Application of Big Data Analysis and Cloud Computing Technology

Jinfang Zhang  
Hebei University of Water Resources and Electric Engineering

Yajuan Li  (liyj366@163.com)  
Hebei University of Water Resources and Electric Engineering

Chunqian Zhang  
Hebei University of Water Resources and Electric Engineering

Research Article

Keywords: big data technology, Cloud Computing, network abnormal behavior detection, data acquisition component, distributed technology, machine learning

Posted Date: November 3rd, 2022

DOI: https://doi.org/10.21203/rs.3.rs-2221076/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License. Read Full License

Additional Declarations: No competing interests reported.
Application of Big Data Analysis and Cloud Computing Technology

Jinfang Zhang 1, Yajuan Li 1*, Chunqian Zhang 2

1 Computer Department, Hebei University of Water Resources and Electric Engineering, Cangzhou 061000, Hebei, China
2 Department of Electrical Automation, Hebei University of Water Resources and Electric Engineering, Cangzhou 061000, Hebei, China

* Corresponding Author: Yajuan Li, liyajuan@hbwe.edu.cn

Abstract

With the swift growth of computer science, technologies such as big data and artificial intelligence are widely used in various fields of modern society. The types of network equipment and the scope of network coverage have also increased rapidly. While the network brings convenience to people, more attention must be paid to the security of the network platform. The purpose is to safely and effectively manage the current rapidly growing Internet data and improve the ability to detect abnormal network behaviors. Combining big data technology and machine learning (ML), the application of big data analysis and cloud computing technology for network security are studied. Firstly, the data collection technology of abnormal network behavior is introduced, and the Flume data collection component and Kafka distributed technology are discussed. Secondly, the data processing process and corresponding algorithm processing of abnormal network behavior are analyzed, including ML framework and stream processing technology. Finally, the model of network abnormal behavior detection based on big data is constructed, and compared with the related model based on the decision tree and random forest (RF) algorithm, and verified by experiments. The verification results reveal that among the 42 attack types against the dataset, the detection accuracy of network abnormal behavior by big data is 96.4%, and the false positive rate is 2.23%, which is higher than that of decision tree and RF algorithm. This experimental study denotes that the network abnormal behavior detection technology of big data based on the ML framework can effectively improve the type and efficiency of network abnormal behavior detection, and has certain reference significance for improving network security management and control capabilities.

Keywords: big data technology; Cloud Computing; network abnormal behavior detection; data acquisition component; distributed technology; machine learning

1. Introduction

With the rapid progress of information science and technology, the amount and type of data generated worldwide are rising rapidly under the wide application of big data, artificial intelligence, cloud computing and other technologies [1]. The world has entered the era of big data. Due to the continuous increase of network equipment and scale, the data generated by users on the Internet is large and very messy [2]. In 2022, China Internet Network Information Center released the 49th Internet coordination report. By December 2021, the number of Internet users in China had reached 1.032 billion, an increase of 42.96 million compared with December 2020, and the Internet penetration rate was 73% [3]. The huge number of network users also implies that people must focus more on the security of the network environment [4]. In 2021, compared with 2020, the number of netizens in China who suffered from Internet attacks dropped, but 40% of them still suffered from Internet attacks. How to build the "Great Wall" of network security defense is still a crucial issue that cannot be avoided in the development of computers [5].
In order to deal with the network security threats in the current era, the advantages of big data technology in data collection, storage and calculation can be considered to be brought into play. Besides, big data technology can be applied to network abnormal behavior monitoring [6]. Big data technology and cloud computing network technology can improve the efficiency and accuracy of network abnormal behavior monitoring and reduce the error rate and processing time of reports [7]. Jnetpcap packet capture technology, Flume data acquisition component, distributed technology Kafka, big data processing framework Spark and cloud computing technology, streaming processing technology Spark Streaming, and data storage technology can provide strong technical support for computer network abnormal behavior monitoring [8].

This exploration combines big data technology and cloud computing network and studies the application of computer big data analysis and cloud computing network technology in the direction of network security. First, the data acquisition technology of abnormal network behavior is introduced, and the Flume data acquisition component and Kafka distributed technology are discussed. Next, the data processing process of abnormal network behavior and corresponding algorithm processing are studied, including the Support Vector Machine (SVM) and Simulated Annealing (SA) algorithm. The two algorithms are combined. Finally, a monitoring model is built based on big data technology and cloud computing network abnormal behavior. The abnormal network behavior monitoring system studied can better complete the parallel collection, feature extraction, real-time monitoring and result storage of massive data. It has certain reference significance for the construction of network security engineering.

2. Methods
2.1. SVM algorithm

SVM is to use the structural risk minimization theory in the feature space to find an optimal hyperplane. This hyperplane can correctly classify all points containing data and has the largest spatial spacing [9]. As a two-classification model, SVM solves the problems of falling into local extremum, dimension disaster and overfitting of traditional machine learning algorithms by solving the optimization of convex quadratic programming [10]. SVM has a strong generalization ability and learning ability as a machine learning algorithm based on Vapnik-Chervonenkis (VC) dimension theory and risk structure minimization [11].

A training sample set 

\[ T = \{(x_1, y_1), (x_2, y_2), \ldots, (x_N, y_N)\} \in (\mathbb{R}^n \times \mathbb{R}) \]

on the feature space is given, where \( x_i \in \mathbb{R}^n, y_i \in \{+1, -1\}, i = 1, 2, \ldots, N \). Under the condition of linear separability, the function of optimal hyperplane for classification is expressed as:

\[ f(x) = w^T \cdot x_i + b \] (1)

\( w = (w_1, w_2, \ldots, w_n) \) is the normal vector of the plane. \( b \) is the coefficient. \( i \) and \( -i \) respectively represent two types of samples. When \( f(x) > 0 \), \( x \) belongs to the sample point of \( i \). When \( f(x) < 0 \), \( x \) belongs to the sample point of \( -i \), as shown in Figure 1.

When a new sample point appears, it is substituted into equation (1). However, the hyperplane that can correctly classify the training set is not unique. Hence, a hyperplane with the largest distance from the sample points needs to be found and determined, which means the optimal hyperplane.

The distance \( \gamma \) from the point in the dataset to the hyperplane is:

\[ \gamma = \frac{y_i(w^T \cdot x_i + b)}{\|w\|} \] (2)

The optimization problem of \( w \) and \( b \) is the maximum distance from the point in the dataset to the hyperplane. The farther the distance between the points in the dataset and the hyperplane is, the more accurate
the classification results are. The required maximum distance is \( \text{max}\, \tilde{y} \). It is assumed that \( \tilde{y} = 1 \) and objective function \( \text{max}\, \tilde{y} \) is:

\[
\begin{aligned}
& \text{max} \frac{1}{\|w\|} \\
\text{s.t.} \; & y_i(w^T x + b) \geq 1, i = 1, 2, \ldots n
\end{aligned}
\] (3)

The maximum distance obtained is \( \text{max} \frac{1}{\|w\|} \), and the minimum \( \frac{1}{2} \| w \|^2 \) needs to be solved first. The problem of solving the maximum distance can be converted into the optimization solution of quadratic programming, as shown in equation (4):

\[
\begin{aligned}
& \text{min} \left( \frac{1}{2} \| w \|^2 \right) \\
\text{s.t.} \; & y_i((w^T x) + b) \geq 1, i = 1, \ldots, l
\end{aligned}
\] (4)

When solving each linear programming problem, the dual linear programming needs to be solved again. The solving process is complicated and it can be simplified by Lagrangian duality, as shown in equation (5):

\[
\begin{aligned}
& \text{max} \; \sum_{i=1}^{n} a_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} a_i a_j y_i y_j x_i^T x_j \\
\text{s.t.} \; & \sum_{i=1}^{n} a_i y_i = 0, a_i \geq 0, i = 1, 2, \ldots, n
\end{aligned}
\] (5)

By finding the optimal solution \( a^* \) of \( a_i \), the optimal solutions \( w^* \) and \( b^* \) of \( w \) and \( b \) are obtained. The equation for solving \( w^* \) and \( b^* \) is as follows:

\[
\begin{aligned}
& w^* = \sum_{i=1}^{n} a_i^* y_i x_i \\
& b^* = y_i - \sum_{i=1}^{n} a_i y_i (x_i \cdot x_j)
\end{aligned}
\] (6)

The optimal hyperplane is \( w^* x + b^* = 0 \) and the decision function is \( f(x) = \text{sign}(w^* x + b^*) \).

In order to change the hard interval between the data points and the optimal plane into a soft interval under the condition that the linearity is inseparable, the relaxation variable and penalty factor caused by data loss are introduced. The equation for constraining a small number of misclassification samples is:

\[
\begin{aligned}
& \text{min} \left( \frac{1}{2} \| w \|^2 \right) + C \sum_{i=1}^{l} \xi_i \\
\text{s.t.} \; & y_i((w \cdot x) + b) \geq 1 - \xi_i, \xi_i \geq 0, i = 1, \ldots, l
\end{aligned}
\] (7)

Nonlinear separability is that there is no optimal hyperplane in the sample space to classify datasets. In the case of nonlinear separability, the non-separable low-dimensional sample space can be transformed into separable high-dimensional sample space \( \Phi(x) \) through nonlinear mapping \( \Phi: \mathbb{R}^n \rightarrow H \) (\( H \) represents high-dimensional feature space). The mapping equation is:

\[
x \rightarrow \Phi(x) = (\Phi_1(x), \Phi_2(x), \ldots, \Phi_l(x))^T
\] (8)

The optimal classification function is:

\[
f(x) = \text{sign} \left( \sum_{i=1}^{l} a_i y_i \Phi(x_i) \cdot \Phi(x) + b \right)
\] (9)

In the mapping process, there will be the problem that the complex mapping function leads to a difficult calculation. The solution is to establish a kernel function and find a function with the same result of vector inner product mapping from the low dimensional space to the high dimensional space [12]. There are four common kernel functions:

1. Linear kernel function of linearly separable dataset:

\[
k(x, y) = x^T y
\] (10)

2. Polynomial kernel function mapping from low dimension to high dimension:
\[ k(x, y) = (x^Ty + 1)^d, d = 1, 2 \ldots \] (11)

Radial basis kernel function of multidimensional mapping:
\[ k(x, y) = \exp (-\gamma \| x - y \|^2) \] (12)

Sigmoid kernel function:
\[ k(x, y) = \text{Sigmoid} \left( b(x^Ty) - c \right) \] (13)

SVM based on Spark ML uses the Hinge loss function to calculate the maximum distance. The Hinge loss function equation is:
\[ L(w; x, y) = \max\{0, 1 - yw^Tx\} \] (14)

Regularization training of SVM can simplify the model and prevent overfitting. Hence, the setting of the regularization coefficient is significant for model training. The samples in the dataset are adopted to construct a classification hyperplane in the high-dimensional space and construct a model. The model can be adopted to predict the new data \( x \). When \( w^Tx > 0 \), the result is normal. Otherwise, the result is abnormal.

![Schematic diagram of classification function value](image)

2.2. SVM based on SA algorithm

SA algorithm simulates the cooling process of solid after heating [13]. At high temperatures, the internal energy of the solid increases continuously due to the violent irregular movement of the particles in the solid. In the cooling process, the temperature gradually cools down, the irregular movement of the particles in the interior slows down, and the internal energy gradually decreases. When the temperature is sufficiently low, the particles in the solid no longer move irregularly, and the internal energy also reaches the minimum [14]. SA is mainly adopted to solve the optimal solution of complex combinatorial problems. The objective function is adopted as the internal energy. During the cooling process, the particles begin to slow down from the violent irregular motion state and finally reach the lowest internal energy, that is, the optimal solution is obtained. In the process of SA search, random perturbation is conducted in combination with the probability jump characteristic, and the optimal global solution of the objective function can be found randomly based on accepting the inferior solution to a certain extent [15].
However, the SA algorithm has some shortcomings in its practical application. On the one hand, in order to solve the optimal solution of a complex function, a high initial temperature is required. The cooling speed of SA is slow, resulting in a slow solution and massive solutions. On the other hand, SA will have jump characteristics in the solving process, and it is easy to lose the optimal solution after multiple jumps [16]. Thereby, SA needs to be improved to avoid the above problems.

The improvement steps of SA are as follows:

1. The cooling function of SA is changed to a piecewise function. When the temperature of the objective function is high, the cooling can be conducted quickly. However, the cooling speed of the objective function needs to be slowed down when the temperature of the objective function is low. The specific expression is as follows:

   \[
   T(t) = \begin{cases} 
   T_0 \ln(k), & k < m \\
   T_0 \cdot 0.99^k, & k \geq m
   \end{cases}
   \]  

   where \( T_0 \) is the starting temperature, \( t \) is the number of iterations, and \( m \) represents a preset value.

2. The temperature when the iteration number of the SA algorithm is changed from constant to the objective function will not change for a while. When the objective function does not change at the set time, the calculation is stopped and the result is directly output.

3. When the temperature of the objective function is low, it is essential to start recording the optimal local solution obtained by SA to avoid missing the optimal solution due to the jumping characteristics of SA.

Figure 2 is the solution process of the improved SA algorithm:

![Figure 2 Flow chart of improved SA algorithm](image)

Due to the good performance in solving the optimal solution, the SA algorithm can be adopted to solve the parameters of SVM and solve the optimization of the SVM model.

2.3. Real-time monitoring of network anomalies based on big data

Figure 3 displays the requirements for real-time monitoring of network abnormalities:
Improving the accuracy of abnormal network behavior monitoring is the most crucial thing for a network real-time monitoring system [17]. Next, it is essential to shorten the data collection and processing time as much as possible, improve the monitoring speed, and reduce the loss caused by abnormal network behavior. Moreover, the network abnormal behavior monitoring system cannot affect the operation of the system and the network when dealing with the network fault [18]. Finally, the system should support extended Windows and Linux systems. The monitoring of abnormal network behavior can be completed by adding nodes.

Big data technology can effectively improve the efficiency of real-time monitoring of abnormal network behavior. Figure 4 is a network abnormal behavior monitoring system framework based on big data technology:

![System frame diagram](attachment:image.png)
In Figure 4, Jnetpcap is adopted to collect network data and extract features. Flume is adopted to summarize and filter data streams and send data to Kafka. Spark streaming will read the data stream in Kafka in real time, pre-operate the data, and monitor the data stream with the abnormal network monitoring model. The database My Structured Query Language (MySQL) is taken to store the monitoring results.

Network abnormal behavior monitoring includes offline models and real-time monitoring. Figure 5 is the overall process:

![Flowchart](image)

Figure 5 Real-time monitoring process of network abnormality

The construction process of the offline monitoring model has three stages: data preprocessing, model training and evaluation. The data preprocessing process is the vectorization and numerical processing of dataset features. In the training phase, the improved SA algorithm and SVM are adopted to divide the dataset into two categories to obtain the monitoring model. The evaluation is based on the feedback of the model training results, adjusting the parameters and retraining until the requirements are met. The real-time data flow monitoring process includes data collection, summary, filtering, and storage in the database after model monitoring.

2.4 Data acquisition module

Any monitoring of abnormal network behavior needs data collection. Stability and sustainability are the primary objectives of data collection work [19]. The data acquisition modules used are the Jnetpcap sub-module and Flume sub-module. The sub-module of Jnetpcap is responsible for network data collection and feature extraction. The Flume sub-module summarizes the extracted data and sends them to Kafka to facilitate subsequent monitoring. The collection of network data is the basis of network abnormal behavior monitoring [20].

2.5 Monitoring module of network abnormality

After the data collection process is completed, the network monitoring module reads the data stream stored in Kafka, and uses the monitoring model for real-time monitoring. Spark Streaming splits the data and
submits them to the task scheduler to facilitate the processing of the Spark kernel. The scheduler sends the data stream to the corresponding computer node for calculation and obtains the classification result. Figure 6 is a flowchart of the monitoring module:

![Flowchart of the network monitoring module](image)

**Figure 6** Flow chart of the network monitoring module

1. **Initialization.** SparkConf is adopted to set the operation, and the parameter is set to local. sparkContext is created to connect Spark clusters and to schedule computing tasks. Then, the task is assigned to the relevant nodes for calculation. StreamingContext is the only data processing channel. Subsequent data streams are processed in Streaming Context. The batch processing interval of the initialized Streaming Context is set to 6s. One Resilient Distributed Dataset (RDD) data is segmented for operation every batch processing interval.

2. **Creating DStream.** DStream of the third-party tool KafkaUtils is adopted to define the data source and realize Kafka to read the data stream. When a data stream is read in Kafka's topic, the interactive interface KafkaUtils converts the data stream into a standard DStream stream, ensuring that the system receives and processes at the same speed without causing memory overflow.

3. **Data preprocessing.** The characteristic format and number of data greatly impact the monitoring results, so the data should be preprocessed. First, the feature format read by Kafka is converted into DataFrame. Next, through the non-continuous dimension feature of index data, the data feature and label are numerically processed. Finally, the data are vectorized. The preprocessed data conform to the rationality of network abnormal behavior monitoring.

4. **Network abnormal behavior monitoring model.** In the Spark Streaming framework, SA-based SVM (SA_SVM) is adopted to monitor the preprocessed data stream.

5. **Storing the results.** The monitoring results are finally stored in the database. Specifying the database reuse connection’s data pool can ensure that the results are completely transmitted to the specific database. The database reuse connection can reduce the loss caused by the database connection and release. The data content output from the database includes time, data characteristics and categories.
2.6 Database storage module

Storing the dataset of distributed computing monitoring results in MySQL database has the following advantages. (1) MySQL, as a relational database, can use lightweight open source code. (2) It is convenient for users to import files into the database [21]. (3) It can provide good SQL query management tools to facilitate database operation and management [22]. (4) With high security, the database has a data backup and recovery function, ensuring data storage security [23].

The data storage module can realize timely and safe storage of monitoring results in the MySQL database. In order to avoid data duplication and facilitate the visual display of subsequent data, the database table can be inserted through the database connection to complete the monitoring of abnormal network behavior and the warehousing of results. This phase needs to create a table `flow_quantity` and table `flow_time` to record time, data characteristics and categories.

Figure 7 is a common operator for processing elements in RDD in Spark:

![Common operators for RDD](image)

Figure 7 Common operators

The map implementation function traverses each element and can return a new RDD. The foreach implementation function traverses each element, but does not return a value. mapPartitions traverses each partition in operation RDD with a return value. foreachPartition implements traversal operation without a return value. mapPartitions is selected because it is efficient and has a return value, which is convenient for subsequent visual operations. mapPartitions only need to connect to each partition during query and other operations, reducing the database's connection and ensuring the performance of database read and write operations [24].

2.7 Visual display module
Figure 8 is a visual display process of data:

The visual display of data can facilitate the user to intuitively and quickly understand the information to be expressed by the huge data, and facilitate the user's analysis and memory. Network abnormal behavior monitoring is to give users an intuitive response to network traffic changes and abnormal behavior analysis and processing results, so it must give users a good interactive experience. The monitoring results stored in the database are the support of the front-end display. In the front-end visualization window, users can also view the changes of the results in the database at any time. The visual display module of data can use the web server. Web can be adopted under Windows, Linux, Solaris and other operating systems [25].

2.8 System test environment and evaluation index

Three virtual machines are adopted for testing, and each virtual machine has a Spark cluster. Spark cluster includes one master node and two slave nodes. Table 1 presents the cluster configuration environment. The system uses CentOS6.5, with 20G hard disk, 4GB memory and a 2-core CPU.

<table>
<thead>
<tr>
<th>Component name</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>System</td>
<td>CentOS6.5</td>
</tr>
<tr>
<td>Hard disk</td>
<td>50G</td>
</tr>
<tr>
<td>Memory module</td>
<td>48GB</td>
</tr>
<tr>
<td>CPU</td>
<td>4 core</td>
</tr>
<tr>
<td>Spark</td>
<td>spark-2.3.0-bin-hadoop2.6</td>
</tr>
<tr>
<td>Flume</td>
<td>flume-ng-1.6.0-cdh5.10.0</td>
</tr>
<tr>
<td>Kafka</td>
<td>kafka 2.11-0.10.2.2</td>
</tr>
<tr>
<td>MySQL</td>
<td>5.1.73</td>
</tr>
</tbody>
</table>
The intrusion network test dataset is a dataset with a record number. Compared with other datasets, the intrusion network test dataset is configured reasonably, avoiding the problem of multiple redundant records caused by massive classifier categories. The experiment can extract data from the whole dataset, saving a lot of time and improving the experiment's accuracy. The intrusion network test dataset also provides a comparison benchmark for different models to facilitate users to evaluate and analyze the datasets and models. The intrusion network test dataset is used as the dataset of abnormal network behaviour’s monitoring performance evaluation. There are three data subsets in the intrusion network test dataset. The data in each subset includes five types: normal, Denial of Service (DOS), port monitoring or scanning (Probe), unauthorized local super user privileged access, User-to-Root (U2R) and unauthorized access from the remote host, Remote to Local (R2L).

A good network abnormal behavior monitoring model has high monitoring accuracy and low false alarm rate, so this exploration uses the accuracy and false alarm rate to evaluate the effect of the network abnormal behavior monitoring model.

3. Result

3.1 Algorithm comparison

SVM, Random Forest (RF) algorithm, Decision Tree (DT) and SA_SVM are introduced into this experiment. Two test sets based on the intrusion network test dataset are adopted to verify the monitoring effect of different algorithms on the test set Train+. Figure 9 is the result:

![Monitoring effect of different algorithm models](image)

Figure 9 Monitoring effect of different algorithm models

Figure 9 proves that the accuracy of the SA_SVM algorithm for Train+ monitoring is 96.4%. Compared with the RF algorithm and the traditional SVM algorithm, the false positive rate is 2.23%, slightly lower than the DT algorithm. In a word, the algorithm proposed has the best monitoring effect.

Figure 10 displays the monitoring effects of different algorithms on different types of attacks on Train+:
Figure 10 Different types of attack monitoring

Figure 10 displays that the monitoring accuracy of the proposed SA_SVM algorithm for Probe, DOS, U2R and R2L in Train+ is 89.32%, 96.47%, 50.25% and 92.83%, respectively. The low accuracy of the SVM algorithm for R2L and U2R monitoring is due to the small proportion of R2L and U2R samples in the training set, resulting in insufficient training of the model. The results of comparative experiments show that the proposed SA_SVM algorithm has high monitoring accuracy and can effectively detect unknown network abnormal behavior.

3.2 Analysis of real-time monitoring results of network anomalies

Jnetpcap is adopted to collect network data, extract data features, and send the data to Kafka. Spark streaming will read the data stream in Kafka in real time and perform pre-operation on the data. SA_SVM model is adopted to monitor data flow. MySQL database is employed to store the monitoring results and upload them to the web. Figure 11 presents the results:
Figure 11 Real-time monitoring of network abnormality (a) network traffic statistics within 6 hours; (b) : network abnormal behavior monitoring within 6 hours

Figure 11(a) reveals that the network traffic increases at the rate of 2000 per hour within 6 hours, and the network traffic has reached 16000 after 6 hours. Figure 11(b) suggests that as the network traffic increases, the number of attacks on the network also increases. The test results show that all functions can operate normally from Jnetpcap data collection, Kafka data processing, Spark Streaming network real-time monitoring and data storage. The network abnormal behavior monitoring system based on big data technology and cloud computing can adapt well to the data environment.

4. Conclusion

Combined with big data technology and cloud computing technology, this exploration studies the application research of computer big data analysis and cloud computing network technology in the direction of network security. First, the data collection technology of abnormal network behavior is introduced, and the Flume data collection component and Kafka distributed technology are discussed. Next, the data processing process of abnormal network behavior and the corresponding algorithm processing are studied, including SVM and SA algorithms. The improved SA algorithm is combined with SVM computing. Finally, the network abnormal behavior monitoring model based on big data is constructed, including a data collection module, network abnormal behavior monitoring module, data storage module and visual display module, and experiments are designed to verify it. The experimental results show that the monitoring accuracy of the SA_SVM algorithm model on Train+ is 96.4%, the false positive rate is 2.23%, and the recognition accuracy of four different network attacks is 89.32%, 96.47%, 50.25%, and 92.83%. In the experimental process, the comprehensive performance is the best. The online network abnormal behavior monitoring system can also operate normally and adapt well to the data environment. It can better complete the parallel collection, feature extraction, real-time monitoring and result storage of massive data. It has certain reference significance for the construction of network security engineering. Monitoring abnormal network behaviors is only the beginning of network security engineering. In the future, the network abnormal monitoring model needs to be further improved to increase the tracing and processing of abnormal network sources based on big data and cloud computing technology, and further analyze and study the network security-related algorithms.
Declarations

Ethical Approval
Not Applicable.

Authors' contributions
YL led the overall study, contributed to the data collection and interpretation, and wrote the manuscript. JZ and CZ contributed to the data collection, data analysis, and manuscript edits.

Funding
This work was funded by Science and Technology Project of Hebei Education Department in 2021, Research on the Prediction Model of Time-delay Infectious Diseases based on Big Data Technology (Project No. 2021237), and funded by Science and Technology Project of Hebei Education Department in 2019, Research on Ecological Monitoring and Restoration of the Grand Canal Based on Big Data Technology-Taking Cangzhou Section as an Example (Project No. QN2019191), and funded by Science and Technology Project of Hebei Education Department in 2022, Research on the Intelligent Early Screening System for Lung Cancer Based on Target Detection (Project No. ZC2022018).

Availability of data and materials
The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest
The authors declare no conflicts of interest.

Acknowledgements

Reference


