Optimization of Random Forests Algorithm Based on ReliefF-SA

Zibo Zhou¹*, Yunfan Wang², Xinrui He³ and Xuecong Zhang⁴

¹School of Jinan University, Guangzhou, China
²School of Jinan University, Guangzhou, China
³School of Jinan University, Guangzhou, China
⁴School of Jinan University, Guangzhou, China

*Corresponding author e-mail: 790500195@qq.com

Abstract. As a widely used classification integration algorithm, the random forests based on decision tree combination is mostly used for classification and regression problems, but its deficiencies still need to be improved. In this paper, we propose simulated annealing algorithm, and its parameters is combined with the global search optimal solution and local optimal solution to optimize search. Then, we propose a random forests hybrid algorithm by combining the Relief algorithm and annealing algorithm, briefly introduce its thoughts and algorithm flow, and also carry out simulation experiments. According to the experimental results, the classification effect of the hybrid algorithm is generally the best, which greatly improves the overall generalization performance of the random forests.

1. Introduction

As one of the classical classification algorithms, it is the focus of people's research to improve and develop the decision tree algorithm. It is obviously hard to complete a complex classification by a single classifier, but Ensemble Learning can solve this problem conveniently. Ensemble Learning [1] regards multiple learners as multiple modules to coordinate and solve the same problem, and the effect of enhancing the generalization ability of the learning system by Ensemble Learning method to [2] is particularly remarkable.

The Ensemble Learning algorithm has experienced the initial development when it uses a large number of individual learners such as the Boosting method [3] and the Bagging method [4]. In 1995, Tin Kam Ho first proposed the concept of Random Decision Forests [5], in 1998. A further Random Subspace method was proposed [6]. Later, Breiman officially proposed the random forests algorithm in 2001 [7], which combines multiple decision trees to construct the Bagging ensemble foundation, and selects the training decision tree by random attribute features selection method to reduce the possibility of over-fitting.

However, due to its algorithmic limitations, random forests also give many scholars the idea of improving research. Generally, starting from the classifier, scholars enhance the accuracy of the classifier and reduce the correlation between classifiers.

There are many related researches on random forests. This paper discusses the feature selection of random forests algorithm and the combination optimization of multiple decision trees.

As for feature selection, Genuer et al. Put forward the idea of feature selection in two stages to avoid redundant data affecting classification effect [8]. However, the disadvantage of this method is
that it is too time-consuming and needs to generate classification model twice. In addition, Xu et al. conducted weighted feature extraction based on Chi square test and information gain ratio in 2011 [9].

As for the combination optimization of multiple decision trees, Bloodgood et al. proposed a weighting concept in 2012 [10], which evaluates the weight according to the performance of the decision tree on OOB and then conducts ensemble voting. This method makes the accuracy of the decision tree model proportional to the weight, but does not consider the diversity of the decision tree combination.

2. Related work

2.1. simulated Annealing

The Simulated Annealing (SA) was first proposed in 1953 by N. Metropolis [11] et al. In 1983, S. Kirkpatrick et al. applied the concept of simulated annealing to combinatorial optimization.

The main idea of SA is to find the global optimal solution by repeating this process: generating the initial solution $\rightarrow$ the new solution of disturbance $\rightarrow$ discriminating whether to accept the new solution.

2.2. ReliefF algorithm

Kononenko proposed the ReliefF algorithm in 1994 [12], which is mostly used for multi-classification. The process of ReliefF algorithm is: set the number of samples, take one sample R randomly from the training sample set each time, record the characteristic data of sample R as $R = (x_{R1}, x_{R2}, x_{R3}, ..., x_{RM})$, and the characteristic data of sample $R'$ is $R' = (x_{R'1}, x_{R'2}, x_{R'3}, ..., x_{R'M})$, measure the distance between two samples $R$ and $R'$ according to the distance formula (1) to search the nearest neighbor samples.

$$
\text{dist}(R, R') = \sqrt{(x_{R1} - x_{R'1})^2 + (x_{R2} - x_{R'2})^2 + \cdots + (x_{RM} - x_{R'M})^2} \quad (1)
$$

Next, k neighbor samples $H$ (nearHits) of R are found out from the sample sets of the same kind as R, k neighbor samples (nearMisses) are found out from the sample sets of different classes of R, and then the weight of each feature is updated, as shown in formula (2):

$$
W(A) = W(A) - \sum_{j=1}^{k} \frac{\text{diff}(A, R, H_j)}{mk} + \sum_{c \in \text{class}(R)} \left[ \frac{p(C)}{1 - p(\text{Class}(R))} \right] \sum_{j=1}^{k} \frac{\text{diff}(A, R, M_j(C))}{mk} \quad (2)
$$

Note: $M_j(C)$ represents the j-th nearest neighbor sample in class C, and $\text{diff}(A, R_1, R_2)$ represents the difference between the sample and the sample on feature A. Its calculation formula is:

$$
\text{diff}(A, R_1, R_2) = \begin{cases} 
\frac{|R_1[A] - R_2[A]|}{\max(A) - \min(A)}, & \text{if } A \text{ is continuous} \\
0, & \text{if } A \text{ is discrete and } R_1[A] = R_2[A] \\
1, & \text{if } A \text{ is discrete and } R_1[A] \neq R_2[A] 
\end{cases} \quad (3)
$$

3. Weighted random forests algorithm based on SA

In this paper, the simulated annealing algorithm is used to reduce the decision tree of random forests and do the secondary training of decision tree weights. Through continuous annealing, we make the weight of the deleted tree to be zero, and find the most suitable weight vector. By this way, decision tree with more generalization performance takes more weight and improves the classification performance of the random forests classifier combination.

3.1. pruning trees

When there is no difference in voting weights, all decision trees are first sorted in ascending order according to generalization performance, and then all trees in the random forests are pre-deleted. If a tree is deleted and the overall generalization performance is improved, then we delete this tree. The deletion method assigns the voting weight to 0, and the decision tree voting is not accepted. The reason why sorting and pre-deletion are needed here is that the decision tree with low generalization performance does not necessarily improve the overall performance. Some decision trees have low
generalization performance, but the correct coverage area is that most other decision trees cannot be correctly judged. In the region, the decision tree can also improve the overall generalization performance of the random forests, so use pre-deletion. Sorting is also due to universal problem. Most decision trees that provide negative generalization are often decision trees with low generalization performance, so sorting is a good solution to the accidental deletion.

Algorithm 1: Pruning tree

**Input:** decision tree voting coefficient vector \( w \) after tree deletion

**Output:** optimal coefficient vector \( w' \)

1. initialization temperature \( t = 500 \)
2. while \( t > 20 \)
3. for \( i \) in range (0, 200)
   4. a new vector \( w' \) is generated by case disturbance, so that the weight of the pruned tree is 0.
   5. calculate \( \Delta f \).
   6. judge whether to accept the new solution.
4. end
5. conduct temperature drop according to the temperature drop Formula \( t = 0.95 \times t \).

3.2. weight training

The weighted secondary training of the random forests with the deletion operation is carried out by the simulated annealing algorithm, is aim to obtain the optimal voting weight ratio, and improve the optimization performance of the random forests.

Algorithm 2: Secondary training base on SA

**Input:** decision tree voting coefficient vector \( w \) after tree deletion

**Output:** optimal coefficient vector \( w' \)

1. initialization temperature \( t = 200 \)
2. while \( t > 20 \)
3. for \( k \) in range (0, 200)
   4. generate new \( w' \) by the renewal rule.
   5. calculate \( \Delta f \).
   6. judge whether to accept the new solution.
4. end
5. conduct temperature drop according to the temperature drop Formula \( t = 0.95 \times t \).

4. Random forests hybrid algorithm

In this section, we will try a hybrid algorithm. In addition to the SA-RF algorithm above, we will use the ReliefF algorithm for feature selection. Through the combination of these two methods, the algorithm can coordinate the global search between the strength and correlation of the decision tree, and get better optimization results.

This is the re-optimization of the random forests model. By synthesizing the two optimization methods, a more effective and systematic improved algorithm is proposed:

Algorithm 3: Random forests hybrid algorithm

**Input:** dataset, the size of decision trees: \( nTree \), the number of partition attributes: \( k \)

**Output:** classification results

1. for \( i \) in range(0, \( nTree \))
   2. a random subset of size \( n \) is selected for each tree by using bootstrap sampling.
   3. get the weight of all features by using ReliefF algorithm.
   4. select the first \( k \) largest features at the node, compare and select the best features, and divide the data set.
   5. generate every decision tree recursively.
4. end
5. prun the decision trees, train the weight of the decision trees, get the best weight.
5. Simulation experiment and analysis
The experiment adopts the 4-fold cross validation method. The simulation software used is Matlab R2016a. The experimental environment was: Windows10 64-bit operating system, Intel Core i5-7300HQ, 2.50Ghz, 8GB memory.

5.1. Experimental data
The dataset used in this paper is from the UCI dataset. The 10 datasets obtained are shown in Table 1 below.

| Dataset                      | Number of sample | Number of attribution | Number of categories |
|------------------------------|------------------|-----------------------|----------------------|
| Glass                        | 214              | 9                     | 6                    |
| Segmentation(ST)             | 210              | 19                    | 7                    |
| Seeds                        | 210              | 7                     | 3                    |
| Winequality-red(WR)          | 1599             | 11                    | 6                    |
| Pima                         | 768              | 8                     | 2                    |
| Wine                         | 178              | 13                    | 3                    |
| Iris                         | 150              | 4                     | 3                    |
| Sonar                        | 208              | 60                    | 2                    |
| Statlog-Heart(SH)            | 270              | 13                    | 2                    |
| Breast-cancer-wisconsin(BCW) | 683              | 9                     | 2                    |

For the missing values and outliers that may exist in the dataset, the processing method of this paper is to delete the corresponding sample data.

5.2. Parameter setting
1) Using the Gini index as a criterion for constructing a split node of decision tree.
2) The number of cycles per training set is 50, a total of 250 trees are constructed, and the updated interval of decision tree is 5.
3) The number of selected features is n.
4) Initialize Temperature t (deleting algorithm t = 500, weight training algorithm t = 200), the temperature drop formula is set to $t = 0.95 \times t$.
5) The number of iteration ‘k’ is initialized to 200 at each temperature.

5.3. Evaluation criteria for classification
In order to quantitatively analyze the classification effect of random forests, the confusion matrix of the following binary classification data is considered, as shown in Table 2.

| Prediction is positive | Prediction is negative |
|------------------------|------------------------|
| Result is positive     | TP                     |
| Result is negative     | FN                     |
|                       | FP                     |
|                       | TN                     |

Where $n = TP + TN + FP + FN$ is the total number of samples. For multi class data, the incorrect data is called negative class.

\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}
\]  

Average model generalization performance: the classification accuracy Test set.
Average model accuracy: the classification accuracy of training set.

5.4. Experimental results and analysis
The purpose of the simulation experiment in this section is to compare and verify the classification performance of random forests hybrid algorithm, original random forests algorithm and weighted random forests algorithm based on SA. The three algorithms here all have tree pruning operations.
The final classification accuracy of all four algorithms is obtained and recorded as follows. First column of each comparison <initial> is the original random forests, the second column <with SA> is the weighted random forests algorithm based on SA, and the third column <with WSA> is the random forests hybrid algorithm.

Table 3. Statistical table of average result from dataset

| Dataset | Average model generalization performance | Average model accuracy |
|---------|------------------------------------------|------------------------|
|         | initial | with SA | with WSA | initial | with SA | with WSA |
| Glass   | 47.29   | 55.93   | 57.93    | 81.91   | 82.22   | 84.47    |
| ST      | 43.34   | 52.84   | 58.87    | 75.94   | 77.44   | 79.89    |
| Seeds   | 92.50   | 96.00   | 96.45    | 99.92   | 99.80   | 100.00   |
| WR      | 67.55   | 71.32   | 71.57    | 94.69   | 94.67   | 95.07    |
| Pima    | 73.20   | 77.69   | 79.10    | 99.82   | 99.64   | 99.74    |
| Wine    | 96.56   | 99.69   | 99.78    | 99.97   | 99.95   | 100.00   |
| Iris    | 93.77   | 96.44   | 94.78    | 99.89   | 99.89   | 99.96    |
| Sonar   | 82.58   | 90.73   | 91.58    | 99.94   | 99.81   | 99.90    |
| SH      | 75.66   | 81.96   | 84.82    | 97.18   | 97.46   | 98.87    |
| BCW     | 97.24   | 97.80   | 97.74    | 99.79   | 99.75   | 99.89    |
| Average dataset | **76.97** | **82.04** | **83.26** | **94.90** | **95.06** | **95.78** |

Based on table 3 above, it can be seen that:

1) from the analysis of the overall data, we can get that: comparing three algorithms’ accuracy of the total average model: initial < with SA < with WSA; from the overall average model generalization performance, the three algorithms are: initial < with SA < with WSA. In term of both the model accuracy and the model generalization performance, the random forests hybrid algorithm (with WSA) has the most significant optimization effect.

2) in terms of the total average model accuracy, the algorithm <with WSA> is better than the algorithm <with SA>, with an average of about 0.72%.

3) compared with the algorithm <with SA>, the average generalization performance of each data set is further enhanced, with an average enhancement of 1.22%, up to 6.03% respectively.

Analyze with the comparison diagram of generalization capability improvement of all data sets.

Figure 1: Average generalization performance

Figure 2: Improvement of average generalization performance

From Figure 1 and Figure 2, we can see the comparison results of the overall average generalization ability of the three algorithms. According to the comprehensive results of each data set, although there are individual data sets (such as iris) whose generalization performance is higher than that of <with WSA> in <with SA> algorithm, the generalization performance of random forests hybrid algorithm is generally improved.
6. Conclusion

Firstly, the algorithm of pruning tree and feature weighting based on SA is proposed to lay the foundation for the following hybrid algorithm.

Then combining the ReliefF algorithm and SA algorithm, we get the random forests hybrid algorithm, which can not only ensure the strength of the decision tree, but also reduce the correlation of the decision tree appropriately. The simulation results show that the hybrid algorithm has high accuracy, the best generalization ability and the best classification performance. According to the performance of classification performance in different tree scales, the reasonable tree range of decision tree is obtained synthetically, and the validity of feature weighting is verified according to the data side of pruning tree.

The hybrid algorithm also has some shortcomings: it is not suitable for all data sets, the effect of data with redundant features is obvious, and the algorithm efficiency is low when the tree size increases due to the search characteristics of SA algorithm, so how to improve these needs further research.

References

[1] Feng Z. Research on Ensemble Learning[D]. Shanghai Jiao Tong University, 2007.
[2] T. G. Dienerich. Machine Learning Research for Current Directions[J]. Al Magazine, 1997, 18(4): 97-136.
[3] Schapire R E. The strength of weak learnability[J]. Proceedings of the Second Annual Workshop on Computational Learning Theory, 1989, 5(2): 197-227.
[4] Breiman, Leo. Bagging predictors[J]. Machine Learning, 1996, 24(2): 123-140.
[5] Ho T K. Random decision forests[C]/ Document Analysis and Recognition, 1995. Proceedings of the Third International Conference on. IEEE Computer Society, 1995.
[6] Ho T K. The Random Subspace Method for Constructing Decision Forests[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998, 20(8): 832-844.
[7] Breiman L. Random Forests[J]. Machine Learning. 2001, 45(1): 5–32.
[8] Genuer R, Poggi J M, Tuleau-Malot C. Variable Selection using Random Forests[J]. Pattern Recognition Letters, 2010, 31(14): 2225-2236.
[9] Xu B, Huang J Z, Williams G, et al. Classifying Very High-Dimensional Data with Random Forests Built from Small Subspaces[J]. International Journal of Data Warehousing & Mining, 2011, 8(2): 44-63.
[10] Bloodgood M, Ye P, Rodrigues P, et al. A Random Forest System Combination Approach for Error Detection in Digital Dictionaries[C]/ The Workshop on Innovative Hybrid Approaches To the Processing of Textual Data. 2012: 78-86.
[11] Steinbrunn M, Moerkotte G, Kemper A. Heuristic and randomized optimization for the join ordering problem[J]. VLDB Journal, 1997, 6(3):191-208.
[12] Kononenko I. Estimating attributes: analysis and extension of relief[C]/ European Conference on Machine Learning. Springer, Berlin, Heidelberg, 1994.