ISBN 978-981-14-4787-7
Proceedings of 2020 the 10th International Workshop on Computer Science and Engineering (WCSE 2020)
Yangon (Rangoon), Myanmar (Burma), February 26- February 28, 2020, pp. 6-10
doi: 10.18178/wcse.2020.02.002

A Novel Clustering-based Class-asociation Rule Mining Method for Handling Class-Imbalanced Datasets

Tien-Dung Phan[1], Thanh-Tho Quan[2+] and Thi-Kim-Anh Vo[3]

[1]Faculty of Foreign Languages and Information Technology, People's Police College II, Vietnam
[2]Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology, Vietnam
[3]Faculty of Information Technology, Ho Chi Minh City Open University, Vietnam

Abstract. Class-association Rules (CARs) mining is a knowledge discovery technique with many practical applications. One of the extensions of mining CARs algorithm is to combine information about data classes to derive rules between item and class. However, in the class-imbalance field, it is difficult to mine the rules related to minor classes. One of the solutions is at first to cluster with the combination with CARs mining, then the items of minor classes can be grouped to some clusters. Thus, the corresponding rules will be easier to detect. The k-means clustering method is often used due to its fast computing speed. However, the clustering results of k-means are non-deterministic, so it may affect the clustering quality. In this study, we propose a new direction for combining k-means and Hierarchical Agglomerative Clustering, and continue with class-based association rule mining. Our method has the same execution time as the k-means method but has better clustering quality, so the generated rules are also more accurate, as illustrated in the experimental results.

Keywords: accuracy, CARs, classification, class-imbalanced dataset, clustering

1. Introduction

The classification method based on association rules has been researched and proved to be better than traditional rule-based methods such as ILA, ID3, etc. [1-3]. In fact, class-imbalanced datasets are quite common. This means there will be some layers with a number of samples that are superior to the others, which greatly affects the training process to classify and to predict classes. Especially, when classifying a classifier, if we choose an inappropriate minimum support threshold *(minSup),* the samples of minor classes will be unfrequent or the rules will be mined mainly the majority class samples.

For the above issues, Nguyen et al. (2016) proposed a clustering method using k-means algorithm to balance the number of samples of each class, then use CAR-Miner algorithm to mine the classification rules [4]. The study has demonstrated a significant improvement in the accuracy of comparisons between with and without implementation of class equilibrium. However, we realize that with this study, there are still limitations of k-means clustering technique and CAR-Miner algorithm:

- *k*-means clustering: although the execution time is relatively fast, it does not guarantee the similarity between the components in the cluster is good enough and it can not handle noises and outliers.

- CAR-Miner: although it is an efficient algorithm to mine classification rules based on MECR tree structure (Modified *Equivalence Class-Rules* tree). However, this algorithm consumes a lot of memory for storing the Obidsets (set of object identifiers containing itemset) of the itemset and requires computation time for the intersection of Obidset sets to each other. So, for a large database, this issue will become significant.

[+] Corresponding author. Tel.: +84 919890203
E-mail address: qttho@cse.hcmut.edu.vn

Based on limitations of k-means clustering methods when balancing class samples, we propose a new method to increase the similarity of data after being clustered in order to increase the accuracy for class prediction. Besides, we also apply CAR-Miner-Diff algorithm to solve the disadvantages of CAR-Miner presented in [4].

2. Mining Class-association Rules

Mining classification rules based on association rules mining (*Class Association Rules - CARs*) is to find a subset of association rules contained in the database [5]. The goal of mining classification rules based on association rule mining is: (i) Mining CARs meeting minimum support threshold (*minSup*) and minimum confidence threshold (*minConf*) ; and (ii) Build classifiers from CARs.

CAR-Miner is an improved algorithm of ECR-CARM algorithm developed by Nguyen et al in 2013 [6]. CAR-Miner mines class association rules based on MECR-tree structure. The MECR-tree structure (Modification of Equivalence Class Rule tree) is an improved tree structure from the ECR-tree structure, each node in the tree contains only a set of itemset with the following information:

- *Obidset* : a set of task object identifiers that contains itemset.

- (c_1, c_2,..., c_k): a list of integers, where c_i is the number of records in *Obidset* belonging to class c_i.

- pos:positive integers stores the position of the class with the highest count, ie. $pos = argmax_{i\in[1,k]}\{c_i\}$.

To solve the limitation of CAR-Miner algorithm based on MECR-tree structure, CAR-Miner-Diff algorithm was born. The CAR-Miner algorithm consumes quite a lot of memory for *storing Obidsets* of itemset sets and requires computation time for the intersection of *Obidset* episodes , this time becomes significant when we consider in a large database . CAR-Miner-Diff is an improved algorithm of CAR-Miner algorithm developed by Nguyen et. al [7]. CAR-Miner-Diff instead of storing the intersection between the Obidset sets, it only stores the difference between those Obidset sets (called *Diffset*), this leads to memory and speed of mining the association rules based on the tree structure are improved .

3. Combination of Clustering Algorithms and Mining Class-Association Rules

3.1. The Balance of the Class and the Clustering Algorithm Combination

In the data of class imbalance, the fact that some classes are in the majority will have a significant influence on the rule-based prediction process due to difficulties in selecting the minimum support thresholds. If the selected threshold is too high, leading to classes containing small samples would not be frequent, so there are no rules containing this class. If we select low threshold to mine the rules containing minority classes, the number of rules of the majority classes is still overwhelming so it also affects the class prediction stage. Therefore, we will balance the data of each class first, then perform the CARs mining.

In this paper, we use the concept of *intra-cluster similarity* to measure similarity between elements in a cluster. If this value is larger, the elements in the cluster will have higher similarity, thus clustering quality is better. Let *C* be a cluster with m elements, the similarity in cluster of *C*, denoted by Θ(C)) is the average of similarity between samples in *C* and is calculated by the following formula:

$$\Theta(C) = (\sum_{i=1}^{m-1}\sum_{j=i+1}^{m} Sim(W_i, W_j)) / (\sum_{i=1}^{m-1} i) \quad (1)$$

where *Sim* (W_i, W_j) is the similarity between the 2 samples in a cluster.

k-means is a simple clustering algorithm, the execution time is quite fast with the algorithm complexity is *O (nkd)* where *k* is the number of clusters, *n* is the number of samples and *d* is the number of times p. However, k-means does not guarantee the similarity in clusters is good enough. In contrast, HAC clustering algorithm has a longer execution time than k-means due to $O(n^2)$ complexity, but it returns cluster results with very similar clustering results [8]. Because we use the k-means method in the first step, the input of the HAC algorithm will be relatively small clusters, which makes the HAC algorithm run much faster in the second step. In addition, clustering can be controlled by HAC for better cluster quality.

3.2. K-means_Car-Miner-Diff

K-means algorithm is combined with the Car-Miner-Diff algorithm shown in Figure 1:

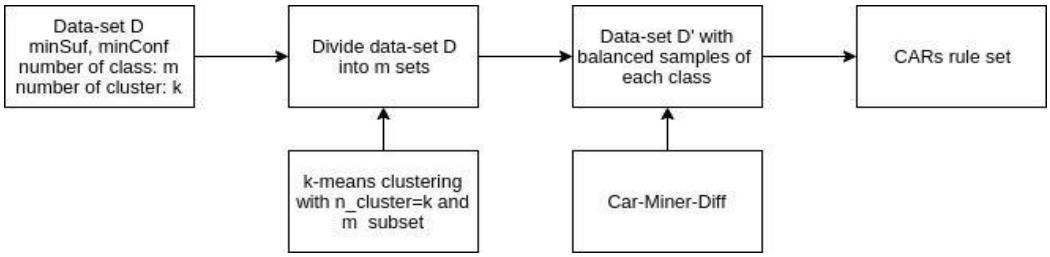

Fig. 1: Steps to combine the k-means algorithm with Car-Miner-Diff

INPUT: Dataset D, minSup, minConf, number of class m, number of cluster k.

OUTPUT: CARs rule set satisfies minSup and minconf.

3.3. K-means_HAC_Car-Miner-Diff

The k-means + HAC algorithm is combined with the Car-Miner-Diff algorithm shown in Figure 2:

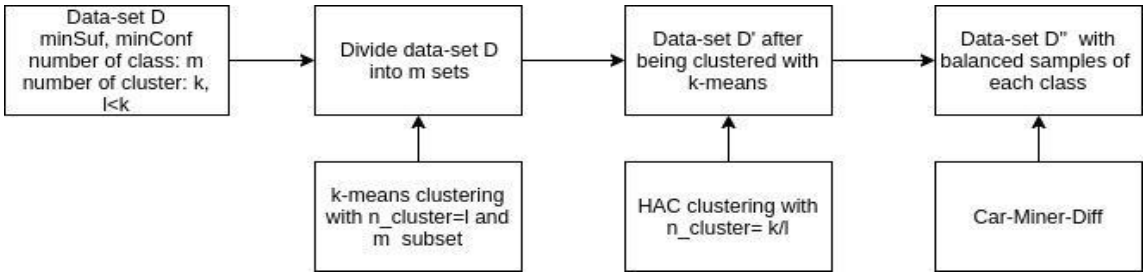

Fig. 2: Steps combination algorithm k-means + HAC with Car-Miner-Diff

INPUT: Dataset D, minSup, minConf, number of class m, number of cluster k, l (l<k).

OUTPUT: CARs rule set satisfies minSup and minconf.

First, we divide dataset D into m subsets corresponding to m values of class attribute. Let k be the number of data rows of the sub-data with the least number of samples.

Then, for each subset with the number of data rows that are greater than k, we apply k-means algorithm on that subset with l clusters (l<k). After that, we have a dataset D'.

For each small cluster which is the result of *k*-means clustering, we continue to apply HAC clustering with k/l clusters. With each cluster created, we only select a representative sample (the sample is the most similar to the center of the cluster). Thus, the result of each original subset will retain k samples and we have dataset D'' with balanced samples of each class.

Finally, we apply the Car-Miner-Diff algorithm on dataset D'' to mine CARs rule set.

4. Experimental Results

The standard empirical databases are taken from the UCI website http://mlearn.ics.uci.edu (Table 1)

Table 1: Experimental standard database

DATA SET	NUMBER OF PROPERTIES	NUMBER OF CLASSES	MODEL NUMBER	DESCRIPTION
Breast Cancer	9	2	683	- Class 0: 444 (65%)- Class 1: 239 (35%)
Chess	10	2	1200	- Class 0: 900 (75%) - Class 1: 300 (25%)
Diabetes	8	2	1400	- Class 0: 942 (67.3%) - Class 1: 458 (32.7%)
Tic-tac-toe	9	2	958	- Class 0: 332 (34.6%)- Class 1: 626 (65.4%)

4.1. Comparative Results on Accuracy

To compare and evaluate the results of accuracy of 03 algorithms: Car-Miner-Diff, k-means_Car-Miner-Diff, k-means_HAC_Car-Miner-Diff, the article uses 04 standard databases, minConf = 60% To proceed with the installation: Experimental results for the accuracy of the three algorithms are presented in Table 2

Table 2: Experimental results on the accuracy of standard databases (%)

Breast cancer	**MINSUP**	**0.5**	**0.3**	**0.1**	**0.08**
	k-means_HAC_Car-Miner-Diff	48.8966	63.1724	94.2759	95.1724
	k-means_Car-Miner-Diff	51.3235	61.5441	90.8088	92.6471
	Car-Miner-Diff	69.8049	69.561	79.3659	91.2195
Chess	**MINSUP**	**0.5**	**0.3**	**0.1**	**0.08**
	k-means_HAC_Car-Miner-Diff	51.1111	77.2222	83.8889	85.5556
	k-means_Car-Miner-Diff	78.9474	78.9474	78.9474	78.9474
	Car-Miner-Diff	75.5556	75.5556	75.5556	75.5556
Diabetes	**MINSUP**	**0.1**	**0.05**	**0.01**	**0.005**
	k-means_HAC_Car-Miner-Diff	75.6345	82.7411	86.802	87.3096
	k-means_Car-Miner-Diff	78.0749	78.0749	83.9572	84.492
	Car-Miner-Diff	71.9048	76.6667	79.7619	80.7143
Tic-tac-toe	**MINSUP**	**0.3**	**0.1**	**0.08**	**0.05**
	k-means_HAC_Car-Miner-Diff	53.2843	62.9902	74.0196	99.0196
	k-means_Car-Miner-Diff	50.7	70	69	88.5
	Car-Miner-Diff	68.4722	67.3611	67.3611	75.6944

The results from Table 2 show that for unbalanced class databases, the improved k-means_HAC_Car-Miner-Diff method results in better accuracy, especially for small minSup thresholds.

4.2. Comparison on Algorithm Execution Time

To compare and evaluate the results of the time of mining the rules of 02 algorithms: Car-Miner-Diff, k-means_HAC_Car-Miner-Diff, the article uses 04 standard databases minConf = 60% to proceed with the installation. Experimental results Perform the execution time between two algorithms are presented in Table 3.

Table 3: Experimental results on the implementation time on the standard database (ms)

Breast cancer	**MINSUP**	**0.5**	**0.3**	**0.1**	**0.08**
	Kmeans HAC Car-Miner-Diff	0	3	43	63
	Car-Miner-Diff	7	16	91	108
Chess	**MINSUP**	**0.5**	**0.3**	**0.1**	**0.08**
	Kmeans HAC Car-Miner-Diff	1	41	270	330
	Car-Miner-Diff	4	43	330	400
Diabetes	**MINSUP**	**0.1**	**0.05**	**0.01**	**0.005**
	Kmeans HAC Car-Miner-Diff	55	130	570	900
	Car-Miner-Diff	83	180	580	950
Tic-tac-toe	**MINSUP**	**0.3**	**0.1**	**0.08**	**0.05**
	Kmeans HAC Car-Miner-Diff	0	67	107	210
	Car-Miner-Diff	1	97	120	260

The results from Table 3 show that for databases with class imbalance, the k-means_HAC_Car-Miner-Diff improvement method due to processing with fewer samples after clustering should result in better processing time, especially for small minSup thresholds.

5. Summary and Future Work

In this paper, we have proposed an improved method combining k-means and HAC clustering techniques, implemented algorithms on standard databases, documented accuracy and execution time between subject methods, exported and original CAR-Miner-Diff algorithm for verification. At the same time, the paper also compares the similarity after clustering for 02 methods k-means and k-means_HAC.

In the future, we will continue to experiment on more types of databases with the increased number of classes to evaluate the applicability of the proposed improvement method. Furthermore, we will also apply this method to other types of classification such as decision trees, ILA, neural networks, etc.

6. Acknowledgements

This research is funded by Vietnam National University Ho Chi Minh City (VNU-HCM) under Grant B2018-20-07

7. References

[1] A. Veloso, W. Meira Jr., M.J. Zaki (2006). Lazy associative classification. In: Proc. of The 2006 IEEE International Conference on Data Mining (ICDM'06), Hong Kong, China, pp. 645- 654.

[2] A. Veloso, W. Meira Jr., M. Goncalves, H.M. Almeida, M.J. Zaki (2007). Multi-label lazy associative classification. In: Proc. of The 11th European Conference on Principles of Data Mining and Knowledge Discovery, Warsaw, Poland, pp. 605-612.

[3] A. Veloso, W. Meira Jr., M. Goncalves, H.M. Almeida, M.J. Zaki (2011). Calibrated lazy associative classification. Information Sciences 181(13), pp. 2656-2670.

[4] L.T.T. Nguyen, TMT. Tran, CH. Giang (2016). Exploiting association clustering rules with class-imbalanced dataset. In Proceedings the 10th National Conference of Fundamental and Applied IT Research (FAIR).

[5] J. Han, M. Kamber, Data Mining: Concepts and Techniques, 3rd Edition. Morgan Kaufmann Publishers, 2011

[6] L.T.T. Nguyen, B. Vo, T.P. Hong, H.C. Thanh (2013). CAR-Miner: An efficient algorithm for mining class-association rules. Expert Systems with Applications, vol.40, no.6, pp. 2305-2311.

[7] L.T.T. Nguyen, Ngoc Thanh Nguyen (2015). CAR-Miner: An improved algorithm for mining class association rules using the difference of Obidsets. Expert Systems with Applications, vol.42, pp. 4361-4369.

[8] K.T.Huynh et al. (2017), "A quality-controlled logic-based clustering approach for web service composition and verification", International Journal of Web Information Systems, Vol. 13 Issue: 2, pp.173-198