application of mathematical statistics method classification are usually divided into three categories, that is, thinking method, descriptive method, and statistical inference method.

3.1.1. Random Simulation. The stochastic differential equations (i.e., the chemical Langevin equations) are described below. First, this paper presents the basic concepts of Poisson random variables and Poisson processes. Poisson random variable: X is a Poisson random variable with parameter q, represented by X to $P(q)$, and obeys a distribution.

$$\mathrm{Prob}(X = n) = \frac{q^n e^{-q}}{n!}, n = 0, 1, 2 \cdots n. \tag{1}$$

And the function form of the stochastic differential equation is as follows:

$$Px_n = q(t, x)et + \sigma(t, x_t)en_t, \tag{2}$$

where the mean and variance of X are both q.

Poisson process: $X(t)$ is a Poisson process with intensity q. If $X(0) = 0$, then the probability of $X(t)$ increasing by more than 2 in a very short time interval h is higher than h. The probability of $O(h) = 1$ is $qh + O(h)$. The Poisson process is a continuous time counting process. One of its important characteristics is that the number of events recorded by the Poisson process with intensity q follows a Poisson distribution with parameters within a period $[t, t + a)$, that is,

$$X(t + a) - X(t) \sim P(qa). \tag{3}$$

Therefore, the evolution of a random chemical reaction system can be written as the following stochastic simulation equation:

$$X(t) = X(0) + \sum_{j=1}^{M} K_j(t)v_j. \tag{4}$$

$Kj(t)$ represents the total number of occurrences of reaction Rj before time t. Since this is a random variable, $X(t)$ is also a random variable, so if $Kj(t + a)$ represents when the system state is $X(t) = x$, the number of times the reaction Rj occurs in time $[t, t + a)$ is as follows.

$$X(t + a) = X(t) + \sum_{j=1}^{M} K_j(t, a)v_j. \tag{5}$$

The above stochastic differential equation gives a good approximation of the stochastic process $X(t + a)$ if the system satisfies the following two conditions: The time interval a is large enough that the expected number of chemical reactions occurring at the time interval $[t, t + a)$ is much greater than 1. That is, it looks like this:

$$a_j(x)a > 1(\forall 1 < j < M). \tag{6}$$

Assuming that the propensity function $aj(X(t))$ corresponding to all responses in the time interval is almost constant,

$$a_j\left(X\left(t'\right)\right) \approx a_j(X(t)), \left(\forall t' \in [t, t + a), \forall 1 \leq j \leq M\right). \tag{7}$$

Typically, each reaction increases or decreases the amount of a particular molecule by one. Therefore, when the number of reactant molecules in the system is much larger than 1, as long as a is small enough, it is easy to satisfy the condition. Since the number of occurrences of reaction channel Rj in $[t, t + a)$ follows a Poisson distribution in the expected $aj(x)a$, equation (4) can be rewritten as follows.

$$X(t + a) = X(t) + \sum_{j=1}^{M} P_j\left(a_j(X(t)a\right)v_j. \tag{8}$$

At this time, the Poisson random variable $Pj(aj(X(t))a)$ can be approximated as a normally distributed random variable with the same mean and variance.

$$P_j\left(a_j(X(t)a\right) \approx N\left(a_j(X(t))a, a_j(X(t)a)\right), a_jX(t)a > 1. \tag{9}$$

So equation (7) can be rewritten approximately as follows:

$$X(t + a) = X(t) + \sum_{j=1}^{M} N_j\left(a_j(X(t))a, X(t)a\right)v_j \tag{10}$$

Note that this article uses a normally distributed continuous random variable to obtain a Poisson random variable with approximately integer values. Taking advantage of the properties of the normal distribution, equation (9) can be rewritten in the following format.

$$X(t + a) = X(t) + \sum_{j=1}^{M} a_jX(t)av_j + \sum_{j=1}^{M} \left[a_jX(t)a\right]^{0.5} v_jN_j(0, 1). \tag{11}$$

Next, when q is dt, time t is white noise $Ej(t)$ and represents a random variable satisfying an independent normal distribution $Nj(0, 1)$, and equation (10) is transformed as follows.

$$X(t + a) = X(t) + \sum_{j=1}^{M} a_jX(t)av_jdt + \sum_{j=1}^{M} a_j^{0.5}X(t)v_jE_j(t)(dt)^{0.5} \tag{12}$$

Through the statistical analysis of the sample orbits, it can obtain the time course of the probability density of the system state and the statistical properties of interest in this paper.

3.1.2. Bayesian Statistics. Bayesian network parameter learning refers to the process of learning the conditional probability distribution table of nodes from training data on the basis of a given network structure. Parameter learning can also be called parameter estimation, and incorrect model parameters will affect subsequent inference results. Bayesian network structure learning is to learn the corresponding DAG structure from the training sample data, as shown in Figure 1.

If the set of random variables $C = \{C1, C2, \cdots, n\}$ and the training dataset $G = \{G(1), G(2), \cdots, G(n)\}$ for these variables, where n is the number of samples, the purpose of structure learning is to output the corresponding DAG graph structure G. The sample carrier holding device may each include a retainer portion including a vacuum cup bonded material electromagnet or a mechanical device configured to hold the sample carrier. When there are few variables, such as only 1 or 2 variables, its structure can be easily determined. However, when the number of nodes increases, the corresponding possible directed acyclic graph structure also shows an exponential increase. The following function is satisfied between the number of DAGs $g(, z)$ and the number of nodes n:

$$g(n) = \begin{cases} 1, & n = 1, \\ \sum_{i=1}^{n} (-1)^{i+1} C_n^i 2^{i(n-i)} g(n-1), & n > 1. \end{cases} \tag{13}$$

Obviously, $g(5) = 29281$ and $g(10) = 4.2 \times 1018$, and it is absolutely impossible to simply build the structure manually. The structure learning method based on the search score regards the Bayesian network structure learning problem as an optimization problem and uses the search algorithm to find the network structure with the highest score

	S	L	B	X	D
1	True	False	False	True	True
2	False	False	False	True	False
3	True	False	True	True	False
⋮					

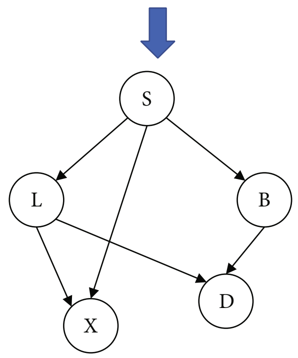


Figure 1: Bayesian network structure learning.

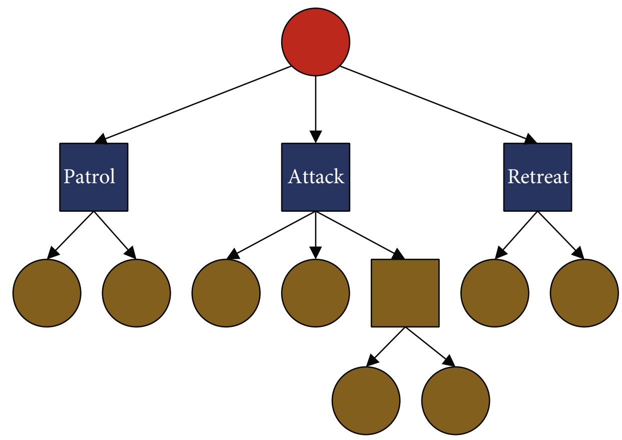


Figure 2: Game character behavior tree structure.

through the scoring function of the specified structure. The mathematical model of structural learning based on search scoring can be expressed as follows:

$$\begin{cases} \max f(G, D), \\ s.t. G \in g, G| = C. \end{cases} \quad (14)$$

Among them, f is the structure scoring function, g is the structure space (that is, the possible structures), and $G| = C$ means that the structure G satisfies the constraint C. Constraint C represents the number of samples, which will be processed according to the formula below when certain parameters are obtained. Therefore, the optimal structure $G*$ can be expressed as follows:

$$G^* = \arg \max_G f(G, D). \quad (15)$$

In structure learning based on search scoring, the first problem is how to calculate its score $f(G, D)$ given training data D and a possible structure G. The Bayesian-based scoring function treats equation (14) as the following MAP problem:

$$G^* = \arg \max_G P(G|D), \quad (16)$$

where $P(G|D)$ is the posterior probability of structure G given D. Assuming that the prior probability of G is $P(G)$, according to the Bayesian formula, it can be known that

$$P(G|D) = \frac{P(D|G)P(G)}{P(D)}. \quad (17)$$

Since $P(D)$ is independent of G, the logarithm of both sides can be obtained:

$$\log P(G|D) = \log P(D|G) + \log P(G). \quad (18)$$

Assume that the parameters of the model structure G are as follows:

$$P(D|G) = \int_{\Delta G}^{0} P(D|G, \Delta G) P(\Delta G|G) d\Delta G. \quad (19)$$

3.1.3. Statistical Decision-Making. Fuzzy Statistical Decision Theory has no such definition in previous research, and here is a brand new definition name. The features of this project have been explained in detail in the previous section and will not be repeated here. Based on the abovementioned characteristics, there is no method fully adapting to the abovementioned problems in the existing methods. Therefore, a new comprehensive integrated method is proposed here to solve the problem of risk assessment of such projects, hoping to achieve a better assessment effect than ever before. The challenge of this chapter is to study the risk factor assessment method of the project, which is mainly based on the combination of fuzzy hierarchical structure and AHP method. On the basis of the two-dimensional comprehensive evaluation method adopted by the previous scholars, it avoids its shortcomings and establishes a four-dimensional comprehensive evaluation system.

3.2. Artificial Intelligence Applied to Digital Games. Games are a peculiar form of human life, but also an integral part of human material production life. From time immemorial or even earlier, there have been games where human civilization has emerged. In ancient China, there were widely circulated and well-known games, such as pitching pots and rolling balls. At the same time, humans also have games for different age groups, from building blocks to childhood jigsaw puzzles, to adult board games, sports games, and more. In fact, the game is not just for fun, and the game itself has certain characteristics. Games can not only bring joy to people but also entertain and educate people, which is very important for the integration of different cultures. Digital games are developing rapidly and have become popular all over the world. As an emerging cultural and creative industry, digital games have better development prospects than other cultural and creative industries. At the same time, the sales of digital game software, hardware, and the game

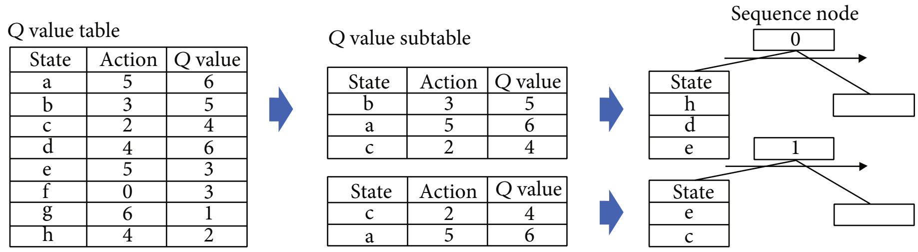


Figure 3: Inserting a conditional node into the behavior tree based on the A value.

itself are growing rapidly at this stage. Game companies represented by Microsoft, Sony, Nin-tendon, etc., are constantly launching refreshing products and achieving great success [9].

Generally speaking, game AI is divided into two layers, decision-making and behavior. The behavior layer can be understood as a middleware in the game. The upper layer is the decision logic, and the lower layer is the animation resources. The work goal is to convert AI requests into animation playback requests. A puppy's behavioral definitions may include the following: walking, running, sleeping, eating, and drinking. These can be thought of as a series of its actions or logical actions. These behaviors form a collection called a behavior pool. A behavior tree is an efficient way to organize these behaviors.

Behavior tree (behavior tree, BT for short) is a tree structure containing hierarchical nodes used to control AI decision behavior. Each time an update is performed, the behavior tree runs a depth-first scan until the underlying leaf node returns success or the status changes to "running" [10]. According to this traversal mechanism, the AI needs to prioritize the behavior tree, and this behavior may need to be placed in the left branch of the behavior tree at design time.

Here we use the behavior of a soldier in the game as an example to introduce the structure of the behavior tree. Figure 2 shows the behavior tree of a game character.

When it comes to deciding what kind of behavior the current soldier is going to do, we are going to search the tree from the top down, through some criteria, and finally determine the leaf nodes that need to be done and execute it. The input to the algorithm is a behavior tree, and the tree is first analyzed to find the deepest sequence nodes, which act as reinforcement learning actions to establish the A values corresponding to the A value table.

The generated A value table can be easily split into subtables, the A value subtable of actions corresponding to actions in the behavior tree. The subtable A values are sorted from high to low, and the parameter selection corresponds to the state with the highest percentage A value [11]. Figure 3 illustrates this process, and the main table is divided into subtables by actions, placed in conditional nodes instead of conditional nodes. Subsequent algorithms filter the corresponding states based on the parameters, leaving only a specific percentage of the highest A values.

In this chapter, we will design and implement a popular AI engine based on Unity3D to support common game types and common game AI key technologies on the market. This article will introduce the general design and overall architecture of the engine. The engine is divided into three parts: basic framework class, intelligent agent architecture, and message mechanism. Next, this paper analyzes the design and implementation of these three parts in detail. The main basic components supporting the engine are the basic framework classes and the design and implementation of decision-making systems, perception systems, and behavioral systems [12]. This article introduces the intelligent agent in detail and finally explains the main role of the message mechanism and the message types used in the engine. The general AI engine implemented in this chapter meets the performance characteristics of generality, efficiency, flexibility, extensibility, customization, and platform expansion. It can help game developers quickly build the artificial intelligence required in the game, make the characters and creatures in the game more lifelike, and bring players a more "immersive" game experience.

3.3. Basic Principles of Sarsa Learning Algorithm. Sarsa learning algorithm belongs to reinforcement learning staggered learning, which is an in-policy learning method for solving decision/control problems. The Sarsa algorithm estimates the action value function $Q^{\pi}(s, a)$ of all executable actions a in any state s under policy π. Here, the TD prediction algorithm can be used to estimate the Q-function [13]. The equation for the Sarsa algorithm to update the action value is as follows:

$$Q(s_t, a_t) = Q(s_t, a_t) + a[r_{t+1} + vQ(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]. \tag{20}$$

Among them, $Q(s_t, a_t)$ is the action value of taking action a in the current state s, and $Q(s_{t+1}, a_{t+1})$ is the action value of taking action in the next state. $rt + 1$ is the instantaneous reward of performing the action in this state, t is the number of iterations, a is the learning rate, and v is the conversion factor.

The Sarsa algorithm relies on two tables:

One is the state-action reward table (called the R table), where each row represents a state. Each column represents an action, and the table value represents the reward for taking that action in that state. By defining the table value of the R table, it can define a reward for the action in each state. With the continuous progress of the algorithm, the rewards of different actions in different states will affect the weights of the corresponding actions in each state. Generally speaking, the value of a row and a column on the R table is

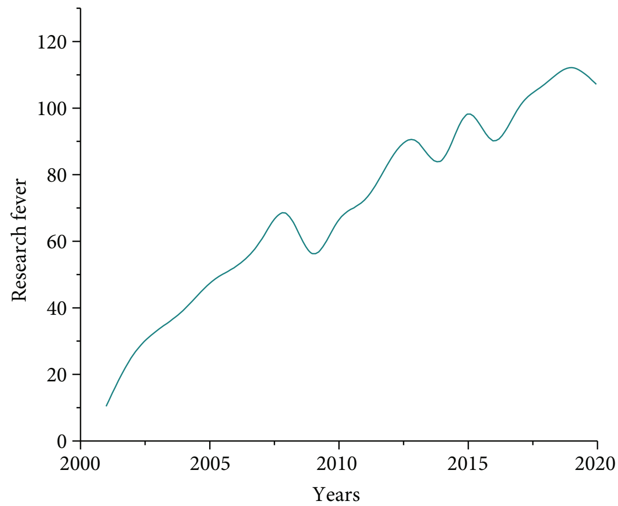


FIGURE 4: Trends in digital games research.

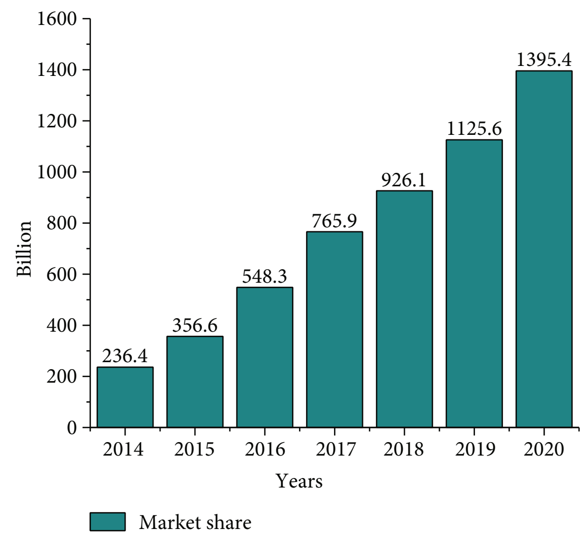


FIGURE 5: The development scale of domestic mobile games.

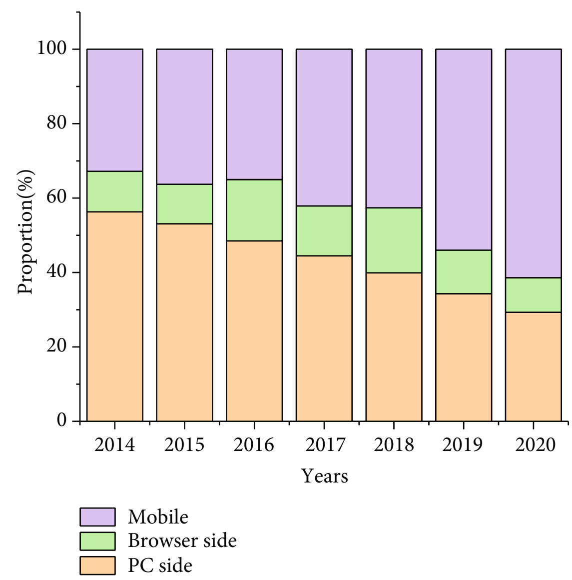


FIGURE 6: Segmentation of the domestic game industry.

positively correlated with the weight of the action corresponding to the final corresponding state [14].

One is a state-action table (called a Q table). Each Q table corresponds to an R table with the same number of rows and columns. The table value of the Q table represents the weight of the corresponding action in the corresponding state, and the weight is used by the artificial intelligence to make judgments. Generally speaking, after determining the state, the algorithm will choose the action with the larger weight [15]. The table value is not constant, and it updates its own value in each iteration according to the update equation. After many iterations (generally greater than 2000), the values in the table tend to stabilize.

$A*$ algorithm is the most effective direct search method to solve the shortest path in static road network and also an effective algorithm to solve many search problems. The closer the distance estimated value in the algorithm is to the actual value, the faster the final search speed. The $A*$ algorithm uses a heuristic estimation function h to predict the remaining distance between the current node and the endpoint. When $f(v)$ is used to represent the path cost from the starting point S to the destination T through node v, the ordered queue usually has nodes v sorted according to $f(v)$ from small to large. The equation for calculating $f(v)$ by the $A*$ algorithm is as follows:

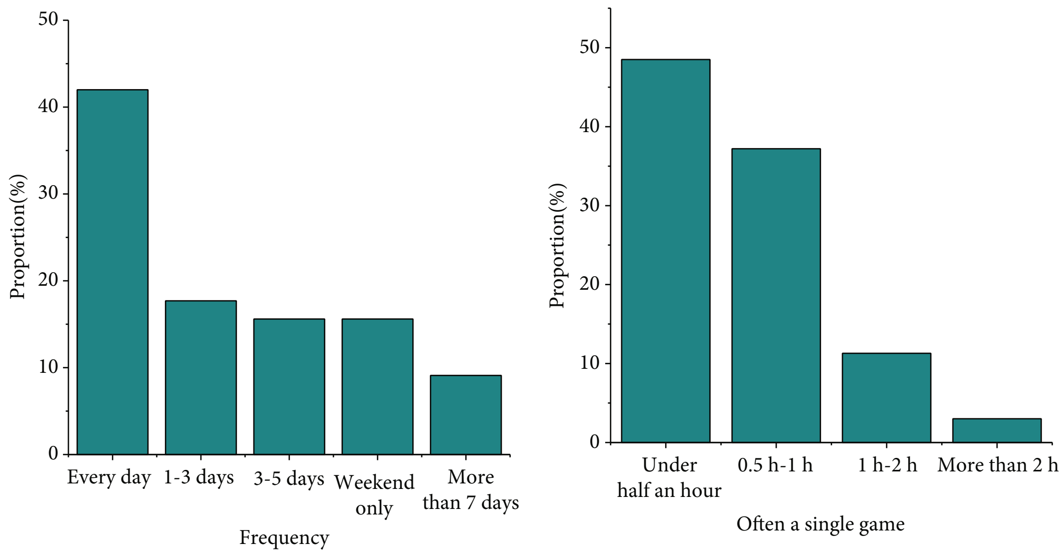
$$f(v) = g(S, v) + h(v, T). \quad (21)$$

In the equation, $g(S, v)$ is the path cost from the starting point S to the node v, and $h(v, T)$ is the estimation of the remaining distance from the current point v to the end point T. Dijkstra's algorithm and A∗ algorithm are traditional shortest path algorithms, they can solve the shortest path problem in static environment. However, in a dynamic environment, the weights on each road section change in real time, and the traditional shortest path algorithm cannot divert vehicles, which will lead to excessive vehicle concentration and congestion transfer [16].

4. AI Experimentation and Analysis in Digital Games

4.1. Status Quo of Digital Game Research. After searching on the Internet using keywords and consulting existing related books, it was found that the research on digital games is relatively complete, and almost all aspects of digital games have been involved. Figure 4 shows the trend in the number of studies on digital games in recent years.

The main reasons why there is so much research on digital games are as follows: The design of digital games itself is a very complex and huge project, which will touch many disciplines and fields such as cultural studies, ethics, design studies, computer science, psychology, and communication



(a) Frequency of children playing number games

(b) Single duration of children playing number games

Figure 7: Frequency and duration of children playing with digital devices.

studies [17]. The industrial value formed by digital games has given a strong impetus to related theoretical research. Figure 5 shows the research data of the survey agency on the volume of China's digital game market from 2014 to 2020.

It can be seen that the game market in the mobile field has a relatively huge development space. After more than half a century of development, digital games have become a mature industry with high technology, high development speed, and high profit. Compared with foreign countries, domestic digital games started relatively late. However, in the domestic market environment where the mobile network is mature and there are many users, mobile games have developed rapidly with the advantages of device popularity, portability, immediacy, and interactivity, and their number is increasing day by day [18].

Figure 6 shows the survey data on the scale of China's online game market.

Some industry insiders pointed out that China is still in the development stage of digital games, but the phenomenon of game homogeneity and piracy is serious. Many games are of low quality, and user stickiness is small. The road of digital game design still needs to be continuously explored [19].

Figure 7 presents the results of a survey on the frequency and duration of digital play by children using digital devices. The largest percentage of children play digitally every day, with more than 90% playing digitally on a weekly basis. 85.7% of children play number games within one hour.

4.2. AI Applications in Digital Games. This chapter mainly conducts the actual operation and test of the prototype system that applies the intelligent decision-making model described in this paper. The configuration and environment of the test machine used are shown in Table 1.

In order to test the correctness of the algorithm, this paper uses a simulation experiment to verify it and completes the simulation experiment with an example of an application scenario [20]. There is only one agent in the experimental scene, called a gladiator. The following is the corresponding behavior definition: defense: resist a certain amount of damage. Reply: restore a certain amount of life. Normal attack: it deals damage based on attack power. Charged attack: it increases damage during consecutive attacks. Finishing attack: take a certain amount of damage, and deal more damage. Escape: go to another area.

After setting the behavior tree, due to the different effects caused by the actions of the gladiators, the effect caused by the behavior needs an accurate measurement standard, so it is necessary to make detailed settings for the interaction between the agents. The detailed rules are shown in Table 2.

It can be seen that in the second layer, the learned behavior tree selects nonaggressive behaviors in most cases. In the last layer, since there is no state corresponding to common attacks, the nodes of common attacks are directly deleted. The reason may be that the design strength of common attacks is unreasonable, resulting in being completely replaced by other behaviors [21]. This paper uses the behavior tree to conduct simulation experiments, and the experimental results are shown in Table 3.

From the above results, it can be seen that the survivability of the type 2 gladiator is higher. When the battle was fierce at the beginning of the experiment, the type 2 gladiator could have a greater survival rate and had an absolute advantage in the final number of survivors.

The training process is relatively long. If a long training process is required for each startup, the ease of use of the system will be greatly reduced [22]. In order to avoid repeated training, it is necessary to save the training results locally, so that the text service to be implemented can be quickly loaded when it is just started. Finally, the information of the obtained results stored in the form of a file is shown in Table 4.

Table 1: Operating environment.

Name	Parameter
Processor	Intel-i3-8100
RAM	DDR4 16G 2400 MHz
Graphics card	GTX750ti
Operating system	Windows10 x64
Emulator version	Android 7.1
Unity version	V2017.2.0f3 (64-bit)
AS version	V2.3.3
Django version	V2.0.3

Table 2: Gladiator's state and corresponding reward table.

(a)

Enemy count return				
Em	Recover	Defense	Attack	Run away
None	0	0	0	10
Have	5	2	5	0

(b)

Current life return				
Hp	Recover	Defense	Attack	Run away
None	-10	-10	-10	-10
Few	0	5	2	5
High	2	0	5	0

(c)

Enemy life return				
Ep	Recover	Defense	Attack	Run away
None	10	0	10	10
Few	2	0	5	0
High	0	0	5	0

Table 3: Comparison table of experimental results.

Evaluation item	Type 1	Type 2
Top 50 deaths	62.00	24.00
Top fifty percent of deaths	0.72	0.28
Average kills	2.50	1.20
The final number of survivors	1.00	4.00

It can be seen that the training results are stored in four files. Among them, npy is the data in the form of the numpy scientific operation library that gensim depends on, model is the model data, and vector is the corresponding vector data. The final single vector file size is 2.43 GB, which is in line with expectations.

In order to realize cross-language calling, the system uses packaging, network calling, and other methods, respectively. This paper measures the latency and compares it to the no-

Table 4: Training results.

File name	Type	Size
wiki.zh.text.model	Model	54658 KB
wiki.zh.text.model.syn0.npy	npy	1073191 KB
wiki.zh.text.model.syn1neg.npy	npy	1073191 KB
wiki.zh.text.vector	Vector	2555843 KB

network test version data, where the performance penalty due to the local network call itself is almost negligible [23]. Among them, the model loading takes a long time, but only when the text service is started for the first time, it has no effect during the system running. Overall, the system can basically meet the requirements of real-time performance. In addition, the proportion of resources used by the behavior tree module is relatively reasonable, and the entire system runs stably.

4.3. Mathematical Statistics of Basketball Number Game. In order to verify the effectiveness of game artificial intelligence, this paper designs and implements a 3V3 basketball game prototype. In order to analyze and compare the training algorithm of the BP neural network mentioned above, this paper records 2675 real-time game data of 25 games in 3V3 basketball game running by the method of automatic sample acquisition. Each game is divided into 15 rounds, and the maximum time for each round is 20 seconds. The real-time data is recorded once per second and output in the format of TXT, and the items are separated by spaces.

The experimental platform chooses MATLAB, and the sample is divided into two parts. The number of training samples is 860, the number of test samples is 1658, the minimum error value is 0.02, and the number of training is 2000. Choose different training algorithms when building a neural network and observe the error curve and memory consumption at the end of training. The error drop graphs for various training algorithms are shown in Figure 8.

The experimental platform is SPSS, and the total data volume is 1684, so the maximum number of clusters is 20. The DB values between 5 and 20 of the number of clusters and the DB values after increasing are counted, as shown in Figure 9. Through the observation of Figure 9, it is found that the DB value decreases with the increase of the number of cluster centers. The maximum DB value is 1, and the corresponding number of cluster centers is 20. At the end of the increase of cluster centers, there is no upward trend in the DB value, and it is necessary to further increase the number of cluster centers.

The sample data of the four neural networks are normalized and compared with the actual operation data. The abscissa is the name of the feature data, and the ordinate is the corresponding feature data value, as shown in Figure 10.

From the test results, the training effect of the total dataset is not ideal. The difference degrees of the first five feature data reached 0.51, 0.83, 0.58, 0.49, and 0.78, respectively, indicating that there is a big difference between the data after cluster analysis and the original data, resulting in inconsistency between the training results and the original data.

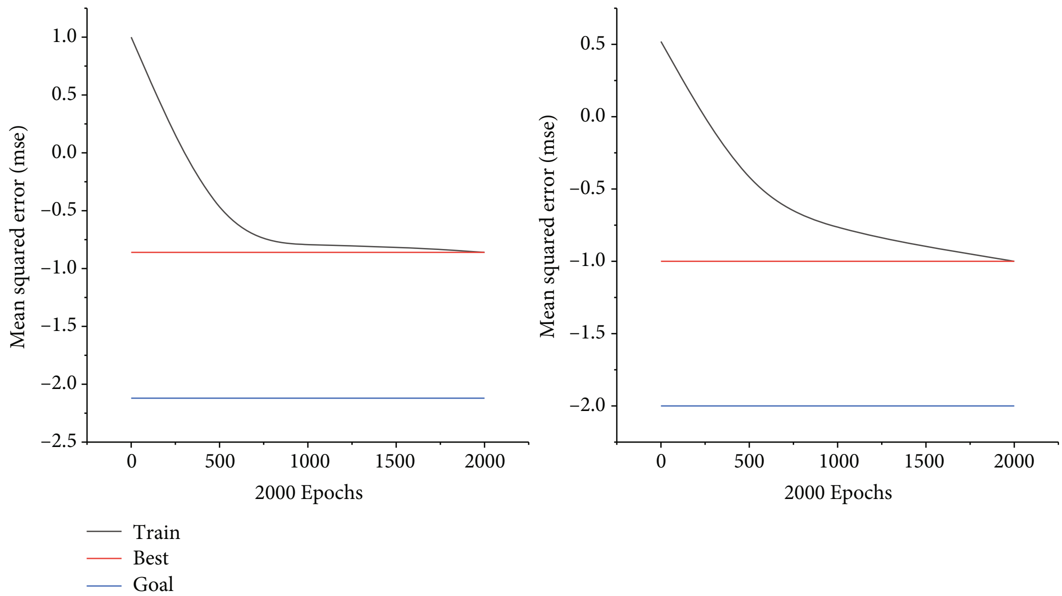


Figure 8: Gradient descent and gradient descent with momentum term.

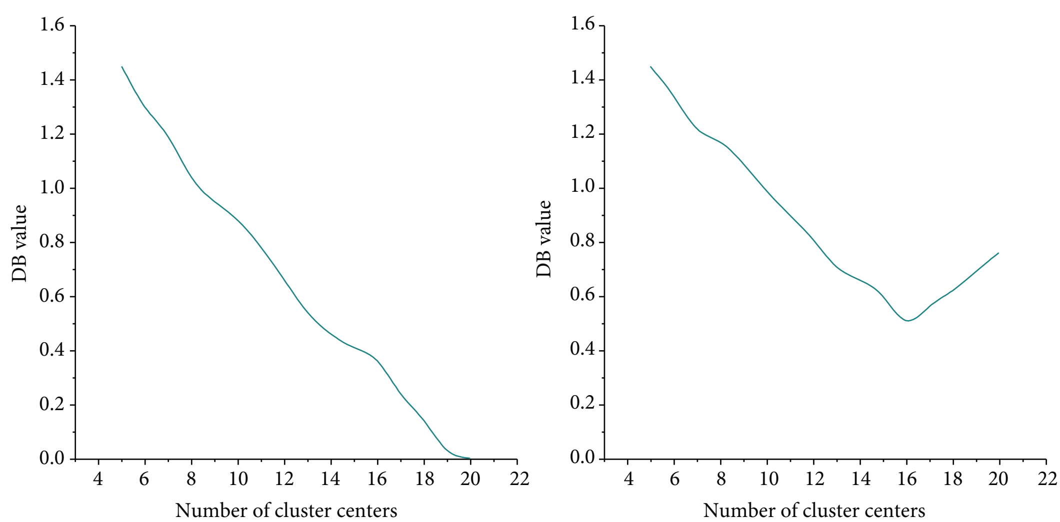


Figure 9: The DB value map corresponding to the number of cluster centers.

The sixth feature data scoring rate is a standard that reflects player intelligence, and the scoring rate of players trained by the total data set drops significantly. The reason is that there are contradictory data with different tactical inclinations in the sample, resulting in poor training effect. But compared to the total dataset, the game decision-making system trained by the sample sets from clusters 1 to N1 basically maintains the characteristics of the samples. Observing Figure 10, the first five feature data of the running data are basically the same as the sample feature values, and the difference does not exceed 20%. The score rate dropped slightly, because the intelligence of the neural network cannot fully reach the height of humans, and the drop is within the acceptable range [24, 25].

The experimental results show that after the secondary clustering, the decision-making system can distinguish the player's data according to the tactical tendency and then learn, realizing the diversification of the decision-making tendency [26, 27].

5. Discussion

The time complexity spectrum sensing based on mathematical statistics studied in this paper mainly starts from reducing the time complexity and is aimed at a single sensing user, without considering the method of multiuser cooperation. Through the process of information acquisition and processing, word segmentation and semantic recognition, and behavior tree decision-making, combined with the specific scenes and functions designed, the logic of the agent and the normal performance of the entire system are basically realized. In the process of design and implementation, some

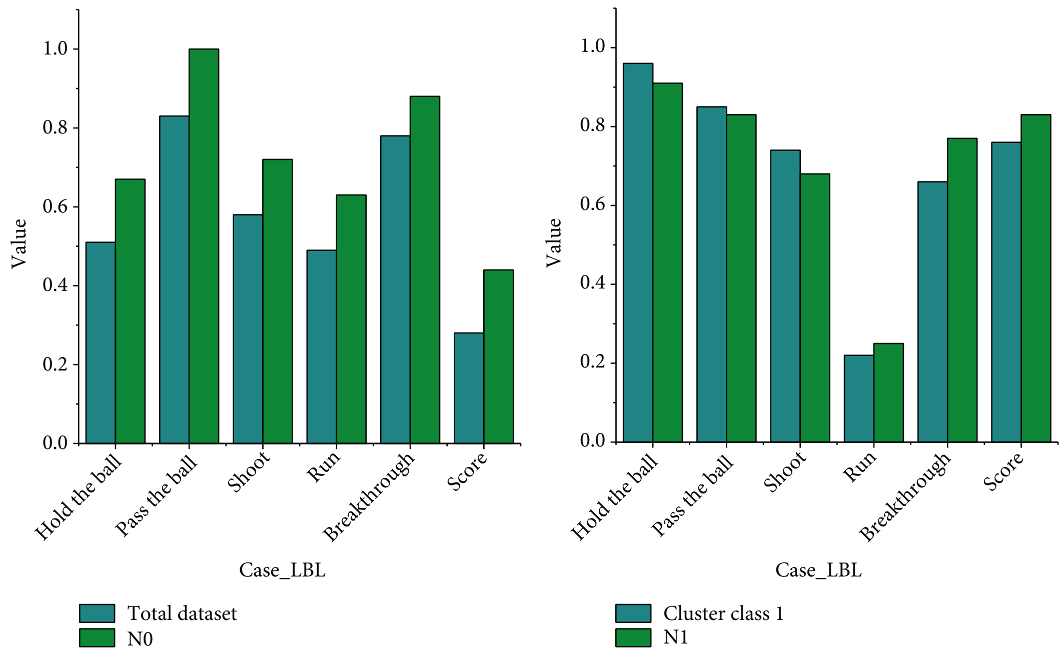


Figure 10: Comparison of total data set and N0 and comparison of clusters 1 and N1.

difficulties and problems occurred. Most of these difficulties and problems have been resolved, but there are still some areas of note or improvement. In the actual communication environment, due to the complex communication environment, it is necessary to adopt a multiuser cooperation method for spectrum detection, and at the same time, the existence of malicious users must be considered. Therefore, further research is needed in this aspect of multiuser collaboration. Although the method of constructing ideas has a wide range of applications in mathematical statistics, it is not easy to describe, so there are few typical examples in mathematical statistics involved in this article, which need to be further summarized.

6. Conclusions

Mathematical thinking is the soul of learning mathematics, and it is the idealism and basic strategy to guide people to deal with mathematical problems. In this paper, experiments use physics engine triggers to implement a perception system that provides input to intelligent models and a digital game-sharing artificial intelligence system that improves perception reliability. This paper proposes an action tree creation method based on Sarsa algorithm, which can optimize the structure of the action tree and improve its intelligence. On top of this, this paper designs a corresponding simulation experiment to verify its effectiveness. This article discusses the types of information and defines the structure and flow of processing. Among them, the implementation principle and method of speech recognition are emphatically studied. At the same time, the speech recognition SDK is introduced into the realized engineering project, and the cross-language call is realized by packaging, and finally, the speech input function is supported.

Data Availability

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

Conflicts of Interest

The author states that this article has no conflict of interest.

References

[1] Z. Zhao, “Mathematical statistics method of residual static torque controlling,” *International Journal of Plant Engineering and Management*, vol. 23, no. 3, pp. 42–45, 2018.

[2] K. Meadows, R. Gibbens, C. Gerrard, and A. Vuylsteke, “Prediction of patient length of stay on the intensive care unit following cardiac surgery: a logistic regression analysis based on the cardiac operative mortality risk calculator, Euro SCORE,” *Journal of Cardiothoracic and Vascular Anesthesia*, vol. 32, no. 6, pp. 2676–2682, 2018.

[3] I. V. Alekseev, “Detection of distributed denial of service attacks in large-scale networks based on methods of mathematical statistics and artificial intelligence,” *Automatic Control and Computer Sciences*, vol. 54, no. 8, pp. 952–957, 2020.

[4] D. Hassabis, D. Kumaran, C. Summerfield, and M. Botvinick, “Neuroscience-inspired artificial intelligence,” *Neuron*, vol. 95, no. 2, pp. 245–258, 2017.

[5] S. Makridakis, “The forthcoming artificial intelligence (AI) revolution: its impact on society and firms,” *Futures*, vol. 90, no. jun, pp. 46–60, 2017.

[6] R. Liu, B. Yang, E. Zio, and X. Chen, “Artificial intelligence for fault diagnosis of rotating machinery: a review,” *Mechanical Systems and Signal Processing*, vol. 108, no. AUG, pp. 33–47, 2018.

[7] F. Sam, “Staffing is not just a numbers game,” *British Journal of Nursing*, vol. 28, no. 7, pp. 481–481, 2019.

[8] A. Zeifman, “Preface of "the third symposium on applied problems in probability theory and mathematical Statistics related to modeling of information systems (APPT+MS)",” *AIP Conference Proceedings*, vol. 1863, no. 1, pp. 1-2, 2017.

[9] L. Angelis, “A course in mathematical statistics and large sample theory,” *Computing Reviews*, vol. 58, no. 6, pp. 334-335, 2017.

[10] L. M. Guzman and M. W. Pennell, “Mathematical Statistics for biologists and other interesting people,” *Trends in Ecology & Evolution*, vol. 34, no. 12, pp. 1064-1065, 2019.

[11] A. Bovier and L. Hartung, “Extended convergence of the extremal process of branching Brownian motion,” *Annals of Applied Probability*, vol. 27, no. 3, pp. 1757–1777, 2017.

[12] I. A. Ibragimov, M. A. Lifshits, A. I. Nazarov, and D. N. Zaporozhets, “On the history of St. Petersburg School of Probability and Mathematical Statistics: II. Random processes and dependent variables,” *Vestnik St*, vol. 51, no. 3, pp. 213–236, 2018.

[13] H. Lu, Y. Li, M. Chen, H. Kim, and S. Serikawa, “Brain intelligence: go beyond artificial intelligence,” *Mobile Networks and Applications*, vol. 23, no. 7553, pp. 368–375, 2018.

[14] C. Cath, S. Wachter, B. Mittelstadt, M. Taddeo, and L. Floridi, “Artificial intelligence and the 'Good Society': the US, EU, and UK approach,” *Science and Engineering Ethics*, vol. 24, no. 7625, pp. 1–24, 2017.

[15] P. Glauner, J. A. Meira, P. Valtchev, R. State, and F. Bettinger, “The challenge of non-technical loss detection using artificial intelligence: a survey,” *International Journal of Computational Intelligence Systems*, vol. 10, no. 1, pp. 760–775, 2017.

[16] J. H. Thrall, X. Li, Q. Li et al., “Artificial intelligence and machine learning in radiology: opportunities, challenges, pitfalls, and criteria for success,” *Journal of the American College of Radiology*, vol. 15, no. 3, pp. 504–508, 2018.

[17] L. Caviglione, M. Gaggero, J. F. Lalande, W. Mazurczyk, and M. Urbanski, “Seeing the unseen: revealing mobile malware hidden communications via energy consumption and artificial intelligence,” *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 4, pp. 799–810, 2016.

[18] M. Hutson, “Artificial intelligence faces reproducibility crisis,” *Science*, vol. 359, no. 6377, pp. 725-726, 2018.

[19] D. T. Bui, Q. T. Bui, Q. P. Nguyen, B. Pradhan, H. Nampak, and P. T. Trinh, “A hybrid artificial intelligence approach using GIS-based neural-fuzzy inference system and particle swarm optimization for forest fire susceptibility modeling at a tropical area,” *Agricultural and Forest Meteorology*, vol. 233, no. Complete, pp. 32–44, 2017.

[20] M. Nasr, A. Mahmoud, M. Fawzy, and A. Radwan, “Artificial intelligence modeling of cadmium (II) biosorption using rice straw,” *Applied Water Science*, vol. 7, no. 2, pp. 823–831, 2017.

[21] J. Lemley, S. Bazrafkan, and P. Corcoran, “Deep learning for consumer devices and services: pushing the limits for machine learning, artificial intelligence, and computer vision,” *IEEE Consumer Electronics Magazine*, vol. 6, no. 2, pp. 48–56, 2017.

[22] C. Chassonnery-Zaïgouche, “How economists entered the 'numbers game': measuring discrimination in the US courtrooms, 1971–1989,” *Journal of the History of Economic Thought*, vol. 42, no. 2, pp. 229–259, 2020.

[23] M. Sabbi and A. Stroh, “The 'Numbers Game': strategic reactions to results-based development assistance in Ghana,” *Studies in Comparative International Development (SCID)*, vol. 55, no. 1, pp. 77–98, 2020.

[24] R. Gonzalez, “Tracking Atlantic salmon smolts a way to win the ‘numbers game’ acoustic telemetry provides researchers the tool that makes the difference,” *Hatchery International*, vol. 19, no. 2, pp. 29–29, 2018.

[25] A. N. Khan, X. Cao, and A. H. Pitafi, “Personality traits as predictor of M-payment systems: aASEM-neural networks approach,” *Journal of Organizational and End User Computing (JOEUC)*, vol. 31, no. 4, pp. 89–110, 2019.

[26] X. Wang, H. Schneider, and K. R. Walsh, “A predictive analytics approach to building a decision support system for improving graduation rates at a four-year college,” *Journal of Organizational and End User Computing (JOEUC)*, vol. 32, no. 4, pp. 43–62, 2020.

[27] M. Daradkeh, “Determinants of self-service analytics adoption intention: the effect of task-technology fit, compatibility, and user empowerment,” *Journal of Organizational and End User Computing (JOEUC)*, vol. 31, no. 4, pp. 19–45, 2019.