ABSTRACT
In order to improve the effect of precision intelligent agricultural decision support system, this paper combines big data technology to carry out precision mining of agricultural data, and combines decision tree algorithm to carry out data classification processing. Moreover, this paper obtains the most effective agricultural decision reference data through data mining, combines the agricultural decision support process to set the functional modules of the decision system, and analyzes the implementation process of each functional module. In addition, this paper studies the theoretical basis and key technologies of the agricultural production structure optimization decision support system, and builds a precision and intelligent agricultural decision support system based on big data analysis. The system mainly performs accurate processing of agricultural data and makes effective predictions, and then makes scientific decision results. Finally, this paper verifies the structure of the model in this paper combined with experimental analysis. From the comparison of experiments, it can be seen that the precision and intelligent agricultural decision support system constructed in this paper has significant effects.

Introduction
The agricultural production management system based on the Internet of Things is a very large system, and each part is worthy of in-depth study and practice. The decision system in the application layer of the agricultural Internet of Things provides decision support for production management to facilitate agricultural production through the processing, analysis, reasoning and decision of data information. Moreover, the level of crop yields will seriously affect the food security system and is an important part of the entire national security system.

Behind the explosive growth of data, there are many more important information hidden. Due to the lack of useful tools, the collected data has far exceeded human processing capabilities. As a result, many collected data become ‘data tomb’ in large databases and are rarely accessed (Aliev et al. 2018). The application of new technologies in agricultural informatization needs to be solved urgently. Its main connotation is the informatization of agricultural production management, agricultural science and technology informatization, agricultural operation and management informatization, rural market circulation informatization, and farmers’ living consumption informatization (Alipio et al. 2019). Therefore, the use of new technologies to replace data with knowledge is a challenge faced by large-scale database information systems. Data mining has brought database technology into a more advanced stage. It can not only query and traverse past data, but also find out the potential connections between past data, thereby promoting the transmission of information, and upgrading people’s application of data from low-level simple query to mining knowledge from data and providing decision support (Aryal et al. 2020). In addition, agricultural production is a complex and open system that requires the integration of knowledge from many aspects. Organising agricultural experts, informatics, economists and other multidisciplinary experts to conduct comprehensive research and development are of great significance to the improvement of China’s agricultural production level (Chae and Cho 2016).

In recent years, intelligent IoT farms have received extensive attention in automated and informatized farm management methods. With the continuous expansion of the farm area, the production environment has become more and more complex, and the traditional wired deployment control methods and
manual management methods have been unable to meet the work needs of the farm production managers. However, the rise of the Internet of Things technology and the continuous development of embedded monitoring systems provide solutions for the management of smart farms (Chandra et al. 2016).

Farmers can use big data to get detailed information on seasonal rainfall, water cycles, fertiliser needs, and much more. This allows people to make informed judgments about which commodities to sow for maximising profits and when to harvest. Farm yields are improved when the appropriate selections are made (Balamurugan et al. 2020).

This paper combines big data technology to construct the precision intelligent agricultural decision support system, and combines experimental research to evaluate the performance of the decision system, so as to improve the reliability of the precision intelligent agricultural decision support system.

Related work
The application of information technology in agriculture has been explored worldwide. Moreover, some developed countries have achieved rich results in agricultural resource monitoring and application, agricultural environmental monitoring, agricultural production management, etc. and they have begun to be applied to actual production (Channe et al. 2015). The United States and other developed countries use its advanced information technology to build an agricultural environment monitoring system with superior performance, and use advanced sensor technology, information collection and transmission technology and computer network technology to establish a nationwide agricultural information system to realise automatic monitoring of the agricultural environment (Clapp et al. 2018). At the same time, the United States has established a three-tier agricultural environmental monitoring system structure for agricultural environmental information collection, information communication and processing, and information release. France uses satellite technology to forecast climate, weather and other information, and to forecast diseases and pests (Elijah et al. 2018). China has built an agricultural production measurement and control network connecting key provinces and cities across the country, and completed real-time monitoring of agricultural production information. The agricultural production measurement and control network based on the Internet of Things has been promoted and applied in many key provinces and cities (Faling et al. 2018).

In Anthony A. Kimaro et al. (2016), the EPIC (Environmental Policy Integrated Climate) crop growth model was used to predict the wheat yield at multiple sites in northern China. Junyong Liu et al. (2018) summarised the principles, technical routes, and research progress of remote sensing technology in crop growth prediction, planting area measurement and yield prediction, and analyzed possible problems in the actual application of remote sensing yield estimation. Peter Newell and Olivia Taylor (2018) established a linear regression model to predict the growth and yield of winter wheat. The research results show that the vegetation index EVI can establish a linear regression equation with yield better than NDVI. Rameshaiah et al. (2015) used the time series method to construct the ARIMA (2,1, (3)) model. The research results show that the established model has a high accuracy. Partha Pratim Ray (2017) selected the number of agricultural fertilisers used, planting area, total mechanical labour, and institutional change factors to construct a food production model through econometric methods such as stepwise regression and weighted least squares regression. The experimental results show that the amount of land and fertiliser used are the main influencing factors of grain yield. Roopaei et al. (2017) established a multiple regression model and a BP (Error Back Propagation) artificial neural network prediction model for food production prediction, and compared the output results of the model. The research results show that the BP artificial neural network can better fit the nonlinear prediction problem in complex situations, and has higher prediction accuracy than the traditional multiple regression model. Laura Scherer and Peter H. Verburg (2017) describes the research and application of artificial neural networks in crop yield prediction and pest prevention.

Ernest C. Shea (2014) assigned reasonable weights to exponential smoothing models, multiple regression models, and C-D production function models, and weights them to build a combination forecasting model for food production. The experimental results prove that the combined model can take advantage of each model to improve the prediction accuracy. Kerri L. Steenwerth et al. (2014) combined grey theory and Markov model and applied it to the prediction of corn yield. The experimental results show that the designed output prediction model can meet the requirements of prediction accuracy. Fanen Terdoo and Olalekan Adekola (2014) proposed a genetic BP algorithm with improved adaptive crossover and mutation operators, and established a sugarcane yield prediction model. The result shows that the effect of the improved genetic BP algorithm is greatly improved compared with the traditional method. Amod K. Thakur and Norman T. Uphoff (2017) proposed a RBF (Radial Basis Function) artificial neural network food production
prediction model based on principal component analysis. The model eliminates redundant information among influencing factors through principal component analysis, reduces the dimensionality of the input variables of the neural network, simplifies the structure of the artificial neural network, and improves the speed and accuracy of food production forecasting. Jonathan Verschuuren (2018) established a discrete process neural network crop yield prediction model, and used simulated annealing-genetic algorithm to revise the neural network weights in layers. The results of the experiment show that this model has a strong generalisation ability and can be applied to crop yield prediction in different regions. Robert B. Zougmore et al. (2018) established a grain yield prediction model based on rough set theory and BP artificial neural network. The experimental results show that the model can significantly improve the accuracy of output prediction.

Application of decision tree classification algorithm in agricultural decision data

The decision tree classification method is the process of classifying it according to the attributes of the training set. Non-linear sets of data are managed effectively with decision trees. Many fields, including construction, civil works, law, and commerce, employ the tree-based tool in application. There are two relating to decision networks such as categorical factor and continuously variable decision trees. The training period for a decision tree is typically longer. Because of the intricacy and time required, decision tree learning is excessively expensive. When it comes to using regression and forecasting continuous variables, the Decision Tree method falls short. For a built tree, its top is the root node, and other nodes are determined by testing and selecting attributes. As shown in Figure 1, it is a decision tree, which represents the ‘prediction to buy a computer’, that is,

Figure 1. Typical decision tree.
the likelihood of a customer in a store buying a computer. Among them, rectangles represent internal nodes, and ellipses represent leaves.

The decision tree classification algorithm can be summarised as follows:

Step 1: The algorithm selects the 'most important attribute' from the training set $D$ as the root, then divides the training set according to the attribute value, and selects a node that can represent the training set data to start constructing the decision tree.

Step 2: If all data objects belong to one class, then the algorithm uses the class label of this class to define the nodes, which are all leaf nodes at this time. If the data objects do not belong to the same class, the algorithm needs to measure the attributes according to a certain strategy (for example, information entropy), and select an attribute $X$ as the attribute to be tested, which is the test node $X$. If the feature of $X$ is assumed to be $V_1, V_2, \cdots, V_n$, then according to the different features of $X$, $D$ can be divided into $n$ subsets $D_1, D_2, \cdots, D_n$, and attribute $X$ can be deleted from the original attribute set.

Step 3: The algorithm repeats the above steps until it generates a decision tree that can properly classify the training set.

The algorithm of the decision tree summarised from the training set is as follows:

Algorithm: The algorithm for establishing a decision tree for a given training set $D$.

Input: Data set $D$ and a collection of attribute class labels of samples in $D$; a collection of candidate attributes;

Output: Decision tree method corresponding to the sample:

1. The algorithm creates node $M$;
2. If the objects in the training set $D$ belong to the same class $C$ at the same time, the algorithm turns the node $M$ into a leaf node and classifies it with class $C$;
3. If the set of candidate attributes is empty, the algorithm returns the node $M$ as a leaf node and marks it as the most common class in the training set $D$. Otherwise, the algorithm selects the attribute with the highest information gain in the set of candidate attributes and uses it to mark the node $M$;
4. A highest information gain attribute is generated by node $M$ as $s_i$;
5. $s_i$ is a subset of the training set $D$, and all elements in the subset satisfy the branch $a_i$;
6. If $s_i$ is empty, the algorithm adds a leaf node to node $M$ and marks it as the class label that accounts for the majority in the training set $D$. Otherwise, the algorithm adds a node $N$ generated by the strategy function.

The pivotal point of the decision tree classification algorithm is how to find the optimal attributes to divide, continuously form high-purity branch nodes and leaf nodes, and finally find a decision tree that can reasonably classify the training set.

The attribute selection metric is a split strategy. One of the algorithms is to use the information gain metric on each node of the tree to select attributes, which is mainly used to determine how to allocate objects on each node. Usually, we target those attributes with maximum information gain. It can minimise the amount of information generated by classifying sample objects. Through the information gain measurement, we can improve the accuracy and efficiency as much as possible in the classification process.

In the ID3 algorithm, information gain is used to determine the choice of attributes at each step. The basic operation process is as follows. We assume that the training set $D$ is a collection of arbitrary sample objects, which contains $m$ objects with different class label attribute values, and assume $C_i$ ($i = 1, 2, \cdots, m$) is $m$ different classes. At the same time, we assume that $C_{iD}$ is the collection of objects of class $C_i$ in data set $D$, $|D|$ g is the number of data objects in data set $D$, and $|C_{iD}|$ is the number of data objects in $C_{iD}$.

All tuples in the training set $D$ are classified, and entropy is used to reflect the average amount of information required for classification. The entropy calculation formula of set $D$ is:

$$\text{Info}(D) = - \sum_{i=1}^{m} p_i \log_2(p_i)$$  \hspace{1cm} (1)

Among them,

$$p_i = \frac{|C_{iD}|}{|D|}$$  \hspace{1cm} (2)

The training set $D$ is classified according to the $n$ characteristics of the attribute $A$, that is, $A$ contains $n$ different objects $\{a_1, a_2, \cdots, a_n\}$, and $D$ is divided into $n$ partitions $\{D_1, D_2, \cdots, D_n\}$. The information required for classification can be obtained by weighted
The number of samples belonging to class $P$ needed to partition the training set $D$ attribute to be partitioned, which can minimise the information gain is:

$$\text{Gain}(A) = \text{Info}(D) - \text{Info}_A(D)$$

The information gain measures the information needed to partition the training set $D$ according to $A$. When Gain($A$) is the largest, the corresponding $A$ is the attribute to be partitioned, which can minimise the amount of information needed to classify the training set $D$.

The conditions for the decision tree classification algorithm to terminate recursion are as follows:

1. At a certain node, the categories of all sample tuples under it are the same;
2. There is no way to divide the remaining attributes of the sample. Under such conditions, the majority voting method can be used to divide the attribute classes. Turn a given node into a leaf node, find the attribute class label that appears most in the training set, and mark the leaf node with this label;
3. If there is no data under the given branch, same as the second point above, use the majority voting method to find the class label of the attribute that appears most in the sample, and use this class label to create and mark the leaf node.

The principle of the naive Bayes algorithm is to assume that the existence of specific features has nothing to do with the existence of other features, which is the independence of class conditions. It supports the assumption of independence between different features using Bayes’ theorem. The Bayesian classifier is dependent on related events, so the probability of an event in the future can be found from the events that occurred before. It makes full use of all the attributes contained in the data, each attribute is equally important, independent of each other, and each attribute is analyzed separately. The basis of the naive Bayes algorithm is the Bayesian conditional probability theorem, which uses the prior probability of an event to calculate the posterior probability. Naive Bayes classifier is a probabilistic method based on Bayes’ Hypothesis and the premise of prediction independent. A Naive Bayes classifier, assumes that the existence of one information in a category is independent of the presence about any other value (Saravanan et al. 2015).

The interpretation of Bayes’ theorem is as follows: It is assumed that $X$ is the data object in the training set, and $X$ is described with $n$ attribute values, and it is assumed that $H$ represents the hypothesis that the data object $X$ belongs to a certain type of $C$. Then, $P(H|X)$ represents the posterior probability of the $H$ event under the premise $X$, that is, the probability that the $H$ event holds under the premise $X$. $P(H)$ is the prior probability, $P(H)$ and event $X$ are independent of each other.

Bayesian rules are as follows:

$$P(H|X) = \frac{P(H)P(X|H)}{P(X)}$$

Bayesian classification is a probabilistic method of training and reasoning that is founded on a distinct interpretation of what it takes to learn from the data, in which probabilities is employed to express ambiguity about just the connection getting learned. The basic idea of naive Bayesian classification can be summarised as: it is assumed that each element in the training set $D = \{D_1, D_2, \ldots, D_m\}$ with $m$ elements can be represented by a vector of $n$-dimensional attributes $X = \{X_1, X_2, \ldots, X_n\}$. Moreover, it is assumed that $A = \{A_1, A_2, \ldots, A_n\}$ represents $n$ attributes, and $X$ is the $n$ measurements of $n$ attributes on the data tuple.

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)}$$

Among them, $P(X)$ is a fixed value for all classes $C_i (i = 1, 2, \ldots, m)$, and $P(C_i)$ is usually considered equal probability, namely:

$$P(C_i) = \frac{\text{Number of samples belonging to class } C_i}{\text{Total number of training set samples } m}$$

Therefore, we only need to maximise $P(X|C_i)P(C_i)$ to get the highest posterior probability. However, because its calculation is very complicated and the amount of calculation is extremely large, the simple setting based on the independence of the class condition can be expressed by the following formula:

$$P(X|C_i) = \prod_{k=1}^{n} P(X_k|C_i) = P(X_1|C_i)P(X_2|C_i) \cdots P(X_n|C_i)$$

Among them, $\eta$ represents the value of attribute $\lambda$ corresponding to tuple $A_k$.

The Least Mean Square (LMS) (also called delta learning rule) is an adaptive learning algorithm. Adaptive learning is a technology-based or digital instructional system that analyses an agricultural productivity in real time and adjusts the techniques accordingly. It establishes a cost function $E(w)$ to describe the gap between the output response and the expected
response, which is continuously differentiable to the weight vector \( w \). It can be seen that the smaller the value of \( E \), the better. The mathematical expression is to find an optimal weight vector \( w^* \), which has the following inequality \( E(w^*) \leq E(w) \) for any \( w \). The weights of the network are adjusted iteratively.

Self-adaptation refers to the automatic adjustment of the processing method, processing sequence, processing parameters, boundary conditions or constraint conditions according to the data characteristics of the processed data in the process of processing and analysis to adapt it to the statistical distribution characteristics and structural characteristics of the processed data. Achieve the best treatment effect. To be precise, the LMS algorithm is an adaptive linear gradient descent algorithm, which is implemented by continuously correcting the initialised filter coefficients according to the minimum mean square error criterion. The

---

**Figure 2.** Basic structure diagram of agricultural decision support system.
weight coefficient of the ‘current moment’ is obtained by adding the proportional term of the negative mean square error gradient to the weight coefficient of the ‘last moment’, and iteratively updated, each time the improvement makes the cost function smaller, and so on. Until the cost function is small enough. In other words, in a multi-dimensional space, starting from a point, keep walking in the direction of decreasing cost function until it reaches the lowest point. There is an adjustment time before the system becomes stable. The gradient descent optimisation method can be applied to converge sharply to the local minimum of the cost function surface. This time is controlled by the convergence factor of the algorithm. Gradient Descent is an optimisation approach for locating a variational stored procedure minimum distance. Gradient descent starts with defining the starting model parameters, and then continuously adjusts the value systems using calculus to minimise the provided cost-function.

Through the adjustment of \( \eta \) in the LMS algorithm, within a certain value range, increasing \( \eta \) will reduce the adjustment time, but when it exceeds this value range, the system will no longer converge. When \( \eta \) is very small, the performance of LMS will become good. However, the rate of convergence to a local minimum is very slow.

The following formula defines the weight vector \( w \) by using LMS learning rules:

\[
w_{p+1} = w_p + \Delta w_p = w_p - \eta \nabla E(w_p)
\]

\[
= w_p - \eta \frac{\partial E}{\partial W}|_{p}
\]

The LMS algorithm is updated in real time. Whenever the data in the training set is trained, the weight vector \( w \) in the network must be revised. Therefore, in order to simplify the content, the subscript \( p \) will be skipped. In a calculation cycle, the training data pair \((x, d)\) is randomly selected from the training data, where \( x \) is the input and \( d \) is the expected response, which is brought into the activation function \( f(u) \). \( u \) can be expressed in matrix form.

\[
u = w^T x
\]

After bringing \( u \) into the activation function, the output of the neuron is obtained:

\[
o = f(u) = f(w^T x)
\]

In order to determine the size of the error of the weight \((x, d)\) of a specific mode \( n \), a direct comparison is made between the calculated expected output \( d \) and the output result \( o \) of the neuron. The error signal is the difference between the expected response and the output:

\[
w = d - o = d - (w^T x)
\]

The error \( e \) will be used to measure the weight of the neuron and adjust it to minimise the cost function \( E \) of the overall network weight.

As a cost function, the error sum of squares must be gradually reduced during the training phase. The continuous nonlinear differentiable cost function \( E \) of the weight vector \( w \) will be a quadric hypersurface geometrically. Moreover, it is a parabolic surface with a concave upwards in the middle, and it is a function with a unique minimum, so that the cost function is minimised, which is equivalent to descending along the parabolic surface to find the minimum. The gradient can be used to find the minimum value. The formula of \( E \) is as follows:

\[
E = \frac{1}{2} e^2 = E(w)
\]
When taking the partial derivative of the cost function $E$ for each element of the weight vector $w$, the result is the gradient $\nabla E(w)$, and the mathematical formula is as follows:

$$\nabla E(w) = \frac{\partial E}{\partial w}$$  \hspace{1cm} (14)

Among them, $\frac{\partial E}{\partial w}$ is called the error signal, which is used to measure the degree of error change when the input of $u$ changes. $\frac{\partial E}{\partial w}$ is used to measure the degree of influence on the weight vector $w$ when calculating a specific input $u$. By applying the chain rule to the above formula again, we can get:

$$\nabla E(w) = \frac{\partial E}{\partial w} = \frac{\partial e}{\partial w} + \frac{\partial e}{\partial u} \frac{\partial u}{\partial w}$$  \hspace{1cm} (15)

When $e$ is differentiated on both sides, there are:

$$\frac{\partial E}{\partial e} = e$$  \hspace{1cm} (16)

At the same time, when $u$ is differentiated by both sides at the same time, there are:

$$\frac{\partial e}{\partial u} = f'(u)$$  \hspace{1cm} (17)

At the same time, when $w$ is differentiated by both sides at the same time, there are:

$$\frac{\partial w}{\partial w} = x$$  \hspace{1cm} (19)

Therefore, the first-order partial derivative of the cost function $E$ with respect to the weight vector $w$ can be expressed as:

$$\nabla E(w) = -ef'(u)x$$  \hspace{1cm} (20)

Therefore, when $p$ is presented in the network, the LMS learning rule can be written as:

$$w_{p+1} = w_p + \Delta w_p = w_p - \eta \nabla E(w_p)$$

\hspace{1cm} $$= w_p - \eta e_p f'(u_p) x_p$$  \hspace{1cm} (21)

Since LMS is applied to the BP algorithm, it is best to specify the error signal term $\delta$ for OL (output layer) in the formula of the LMS algorithm. Backpropagation is a technique for swiftly calculating variables. It facilitates in the computation of a gradient descent with loads. The weights are adjusted reverse, from output to input, which gives the technique its meaning. The number of sequences which can be learned into a system with mean zero squared error can be defined as the LMS Capability. The synapse precepts are followed either by Algorithms. The orientation of synapse weight fluctuations, increases, is indicated by these precepts, but just not the pace of change. The network’s LMS capability will be equivalent to the number of assigning weights to every individual contribution. The error signal is given by the following formula:

$$d = ef'(u) = (d - o)f'(u)$$  \hspace{1cm} (22)

Therefore,

$$w_{p+1} = w_p + \Delta w_p = w_p + \eta \delta x_p$$  \hspace{1cm} (23)

For the mode $p$, it can be written in the form $w_{j,p}$ of each vector component of the weight vector $w_p$:

$$w_{p+1} = w_p + \eta (d_p - o_p)$$

$$f'(u)x_{j,p} = w_{j,p} + \eta \delta x_{j,p}$$  \hspace{1cm} (24)

For the linear activation function in the OL neuron that can be applied to the MLP model, the derivation of the activation function is equivalent to $f'(u) = 1$. Therefore, the error signal $\delta$ is equal to the error $e$:

$$d = ef'(u) = e = d - o$$  \hspace{1cm} (25)

Therefore, the LMS learning algorithm of linear neurons is given by:

$$w_{p+1} = w_p + \Delta w_p = w_p + \eta \delta x_p$$

\hspace{1cm} $$= w_p + \eta e_p x_p$$  \hspace{1cm} (26)

**Precision and intelligent agricultural decision support system**

The agricultural decision support system is an integrated system that aims to provide agricultural decision support and coordinate multiple functions. Data collection and analysis systems (DSS) are software-based platforms that gather and analyze data from a number of inputs. Their goal is to make decision-making easier for administration, operation, budgeting, and recommending the best solution path. It aids producers in the agriculture industry in solving complicated crop production problems. An DSS can not only present a list of possibilities for current actions, but it can also assist decision – making in achieving better results in future positions. The conceptual structure of DSS is mainly composed of six subsystems: conversation system, control system, problem handling system, database system, model library system and knowledge base system. The logical structure of a simple agricultural decision support
system only needs to include a database, a model library
and a knowledge base. Conversational Systems are in-
telligent machines that can comprehend language and
engage in a textual or vocal discussion with a consumer.
Besides, maintaining an increasing population, ensuring
a subsistence for farming, and preserving the environ-
ment are all solved via problem-solving functions. If
we are to make endless development in every one of
these, we should work collectively. However, setting a
standard, monitoring overall performance, and imple-
menting corrective measures in judgement are all
examples of managerial control. Moreover, the data-
bases provide for versatility in query creation by select-
ing columns from attribute values and measurements
from the table structure, allowing the responsible party
to select which economic data are helpful to analyses,
as well as the creation of various types of graphs.
Further, to detect growth advantages, challenges, and
emergencies, an experience and understanding man-
agement information systems (dss) that directs,
teaches, and offers rationale was deployed. The state's
focus is to enhance beginner users’ decisions to a
degree that is comparable to those of more knowledge-
able decision – making (Deepa et al. 2020).
The simple logical structure of DSS is shown in
Figure 2.
The agricultural decision support capability of the
agricultural decision support system does not directly
rely on the data in the database system. The data in the
database is passed to the model library as the input of
the model library. The model library uses these data to
model and calculate the calculation results to provide
DSS with agricultural decision-making. Support, it is at
the core of the agricultural decision support system.
The model library is a component that provides
decision-makers or managers with decision-making
analysis capabilities. It can analyze and answer questions
through reasoning and calculation, and then return the
results to the decision-maker. Models are a way for
people to study things based on their objectivity. The
model library is composed of different models con-
structed for different problems in the decision-making
process. Model library management is used to organise
and manage these decision-making models. The model
library is the core component of the model library
system. Therefore, it can be considered that the agricul-
tural decision support system is ‘model-driven’. Model
library management aids in the development of numerical,
subjective, and description forms for successful plan-
ning and decision-making by library administrators.
Nevertheless, the focus is mostly on numerical analyses,
which are made up of mathematical formulas that calcu-
late the workload that govern library activities. The

![Figure 4. Structure diagram of the optimisation model of agricultural production structure.](image)

![Figure 5. Data flow chart.](image)
library system’s primary goal is to gather, preserve, organise, extract, and make sources of information accessible to online consumers. The model library system is mainly responsible for the maintenance and management of the model. It is mainly composed of the model library that stores the model, the modelling system that is responsible for establishing the model, the model library that generates, modifies, maintains and updates the model, the maintenance system, and the model library management system. The model library system structure is shown in Figure 3.

The production structure planning model is to use the grey linear programming model to establish the optimal resource utilisation structure and production structure to fully tap the production potential. The linear programming methodology was employed to information compiled by a farmer to maximise farm earnings. In more precise terms, linear programming is the process for maximising a linear optimal solution that is constrained by linear conditions. Figure 4 is the structure diagram of the mathematical model of agricultural production structure.

Data flow refers to the flow of data in the system. The operations on the data in the system mainly include adding, deleting, modifying, and querying. Moreover, the model is the key to the agricultural decision support system, and the main flow of data is the mathematical model. According to the needs of users, the system queries related data from the database, and then users call the model to perform calculations to obtain supporting data. In order to forecast prediction performance, a data acquisition procedure that records meaningful information about the product condition should first be completed. The grey approach is used to anticipate future situations after obtaining meaningful data. Before the deploy of grey model, the collected

---

**Figure 4.** Data maintenance activity diagram.

---

**Figure 6.** Data maintenance activity diagram.
time series data must first be distorted. The model is built using training data, and the model’s performance is compared using testing data. The root-mean square error and linear correlation are used to measure the forecasting performance.

Figure 5 shows the data flow chart of the system. Recognising producers’ objectives as well as how the diversity of existing agricultural systems helps to such goals is the first step in any attempt to engage with farm owners. By automating manual collecting and contextual understanding activities while taking external elements into consideration, Administrator intends to help managerial chores and real-time decision building, and also compliance management. The owner of the farm Information triggers both internally and externally filtering, taking into consideration the operational activities that have been scheduled. An implementation strategy can be prepared and delivered to the various facets of life based on information.

There are four main functions of data maintenance, namely, adding, modifying and deleting data. For security reasons, data maintenance must be the authority of the system administrator to unify the management of data. The UML activity diagram of the data maintenance function is shown in Figure 6.

The comprehensive query is divided into three sub-modules: output query, material consumption query and population query. There are two ways to query output, one is vertical query according to time, and the other is horizontal query according to crops. Output Query described the output indexes about the particular annual yield and productivity of a specified
crops. The materials consumption query report is a document that is used in the manufacturing process to summarise the items and products used within a given accounting period. Population query estimated the number of individual items or group of crops that results in good or poor productivity. The UML activity diagram of the comprehensive query is shown in Figure 7.

The system performance verification

This paper constructs a precision and intelligent agricultural decision support system based on big data analysis. The system mainly performs accurate processing of agricultural data and makes effective predictions, and then makes scientific decision-making results. The system in this article needs to have strong agricultural data processing capabilities, and can make scientific decisions based on predictions. Finally, large-scale agricultural information analysis to assist business insights at speed and scale required network infrastructure expenditures, while big data processing requires improved parallel and distributed computing approaches. Therefore, when verifying system performance, this article mainly starts from two perspectives: systematic agricultural data mining and agricultural decision-making. This article obtains a large amount of data from the network, and constructs the system through simulation, and inputs the data collected by the network into the simulation system to count the test results. The results of the data mining experiment are shown in Table 1 and Figure 8.

From the above research results, it can be seen that the precision and intelligent agricultural decision support system based on big data analysis constructed in this paper has good agricultural data mining effects.
Figure 9. Decision effect of agricultural decision support system.

capabilities. On this basis, this paper verifies the decision effect of the system, and the results obtained are shown in Table 2 and Figure 9.

From the above research results, the precision and intelligent agricultural decision support system based on big data analysis constructed in this paper can play an important supporting role in agricultural decision-making.

Conclusion

In the era of big data, in order to improve the effect of agricultural decision-making and the effect of agricultural economic development, this article combines big data technology for agricultural data mining, obtains effective data from massive agricultural data, and optimises the theoretical basis and key technologies of the decision support system for agricultural production structure. We studied. Moreover, this article divides the mathematical models applied by the agricultural production structure optimisation decision support system into three categories according to their functions: prediction models, input-output analysis models, and production structure planning models, and introduces the corresponding mathematical models, and determines the research route for agricultural production structure optimisation. In addition, this article builds a precision and intelligent agricultural decision support system based on big data analysis. The system mainly uses accurate processing of agricultural data to make effective predictions, and then make scientific decision results. Finally, this paper designs an experiment to analyze the performance of the agricultural decision support system proposed in this paper. Through data analysis, we can see that the agricultural decision support system constructed in this article has good agricultural data processing effects and agricultural decision-making capabilities.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Notes on contributor

Qiao Jie (1982- ), female, born in Nanjing, Jiangsu Province, received her master's degree from North University of China in 2009, majoring in Measurement Technology and Instrument. Now she is a lecturer and engineer in Nanjing Institute of Information Technology. Now she is the director of the big data and computer application technology teaching and research section. Main research fields: server operating system, big data technology application. More than 10 papers published and 4 books published.

References

Aliev K, Pasero E, Jawaid MM, Narejo S, ulatov A. 2018. Internet of plants application for smart agriculture. Int J Adv Comput Sci Appl. 9(4):421–429.
Alipio MI, Cruz AEMD, Doria JDA, Fruto RMS. 2019. On the design of nutrient film technique hydroponics farm for smart agriculture. Eng Agric Environ Food. 12(3):315–324.
Aryal JP, Sapkota TB, Rahut DB, Jat ML. 2020. Agricultural sustainability under emerging climatic variability: the role of climate-smart agriculture and relevant policies in India. Int J Innov Sust Dev. 14(2):219–245.
Balamurugan S, Muthu BA, Peng SI, Wahab M HA. 2020. Call for special issue papers: big data analytics for agricultural disaster management. Big Data. 8(6):544–545.
Chae CJ, Cho HJ. 2016. Smart fusion agriculture based on Internet of Thing. J Korea Convergence Soc. 7(6):49–54.
Chandra A, McNamara KE, Dargusch P, Damen B, Rioux J, Dallinger J, Bacudo I. 2016. Resolving the UNFCCC divide on climate-smart agriculture. Carbon Manage. 7(5–6):295–299.
Channe H, Kothari S, Kadam D. 2015. Multidisciplinary model for smart agriculture using internet-of-things (IoT), sensors, cloud-computing, mobile-computing & big-data analysis. Int J Comput Technol Appl. 6(3):374–382.
Clapp J, Newell P, Brent ZW. 2018. The global political economy of climate change, agriculture and food systems. J Peasant Stud. 45(1):80–88.
Deepa N, Khan MZ, Prabadevi B, P.m. DR, Maddikunta PK, Gadekalu TR. 2020. Multiclass model for agriculture development using multivariate statistical method. IEEE Access. 8:183749–183758.
Elijah O, Rahman TA, Orikumhi I, Leow CY, Hindia MN. 2018. An overview of Internet of Things (IoT) and data analytics in agriculture: benefits and challenges. IEEE Internet Things J. 5(5):3758–3773.
Faling M, Biesbroek R, Karlsson-Vinkhuyzen S. 2018. The strategizing of policy entrepreneurs towards the global alliance for climate-smart agriculture. Global Policy. 9(3):408–419.
Kimaro AA, Mpanda M, Rioux J, Aynekulu E, Shaba S, Thiong’o M, Mutuo P, Abwanda S, Shepherd K, Neufeldt H. 2016. Is
conservation agriculture ‘climate-smart’ for maize farmers in the highlands of Tanzania? Nutr Cycling Agroecosyst. 105 (3):217–228.
Liu J, Chai Y, Xiang Y, Zhang X, Gou S, Liu Y. 2018. Clean energy consumption of power systems towards smart agriculture: roadmap, bottlenecks and technologies. CSEE J Power Energy Syst. 4(3):273–282.
Newell P, Taylor O. 2018. Contested landscapes: the global political economy of climate-smart agriculture. J Peasant Stud. 45(1):108–129.
Rameshaiah GN, Pallavi J, Shabnam S. 2015. Nano fertilizers and nano sensors—an attempt for developing smart agriculture. Int J Eng Res Gen Sci. 3(1):314–320.
Ray PP. 2017. Internet of things for smart agriculture: technologies, practices and future direction. J Ambient Intell Smart Environ. 9(4):395–420.
Roopaei M, Rad P, Choo KKR. 2017. Cloud of things in smart agriculture: intelligent irrigation monitoring by thermal imaging. IEEE Cloud Comput. 4(1):10–15.
Scherer L, Verburg PH. 2017. Mapping and linking supply-and demand-side measures in climate-smart agriculture. A review. Agron Sustainable Dev. 37(6):1–17.
Saravanan V, Pralhaddas KD, Kothari DP, Woungang I. 2015. An optimizing pipeline stall reduction algorithm for power and performance on multi-core CPUs. Hum Cent Comput Inf Sci. 5(2), 1–13.
Shea EC. 2014. Adaptive management: The cornerstone of climate-smart agriculture. J Soil Water Conserv. 69 (6):198A–199A.
Steenwerth KL, Hodson AK, Bloom AJ, Carter MR, Cattaneo A, Chartres CJ, Hatfield JL, Henry K, Hopmans JW, Horwath WR. 2014. Climate-smart agriculture global research agenda: scientific basis for action. Agric Food Sec. 3(1):1–39.
Terdoo F, Adekola O. 2014. Assessing the role of climate-smart agriculture in combating climate change, desertification and improving rural livelihood in Northern Nigeria. African J Agric Res. 9(15):1180–1191.
Thakur AK, Uphoff NT. 2017. How the system of rice intensification can contribute to climate-smart agriculture. Agron J. 109(4):1163–1182.
Verschuuren J. 2018. Towards an EU regulatory framework for climate-smart agriculture: the example of soil carbon sequestration. Transnational Environ Law. 7(2):301–322.
Zougmoré RB, Partey ST, Ouédraogo M, Torquebiau E, Campbell BM. 2018. Facing climate variability in sub-Saharan Africa: analysis of climate-smart agriculture opportunities to manage climate-related risks. Cahiers Agric (TSI). 27(3):1–9.