Journal of Scientific & Industrial Research
Vol. 79, December 2020, pp. 1095-1100

# A New Improved Approach for Feature Generation and Selection in Multi-Relational Statistical Modelling using ML

Vikash Yadav[1]*, Mayur Rahul[2] and Rati Shukla[3]

[1]ABES Engineering College, Ghaziabad, Uttar Pradesh, India

[2]Department of Computer Application, UIET, CSJM University, Kanpur, India

[3]GIS Cell, MNNIT Prayagraj, Allahabad, India

*Received 14 July 2020; revised 04 September 2020; accepted 15 September 2020*

Multi-relational classification is highly challengeable task in data mining, because so much data in our world is organised in multiple relations. The challenge comes from the huge collection of search spaces and high calculation cost arises in the selection of feature due to excessive complexity in the various relations. The state-of-the-art approach is based on clusters and inductive logical programming to retrieve important features and derived hypothesis. However, those techniques are very slow and unable to create enough data and information to produce efficient classifiers. In the given paper, we proposed a fast and effective method for the feature selection using multi-relational classification. Moreover we introduced the natural join and SVM based feature selection in multi-relation statistical learning. The performance of our model on various datasets indicates that our model is efficient, reliable and highly accurate.

**Keywords**: Feature Selection, Inductive Logical Programming, Natural Join, SVM, Statistical Learning

## Introduction

Multi-relation exists in many real-world databases. The mining in the multi-relational database is very important task in many areas such as predicting business trends etc. The classification of multiple relations is one of the most difficult tasks in data mining of multi-relation database.

Classification of multiple relations is found to be very difficult problem in today's era of research due to presence of large number of entities. Further, day by day increasing of high dimensional search space also makes the classification more challenging. Moreover, complicated nature of various relations in multiple relational databasescreates lots of problem. The highly complicated database leads to high cost in feature generation and feature selection.

The existing approaches are mainly based on the Inductive logical programming and clusters to extract important features from the relational database. Those techniques are comparatively slow and searching are highly expensive in the search space. Moreover, they are unable to utilize the information stored in all types of entities. The use of SVM classifiers utilizes information more than the previous methods because they work on entities instead of rules.[1]

*Author for Correspondence
E-mail: vikash.yadav@abes.ac.in

There are so many techniques used for feature extraction in image processing, computer vision and HCI such as principal component analysis (PCA), independent component analysis (ICA), Gabor filters, layered approach, geometric features, moments invariants, Zernike moments and local binary patterns.[2–9] The SVM, decision trees and regression can be applied very effectively in single relations.[10] They cannot be applied directly to multi-relations because they create some complications. The aggregation of the data or normalisation of data is done before sending to the classification with the help of cutting edge methods such as SVM.

In our proposed work, we are able to solve the problem arises in the classification of multiple relations by proposing a general method using extraction of important features from the feature space. Our method is the combination of natural join and SVM to extract features in multi-relational feature space. The main objective of this method is to apply our framework to multi-relational data and to improve the accuracy and processing time. The performance of our method can also be applied in various datasets which indicates its better accuracy, processing time and scalability with other existing methods.

The remaining paper is organised as follows: In section 2, related works are explained, methodology is

discussed in section 3, Experiments and results are in section 4 and finally concluded in section 5.

## Related Work

The problem of Multi-relational classification is resolved generally by the inductive logical programming (ILP) and regressions.[10–12] ILP is used to represent problems in terms of background knowledge and logical database of facts. It derived some hypothesised logical programs which favours positive samples but not negative samples.[13]

The PKDD CUP 99 Financial database which we used for the experiment purpose is represented in Fig. 1.

Quinlan *et al*. proposed a set of conjunctive rules called FOIL, used to differentiate between positive samples with the negative samples. The main objective of the FOIL is to look for the finest predicate rule and add those to the candidate rule.[14] Muggleton *et al*. introduced a normal ordinary clause for each case using A* like search called PROGOL, which removes all the redundant clauses.[15] Blockeel *et al*. proposed C4.5 with the help of heuristic search and literals conjunction using tree nodes called TILDE, used to represent background observation.[16–17] The main disadvantage of above techniques is that they were expensive and slow to handle large search space of rules and clauses. Yin *et al*. developed CrossMine to produce tuple based method to avoid physical relations to some extent.[18]

Besides above all techniques, another idea is to change the multiple relation databases to single universal relation. The propositional based approaches used ILP to convert different relational data to produce features.[19] The universal database features are added using first order clauses of ILP.[20] Lavrac *et al*. proposed a propositional method which converts clauses into propositional expressions according to the determination of all the literals in the clauses.[21] Srinivasan *et al*. proposed the framework which is capable to solve problem of unknown clauses by using Boolean features in PROGOL clauses.[22] On Comparison with other existing methods, cost of computation is high and loss of information is found when Propositionalization method has applied to construct the binary features in the multiple relational databases. Krogel *et al*. proposed a method which is efficient in feature generation using aggregate function called RelAggs.[23] The accuracy is the issue which is still becoming great challenge for the researchers due to high range of relations with high range of features constructed.

Although many researchers have been done in multi-relation database, some problems like accuracy and processing time is still challenge for the researchers. Many machine learning techniques like Clustering, SVM works well in single relations but unable to get extended in multi-relations effectively. They work in feature space instead of set of rules. Propositionalization is capable of converting different relations into single relations, but the computational cost is very large which makes the classifier model more costly.

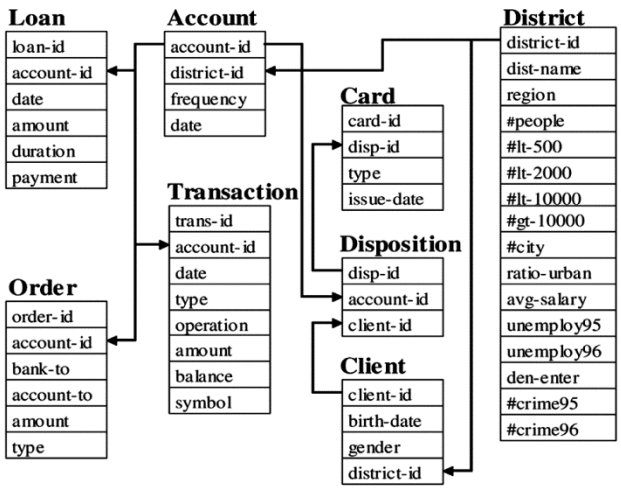


Fig. 1 — PKDD CUP 99 Financial database example[18]

## Proposed Methodology

To effectively use the general supervised learning technique (SVM) on multi relational database, we proposed following model (Fig. 2) for feature generation and selection:

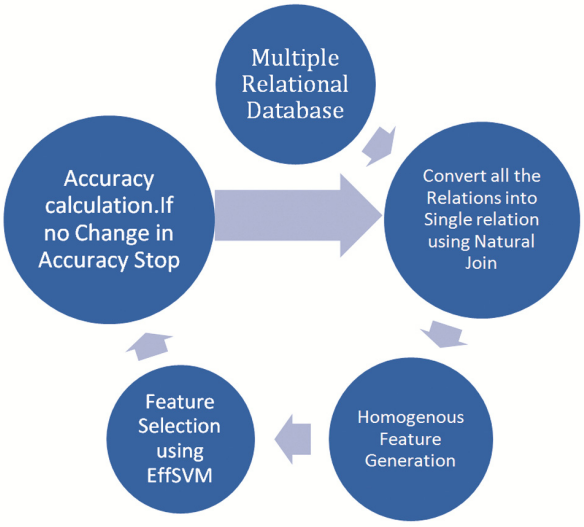


Fig. 2 — Proposed model

It usually contains four steps: (1) Converting all the relations into single relation. (2) Homogenous feature generation for the given single relation (3) Feature selection according to its usefulness in the model using EffSVM. (4) It is an iterative process; if accuracy changes repeat steps otherwise stop. The training and testing data follows n-fold cross validation rule. In this rule, all features are divided into n sets where n-1 set is used to train the model and remaining is used to test the model.

**/*Algorithm for Our Model*/**

1. Set Accuracy = 50
2. Transform all the relation into single relation using natural join
3. While TRUE
4. Feature generation based on similar type
5. Feature selection using our EffSVM
6. Accuracy calculation
7. If accuracy changes repeat steps 4 to 6 otherwise exit from the loop
8. End While

Let us start our general approach using natural join. There are n relations such that $R_1$, $R_2$, $R_3$….$R_n$. The combination of all the relations is $R_S$ (See Fig. 3).

The single relation contains large amount of attributes and therefore contains large amount of features. They are very difficult to manage. They form the search space for the classifiers. Some features in the single universal relation cannot be appropriate for the classifiers. For example, the 'date'attribute in transaction table[18] gives very low information whether the transaction happens at that date or in next date based on transaction completion. We want to develop an efficient approach to get small features which are helpful in multiple relational classifications. Preferably, features should not be similar in the single universal relation.

Many techniques have been studied in selecting feature.[24] However, majority of the techniques are made preferably for single relation. Those techniques are not suitable for the single universal relations $R_S$ because it contains so many instances for the single tuple.

To overcome this problem, we introduced homogenous-based feature extraction method for multiple relational classifications. We started with the method used by Yin *et al*. to measure homogenous-based feature generation.[25] First we calculate the similarity between tuples of the relation. Further it is used to create similarity matrix. The similarity matrices are used to represent feature similarity between tuples. This technique is used to create relation between tuple and its existing class ranges from 0 to 1. Comparison of this technique to other techniques, the results from this technique supports the feature selection in different classes for efficient classification.

Based on the homogenous-based feature generation, we propose a technique for selecting features for multiple relational classifications. We follow the method used by the Yin *et al*.[18] The homogenous-based feature generation and selection works very good under the hypothesis that the similarity factor between the tuples in the similar class is high and in other class is very low. The implementation of this hypothesis is applied on the feature selection procedurewhere all the features selection is based on the high similarity.

## Experiments and Results

In order to assess the performance of our proposed method for multiple relational classifications, we starts with building of our efficient SVM called EffSVM. We evaluate our proposed method with other existing methods in terms of accuracy and processing time. We also used our method to build with other method for proposition and compare our result with other ILP approaches in terms of processing time and accuracy.

We compare our results with some other existing approaches like RelAggs and CrossMine.[18,23] All the simulation has been done in Python. All the experiments are run on the 2.00 Ghz Intel i3 processor having 4gb ram with windows 10. We follow n-cross validation rule.

We have used three datasets for our experiment and results.[26] They are: (1) Mutagenesis (Muta), a

Student_Details

| Roll No. | Name | Address |
|---|---|---|
| 1 | Anoop | Delhi |
| 2 | Anurag | Noida |
| 3 | Ganesh | UP |

⋈

Student_Result

| Roll No. | Marks |
|---|---|
| 1 | 20 |
| 2 | 30 |
| 3 | 10 |

Student_Details ⋈ Student_Result

| Roll No. | Name | Address | Marks |
|---|---|---|---|
| 1 | Anoop | Delhi | 20 |
| 2 | Anurag | Noida | 30 |
| 3 | Ganesh | UP | 10 |

Fig. 3 — Example of Natural join

relational learning dataset. (2) Finance Database (FDB), a bank finance database (3) East-West (EW), relational learning issues in machine learning.

**Evaluation of Our Proposed Method**

In this simulation, we assess the performance of feature generation and selection in our method called EffSVM. The following methods are used to compare our results.(1) The Naïve method without using feature selection having aggregate features. (2) MulSVM used both similarity and distance based methods with feature generation and selection.

The accuracies and processing time of all the methods including our method is shown in Table 1. Our proposed method gives higher accuracy and processing time except for EW. Compare to Naïve and MulSVM, our method due to homogenous-based methods perform better than the other methods. Our experiments states that homogenous-based and natural join both in the EffSVM improve the accuracy and the processing time.

**Evaluation of Our Method with Other Propositional Methods**

We implement our proposed method with other algorithms used for proposition. We also compare it with the RelAgg and CrossMine. The accuracies and processing time are shown in Table 2.

Our proposed method works best with compare to other state-of-the-art methods except in the E-W. On comparison with our proposed method without using RelAgg with the CrossMine, accuracy increase by 7.93% and processing time decrease by 29.23% and with RelAgg, it is increases by 0.48% and 33.8% respectively. It is shown that when compare with CrossMine our proposed method works better without RelAgg as compare to our proposed method with RelAgg in Muta database. In E-W database, performance of our proposed method is not good in comparison with CrossMine.

CrossMine is able to challenge all the process without RelAgg and with RelAgg both in accuracy and processing time in both Matagenesis and Financial database but little less efficient in East-West database. This experiment is able to prove the better performance of our proposed method in respect to accuracy and processing time.

For the assessment of scalability and efficiency, we follow the method of Zou *et al*. for the construction of synthetic database.[27] We develop same relational schema with r relations. Every relation has anattribute. Every relation has a primary key which is generated randomly. Foreign keys f is generated for each primary key and tuples are form for each relation.

After design a series of database according to Zou *et al*.[27], we compare the processing time of MulSVM, CrossMine with our proposed method. The results are given in Table 3 and comparison of all these methods are given in Fig. 4. We design another series according to the work of Zou *et al*.[27] and the results are depicted in Table 4 and the comparison between various methods is shown in Fig. 5.

When number of tuples increases our proposed method slow in starting but after that it is better than the MulSVM and CrossMine depicted in Fig. 4. The

Table 1 — Performance analysis of our proposed method

| | Muta | | F-DB | | E-W | |
|---|---|---|---|---|---|---|
| | Accuracy (%) | Processing Time (Sec) | Accuracy (%) | Processing Time (Sec) | Accuracy (%) | Processing Time (Sec) |
| Naive | 86.2 | 1.4 | 87 | 69 | 80 | 0.1 |
| MulSVM[27] | 87.8 | 1.0 | 87.3 | 4.4 | 80 | <0.1 |
| Our Proposed Method | 88.4 | 0.92 | 89.1 | 4.8 | 78.6 | 0.12 |

Table 2 — Performance analysis of our proposed method with some existing models

| | Muta | | F-DB | | E-W | |
|---|---|---|---|---|---|---|
| | Accuracy (in %) | Processing Time in sec | Accuracy (in %) | Processing Time in sec | Accuracy (in %) | Processing Time in sec |
| Our Proposed Method | 88.4 | .92 | 89.1 | 4.8 | 78.6 | 0.12 |
| Our Proposed Methodwith RelAggs[23] | 82.3 | 5.7 | 83.1 | 12.35 | 74.2 | 5.3 |
| MulSVM[27] | 87.8 | 1 | 87.3 | 4.4 | 80 | <0.1 |
| RelAggs_SVM[27] | 79.8 | 7.3 | 82.6 | 125 | 80 | 9 |
| CrossMine | 81.9 | 1.3 | 87.3 | 9.7 | 80 | 0.1 |

Table 3 — Processing time of our proposed method as number of tuples increases

| | | | | | |
|---|---|---|---|---|---|
| Processing Time (Sec) | 115 | 221 | 278 | 442 | 975 |
| Number of Tuples | 10000 | 30000 | 50000 | 80000 | 100000 |

Table 4 — Processing Time of our proposed method as number of relation increases

| | | | | | |
|---|---|---|---|---|---|
| Processing Time in (Sec) | 0.5 | 2.9 | 9.1 | 10.7 | 12.5 |
| Number of Relations | 10 | 20 | 50 | 100 | 200 |

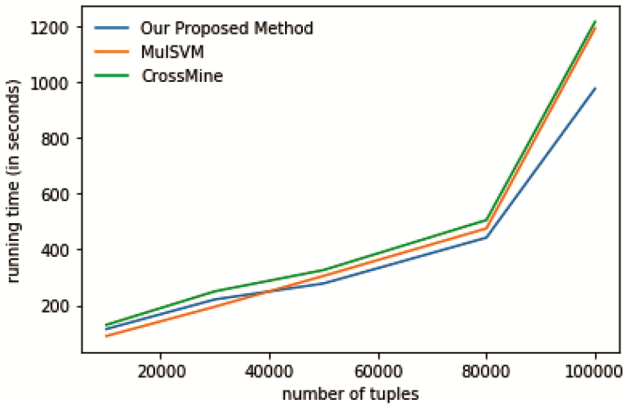


Fig. 4 — Comparison between different methods with our proposed method with increase in number of tuples

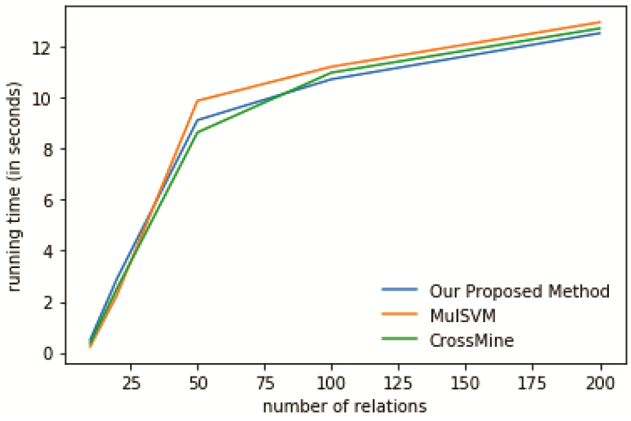


Fig. 5 — Comparison between different methods with our proposed method with increase in number of relations

number of relation increases, our proposed method works better than the other methods on the basis of processing time as depicted in Fig. 5.

## Conclusions and Future Works

In this paper, we able to proposed a method for the feature generation and selection in multiple relational classifications. We proposed a homogenous-based feature selection technique to improve the efficiency and processing time in relational database to generate feature space. A rigorous study shows that our proposed method is able to improve efficiency, accuracy and scalability with respect to other state-of-the-art. In future, we incorporate some more machine learning techniques in high dimensional dataset to improve the performance of feature generation and selection in multiple relational classifications.

## References

1 Burges C J, A Tutorial on support vector machines for pattern recognition, datamining knowledge discovery, **2(2)** (1998) 121–167.

2 Rahul M, Kohli N, Agarwal R, Mishra S, Facial expression recognition using geometric features and modified hidden markov model, *Int J Grid Util Comput*, **10(5)** (2019) 488–496.

3 Rahul M, Kohli N, Agarwal R, Layered recognition scheme for robust facial expression recognition using modified hidden markov model, *J Multimed Process Technol*, **10(1)** (2019) 18–26.

4 Rahul M, Kohli N, Agarwal R, Facial expression recognition using moment's invariants and modified hidden markov model, *Int J Appl Eng Res*, **13(8)** (2018) 6081–6088.

5 Rahul M, Kohli N, Agarwal R, Facial expression recognition using multistage hidden markov model, J *Theor Appl Inf Technol*, **95(23)** (2017) 6379–6388.

6 Rahul M, Shukla R, Yadav D K, Yadav V, Zernike moment-based facial expression recognition using two-staged hidden markov model, *Adv Comp Comm Comput Sci*, **924** (2019) 661–670.

7 Rahul M, Kohli N, Agarwal R, Facial expression recognition using local binary pattern and modified hidden markov model, *Int J Adv Intell Paradig*, **17(3-4)** (2020) 367–378.

8 Rahul M, Kohli N, Agarwal R, Partition based feature extraction technique for facial expression recognition using multi-stage hidden Markov model, *J Appl Eng Sci* **13(9)** (2018) 2651–2658.

9 Rahul M, Kohli N, Agarwal R, Facial expression recognition using local multidirectional score pattern (LMSP) descriptor and modified hidden Markov model, *Int J Adv Intell Paradig*, (In Press).

10 Lavrac N, Dzeroski S, *Inductive logic programming: Techniques and applications* (Ellis Horwood, New York) 1994.

11 Muggleton S, Inductive logic programming, *New Generat Comput*, **8(4)** (1991) 295–318.

12 Muggleton S, Raedt L, Inductive logic programming: Theory and methods, *J Log Program*, **19** (1994) 629–679.

13 Lloyd J W, *Foundations of Logic Programming*, (Springer, New York) 1987.

14 Quinlan J R, Cameron-Jones R M Foil, A Midterm Report, in *ECML 1993*, edited by Brazdil P B, LNCS, 667 (Springer, Heidelberg) 1993, 3–20,.

15 Muggleton S, Inverse entailment and progol, *New Generat Comput*, **13** (1995) 245–286.

16 Blockeel H, Raedt L D, Top-down induction of first-order logical decision trees, *Artif Intell*, **101(1-2)**, (1998) 285–297.

17 Quinlan J R, *C4.5 Programs for Machine Learning* (Morgan Kaufmann, California) 1993.

18 Yin X X, Han J W, Yang J, Yu P S, Crossmine: Efficient Classification across Multiple Database Relations, in *ICDE*, 2004, 399–411.

19 Kramer S, Relational learning VS Propositionalization: Investigations in Inductive Logic Programming and Propositional machine learning, Technical report, Vienna University of Technology, 1999.
20 Quinlan J R, Induction of Decision Trees, *Mach Learn*, **1** (1986) 81–106.
21 Lavrac N, Principles of Knowledge Acquisition in Expert Systems. PhD thesis, Facultyof Technical Sciences, University of Maribor, 1990.
22 Srinivasan A, King R D, Feature construction with inductive logic programming: A study of quantitative predictions of biological activity aided by structural attributes, *Data Min Knowl Discov*, **3(1)** (1999), 37–57.
23 Krogel M A, On Propositionalization for knowledge discovery in relational databases, PhD thesis, Fakultat fur Informatik, Germany, 2005.
24 Molina L C, Belanche L and A Nebot, Feature selection algorithms: a survey and experimental evaluation, 2002 Proc IEEE Int Conf on Data Mining (Maebashi City, Japan) 2002, 306–313.
25 Yin X X, Han J W & Yu P S, Cross-Relational Clustering with User's Guidance, in *KDD 2005*, 2005.
26 http://www.cs.waikato.ac.nz/ml/proper/datasets.html.
27 Zou M, Wang T, Li H & Yang D, A general multi-relational classification approach using feature generation and selection, *ADMA 2010*, **Part II**, LNCS 6441, 2010, 21–33.