processes

MDPI

*Article*

# A Comparative Study of Oil–Water Two-Phase Flow Pattern Prediction Based on the GA-BP Neural Network and Random Forest Algorithm

**Yongtuo Sun [1,2], Haimin Guo [1,2,*], Haoxun Liang [1,2], Ao Li [1,2], Yiran Zhang [1,2] and Doujuan Zhang [1,2,3]**

[1] College of Geophysics and Petroleum Resources, Yangtze University, Wuhan 430100, China; 2022720524@yangtzeu.edu.cn (Y.S.); 2022710374@yangtzeu.edu.cn (H.L.); 2022730026@yangtzeu.edu.cn (A.L.); 2021720527@yangtzeu.edu.cn (Y.Z.); 2021730028@yangtzeu.edu.cn (D.Z.)
[2] Key Laboratory of Exploration Technologies for Oil and Gas Resources, Yangtze University, Ministry of Education, Wuhan 430100, China
[3] Research Institute of Exploration and Development, Sinopec Shengli Oilfield Company, Dongying 257000, China
* Correspondence: ghm@yangtzeu.edu.cn

**Abstract:** As global oil demand continues to increase, in recent years, countries have continued to expand the development of oil reserves, highlighting the importance of oil. In order to adapt to different strata distribution conditions, domestic drilling technology is becoming more and more perfect, resulting in a gradual increase in horizontal and inclined wells. Because of the influence of various downhole factors, the flow pattern in the wellbore will be more complex. Accurately identifying the flow pattern of multiphase flow under different well deviation conditions is very important to interpreting the production log output profile accurately. At the same time, in order to keep up with the footsteps of artificial intelligence, big data and artificial intelligence algorithms are applied to the oil industry. This paper uses the GA-BP neural network and random forest algorithm to conduct fluid flow pattern prediction research on the logging data of different water cuts at different inclinations and flow rates. It compares the predicted results with experimental fluid flow patterns. Finally, we can determine the feasibility of these two algorithms for predicting flow patterns. We use the multiphase flow simulation experiment device in the experiment. During the process, the flow patterns are observed and recorded by visual inspection, and the flow pattern is distinguished by referring to the theoretical diagram of the oil-water two-phase flow pattern. The prediction results show that the accuracy of these two algorithms can reach 81.25% and 93.75%, respectively, which verifies the effectiveness of these two algorithms in the prediction of oil–water two-phase flow patterns and provides a new idea for the prediction of oil–water two-phase flow patterns and other phases.

**Keywords:** inclined well; horizontal well; vertical well; GA-BP neural network; random forest algorithm; flow pattern prediction



**Citation:** Sun, Y.; Guo, H.; Liang, H.; Li, A.; Zhang, Y.; Zhang, D. A Comparative Study of Oil–Water Two-Phase Flow Pattern Prediction Based on the GA-BP Neural Network and Random Forest Algorithm. *Processes* **2023**, *11*, 3155. https://doi.org/10.3390/pr11113155

Academic Editors: Mehdi Ostadhassan, Xin Nie, Liang Xiao and Hongyan Yu

Received: 16 October 2023
Revised: 30 October 2023
Accepted: 2 November 2023
Published: 5 November 2023



## 1. Introduction

In recent years, the study of multiphase flow patterns [1–5] has been one of the research hotspots concerned by many scholars. At the same time, studying oil–water two-phase flow patterns is the basis for studying related multiphase flow patterns. As the oil field enters the late development period, the influence of water on the fluid flow pattern will be significantly increased. Hence, predicting the oil–water two-phase flow pattern is essential to mining. However, the flow state of oil–water two-phase flow in horizontal and deviated wells is more difficult to predict than that in vertical wells, especially the change of well deviation angle or flow velocity, which will have a more significant impact on the change of the flow pattern. For oil–water two-phase flow in horizontal wells, Tarllero [6] conducted

experiments with a pipe with an inner diameter of 5.013 cm and a length of 15.54 m at a temperature of 25.6 °C and determined six flow patterns, including segregated flow and dispersed flow. The segregated flow includes stratified flow and stratified flow with a mixed stratified interface. The dispersed flow includes a water-led flow pattern in which the dispersion of oil in water and the emulsion of oil in water occur, and an oil-led flow pattern in which the emulsion of water in oil and the dual dispersion of water in oil occur. Figure 1 shows the above six flow patterns. For inclined wells, Flores et al. [7] tested the flow patterns of 75°, 60°, and 45° inclination in a pipeline with an inner diameter of 5.08 cm and a length of 15.3 m at a temperature of 32.22 °C and obtained seven flow patterns, mainly consisting of water continuous phase and oil continuous phase. One type of transition flow pattern occurs between the continuous phase with water and the continuous phase with oil. It is worth mentioning that Flores concluded from the experiment that when the inclination angle of the pipeline is greater than 33°, the flow pattern of oil–water two-phase flow will not appear. Figure 2 shows the seven flow patterns in the above-inclined pipe. For vertical wells, Govier [8] et al. experimented with oil–water two-phase flow in a vertical transparent pipe with an inner diameter of 2.63 cm and a length of 11.3 m and concluded that the flow pattern of oil–water two-phase flow is similar to that of gas-water two-phase flow. Therefore, the flow patterns of oil–water two-phase flow obtained by them are:

bubble flow (water is a continuous phase).
slug flow (water is a continuous phase).
froth flow (no fixed continuous phase).
mist flow (oil is a continuous phase).

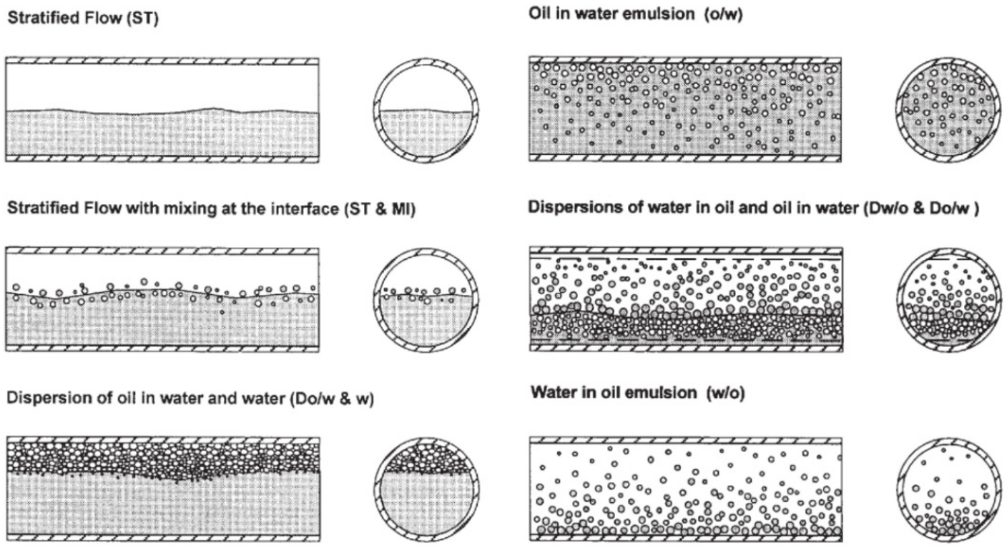



**Figure 1.** Flow pattern of oil–water two-phase flow in horizontal pipe.

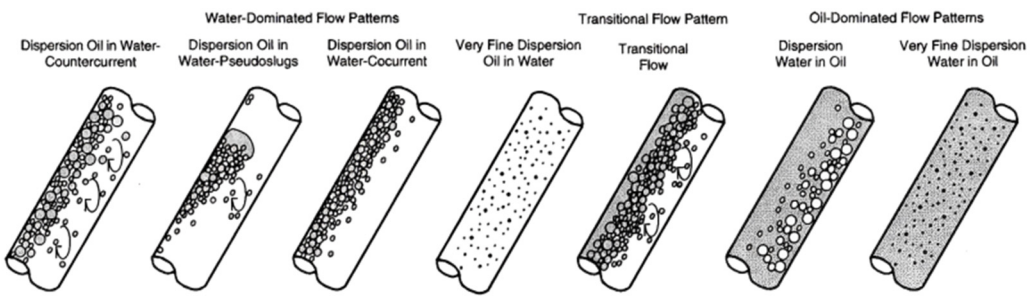



**Figure 2.** Flow pattern of oil–water two-phase flow in inclined pipe.

In addition, Flores et al. conducted an oil–water two-phase flow test with a pipeline inclination of 90°. They determined six flow patterns, with water as a continuous phase and oil as a continuous phase. The flow patterns determined by the two are more similar, but because of the difference in the choice of experimental equipment, the conclusions drawn are still slightly different. Figure 3 shows the flow pattern obtained by Flores et al. in a vertical pipe.

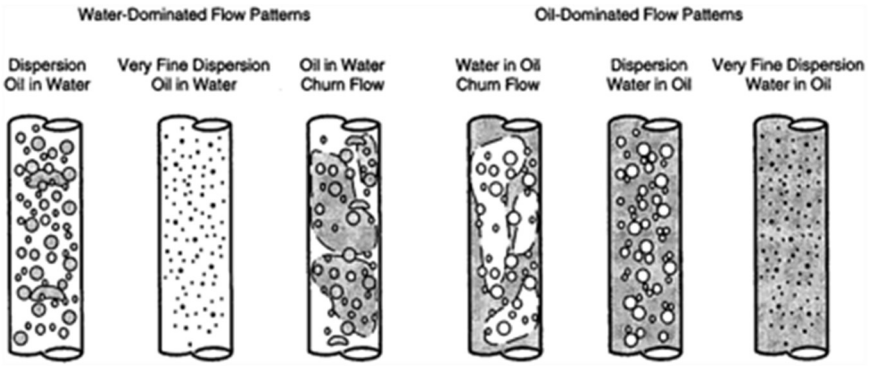



**Figure 3.** Flow pattern of oil–water two-phase flow in vertical pipe.

Despite ongoing research, due to different research focuses and perspectives and the influence of various factors, there has never been a unified conclusion on the definition of flow patterns. Currently, most of the flow pattern research is still in the subjective observation stage and relies on the flow pattern diagram. The lack of a qualitative judgement method is also a prominent reason for the absence of an overall definition of flow patterns. Therefore, scientific and accurate prediction of oil–water two-phase flow patterns is crucial for the safety and economy of process program design and operation. Secondly, it can also promote the innovation and development of related technologies, improve production efficiency, reduce environmental and other risks, enhance the controllability of the production process, and optimize human and material resources.

In recent years, many scholars have carried out relevant research on fluid flow patterns using computer numerical simulation [9–12]. Among them, Gupta et al. [11] used the ANSYS software package to model the Taylor flow in microchannels and proposed a standard for the fine mesh capture of films. Etminan et al. [12] analyzed the influence of a microchannel with a sudden increase in diameter on the fluid dynamics in the pipe. Through numerical simulation, researchers can simulate the fluid flow pattern under the influence of different factors. However, in the actual well, to determine the flow pattern, we need to use a logging instrument that is lowered into the well to measure a series of parameters. The numerical simulation mentioned above ignores the effect of the logging instrument on the fluid in the well. It is conceivable that such simulation results and the actual downhole flow pattern will have a particular deviation. On the other hand, the physical experiment must be consistent with the real wellbore, which is not very convenient. At the same time, necessary factors such as temperature and pressure physical experiments cannot fully reproduce these conditions in the real wellbore. In addition, the physical experiment also limits data points and a heavy workload of test personnel, resulting in errors and other problems.

With the emergence of deep learning and machine learning, the processing and analysis of data have become more efficient and accurate. Using these techniques can continually increase productivity and improve old methods. There are already many machine learning algorithm applications with the ability to predict relevant data in many fields, such as the application of a genetic algorithm in the field of construction [13], the prediction of the skid resistance of hybrid materials in the field of materials [14], and the prediction of protein secondary structure in the field of molecular biology [15]. Many scholars have also used these methods to predict fluid flow patterns, such as Qian et al. [5], who used support

vector machines to identify the flow patterns of oil–water two-phase flow. Mask et al. [16] improved the flow pattern prediction model based on a machine learning algorithm. Jefferson et al. [17] used void rate time series, signal processing, and machine learning for the classification prediction of flow patterns. Alhashem [18] tested the performance of five algorithms for flow pattern prediction based on a database. However, most of them are learning to predict using older data, and no newer experiments match their predictions. In this paper, we use a multiphase flow simulation experimental setup to conduct a variety of scenarios and collect 60 sets of data about the flow pattern. Table 1 describes the density versus viscosity of the oil and water used in the experiment. After sorting out the experimental data, the authors use two algorithms to learn and predict the sorted flow data, establish the corresponding prediction model after continuous parameter tuning, and compare the two algorithms. The purpose is to improve the accuracy and precision of flow pattern identification, provide a scientific basis for practical engineering applications, and promote the combination of traditional industrial technology and new-era technology.

**Table 1.** Parameters oil and water.

| | Density (g/cm$^3$) | Viscosity (mPa·s) |
|---|---|---|
| Oil | 0.826 | 2.92 |
| Water | 0.988 | 1.16 |

## 2. Algorithmic Principle

### *2.1. GA-BP Neural Networks*

The BP (Back Propagation) neural network [19] is one of the multilayer feedforward neural network models proposed in 1986. It has a powerful nonlinear mapping ability to recognize noisy samples without having to know in advance the mathematical equations describing the mapping relationship between inputs and outputs, and it is highly fault-tolerant so that when there is some damage or change in the middle, the overall performance is only slightly degraded. It uses the fastest descent method to constantly update and adjust the parameters of the network by constantly backpropagating the error to minimize its sum of squares. Its arithmetic speed is slow, hidden nodes are difficult to determine, and model selection is complex and easy to fall into the local minimum. Figure 4 shows the BP neural network flow chart.

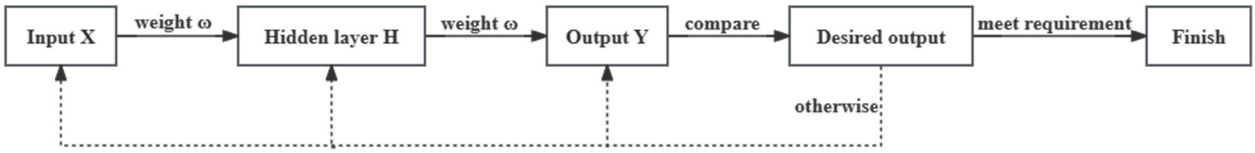



**Figure 4.** BP neural network flow chart.

Assume that the number of neurons in the input and output layers are m and n, respectively, and the number of neurons in the hidden layer is $d$ where $m = n$. First, the data enter the hidden layer from the input layer and are calculated to determine the output layer—the data calculation from the input layer to the hidden layer as in Equation (1).

$$\alpha_h = \sum_{i=1}^{m} v_{ih} x_i + \theta_h \tag{1}$$

The computation of the hidden layer to the output layer as in Equation (2).

$$\beta_j = \sum_{h=1}^{d} w_{hj} b_h + \theta_j \tag{2}$$

where $v_{ih}$ and $w_{hj}$ are the weights from the input layer to the hidden layer, $\theta_h$ and $\theta_j$ are the bias variables (activation functions) from the hidden layer to the output layer. $\alpha_h$ is the input to the $h$ hidden neuron, and $\beta_j$ are the input to the $j$ output neuron.

The calculation of the first process is random due to the weights and thresholds, resulting in a large error between the result and the expectation. It also requires back-propagation of the error to continuously adjust the parameters to be fitted better to minimize the error. When the calculated result does not equal the expectation, define this error as calculated in Equation (3).

$$E = \frac{1}{2}\left(\sum_{j=1}^{n} T_j - \beta_j\right)^2 \tag{3}$$

where $T_j$ is the $j$th expectation.

According to Formula (3), the error can be expanded into the hidden and input layers so that the weights of each layer can be updated and adjusted. Adjustment needs to set the learning rate of the model to control the pace of parameter adjustment, and the appropriate learning rate can converge the objective function to the local minimum at the appropriate time. The learning rate is set too small, which will result in slow convergence. Setting it too large will make it easier to converge the results. In general, set the learning rate to 0.01~0.8. Through the continuous learning of the training data and iteration, we can determine a suitable classification model to classify the dataset reasonably. Figure 5 shows the architecture of the BP neural network.

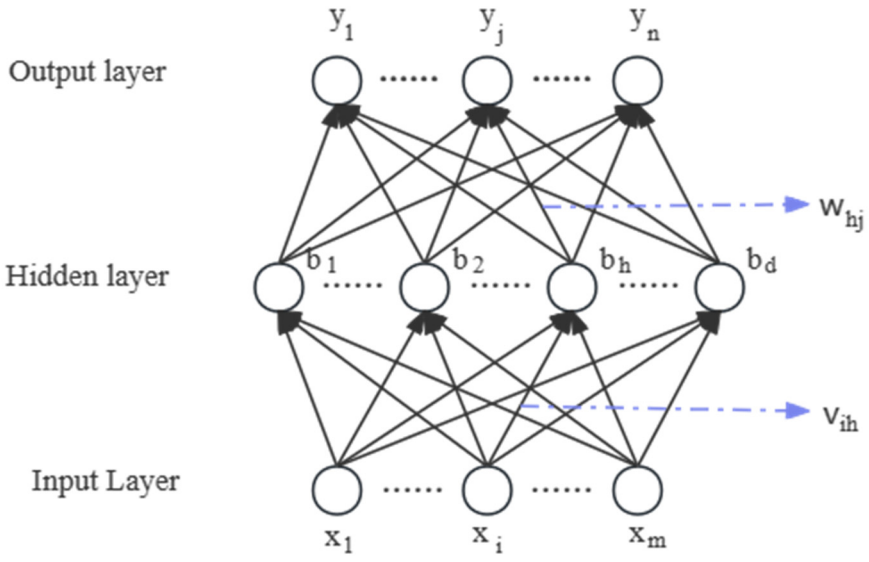



**Figure 5.** Architecture of the BP neural network.

Genetic Algorithm (GA) is a model that simulates the process of biologization, motivated by natural selection and the genetics of biological evolution. It can overcome the obstacles encountered by the traditional optimization algorithm, has the inherent parallelism and ability of parallel computation, takes the value of the objective function as the search information directly when searching, and is easy to combine with other technologies. However, it is easy to premature convergence and has low efficiency in processing data. The genetic algorithm optimizes the BP neural network, i.e., the GA-BP neural network [20], by optimizing the configuration of the network parameters and the minimum prediction error of the test set. The optimization of the BP network mainly includes the evolution of connection rights, the evolution of the network structure, the evolution of learning parameters, and the determination of the fitness function. The current optimization mainly improves the initialization weights of the BP network. Figure 6 shows the flowchart of the GA-BP algorithm.

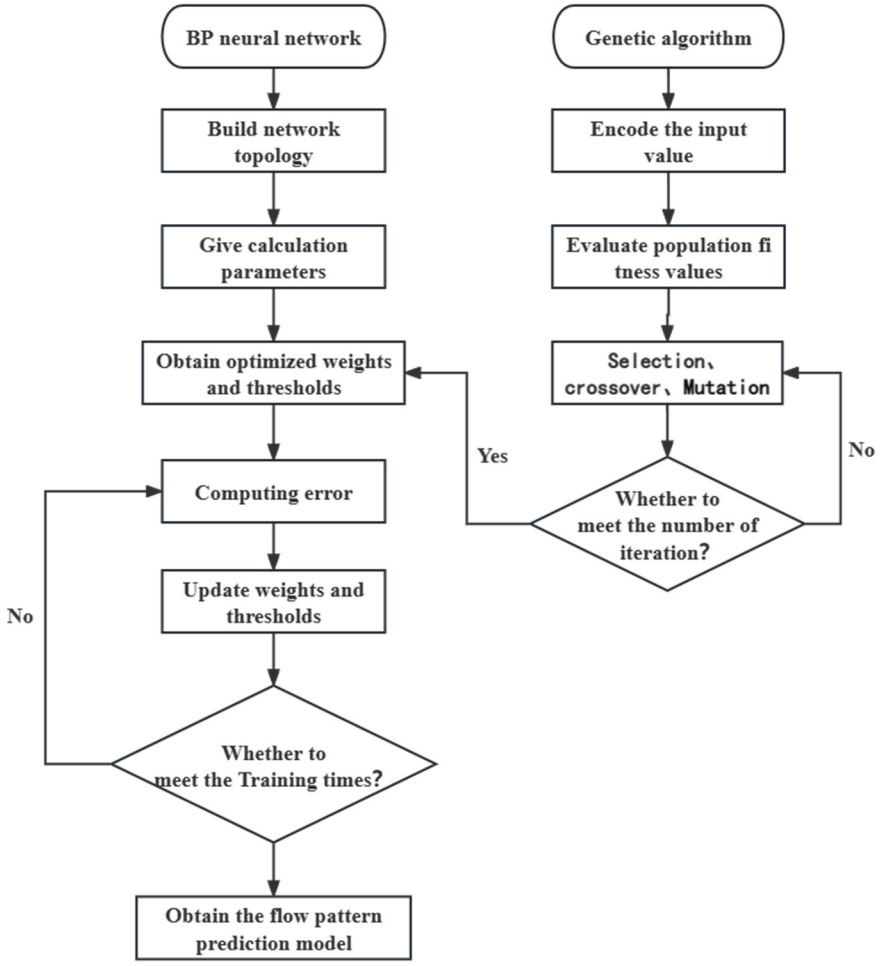



**Figure 6.** Flowchart of GA-BP algorithm.

### 2.2. Random Forest Algorithm

A decision tree is a classifier [21], a directed tree formed from a root node without input. Nodes with output are called internal nodes (test nodes), and other nodes are called leaf nodes (decision nodes). When using a decision tree, each internal node classifies samples into two or more categories by attribute values. Figure 7 depicts a decision tree concerning whether a customer will buy back a product.

Figure 7 shows a simple example of a decision tree, with circular nodes representing internal nodes and triangular nodes representing leaf nodes. We can turn that into a rule: "If the customer is male, older than 22, and has started work, then the customer will buy back the product". Generally, the complexity of a tree is measured by the depth of the tree, the total number of nodes, and other factors, and the complexity of a tree affects the accuracy of its classification results [22]. Decision tree generation is a process of recursively generating the optimal decision tree, which mainly includes three parts: feature selection, tree generation, and pruning. Pruning is to enhance the generalization ability of the model. Over the years, there have been many decision tree algorithms [23–25], and the random forest algorithm used in this paper is a combination of decision trees generated by the CART pruning algorithm [20].

The random forest algorithm is a classic machine learning algorithm in integrated learning and can improve the Bagging algorithm [26]. In 2001, Breiman [27] combined the tree model into a random forest to solve the problem and improve the prediction accuracy without significantly increasing the amount of computing. Compared with a traditional decision tree, which selects the optimal feature in each sample feature as the basis for the division of the left and right sub-trees of the decision tree, the random forest algorithm will

only randomly select the best feature among some sample features as the basis for division. There is no correlation between the decision trees, making the model more capable of generalization. After obtaining a forest, when predicting a new sample, the forest will let each decision tree make a judgment separately and then see which category is selected the most and predict which category this sample is in. Figure 8 depicts the process of generating a random forest.

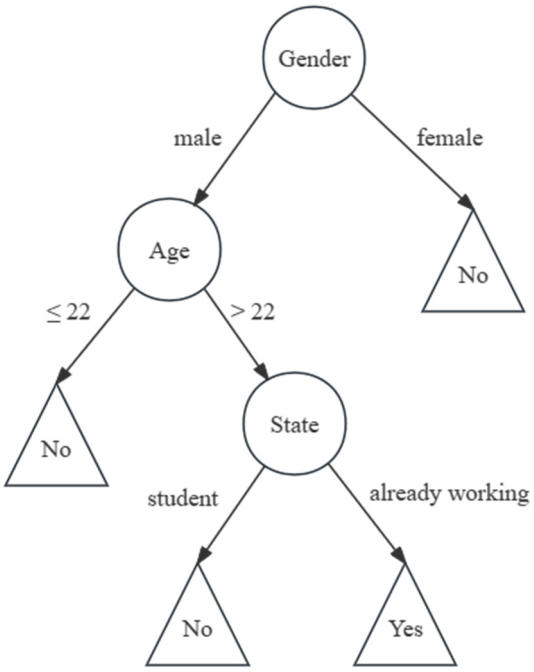



**Figure 7.** A decision tree that describes whether customers will buy back.

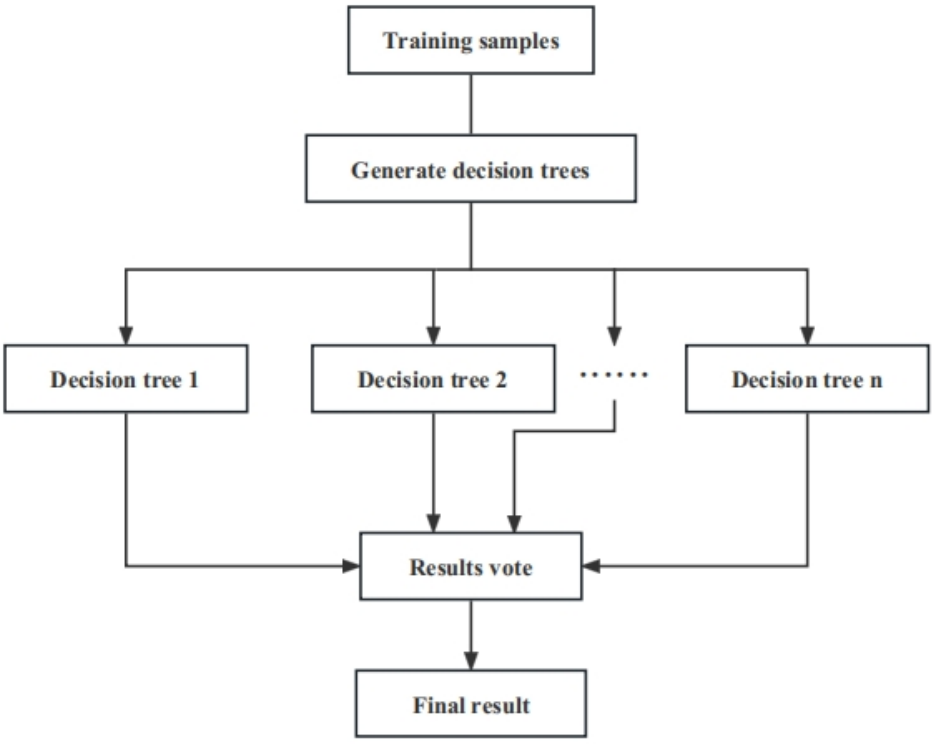



**Figure 8.** Random forest process schematic.

The random forest algorithm uses the Gini index to select features. The selection criterion of the Gini coefficient is that all the data in the child nodes belong to the same classification. The smaller the coefficient, the smaller the uncertainty, and the more thorough the data segmentation. In addition, the random forest can also use out-of-bag error for feature selection because the algorithm will randomly select some sample features, so there will be other samples that are not collected, and these samples can be used to measure the quality of the features. The Gini index is as in Equations (4) and (5) for a given set of samples.

$$Gini(D) = 1 - \sum_{k=1}^{K} \left( \frac{|C_k|}{|D|} \right)^2 \tag{4}$$

$$Gini(p) = \sum_{k=1}^{K} p_k(1 - p_k) = 1 - \sum_{k=1}^{k} p_k^2 \tag{5}$$

where $K$ is the number of sample categories, $D$ is the total number of samples, $p_k$ is the probability of each category, and $C_k$ is the number of samples belonging to the $k$th category in the set.

When constructing a random forest, the CART algorithm recursively bisects each feature, and the process consists of decision tree generation and pruning. Decision tree generation is to construct a binary tree recursively, and the generation process follows the minimization of the squared error and the minimization of the Gini index, followed by the generation of a binary tree. It is assumed that feature $A$ has two values, i.e., it can be divided into two nodes, the Gini index, after splitting as in Equation (6). When $Gini(D, A) = 0$, all samples belong to the same class.

$$Gini(D, A) = p_1 Gini(D_1) + p_2 Gini(D_2) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2) \tag{6}$$

After all the binary trees are generated, the next step is to use the CART pruning algorithm to obtain the optimal subtree. The algorithm is divided into two steps. First, the generated decision tree $T_0$ is pruned continuously until the root node, thus forming a sequence of subtrees $\{T_0, T_1, \ldots, T_n\}$, from which the optimal subtree can be chosen. The general process is as follows:

Define the tree model loss function as shown in Equation (7).

$$C_a(T) = C(T) + a|T| \tag{7}$$

where $C(T)$ is the model prediction error, $a|T|$ is the model complexity, and $|T|$ is the number of leaf nodes of the model. The coefficient $a$ is used to weigh the fit of the training set and the complexity of the model. When $a = 0$, the overall loss of the tree is minimized at this point, and the tree is the most lush. If $a$ tends to infinity, the tree at this point has only one root node, so as $a$ continuously increases in size, the tree continually decreases.

The loss function of any node t is given in Equation (8).

$$C_a(T_t) = C(T_t) + a|T_t| \tag{8}$$

When $a = 0$, there is the following Equation (9):

$$C_a(T_t) < C_a(t) \tag{9}$$

Clearly, as $a$ increases, it allows the following Equation (10):

$$C_a(T_t) = C_a(t) \tag{10}$$

At this point, the $t$ node can be considered to have the same amount of loss as the subtree with the $t$ node as the root node, at which point a can be made equal to the following:

$$a = g(t) = \frac{C(t) - C(T_t)}{|T_t| - 1} \tag{11}$$

When $a = g(t)$, pruning is performed, node $t$ is taken as a leaf node, its class is decided by majority voting on the leaf node, and the resulting subtree is taken as $T_1$.This continues until the root node is obtained. Finally, the optimal tree $T$ is selected among the sequence of subtrees using the cross validation method, where each decision subtree corresponds to one $a$.

## 3. Method Applications

Since both the GA-BP neural network and random forest algorithm have classification prediction functions, this study will use these two algorithms to analyze and predict the experimental flow pattern data and then conduct a comparative study on the prediction results of these two algorithms to prove the feasibility and accuracy of the flow pattern prediction results of these two algorithms. Because there are many different indicators in the experimental data, such as different water cuts, flow rates, and well slope, different indicators have different dimensions, and there is a large gap between the data. Therefore, the author adds data standardization processing to the two algorithms involved in this research. Here, the authors use the most typical data normalization process (maximum and minimum standardization) and convert the experimental data to the range [0,1]. Equation (12) normalizes the data.

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{12}$$

The first is the application of the GA-BP neural network. The standardized training set and test set are read into the algorithm, and the genetic algorithm encodes the four columns of data (three columns of influence factors and one column of flow-type data) in the training set; that is, the algorithm maps the flow pattern corresponding to each group of influencing factors into its representation. After encoding the flow pattern, the algorithm calculates the fitness value of each flow pattern data sample in the training set. Then, it selects better flow data samples according to the fitness values and crosses and mutates these good samples to generate new samples with excellent characteristics. The genetic algorithm will end the calculation when it reaches the set maximum number of iterations in the above operations and pass the obtained optimal parameters to the BP neural network, and the GA genetic algorithm ends. The BP neural network uses the transmitted parameters to classify and predict the data in the training set and calculates the error between the predicted results and the actual results. Then, the information is constantly backpropagated to the previous layers to update the weights and thresholds so that the error between the predicted results and the actual results can meet the set requirements as much as possible. When the BP neural network reaches the set training times, the calculation terminates. Then, the BP neural network will use the optimized parameters to classify and predict the data of the training set and the test set and output the results for convenient analysis. At this point, the BP neural network algorithm ends.

Next is the application of the random forest algorithm in this study. Read the above training set and test set into the random forest algorithm, and the random forest algorithm will randomly select $n$ samples from the training set to train a decision tree to take the sample of the decision root node. When training the decision tree, the algorithm will take any value $m$ ($m$ is less than the number of sample features) and select one attribute from the $m$ attributes as the split attribute of this node by using the principle of the Gini index, and repeat this training method until the node cannot be split. According to the above steps, many decision trees are established, and then the optimal decision tree sequence is

obtained by the pruning algorithm. At this point, a corresponding random forest model is established. Then, the algorithm will use the built model to predict the data of the training set and the test set, determine the final classification result, and output it for easy analysis. The random forest algorithm is then over.

## 4. Experiment Overview

This experiment was conducted under normal temperature and pressure (20 °C, 95.89 Kpa). We used a multiphase flow simulator to simulate the flow pattern mechanism under different influencing factors. The experimental equipment is shown in Figure 9, in which the simulated wellbore is a transparent glass pipe with an inner diameter of 12.4 cm and a length of 12 m. The parameters of oil and water used in the experiment can be referred to in Table 1.

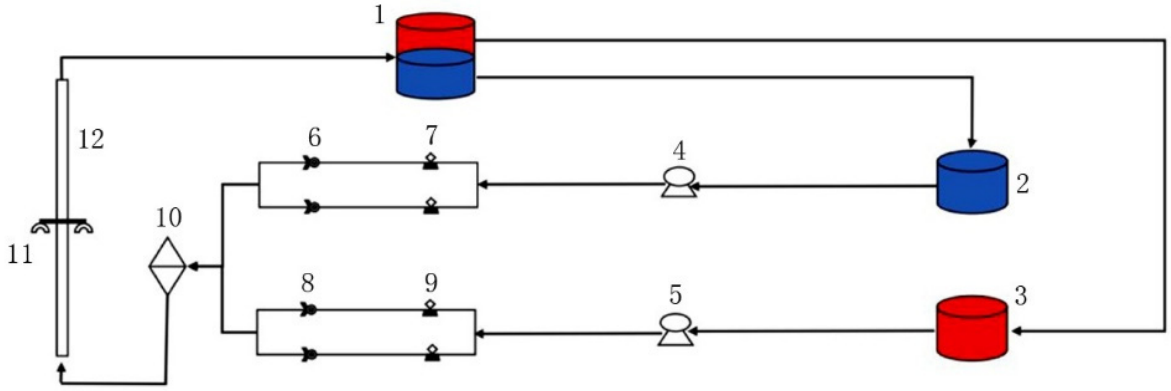



**Figure 9.** Schematic diagram of multiphase flow experimental device. 1. Oil–water separation tank; 2. Water storage tank; 3. Oil storage tank; 4, 5. Pressure pump; 6, 7, 8, 9: console; 10. mixing tank; 11, 12. Simulated wellbore.

As shown in Figure 9, we use blue for water and red for oil. at the beginning of the simulation experiment, after the fluid flow rate, water cut, and pipe inclination angle were adjusted by the console to allow the fluid to flow out of the liquid storage tank. It then entered the simulated wellbore through the pressure pump and pipe row area. After some time, the fluid flow state in the tube was stable. At this time, we used the camera to take pictures at the observation point, shoot videos for our records, and archive them after the experiment was over for follow-up inspection. After the simulation experiment, the fluid was separated into pure water and oil through the separation tank. Then, it flowed into the corresponding liquid storage tank to facilitate the recycling of subsequent experiments.

In this experiment, we collected 60 sets of valid flow pattern data. By comparing the flow pattern diagrams of pipes with different tilt angles, we roughly divided these 60 groups of flow patterns into five types, namely bubble flow, emulsion flow, froth flow, wavy flow, and stratified flow, and successively coded these five flow patterns into numbers 1 to 5 for differentiation, as shown in Table 2.

**Table 2.** Flow patterns and coding.

| Flow Pattern | Coding |
|---|---|
| bubble flow | 1 |
| emulsion flow | 2 |
| froth flow | 3 |
| wavy flow | 4 |
| stratified flow | 5 |

Before using the algorithm to predict the flow pattern, we collated the flow pattern data obtained from the simulation experiment. After sorting, the parameters affecting the flow pattern were found to include the slope of the wellbore, the flow rate of the fluid, and the water cut. Finally, the training data and the three features in the training data were read into the algorithm for model construction and prediction to test whether the algorithm is suitable for the prediction of oil–water two-phase flow and the accuracy of the predicted flow pattern of the algorithm. In the division of training data and test data, 16 data groups were randomly selected from the four groups, with 20%, 60%, 80%, and 90% water cuts as the training set, and all the remaining data were used as the test set.

After reading these two data sets into the algorithm, the algorithm will learn and build the model on the data of the training set. Among them, the GA-BP neural network algorithm will first preprocess the training data through the genetic algorithm, such as data coding and fitness value calculation, which is related to how well a single sample adapts to the population, so the larger the fitness value, the better. Then, the genetic algorithm obtains the optimal weights and thresholds according to a series of operations. Finally, the BP neural network continuously updates these parameters to ensure that the predicted flow patterns are closer to the actual flow patterns. The random forest algorithm only randomly selects part of the samples to build a large number of prediction models according to the rules of sample random, feature random, and sampling with replacement.

After the model is constructed, the two algorithms will automatically extract features from the data of the training set and the test set and predict the flow pattern. We compare the predicted flow pattern results with the experimental flow pattern under the corresponding conditions and then compare and analyze the prediction results of the two algorithms.

## 5. Analysis of Projected Results

When the two algorithms are predicted, we count all the predicted results and then compare the predicted results with the experimental results to analyze the predicted results and give a conclusion. Figure 10a shows the confusion matrix of the prediction results of the GA-BP algorithm on the training set, and Figure 10b shows the confusion matrix of the prediction results of this algorithm on the test set. The numbers 1, 2, 3, 4, and 5 on the abscissa and ordinate correspond to the five flow patterns in Table 2, respectively. The blue squares and the number in the square on the diagonal in the figure indicate the predicted correct flow pattern and the number of samples, and the red square and the number in the squares indicate the predicted incorrect flow pattern and the number of samples. The percentages in the blue and red squares in the figure indicate the number of correctly and incorrectly predicted flow pattern samples in each row and column as a proportion of the total samples. Figure 12a,b are consistent with the representations of Figure 10a,b.

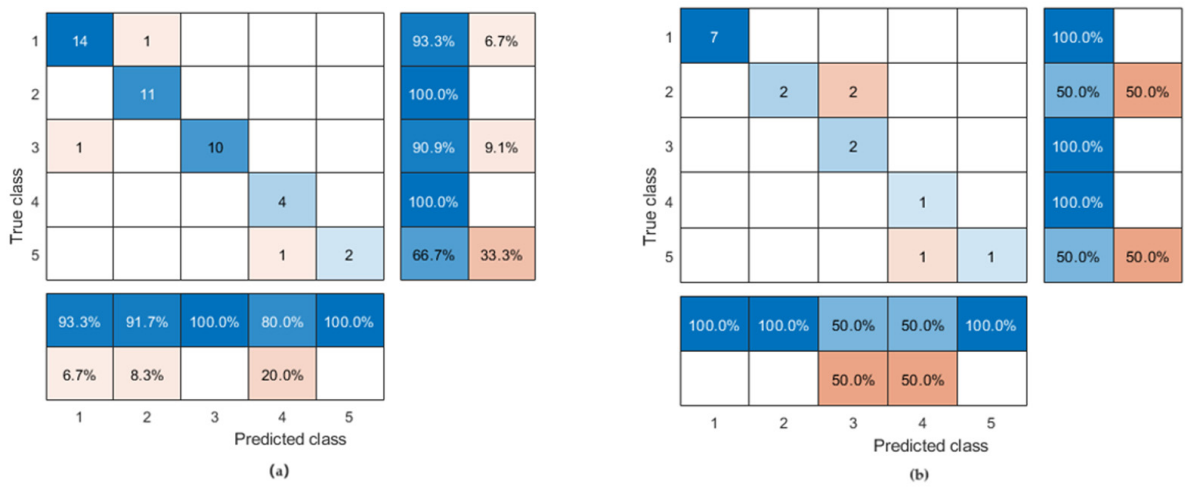



**Figure 10.** Confusion matrix of the predicted results of GA-BP algorithm.

For the two confusion matrices above, the rows represent the observed flow class for the trial, and the columns represent the predicted flow class. Therefore, it can be seen from Figure 10a that the GA-BP neural network has three prediction errors in the prediction of the training set after training. These predict a bubble flow as an emulsion flow, a froth flow as a bubble flow, and a stratified flow as a wavy flow, and the accuracy is 93.18%. The prediction results of the test set in Figure 10b show that the algorithm predicts two emulsion flows as froth flows and a stratified flow as a wavy flow with an accuracy of 81.25%. It is proved that this algorithm has high performance. Figure 11a shows the scatter plot of the prediction results of the GA-BP algorithm on the training set, and Figure 11b shows the scatter plot of the prediction results of the GA-BP algorithm on the test set. The numbers 1, 2, 3, 4, and 5 on the ordinate correspond to the five flow patterns in Table 2, respectively, and the numbers on the abscissa represent the number of samples. The red and blue lines in the figure represent the broken lines of the actual and predicted flow patterns, respectively. Figure 13a,b are consistent with the representations of Figure 11a,b.

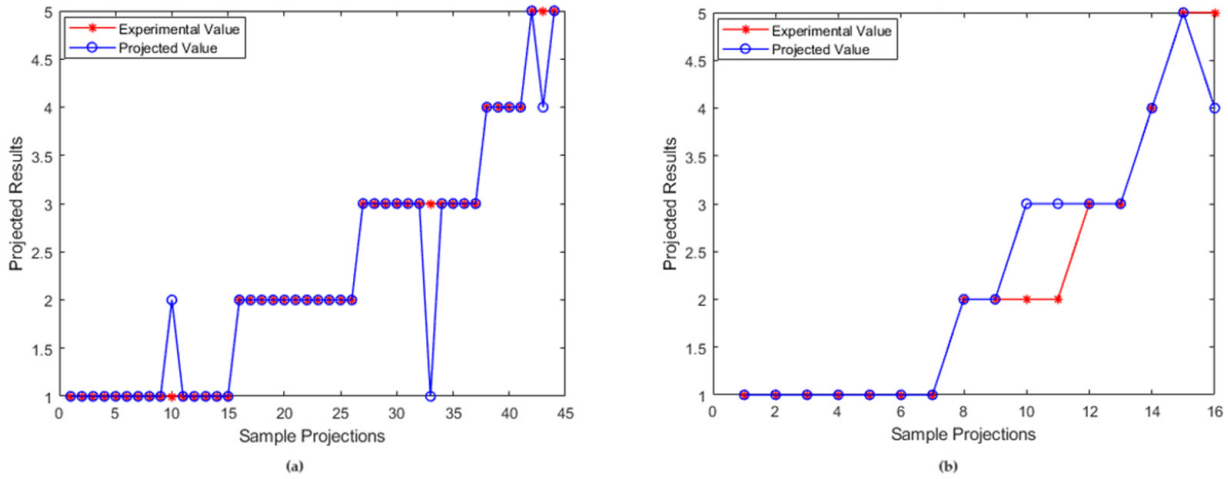



**Figure 11.** Scatterplot of the predicted results of GA-BP algorithm.

Figures 12 and 13 show the confusion matrix and scatter plot of the random forest predictions for the training set and test set.

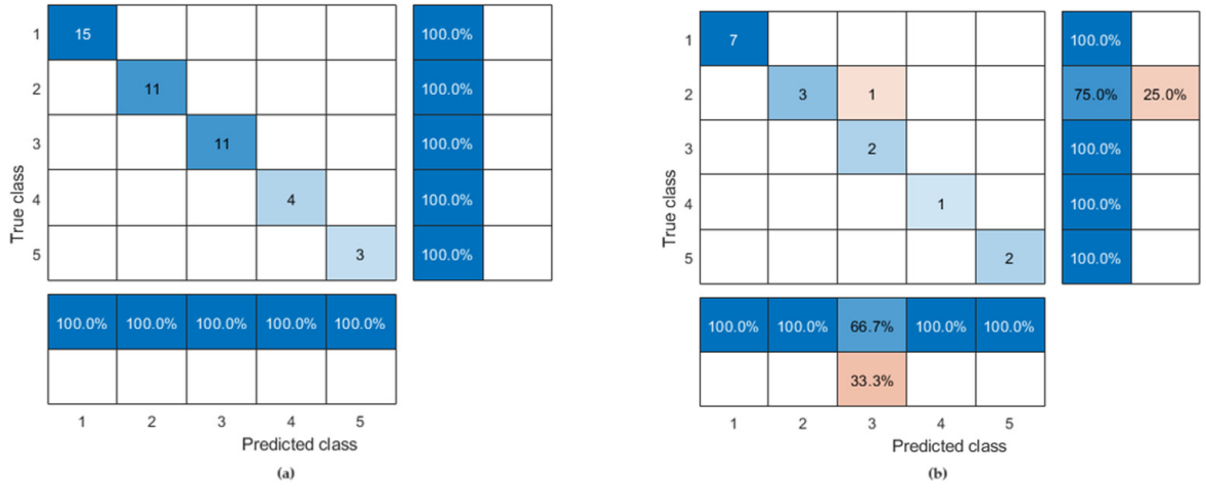



**Figure 12.** Confusion matrix of the predicted results of random forest algorithm.

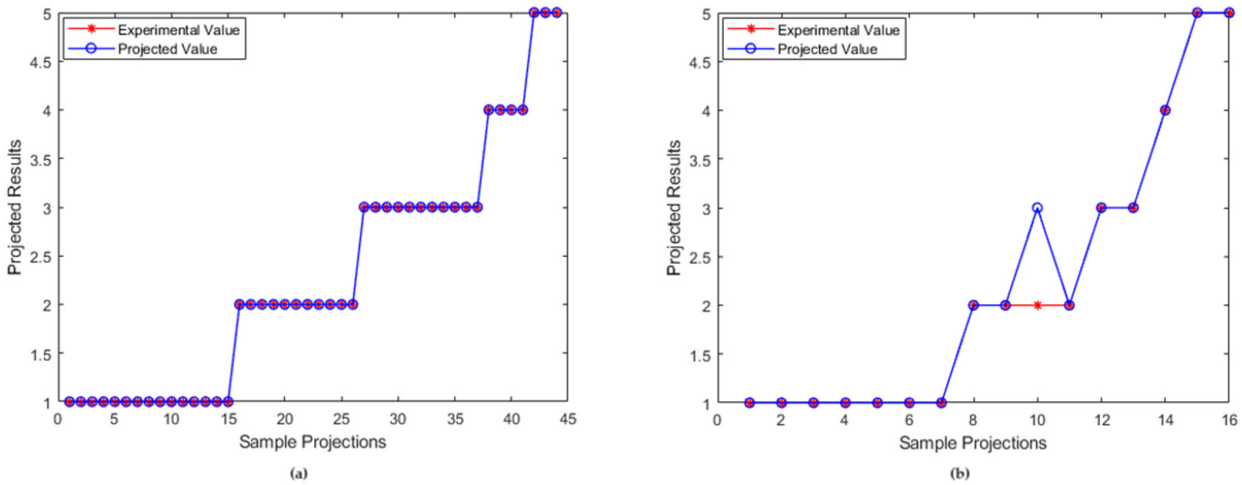



**Figure 13.** Scatterplot of the predicted results of random forest algorithm.

As can be seen from Figure 12a, the prediction accuracy of the training set used in this experiment after the random forest training can reach 100% at most, which shows that the random forest algorithm has a perfect prediction of the training set after training. The prediction accuracy of the test set can reach 93.75%, and only an emulsion flow pattern is predicted to be froth flow. It is proved that the random forest algorithm dramatically improves the learning and prediction ability of the flow-type compared with the GA-BP neural network.

Detailed information about the prediction results of the two algorithms for the flow patterns is in Table 3.

**Table 3.** GA-BP neural network and random forest prediction results.

| Water Cut (%) | Angle of Inclination (°) | Flow Rate ($m^3$/d) | Experimental Flow Pattern | GA-BP | Accuracy Rate | Random Forest | Accuracy Rate |
|---|---|---|---|---|---|---|---|
| 20 | 0 | 100 | 1 | 1 | 81.25% | 1 | 93.75% |
| | 85 | 600 | 2 | 2 | | 2 | |
| | 85 | 100 | 4 | 4 | | 4 | |
| | 90 | 600 | 2 | 2 | | 2 | |
| 40 | 60 | 100 | 1 | 1 | | 1 | |
| | 85 | 300 | 1 | 1 | | 1 | |
| | 90 | 100 | 5 | 5 | | 5 | |
| | 90 | 600 | 3 | 3 | | 3 | |
| 60 | 0 | 600 | 2 | **3** | | **3** | |
| | 60 | 100 | 1 | 1 | | 1 | |
| | 60 | 600 | 2 | **3** | | 2 | |
| 80 | 0 | 100 | 1 | 1 | | 1 | |
| 90 | 0 | 100 | 1 | 1 | | 1 | |
| | 90 | 100 | 5 | **4** | | 5 | |
| | 90 | 300 | 1 | 1 | | 1 | |
| | 90 | 600 | 3 | 3 | | 3 | |

As can be seen from Table 3, when at 60% water cut, the accuracy of the prediction results of both algorithms is reduced compared to other water cut groups; in particular,

both algorithms miss the prediction for a water cut of 60%, well inclination of 0°, and a flow rate of 600 $m^3/d$. For water cuts of 20%, 40%, and 80%, both of them perform well in predicting the flow pattern, but the prediction of 80% water cut flow pattern still needs to be verified subsequently. From the overall accuracy, the prediction accuracy of the GA-BP neural network is 81.25%. When the number of decision trees is moderate, the random forest has good prediction accuracy in different water cuts, well deviation, and flow, and its total accuracy can reach more than 90%. The above data illustrate the feasibility of these two algorithms for flow pattern prediction with some reasonable accuracy. Figure 14 illustrates the prediction accuracy of the two algorithms for the five flow patterns.

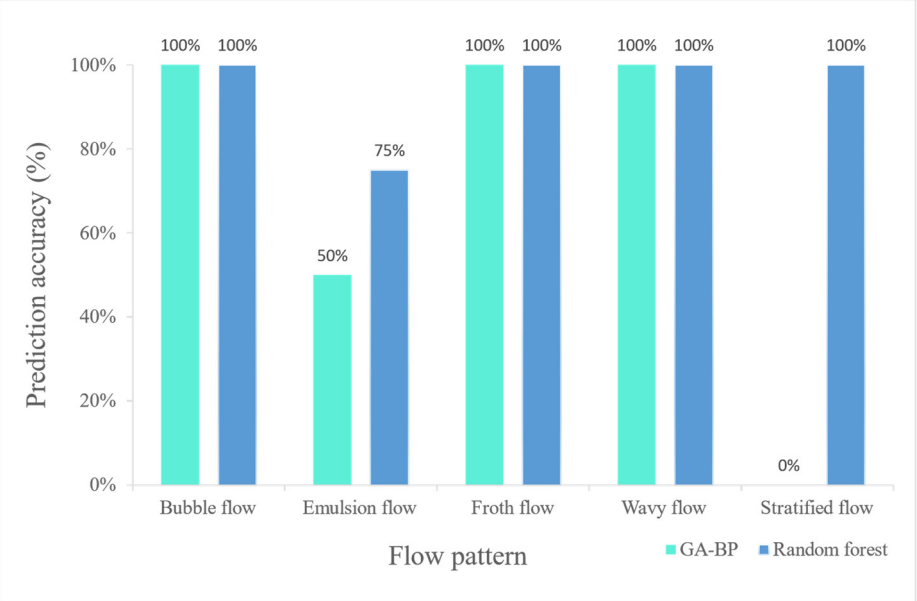
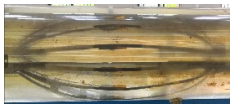



**Figure 14.** Accuracy of flow pattern prediction Conclusions.

From Figure 13, we can intuitively see that the prediction accuracy of the random forest algorithm is higher than that of the GA-BP neural network. At the same time, the accuracy of these two algorithms in predicting bubble flow, froth flow, and wavy flow can reach 100%. However, both algorithms have a specific decrease in the prediction accuracy of emulsion flow, where the prediction accuracy of the random forest algorithm is 75%, and the prediction accuracy of the GA-BP neural network is 50%. In addition, the GA-BP neural network does not identify the stratified flow, which may be related to the small number of samples, because it will lead to errors when the genetic algorithm optimizes the parameters according to the features and then affects the prediction results of the BP neural network. However, the random forest algorithm can accurately identify the stratified flow under the training of only one stratified flow data. It is related to the fact that it constructs many decision trees. All decision trees will contribute to a prediction result, which also shows the feasibility and high accuracy of the random forest algorithm for predicting the flow pattern of oil–water two-phase flow. In order to make the experimental results more convincing, more effective flow data will be counted in the future, and more training sets will be randomly selected from the experimental data for prediction. This will make the prediction accuracy of the two algorithms higher. Table 4 shows the flow pattern details of the prediction errors.

**Table 4.** Comparison of actual and predicted flow patterns.

| Experimental Flow Patterns | Actual Flow Pattern | GA-BP Predictive Flow Patterns | Random Forest Predictive Flow Patterns |
|---|---|---|---|
|  | Emulsion flow | Froth flow | Froth flow |
|  | Emulsion flow | Froth flow | Emulsion flow |
|  | Stratified flow | Wavy flow | Stratified flow |

## 6. Conclusions

After the above two machine learning algorithms predict the experimental flow pattern data, we can obtain the following conclusions through the comparison of the results and the analysis of the accuracy of the flow pattern prediction:

(1) Good at combining different machine learning algorithms to predict the wellbore fluid flow pattern. New ideas or methods of flow pattern prediction can be proposed by comparing and analyzing the accuracy of the flow pattern predicted by different methods.

(2) Randomly select different training sets from the experimental data, and then train the two algorithms according to the experimental design and make predictions. After a large number of prediction results are compared, the prediction accuracy of the random forest algorithm is better than that of the GA-BP neural network. The random forest algorithm has good prediction results under different flow states with different parameters. Many prediction results show that the prediction accuracy can be stable above 90%.

(3) Selecting the appropriate data set from the experimental data as the training set and performing appropriate data processing on it, such as normalization processing, dimension reduction processing, and, more importantly, continuously tuning the parameters of the algorithm, all of which significantly improve the efficiency of computer data processing.

(4) After that, the parameters of the random forest algorithm can be continuously tuned, and more experimental data can be expanded to increase the number of training set data to continuously improve the algorithm's accuracy.

**Author Contributions:** Conceptualization, Y.S.; Methodology, Y.S.; Software, Y.S.; Validation, Y.S., H.L., A.L., Y.Z. and D.Z.; Formal analysis, Y.S.; Investigation, Y.S., H.L., A.L. and Y.Z.; Data curation, Y.S.; Writing—original draft, Y.S.; Writing—review & editing, H.G., A.L. and D.Z.; Project administration, Y.S. and H.G.; Funding acquisition, H.G. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** Where data is unavailable due to privacy or ethical restrictions.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Wu, Y.; Guo, H.; Song, H.; Deng, R. Fuzzy inference system application for oil-water flow patterns identification. *Energy* **2022**, *239*, 122359. [CrossRef]
2. Ohnuki, A.; Akimoto, H. Experimental study on transition of flow pattern and phase distribution in upward air-water two-phase flow along a large vertical pipe. *Int. J. Multiph. Flow* **2000**, *26*, 367–386. [CrossRef]
3. Xu, X.X. Study on oil-water two-phase flow in horizontal pipelines. *J. Pet. Sci. Eng.* **2007**, *59*, 43–58. [CrossRef]
4. Bannwart, A.C.; Rodriguez, O.M.; Trevisan, F.E.; Vieira, F.F.; De Carvalho, C.H. Experimental investigation on liquid-liquid-gas flow: Flow patterns and pressure-gradient. *J. Pet. Sci. Eng.* **2009**, *65*, 1–13. [CrossRef]

5. Su, Q.; Li, J.; Liu, Z. Flow Pattern Identification of Oil–Water Two-Phase Flow Based on SVM Using Ultrasonic Testing Method. *Sensors* **2022**, *22*, 6128. [CrossRef] [PubMed]
6. Trallero, J.L. *Oil-Water Flow Patterns in Horizontal Pipes*; The University of Tulsa: Tulsa, OK, USA, 1995.
7. Flores, J.G.; Chen, X.T.; Sarica, C.; Brill, J.P. Characterization of oil-water flow patterns in vertical and deviated wells. *SPE Prod. Facil.* **1999**, *14*, 102–109. [CrossRef]
8. Govier, G.W.; Sullivan, G.A.; Wood, R.K. The upward vertical flow of oil-water mixtures. *Can. J. Chem. Eng.* **1961**, *39*, 67–75. [CrossRef]
9. Yi, X.; Hao, P.; Yi, L.; Wang, H. Flow pattern identification for gas-oil two-phase flow based on a virtual capacitance tomography sensor and numerical simulation. *Flow Meas. Instrum.* **2023**, *92*, 102376.
10. Gao, H.; Gu, H.Y.; Guo, L.J. Numerical study of stratified oil-water two-phase turbulent flow in a horizontal tube. *Int. J. Heat Mass Transf.* **2003**, *46*, 749–754. [CrossRef]
11. Gupta, R.; Fletcher, D.F.; Haynes, B.S. On the CFD modelling of Taylor flow in microchannels. *Chem. Eng. Sci.* **2009**, *64*, 2941–2950. [CrossRef]
12. Etminan, A.; Muzychka, Y.S.; Pope, K. Numerical investigation of gas–liquid and liquid–liquid T aylor flow through a circular microchannel with a sudden expansion. *Can. J. Chem. Eng.* **2022**, *100*, 1596–1612. [CrossRef]
13. Yu, W.; Li, B.; Jia, H.; Zhang, M.; Wang, D. Application of multi-objective genetic algorithm to optimize energy efficiency and thermal comfort in building design. *Energy Build.* **2015**, *88*, 135–143. [CrossRef]
14. Zheng, D.; Qian, Z.D.; Liu, Y.; Liu, C.B. Prediction and sensitivity analysis of long-term skid resistance of epoxy asphalt mixture based on GA-BP neural network. *Constr. Build. Mater.* **2018**, *158*, 614–623. [CrossRef]
15. Hua, S.J.; Sun, Z.R. A novel method of protein secondary structure prediction with high segment overlap measure: Support vector machine approach. *J. Mol. Biol.* **2001**, *308*, 397–407. [CrossRef] [PubMed]
16. Mask, G.; Wu, X.; Ling, K. An improved model for gas-liquid flow pattern prediction based on machine learning. *J. Pet. Sci. Eng.* **2019**, *183*, 106370. [CrossRef]
17. Ambrosio, J.D.S.; Lazzaretti, A.E.; Pipa, D.R.; da Silva, M.J. Two-phase flow pattern classification based on void fraction time series and machine learning. *Flow Meas. Instrum.* **2022**, *83*, 102084. [CrossRef]
18. Alhashem, M. Machine learning classification model for multiphase flow regimes in horizontal pipes. In Proceedings of the International Petroleum Technology Conference, Dhahran, Saudi Arabia, 13–15 January 2020; p. D023S042R001.
19. Zhou, Y.M.; Wang, S.W.; Lin, L. An Application of BP Neural Network Model to Predict the Moisture Content of Crude Oil. *Adv. Mater. Res.* **2012**, *524–527*, 1327–1330. [CrossRef]
20. Shi, S.; Liu, J.; Hu, H.; Zhou, H. A research on a GA-BP neural network based model for predicting patterns of oil-water two-phase flow in000 horizontal wells. *Geoenergy Sci. Eng.* **2023**, *230*, 212151. [CrossRef]
21. Rokach, L.; Maimon, O. Decision trees. In *Data Mining and Knowledge Discovery Handbook*; Springer: New York, NY, USA, 2015; pp. 165–192.
22. Breiman, L.; Friedman, J.; Olshen, R.; Stone, C. *Classification and Regression Trees*; Wadsworth Int. Group: Belmont, CA, USA, 1984; Volume 37, pp. 237–251.
23. Quinlan, J.R. *C4. 5: Programs for Machine Learning*; Elsevier: Amsterdam, The Netherlands, 2014.
24. Shafer, J.C.; Agrawal, R.; Mehta, M. A scalable parallel classi er for data mining. In Proceedings of the 22nd International Conference on VLDB, Mumbai, India, 3–6 September 1996.
25. Mehta, M.; Agrawal, R.; Rissanen, J. SLIQ: A fast scalable classifier for data mining. In *Advances in Database Technology—EDBT'96: 5th International Conference on Extending Database Technology Avignon, France, 25–29 March 1996 Proceedings 5*; Springer: Berlin/Heidelberg, Germany, 1996; pp. 18–32.
26. Wang, S.M.; Zhou, J.; Li, C.Q.; Armaghani, D.J.; Li, X.B.; Mitri, H.S. Rockburst prediction in hard rock mines developing bagging and boosting tree-based ensemble techniques. *J. Cent. S. Univ.* **2021**, *28*, 527–542. [CrossRef]
27. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [CrossRef]