International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-2, July 2019

Hybrid ACO-PSO-GA-DE Algorithm for Big Data Classification

Anju Bala, Priti

Abstract: ***This paper designs a technique to classify big data efficiently. This work considers the processing of big data as an optimization problem due to the trade-off between accuracy and time and solves this optimization problem by using a meta-heuristic approach. The HAPGD (Hybrid ACO (Ant Colony Optimization), PSO (Particle Swarm Optimization), GA (Genetic Algorithm), and DE (Differential Evolution)) classification algorithm is designed by using the support vector machine (SVM) along with hybrid ACO-PSO-GA-DE algorithm that hybrids exploration capability of ACO with exploitation capability of PSO whose balance is maintained using modified GA. The GA has been modified by using the DE algorithm. The presented technique performs classification efficiently as shown in results on seven datasets using different analysis parameters due to balanced exploration and exploitation search with fast convergence.***

Index Terms**: *Accuracy, ACO, Big Data, Classification, DE, GA, PSO.***

I. INTRODUCTION

Technological advancement and enhanced social media usage have led to a huge generation of data. The efficient processing of the data is the basic need for any company to survive in this competitive world[1]. The efficient processing of big data is a challenging task due to seven V's property of big data that includes volume, variety, velocity, value, veracity, visualization and variability [2][3]. The challenge of processing big data becomes critical due to the trade-off between different processing parameters as per the properties of big data. The processing of big data with better classification accuracy gets results in large evaluation time [4][5]. This leads to process the big data as an optimization problem due to optimization required between the processing parameters of big data [6]. To find an approximate solution to hard combinatorial optimization problems, a lot of Meta heuristic-based approaches were suggested. ACO is one of the most popular and efficient Metaheuristic algorithms which explores and exploits the solution space to find an accurate and optimal solution in less amount of time[7]. ACO is based on the foraging behavior of real ants. In searching for the shortest path, ants randomly explored the area around their nest which is taken as solution search space. In order to recognize the quantity and quality of food, a chemical substance is pheromone is used as a basis for ACO, which converges the search towards high-quality regions. An optimal solution is constructed by selecting the solution components one after the other from the set of solutions. A pheromone model (probabilistic distribution) is a set of pheromone values which are assembled with solution components to converge the search towards global optima. New generated candidate solution is used to update the pheromone model for the next phase of updated solution reaching towards the optimality[7]. PSO (Particle Swarm Optimization) is a Metaheuristic technique which didn't provide a guarantee for achieving the optimal solution of any complex optimization problem yet it proved as an efficient aid in achieving the optimal solution[8]. PSO is based on the behavior of animal societies where leadership is absent like flocking of birds, schooling of fishes, etc. Any of the animal, who is closest to the food source becomes the potential solution or particle. Every particle keeps its position and velocity updated, to acquire optimality. While exploring the search space, particles' movements are guided by the by their own best position (pbest) and overall best position among the whole population (gbest). Fitness values are used for constructing the solution by comparing fitness value with pbest and gbest and selecting the one which one is minimum. Particle's velocity and positions go on updating until the desired best solution is achieved. Flying of particles across the search space is adjusted in terms of position and velocity by clubbing together the history of its own best and other members of swarms. In this way, particles' position and velocity go on updating until the desired outcomes are achieved[9]. The ACO, as well as PSO, are promising techniques to solve the optimization problems. However, the hybridization of these techniques along with various other techniques had shown effective results. This paper proposes a hybrid technique by inculcating the ACO, PSO, GA, and DE to solve the optimization problem discussed in the next section.

II. HYBRID ACO-PSO-GA-DE ALGORITHM

The main motive of this hybrid algorithm is to remove the limitation of individual algorithms by combining the strengths of each algorithm. The exploration strength of ACO is combined with exploitation property of PSO which are being balanced by the GA[10]. The crossover phase of GA is replaced DE perturbation phase to improve the convergence of the overall algorithm to produce better results in less iteration. The ACO combined with the GA-DE gives the fast

Revised Manuscript Received on 30 July 2019.
***** Correspondence Author
Anju Bala*, Research Scholar, Department of Computer Science and Applications, M.D.U, Rohtak, India.
Priti, Assistant Professor, Department of Computer Science and Applications, M.D.U, Rohtak, India.

converging algorithm that explores the search space efficiently while the PSO combined with GA-DE results in an algorithm that exploits the search space effectively. The random initiation problem of the GA is suppressed by passing vectors processed by ACO and PSO. Overall, randomly generated vectors are processed for one iteration by the ACO and PSO to explore and exploit the search space respectively. Then, the solution of the first iteration is passed to the GA which performs the Selection and mutation operation on vectors. Then the resultant vectors undergo the differential perturbation to generate input for the ACO and PSO. This step is repeated until the algorithm converges. The complete details can be understood by the algorithm given in the next subsection.

A. *Hybrid ACO-PSO-GA-DE Algorithm*

1. Perform Initialization i.e. initiate ACO and PSO parameters and initiate number of iteration (nitr) to 0.
2. Initiate Random vector say H

$$H = [H_1 \quad H_2 \quad \cdots \quad H_n]$$

// this vector act as an initial vector for ACO as well as PSO

3. H_a=Apply ACO on H
4. Update Pheromone Value
5. H_p=Apply PSO on H
6. Update local and global path
7. H_a=QuickSort(H_a)
8. H_p=QuickSort(H_p)
9. H_1= H_a(1)
10. H_2= H_p(1)
11. F1=f(H_1)
12. F2=f(H_2)
13. While(nitr<MAX_ITERATION && change<e^{-10})
    a. $U_1 = \frac{H_1 - LB_a}{UB_a - LB_a}$
    b. $H_{1n} = H_1 + U_1 * (UB_a - LB_a)$
    c. $U_2 = \frac{H_2 - LB_p}{UB_p - LB_p}$
    d. $H_{2n} = H_2 + U_2 * (UB_p - LB_p)$
    e. $r_1 = rand(1:n)$
    f. $r_2 = rand(1:n)$
    g. $H_{1n} = H_{1n} + rand * (H_{r1a} - H_{r2a})$
    h. $H_{2n} = H_{2n} + rand * (H_{r1p} - H_{r2p})$
    i. H_a=Apply ACO on H_{1n}
    j. Update Pheromone Value
    k. H_p=Apply PSO on H_{2n}
    l. Update local and global path
    m. H_a=QuickSort(H_a)
    n. H_p=QuickSort(H_p)
    o. H_1= H_a(1)
    p. H_2= H_p(1)
    q. Change=F1-f(H_1)+F2-f(H_2)
    r. F1=f(H_1)
    s. F2=f(H_2)
    t. nitr++;

    End while
14. If(F1>F2)

        Return H1

    Else

        Return H2

    End if

The above algorithm explains the hybrid ACO-PSO-GA-DE algorithm that is capable to solve optimization problems efficiently. In this algorithm, A random vector is generated which act as an input to ACO as well as PSO whose parameters are initiated first. The first iteration updates the H to H_a by using ACO and H to H_p by using PSO algorithms. Then, the quick sort is applied to sort the components (vectors) of H_a and H_p based on their fitness values. Then the first i.e. best vector is selected as H_1 and H_2 from H_a and H_p respectively to be processed for other iterations. The fitness values calculated for these vectors are F1 and F2. The selection, mutation and differential perturbation is applied on these vectors to generate the H_{1n} and H_{2n} vectors which undergo ACO and PSO process respectively to generate H_a and H_p vectors respectively. The change in the fitness values of the updated and existing vectors is calculated. The process is repeated until the change minimizes or the number of iterations exceeds maximum iterations. The best vector from the resultant vector is the output of the algorithm. This algorithm is applied to the data mining discussed in the next section.

III. HAPGD CLASSIFICATION ALGORITHM

The algorithm discussed in the previous section i.e. hybrid ACO-PSO-GA-DE is applied in data mining to perform the classification. The **H**ybrid **A**CO-**P**SO-**G**A-**D**E (HAPGD) classification algorithm classifies the elements by using the SVM classifier along with hybrid meta-heuristic. In this algorithm, Feature selection is applied to the dataset by using the hybrid ACO-PSO-GA-DE algorithm. Then, the dataset is classified into K equal parts where the value of K depends upon the system configuration and the Dataset size. Then, the classification process using SVM classifier is applied to the dataset K times. Each time one part is selected for the testing and remaining for the training purposes. The overall classification accuracy is the average of accuracy achieved in each case. The overall process of HAPGD classification algorithm is also stated in fig 1.

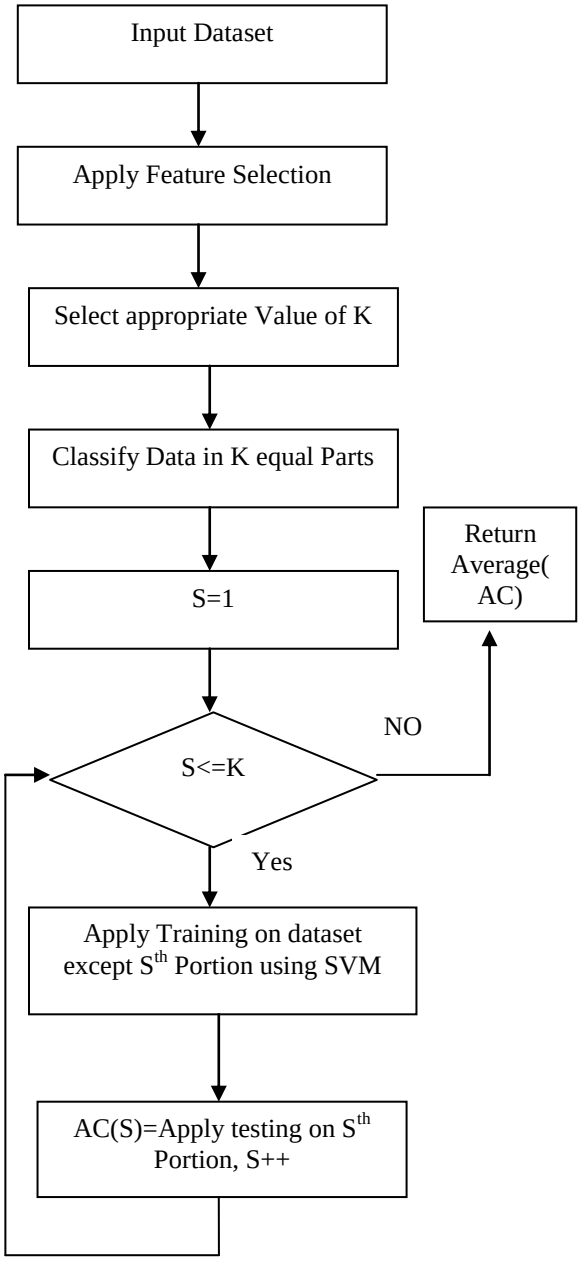

Fig 1: HAPGD Classification Algorithm

The process of HAPGD algorithm defined in the fig 1 is used to perform the classification uses the hybrid meta-heuristic technique to select the significant feature of the dataset. This step is necessary to remove the irrelevant and redundant feature of the dataset resulting improved performance of algorithm. The feature selection is most important as the irrelevant and redundant features degrade the performance of classification. Then, to improve the reliability of algorithm k-fold cross validation is applied by using the SVM classifier to compute the classification analysis parameters of the algorithm. The step by step description of HAPGD algorithm is given in the algorithm mentioned in next subsection.

A. HAPGD Classification Algorithm

1. Input Dataset
2. Initiate Random vector say H
   $H = [Attributes\ of\ dataset]$
3. H_a=Apply ACO on H to select different attribute of dataset
4. Update Pheromone Value
5. H_p=Apply PSO on H to select different attribute of dataset
6. Update local and global path
7. $H_1 = H_a$
8. $H_2 = H_p$
9. $F1=f(H_1)$
10. $F2=f(H_2)$
11. While(nitr<MAX_ITERATION && change<e-10)
    a. $U_1 = \frac{H_1 - LB_a}{UB_a - LB_a}$
    b. $H_{1n} = H_1 + U_1 * (UB_a - LB_a)$
    c. $U_2 = \frac{H_2 - LB_p}{UB_p - LB_p}$
    d. $H_{2n} = H_2 + U_2 * (UB_p - LB_p)$
    e. $r_1 = rand(1:n)$
    f. $r_2 = rand(1:n)$
    g. $H_{1n} = H_{1n} + rand * (H_{r1a} - H_{r2a})$
    h. $H_{2n} = H_{2n} + rand * (H_{r1p} - H_{r2p})$
    i. H_a=Apply ACO on H_{1n}
    j. Update Pheromone Value
    k. H_p=Apply PSO on H_{2n}
    l. Update local and global path
    m. H_a=QuickSort(H_a)
    n. H_p=QuickSort(H_p)
    o. $H_1 = H_a(1)$
    p. $H_2 = H_p(1)$
    q. Change=F1-f(H_1)+F2-f(H_2)
    r. F1=f(H_1)
    s. F2=f(H_2)
    t. nitr++;

    End while
12. If(F1>F2)
    D1=Select features having value 1 in round(H_1)
    Else
    D1=Select features having value 1 in round(H_2)
    End if
13. Apply SVM classification using K fold cross-validation to compute classification analysis parameters.

The above algorithm should classify the big data effectively, the algorithm has used the fitness function discussed in the next subsection.

B. Fitness Function

The HAPGD classification algorithm has used the following fitness function to classify the data effectively.

$$f = \theta_1 * senstivity + \theta_2 * accuracy + \theta_3 * \frac{1}{selection\ size\ ratio}$$

Here, $\theta_1 + \theta_2 + \theta_3 = 1$. Here the values of $\theta_1, \theta_2, \theta_3$ are taken to be 0.35, 0.35 and 0.3 respectively on the

experimental basis. The implementation and the result analysis of the HAPGD algorithm have been done in the next section.

IV. RESULT AND DISCUSSION

The algorithm has been analyzed on 7 different datasets downloaded from the UCI repository [11]. The performance comparison has been done with five states of art hybrid algorithms using sensitivity, specificity, accuracy and selection size as the parameters. The detail of datasets including dataset name along with its resource, attributes, and instances are given in table 1. Moreover, the total number of elements available in the dataset is also given in table 1.

Table 1: Datasets Description

S. No.	Dataset Name	Attributes	Instance	Total Elements	Resource
1	AUSTRALIAN	14	690	9660	UCI Repository
2	WDBC	30	569	17070	
3	PIMA	8	768	6144	
4	ARRHYTHMIA	452	279	126108	
5	ADVERTISEMENT	1558	3279	5108682	
6	HAR	561	10299	5777739	
7	MADELON	500	2000	1000000	

Table 1 exhibits the description of the datasets used in the work to analyze the performance of HAPGD algorithm for classification. The analysis parameters i.e. accuracy, sensitivity, specificity and selected feature ratio described in the next subsection.

A. *Accuracy*

Accuracy represents the number of instances correctly classified by the classifier. It is given as:

$$A = \frac{1}{n} * Correctly\_classified\_intsnaces \quad (1)$$

Here, n is the number of instances in the dataset

B. *Sensitivity*

It gives the correctly classified true instances. In other words, it is sensitivity to the correct classification. The sensitivity can be represented by (2).

$$Senstivity = \frac{TP}{TP+FN} \quad (2)$$

Here TP, FN shows true positive and false negative respectively.

C. *Specificity*

It is the correctly classified negative instances. It can be given by (3).

$$Specificity = \frac{TN}{TN+FP} \quad (3)$$

Here TN, FP denotes true negative and false positive respectively.

D. *Selected feature ratio*

It is the ratio of the number of features selected to the total number of features. It is given by (4):

$$SFR = \frac{NumberofselectedFeatures}{n} \quad (4)$$

Here, n is the number of features in the dataset.

V. PERFORMANCE ANALYSIS

The performance of HAPGD algorithm is compared with hybrid meta-heuristic algorithms i.e. ACO-PSO [12], ACO-GA [13], PSO-GA [14], GA-DE[15] and ACO-PSO-GA[10] using the parameters described above on 7 datasets given in table 1 has been done in this section. The results are evaluated by executing the algorithms 20 times and taking the average of results.

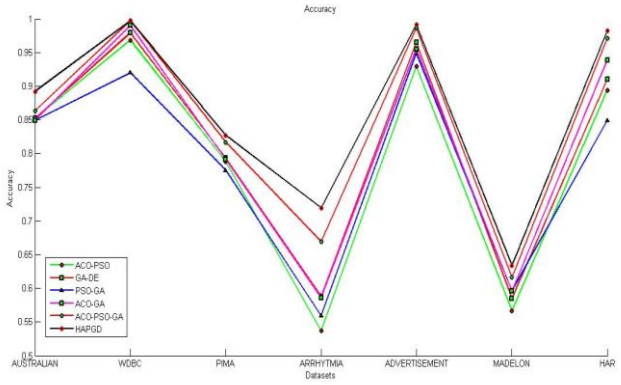

Fig 2: Comparison of Accuracy on Different Datasets

The initial parameter setting has been done as per the referred paper for each algorithm. The SVM classifier is used with RBF kernel having sigma=15. The accuracy is given in fig 2The comparison of classification accuracy on different datasets of HAPGD algorithm with other hybrid algorithms is shown in fig 2. The HAPGD algorithm exhibits more accuracy on each dataset as compared to other existing state of art algorithms. This is due to balancing of exploration phase (due to ACO) and exploitation phase (due to PSO) by the GA. The comparison of ratio for a number of features selected to the total number of features is given in fig 3.

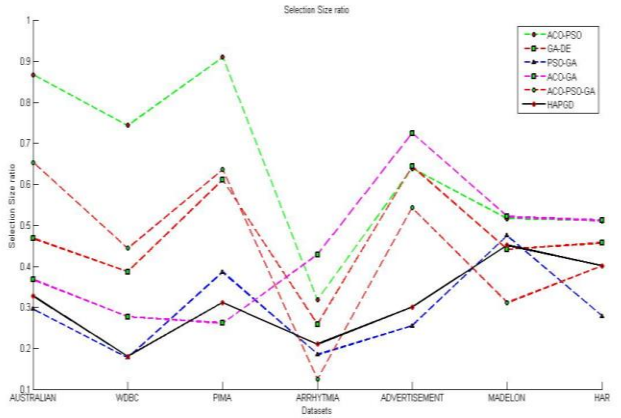

Fig 3: Comparison of Selection Size ratio on Different

Datasets

Fig 3 compares the selection size ratio of the hybrid algorithm on different datasets. It can be seen that the selection size of the ACO-PSO-GA-DE is high as compared to other few hybrid algorithms. It means higher accuracy of ACO-PSO-GA algorithm costs the number of features. It may include the significant features that are ignored by the other existing algorithms.

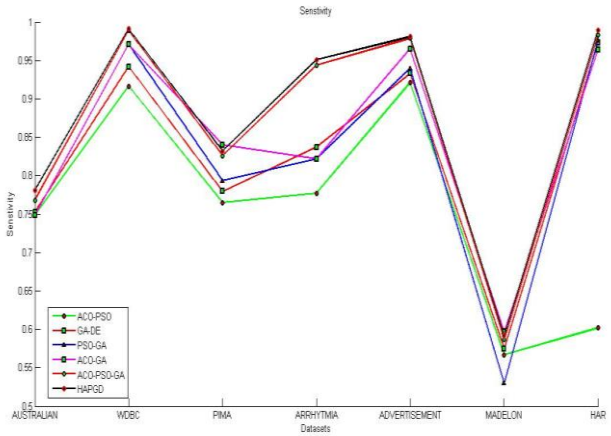

Fig 4: Comparison of Sensitivity on Different Datasets

The comparison clearly shows the sensitivity of the HAPGD algorithm is better as compared to other hybrid techniques. However, the exception is shown for the PIMA dataset.

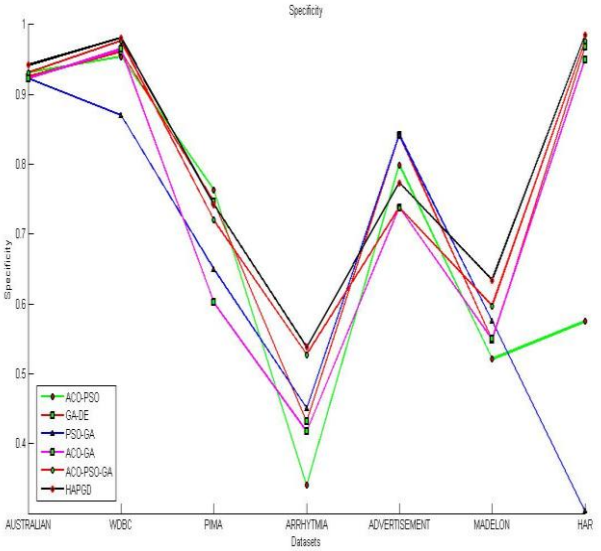

Fig 5 Comparison of Specificity on Different Datasets

The comparison in fig 5 clearly denotes that the specificity of HAPGD algorithm is better as compared to other hybrid algorithms due to more balanced exploration and exploitation search.

VI. CONCLUSION

This paper presents HAPGD algorithm to classify the big data and analyze the same on seven datasets with varying size downloaded from the UCI repository. The comparison has been done with other states of art techniques including ACO-PSO, GA-DE, PSO-GA, ACO-GA, and ACO-PSO-GA based classification algorithm by using accuracy, selection size ratio, specificity and sensitivity as the parameters. The analysis clearly shows that the accuracy, as well as the sensitivity and specificity of the HAPGD algorithm, are better than the other algorithm due to balanced exploration and exploitation search. The selection size ratio improvement is also shown as compared to ACO-PSO-GA algorithm. In the future, the exploration phase can be improved to enhance performance.

REFERENCES

1. A. Oussous, F. Z. Benjelloun, A. Ait Lahcen, and S. Belfkih, "Big Data technologies: A survey," *Journal of King Saud University - Computer and Information Sciences*, vol. 30, no. 4, pp. 431–448, 2018.
2. R. Iqbal, F. Doctor, B. More, S. Mahmud, and U. Yousuf, "Big Data analytics: Computational intelligence techniques and application areas," *International Journal of Information Management*, vol. In Press, 2018.
3. M. B. Dezfouli, M. H. Nadimi Shahraki, and H. Zamani, "A Novel Tour Planning Model using Big Data," in *2018 International Conference on Artificial Intelligence and Data Processing, IDAP 2018*, 2019, pp. 1–6.
4. Y. Wang, L. A. Kung, and T. A. Byrd, "Big data analytics: Understanding its capabilities and potential benefits for healthcare organizations," *Technological Forecasting and Social Change*, vol. 126, pp. 3–13, 2018.
5. Y. Wang, L. A. Kung, W. Y. C. Wang, and C. G. Cegielski, "An integrated big data analytics-enabled transformation model: Application to health care," *Information and Management*, vol. 55, pp. 64–79, 2018.
6. H. Wang *et al.*, "A hybrid multi-objective firefly algorithm for big data optimization," *Applied Soft Computing Journal*, vol. 69, pp. 806–815, 2018.
7. M. Dorigo and C. Blum, "Ant colony optimization theory: A survey," *Theoretical Computer Science*, vol. 344, no. 2–3, pp. 243–278, 2005.
8. T. Newbould, "Industrial coatings," *Engineer*, vol. 293, no. 7661, p. 44, 2004.
9. Y. Lu, M. Liang, Z. Ye, and L. Cao, "Improved particle swarm optimization algorithm and its application in text feature selection," *Applied Soft Computing Journal*, vol. 35, pp. 629–636, 2015.
10. J. H. Tam, Z. C. Ong, Z. Ismail, B. C. Ang, and S. Y. Khoo, "A new hybrid GA−ACO−PSO algorithm for solving various engineering design problems," *International Journal of Computer Mathematics*, vol. 0, no. 0, pp. 1–37, 2018.
11. D. Dheeru and E. Karra Taniskidou, "{UCI} Machine Learning Repository." 2017.
12. K. Menghour and L. Souici-Meslati, "Hybrid ACO-PSO based approaches for feature selection," *International Journal of Intelligent Engineering and Systems*, vol. 9, no. 3, pp. 65–79, 2016.
13. S. Nemati, M. E. Basiri, N. Ghasem-Aghaee, and M. H. Aghdam, "A novel ACO-GA hybrid algorithm for feature selection in protein function prediction," *Expert Systems with Applications*, vol. 36, no. 10, pp. 12086–12094, 2009.
14. S. M.Uma, K. Rajiv Gandhi, E. Kirubakaran, and D. E. K. Dr.E.Kirubakaran, "A Hybrid PSO with Dynamic Inertia Weight and GA Approach for Discovering Classification Rule in Data Mining," *International Journal of Computer Applications*, vol. 40, no. 17, pp. 32–37, 2012.
15. W. Y. Lin, "A GA-DE hybrid evolutionary algorithm for path synthesis of four-bar linkage," *Mechanism and Machine Theory*, vol. 45, no. 8, pp. 1096–1107, 2010.

AUTHORS PROFILE

Anju Bala is MCA, Research Scholar (JRF), Department of Computer Science and Applications, M.D.U, Rohtak, India. She has published more than 10 publications in various journals of national /international repute.Her area of research includes Big Data and Data mining.

Dr. Priti, MCA, PhD (Computer Science), Assistant Professor**,** Department of Computer Science and Applications, M.D.U, Rohtak, India. She has published more than 50 publications in various journals/magzines of national and international repute. She has 12 years of experience in teaching and research. Her area of research includes Software Re-engg, Data Mining, Big Data, and Machine Learning.