Extraction of Association Rule Mining using Apriori algorithm with Wolf Search Optimisation in R Programming

Garima Jain, Diksha Maurya

Abstract*: Association rules mining (ARM) is a standout amongst the most essential Data Mining Systems. Find attribute patterns as a binding rule in a data set. The discovery of these suggestion rules would result in a mutual method. Firstly, regular elements are produced and therefore the association rules are extracted. In the literature, different algorithms inspired by nature have been proposed as BCO, ACO, PSO, etc. to find interesting association rules. This article presents the performance of the ARM hybrid approach with the optimization of wolf research based on two different fitness functions. The goal is to discover the best promising rules in the data set, avoiding optimal local solutions. The implementation is done in numerical data that require data discretization as a preliminary phase and therefore the application of ARM with optimization to generate the best rules.

***Index Terms*: Association Rule Mining, Apriori algorithm, Fitness Function, Wolf search optimization.**

I. INTRODUCTION

Without a doubt, now the Internet and the digitization of all information have transformed the way we share knowledge. This transformation of knowledge on the digital environment leads to the rapid growth of the data repository in which we store information for various operations such as data compression, data mining and data analysis, etc. These are the challenges in the big data area. These days, information mining is a fundamental procedure with various organizations from dissimilar sources. ARM is one of the important tasks and the famous research area in data mining. ARM is implemented in an extensive variety of requests, such as card transactions, supermarket telecommunications, intruder detection, insurance claims, banking services, web mining [1]. Association rules are defined by the IF / THEN declarations for signifying repeat relationships between data set attributes. First proposal by Rakesh Agrawal, Tomasz Imieliński and Arun Swami [2]. In general, the goal of any optimization problem [3] is to find:

$$x^{opt} = \max_{x \epsilon x} f(x) \qquad (1)$$

Where x^{opt} is the optimal value of the data set and there is a fitness function f (x) to judge the results. The technique which is well known as global optimum that symbolizes a better x^{opt} solution that should be present in the space of the specified problem. It is not necessary that every time real life situations occur, an optimization function has a good mathematical performance, so it is a well-known problem in the search for an optimal global solution. The number of local minibuses increases exponentially when the number of dimensions increases in which the data sets have high dimensional variables, in these cases it is exhaustive to find the optimal overall value. The information mining instruments connected in the instructive information were Orange, Weka and R Studio [13]. By optimizing the problem, when the size of problems increases, the search space of the candidate solution also increases more than exponentially, leading to a comprehensive search for the optimal global solution.

The following sections of this paper are defined as:- In section 2, we provide associated work on the mining association and various optimization techniques. In section 3, the hybrid algorithm is presented. Section 4 presents the computational outcomes of the selected fitness function and other parameters. The conclusion is discussed in section 5.

II. THEORETICAL BACKGROUND

This segment gives a short audit of the four primary areas which structure the background of the work: Association Rule Mining, Apriori, nature-inspired optimization and wolf search optimization.

A. Association Rule Mining

Following equation is a representation of an association rule is in the form of:

$$X \rightarrow Y \qquad (2)$$

Where given X and Y are transactions or we also denote as an items. The first part X is defined as antecedent (IF) and other one Y is called consequent (THEN). It means IF X appears then how supportively Y appears. To find association rule, user must define their threshold value of interest. There are two thresholds values support and confidence. These are two most famous parameters and most studied in literature.

Support: It is an evidence of how frequently the items appear in the database or in set of transactions. [4]

$$supp(X) = X/T \qquad (3)$$

Revised Manuscript Received on July 5, 2019.

Garima Jain, Swami Vivekanand Subharti University, Meerut, India.

Diksha Maurya, Swami Vivekanand Subharti University, Meerut, India

Where X is item set in a transaction and T is total number of the transactions.

Confidence: [4] It interprets that X→ Y rule with detail to a set of transactions is the ratio of the transactions that covers X also comprises Y. It is likewise characterized:

$$conf(X \rightarrow Y) = \frac{supp(XUY)}{supp(X)} \qquad (4)$$

Where Y is also an item set, XU Y is union of X and Y.

B. Apriori Algorithm

For the extraction of sets of frequent elements, the Apriori algorithm is the most consolidated and vital, it was proposed by R. Agrawal and R. Srikant in 1994 [5]. The key idea of the Apriori algorithm is to create several repetitions in the database to find the rules of different groups of elements. An iterative approach that is primarily research is used throughout the search space, where groups of k elements are used to explore sets of elements (k + 1).

The main part of the Apriori algorithm is that all the non-empty subsets of a set of frequent elements must be frequent. He also declares that if a subset cannot pass through the minimum support threshold, then its entire superset cannot pass through the process. The performance is defined and proposed of PSO-SVM and is then compared with the various surviving feature selection algorithms such as Info gain, Chi-squared[14].

C. Related work with Optimizing Algorithms

To obtain the most appropriate association rules from the data sets, many optimization approaches have been used in the literature. Nature works for one of the best inspirations for the development of optimization algorithms inspired by nature.

These nature-inspired methodologies neglect the ideal neighborhood arrangement and spotlight on meta-heuristics to locate the most ideal arrangement. There are different types of various algorithms have been established that signify the stimulation for the optimization of particle swarms [2], the optimization of the colony of ants [6], the bat [7], the search for the cuckoo [8], l Optimization of bacterial colonies [4] and the wolf research optimization [9].

III. Wolf Search Optimization

One of the most common predators that always hunt in packs is known as Wolves. They regularly travel as combined family unit, not at all like Particle Swarm and Ants, which more often than not move in very expansive groups. [3]

Hence Wolves do their hunt very silently. Wolves move in a very loosely coupled group but they prey for food individually. They do not have any communication between them, which will reduce the searching time. Wolves often locate prey from distance of miles away by scent as they have an amazing sense of smell. Every individual wolf has a detecting space that creates a sensing radius or coverage zone named as visual distance Wolves move in Brownian motion when they target their prey. Each step of wolf movement is generally smaller than the visual distance[3].

This algorithm is implemented in the field of attribute reduction in classification[10] and with Ephemeral memory[3]

A. A Wolf search Algorithm

```
Objective function f(x), x=(x1,x2,..xd)^T
Initialize the population of wolves, xi(i=1,2,..,W)
Define and initialize parameters:
r = radius of the visual range
s = step size by which a wolf moves at a time
α = velocity factor of wolf
pa = a user-defined threshold [0..1], determines how frequently an enemy appears
WHILE ( t < generations && stopping criteria not met)
  FOR i=1:W // for each wolf
    Prey_new_food_initiatively();
    Generate_new_location();   // check whether the next location suggested by the random number generator is new. If not, repeat generating random location.
    IF(dist(xi,xj) < r  &&  xj is better as f(xi) < f(xj))
      xi moves towards xj  // xj is a better than xi
    ELSE IF
      xi = Prey_new_food_passively();
  END IF
  Generate_new_location();
    IF(rand() > pa)
     xi = xi + rand() + v; // escape to a new pos.
    END IF
  END FOR
END WHILE
```

In this Algorithm the authors have proposed to optimize the results of The Apriori with wolf search optimization.

IV. Methodology

The Flow diagram for the methodology is show in Figure 1 that shows how the various steps are performed.

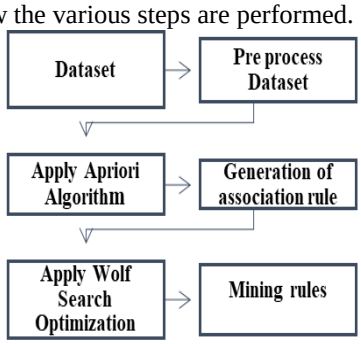

Fig. 1. Methodology

The dataset is imported in the Integration Development Environment (IDE) which is then pre-processed to remove the missing values. Thereafter, Apriori algorithm is applied to get frequent item sets and association rules. The rules are then optimized with wolf search algorithm to get globally best association rules. Fitness function play a significant role in optimization approaches to mine the best fit rules.

Fitness functions are problem dependent. The need is to get optimum fitness value, which can be achieved by evaluating fitness function for each search agents in wolf search optimization.

$$fitness function\ 1 = confidence(i) \times \log(support(i) \times length(i) + 1) \quad \ldots\ldots\ldots(5)$$

Equation (1) is a fitness function discussed in [11] which is applied to PSO. The parameter length is a marker of standard's unpredictability (bigger length shows progressively complex rules) which is equivalent to the quantity of things in precursor.[12]

$$fitness\ function\ 2 = w1 * support\ (i) + w2 * confidence(i) \quad \ldots\ldots (6)$$

Equation (2)is the fitness function used in Bacterial colony optimization where w1 = w2 and w1 + w2 =1 [4].

V. IMPLEMENTATION RESULTS

The Apriori algorithm with wolf search optimization is implemented in R studio (IDE) with R programming language version3.2. The numeric dataset is imported in IDE. The dataset wine is taken from UCI Repository [11].

Wine dataset has 13 attributes and 178 instances**.**

Table1. No. of rules generated on application of Apriori algorithm.

support	confidence	number of rules
0.02	0.01	243
0.02	0.02	175
0.02	0.03	148
0.02	0.04	139
0.02	0.05	123
0.02	0.06	103
0.02	0.07	97

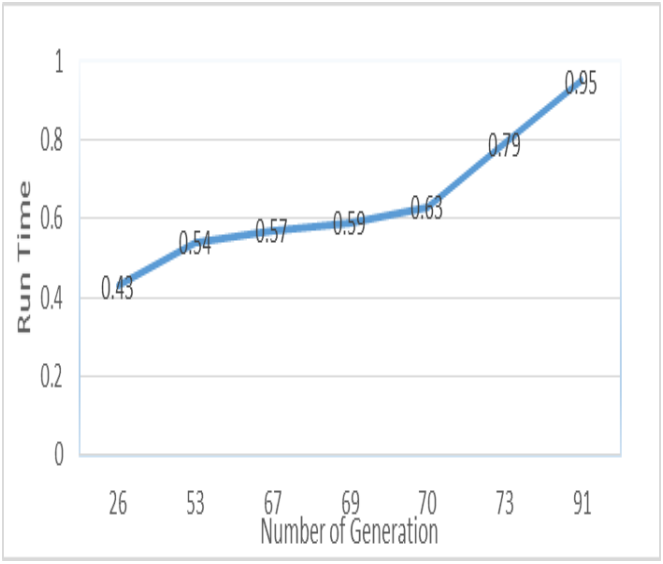

Fig. 2. Representation of Number of wolf population v/s CPU runtime.

Table 2. Globally Best Association Rules

Number of Association	Association Items	BestFitness
26	{Alcalinity of ash=18.5, Proline=675} => { class identifier=3}	0.01123596
53	{Hue=1.23, Proline=428} => {Alcalinity of ash=16	0.17839426
67	{class identifier=3, Alcalinity of ash=18.5 } => {Proline=675}	0.49274932
69	{Magnesium=87, Proanthocyanins=1.87} => {Alcalinity of ash=19}	0.56892696
70	{class identifier=2, Proanthocyanins=1.87} => {Hue=0.93}	0.27453784
73	{Alcalinity of ash=20, Nonflavanoid phenols=0.53} => {class identifier=3}	0.61948525
91	{class identifier=2, Magnesium=88, Hue=1.04,OD280/OD315 of diluted wines=2.77} => {Alcalinity of ash=18.5}	0.57829423

VI. CONCLUSION

This research is defined in terms to propose a significant methodology in the field of association rule. A new nature inspired WSO is applied to optimize the association rule. The research concludes that WSO will able to find the globally best association rules among the various associations which is extracted from dataset. This study shows that WSO is efficient optimizing algorithm for the association rules. In future we will work on large dataset and compare it with other algorithms.

REFERENCES

1. GL. Tsay, R. S. Sreenivas, and R. larry W, "Scalable Association Rule Mining with Predication on Semantic Representations of Data," pp. 180–186, 2015.
2. Mayank Agrawal, M. Manuj, and P. S. S. Kushwah, "Association Rules Optimization using Improved PSO Algorithm," pp. 395–398, 2015.
3. R. Tang, S. Fong, X. S. Yang, and S. Deb, "Wolf search algorithm with ephemeral memory," 7th Int. Conf. Digit. Inf. Manag. ICDIM 2012, pp. 165–172, 2012.
4. D. S. Cunha, R. S. Xavier, D. G. Ferrari, L. N. De Castro, and S. Paulo, "Association Rule Mining using a Bacterial Colony Algorithm," 2015.
5. K. S. Kumar and R. M. Chezian, "A Survey on Association Rule Mining using Apriori Algorithm," Int. J. Comput. Appl., vol. 45, no. 5, pp. 47–50, 2012.
6. Manju and C. Kant, "Mining association rules directly using ACO without generating frequent itemsets," Int. Conf. Energy Syst. Appl. ICESA 2015, no. Icesa, pp. 390–395,
7. X. S. Yang, "A new metaheuristic Bat-inspired Algorithm," Stud. Comput. Intell., vol. 284, pp. 65–74, 2010.
8. N. Optimisation, "Engineering optimisation by cuckoo search Xin-She Yang*," vol. 1, no. 4, pp. 330–343, 2010.
9. I. E. Agbehadji, S. Fong, and R. Millham, "Wolf Search Algorithm for Numeric Association Rule Mining," no. ICCKE, pp. 1–5, 2016.
10. W. Yamany, E. Emary, and A. E. Hassanien, "Wolf search algorithm for attribute reduction in classification," IEEE SSCI 2014 - 2014 IEEE Symp. Ser. Comput. Intell. - CIDM 2014 2014 IEEE Symp. Comput. Intell. Data Mining, Proc., pp. 351–358, 2015.
11. R. J. Kuo, C. M. Chao, and Y. T. Chiu, "Application of particle swarm optimization to association rule mining," Appl. Soft Comput., vol. 11, no. 1, pp. 326–336, 2011.
12. T. Watanbe, A. Monden, Y. Kamei, and S. Morisaki, "Identifying Recurring Association Rules in Software Defect Prediction.," 2016.
13. Hussain, Sadiq, et al. "Classification, clustering and association rule mining in educational datasets using data mining tools: A case study." Computer Science On-line Conference. Springer, Cham, 2018.
14. Vijayashree, J., and H. Parveen Sultana. "A Machine Learning Framework for Feature Selection in Heart Disease Classification Using Improved Particle Swarm Optimization with Support Vector Machine Classifier." Programming and Computer Software 44.6 (2018): 388-397.

AUTHORS PROFILE

Garima Jain was born on November 3, 1992. She has received her M.Tech. degree from Galgotias College of Engineering, in 2017. Presently she is working in Swami Vivekanand Subharti University, Meerut. She has 2 year of academics experience. She has worked with industriy. She has good understanding of emerging technologies like R Language and Python. She is also a NAAC coordinator at College Level. She is also completed a Swayam courses. She also Attend Faculty Development Programme (FDP) -2019 organized by DTU Delhi on Machine Learning in Pattern Recognition. She has published various papers in National & International Conference and Journals. She is also awarded with a Best paper Presentation Award in year 2017.

Diksha Maurya was born on 15th December 1992. She has received her M.Tech. degree from PEC Engineering College Punjab, in 2017. Presently she is working in Swami Vivekanand Subharti University, Meerut. She has 2 year of academics experience. She has attended many workshops. She has a good Command on C language, Python, R Language. She has completed her M.Tech degree in Cyber Security. She is always very curious to work in Security related Area. She has deals with various under graduate and post graduate courses.