ISBN 978-981-14-1684-2
Proceedings of 2019 the 9th International Workshop on Computer Science and Engineering
(WCSE 2019 SUMMER)
Hong Kong, 15-17 June, 2019, pp. 630-637
doi: 10.18178/wcse.2019.06.093

A Dynamic Integrated Classification Algorithm Based on Big Data Environment

Dan Ma[1], Ji-chun Jiang[1+], Wei Wang[2]

[1] College of Computer Science &Technology, GuiZhou University Guiyang, Guizhou Province China
[2] Guizhou Gas Group Corporation Ltd, Guiyang, Guizhou Province China

Abstract. With the developing of big data application, classification algorithm has been expanded to distributed datasets from the single dataset. So a dynamic integrated classification algorithm based on big data environment was proposed. This algorithm gain integrated classifiers of high classification accuracy for each local dataset, and dynamically generate the recognition model according to the distribution characteristics of local samples to be tested. In the application process, after numerous new sample data join the datasets, the classifier performance will drop gradually. By aiming at the above problem, this algorithm will retrain the classification model in the dynamic expansion process of datasets. According to the experimental results, the algorithm proposed in this paper has high classifier training performance and classification accuracy. At the same time, it also possesses high adaptive capacity when faced with dynamically changing distributed datasets.

Keywords: Classification Algorithm; Integrated Algorithm; Big Data; DIC

1. Introduction

With the implicit values of big data calling more people's attention, classification algorithm has been expanded to distributed datasets from the single dataset. Research on classification technology in big data mainly focuses on two aspects. One is how to achieve fast classification in the big data environment, and the other is how to improve the performance of integration classification. Zhang mingwei et al.[1]established a distributed assistant association classification model in the big data environment. The model solves the problem of how to build distributed datasets and improve the performance of classification based on dynamic datasets. Chen xuebin et al.[3] proposed a parallel classification hybrid algorithm for big data. This algorithm improves the classification efficiency of massive data, and the accuracy of classification, reduces the system overhead. Wang yanbin et al.[4] proposed the integrated method of gradient optimization decision tree. This method proposes to integrate classification accuracy. Zhang feifei et al.[5] proposed an multiple classifiers on the basis of stratification to improve the improved over-sampling unbalanced data integration classification algorithm. The algorithm use AdaBoost technology to deal with the unbalance data to improve the accuracy of classification.

A dynamic integrated classification algorithm based on big data is proposed in this paper. The model is shown in Figure. 1.It covers three parts: generating the classification model in big data environment; dynamically integrating the recognition model; and re-training the classification model. This algorithm can dynamically generate a recognition model according to the distribution characteristics of local samples to be tested. It can gain integrated classifier of high classification accuracy for every local dataset.

+ Corresponding author. Tel.: +86 13339607116
E-mail address: 895439149@qq.com

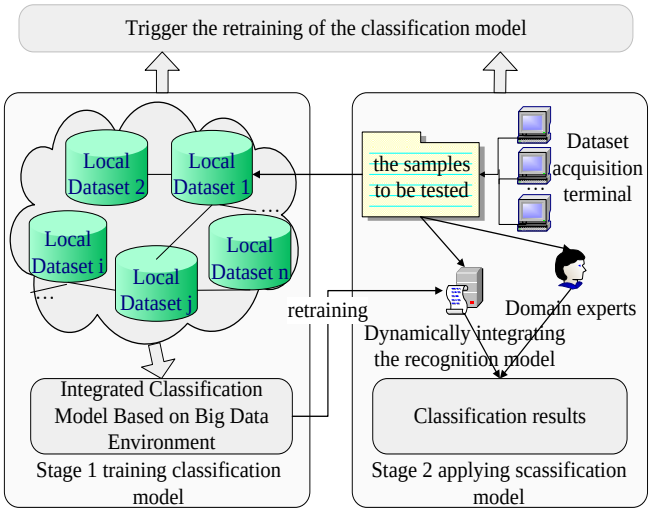

Fig. 1: integration classification model in big data environment

2. Construction of Dynamic Integrated Classifier

This algorithm has inherited the framework of the classic integrated classification technique boosting, and can dynamically generate a recognition model according to the distribution characteristics of local samples to be tested.

2.1. Setting of re-training conditions for the classification model

In big data environment, as many new sample data join the datasets continuously, the classifier performance will drop gradually, so a new classification model should be generated through re-training. The conditions to trigger re-training of classification model are as follows. (1) In the local datasets, suppose that the time to complete previous retraining and generate the classification model is t, the classification model can be re trained after the certain time interval Δt. (2) Within $t+\Delta t$, and the number of classified samples reaches the threshold value N or the error rate of classification reaches the threshold value ER, then the classification model will be re-trained. The classification model can be re-trained when the datasets presents dynamic expansion changes with time, so as to improve the classification accuracy[1].

2.2. Design of dynamic integrated classification algorithm

The following two problems should be considered when this algorithm is designed. (1) In big data environment, the samples in various local datasets have their own distribution characteristics. Besides, the similarity level between various local datasets in category distribution also varies. When the classification model is trained for different local datasets, we should utilize not only current dataset as the main training samples to generate the classification model, but also training samples of other local datasets as the supplement. (2) When the local samples to be tested are classified, the problem is how to dynamically generate the final classification recognition model according to the difference between samples to be tested and various local training sets.

The category distribution of training samples at each local dataset is different. Therefore, it is considered that the current dataset is the main dataset, and other datasets are the auxiliary, so as to train the dataset of each data sources and obtain the classification model suitable for the distribution characteristics of samples. Let global datasets D be $\{D_0,D_1,...,D_{n-1}\}$, where D_i is the ith local dataset. Let the degree of difference in distribution be $\{\rho_0,\rho_1,...,\rho_n\}$, where ρ_j represents the degree of difference in the distribution of samples categories between local dataset D_j and local dataset D_i. For example, distribution characteristics of samples categories between D_j and D_i are closer, then $\rho_j>\rho_k$. The classification is auxiliary application method in big data environment. So the value of ρ_j is provided by domain experts according to experience. Suppose a global datasets $\{D_0,D_1,D_2\}$. Put more than half of the samples of D_0 into the training set, and extract training

samples from other local datasets in proportion. The training sample structure built for D_0 is shown in table 1. Let $\rho_0=1$, $\rho_1=0.8$,$\rho_2=0.6$.

Table 1: The training set for D_0

Id	X	Y	Z	Category	D_i	ρ
0001	x_1	y_4	z_3	C_1	D_0	1
0002	x_3	y_4	z_1	C_1	D_0	1
0003	x_2	y_1	z_2	C_3	D_0	1
0004	x_4	y_3	z_1	C_2	D_0	1
1001	x_2	y_3	z_1	C_2	D_1	0.8
2002	x_2	y_4	z_4	C_1	D_2	0.6
1003	x_2	y_2	z_4	C_1	D_1	0.8
2004	x_1	y_3	z_2	C_2	D_2	0.6
2002	x_4	y_2	z_1	C_2	D_2	0.6

In the process of training the classification model for the local dataset D_0, C4.5 decision tree is used to generate the sub-classifier for a single iteration. The parameters of sample size in the formula for Entropy and Gain need to be redefined as the sum of weights of corresponding samples, so that the algorithm can directly process weighted samples[2]. This paper uses the Adaboost to iteratively train multiple sub-classifiers. In the (i+1)th iteration, the weight of each correctly classified training sample in the ith iteration needs to be recalculated. Since the training samples in the big data come from different local datasets, the algorithm need to add the parameter ρ(the degree of difference in categories distribution between local datasets) into the process of training sub-classifier. So the weight of correctly classified training samples is:

$$\omega_j(i+1)=\frac{\omega_j(i)\times\rho\times error(M_i)}{1-error(M_i)} \tag{1}$$

In which, $\omega_j(i)$ represents the weight of a correctly classified sample in the ith iteration, and error(M_i) represents the error rate of sub-classifier M_i, the formula is as follows:

$$error(M_i)=\sum_{1}^{d}\omega_j(i)\times error(X_j) \tag{2}$$

where error(X_j) represents the rate of misclassification of sample X_j. If the sample is misclassified, error(X_j) is 1, else the value is 0. If the error of sub-classifier M_i is over 0.5, it is discarded. When the weights of all correctly classified samples have been updated, the weights of all samples must be normalized. After the iteration is completed, a set of sub-classifier sequence $\{M_1,M_2...M_k\}$ for D_0 is obtained in the global datasets.

The rule tree of the sub-classifier M_i is shown in figure 2. In this tree, branches represent decision attributes and leaves represent categories. Suppose that the impact factors of decision attributes in sub-classifier M_i is θ. The attributes of the same layer have the same θ, as well as being reorganized $\theta_l>\theta_l+1$ ($(1\le l\le h-1)$), l is the number of layers that nodes are in, and h is the depth of the rule tree.

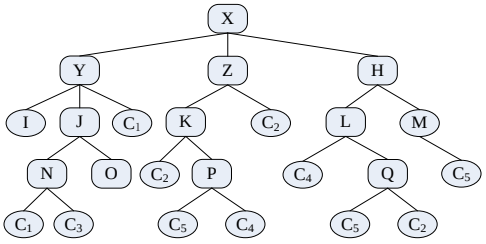

Fig. 2: The rule tree of the sub-classifier M_i

In order to distinguish the difference in attribute distribution between testing set and the training sets extracted from different local datasets, the attribute difference coefficient (ADC) is used. Each training set of sub-classifier M_i is adjusted, and each attribute of training sample that is not stored in the rule tree T_i is clipped out. The adjusted sample set is $t_m=\{(t_{1\ 1},t_{1\ 2},...,t_{1\ n}),\ (t_{2\ 1},t_{2\ 2},...,t_{2\ n}),...,(t_{m\ 1},t_{m\ 2},...,t_{m\ n})\}$. Let $t_{i\ j}(1\le i\le m,1\le j\le n)$ is a sample in the training set after cutting, where m represents the number of samples, and n

represents the number of remaining attributes after cropping. The attributes of the samples S to be tested are similarly clipped, and only the attributes appearing in the rule tree T_i are retained. Countj($t_{i+1\,j}$,S_j) is used to compute the difference between the jth attribute distribution of each sample in the training set tm and the jth attribute distribution of the sample to be tested S. The formula is:

$$Count_j(t_{i+1\,j}, S_j) = \begin{cases} Count_j(t_{i\,j}, S_j) + 1, S_j \neq t_{i+1\,j} \\ Count_j(t_{i\,j}, S_j), S_j = t_{i+1\,j} \end{cases} \tag{3}$$

The formula of ADC is:

$$ADC(T,S) = \frac{\sum_{j=1}^{n}(\sum_{i=1}^{m} Count_j(t_{i\,j}, S_j) * \frac{\theta_j}{m})}{n} \tag{4}$$

The threshold value e of ADC can be set according to the experience of domain experts. When ADC> e, it indicates that the attribute distribution of the test samples differ greatly from the training samples, and the sub-classifier cannot be selected into the final classification model, In this way, a set of sub-classifier sequences are selected dynamically for the testing samples.

2.3. The process description of dynamically integrating classification algorithms

The DIC algorithm is divided into two stages: training and decision-making. The local dataset D_0 is taken as an example to describe the two stages of the algorithm. The training stage of DIC algorithm is described as follows:

(1)Initialize the weight of each sample in D, $\omega_j(1)=1/d$;

(2)for i=1 to k do

(3) according to the weight, the training set D_i is obtained by sampling from D;

(4) the sub-classifier M_i on D_i was produced by training the weighted training set with C4.5 decision tree algotiyhm;

(5) calculate error(M_i)

(6) if error(M_i)≥0.5 then

(7) the sample weight is re-initialized to 1/d;

(8) goto (3);

(9) endif

(10) according to the rule tree corresponding to sub-classifier M_i, the decision attribute of training set D_i was clipped and produce the sample set t_m(i);

(11) For each correctly classified sample in D_i do

(12) Calculate the weight of the sample $\omega_j(i+1)$;

(13) endfor

(14) normalize the weight of each sample.

(15)endfor

When a large number of new sample data is added to the dataset, the performance of the classifier will gradually decline, which requires retraining to generate a new integrated classification model after a certain interval. If the number of classified samples or the error rate reaches the threshold, it is necessary to retrain the integrated classification model.

When classifying a test sample in the local datasets, It constitutes the final recognition model according to dynamically extracting a set of sub-classifier sequences{M_1,M_2,…,M_t} from the integration classifier {M_1,M_2,…,M_k}. When predicting categories, voting rights are assigned to each sub-classifier, and the lower the error rate is, the higher the voting weight will be assigned to the sub-classifier. The categories of the testing samples will be gained by weighted voting.

The decision-making stage of DIC algorithm is described as follows:

(1) Initialize the weight of each class to 0;

(2) for i=1 to k do

(3) the value of ADC which is the difference in attribute distribution between the sample S to be tested and the sample attributes of the sub-classifier M_i is calculated;

(4) if ADC >e then

(5) Sub-classifier M_i is excluded from the final recognition model;

(6) Goto (3);

(7) endif

(8) Sub-classifier M_i was included in the final recognition model for sample S;

(9) endfor

(10) for i=1 to t do

(11) Calculate the voting weight of sub-classifier M_i;

$$\omega(M_i) = \log \frac{1-error(M_i)}{error(M_i)} \tag{5}$$

(12) produce class identifer

$$c = M_i(S) \tag{6}$$

(13) $\omega(M_i)$ is added to the weight $\omega(c)$;

(14) endfor

(15)return the class identifer with the maximum weight $\omega(c)$.

3. Experimental Results and Analysis

In order to verify the validity of the dynamic integrated classification algorithm DIC under big data environment proposed in this paper, an analysis was made on the classification accuracy of the model and the adaptive capacity of the classification model when faced with distributed dynamic datasets. The sample data applied in this experiment came from user behavior log data provided by China Internet data mining competition in 2013. the datasets are internet behavior log for 4 weeks created by users randomly selected in 30 provinces and cities of China. Every province, city were treated as an acquisition point of sample datasets, including about 1000K classification samples. The sample size distribution at each dataset is presented in Figure. 3.

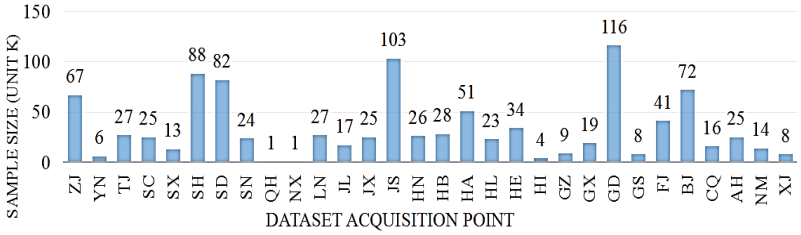

Fig. 3: Distribution diagram of sample sizes at each data acquisition points

3.1. A analysis on the classification accuracy of DIC algorithm

In order to better analyze the performance of DIC algorithm, a comparison was made on the classification accuracy of DIC algorithm and various popular classification algorithms in the single local dataset. Then the dataset was expanded to the global datasets dominated by current dataset and assisted by other datasets. Finally, the experimental results of different algorithms were compared.

First, 10 datasets including GZ, CQ, FJ and SH were selected as the local datasets. Then a comparison was made on the classification accuracy of algorithms in the single local dataset via 10-fold cross-validation. Specifically speaking, all samples in each local dataset were divided into 10 grades: nine of them were used as the training sets in turn, and one of them was used as the test set to conduct verification. The average

value of 10 tests was taken as the classification accuracy of the algorithm for the datasets. Table 2 shows the classification performance of various algorithms in 10 different single local dataset.

Table 2: Comparison of accuracy among various algorithms in 10 single local datasets

Local dataset	Sample size (Unit: K)	CART (%)	AdaBoost (%)	DIC (%)
GZ	9	78.2	86.5	87.2
CQ	16	72.5	84.0	83.1
HL	23	81.1	84.5	85.8
TJ	27	87.9	91.5	92.7
HE	34	69.3	80.1	86.6
FJ	41	70.1	78.5	85.4
HA	51	88.5	91.8	93.8
ZJ	67	86.4	89.6	92.6
SH	88	73.3	76.2	83.1
GD	116	72.5	83.3	88.3
Average	—	78.0	84.6	87.9

According to Table 2, the classification accuracy of two integrated algorithms including AdaBoost and DIC is higher than that of the other simple classification algorithm. Moreover, the classification model established with the DIC algorithm proposed in this paper shows the highest average classification accuracy in the 9 single local dataset of different sample scales. The accuracy of DIC algorithm is slightly lower than that of AdaBoost algorithm only in one single local dataset CQ. In several unbalanced single local dataset like HE, FJ and GD, the classification accuracy of DIC algorithm is obviously higher than that of other algorithms, showing relatively great performance superiority. Generally speaking, the classification accuracy of DIC algorithm is superior to that of the other two popular classification algorithms.

In order to better verify the classification accuracy of DIC algorithm in big data environment, the datasets was expanded to the global datasets from the single dataset. The point of GZ was selected as the local dataset, and 1/2 of local data samples were chosen as the training set. Data in other datasets are added to this training set in equal proportion, to gain the training sample set of the local dataset GZ in the global datasets. The training set was divided according to the proportions of 20%, 40%, 60% and 80% via cross test method, and the corresponding category distribution difference coefficient ρ_i was set for training samples from various local datasets. Each algorithm generated an integrated classification model in the training sets of four proportions. The remaining 1/2 of samples in the local dataset GZ were used as the test set, and multiple groups of samples to be tested were chosen to verify the classification models gained through the three integrated algorithms. The classification accuracy of Bagging, AdaBoost and DIC in the training sets of four proportions was obtained after prediction, as shown in Figure. 4.

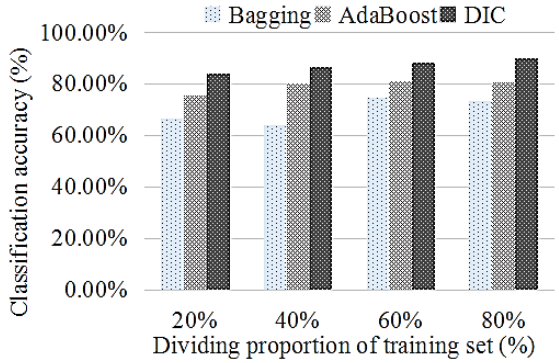

Fig. 4: Comparison of results among 3 integrated classification algorithms for the local dataset GZ

According to the experimental result of Fig. 4, in the global datasets established for different local datasets, when the training set is divided with different proportions, Bagging algorithm shows the lowest classification accuracy in the distributed global datasets, while DIC algorithm presents relatively higher accuracy than that of the other 2 popular integrated classification algorithms. With the increase of training samples, the rising range of classification accuracy of DIC algorithm is the smallest. Moreover, it also

presents comparatively high classification accuracy in imbalanced datasets, showing that the classification performance of DIC algorithm is more stable.

3.2. Analysis on the adaptive capacity of DIC algorithm for distributed dynamic datasets

In order to verify the adaptive capacity of DIC algorithm when faced with distributed dynamic datasets, the local dataset acquisition point JS of bigger data size was used to construct the global datasets and conduct experimental verification. First, datasets collected in the first week were used to generate the training set, and the classification model was gained via algorithms. Besides, the classification accuracy was tested. Then, the data samples collected in the second, third and fourth weeks were added successively. Later, an experimental analysis was made on the algorithms. Fig. 5 shows the change situations about the classification accuracy of AdaBoost and DIC with time.

According to the comparison of classification performance between the 2 integrated classification algorithms in Fig. 5, when numerous samples are added to the dataset continuously, the classification accuracy of AdaBoost algorithm shows a great change, while the classification accuracy of DIC algorithm presents a relatively stable. Therefore, in the dynamic expansion process of dataset, the classification performance of DIC algorithm is superior to that of the AdaBoost algorithm. Besides, its adaptive capacity to distributed dynamic datasets is enhanced. Because DIC algorithm considers the change of dataset category distribution in time, re-train the classifier, and dynamically generate the recognition model in test.

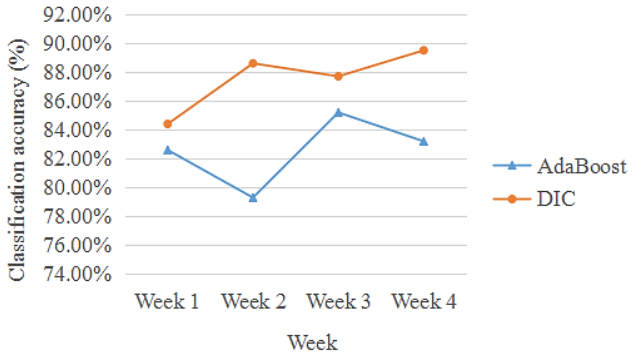

Fig. 5: Change situations about the classification accuracy of 2 integrated classification algorithms.

According to the comprehensive analysis on the above experimental results, DIC algorithm in big data environment proposed by this paper has high classification accuracy, and can stably improve the classification performance of the classifier when faced with distributed dynamic datasets.

4. Conclusion

A dynamic integrated classification algorithm based on big data environment is proposed. This algorithm introduced distribution difference coefficient between local datasets, the influencing parameters of decision attributes for classification in sub-classifier, and the computing formula for attribute distribution difference between samples to be tested and various local training samples. The classification model could be generated dynamically in the application process. The experimental results show that the dynamic integrated algorithm proposed in this paper presents a good classification effect in mass data classification, and is a feasible integrated classification model based on big data environment.

5. Acknowledgments

Our thanks to Youth science and technology talent growth project of guizhou province(KY[2018]112). Our works are supported by this project.

6. References

[1] Zhang Ming-Wei, zhu zhi-Liang, liu Ying et al. A distributed assistant associative classification algorithm in big data environment[J]. Journal of Software 2015,26(11):2795-2810.

[2] Jiang Ji-chun, Ma Dan. Dynamic integration algorithm of multiple classifiers based on improved AdaBoost [J].

Computer Engineering and Design.2015,36(11):3000-3004.

[3] Chen Xue-bin, Wang Shi, Dong Yan-yan. Research on Parallel Classification Hybrid Algorithm for Big Data [J]. Microelectronics and Computer. 2016,33(4):138-140.

[4] Wang Yan-bin, Wu You-xi, Liu Hong-pu. Research and Application of Ensemble Learning Using Gradient Optimization Decision Tree [J]. Computer Science. 2008,45(11A):121-125.

[5] Zhang Fei-fei, Wang Li-ming, Chai Yu-mei. Over-sampling Based Ensemble Classification Algorithm on Imbalanced Data. Journal of Chinese Computer Systems. 2018,10(10):2162-2168.

[6] Wang Guo-yin, Liu qun, Yu Hong et al. Data mining and application of big data [M]. Beijing: Tsinghua University Press, 2017:197-201.

[7] Gema, B.O. Jason, J. J. David, C. Social big data: Recent achievements and new hallenges[J]. Information Fusion. 2016,28:45-59.

[8] Cao, J.Z.; Wu, H.Y. POI Location Updating Method Based on ttendance Data[J]. Geospatial Information. 2013, 11(2), 15-18.

[9] Zhang L S, Yang M J, Lei D J. An improved PAM clustering algorithm based on initial clustering centers. Appl Mech Mater. 2012, 135/136: 244