

# Incremental Integration Algorithm based on Incremental RLID3

Hongbin Wang<sup>a</sup>, Lei Hu<sup>b</sup>, Xiaodong Xie<sup>a</sup>, Lianke Zhou<sup>a,\*</sup>, and Huafeng Li<sup>a</sup>

<sup>a</sup>College of Computer Science and Technology, Harbin Engineering University, Harbin, 150001, China

<sup>b</sup>General Office, Systems Engineering Research Institute, Beijing, 100094, China

---

## Abstract

In the research process of ID3 algorithm, some deficiencies were found. RLID3 algorithm is on the improvement of ID3 algorithm in terms of the number of leaf nodes. RLID3 algorithm uses ensemble learning method to integrate multiple incremental RLID3 model and the predictive ability of the algorithm is further improved. Incre\_RLID3 is an incremental learning algorithm that is based on a decision tree constructed by RLID3. It adjusts construction of the tree using incremental data set. The goal of this algorithm is to use the new data on the basis of the original decision tree. In order to further improve the accuracy of the algorithm, this paper proposes an ensemble algorithm PAR\_WT. The basic idea of this algorithm is to use the data set to generate multiple RLID3 decision tree. Then, the test samples are classified by each decision tree. Finally, combined with the PAR\_WT algorithm and Incre\_RLID3 algorithm, an incremental ensemble algorithm Incre\_RLID3\_ENM algorithm with incremental learning ability is obtained.

*Keywords:* incremental integration; ensemble learning; incremental learning; decision tree

(Submitted on October 12, 2018; Revised on November 11, 2018; Accepted on December 23, 2018)

© 2019 Totem Publisher, Inc. All rights reserved.

---

## 1. Introduction

In recent years, the research on incremental ensemble method is relatively hot. Some incremental ensemble methods that have emerged include the typical Learn++ algorithm [1-2] and selective ensemble method [3-5]. These methods play a leading role in the research of incremental integration. The effectiveness of these methods is verified by a lot of experiments and applications.

## 2. The Deficiency of RLID3 Algorithm

In actual production, the data are usually produced in an irregular time. In this case, the data analysis system is required to learn the information incrementally; it must learn from new data while not forgetting about the model learned before. This learning method is called incremental learning. Incremental learning [6] is a learning algorithm that can be used to train the model incrementally. Incremental learning can be applied in cases of uncertain time when new data arrives.

Incremental algorithm is a hot research direction [7]; in recent years, there are many experts and scholars to study it. At present, the research on incremental algorithm is to improve and transform the existing classical classifier, so that it becomes a classifier with incremental function. For example, ID4 algorithm [8] is an improved ID3 algorithm. When the incremental data comes, ID4 algorithm can adjust the decision tree generated by the ID3 algorithm. In this paper, according to the above ideas of ID4 algorithm, the decision tree generated by RLID3 algorithm is adjusted according to the incremental data; Incre\_RLID3 algorithm with incremental learning function is obtained.

After years of experiment and application, it is proved that ensemble learning is a kind of machine learning method worth studying [9-10]. Ensemble learning can make up for the shortcoming of the single classifier trained by the data set, which is difficult to fit the actual data well. Ensemble learning uses multiple classifiers to train data sets in some way, and the classification results of these classifiers are used as the final classification results.

\* Corresponding author.

E-mail address: [zhoulianke@hrbeu.edu.cn](mailto:zhoulianke@hrbeu.edu.cn)

It is found that the ensemble learning method of Bagging [11] can choose the incremental classifier as the base classifier. Integration of these incremental classifiers can be an incremental integration algorithm. This incremental ensemble algorithm has good accuracy and incremental learning ability.

### 3. Incremental RLID3 Algorithm Incre\_RLID3

According to the data set, RLID3 algorithm selects the attribute of the maximum decision tree optimization ratio; this attribute is used as the split attribute. Then, according to the value of this attribute, the data subset is divided, and decision tree optimization ratio of each subset is calculated recursively.

Among them, decision tree optimization ratio is the ratio of the information gain of the current attribute and the number of leaves of the decision tree, which is generated by the current node. The formula is as Equation (1).

$$DTOR(S, A) = \frac{Gain(S, A)}{LeafNum(S, A)} \quad (1)$$

In the above formula,  $DTOR(S, A)$  presents decision tree optimization ratio,  $Gain(S, A)$  represents the information gain of attribute  $A$  on the data set  $S$ , and  $LeafNum(S, A)$  represents the number of leaves of a decision tree that is formed using a data set  $S$  and attribute  $A$ .

The concepts used in the algorithm are introduced. Sample feature code is a string used to store data sets; this string represents a sample data. It can obtain the value of each attribute from .arff file in Weka. The value of each attribute is labelled as 0, 1, ...,  $(n - 1)$  form. Among them,  $n$  is the number of attributes. Each column represents an attribute, and the order of attributes is the same as that of attributes in .arff file in Weka. Sample feature code set refers to a plurality of sample signatures in a form of organization.

The data set of Table 1 that is stored as sample feature code set is represented as the following: 000, 000, 101, and 201.

In the outlook attribute, the value of sunny is 0, the value of overcast is 1, and the value of rainy is 2. The other attribute marking way is the same as the outlook attribute.

The samples of the data set is added to the sample feature code set by reading each sample data in the data set. Examples of data collection are shown in Table 1.

Table 1. Examples of data collection

No.	Outlook	Humidity	Play
1	sunny	high	no
2	sunny	high	no
3	overcast	high	yes
4	rainy	high	yes

#### 3.1. Incre\_RLID3 Algorithm

Based on the analysis of the last section, it is found that the incremental base classifier is the key problem of the algorithm. In this paper, we use Incre\_RLID3 algorithm as the base classifier. The core idea of Incre\_RLID3 algorithm is to dynamically adjust decision tree generated by RLID3 algorithm using the incremental data.

Incre\_RLID3 algorithm can be divided into the following two stages.

##### (1) Initial stage

The initial stage is to get the training data set. Before the start of the stage, there is no classifier training. In this stage, RLID3 algorithm is used to train the training set, and the decision tree model is obtained. After the end of this stage, the training model of decision tree is obtained.

##### (2) Increment stage

The incremental phase happens after completing initial stage and a period of time; it is able to use the incremental data.

Before the start of the stage, there is already a good classifier in the initial stage of training. In this stage, Incre\_RLID3 algorithm based on the incremental data set is used to adjust the initial stage of the decision tree trained model. After the end of this stage, the decision tree model based on the incremental data set is obtained.

The initial stage algorithm of Incre\_RLID3 algorithm uses the RLID3 algorithm. Increment stage of Incre\_RLID3 algorithm is shown below. Before using the incremental data set, each data of incremental data set can be classified according to the decision tree of initial stage. The data can be removed from the incremental data set if it can be classified correctly. The incremental data set used by algorithm 1 refers to the set after the above treatment.

---

**Algorithm 1** Update RLID3Tree (Root node of a RLID3 decision tree, Incremental data set)

---

```

1. Input: Root node of a RLID3 decision tree, Incremental data set
2. Output: adjusted decision tree
3. declare a variable curNode, decision tree is used to represent the nodes in the traversal algorithm
4. set the curNode to the root node of the input RLID3 decision tree
5. adding the incremental data set to the sample feature code of the curNode
6. if (The classification of each sample is the same in the sample feature code set of the curNode) { //All samples are divided into similar
7.     curNode is set to a leaf node, namely: the curNode attribute is set to null
8.     class attribute value of curNode is set to the classification of the sample feature code set of the curNode
9. }else{//The samples belong to the category is not exactly the same
10.     if (curNode is a leaf node) { //The attribute of the curNode is null
11.         using the feature sample code set of the curNode, according to the formula (1), choosing the attribute with the maximum value of decision tree
            optimization ratio, the attribute of the curNode is set to the attribute
12.         if (decision tree optimization ratio of the attribute of the curNode is 0) {
13.             set the curNode to the leaf node, namely: the attribute of the curNode is set to null
14.             class attribute value of the curNode is set to the most class of sample feature code set
15.         }else{//The maximum value of decision tree optimization ratio is not 0
16.             the attribute of the curNode as the attribute of the root node, call the RLID3 algorithm makeRLID3Tree (Sample feature set of the
            curNode, the attribute of the curNode) to build RLID3 tree}
17.         }else{//the curNode is a branch node
18.             using sample feature set of the curNode. According to the formula (1), selecting the attribute of the maximum value of decision tree optimization
            ratio, denoted as A
19.             if(decision tree optimization ratio of A is 0){ //
20.                 the curNode is set to leaf node
21.                 class attribute value of the curNode is set to the highest value in sample feature code set of the curNode
22.             }else{//decision tree optimization ratio of A is not 0
23.                 The attribute of the curNode is the same as A.
24.                 while(sample: sample feature code set of the curNode){
25.                     add sample to the childrenTrainData[sample partition attribute's value number] array
26.                 }
27.                 for (j = 0; j < the number of the curNode's available; j++)
28.                     the node that corresponds to the first j value is used as the root node of the sub tree,
29.                     call RLID3 algorithm,
30.                     updateRLID3Tree(the node that corresponds to the first j value, childrenTrainData[j]), subtree traversal
31.                 }
32.             }else{//the attribute of curNode is not the same as A
33.                 the node that corresponds to the first j value is used as the root node of the sub tree,
34.                 call RLID3 algorithm,
35.                 updateRLID3Tree(the node that corresponds to the first j value, childrenTrainData[j]), subtree traversal
36.             }
37.         }
38.     }
39. }
```

---

### 3.2. Ensemble Learning based on Parallel Weighted

In this paper, parallel weighted algorithm is a combination of multiple base classifier algorithm, which belongs to ensemble learning algorithm. Parallel weighted refers to the training of multiple base classifiers in parallel, and retains the trained classifier classification accuracy. The accuracy of classification is obtained according to the Equation (3). In this section, RLID3 algorithm is used to generate decision tree, which is used as the base classifier.

Define 1 (weight recorder) using a two-dimensional matrix represents weight recorder. It is used to represent the relationship between the base classifiers and the classified categories. For two dimensional matrix, Line represents base classifier and column represents classification of data sets. The number of rows in a two-dimensional matrix is set to  $m$ , and

the number of columns is set to the value  $n$  of class attribute of the data set. Using  $A$  to represent the two-dimensional matrix,  $a_{ij}$  represents the weight that the  $i$  base classifier classifies a sample into category  $j$ . The structure of a two-dimensional matrix can be expressed as follows.

$$A = \begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix}$$

The weight recorder is used in the PAR\_WT algorithm, as shown below.

---

**Algorithm 2** Weight recorder

---

**Input:** data set, trained classifier set

**Output:** weight recorder

1. Initialization of the weight recorder, making  $a_{ij}=0$
  2. for (instance: data set) {
  3.     for (classifier: classifier set) { //classifier represents the  $i$  classifier
  4.          $j = \text{classifier.classify}(\text{instance})$  //  $j$  represents the classification results of the classifier for the sample
  5.          $a_{ij} += \text{classifier.getWeight}()$  //  $a_{ij}$  plus the weight of the  $i$  classifier
  6.     }
  7. }
- 

Ensemble learning algorithm based on parallel weighted is divided into two parts.

(1) Calculate the weights of each base classifier

In this part, we need to train the multiple base classifiers in parallel to get the accuracy of the classifier. The formula of the classification accuracy is shown in Equation (2).

$$\text{precision} = \frac{\text{the number of samples that can be correctly classified}}{\text{the numbers of samples}} \quad (2)$$

The base classifier's accuracy Equation (2) is brought into the Equation (3).

$$\text{weight} = \log_2 \left( \frac{1}{1 - \text{precision}} \right) \quad (3)$$

In the above Equation (3), weight represents the weight of the base classifier, and precision indicates the classification accuracy. Get a weight with the accuracy of the base classifier as the weight of the base classifier.

(2) Using weights and base classifiers integrates the classification results of the classifiers

i. Using the test set, the base classifier and the weights are assigned to weight recorder.

ii. The sum of each column in a two-dimensional matrix is calculated; the maximum value is used as the classification result.

Bagging is a common way of ensemble learning. Bagging can be trained in parallel to multiple base classifiers, but it simply uses the voting method to comprehensively classify the voting results. The results are used as the prediction results of the classifier. In this paper, PAR\_WT algorithm is used to get the weight of the Equation (3). The classification results of the weights are used to classify the classifiers, and the classification results are obtained as the final classification results.

PAR\_WT algorithm has two stages. The first stage includes the training base classifier and the weight of the base classifier. The second stage is to combine all the base classifiers to predict the results. Because the main research of this section is ensemble learning, incremental learning is not added to this section. This section selects non-incremental RLID3 algorithm to train the base classifier.

The steps of the PAR\_WT algorithm are shown below.

---

**Algorithm 3-1** PAR\_WT algorithm

---

**Input:** data sets, the number of base classifiers

**Output:** base classifier and its weight

1. Parallel implementation of the following procedure { //T times T said the number of base classifiers.
  2. Being randomly divided into 10 parts of data.
  3. Taking out the data set of 9 parts as the training set of the i base classifier.
  4. Taking out the data set of 1 parts as the test set of the i base classifier.
  5. Using RLID3 algorithm to train the i based classifier.
  6. Using the formula (2), the classification accuracy of the i base classifier is calculated.
  7. Using the formula (3), calculating the weight of the i base classifier.
  8. }
- 

The second step of the PAR\_WT algorithm is shown as follows.

---

**Algorithm 3-2** ParallelAndWeightSecond (pre-classification sample data)

---

**Input:** data sets, the number of base classifiers

**Output:** base classifier and its weight

1. Initialization of the weight recorder, making  $a_{ij}=0$
  2. for (classifier: classifier set) { //classifier represents the i classifier
  3.      $j = \text{classifier.classify}(\text{instance})$  //j represents the classification results of the classifier for the sample
  4.      $a_{ij} += \text{classifier.getWeight}()$  //  $a_{ij}$  plus the weight of the i classifier
  5. }
  6. Calculating the sum of each column of weight recorder
  7. The maximum value of each column is used as the classification results of pre classified data.
- 

#### 4. Incremental Integration Algorithm based on Incre\_RLID3 Algorithm

By combining Incre\_RLID3 algorithm and PAR\_WT algorithm, an incremental integration algorithm based on RLID3 is proposed. The main task of this section is to use the Incre\_RLID3 algorithm to train the base classifier in PAR\_WT algorithm. PAR\_WT algorithm is divided into two stages. It leads to the incremental integration algorithm of initial stage and incremental stage, which is divided into two stages.

Initial stage uses PAR\_WT algorithm. At this stage, RLID3 algorithm is used as the training base classifier. Initial stage is divided into 2 stages.

##### (1) Initial stage

Initial stage uses PAR\_WT algorithm. At this stage, RLID3 algorithm is used as the training base classifier. Initial stage is divided into 2 stages, which are first stages and second stages, respectively. These two stages are the same as the two stages of the algorithm 2 PAR\_WT algorithm.

##### (2) Incremental stage

Incremental stage is divided into 2 stages, which are first stages and second stages respectively. Incremental integration algorithm is based on the incremental stage of PAR\_WT algorithm. The second stage is the same as the PAR\_WT algorithm, but the first stage is slightly different. In the first stage, we need to input the parameters for initial stage of all the base classifier, and PAR\_WT algorithm does not need to enter this parameter.

The algorithm steps of the incremental stage of Incre\_RLID3\_ENM algorithm are shown in Algorithm 4.

---

**Algorithm 4** RLID3IncreEnsemble (incremental data set, initial stage of all base classifiers)

---

**Input:** incremental data set, initial stage of all base classifiers

**Output:** adjusted base classifiers

1. Perform the following procedure T times in parallel { //T represents the number of base classifiers
  2. Call algorithm 1, using the incremental data set to adjust the i base classifier to obtain the adjusted i base classifier
  3. Test set of the i base classifier of initial stage is used as the test set of the adjusted i base classifier
  4. Using the formula (2), the accuracy of the i base classifier is calculated.
  5. Using the formula (3), the weight of the i base classifier is calculated.
  6. }
-

The steps of the first stage are shown below. The steps of the second stage of the algorithm are the same as the second stage of PAR\_WT algorithm.

## 5. Experiment Results and Analysis

The classification problem is studied in the paper. The experiment was carried out using 10 data sets related to classification in the UCI database. The data sets contain a wide range of areas: life, computer, social and games. The basic information of the data sets is shown in Table 2.

Table 2. Basic information of the data sets

Name	Instances	Attributes	Classifications	Missing Values
Letter	20000	16	26	No
HIV	6590	8	2	No
Nursery	12960	8	5	No
Tic-Tac-Toe	958	9	2	No
Connect-4	67557	42	3	No
Chess	3196	36	2	No
Splice	3190	61	3	No
Mushroom	8124	22	2	Yes
Lymph	148	18	4	No
Breast-w	699	10	2	Yes

Because the algorithm can deal with the samples that must be non-missing value, Mushroom and Breast-w are processed. There are two methods to pre-process the data set with missing values. The first is to remove the stalk-root from the data set; the second is to remove the missing data from the data set. In the Mushroom data set, only the stalk-root attribute misses values, and the proportion of missing data in the data set is large. So, Mushroom is used after removing the stalk-root attribute. After processing, data sets do not have missing value of sample, and small and categorical attributes. The attribute value is not enumerable phenomenon.

### 5.1. Evaluation Method

In this paper, precision is used as the index evaluation of algorithm; the formula is shown as Equation (4).

$$precision = \frac{\text{the numbers of samples that can be correctly classified}}{\text{the numbers of samples}} \quad (4)$$

### 5.2. Result and Analyze

The purpose of this section is to record the classification accuracy of Incre\_RLID3 algorithm. In this paper, the data set is divided into 4 parts. The part of the initial training set accounts for 40% of the total sample size of the data set, test set accounts for 10%, the first increment accounts for 30%; and the second increment accounts for 20%.

The experimental procedure is as follows.

(1) Use initial training set as a parameter to call initial stage of Incre\_RLID3 algorithm to train the classifier. The accuracy of the classifier is calculated using the test set.

(2) Incremental training is performed by using the first increment data set and the classifier as initial stage of Incre\_RLID3 algorithm. The accuracy of the classifier is calculated using the test set.

(3) Incremental training is performed by using the second increment data set and the classifier obtained at the last stage as the parameters of Incre\_RLID3 algorithm. The accuracy of the classifier is calculated using the test set.

In order to visually show the experimental results of Incre\_RLID3 algorithm, the experiment results are shown in Figure 1.

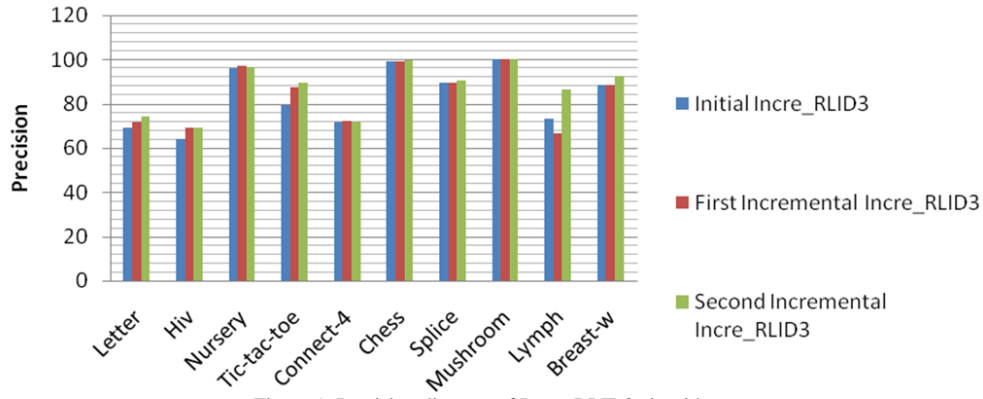


Figure 1. Precision diagram of Incre\_RLID3 algorithm

The results show that the classification accuracy of the next stage of Incre\_RLID3 algorithm is not necessarily higher than that of the previous stage. The accuracy of each stage will not use the entire data set than direct training and testing of the RLID3 algorithm with high accuracy. The advantage of using Incre\_RLID3 algorithm is that it is able to incrementally train data.

From parallel weighted, it can be seen that the parallel weighted experiment needs to determine the kind of classification algorithm and the number of base classifiers. In the experiment, 3 base classifiers based on the RLID3 algorithm are used.

Ensemble learning needs to consider how to split the data set efficiently and reasonably. The way in which the data set is processed is as follows.

- 01: The data set is divided into data set 1 and data set according to the ratio of 9:1.
- 02: Perform the following procedure 3 times in parallel {
- 03: The samples of 9/10 is extracted randomly from the data set 1; these samples are used as the training set of base classifier  $i$
- 04: The remaining 1/10 samples of the data set 1 are taken as the test set of base classifier  $i$
- 05: }
- 06: The data set 2 is used as the test set of the integrated classifier.

Using the above method to process data sets, it is ensured that the test set has no intersection with the training set, and the base classifier can be trained with more data. The training set is randomly extracted. Generally, a training set of 3 base classifiers is not exactly the same.

Figure 2 shows the accuracy of the integrated classifiers.

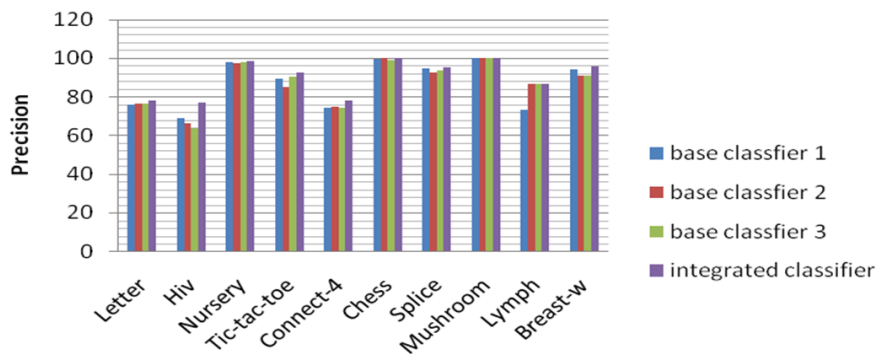


Figure 2. Precision diagram of PAR\_WT algorithm

From the experimental result, it can be seen that after the parallel weighted algorithm is processed, the accuracy of integrated RLID3 algorithm is higher than or equal to the base classifier's accuracy. It shows that the prediction result of ensemble classifier is better than the single classifier.

Incremental integration experiment is a combination of incremental and ensemble learning. In the experiment, the data set is divided into the initial data set, the incremental data set for the first increment experiment, the incremental data set for the second increment experiment and the test set with 4 disjoint sets of data. In this section of the experiment, 3 base classifiers are used. Incre\_RLID3 algorithm are used to train the classifier.

The experimental procedure using the data set to carry out this section is shown below.

(1) Samples are randomly selected from the data set: 40% for the initial data set, 30% for the first incremental data set, 20% for the second incremental data set, and 10% for the test at each stage of the algorithm.

(2) Initial data set is processed 3 times in parallel; each time the initial data set randomly draws 9/10 of the samples as the training set of base classifier  $i$ . The remaining 1/10 of the initial data set is used as the test set for base classifier  $i$ .

(3) 3 test sets obtained from step (2) are used as the test set of the first increment experiment and the second increment experiment. According to the Formula (3), the weights of the 3 base classifiers are calculated.

Figure 3 shows the classification results of the ensemble classifier in the initial stage, the first increment stage and the second increment stage.

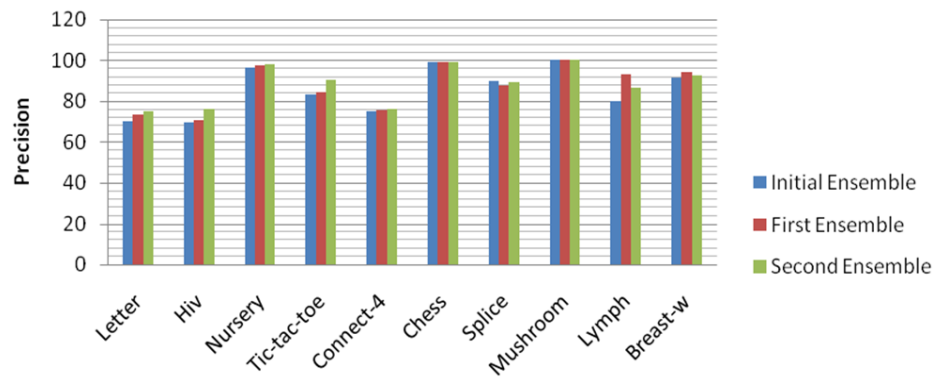


Figure 3. Precision of each stage of Incre\_RLID3\_ENM

From the experimental result, it can be seen that Incre\_RLID3\_ENM algorithm has a good precision. And the algorithm can process the incremental data.

## 6. Conclusion

At present, the research on incremental integration algorithm is very hot; it is mainly focused on the improvement and application of incremental integration algorithm. Incre\_RLID3 algorithm is proposed in this paper. The algorithm is based on RLID3, which does not have the increment function. Although the bagging method can be used to train a number of models in parallel, it simply uses the voting method to synthesize the voting results of each model. The results are used as the final prediction results of all models. PAR\_WT algorithm will use the Formula (2) to get the weight, use weights to predict the results of all models, and the obtained prediction results are used as the final prediction results of the model. Incre\_RLID3 and PAR\_WT are combined, and the Incre\_RLID3\_ENM algorithm is obtained. Further research can consider how to further improve the speed of incremental learning algorithm construction, as well as to improve the form of parallel construction. Further research can also consider the possibility of further reducing the spatial storage of incremental integration algorithms.

## Acknowledgement

This work was funded by the National Natural Science Foundation of China under Grant (No. 61772152 and No. 61502037), and the Basic Research Project (No. JCKY2016206B001, JCKY2014206C002 and JCKY2017604C010).

## References

1. C. L. Cassano, A. J. Simon, L. Wei, C. Fredrickson, and Z. H. Fan, "Use of Vacuum Bagging for Fabricating Thermoplastic Microfluidic Devices," *Lab on a Chip*, Vol. 15, No. 1, pp. 62, 2015



2. Y. Chen, J. Cao, S. Feng, and Y. Tan, "An Ensemble Learning based Approach for Building Airfare Forecast Service," in *Proceedings of IEEE International Conference on Big Data*, pp. 964-969, 2015
3. B. Gu, V. S. Sheng, Z. Wang, D. Ho, S. Osman, and S. Li, "Incremental Learning for v-Support Vector Regression," *Neural Networks*, Vol. 67, No. C, pp. 140, 2015
4. Y. Guo, L. Jiao, S. Wang, S. Wang, F. Liu, K. Rong, et al., "A Novel Dynamic Rough Subspace based Selective Ensemble," *Pattern Recognition*, Vol. 48, No. 5, pp. 1638-1652, 2015
5. B. Han, B. He, N. Rui, M. Ma, S. Zhang, M. Li, et al., "LARSEN-ELM: Selective Ensemble of Extreme Learning Machines using LARS for Blended Data," *Neurocomputing*, Vol. 149, pp. 285-294, 2015
6. L. Hu, C. Shao, J. Li, and H. Ji, "Incremental Learning from News Events," *Knowledge-based Systems*, Vol. 89, No. C, pp. 618-626, 2015
7. N. Li, Y. Jiang, and Z. H. Zhou, "Multi-Label Selective Ensemble," in *Proceedings of International Workshop on Multiple Classifier Systems*, pp. 76-88, 2015
8. D. Peressutti, W. Bai, T. Jackson, M. Sohal, A. Rinaldi, D. Rueckert, et al., "Prospective Identification of CRT Super Responders using a Motion Atlas and Random Projection Ensemble Learning," in *Proceedings of International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 493-500, 2015
9. J. C. Schlimmer and D. H. Fisher, "A Case Study of Incremental Concept Induction," in *Proceedings of National Conference on Artificial Intelligence*, pp. 496-501, Philadelphia, Pa, August 11-15, 1986
10. X. Z. Wang, H. J. Xing, Y. Li, Q. Hua, C. R. Dong, and W. Pedrycz, "A Study on Relationship Between Generalization Abilities and Fuzziness of Base Classifiers in Ensemble Learning," *IEEE Transactions on Fuzzy Systems*, Vol. 23, No. 5, pp. 1638-1654, 2015
11. M. S. Zia and M. A. Jaffar, "An Adaptive Training based on Classification System for Patterns in Facial Expressions using SURF Descriptor Templates," Kluwer Academic Publishers, 2015

**Hongbin Wang** received his PhD degree in computer application technology from Harbin Engineering University in 2010. He is currently an associate professor in the College of Computer Science and Technology of Harbin Engineering University. His research interests include information management, data integration, data space, semantic web, ontology and information system design.

**Lei Hu** received his Master's degree in computer application technology from Wuhan University in 2004. He is currently a senior engineer of SERI. His research interests include information management, systems engineering, and system integration.

**Xiaodong Xie** is now studying for a Bachelor's degree in computer application technology at the Harbin Engineering University in 2018. His research interests include deep learning, big data, AI, machine learning.

**Lianke Zhou** received his PhD degree in computer architecture from Harbin Institute of Technology in 2011. He is currently a lecturer in the College of Computer Science and Technology of Harbin Engineering University. His research interests include data visualization, dataspace, distributed computing and mobile computing.

**Huafeng Li** received his master degree in software engineering from Harbin Engineering University in 2016. His research interests include information management, data integration, and data classification.