International Journal of Scientific Research in Computer Science, Engineering and Information Technology

ISSN : 2456-3307

OPEN ACCESS

Available Online at : www.ijsrcseit.com
doi : https://doi.org/10.32628/CSEIT2390656

A Survey on Relational Database Based Multi Relational Classification Algorithms

Komal Shah[1], Dr. Kajal S. Patel*[2]

[1]Research Scholar, Gujarat Technological University, Ahmedabad, Gujarat, India

*[2]Associate Professor, Vishwakarma Government Engineering College affiliated to Gujarat Technological University, Ahmedabad, Gujarat, India

ARTICLEINFO

Article History:

Accepted: 08 March 2024
Published: 15 March 2024

Publication Issue
Volume 10, Issue 2
March-April-2024

Page Number
140-147

ABSTRACT

Classification on real world database is an important task in data mining. Many classification algorithms can build model only for data in single flat file as input, whereas most of real-world data bases are stored in multiple tables and managed by relational database systems. As conversion of relational data from multiple tables into a single flat file usually causes many problems, development of multi relational classification algorithms becomes popular area of research interests. Relational database based multi relational classification algorithms aim to build a model that can predict class label of unknown tuple with the help of background table knowledge. This method keeps database in it normalized form without distorting structure of database. This paper presents survey of existing multi relational classification algorithms based on relational database.

Keywords: Data Mining, Classification, Relational Data; Multirelational Classification

I. INTRODUCTION

Over the past few years, extensive data collection has become prevalent across diverse fields including science, medicine, banking, and chemistry. Much of this data is organized into multiple tables within relational databases. Consequently, there's been a notable surge in interest towards learning from such relational databases without the need to merge data from different tables explicitly. This field, known as multi-relational data mining, focuses on extracting valuable insights directly from these disparate tables. Various techniques within multi-relational data mining, such as association rule mining, classification, and clustering, have demonstrated successful applications across numerous domains including marketing, healthcare, finance, fraud detection, and the natural sciences [1]

There are two approaches to extracting knowledge from relational databases. The first approach, known as propositional mining method [2], relies on traditional data mining algorithms that assume the data exists within a single table. This method involves consolidating multiple relational tables into a single flat table through various join and aggregation operations. However, this process often leads to the creation of a large table containing all attributes from related tables, including many irrelevant attributes such as IDs and primary keys. Additionally, joining tables can result in the loss of valuable semantic information represented by database links. The resulting table may also contain numerous NULL or missing values, negatively impacting the accuracy of the mining algorithm.

The second approach, relational mining method [3], aims to directly extract knowledge from relational databases without merging all tables into one. Instead, it focuses on developing new algorithms capable of handling relational databases directly, thereby preserving the inherent structure and relationships within the data.

Multi-relational classification within relational databases seeks to develop classification models while maintaining the inherent structure of the database. This approach primarily encompasses two types of classification algorithms: i) Selection Graph-Based Relational Classification and ii) Tuple ID Propagation-Based Relational Classification.

The Selection Graph model leverages SQL, the database query language, to directly interact with the relational tables of the database. Through this model, the relational structure of the database is transformed into a selection graph, which can be easily represented using SQL. Utilizing SQL queries, classifiers can be constructed within this framework. The Multi-Relational Decision Tree learning framework is rooted in the Selection Graph-based relational classification. It shares significant similarities with classic decision tree algorithms but undergoes a series of refinements to add decision tree nodes from various tables until meeting termination criteria, with the leaf nodes obtaining class labels [4].

Tuple ID propagation based relational classification joins relational tables through propagating tuple ID. Relational database tables are classified into one target table and others as non-target tables. Target table contains class label and tables which are joined directly or via some foreign key chain with target table are known as non-target or background tables. Tuple ID propagation propagates class IDs from target table to background tables in relational database. The method does not actually create physical connections like propositional mining method but it performs virtually joining to reduce the costs of time and space [5]. As tuple ID propagation performs virtually join of non-target relations with the target one, it is a simple and efficient method to perform classification.

Following section presents classification algorithms using relational database method in detail.

II. SELECTION GRAPH BASED ALGORITHMS

Numerous algorithms for Multi-Relational Decision Tree learning are rooted in the concept of the selection graph. These algorithms build decision trees where nodes represent multi-relational patterns, also known as selection graphs.

One such algorithm, MRDTL (Multi Relational Decision Tree Learning) [6], is an extension of the TILDE algorithm, which utilizes first-order logic clauses to depict decisions or nodes within the tree. However, in relational databases, data are structured as records in tables rather than in first-order logic. MRDTL extends the TILDE approach to accommodate records within relational tables.

MRDTL augments the decision tree by iteratively refining the nodes until a termination condition is reached, such as accurate classification of tuples in the training set. Upon meeting a termination condition, a leaf node with its corresponding class label is incorporated into the decision tree. The addition of a node to the decision tree is guided by an impurity measure, such as information gain.

The authors of [6] also introduce refinements to the MRDTL algorithm based on suggestions presented in [7]. These refinements include add condition, add edge and node, look ahead, multiple instantiations of associations, and mutual exclusion.

The MRDTL-2 algorithm, introduced in [8], is a refined version of the MRDTL algorithm, incorporating enhancements to improve runtime efficiency and address missing values by employing Naive Bayes classifiers. Utilizing SQL operations, MRDTL-2 calculates the necessary counts for information gain associated with these refinements.

Compared to MRDTL, MRDTL-2 aims to mitigate the slowdown caused by the growing selection graph at deeper nodes in the decision tree. As the decision tree expands with additional nodes, MRDTL experiences extended execution times due to the increasing complexity of SQL queries.

MRDTL-2 tackles this challenge by reusing computed results from higher levels of the decision tree when refining lower levels, thereby reducing execution time. This strategic reuse of computations optimizes runtime efficiency, distinguishing MRDTL-2 from its predecessor.

III. TUPLE ID PROPAGATION BASED ALGORITHMS

Tuple ID propagation is efficient approach for propagating information among different tables. This method propagates the IDs of target tuples along with associated class labels to background relations. In background relation each tuple is associated with a set of IDs representing the target tuples. The propagated IDs will help to find useful features from background table. Tuple ID propagation is a low cost, convenient and flexible method that virtually joins different relations. Since it is easy to propagate IDs between any two relations, searching via any join path in multiple relations is easy, and no repeated computation is required when searching along different join paths that share common prefixes.

CrossMine algorithm [5] uses tuple id propagation and generates rules for classification. The main idea of CrossMine is to repeatedly divide the target relation into partitions, and recursively work on each partition. This algorithm is based on sequential covering algorithm, which repeatedly constructs rules and removes positive tuples covered by each rule. To construct a rule, CrossMine repeatedly searches for the best predicate and appends it to the current rule. During the search process, CrossMine reduces the search space to relations related to the target relation or related to relations used in the rule. In this way the strong semantic links can be identified and the search process is reduced. Thus CrossMine [5] is a scalable and accurate method for multirelational classification based on tuple id propagation.

There are two different algorithms proposed in [9] based on CrossMine method: CrossMine-Tree and CrossMine-Rule. Cross-Mine-Tree [9] is a decision tree based classifier. It recursively selects the best attribute and divides the target tuples into partitions. Each tree node in cross mine decision tree contains two parts: i) prop-path, that shows how tuple IDs are propagated and ii) a splitter that divides all target tuples into several partitions, and creates child nodes corresponding to each partition.

CrossMine-Rule [9] is a rule based classifier. It repeatedly builds predictive rules and then concentrates on remaining target tuples. CrossMine-Rule builds classification rules that can differentiate positive examples from negative ones. Each rule is a list of predicates, associated with a class label. A target tuple satisfies a rule if and only if it satisfies every predicate of the rule. If the numbers of positive and negative tuples are unbalanced, CrossMine-Rule uses a selective sampling method to reduce the number of negative tuples. This provides high scalability with respect to the number of tuples.

A multi relational Naive Bayes algorithm called Graph-NB was proposed in [10]. It uses semantic relationship graph (SRG) and the extended version of Naïve Bayesian formula to support multi relational classification. A semantic relationship graph is a directed acyclic graph describes relationships between tables. To perform virtual join of tables on each path of semantic relationship graph, it uses tuple-id propagation method. To achieve better classification accuracy, a pruning strategy known as “cutting off” is proposed in this algorithm. This pruning method decides best part of semantic relationship graph that needs to be considered while building classification model so that weakly connected tables can be ignored.

A multi-relational Bayesian Classification algorithm named SRG-BC [11] is also based on semantic relationship graph. It uses queue data structure to traverse the SRG in width-first manner. It starts construction of SRG from target table. Then eventually adds all join edges of the current table to the rear of queue. It performs chi-square tests on current table’s attributes and selects discriminatory ones, then pick out the front edge of the queue as the next join route. It uses the tuple ID propagation method to load data of the join edge's right table. This process continues till queue becomes empty. In next step this algorithm performs an iterative selection from the set of features and constructs an optimal feature set. Also this algorithm deletes non-intermediary tables in the SRG that do not include any feature in the optimal set to produce smaller and more compact SRG. This method optimizes the SRG and running time of algorithm.

Algorithm NB-Split [12] is two phase multi relational classification algorithm with a semantic divide and conquer approach. Training phase of NB Split algorithm handles the preparation and training phase of building the classification model. The database schema is directly fetched from reading the metadata of database. In [13] a classification model is proposed that is based on path independence assumption. It uses Naive Bayes classification algorithm as base classifier.

The Relational Decision Tree (RDC) algorithm [4] is derived from the previously discussed MRDTL algorithm. Like MRDTL, RDC employs a decision tree to construct a classification model. It initiates the construction process with the target table serving as the root node of the decision tree. Information gain is computed at each node to guide the refinement of the tree.

Operating recursively, the algorithm selects the best attribute via information gain to partition the data, thus expanding the leaf nodes of the tree until a stopping condition is reached. This stopping condition is evaluated by determining whether all records belong to the same class label, whether they possess identical attribute values, and whether the number of records falls below a specified threshold. To address missing values, the algorithm employs a Naïve Bayes predictor to predict the most likely value for the corresponding attribute. This approach ensures robust handling of missing data within the relational context.

The Classification with Aggregation of Multiple Features (CLAMF) [14] approach constructs classification models from multi-relational data by employing aggregation techniques with both single and multiple features. It utilizes an ILP framework built upon the sequential covering algorithm and employs tuple ID propagation for aggregation across related tables. This technique introduces a systematic approach for selecting and applying suitable aggregation functions tailored to various numbers of features and data types. By integrating multi-feature aggregation predicates, CLAMF enhances classification performance and generates insightful rules with enhanced interpretability.

Another Feature And Relation Selection (FARS) algorithm is proposed in [15]. In this approach to measure correlation between features in a table or features in cross table symmetrical uncertainty is used. The same technique is also extended to measure the correlation between table and class attribute. Authors propose a greedy method to select relevant features and tables based on correlations of features and tables with class attribute. The database schema is reconstructed for selection of highly relevant tables and attributes so that they could be given top priority during classification.

A multi view based multi relational classification [16] also uses Tuple Id propagation as first step in its frame work. In multi view based multi relational classification, database can retain its original structure, multiple views from target and non-target tables are generated and propositional learning algorithms are applied on these views. Then meta learning is performed to build final classification model. Algorithm Multi View Classification (MVC) [16] is based on this technique. First it propagates tuple id an class labels from target relation to all non-target relations and a view combination technique is employed to build final classification model.

Artificial Neural networks (ANNs) can be used for single table classification and cannot be applied for multiple table classification directly. Algorithm Multiple View artificial Neural Networks (MVNNs) [17] bridges the gap between ANNs and relational databases. Algorithm MVNNs first propagates tuple id and class label to non-target relations. Then it uses neural networks to build classification model for each view.

Correlation based multiple view validation [18] constructs multiple views based on both target and non-target relations. This algorithm first partition attributes into multiple subsets. Then these subsets are used to construct multiple uncorrelated views, based on a correlation-based view validation method, against the target concept. These views are learned independently. Then, the knowledge possessed by multiple views are combined via meta learning algorithm a to construct final model.

A different meta learning technique was proposed in [19] that is MVC based on voting combination technique. This algorithm gives weights to each constructed views based on their accuracy. Thus individual performance of views can be considered for construction of final model. Voting is uses as meta classifier in [19]. Algorithm MVC_WV_RST [20] [21] considers success rate as a measure of benefit and running time as a measure of cost to select most appropriate and efficient classifier for particular view. It selects appropriate classifiers based on characteristics of table and give ranking based on multi criteria function using Ratio of Success Rate and Time (RST).

Some of above discussed algorithms are analyzed in terms of their accuracy in next section.

IV. COMPARATIVE ANALYSIS

Many algorithms discussed in previous sections are analyzed with respect to their accuracies in following tables and figures. Two databases Mutagenesis [22] and Financial [23] are used to test accuracies of various algorithms. Both databases are from different application domains and they have variant relational structures. They contain different numbers of tuples in the entire database having varying degree of class distribution in the target table.

Mutagenesis [24] database contains structural descriptions of Regression Friendly molecules that are to be classified as mutagenic or not. The background tables contain information about the atoms and bonds that make up the molecules. The *Atom* and *Bond* tables are linked to the target relation *Molecule* through the *Molecule Atom* table.

Financial [25] database is from financial domain and contains typical business data. It contains eight tables. The target table, i.e. the *Loan* table and the related information for each loan is stored in the tables *Account, Client, Order, Transaction, Card, Disposition* and *District*. All background tables are linked to the target table via directed or undirected foreign key chains. Classification model finds out if loan is good or bad.

Mutagenesis is comparatively smaller dataset than Financial dataset with respect to number of relations, number of attributes and number of tuples.

Table 1 and Table 2 show accuracies and running time obtained by various algorithms on Mutagenesis and Financial datasets respectively. The running times are shown mainly to give a general idea about speed of algorithms. The exact values of the running times are not comparable because of the differences in hardware and software platforms used in the different implementations.

As per accuracies shown in table 1 and table 2, Tuple Id propagation is more accurate and efficient method for relational database classification than selection graph method. It requires only small amount of data transfer among tables of given dataset.

TABLE I

ACCURACY (%) ON MUTAGENESIS DATASET

Algorithm Name	**Accuracy (%)**	**Running Time (seconds)**	**Reference**
MRDTL	87.5	52.15	[6]
MRDTL-2	87.5	28.45	[8]
CrossMine	87.7	1.92	[5]
CrossMine-Tree	75	0.66	[9]
CrossMine-Rule	89.3	2.57	[9]
Graph-NB	86.2	1.1	[10]
SRG-BC	87.24	1.7	[11]
MVC	83	3.8	[16]
MVNN	84.3	-	[17]
MVC_WV	**100**	1.75	[19]
MVC_WV_RST	94.14	**0.9**	[21]

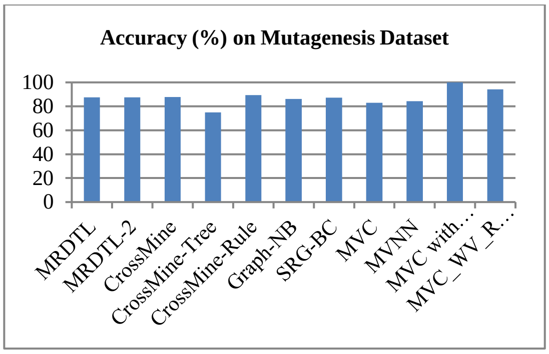

Figure 1. Accuracy(%) on Mutagenesis dataset

TABLE III

ACCURACY (%) ON FINANCIAL DATASET

Algorithm Name	**Accuracy (%)**	**Running Time (seconds)**	**Reference**
CrossMine	87.5	13.9	[5]
CrossMine-Tree	87.3	8.23	[9]

CrossMine-Rule	88.3	16.8	[9]
Graph-NB	85.25	1.9	[10]
SRG-BC	86.5	**1.28**	[11]
MVC	92.5	5.6	[16]
MVNN	93.1	-	[17]
MVC_WV	94.13	2.9	[19]
MVC_WV_RST	**94.41**	5.8	[21]

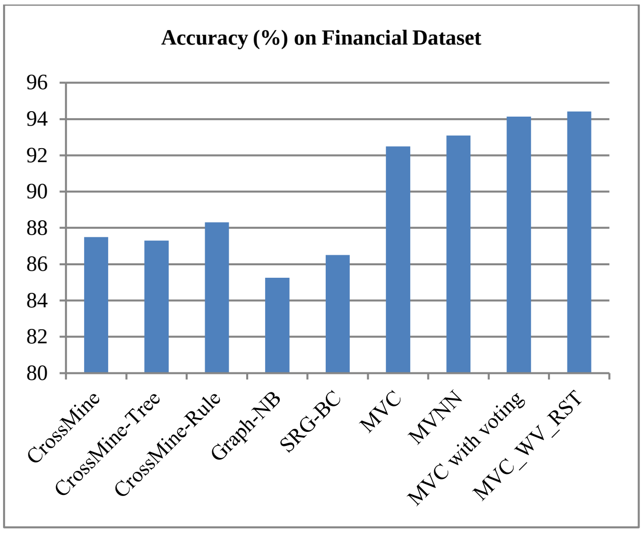

Figure 2. Accuracy(%) on Financial dataset

Furthermore multi view based algorithms are more accurate and faster. They do not require development of new classification algorithm instead of that they uses existing propositional learning algorithms for multi relational learning. Selection of different classification algorithms as per different characteristic of views is also possible in multi view learning. For financial data set, MVC_WV_RST gives highest accuracy while for mutagenesis dataset, MVC with weighted voting gives highest accuracy. Hence for both dataset multi view based tuple id propagation method is most efficient relational database based multi relational classification method.

V. CONCLUSION

In this paper, we discussed classification algorithms for multi relational database. These all algorithms are based on relational database approach where structural information of database kept intact while building classification model. We discussed selection graph and tuple id propagation methods for multi relational classification. Multi view classification also makes use of tuple id propagation method before construction of multiple views. So algorithms on multi view approach are also discussed in this paper. By comparing accuracies of different algorithms we found that multi view based tuple id propagation method is more efficient method for relational classification.

REFERENCES

[1] Jiawei Han, Micheline Kamber, *Data Mining: Concepts and Techniques 2nd*, China Machine Press, Beijing, 2006.

[2] Kramer, S., N. Lavrac and P. Flach. Propositionalization approaches to relational data mining. In S. Dzeroski and N. Lavrac, eds. *Relational Data Mining*. pp 262-291, Springer-Verlag, 2001.

[3] H. Blockeel, "Statical relational learning," handbook on Neural network Information Processing, 2013.

[4] Jing-Feng Guo, An Efficient Relational Decision Tree Classification Algorithm, Third International Conference on Natural Computation (ICNC 2007).

[5] X., Han, J., Yang, J., & Philip, S. Y. Yin, "Crossmine: Efficient classification across multiple database relations," *In Constraint-Based mining and inductive databases. Springer, Berlin, Heidelberg.*, pp. 172-195, 2006.

[6] Héctor Ariel Leiva, "MRDTL: A multi-relational decision tree learning algorithm," 2002.

[7] Knobbe, J., Blockeel, H., Siebes, A., and Van der Wallen, D. M. G. Multi-relational Data Mining. In *Proceedings of Benelearn '99*, 1999.

[8] A., Leiva, H., & Honavar, V. Atramentov, "A multi-relational decision tree learning algorithm–implementation and experiments.," 38-56, 2003.

[9] X., Han, J., Yang, J., & Philip, S. Y. Yin, "Efficient classification across multiple database relations: A crossmine approach," *IEEE Transactions on Knowledge & Data Engineering, (6)*, pp. 770-783.

[10] H., Yin, X., & Han, J. Liu, "An efficient multi-relational Naïve Bayesian classifier based on semantic relationship graph," *In Proceedings of the 4th international workshop on Multi-relational mining. ACM.*, pp. 39-48, 2005.

[11] H., Liu, H., Han, J., Yin, X., & He, J. Chen, "Exploring optimization of semantic relationship graph for multi-relational Bayesian classification," *ecision Support Systems, 48(1)*, pp. 112-121, 2009.

[12] G., Murty, M. N., & Sitaram, D. Manjunath, "A heterogeneous naive-bayesian classifier for relational databases.," *KDD, Paris.*, 2009.

[13] O., Bina, B., Crawford, B., Bingham, D., & Xiong, Y. Schulte, "A hierarchy of independence assumptions for multi-relational Bayes net classifiers," *IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*, pp. 150-159, 2013.

[14] R., Moser, F., & Ester, M. Frank, "A method for multi-relational classification using single and multi-feature aggregation functions.," *In European Conference on Principles of Data Mining and Knowledge Discovery. Springer, Berlin, Heidelberg.*, pp. 430-437, 2007.

[15] J., Liu, H., Hu, B., Du, X., & Wang, P. He, "Selecting Effective Features And Relations For Efficient Multi-Relational Classification," *Computational Intelligence, 26(3)*, pp. 258-281, 2010

[16] H., & Viktor, H. L. Guo, "Mining relational databases with multi-view learning," *In Proceedings of the 4th international workshop on Multi-relational mining, ACM*, pp. 15-24, 2005.

[19] H., & Viktor, H. L. (2006, July). Guo, "Multi-view ANNs for multi-relational classification. In Neural Networks," *IJCNN'06. International Joint Conference on (pp. 5259-5266). IEEE.*, 2006.

[20] Viktor Herna L. Guo Hongyu, "Mining relational data through correlation-based multiple view validation," *In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining ACM.*, pp. 567-573, 2006.

[21] S. Modi, "Relational classification using multiple view approach with voting," *International Journal of Computer Applications, 70(16).*, 2013.

[22] A., & Kosta, Y. P. Thakkar, "Efficient Heterogeneous Multi-relational Classification Using Multi-criteria Ranking Approach Based on Characteristics of Multiple Relations.," *JCP, 10(6)*, pp. 418-426, 2015.

[23] Kosta, Y. P. Thakkar Amit, "Improving efficiency of heterogeneous multi relational classification by choosing efficient classifiers using ratio of success rate and time," *Intelligent Automation & Soft Computing, 23(1)*, pp. 75-86, 2017.

[24] Srinivasan, A., Muggleton, S.H., Sternberg, M.J., & King, R.D. (1996). Theories for mutagenicity: A study in first-order and feature-based induction. *Artificial Intelligence, 85*, 277–299.

[25] Berka, P., (2000) Guide to the financial data set. In A. Siebes & P. Berka (Eds.), PKDD2000 Discovery Challenge.