Intelligent Hybrid Swarm based Feature Selection Methods using Rough Set

Tarun Maini, Rakesh Kumar Misra, Devender Singh

Abstract: New feature selection methods based on Rough Set and hybrid optimization technique are proposed in this paper. In this work Feature Selection (Feature Reduction) has been implemented using Rough Set. Lower approximation based Rough Set has been used to calculate Positive Region which is consequently used to calculate Rough Dependency measure. Weighted sum of rough dependency measure and difference of total features of dataset and reduct normalized with respect to total feature, is used as fitness function. To optimize (maximize) this fitness function, a hybrid method of swarm intelligence algorithms like Intelligent Dynamic Swarm (IDS) and Particle Swarm Optimization (PSO) has been proposed and new method of population initialization has also been proposed. This method has been implemented on UCI repository based benchmark datasets of and it is shown that it results in improved reducts in terms of number of features, execution time with acceptable classification accuracy.

Index Terms: Feature Selection, Rough Set, Swarm Intelligence, Intelligent Dynamic Swarm, Particle Swarm Optimization

I. INTRODUCTION

Feature Selection involves removing superfluous, irrelevant and noisy features resulting in the reduced subset of features mapping the class labels even more accurately [1], preserving the predictive capability. Many researchers have been exploring this area to find appropriate and effective techniques to perform this task [2-6].

If we have \(K\) number of features in a given dataset and we are required to perform feature selection, we may have several methods to obtain best subset of features, out of given set. Few of them are as follows: First of all exhaustive method can be tried, in which we have to enumerate and validate (based on some predefined criteria) all possible subset solutions, for selecting best subset solution. But for medium and large datasets this method is practically impossible. So we need the alternative methods having reasonable execution time.

The alternative method may be Hill Climbing approach. Which may be used with a suitable fitness function/predefined criteria. Two important techniques like forward selection and backward elimination are the well-known strategies using this approach. Hill climbing methods for selecting the useful features from a given datasets are suggested in [4] and [7]. These methods may get stuck in local minima.

Instead of Hill climbing, for improved search, meta-heuristic search methods can be used. In this method we can use IDS, PSO and GA etc. These methods may provide near optimal solutions, with less computation. Also these methods facilitate simultaneous feature selection, which is different from hill climbing.

Rough Set [8] is an efficient technique for feature selection [9-10]. For picking all the appropriate features at one go, Wang et al. have used rough dependency measure as fitness function [5, 11].

Rough Sets are useful for performing feature selection tasks, provided a suitable method is devised to select useful features. If we optimize fitness value based on some metric of dataset, it may provide near optimal reduct.

Ref. [5] reported a Rough Set Theory (RST) based fitness function approach in their work. The authors used PSO for optimizing the fitness function. Ref. [6] introduced Rough Set - Intelligent Dynamic Swarm (RST-IDS) claiming to be 30% faster than RST-PSO.

In the present work, an attempt is made to further improve the method reported in [5] through implementations of Hybrid methods of IDS and PSO along with Distributed Sampled (DS) initialization suggested in [12].

This work has used PSO, IDS and PSO-IDS hybrid technique for RST, all methods used DS initialization. Using these algorithms, reducts can be obtained in an efficient and faster manner. The proposed hybrid feature selection method is efficient and fast for datasets having feature variation from small features (7 features) to reasonably large features (300 features).

To select the features the dependency of the features on the class, has been used. Rough dependency measure has been used as fitness function in our evolutionary methods which is to be maximized. The rough dependency measure has been calculated with the help of lower approximation and positive regions. Swarm based methods are used for picking up the appropriate features. In these algorithms, features in a probable solution feature set are selected simultaneously resulting in improved performance. For PSO, the binary version of particle swarm optimization suggested by [5] and for IDS, the method used in [6] have been used. The given dataset has been used to calculate lower approximation, positive regions and rough dependency measure \(\gamma\).

The structure of the remaining paper is presented.

Preliminaries of the underlying theory, have been discussed in section II. In section III, RST based Feature Selection using IDS and PSO. In section IV, some hybrid methods have been proposed is presented. In section V Experimental Set-up have been described. In Section VI observations from the resulting tables are discussed in Result and discussion, and finally in section VII conclusion of present work is presented.
II. PRELIMINARIES

A. Rough Set Feature Selection preliminaries

Rough Set is a suitable feature selection technique [8]. It is able to predict the decision concepts and the original features. This technique does not need any other information or parameter beside the data provided. To explain the procedure of Rough Set based feature selection [10, 11], through an example using sample dataset, (given in Table I) is attempted here.

Let, \( I = (U, H) \), be an information system, where finite objects are denoted by \( U \) and \( H \) is the non-empty finite set of features including the class labels.

With any \( G \subseteq H \), an equivalence class denoted by \( IND(G) \) is associated.

\[
IND(G) = (x, w) \in U^2 \mid \forall h \in G, h(x) = h(y)..............(1)
\]

The partition of \( U \) can be generated by \( IND(G) \). It is denoted as \( U/IND(G) \) and it can be computed as described below.

\[
U / IND(G) = \Xi U / IND(h) \mid h \in G................(2)
\]

Where \( \Xi \) is defined for sets E and F:

\[
E \Xi F = X \cap W \mid X \in E, W \in F, X \cap W \neq \phi............(3)
\]

x and w are indiscernible by features from G, if \( IND(G) \) contains \((x,w)\). \([x]_G \) denotes equivalence class for \( G \)-indiscernibility relation. For example, if \( G = \{j,k\} \), the objects 0, 4 are indiscernible, and similarly objects 1 and 7 are also indiscernible.

\( IND(G) \) creates following partitions of \( U \) as per eq (2).

\[
U / IND(G) = \{\{1,7\},\{0,3,4\},\{2,5,6\}\} \Xi \{\{0,2,4\},\{1,3,6,7\},\{5\}\}
\]

\[
U / IND(G) = \{\{0,4\},\{2\},\{1,7\},\{3\},\{6\},\{5\}\}
\]

Let \( X \subseteq U \). \( G \)-lower and \( G \)-upper approximations of \( X \), can be approximated as follows using the information in \( G \).

\[
GX = \{x \in U \mid [x]_G \subseteq X\}..............................(4)
\]

\[
\overline{GX} = \{x \in U \mid [x]_G \cap X \neq \phi\}..............................(5)
\]

Rough Set denoted with tuple \( <GX, \overline{GX}> \). Let us have an information system consisting of set of conditional features \( G \) and class label features \( L \), inducing equivalence relations over complete dataset, consequently the positive regions are defined as follows:

\[
POS_G(L) = \bigcup_{x \in U / H} GX..............................(6)
\]

For example:

\[
POS_{(a,b)}([h]) = \{\phi,\{2,5\},\{3,6\}\}
\]

\[
POS_{(a,b)}([h]) = \{2,3,5,6\}
\]

As the information that would make objects 0, 1, 4, and 7 discernible, is not there, so they cannot be classified.

B. Rough Dependency Measure

An important issue in the analysis of data is discovering dependencies between the features of the dataset. The feature set \( L \) fully depends on \( G \), if all the set of features \( G \) uniquely determine feature values for set \( L \).

The degree of dependency in RST, may be defined as follows:

Degree of dependency (denoted \( G \rightarrow_d L \), of \( L \) on \( G \) is \( d(0 \leq d \leq 1) \), if

\[
d = \gamma_G(L) = \frac{|POS_G(L)|}{U}..............................(7)
\]

\( G \) is assumed as set of conditional features and \( L \) as decision feature. If \( d = 0 \) then \( L \) does not depends on \( G \), if \( 0 < d < 1 \), \( L \) partially depends on \( G \), else if \( d = 1 \) then \( L \) fully depends on \( G \).

For the example dataset under consideration (Table I), for feature \{h\} (class label), its degree of dependency on the attributes \{j,k\} is

\[
\gamma_{(j,k)}([h]) = \frac{|POS_{(j,k)}([h])|}{U}
\]

\[
= \frac{2,3,5,6}{\{0,1,2,3,4,5,6,7\}}
\]

\[
= \frac{4}{8}
\]

\[
= \frac{1}{2}
\]

A reduct R i.e. reduced feature set, \( R \subseteq G \), should have same dependency measure as it was in unreduced dataset, for \( H \), to preserve the predictive capability. Reduct sets are not unique.
III. RST-BASED FEATURE SELECTION (RST-FS) USING PSO AND IDS

Swarm based technique performs simultaneous feature selection. The dependency measure $\gamma$ is used as part of fitness function [5]. This method returns optimized $R$.

The particle in swarm algorithm is the combination of 1 and 0. Index of ‘1’ shows that feature should be selected. In PSO and IDS all the particles are initialized in distributed manner (DS initialization) [12], and lower approximation for all classes of all the benchmark dataset, are computed using RST. In the next step, the Positive regions and Rough dependency measures, are computed.

In fitness function weightage assigned to dependency measure term related to the number of selected features are 0.9.

In this paper hybridization based methods of swarm algorithms viz. PSO and IDS are proposed. Hence PSO and IDS are discussed consequently.

A. RST based Feature Selection using PSO

Feature selection has been implemented using PSO based search technique and fitness function is based on rough dependency measure [5]. Here population has been initialized using DS-initialization [12]. PSO is described as follows.

1. Fundamentals of PSO

PSO is a swarm intelligence [13-15] inspired technique. In PSO we have population of several solutions. Every constituent of population tries to move towards the solutions having global value and tries to move towards its own best values with a velocity, to optimize a given fitness function.

$$V_i = wt \cdot V^i + r_1 \cdot c_1 \cdot (P^i - X^i) + r_2 \cdot c_2 \cdot (P^g - X^i)........(8)$$

$$X^i = X^i + V^i.......................(9)$$

where velocity of particles: $V^i$, inertia weight: $wt$, randomly generated numbers: $r_1$ and $r_2$, and constants: $C_1$ and $C_2$ having value 2, $P^i$ (personal or local best) is its own previous best for any particle, and $P^g$ (global best) denotes best of bests, $X^i$ denotes the position of the particle.

2. PSO modification for feature selection

In this work modified version of PSO, suggested in [5] has been used for selection of features where, velocity is obtained as follows.

$$V^i = \text{round}(wt \cdot V^i + r_1 \cdot c_1 \cdot \text{velocityupdate}(P^bi, - X^i) + r_2 \cdot c_2 \cdot \text{velocityupdate}(G^bi, - X^i)).......................(10)$$

Here velocity of $i^{th}$ particle is represented as $V^i$, and it should be rounded to an integer value. In modified PSO, function velocityupdate($G^bi, - X^i$) is defined explicitly and the working of this function can be explained with an example as follows:

Suppose, $P^{bi} = [0 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0]$, and $P^{g} = [1 \ 1 \ 0 \ 0 \ 1 \ 1 \ 1 \ 0 \ 1]$.

Number of ‘1’s, represented as $K^i = 1$, and

Number of ‘1’s, represented as $K^{g} = 3$

The value returned by function velocityupdate = $K^i - K^{g} = 3 - 1 = 2$

In the function velocityupdate, the boundary conditions of $V^i$ are described as follows, $V^i = 1$, if $V^i \leq 1$, and $V^i = \text{round}(K/3)$ if $V^i > K/3$. For other values of $V^i$ it remains unchanged. $K$ is total number of features.

Particle $X^i$ is updated using following function.

$$X^i = \text{positionupdate}(G^bi, X^i, V^i).......................(11)$$

Further details can be found in [5].

B. RST based Feature Selection using IDS


1. IDS Fundamentals

In this method again we initialize the population using distributed sampled (DS) initialization [12]. Parameters required are $C_w$, $C_p$, $C_g$ and random number ‘r’ is generated.

2. Procedure: IDS

IDS is closely related to PSO and it is a kind of adaption of PSO for problems involving discrete variables. The IDS implements the PSO procedure by using following expression for the $j^{th}$ element, $X^{ij}$ of individual $X^{i}$ described in equations (8) and (9).

$$X^{ij} = \begin{cases} X^{ij}, & \text{if } R^{ij} \in [0, C_w) \\ P^{ij}, & \text{if } R^{ij} \in (C_w, C_p) \\ P^{g}, & \text{if } R^{ij} \in (C_p, C_g) \\ x, & \text{if } R^{ij} \in (C_g, 1) \end{cases}.......................(12)$$

From the above equation it may be said that element $X^{ij}$ can take corresponding $j^{th}$ value of $P^i$ (personal best of individual) $P^g$, or $P^g$ (Global best of population) or a random number $x$ or will remain unchanged depending on the range in which the random number $R_{ij}$ falls. The values of the parameters $C_w$, $C_p$ and $C_g$ in eq. (12) are selected such that $0 \leq C_w < C_p < C_g < 1$.

IV. PROPOSED METHODS

In this work, authors propose two hybrid methods viz. (i) Proposed method 1: Parallel PSO-IDS hybrid method, (ii) Proposed method 2: Series PSO-IDS hybrid method. In the proposed algorithms attempt has been made to develop a hybrid algorithm utilizing the capabilities of both PSO and IDS. The proposed algorithms are robust and capable in computing the solution efficiently and faster, maintaining the high classification accuracy with low standard deviation.

A. Proposed Method-1: Parallel PSO-IDS Hybrid

In this method PSO and IDS are hybridized in such a way that half of the population is supplied to PSO and rest half of the population is supplied to IDS. Both PSO and IDS methods will have their own gbests, and the better one is considered as gbest of the population.

B. Proposed Method-2: Series PSO-IDS Hybrid

In this method we hybridized both techniques in such a way that whole population is first supplied to PSO for one iteration and then resulting population is supplied to IDS. The above cycle is again repeated with resulting population, till the termination criterion is satisfied.

Thus in this technique every generation consists of two generations one for PSO and one for IDS.

V. EXPERIMENTAL SETUP

Assumptions and parameters related to experiments are discussed as follows.

A. Dataset

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Dataset</th>
<th>No. of Objects</th>
<th>No. of Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Lung</td>
<td>32</td>
<td>56</td>
</tr>
<tr>
<td>2</td>
<td>Soybean small</td>
<td>47</td>
<td>35</td>
</tr>
<tr>
<td>3</td>
<td>Wine</td>
<td>178</td>
<td>13</td>
</tr>
<tr>
<td>4</td>
<td>LSVT</td>
<td>126</td>
<td>310</td>
</tr>
</tbody>
</table>

All the dataset used here are from UCI repository [16]. Characteristics of the same are described in Table II.
B. DS Initialization

The number of features in a reduct of input dataset may be small, moderate or large and samples of particles picked up from search space by any initialization technique, should be distributed evenly consisting of samples (particles) having small, moderate or large number of features.

In all the methods, used in this work, DS-initialization [12] technique has been used.

C. Parameter Setting

In eq. (10), \( C_1 \) and \( C_2 \) both is set as 2. Parameters for IDS algorithm are chosen as follows: \( C_a = 0.1, C_p = 0.4, C_c = 0.9 \).

As algorithm proceeds iteration by iteration the inertia weight \( wt \) varies gradually from 1 to 0.4 in PSO-DS and in PSO part of proposed methods.

\[
wt = (wt - 0.4) \left( \frac{\text{Max} \_\text{iter} - \text{Current} \_\text{iter}}{\text{Max} \_\text{iter}} \right) + 0.4 \ldots (13)
\]

Fitness Value and Classification Accuracy

The fitness value will depend upon the dependency measure as shown in (7). In order to find the minimal reduce incorporation of other second term is being done as in [5] with low weightage. Formula for fitness function used here is as under.

\[
\text{Fitness} \_\text{Value} = 0.9 \* r(H) + 0.1 \* \left( \frac{|G| - |R|}{|G|} \right) \ldots (14)
\]

Rough dependency measure for a reduct \( R \) with respect to decision feature \( H \) is denoted as \( \gamma(H) \). \(|G|\) is the number of features in the dataset.

Classifiers J48 [17], JRip [18] and PART [19], are used to compute classification accuracy using WEKA [20, 21], a data mining workbench.

VI. RESULT AND DISCUSSION

Tables III and IV presents the comparison of performance of DS-initialized PSO and IDS with proposed parallel and series hybrid methods.

### TABLE III: Comparison of Selected Number of Features and Execution Time

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Feature Selection Method</th>
<th>No. of Features Selected (Execution Time)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Min</td>
<td>Mean</td>
</tr>
<tr>
<td>Lung (56)</td>
<td>PSO-DS</td>
<td>3(17.57)</td>
</tr>
<tr>
<td></td>
<td>IDS-DS</td>
<td>5(11.06)</td>
</tr>
<tr>
<td></td>
<td>Proposed-1</td>
<td>4(10.61)</td>
</tr>
<tr>
<td></td>
<td>Proposed-2</td>
<td>3(11.17)</td>
</tr>
<tr>
<td>Soybean</td>
<td>PSO-DS</td>
<td>2(15.51)</td>
</tr>
<tr>
<td>small (35)</td>
<td>IDS-DS</td>
<td>2(12.75)</td>
</tr>
<tr>
<td></td>
<td>Proposed-1</td>
<td>2(9.91)</td>
</tr>
<tr>
<td></td>
<td>Proposed-2</td>
<td>2(11.57)</td>
</tr>
<tr>
<td>Wine (13)</td>
<td>PSO-DS</td>
<td>2(23.95)</td>
</tr>
<tr>
<td></td>
<td>IDS-DS</td>
<td>2(230.83)</td>
</tr>
<tr>
<td></td>
<td>Proposed-1</td>
<td>2(228.66)</td>
</tr>
<tr>
<td></td>
<td>Proposed-2</td>
<td>2(211.28)</td>
</tr>
<tr>
<td>LSVT (310)</td>
<td>PSO-DS</td>
<td>14(185.92)</td>
</tr>
<tr>
<td></td>
<td>IDS-DS</td>
<td>18(159.84)</td>
</tr>
<tr>
<td></td>
<td>Proposed-1</td>
<td>3(172.11)</td>
</tr>
<tr>
<td></td>
<td>Proposed-2</td>
<td>12(157.47)</td>
</tr>
</tbody>
</table>

In the case of reasonably smaller dataset Wine the proposed methods do not have any special or significant improvement. But it certainly improves the classification accuracy of the selected feature set very frequently (Table IV).

In the case of Lung dataset both the proposed hybrid methods reduces the average number of selected features up to less than 5. Both the hybrid methods Proposed-1 (parallel hybrid PSO-IDS) and Proposed-2 (series hybrid PSO-IDS) either improves or have the equivalent accuracy than any other method used in this paper.

In the case of Soybean small it is observed from Table IV that the proposed methods provide the guaranteed results every time, which is the optimal result also, which means lowest number of selected features with highest classification accuracy and s.d. is zero.

In the case of LSVT, proposed-1 (parallel hybrid PSO and IDS) method provides the best feature selection. Here the minimum number of selected feature are only 3 out of 310 and average number of selected feature are less than 9, without compromising the classification accuracy.

Proposed-2 (series hybrid PSO-IDS) method gives minimum and average number of selected features 12 and 14.75 respectively (Tables III and IV).

The 3 selected features (68, 200, 268) are the features having more classification accuracy compare to that of the 12 selected features (3, 8, 59, 111, 126, 154, 181, 214, 224, 228, 231) out of total 310 features.

In case of proposed hybrid algorithms it seems that the search saturates before 100 iterations in the case of Soybean small and Wine as they have zero standard deviation for selected features.

Further results on the datasets with large number of features (Lung, Soybean small and LSVT) show the power of proposed methods. In these cases, it improves classification accuracy of the dataset with reduced number of features. Both the proposed methods yield comparable selected features and fitness values.

The proposed methods drastically reduces the selected features and execution time especially in case of LSVT dataset. This shows the improved power and robustness of proposed methods for high dimensional dataset.

### TABLE IV: Comparison of Classification Accuracy

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Classifier</th>
<th>Feature Selection Method</th>
<th>Classification Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>Max</td>
</tr>
<tr>
<td>Lung (56)</td>
<td>J48</td>
<td>PSO-DS</td>
<td>87.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IDS-DS</td>
<td>87.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Proposed-1</td>
<td>84.37</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Proposed-2</td>
<td>87.5</td>
</tr>
<tr>
<td>Soybean</td>
<td>JRip</td>
<td>PSO-DS</td>
<td>7.5</td>
</tr>
<tr>
<td>small (35)</td>
<td></td>
<td>IDS-DS</td>
<td>87.5</td>
</tr>
</tbody>
</table>

Published By: Blue Eyes Intelligence Engineering & Sciences Publication
REFERENCES


VII. CONCLUSION

In this work improvement is demonstrated in selecting appropriate features using RST method by introducing two new hybrid methods. The performances of the proposed methods improve with DS-initialization. Effect of hybrid methods in terms of classification accuracy is more visible for reasonably large sized dataset. It is observed that in all of the datasets Parallel Hybrid method performs better than Series Hybrid method. However, in certain datasets the performance of Series Hybrid method was found better. Hence both the proposed methods should be attempted for a given dataset.
AUTHORS PROFILE

Tarun Maini is presently a research scholar in Systems Engineering, run by Department of Electrical Engineering, IIT(BHU), Varanasi, India. He has done his B.E. in 2001 and M.Tech. in 2012 in Computer Science and Engineering. He has published several papers in indexed international journals and conferences. His research interest includes computational intelligence, machine learning, feature selection, swarm and evolutionary computation, rough and fuzzy rough sets, etc.

Rakesh Kumar Misra received his B.Sc. Engg. in Electrical Engineering from DEI, Dayalbagh, Agra India in 1995. He received his M.Tech. in Engineering Systems from DEI, Dayalbagh, Agra, India in 1997. He received his Ph.D. degree from IIT BHU, Varanasi India. He is currently a Professor in the department of Electrical Engineering, IIT(BHU), Varanasi, India (erstwhile IT, BHU Varanasi). He has published several papers in international journals and conferences. His research interest includes evolutionary computation, computational intelligence, Power distribution systems, AI applications in power systems etc.

Devender Singh received his B.E. in Electrical Engineering from SVREC, Surat, India in 1993. He received his M.E. in Electrical Engineering from MNREC, Allahabad, India in 1999. He received his Ph.D. degree in Electrical Engineering from IT BHU, Varanasi India. He is currently a Professor in the department of Electrical Engineering, IIT(BHU), Varanasi (erstwhile IT, BHU Varanasi), India. He has published several papers in different IEEE Transactions and IEEE international conferences. His research interest includes Evolutionary computation, Computational intelligence, Power systems, Distribution generation and planning, state estimation, short term load forecasting, AI applications in power systems etc.