IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 7, July 2023

DOI: 10.17148/IJARCCE.2023.12748

A Feature Selection Method using FSPSO

J. Vimal Rosy[1]* and Dr. S. Britto Ramesh Kumar[2]

Head & Assistant Professor, Soka Ikeda College of Arts & Science for Women, Chennai,

St. Joseph's College (Autonomous), Trichy[1]

Assistant Professor, Soka Ikeda College of Arts & Science for Women, Chennai,

St. Joseph's College (Autonomous), Trichy[2]

Abstract: A key large data activity, feature selection (FS), reduces the "curse of dimensionality" by choosing a meaningful feature subset to improve classification performance. Search algorithms may be constrained by FS techniques as the number of attributes grows. To achieve equivalent or better classification performance and increase computing efficiency, a subset of pertinent characteristics are chosen from a large number of original features using the feature selection process. Particle swarm optimization (PSO) is a global search metaheuristic that can swiftly and with few presumptions search a space with many dimensions. An FSPSO method known as particle swarm optimization (PSO) has recently attracted a lot of attention from experts in the field. Following a basic explanation of feature selection and PSO, a review of recent PSO for feature selection work is given.

Keywords: Particle Swarm Optimization, Feature Selection, Classification, Mutation Operator.

I. INTRODUCTION

A dataset for classification problems typically contains a lot of features, frequently containing useful, unnecessary, and redundant features. Due to the huge search space—known as "the curse of dimensionality"—irrelevant and redundant features are not helpful for classification and may significantly worsen classification performance [1,2]. In order to obtain comparable or even higher classification performance than using all features, feature selection is offered as a method for choosing a subset of pertinent characteristics from a vast number of accessible features [2]. Feature selection can minimize the number of features, shorten training time, streamline taught classifiers, and/or enhance classification performance by removing/reducing redundant and irrelevant information [2].

The tough combinatorial problem of feature selection. Because of the issue with feature interaction, the best feature subset is typically a collection of characteristics that exhibit feature complementarity. Features may interact in a two-way or multiple-way fashion [1,5]. As a result, removing some of these features will remove or decrease unneeded complexity because they may become redundant when used in conjunction with other features. On the other hand, a feature that is redundant or just marginally useful when used alone may become extremely relevant when used alongside others. Consequently, a group of complementary traits should make up an ideal feature subset.

The work of feature selection is difficult primarily because of the size of the search space. With respect to the quantity of features present in the dataset, the size of the search space grows exponentially [1]. Therefore, in most cases, a thorough search is almost impossible. The majority of these algorithms still experience issues with local optima stagnation or being computationally expensive, despite the fact that numerous alternative search approaches have been used for feature selection [4]. An effective global search method is required to more effectively solve feature selection issues.

Global search capabilities of evolutionary computation (EC) approaches are well recognized. A more modern EC technique called particle swarm optimization (PSO) [6,7] is less computationally intensive than certain previous EC techniques. PSO has thus been employed as a successful feature selection technique [8,4]. The current PSO does have some restrictions regarding feature choices, though. The feature selection task has not been adjusted into PSO, to start. According to Gutierrez et al. [9], initialization procedures in PSO perform differently in various high-dimensional search space situations.

With the exception of our earlier work [10], no known initialization solutions are, however, expressly given for feature selection difficulties. Furthermore, the conventional personal and global best updating mechanism may overlook some feature subsets with excellent classification performance but few characteristics . As a result, the potential of PSO for feature selection hasn't been fully explored, and we'll keep working on our earlier research [10] to learn more about the initialization and updating processes in PSO for feature selection.

II. RELATED WORKS

A dynamic PSO model with escaping prey schemes (DPSOEP) was put forth by Chen et al. [1]. According to their fitness scores, swarm particles in DPSOEP were divided into three sub-swarms: "preys" (top-ranked particles), "strong particles"

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 7, July 2023
DOI: 10.17148/IJARCCE.2023.12748

(middle-ranked particles), and "weak particles" (lower-ranked particles). Following distinct search operations, such as Lévy flights, the initial PSO location, and a multivariate normal distribution, the particles in the aforementioned groups then searched for global optimality.

Vimalrosy et al[13]. optimized the Sine Swarm to find the best features for the UNSW-NB15 dataset. The Random forest classification method is used to make the correct classification. 98.15% accuracy was reached with the suggested system, OSS-RF.

A modified PSO approach for choosing the best hyper-parameters for Gaussian process regression (GPR) was put out by Kang et al. in their proposal [2]. A momentum element was suggested in place of the inertial component as in PSO and was dependent on the mean distance of the swarm in two subsequent cycles. The global best solution was then further improved via a mutation method built on a perturbation function.

A multiswarm PSO technique is presented by Liu et al. [3] to simultaneously search for the ideal feature subset and optimize the SVM's parameters. Experiments demonstrate that the suggested feature selection method might outperform grid search, traditional PSO, and GA in terms of classification accuracy. Due to the enormous population size and complex communication rules between many subswarms, the proposed technique is computationally more expensive than the other three methods.

According to Chuang et al. [12] technique for PSO feature selection, gbest will be reset to zero if its value remains constant for a number of rounds. However, neither PSO or EC based algorithms are employed for comparisons; instead, the suggested approach is merely put up against one standard method in terms of classification performance.

A filter feature selection approach is put forth by Wang et al. [11] and is based on an enhanced binary PSO and rough set theory. The degree of reliance between class labels and chosen features, as determined by a rough set, is used to award a particle's quality. This study also demonstrates another issue of applying rough set theory to feature selection problems: computing the rough set takes up the majority of the running time. The reliance between features and class labels can also be demonstrated using fuzzy sets.

A filter feature selection approach is put forth by Wang et al. [11] and is based on an enhanced binary PSO and rough set theory. The degree of reliance between class labels and chosen features, as determined by a rough set, is used to award a particle's quality. This study also demonstrates another issue of applying rough set theory to feature selection problems: computing the rough set takes up the majority of the running time. The reliance between features and class labels can also be demonstrated using fuzzy sets.

III. DATASET DESCRIPTIONS

According to (Slay N. M., 2016), UNSW-NB15 was developed in the Australian Centre for Cybersecurity's cyber range lab in 2015. One of the dataset's formats is CSV files. The original CSV files, which were divided into four files and had more than 2.5 million records, are not used.

Since the training and testing sets of the polished CSV files contain 175,341 transactions and 82,332 entries, respectively, we are employing them in our study. The dataset has 47 features, encompassing category, nominal, and numeric data types. It is a multi-class, binary labelled dataset. Table 1 displays the distribution of each assault in training and test sets.

Table 1. Number of records in training and testing subsets for each class

Classes	Training Subset	Testing Subset
Normal	56,000	37,000
Analysis	2,000	677
Backdoor	1,746	583
DoS	12,264	4,089
Exploits	33,393	11,132
Fuzzers	18,184	6,062
Generic	40,000	18,871
Reconnaissance	10,491	3,496
Shellcode	1,133	378
Worms	130	44
Total Number of Records	175,341	82,332

IV. FEATURE SELECTION

The goal of feature selection is to choose the smallest possible subset of features that are both required and adequate to represent the target notion [12]. Its goals include decreasing the amount of data required for learning, reducing running time, enhancing system accuracy, and improving the understandability of the learnt model . The five fundamental steps of a common feature selection approach [11] are depicted in Fig. 1:

ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 8.102 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 12, Issue 7, July 2023

DOI: 10.17148/IJARCCE.2023.12748

1.Initialization: The initialization process for a feature selection method is based on all of the original features.
2. Subset discovery : It is a process for finding potential subsets. It is a search process that can begin with none, all, or a randomly chosen subset of features. In this generation step, the best subset of features is sought using a variety of search approaches, including both traditional search methods and FCPSO techniques.
3. Subset evaluation: A function to gauge how well the created feature subsets performed.
4. Stopping criteria: The algorithm will terminate depending on a predetermined criterion, which may be based on the evaluation function or the creation process. The former can be a predetermined maximum number of iterations completed or a predetermined number of features chosen. The latter entails determining if a particular evaluation function yields an optimal feature subset and whether the addition or deletion of any feature results in a worse subset.
5. Results validation: Tests on unseen data are conducted to determine the correctness of the chosen subset.

Algorithm: MFPSO

Begin
2 divide Dataset into a Training set and a Test set; initialize the swarm;
3 initialize the set of head XSet and Archive
4 calculate the crowding distance of each member in XSet;
5 while Maximum Iterations is not reached do
6 for each particle do
7 select a head (gbest) from XSet for each particle by using a binary tournament selection based on the crowding distance;
8 update the velocity and position of particle i
9 apply mutation operators;
10 evaluate two objective values for each particle; /∗ number of features and the classification error rate on the Training set ∗/
11 update the pbest of each particle;
12 end 13 identify the nondominated solutions (particles) to update XSet;
14 send head to Archive;
15 calculate the crowding distance of each member in XSet;
16 end 17 calculate the classification error rate of the solutions in Archive on the test set;
18 return the solutions in Archive and their training and test classification error rates;
19 end

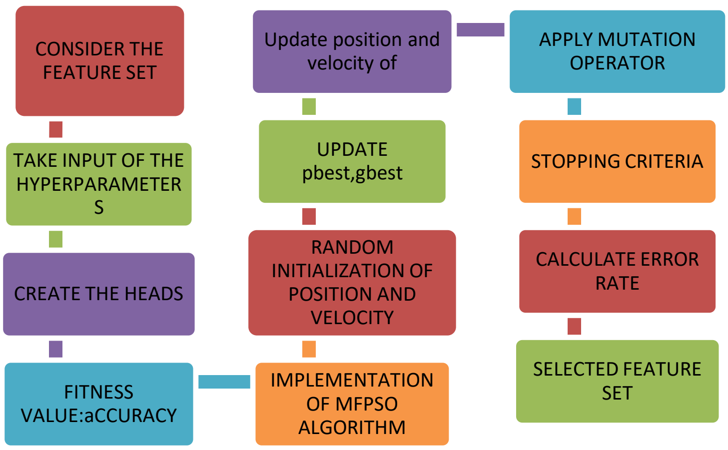

Fig 1 : Flow of FSPSO Method

Algorithm displays the FSPSO pseudocode. The primary problem of finding a good leader (gbest) is addressed by FSPSO, which uses a leader set to store the nondominated solutions as prospective leaders for each particle. A gbest is chosen from the leader set using a binary tournament selection and their crowding distances. To choose which nondominated solutions should be included in the X set and retained throughout the evolutionary process, a crowding factor is specifically used. The less congested of the two solutions picked from the head established by the binary tournament selection is determined to be the best option. The number of particles in the swarm is typically used to determine the maximum size of the X set. Mutation operators are used to maintain the swarm's variety and to enhance the algorithm's search capabilities.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁘ Impact Factor 8.102 ⁘ Peer-reviewed / Refereed journal ⁘ Vol. 12, Issue 7, July 2023

DOI: 10.17148/IJARCCE.2023.12748

V. EXPERIMENTAL RESULTS

The algorithms often increased accuracy by roughly 2–3%, which was a promising outcome. Even though none of the three classification methods performed optimally for a given dataset, they all produced promising results. FSPSO typically outperformed PSO (with a constant threshold of 0.5) in these situations.

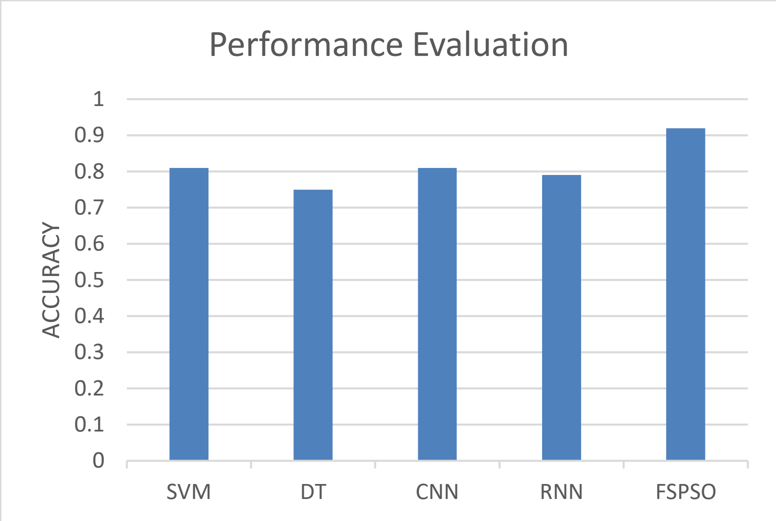

Fig 1.2 Performance Evaluation

CONCLUSION

Wrapper-based approaches take a lot of work, but with the right optimization methodology, they can yield an ideal feature set and a classification model that performs satisfactorily. In this study, we presented FSPSO with a threshold value depending on the mean of feature-class and compared the outcomes with a straightforward classifier and PSO-based feature selection with a constant threshold of 0.5. The most effective model, FSPSO, performed well across all datasets. The technique may now be expanded to work on redundant data and is most effective for deleting irrelevant data. The algorithm's time complexity for convergence and producing a good result is higher with high-dimensional datasets. By choosing a strong fitness value, the model can be improved even more. To confirm its validity, this method can be further applied to a larger pool of datasets.

REFERENCES

[1]. Chen, K.; Zhou, F.-Y.; Yuan, X.-F. Hybrid particle swarm optimization with spiral-shaped mechanism for feature selection. Expert Syst. Appl. 2019, 128, 140–156.

[2]. Kang, L.; Chen, R.-S.; Xiong, N.; Chen, Y.-C.; Hu, Y.-X.; Chen, C.-M. Selecting Hyper-Parameters of Gaussian Process Regression Based on Non-Inertial Particle Swarm Optimization in Internet of Things. IEEE Access 2019, 7, 59504–59513.

[3] Y. Liu, G. Wang, H. Chen, and H. Dong, “An improved particle swarm optimization for feature selection,” J. Bionic Eng., vol. 8, no. 2, pp. 191– 200, Jun. 2011.

[4] K. Neshatian, M. Zhang, Pareto front feature selection: using genetic programming to explore feature space, in: Proceedings of the 11thAnnual conference on Genetic and evolutionary computation (GECCO’09), New York, NY, USA, 2009, pp. 1027–1034. [18]

[5] D. Muni, N. Pal, J. Das, Genetic programming for simultaneous feature selection and classifier design, IEEE Transactions on Systems, Man, and Cybernetics. Part B: Cybernetics 36 (2006) 106–117. [19]

[6] K. Neshatian, M. Zhang, P. Andreae, A filter approach to multiple feature construction for symbolic learning classifiers using genetic programming, IEEE Transactions on Evolutionary Computation 16 (2012) 645–661.

[7] Zhang, L.; Mistry, K.; Lim, C.P.; Neoh, S.C. Feature selection using firefly optimization for classification and regression models. Decis. Support Syst. 2018, 106, 64–85.

[8] Xue, B.; Zhang, M.; Browne, W.N.; Yao, X. A Survey on Evolutionary Computation Approaches to Feature Selection. IEEE Trans. Evol. Comput. 2016, 20, 606–626.

[9] L. Cervante, B. Xue, M. Zhang, and L. Shang, “Binary particle swarm optimisation for feature selection: A filter based approach,” in Proc. IEEE CEC, 2012, pp. 1–8. [46]

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified ⁘ Impact Factor 8.102 ⁘ Peer-reviewed / Refereed journal ⁘ Vol. 12, Issue 7, July 2023
DOI: 10.17148/IJARCCE.2023.12748

[10] B. Xue, M. Zhang, and W. N. Browne, "New fitness functions in binary particle swarm optimisation for feature selection," in Proc. IEEE CEC, 2012, pp.

[11] X. Wang, J. Yang, X. Teng, W. Xia, and R. Jensen, "Feature selection based on rough sets and particle swarm optimization," Pattern Recognit. Lett., vol. 28, no. 4, pp. 459–471, Mar. 2007.

[12] L. Y. Chuang, H. W. Chang, C. J. Tu and C. H. Yang, "Improved binary PSO for feature selection using gene expression data", Computational Biology and Chemistry, 32(1), 29-38, 2008.

[13] Vimal Rosy and Dr. S. Britto Ramesh Kumar, "OSS- RF: Intrusion Detection using optimized Sine swarm based random forest classifier on UNSW-Nb 15 dataset", International Journal of Technical & physical problems of Engineering (IJTPE) Issue : 51, Vol.14, No. 2, PP, 275-283, ISSN:207723528 June 2022.

[14]. J. Vimal Rosy and Dr. S. Britto Ramesh Kumar, "SC-CVAR : Intrusion detection using Feature selection and Machine Learning Techniques on UNSW-NB15 dataset", International Journal of Computer Science and Network Security (IJCSNS), ISSN: 1738-7906 , Vol 22, No.4 April 2022.

[15] I. A. Gheyas and L. S. Smith, "Feature subset selection in large dimensionality domains," Pattern Recognit., vol. 43, no. 1, pp. 5–13, Jan. 2010.

[16] M. Dash and H. Liu, "Feature selection for classification," Intell. Data Anal., vol. 1, no. 1–4, pp. 131–156, 1997.

[17] A. Unler and A. Murat, "A discrete particle swarm optimization method for feature selection in binary classification problems," Eur. J. Oper. Res., vol. 206, no. 3, pp. 528–539, Nov. 2010.

[18] N Moustafa, J Slay , "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set) , - 2015 military communications and information system.

[19] B. Xue, M. Zhang, and W. N. Browne, "New fitness functions in binary particle swarm optimisation for feature selection," in Proc. IEEE CEC, 2012, pp. 1–8.

[20] K. Neshatian and M. Zhang, "Using genetic programming for contextsensitive feature scoring in classification problems," Connect. Sci., vol. 23, no. 3, pp. 183–207, 2011.

.