A Novel GSA-SVM Based Attribute Selection Algorithm for Movie Recommendation

Kunika Arora[1], Shekhar Singh[2], Dr. Vikram Bali[3]
[1](M.Tech Student, Department of CSE, Panipat Institute of Engineering & Technology, Samalkha, Haryana, India
Email: kunika.arra@gmail.com)
[2] (Assistant Professor, Department of CSE, Panipat Institute of Engineering & Technology, Samalkha, Haryana, India
Email: Shekhar.cse@piet.co.in)
[3] (Associate Professor, Department of CSE, Panipat Institute of Engineering & Technology, Samalkha, Haryana, India
Email: Vikram.cse@piet.co.in)

ABSTRACT

Until now, recommender systems have been extensively used within e-commerce and communities where items like movies, music and articles are recommended. More recently, recommender systems have been deployed in online music players, recommending music that the users probably will like. This thesis presents the design and implementation of recommendation system for movies. Our work is based on the fact that collaborative recommendation system is suitable for this application as it considers user's behavior as well as movie data. We used multiclass Support vector machine (SVM) as machine learning model and to choose more optimal attributes we chose Gravitational Search Algorithm (GSA) as it's a global Optimization algorithm and converge early with maximum accuracy as in our case. It gives an optimal features set out of 8 in total. We compared the results with PSO and Genetic Algorithm to prove GSA selects the best set of attributes and all selected features are trained and tested by SVM to give high rating recommendation of movies.

Keywords *– Collaborative Filtering (CF), Gravitational Search Algorithm (GSA), Support Vector Machine (SVM).*

I. INTRODUCTION

The term collaborative filtering means people are getting together to share their views about particular items and checking out each other's reviews on the basis of whatever reading material they are having at that time Paul Resnick and Hal R.Varian, in the year 1997 introduced the term Recommender System'. There were basically two prominent reasons behind finding this particular name as the first and foremost was recommenders could not work properly as the users were unaware of each other's identity, apart from that the other reason was their recommendation also included the interest of the user. Implicit feedback for recommendations was also being provided in combination with the aggregation techniques. To exemplify, collaborative filtering of internet news was done by a system called Group lens, it worked on two strategies first one was implicit feedback through reading time and explicit feedback through ratings. Furthermore, taking into consideration the privacy concerns of users a new strategy was also recommended and it was pseudonyms and to combine various recommendations weighted voting was used. Not only the efficiency but the cost was also given attention and it was found that maintenance and progress of these systems was hefty and it was important to identify whether the merits outweigh the demerits.

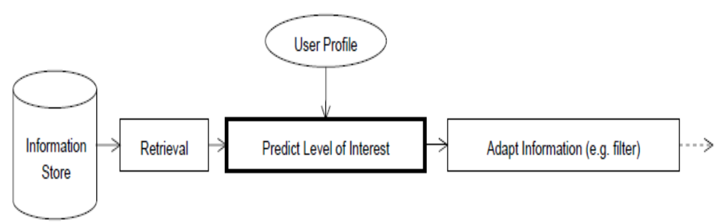


Fig. 1: Information Filtering in Recommender Systems

Recommendation systems [1] find numerous applications when it comes to purchasing or surfing information about any particular product .To substantiate my view, internet sites, shopping websites, electronics etc. are visited by people on daily basis foe meeting their needs. In addition to this, recommendation systems are conspicuously used in guiding people about variety of products that they enquire or they are interested in buying [2].

1.1 Classification of Recommender Systems

Recommendation systems are broadly classified into three categories. In Content-Based approach recommendations are made solely based on the attributes of the products which the user preferred previously. In Collaborative approach recommendations are made based on the similarity between the preferences of users and not the content of the products. The recommended items will depend on what other similar users liked. In Hybrid approach recommendations are generated based on the resemblances between users and the analogy between products along with their given feedback.

1.2 Objective

This work is done by machine learning algorithms which learns from user's past visiting behavior like type of movies browsed, movies search by which actor name, how old/new movies are watched etc and most importantly what rating is provided by users. Previously this work is done by SVM (Support Vector Machine) algorithm in which a model is trained based on history and movies are recommended. To get higher accuracy, the data attributes are optimally selected using Gravitational Search Algorithm which is an iterative optimization algorithm and good in terms of convergence speed and accuracy rate. So we focused to improve the accuracy as well as convergence speed for movie recommendation system. The objectives include firstly using the SVM classification to recommend the movies based on query input in such a manner to optimize the parameters of data attributes to achieve higher accuracy using optimization Gravitational Search Algorithm (GSA) and to use the overall accuracy as the objective function. Moreover, Movie lens recommendation dataset will be used for training and testing the SVM which is available on https://grouplens.org/datasets/movielens/ to use freely.

II. LITERATURE REVIEW

V. Adi Lakshmi et al. [3] demonstrated a hybrid approach of implementing recommendation systems using content based methods as well as collaboration methods. They did experimentations on a movie data to show optimum levels of accuracy. Moreover, they increased the accuracy level in comparison to matrix factorization approach. Owing to the fact that required results are not obtained when merely collaborative methods are used as they depend on user’s ratings, they presented the use of content information. Additionally, it was proposed that demographic information of the users along with the genre information of the users can further help in inclining the accuracy levels.

Karzan Wakil et al. [4] presented a new factor termed as human emotions while using recommendation systems along with Content Based Filtering (CBF), Collaborative Filtering (CF), emotions detection algorithm and their own algorithm, which is represented by matrix. In addition to this, the advantage lies in the fact that there is a relation between people’s emotional states and the recommended movies. They recommended the researchers to enhance the idea by following some strategies first of all, using not only one recommendation technique to attain better result of the movies. Secondly, using colours which can be more than three to find human emotions. Lastly, designing a new algorithm to solve the movie recommender system.

Rahul Katarya et al. [5] discussed movie recommendation system used with Movielens dataset using k-means clustering with cuckoo search optimization algorithm .In a systematic manner the approach has been explained and outputs were shown. Comparison of already prevailing approaches is done with it and consequently analyzation of results is performed. It is also compared with existing approaches, and the results have been analyzed and interpreted. Evaluation metrics such as mean absolute error (MAE), standard deviation (SD), root mean square error (RMSE) and t-value for the movie recommender system delivers better results as our approach offers lesser value of the mean absolute error, standard deviation, and root mean square error. The results obtained on Movielens stipulate that recommended approach may provide high performance regarding reliability, efficiency and delivers accurate personalized movie recommendations in comparison to other methods.

Kathpal Mohit et al. [6] worked on K-Means based crowd-aware recommender system which considered the nearest crowd of a specified user then further located preferences of those which can be suggested to the user at that point of time. In a particular dataset, to make a cluster of spots among the set of spots a K-means clustering algorithm is used. Depending on the belonging of a particular user in a cluster groups are made and they are placed in that cluster. In addition to this, user is classified in a particular crowd following that set of movies is searched by the system which is mostly liked by the user group. From this particular idea there are chances of attaining high precision, effectiveness and less time consuming.

Peng Yi et al. [7] discussed the cold start problem which arises due to data irregularity. In search of similar set movie labels are used for improving the user recommended rate of the new movie in cold start problem. The common features of actors and directors were considered to optimise the similar movie searching. In the testing with movie lens dataset, this particular method of optimization outweighs the original algorithm.

Jesse Dodge et al. [8] presented a new set of benchmark tasks designed to evaluate end-to-end dialog systems. The movie dialog dataset measures how well such models can perform at both goal driven dialog, of both objective and subjective goals thanks to evaluation metrics on question answering and recommendation tasks, and at less goal driven chit-chat. A true end-to-end model should perform well at all these tasks, being a necessary but not sufficient condition for a fully functional dialog agent.

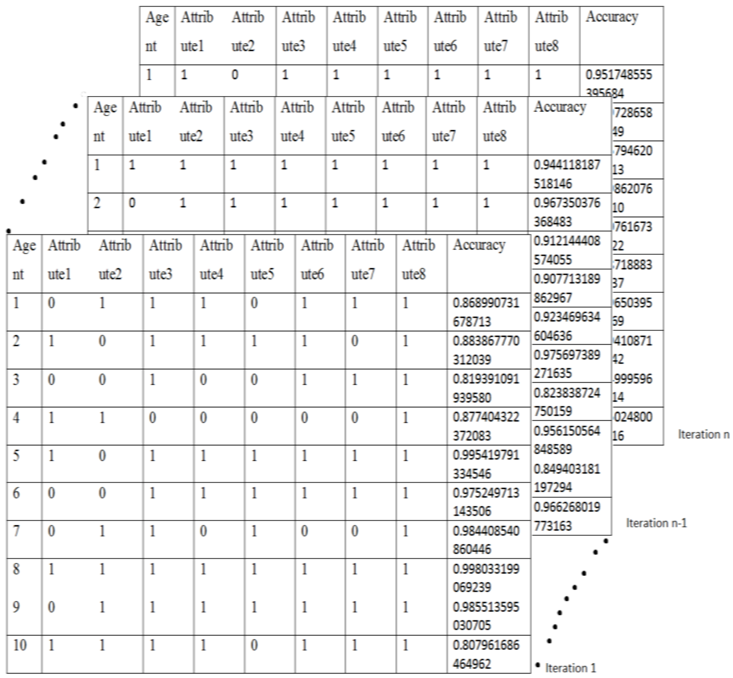
III. PROPOSED WORK

The Collaborative filtering (CF) is widely used for music recommendation, movie recommendation, news recommendation etc. In this work we used it for movie recommendation engine. It is purely based on movies rating whether explicit or implicit by the user. A set of user's behavior data is collected and any machine learning model is trained to learn user's behavior. Using that trained model, movies are recommended to any user. In our case Support Vector Machine (SVM) which has advantage of an overall optimum and a strong generalization ability over other ML algorithms. The setback point here is the data attributes to create a hurdle in ML modeling to recommend effectively as not all attributes contribute into recommendation every time. Some of them can reduce the accuracy too. So we must choose the attributes which contribute the most and also reduction of dataset will train the model faster. For this purpose we used optimization to choose the optimum feature set of input attributes. Previously this is done by using particle swarm optimization (PSO) but this optimization method has the issue of premature convergence since it is a local search heuristic algorithm. The optimum solution searched by it is not necessarily best due to skipping of some minima/maxima points. So we replaced this optimization with the Gravitational Search Algorithm. The data used here is movie lens dataset which has a total of 8 attributes to be optimally selected. Dataset description is detailed in next section of this chapter. Out of these attributes, GSA will choose all those only which contribute more in the accuracy improvement in recommendation. The number of agents in GSA acts as the number of options available for attributes selection at an iteration and their position is the column choice in the data. The position of agents will be either 0 or 1. '1' stands for selection of that attribute and '0' stands for that attribute is not selected in that iteration. For iteration, 10 different set of attributes choice is chosen which means 10 agents are used for GSA to optimise attributes choice. The whole data is first divided into testing and training datasets. 80% of data is used for training and rest is used for testing. For the first iteration, each position of agent is chosen randomly which is for number of attributes set for an iteration. The selected attributes for them are used to train the multiclass SVM and tested for the query data. This accuracy is noted down in a matrix for each GSA agent. An example matrix is shown in Table 1.

Table 1
Accuracy Matrix for GSA Agents

Agent	Attribute1	Attribute2	Attribute3	Attribute4	Attribute5	Attribute6	Attribute7	Attribute8	Accuracy
1	0	1	1	1	0	1	1	1	0.868990731678713
2	1	0	1	1	1	1	0	1	0.883867770312039
3	0	0	1	0	0	1	1	1	0.819391091939580
4	1	1	0	0	0	0	0	1	0.877404322372083
5	1	0	1	1	1	1	1	1	0.995419791334546
6	0	0	1	1	1	1	1	1	0.975249713143506
7	0	1	1	0	1	0	0	1	0.984408540860446
8	1	1	1	1	1	1	1	1	0.998033199069239
9	0	1	1	1	1	1	1	1	0.985513595030705
10	1	1	1	1	0	1	1	1	0.807961686464962

Out of these accuracy values, the best one is saved. Now the positions of agents are updated or in other words the attributes selection is changed and again accuracy for the new selection is calculated and saved for the best value. This process keeps going on and a n-dimensional matrix is formed out of which corresponding row to highest accuracy is considered to be the best selection for attributes as in following figure 2.



Agent	Attribute1	Attribute2	Attribute3	Attribute4	Attribute5	Attribute6	Attribute7	Attribute8	Accuracy
1	0	1	1	1	0	1	1	1	0.868990731678713
2	1	0	1	1	1	1	0	1	0.883867770312039
3	0	0	1	0	0	1	1	1	0.819391091939580
4	1	1	0	0	0	0	0	1	0.877404322372083
5	1	0	1	1	1	1	1	1	0.995419791334546
6	0	0	1	1	1	1	1	1	0.975249713143506
7	0	1	1	0	1	0	0	1	0.984408540860446
8	1	1	1	1	1	1	1	1	0.998033199069239
9	0	1	1	1	1	1	1	1	0.985513595030705
10	1	1	1	1	0	1	1	1	0.807961686464962

Fig. 2: Matrix created for accuracy and attributes selection for each agent for n number of iterations.

A complete step by step algorithm is explained below:

- Load the movie lens dataset in numeric format and divide that into random 80:20 ratio for training and testing of recommendation engine.
- Initialise the GSA parameters like number of iterations, number of agents, initial G0 and alpha.
- Randomly initialise the agent's new positions which must be either 1 or 0 and will choose the attributes out of 8 in total.
- Call the objective function to train the model for selected attributes in training data and test the model for testing data to get the recommendation accuracy.
- To update the random positions of agents, force and mass has to be calculated by using the equations

$$F_{ij}^{d}(t) = G(t)\frac{M_{pi}(t) \times M_{aj}(t)}{R_{ij}(t) + \varepsilon}\left(X_j^d(t) - X_i^d(t)\right), \quad (1)$$

$$m_{it} = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \quad (2)$$

- The new updated position is obtained from the formula

$$X_i^d(t+1) = X_i^d(t) + V_i^d(t+1) \quad (3)$$

The velocity in this case is calculated by using acceleration which is based on force and mass calculated in previous step.

- For this new updated position or values of weights and biases, objective function is again called and accuracy is saved.
- The attributes' positions for which minimum of accuracy is obtained out of previous two set of values, is further considered for updating.
- This process continues till all iterations are not completed.
- The final maximum accuracy is obtained and attributes selected for them are used as final set of attributes which gives higher accuracy.

Following figure 3 shows the flow chart of the proposed algorithm.

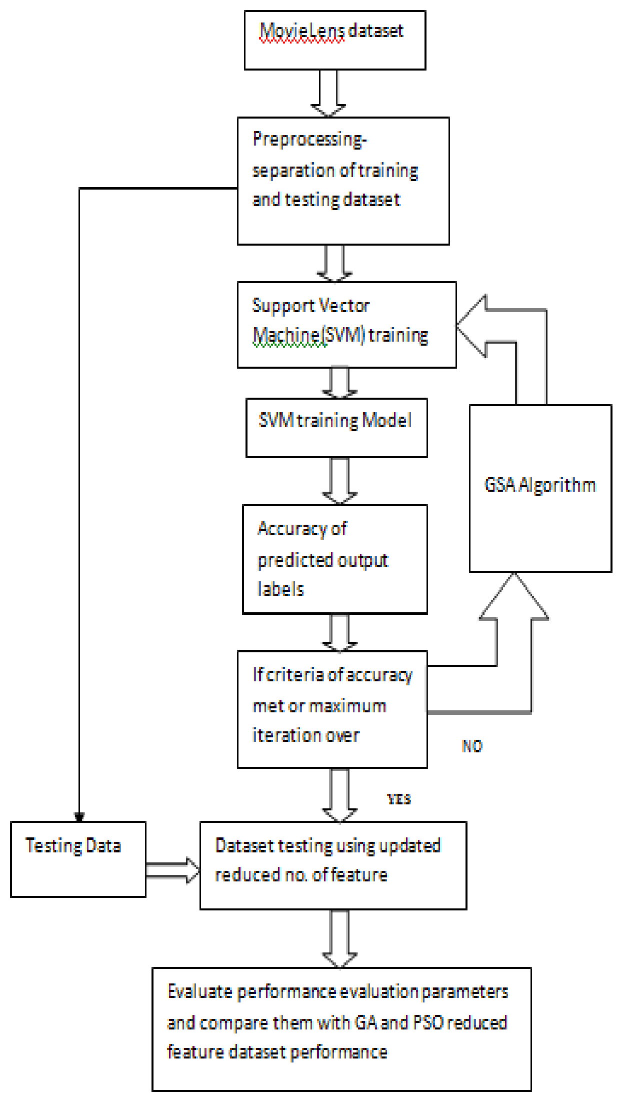


Fig. 3: Flowchart of Proposed Model

IV. RESULT

We used multiclass SVM to recommend the movies similar to user selected movie. The multiclass here because the rating of movie is the criteria on which SVM modeling will recommend the movie and this rating is in between 1-5.So a total of 5 classes are there to be identified and user will get those similar movies whose rating is highest among that genre. We further optimized the attributes for the accuracy improvement using GSA optimization and compared the results with GA and PSO.

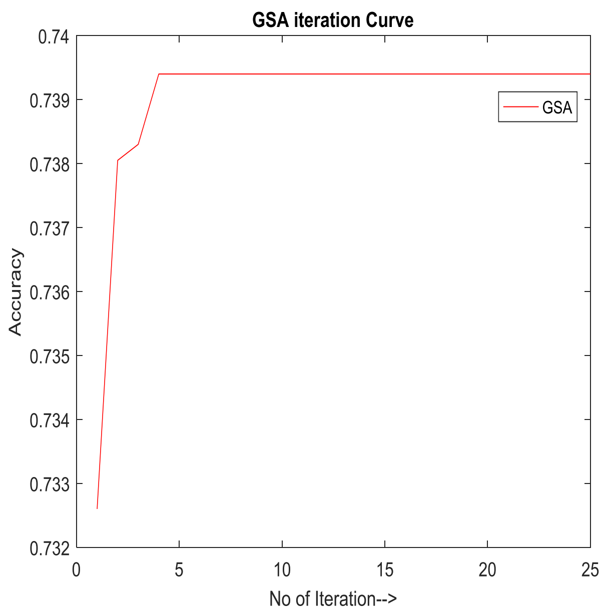


Fig. 4: GSA Iteration Curve

Figure 4 shows the comparative curve which are plotted for highest accuracy out of all agents for all iterations. It is clear from the figure that GSA is settling at maximum value at 4th iteration and not changing after that. This proves we can expect a good optimal selection of attributes from this optimization.

On the basis of five evaluation parameters we compared the results of all three optimization algorithms for data division of 80/20 % ratio. Table 2 shows the corresponding values for each evaluation parameter or GSA, GA and PSO.

Table 2
Evaluation of Results of Comparison of all three Optimization Algorithms

	Accuracy	Sensitivity	Specificity	Precision	Recall
GSA	0.7095	0.9247	0.6955	0.0979	0.7397
PSO	0.3457	0	0.3678	0	0
GA	0.3374	0	0.3590	0	0

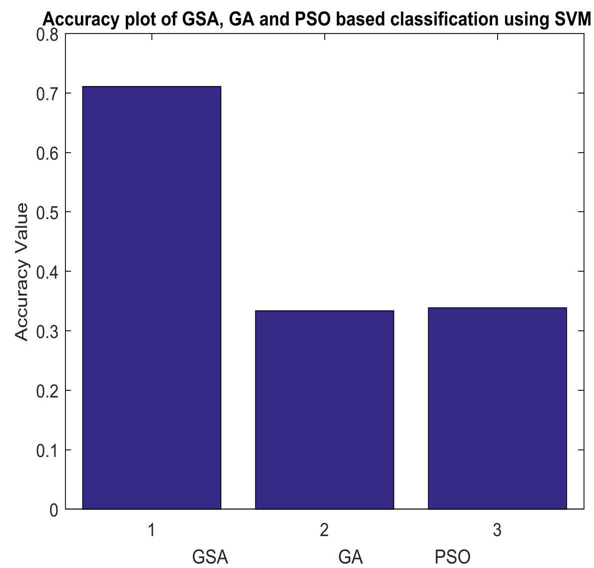


Fig. 5: Accuracy comparison plot for all three optimization algorithms using multiclass SVM

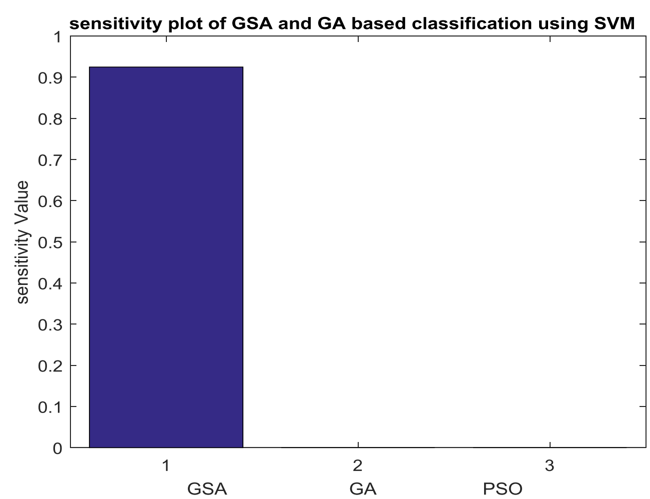


Fig. 6: Sensitivity comparison plot for all three optimization algorithms using multiclass SVM

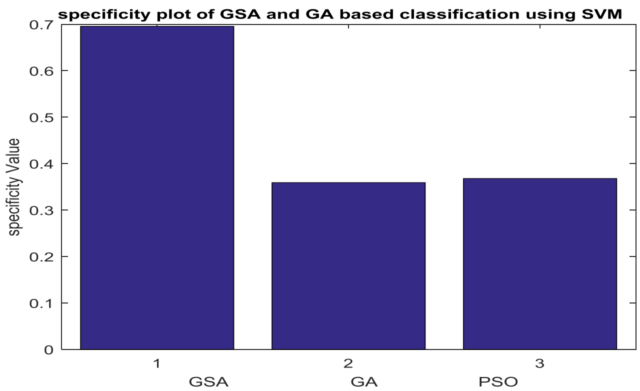


Fig. 7: Specificity comparison plot for all three optimization algorithms using multiclass SVM

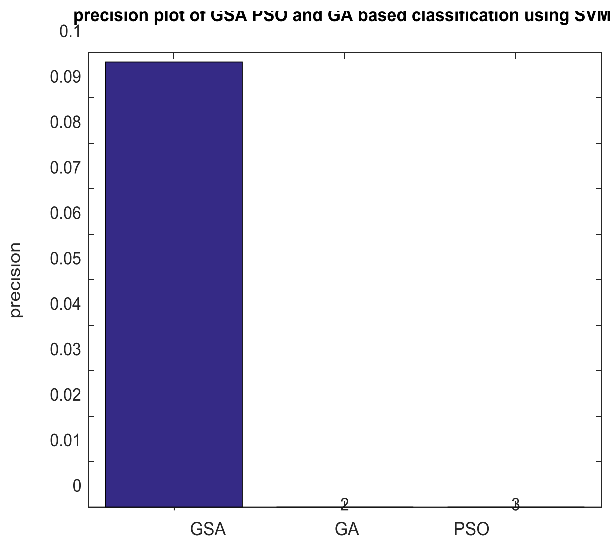


Fig. 8: Precision comparison plot for all three optimisation algorithms using multiclass SVM

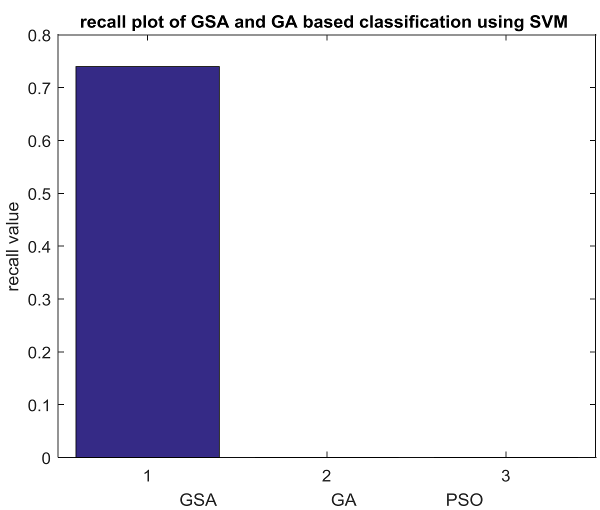


Fig. 9: Recall comparison plot for all three optimisation algorithms using multiclass SVM

Above plots are done for single class out of five. So we used confusion matrix plots to analyze the accuracy performance by each class. Confusion matrix will tell us how many samples were picked for each class and exact number of samples detected in each class after model testing. Following figures shows the confusion matrix plot for these three optimizations. The yellow highlighted box in confusion map represents the accuracy % for that class. As is clear from the figure 10, for class 1, actual number of samples collected were 1435, out of which 904 were selected correct and 531 were classified as class 2 which is wrong detection, while in figure 11 and figure 12 the true detection is zero for class1.

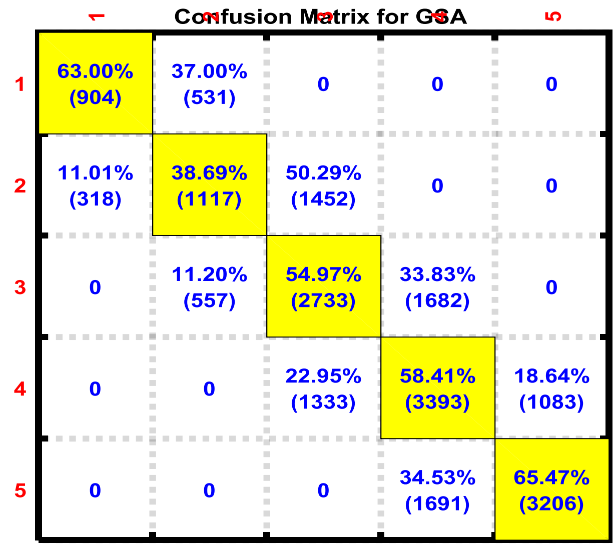


Fig. 10: Confusion matrix plot for GSA selected attributes

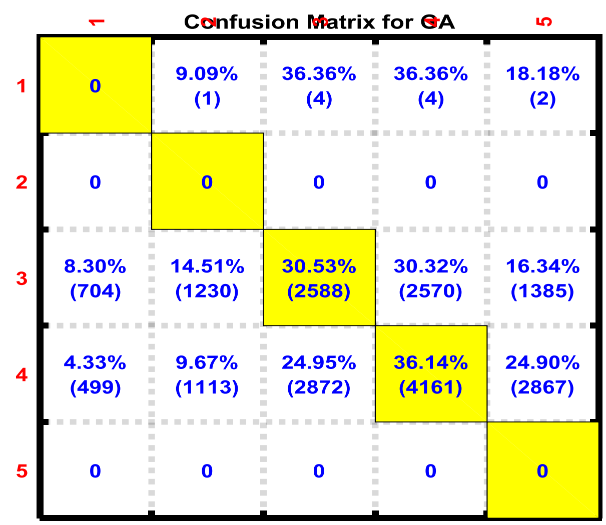


Fig. 11: Confusion matrix plot for GA selected attributes

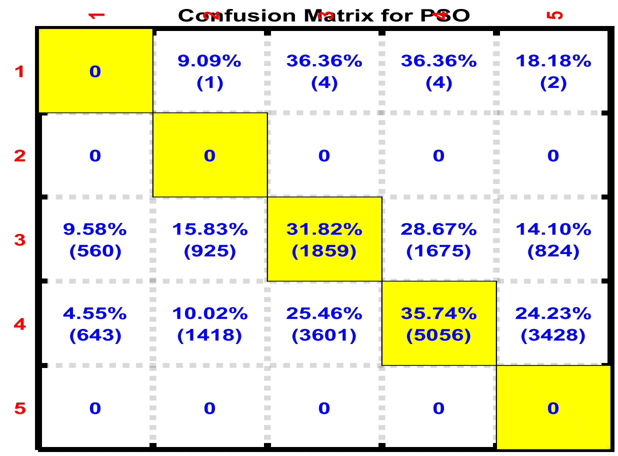


Fig. 12: Confusion matrix plot for PSO selected attributes

V. CONCLUSION AND FUTURE SCOPE

Majority of movie recommendation systems are based on the concept of training and testing the ML model for a given dataset. They forget to analyze that whether given attributes of whole data are actually taking part in recommendation or they are decreasing the recommendation accuracy. To avoid this situation we worked on optimally choosing the attributes for better accuracy. Gravitational search optimization is selected for this work which is better than PSO optimization as it converge at higher accuracy value than PSO. Multiclass SVM modeling is used due to movies rating form 1-5. Results are compared with PSO and GA in terms of precision, recall, specificity, accuracy and sensitivity. In all evaluation parameters GSA results are winning the competition.

The problem occurs when new data or user has to be introduced in already trained model. A combination of content-based and collaborative filtering would reduce the issue. A safe user feedback should be present. Like if a user wants to watch romantic movie then he must be introduced some goof romantic movie. Scalability can be made available as it helps in transforming the system in such a way that it is suitable for large number of users worldwide. More attributes for user information and movie information must be added which contribute to accuracy improvement like user's previous interest, watch time of a recommended movie etc.

REFERENCES

[1] Paul Resnick and Hal R. Varian. *Recommender systems*, volume 40. ACM Press

[2] J. B. Schafer, J. A. Konstan, and J. Riedl. *E-Commerce Recommendation Applications*.

[3] V.Adi Lakshmi, J.Kanaka Priya, “A Hybrid Approach to Movie Recommender System”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 9, September 2013.

[4] K.Wakil, K. Ali, R. Bakhtyar, “Improving Web Movie Recommender System Based on Emotions”, International Journal of Advanced Computer Science and Applications, Volume 6, Issue 2, 2015.

[5] R. Katarya, O. Prakash Verma," An effective collaborative movie recommender system with cuckoo search", Egyptian Informatics Journal 18 , Pages 105–112,2017.

[6] K. Mohit, B. Rajdavinder Singh and S. Gurpreet," K-Means Demographic based Crowd Aware Movie Recommendation System", Indian Journal of Science and Technology, Volume 9, Edition 23, June 2016.

[7] P. Yi, C. Yang, X. Zhou and C. Li, "A movie cold-start recommendation method optimized similarity measure," 2016 16th International Symposium on Communications and Information Technologies (ISCIT), Qingdao, 2016.

[8] J. Dodge, A. Gane , X. Zhang , A. Bordes, S. Chopra, A. H. Miller, A. Szlam & J. Weston," Evaluating Prerequisite Qualities For Learn - Ing End -To-End Dialog Systems", arXiv:1511.06931v6, April 2016.