Efficient Heterogeneous Multi-relational Classification Using Multi-criteria Ranking Approach Based on Characteristics of Multiple Relations

Amit R. Thakkar[1*], Yogeshwar P. Kosta[2]

[1] Department of Information Technology, Charotar University of Science & Technology (CHARUSAT), CHARUSAT Campus, Changa, Gujarat, India.
[2] Faculty of Engineering, Marwadi Group on Institutions, Rajkot, Gujarat, India.

* Corresponding author. Tel.:+91-9601290990; email: amitthakkar.it@charusat.ac.in
Manuscript submitted April 22, 2015; accepted July 15, 2015.
doi: 10.17706/jcp.10.6.418-426

Abstract: Traditional data mining algorithms will not work efficiently for most of the real world applications where the data is stored in relational format. Even well-known traditional classification technique such as J48, Naïve Bayes often suffers from poor scalability and unsatisfactory predictive performance when it comes to working with relational data. Moreover the performance of existing relational classification is also limited as the existing algorithms are not able to use different classifiers based on characteristics of different relations. Proposed approach in this paper is to select appropriate classifiers based on characteristics of dataset and give ranking based on multi criteria function using Ratio of Success Rate and Time (RST). In RST we combine success rate as a measure of benefit and running time as a measure of cost. The goal of the proposed relational classification is to use most appropriate and efficient classifier for the relation to achieve better efficiency as compared to the common classifiers. The experimental results show that the performance of proposed relational classification is better in terms of accuracy and efficiency when compared to all other existing algorithms available in the literature.

Key words: Multi relational data mining, relational classification, meta learning.

1. Introduction

The primary objective of any data mining activity is to check the given database for patterns that may give better understanding for given task. Unfortunately, the widespread application of data mining has been limited by an assumption that all data resides in a single table [1]. Most of the data mining algorithms which are currently available are based on single table setting which restricts their use to datasets consisting of a multiple tables and such algorithm only allows the analysis of fairly simple objects [2]. To be able to analyze relational databases containing multiple relations properly, new algorithms will have to be written to deal with the structural information that occurs in relational databases. Multi relational classification method searches for relevant features both from a target relation and relations related to the target, in order to better classify tuples in the target relation.

There are basically two approaches available in the literature to classify multi-relational data. In the first approach relational database has to be converted to a single table format, so that propositional data mining algorithms will be able to work with database which is known as Propositional Data Mining [3]. Methods

such as RELAGGS have been developed to perform this translation. Although more data is produced but a lot of information about how the data was originally structured is lost. The class of techniques that support the analysis of structured objects is known as Structured Data Mining [4]. These techniques are typically upgrades from well-known and accepted Data Mining techniques for tabular data, and focus on dealing with the richer representational setting [2], [3]. Thus the task of learning from relational data has begun to receive significant attention in the literature, especially for the relational classification task [3], [5]-[10]. The rest of the paper is organized as follow. We first describe the related work and then current research challenges. Then we have discussed proposed algorithms. The final section evaluates the classification performance of the different models on benchmark datasets.

2. Related Work

The two major families of approaches one which upgrades the traditional learning algorithms and other extensively preprocess ("flatten") the multiple relations to a single universal flat file, have demonstrated some shortcomings. Existing “upgrading” approaches give poor performance when applied to data having noise or numeric values. The “flattening” strategies require lot of time for the data transformation. So a new approach multi-view learning [8], [11], [12] has received significant attention in literature. The algorithm (MVC) proposed in [12] , successfully used decision tree as a relational classification and classifying objects across multiple relations efficiently, without converting multiple relations to a single one. In [9], author proposed a scalable, two phase classification algorithm for classifying relational data. Using the semantic information, authors proposed a new recursive aggregation technique to collect heterogeneous classification applied at individual tables. In [13], the algorithm is proposed to classify hierarchical continuous label data. In this method, first the class hierarchical decision tree is generated which shows the predefined ranges. Then the new attribute is selected based on the goodness of splitting node and finally the class label is determined. In approach of [14] given a set of extended join tables, probabilistic classifiers are learn independently from each extended join table and appropriate weight are given to different join tables before combining them using Bayes combination scheme. In multi relational classification different relation are having different characteristics so it is hard to classify all instances of different relations using a common classifier still obtaining a good generalization performance. Moreover, for certain types of data, certain choices of local classifier may be known to perform better therefore it could be better to use different choices of classifiers for each relation [15]. As a result, these difficulties have prevented a wider application of multi relational mining, and post an urgent challenge to the data mining community. In the subsequent section, we have proposed new relational classification model based on most efficient heterogeneous classifiers based on their characteristics.

3. Introduction to the Proposed System

To address the above mentioned problems, we proposed a new multiple relational classification approach based on weighted voting technique and heterogeneous classifiers. The proposed approach also enables to use most appropriate traditional data mining methods to improve the overall efficiency of relational classification. In this section we describe an algorithm called Multi Relational Classification with weighted voting using heterogeneous classifiers (MRC_WV_RST).

3.1. Steps of Proposed Multi-relational Classification Model (MRC)

Fig. 1 shows five stages of MRC framework for relational classification which are briefly described below.

Class label propagation Stage: In this step, the training data sets, which are used by a number of relational learners, are built by propagating information of class label from the target relation to the background relations, based on the foreign key links.

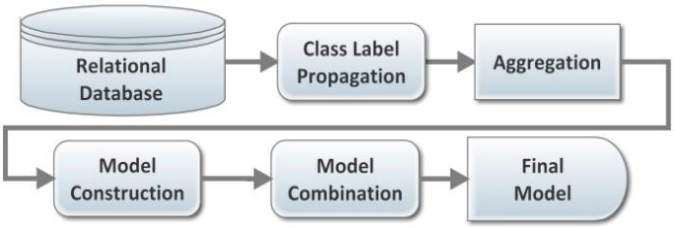


Fig. 1. Steps of proposed multi relational classification model.

Aggregation Stage: The training data built after class label propagation may have one-to-many relationship. So in this step we need to apply an aggregation function which summarizes information from multiple tuples into one row using the key from the target relation. Each newly built background relation is then used as training data by different classifiers.

Model Construction Stage: In this step, traditional single-table classification algorithms are used to learn from each relation of relational database. In multi relational classification different relations are having different characteristics so it is hard to classify all those relations using a common classifier while still obtaining a good generalization performance. So the goal of this stage is to use classifier for different relations where it is most suitable and thus to achieve overall results that can be considerably better than using common classifier for relations. Meta Learning is used to select appropriate classifier depending of dataset characteristics. Meta Learning System can be divided into two modes: Acquisition Mode and Advisory Mode [15]. In Acquisition mode, Meta-knowledge base is created using the features that are extracted and the performance on the training datasets. The Aim of this Advisory mode is to provide ranking for a given dataset based on the Meta knowledge base created during acquisition mode. Ranking is provided based on either considering Ratio of Success Rate method (RSR) or Ratio of Success Rate and Time (RST) method. But in RSR, only performance is considered as a success measure to rank the classifier; so it does not give best performance in terms of efficiency of algorithm. So we have extended the RSR to incorporate time and give ranking based on multi criteria function using Ratio of Success Rate and Time (RST).

Ratio of Success Rate and Running Time ranks algorithms according to the relative advantage/disadvantage they have over other algorithms using multi criteria ranking approach RST using following steps.

Step 1: Find ratio of success Rate and time of algorithm a_p and a_q for algorithm *i*.

$$RST_{a_p,a_q}^{d_i} = \frac{\frac{SR_{a_p}^{d_i}}{SR_{a_q}^{d_i}}}{1+ log_{10}\left(\frac{T_{a_p}^{d_i}}{T_{a_q}^{d_i}}\right)} \tag{1}$$

where, $SR_{a_p}^{d_i}/SR_{a_q}^{d_i}$ gives relative advantage of algorithm a_p over a_q while $T_{a_p}^{d_i}/T_{a_q}^{d_i}$ gives relative disadvantage of algorithm a_p over a_q. So by dividing a measure of benefit by a measure of cost, we get the overall quality of an efficient algorithm. As the range of time is much wider than the range of accuracy ***log*** based reduction is used to normalize the value of time.

Step 2: Calculate a pair wise ratio of success Rate and time for each pair of algorithms.

$$RST_{a_p,a_q} = \frac{\left(\sum_{di} RST_{a_p,a_q}^{di}\right)}{n}, \tag{2}$$

where *n* is number of datasets

Step 3: Find overall Rank for each algorithm

$$RST_{a_p} = \frac{\left(\sum_{a_q} RST_{a_p,a_q}\right)}{(m-1)}, \quad (3)$$

where n is number of algorithm

The classifier with highest value of RST is selected by MRC algorithm for classification.

Model Combination Stage: in this stage, multiple learners are pruned and combined to construct the final classification model to improve the accuracy and efficiency of the proposed approach based on score of classifiers. The score for different classes can be combined using either un-weighted or weighted combination techniques. As in the relational classification each relation has different competency to classify the tuple, un-weighted method is not useful. All existing weighted schemes like weighted Majority vote [16] or Best Worst Weighted Voting [17] techniques are not directly applicable because in relational classification we want to assign relatively high weight to classifiers which are performing better to classify the tuples from relevant background table. At the same time we have to remove those classifiers which are not contributing in the classification decision. So below we have provided new weight assignment scheme for our multi relational classification model. In proposed combination scheme, weight zero is assigned if misclassification error of classifier is greater than 0.5 means eliminating non-contributing classifiers from relational learning. The weight of rest of the classifiers is calculated as follows:

$$W_i = \begin{cases} 0 & \text{if misclasification error} \geq 0.5 \\ \left(\frac{(\text{Error}_W - \text{Error}_i)}{(\text{Error}_W - \text{Error}_b)} * \left(\text{Max}_{\text{Weight}} - \text{Min}_{\text{Weight}}\right)\right) + \text{Min}_{\text{Weight}} \end{cases} \quad (4)$$

where Error_b is misclassification of best accurate classifier and Error_w is error misclassification error of lowest accurate classifier. Now combining all the steps of MRC Framework, we have designed the flowchart for training and testing phase of our proposed algorithm.

- **Description of Training Phase**

As shown in the Fig 2, First of all, the algorithm construct multi-relation training set from each background relation using propagation of class label and aggregation. After construction of the individual training data sets, meta learning is used to select most appropriate classifier. The selected algorithm is called to learn the target concept from each of the training sets. In this way, all classifiers make different observations on the target concept. After that weighted vote is assigned to every classifier as per equation given in the algorithm. The final output of training phase is score matrix and weight assignment table.

- **Description of Testing Phase**

As shown in Fig 3, in testing phase, for each tuple x from the target table, a classifier will retrieve the tuple from background table RBi corresponding to the key reference. If a corresponding tuples is found, aggregation operations are applied on tuples and score matrix is obtained. Then voting technique is applied to combine result of different relations. Here weight acquired during voting is multiply with score matrix and then sum of probability is applied which returns accurate class label to unknown testing tuples.

4. Experimental Analysis

This section provides the results obtained for the MRC_WV_RST algorithm on three benchmark real-world databases. These results are compared with other well-known multi-relational data mining systems, namely RelAggs flattening approach, latest relational classifier based on multi view learning i.e MVC, MRC based on weighted voting MRC_WV and MRC based on weighted voting using Heterogeneous learners MRC_WV_RSR. We have implemented the MRC based algorithms using Java and Weka open source data mining tool. Also, in our experiments appropriate classifiers are selected using Meta learning. All methods were run on an Intel core to duo based machine with a 2.2 GHz processor and 3 GB of main memory.

Proposed Algorithm: Multi Relational Classification Using Heterogeneous Classifiers using Ratio of Success Rate and Time (RST): (MRC_WV_RST) Training Phase

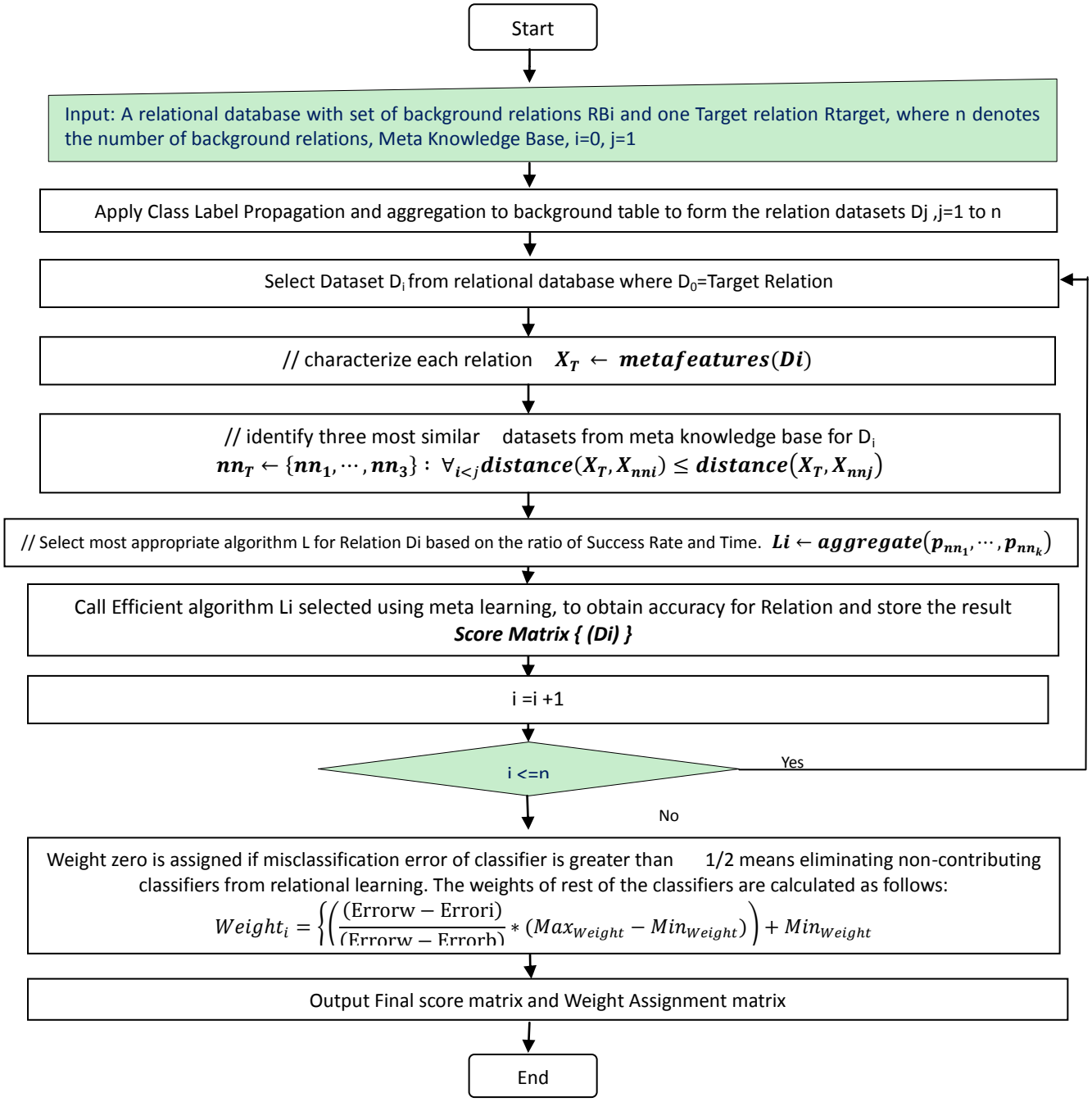


Fig. 2. Flowchart for training phase of algorithm.

4.1. Database Details

The three benchmark databases, namely Mutagenesis [18], Thrombosis [19] and Financial [20] are used to test the algorithm. All three database comes from different application domains, have variant relational structures, consist of different numbers of tuples in the entire database and in the target relation, and present varying degree of class distribution in the target relation.

Experiment 1: In the first experiment, we have used Meta learning to select appropriate classifiers for MRC_WV_RSR and MRC_WV_RST algorithms as no classifier is able to perform best for all the relation because different relations have different characteristics. Similarly for all other relations of all databases

RSR and RST Based Ranking are calculated. For Loan relation of Financial Database the RSR based method select IB1 algorithm as it only consider the accuracy for ranking the classifier while RST based method select Naïve Bayes algorithm (NB) by considering both time and accuracy to improve overall efficiency of relational classifier without affecting the performance much. Similarly for all the relation RSR and RST based ranking is calculated.

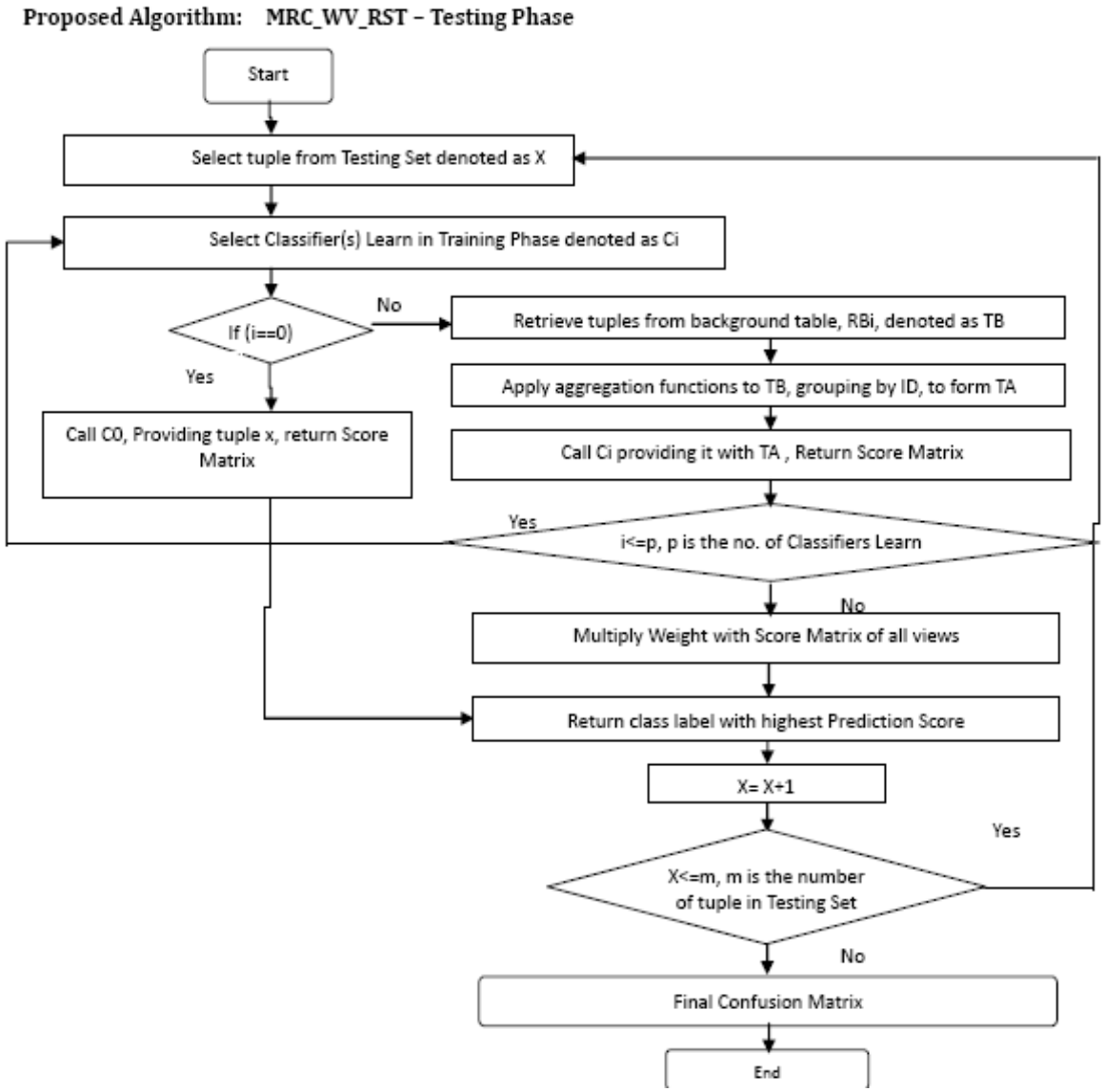
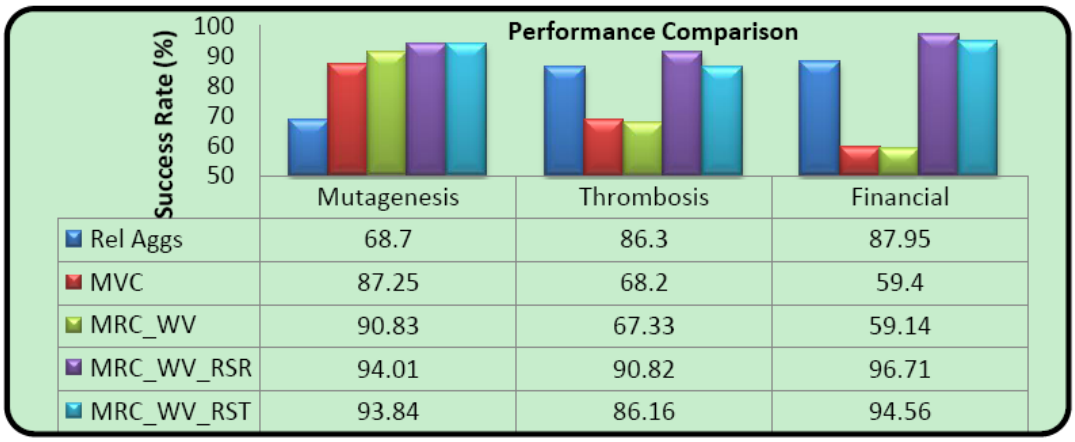


Fig. 3. Flowchart for testing phase of proposed algorithm.

Experiment 2: In this experiment, we examine the performance of the proposed algorithm in terms of accuracy obtained. In this experiment, Decision Tree Classifier is used as propositional learners in Flattening Approach Rel Aggs , as base leaner and meta learner in MVC [12], as a base learner in MRC_WV approaches. In this experiment, Heterogeneous Classifiers are selected using RSR and RST based ranking method and used as a relational learner by the MRC_WV_RSR and MRC_WV_RST algorithm respectively. In this experiment, we examine the performance of the MRC_WV_RST using Heterogeneous learning algorithms in terms of Success Rate. As we have used two class problems, accuracy will not give us the actual performance so we have used success rate which is average of true positive and true negative count from confusion matrix.

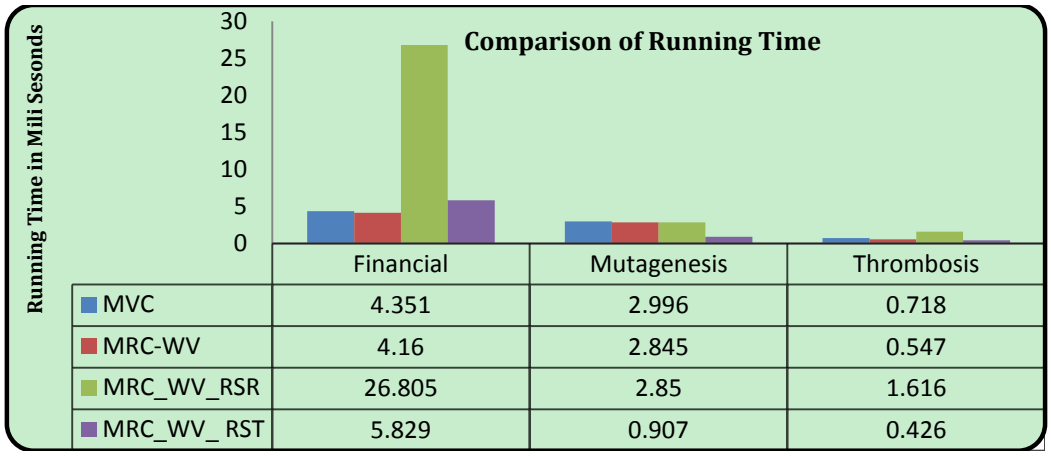


	Mutagenesis	Thrombosis	Financial
Rel Aggs	68.7	86.3	87.95
MVC	87.25	68.2	59.4
MRC_WV	90.83	67.33	59.14
MRC_WV_RSR	94.01	90.82	96.71
MRC_WV_RST	93.84	86.16	94.56

Fig. 4. Comparison of success rate.

By selecting appropriate classifier using RST will improve the efficiency and performance of MRC as individual table is trained by the best appropriate classifier. As presented in Fig. 4, the algorithm MRC_WV_RST outperforms the all other existing algorithm as in MRC_WV_RST best efficient classifier is selected. The proposed MRC_WV_RST gives slightly less performance compare to MRC_WV_RSR as in RSR the ranking is providing based on accuracy only while RST select the best efficient classifier which may not be the best performing classifier.

Experiment 4: To evaluate the performance of MVC [12], MRC with weighted voting and MRC with Heterogeneous learning in terms of running time, we provide the execution time needed for each database.



	Financial	Mutagenesis	Thrombosis
MVC	4.351	2.996	0.718
MRC-WV	4.16	2.845	0.547
MRC_WV_RSR	26.805	2.85	1.616
MRC_WV_ RST	5.829	0.907	0.426

Fig. 5. Running time of MRC with weighted voting, MRC with het learning and MRC with RST.

From the Fig. 5, MRC_WV_RST algorithm took less time compared to MRC_WV_RSR with Heterogeneous Learning as in MRC_WV_RST best performing classifier is selected using multi criteria function RST which give equal importance to accuracy and time for selecting most appropriate classifier as compared to MRC_WV_RSR which select appropriate classifier based on only accuracy. So if we select appropriate classifier using multi criteria function it will improve the overall efficiency of the relational classification.

5. Conclusion and Future Work

In this paper, algorithm called 'MRC with weighted voting using heterogeneous classifiers with RST (MRC_WV_RST)' is proposed. The proposed algorithm is more efficient because of voting technique used to prune non contributing relation and suitable classifier is selected using meta learning which has further improved the performance of the algorithm compared to MRC_WV_RSR. In future other classifier measure

will be used to produce ranking using meta learning and result can be compared with developed algorithm in terms of accuracy and running time. Pre-processing techniques can be incorporated to algorithm which removes irrelevant features from database and may improve result.

References

[1] Arno, J. K. (2004). *Multi Relational Data Mining, SIKS Dissertation*, *Series No. 15*, Universiteit Utrecht, Nederlands.

[2] Hand, D. J., Heikki, M., & Padhraic, S. (2001). *Book of Principles of Data Mining*. MIT Press.

[3] Uwents, W., Gabriele, M., Hendrik, B., Marco, G., & Franco, S. (2011). Neural networks for relational learning: An experimental comparison. *Journal of Machine Learning*, *82(3),* 315-349.

[4] Blockeel, H., Luc, De R., Nico, J., & Bart, D. (1999). Scaling up inductive logic programming by learning from interpretations. *Journal of Data Mining and Knowledge Discovery*, *3(1),* 59-93.

[5] Li, Y., Luan, L., Yan, S., & Yuan, Y. H. (2009). Multi-relational classification based on the contribution of tables. *Proceeding of International Conference on Artificial Intelligence and Computational Intelligence: Vol. 4* (pp. 370-374).

[6] J. F., Guo, Li, J., & W. F., Bian. (2007). An efficient relational decision tree classification algorithm. *Proceedings of Third International Conference on Natural Computation (ICNC)* (pp. 530-534).

[7] H. Y., Liu, Yin, X. X., & Han, J. W. (2005). An efficient multi-relational Naïve Bayesian classifier based on semantic relationship graph. *Proceedings of the 4th International Workshop on Multi-relational Mining* (pp. 39-48). Chicago.

[8] Guo, H. Y., & Herna, L. V. (2005). Mining relational databases with multi-view learning. *Proceedings of the 4th International Workshop on Multi-relational Mining* (pp. 15-24). ACM. Chicago, Illinois, USA.

[9] Manjunath, G. M., Narasimha, M., & Dinkar, S. (2013). Combining heterogeneous classifiers for relational databases. *Journal of Pattern Recognition*, *46(1)*, 317-324.

[10] Yin, X. X., Han, J. W., Jiong, Y., & Philip, S. Y. (2006). Efficient classification across multiple database relations: A crossmine approach. *IEEE Transactions on Knowledge and Data Engineering*, *18(6),* 770-783.

[11] Guo, H. Y., & Herna, L. V. (2006). Multi-view ANNs for multi-relational classification. *Proceedings of International Joint Conference on Neural Networks* (pp. 5259-5266). Vancouver, BC, Canada.

[12] Guo, H. Y., & Herna, L. V. (2008). Multirelational classification: A multiple view approach. *Journal of Knowledge and Information Systems, 17(3),* 287-312.

[13] Hu, H.-W., Chen, Y.-L., & Kwei, T. (2012). A novel decision-tree method for structured continuous-label classification. *IEEE Transactions on Cybernetics, 43(6)*, 1734-1746.

[14] Schulte, O., Bahareh, B., Branden, C., Derek, B., & Yi, X. (2013). A hierarchy of independence assumptions for multi-relational Bayes net classifiers. *Proceedings of IEEE Symposium on Computational Intelligence and Data Mining (CIDM)* (pp. 150-159).

[15] Pavel, B. B., & Carlos, S. (2003). Ranking learning algorithms: Using IBL and meta-learning on accuracy and time results. *Journal of Machine Learning*, *50*, 251–277.

[16] Ludmila, K. (2004). *Combining Pattern Classifiers Methods and Algorithms*. Wiley-Interscience.

[17] Tsoumakas, G., Ioannis, K., & Ioannis, V. (2004). Effective voting of heterogeneous classifiers. *Proceedings of 15th European Conference on Machine Learning (ECML)* (pp. 465-476). Springer Berlin Heidelberg.

[18] Srinivasan, A., Stephen, H. M., Michael, J. E. S., & Ross, D. K. (1996). Theories for mutagenicity: A study in first-order and feature-based induction. *Journal of Artificial Intelligence*, *85(1),* 277-299.

[19] Coursac, I., Duteil, N., & Lucas, N. (2002). PKDD 2001 discovery challenge — medical domain. *The*

PKDD Discovery Challenge, 3(2).
[20] Berka, P. (2000). Guide to the financial data set. *Proceedings of PKDD 2000 Discovery Challenge*.

Amit Thakkar has received his B.E degree in I.T. from Gujarat University in 2002 and the master degree from Dharmsinh Desai University, Gujarat, India in 2007. He has got his PhD degree in the area of multi relational classification at Kadi Sarva Vishwa Vidyalaya, Gandhinagar, India in June 2010. He is currently working as an associate professor in the Department of Information Technology, Faculty of Engineering and Technology, CHARUSAT University, Changa, Gujarat Since 2002. He has published more than 35 research papers in the field of data mining and web technology. His current research interests include multi relational data mining, relational classification and big data analytics.

Yogeshwar P. Kosta has received his M.Tech degree in micro electronics from Delhi University, India in 1991 and his PhD degree in telecommunication engineering in 1997 from Rani Durgavati Vishwavidyalaya, India. He has also completed the advanced project management from Stanford University in 2008. Since 2011 he has been working as a professor and the director at Marwadi Education Group of Institutions, Rajkot Gujarat. He has published more than 100 research papers in the field of electronics, communication & computer science.