A swarm intelligence based multi-label feature selection method hybridized with a local search strategy

A. Rafiee[1], P. Moradi[2*], A. Ghaderzadeh[3]

[1,3]Department of Computer Engineering, Islamic Azad University, Sanandaj Branch, Sanandaj, Iran.
[2]Department of Computer Engineering, University of Kurdistan, Sanandaj, Iran.
[1]arafie@iausdj.ac.ir, [2]p.moradi@uok.ac.ir, [3]b.ghaderzadeh@iausdj.ac.ir
[*]Corresponding author

Received: 30-10-2021, Revised: 02-02-2022, Accepted: 26-04-2022.

Abstract
Multi-label classification aims at assigning more than one label to each instance. Many real-world multi-label classification tasks are high dimensional, leading to reduced performance of traditional classifiers. Feature selection is a common approach to tackle this issue by choosing prominent features. Multi-label feature selection is an NP-hard approach, and so far, some swarm intelligence-based strategies and have been proposed to find a near optimal solution within a reasonable time. In this paper, a hybrid intelligence algorithm based on the binary algorithm of particle swarm optimization and a novel local search strategy has been proposed to select a set of prominent features. To this aim, features are divided into two categories based on the extension rate and the relationship between the output and the local search strategy to increase the convergence speed. The first group features have more similarity to class and less similarity to other features, and the second is redundant and less relevant features. Accordingly, a local operator is added to the particle swarm optimization algorithm to reduce redundant features and keep relevant ones among each solution. The aim of this operator leads to enhance the convergence speed of the proposed algorithm compared to other algorithms presented in this field. Evaluation of the proposed solution and the proposed statistical test shows that the proposed approach improves different classification criteria of multi-label classification and outperforms other methods in most cases. Also in cases where achieving higher accuracy is more important than time, it is more appropriate to use this method.

Keywords
Feature selection, Multi-label classification, Local search strategy, Swarm intelligence, Particle swarm optimization.

1. Introduction

Supervised learning is associated with inferring the relations between input instances and class labels. In traditional classification tasks, each instance is associated with a class label. However, in many real-world scenarios, an instance may be associated with multiple labels. For example, in the news classification, a piece of news related to the publication of a new iPhone cell phone is associated with both the business and the technology labels. In other words, each instance could be related to a set of labels instead of one label. The existing methods can be divided into two main categories. The first category is transforming the multi-label learning problem into a set of single classification problems and using the single-label learning algorithms. The second category is adapting the existing classifier algorithms to address the multi-label classification task. Achievement of high classification accuracy is an essential task in the real world because it determines the success of applications. The feature selection methods have attracted much attention because they can significantly improve classification accuracy by selecting of most prominent features. On the other hand, many irrelevant and redundant features may substantially degrade the accuracy of learned models and reduce the performance of the learning models. So the main problem is reducing the dimensions while considering the computational complexity of the model. For this reason, feature selection is the first step in the classification process. Real data may contain features from variable communications to predict class labels. For example, to predict a disease label such as "diabetes," the feature gender is less relevant than the feature age. The irrelevant features typically decrease the accuracy of the classification model and the efficiency of the computational resources. Therefore, the main idea of feature selection is to select a subset of salient features by eliminating irrelevant features and highly correlated redundant features. This reduction helps to accelerate the learning process.

There are three main types of feature selection methods that are used in a classification called filter [1-7] wrapper [8-12], and embedded [13-15] The filter method seeks features that maximize a criterion, without depending on any particular learning model. Wrapper methods employ a learning algorithm to evaluate feature subsets. In other words, in each iteration for each found solution, it needs to run a classifier. The filter-based methods are faster than wrapper-based methods, but due to the lack of a learning model in their search process, the final result will be less informative than wrapper-based

methods. Finally, embedded methods use the benefit of both wrapper and filter methods.

The wrapper-based methods are divided into sequential and global search methods [16-18]. The sequential search methods are divided into backward and forward search methods. The backward search method starts with a complete set of features. In each step in the method, one feature is removed concerning the performance of the classifier. The forward search method begins with an empty set of features. In each step in the process, one feature is added to the features set to increase the classifier's performance. In recent years, the metaheuristic algorithms have been attracted a lot of attention due to their excellent performance in solving feature selection problems such as Ant Colony Optimization (ACO) [5, 6, 19-21], Genetic Algorithm (GA) [11, 22-24], Simulated Annealing (SA) [25, 26] and Particle Swarm Optimization (PSO) [10, 27-31].

Recently, a wrapper algorithm for feature selection based on particle swarm optimization has been proposed [10]. The results of this paper show the efficiency of this algorithm. However, in some situations, the results show low accuracy and trapping in a local optimum. To solve this issue, in this paper a local search strategy has been proposed, which improves the algorithm's performance and accuracy and increases the convergence speed. According to this strategy, the similarity of each feature with other features is calculated based on the Mutual Information (MI) theory. Redundant features are usually more similar to the other features. Also, the relation of each feature is calculated with the class labels. Finally, a criterion is proposed based on these two parameters that show the feature's relevance. The features are sorted by this criterion and divided into two categories. The first are those of the independent and related features, and the second category contains less relevant and more redundant features. To employ this strategy into the search process, a mutation phase is added to the Particle Swarm Optimization (PSO) algorithm. Based on this step, relevant features are added from the first group to the solution and remove those which are belong to the second category for each response in the population. This strategy will lead to a rapid convergence and convergence speed. In other words, the proposed method uses the correlation between features and problem labels to improve the search process. in addition, this method tries to decrease the number of features in the final subset by estimating the size of the subset. The main objectives and novelties of the proposed method are summarized as follows:

(1) A local search strategy integrated with PSO is proposed to reduce the dimensionality of the feature subset and select a desired subset.

(2) The goal of the local search strategy is to guide the search process of PSO to select distinct features to their correlation information.

(3) The use of correlation information to guide the search process in particle swarm optimization is that the non - correlated (non - similar) characteristics are more likely to be selected from correlated features.

(4) To select a small number of salient features using a specific subset size scheme is proposed.

(5) This algorithm has better performance and faster convergence speed compared to the algorithm proposed in [10].

The remainder of this paper is organized as follows. Section 2, provides the existing multi-label feature selection methods with their pros and cons and summarize them in a table. In the third section, the proposed method is briefly described. The experiments and results are presented in Section 4. Finally, in the fifth section, the paper concludes.

2. Related works

The primary purpose of feature selection methods is to reduce feature space dimensions and improve classification performance by eliminating irrelevant and redundant features. In general, feature selection methods are divided into three categories: Filter, Wrapper, and Embedded. The filter methods independently reduce the learning dimension of the data. These methods rank features based on some criteria and delete features that do not have good scores. Various criteria have been proposed for assessing the quality of selected features, including Mutual Information (MI) [32], Information Gain (IG)[3], Relief-F, and its extensions[2].

To solve the multi-label feature selection problems, one should first transform the multi-label data into one or more single-label data. Then The feature selection algorithms are performed on these single label data. Chen et al. [33] proposed a systematic document transformation framework that assigns the labels weights to a multi-label document using an entropy-based conversion technique. Also, **Spolaôr** et al. [34] used **Label – Powerest** (LP) and Binary Relevance (BR) techniques to data transformation and compare the efficiency of these methods aim to select the Relief-F and Information Gain (IG). However, BR is not able to distinguish between tag and LP in the training phase. Also, there is no distinction between the features of the sample. In contrast, BR is simple, and LP can detect the correlation between labels. This feature has fewer features and is due to the interaction between the features. It is a multivariate search capable of distinguishing two samples from a class or different classes. Dimensionality reduction without hurting the classifier performance determines the dependence between each feature and class label. In [35], a pruned problem procedure (PPT) has been introduced as a greedy feature selection method based on multi-dimensional information. This method is ineffective in the high dimensions. In this method, LP can also detect the correlation between the labels and the dependency between the labels. In [36] has been employed to transform multi-label data into single label data. Next, a greedy feature selection method is introduced based on the multiple information. The method presented in [37] expands the initial results of [35] and suggests a method of pruning parameter selection. This method considers the possible dependencies between class labels and between features during the selection process. Also, a method for automatic tuning of

pruning parameters based on permutation testing has been proposed, and scalability is reduced. Authors in [2] propose a technique that transforms the multi-label problem to a single label problem using PPT and then employs the Relief-F algorithm to assign weight to each feature. They also present a further development of Relief-F for the multi-label feature selection problem. This method is only for binary classes with no missing value. Relief also selects fewer features and is since it considers the interaction between features. It is a multivariate search and can clean two instances of a different class. In another study, a fast feature selection method based on information-theoretic has been proposed. The approach increased the speed of the search process by removing unnecessary measures and identifying the combinations of important labels [38]. This search algorithm finds random subsets in each iteration, and therefore a large number of features are selected. In this way, the cost of complexity is reduced by the Score function. Lin et al. proposed a multi-label feature selection method that selects outstanding features based on the neighbour-hood mutual information. The proposed approach introduces the margin of the instance to granulate all instances under different labels at first. They also define three other concepts of the neighbour-hood mutual information based on various cognitive viewpoints. Then, they have introduced an optimization objective function to evaluate the quality of candidate features. In this method, the classifier has high computational complexity and does not consider the dependency between the labels. Also, three different neighbour-hoods are defined that are used to evaluate the quality of selected features [1].

In embedded methods, a search algorithm is performed to find a suitable subset of features by a learning algorithm. Lin et al. [13] proposed an embedded feature selection approach into multi-label classification, Multi-label Embedded Feature Selection (MEFS). To evaluate the features, "MEFS" adopts an evaluation criterion and uses a search strategy to find a suitable feature subset. The experimental results show that the performance of the proposed algorithm is superior to the filtering methods in the Music Emotion dataset. In [14], an embedded feature selection method has been proposed for classification with missing labels. The missing labels are covered using the robust linear, and l2 determines the specified features, p-norm effectively ($0<p\leq1$). In this method, the classifier is of high computational complexity, and the dependency between the labels is not intended. Also, missing labels are recovered and are predictable.

Wrapper methods use a predefined learning algorithm. The method selects features that increase the efficiency of the learning algorithm. In [11], the feature extraction method is presented based on PCA, combined with a genetic algorithm selection method. In[8], the authors present a memetic selection method to redefine the subset of the features found by the genetic search. In this method, a genetic algorithm is a random method that can select many features. Also, the GA algorithm is simple to implement, and the coating method is highly accurate. In [36], a multi-label feature selection algorithm is introduced based on mutual information and GA. In the first step, the mutual information is used to complete the selected features locally. Then, based on the results of this step, the GA selects a subset of optimal global features.

Some tasks have been done in the multi-label feature selection that uses the concept of multi-objective optimization. In all of these methods, wrapper methods are used that are usually based on metaheuristics. For example, Zhang et al. [10] used the PSO algorithm, and the authors in [12] used GA. In both studies, two or three evaluation criteria are simultaneously optimized. For example, in [12], the authors used accuracy, Hamming, and micro-Average measures. This method is very high in high-dimensional data. In this method, BR is used with a simple implementation. LP is also able to detect the correlation between labels. The accuracy of this method is higher than the filtering methods. Authors in [39] used the Hamming loss and accuracy measures, while Zhang et al. in [10] used the hamming loss and the number of selected features. This algorithm is random and therefore increases the probability of a large number of selected features. The PSO has fast convergence and utilizes multi-objective features.

The related publications to the multi-label feature selection methods are summarized in Table 1. The first column, feature selection category, demonstrates the feature selection classification method (Embedded, Filter, or Wrapper). Also, the second column, data transformation, indicates whether this method is used in the transformation technique. The next column, the classification algorithm, shows the classification strategy, while the feature selection column shows the feature selection technique used in each method. Finally, in the data sets column, the datasets that are used in each method are listed.

3. Proposed method

The PSO is a swarm-intelligence optimization approach that was first proposed in 1996 that inspired by the behaviour of birds when searching for food [40]. In PSO, first, an initial response set is generated. Then, to find an optimal solution in the possible response space, or for generations, a response is made. Each particle is defined as multi-dimensional with two values of location and velocity. In each stage of the particle movement, velocity and position of each particle, guides the best answers are determined in terms of merit for all particles.

The most important advantage of the PSO algorithm over other algorithms is finding a better solution for feature selection within shorter time. Also, the premature convergence and weak point in the appropriate adjustment near the local optimum points are disadvantages of the PSO algorithm. In this paper a new feature selection method based on the PSO algorithm is developed to overcome these weaknesses. In the proposed method, the final subset of features is selected through a few steps. Figure 1 shows the general procedure of the proposed method in multi - label feature selection. By this figure, first, the size of the feature set is estimated automatically. In the next step, all the features are classified into similar and dissimilar groups using correlation information. Then, the binary PSO algorithm is integrated with a specific local search strategy that incorporates local information into the

search space. In the next step, several predetermined particles are produced. These particles are moved to their new positions according to local best positions and the global best of the swarm. Next, each particle searches in the local area concerning the features correlation information. In this step, the fitness value of each particle is calculated. In the last step, global and local best particles are replaced with those of previous values. When the stopping criterion is satisfied, the final feature set is presented. Also, the pseudo-code of the proposed method is presented in Fig.2.

Table 1. Summary of publications on multi-label feature selection.

Methods	Category	Transformation	Classification Algorithm	Feature selection	Application
ELA [33]	Filter	Entropy	SVM	Information Gain, OCFS	Text
RF-BR [34]	Filter	LP, BR	BRKNN	Information Gain, Relief-F	Various domains
RF-LP [34]	Filter	LP, BR	BRKNN	Information Gain, Relief-F	Various domains
IG-BR [34]	Filter	LP, BR	BRKNN	Information Gain, Relief-F	Various domains
IG-LP [34]	Filter	LP, BR	BRKNN	Information Gain, Relief-F	Various domains
MI [37]	Filter	PPT	MLKNN	Mutual Information	Various domains
Kim [8]	Wrapper	-	MLNB	Genetic Algorithm	Various domains
MLFS [9]	Wrapper	-	MLKNN	Mutual information	Various domains
MEFS [13]	Embedded	LP	Various classifiers	Max Average, LP-Chi	Music
MLNB [11]	Wrapper	-	MLNB	Genetic Algorithm	Various domains
MI [32]	Wrapper	-	MLNB	Mutual Information	Various domains
RELIEF-F [2]	Filter	PPT	MLKNN,BRKNN	Relief-F	Various domains
Khan [12]	Wrapper	-	SVM, DT	Genetic Algorithm	Music
MPSOFS [10]	Wrapper	-	MLKNN	PSO	Various domains
MMFS [39]	Wrapper	-	MLKNN	NSGA-II	Various domains
MLINFOGAIN [41]	Filter	-	Various Classifiers	ML Information Gain	Various domains
MFNMI [1]	Filter	-	MLKNN	Mutual information	Various domains
MDMR [4]	Filter		MLKNN, LIFT	max-dependency, min-redundancy	Image
MLMLFS [14]	Embedded	-	MLKNN	$l2,p$-norm	Various domains
Doquire [35]	Filter	PPT, LP	MLKNN, SVM	Mutual information	Various domains
IGMF [3]	Filter	-	SVM, MLKNN	ML Information Gain	Various domains

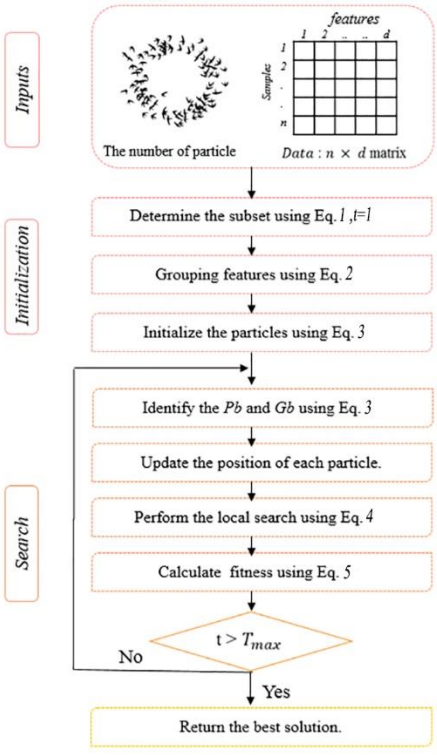


Fig.1. The general procedure of the proposed method in multi - label feature selection.

3.1.Determining the initial number of features

In the proposed method, a probabilistic random function is used to provide a lower-valued random number. The aim is to determine the subset size s in a bounded region rather than the boundless region [42]. To this end, a probabilistic formula (Eq.1) is used to determine the initial size of the feature subset ($s \leq f$) [43].

- $P_s = \frac{f-s}{\sum_{i=1}^{l}(f-i)}$ (1)

where f refers to the number of original features in a given dataset, s is the number of selected features, P_s describes the difference between s and ($P_s = f - s$), and P_s is the value of the estimate s as an initial number of features. Note that the goal of feature selection is to select the minimal number of informative features. To this end, in this equation, P_s decreases with increase in s. The number of initial features is estimated by the roulette wheel strategy and is based on the probability P_s. The value s is randomly selected in the range of $[\alpha.\beta]$, where $\beta = \varepsilon.f$ and α depends on the given dataset and generally it is set to 3. The parameter ε are defined by the user to control β. In this case, the search space becomes more significant for finding the salient features, with a high computational cost. Thus, it should generate ineffective subsets of features. It is necessary to note that the presented scheme tries to provide a

smaller set size. It only depends on the value of parameter ε determined by the user based on his knowledge about the datasets.

Algorithm 1. The proposed multi-label feature selection algorithm

Input: Dataset, mcn: maximum cycle number, np: number of particles
Output: F: Final set of features
Begin Algorithm
Begin (initialize)
Determining number of feature (s).
for i= 1 to np
Create random particle X_{ij}.
Create random velocity V_{ij}.
End
Compute CC_{ij} and Cor_i for all features.
$Cor_i \geq Cor_{mid}$ Similar feature
$Cor_i < Cor_{mid}$ Dissimilar feature
End (initialization)
For i= 1 to mcn
Compute updating the particle velocity.
Compute updating the position of particle
X_s includes the similar features.
X_d includes the dissimilar features.
Remove all feature in X_s that is 0 in particle x.
Remove all feature in X_d that is 0 in particle x.
Calculate the value of n_s and n_d
Apply local search strategy
Calculate fitness for each particle
Choose the features that have less fitness.
End (iteration)
Return "F: subset features"
Eng Algorithm

Fig.2. The pseudo-code of the proposed method.

3.2. Grouping of the features

To find the relationships among the features, they are divided into similar and dissimilar groups. Then the algorithm can choose distinct features for strong learning models.[44]. Correlation is one of the most common and useful statistics that describe the relationship between the two variables. To measure the correlation between different characteristics of a Pearson correlation coefficient, the Pearson correlation coefficient [45] is derived from the following equation at first:

$$CC_{ij} = \frac{\sum_{k=1}^{m}(x_i(k) - \bar{x_i})(x_j(k) - \bar{x_j})}{\sqrt{\sum_{k=1}^{m}(x_i(k) - \bar{x_i})^2}\sqrt{\sum_{k=1}^{m}(x_j(k) - \bar{x_j})^2}} \quad (2)$$

where CC_{ij} is the correlation coefficient between two properties of i and j, and m is the number of samples. x_i (k) and x_j (k) denote the values of the feature vector i and j for sample k. If the correlation coefficient is a high value, then it means that the two features are very similar. On the other hand, lower values indicate lower similarity. After calculating the correlation coefficient for all possible combinations of properties, the correlation value for feature i is given as follows:

$$Cor_i = \frac{\sum_{j=1}^{f}|cc_{ij}|}{f-1} \qquad if\ i \neq j \quad (3)$$

where f is the total number of features and CC_{ij} is the correlation coefficient between the properties i and j. finally, two groups are created with the size of f/2. The first group includes features with the highest degree of correlation, called the similar group (S). Another group has a lower degree of correlation than features in S, called the dissimilar group (D). The first feature in group S and the last feature in group D are the most and least correlated features in the dataset.

3.3. Initializing and updating the particles

In this step, each particle is defined by a binary vector. The length of the vector is equal to the number of original features. If the value of a cell in this vector is set to 0, it means the corresponding feature is not selected, and when the value is set to 1, it means the related feature is selected. Then, for each particle, a velocity vector is generated. Each cell of the velocity vector is set to a random value in the range of (0, 1). The length of the velocity vector is equal to the length of the particle vectors. The velocity of each particle is changed according to the following equation (Eq.8):

$$V_i(t+1) = V_i(t) + C_1.rand_1.\left(pbest_i(t) - X_i(t)\right) + C_2.rand_2.\left(gbest_i(t) - X_i(t)\right) \quad (4)$$

where t represents the number of iterations, pbest is the best value obtained by particle i from the start of the algorithm's running to iteration t. gbest is the best value obtained by all particle from the beginning of the algorithm's running to iteration t. and C_1, and C_2 are two real numbers. The parameters $rand_1$ and $rand_2$ are random numbers in the range of (0,1).

It should note if the velocity of the particles increases and exceeds the V_{max} , the speed of this dimension must be restricted to V_{max}, according to the following equation:

$$if\ V_i(t+1) \notin (V_{min}.V_{max})\ then\ V_i(t+1) = max(min(V_{max}.V_i(t+1)).V_{min}) \quad (5)$$

where V_{max} and V_{min} are user-specific parameters (here $V_{max} = 4$, $V_{min} = -4$).
Note the position of a particle is updated with the function $S\left(V_i(t+1)\right)$ according to the following equation (6):

$$S\left(V_i(t+1)\right) = \frac{1}{1+e^{v_i}} \quad (6)$$

$$if\ rand < S\left(V_i(t+1)\right) then\ X_i(t+1) = 1 \qquad else \qquad X_i(t+1) = 0$$

If $V_i(t+1)$ is larger than a random value, then its position value is denoted by 1. It means that the corresponding feature on the next update is also selected. However, if $V_i(t+1)$ is smaller than the random value, the position value is displayed with 0 and means that the corresponding feature is not selected.

3.4. Local search operations

At this stage, the "ADD" and "Delete" operators are used to improving the local search for the particle for a specific particle. A particle uses the "ADD" operator to select a number of the desired features of the operator and uses the "Delete" operator to remove several features available [42].

In the local search operation, all selected features are extracted by the particle at first. Then, in each particle,

the algorithm identifies several bits of a newly generated particle, i.e., 10011001, and puts them into a subgroup X, i.e., X= f1, f4, f5, f8. Each element of X is compared with the groups D and S; the X is divided into subgroups X_d and X_s. X_d contains the features in D, whereas the remaining features of X are located in X_s [46]. Then, in the next step, all of the features in X_d and X_s are sorted ascending, respectively, based on their correlation values.

Therefore, the first and last features of X_d and X_s have the highest distinction and the most similarity, respectively. Finally, the number of similar and dissimilar features are determined by computing the ns and nd values, respectively. Here, $ns = \mu.s$ and $nd = (1-\mu).s$.Where μ is a specific user parameter and s is the initial subset of features estimated at the first stage. If the number of dissimilar features in a particle becomes smaller than nd ($|X_d| < nd$), then $(nd - |X_d|)$ features are added to the particle, otherwise $(|X_d| - nd)$ must remove features from the particle. Instead, if the number of similar features becomes larger than ns ($|X_s| > ns$) , $(|X_s| - ns)$ features in X_s have to be removed from the particle. In other words, when the number of similar features is smaller in the generated particle from ns, $(ns - |X_s|)$ features are added to the particle (e.g. X={f1, f3, f4, f8, f9, f10}.

3.5. Calculating fitness

The proposed method is used the ML-KNN [47] classifier to evaluate the selected feature subset. ML-kNN is the first ML lazy learning algorithm proposed by Zhang which is based on the maximum-posterior principle. It is derived from traditional k-Nearest Neighbor (kNN) algorithm. For each unseen instance, firstly its k nearest neighbours are identified and then the label set for the unseen instance is identified by considering the label sets of identified neighbors instances. Different criteria for assessing the efficiency of a multi-label classifier are designed, including hamming loss, ranking loss, one-error, etc. In this paper, the Hamming loss is used to evaluate the classification error rate of a particle. Then, the particle with a minimum value of Hamming is chosen (Eq.11). The lower the Hamming loss measure is, the better performance of the classifier is obtained[10].

- $F(P_i) = Min\ (Hammingloss(P_i.S))$ (7)

4. Experimental studies

4.1. Data sets

In this paper, five datasets of different applications have been developed from the various databases[1,2]. Table 2 shows the characteristics of these data sets, including the name, number of samples, number of features, the number of labels, and their range.

Table 2. Description of multi-label datasets

Name	Instance	Features	Labels	Domain
Image	2000	294	5	Image
Yeast	2417	103	14	Biology
Birds	645	260	19	Audio
Scene	2407	294	6	Image
Emotion	593	72	6	Music

4.2. Performance evaluation criteria

Different criteria for evaluating the efficiency of a multi-label classifier are designed, including hamming loss, one-error, coverage, ranking loss, average, and so on [47]. In this paper, we use a number of these criteria to evaluate this method and compare it with other methods. These metrics are defined in (Equations 8-13).

- Hamming loss (HL): These criteria evaluate how many times an instance-label pair is misclassified.

$$HL = \frac{1}{p}\sum_{i=1}^{p}\frac{1}{q}\,|h(x_i)\Delta Y_i| \tag{8}$$

where Δ denotes the symmetric difference between two sets.

- Ranking loss (RL): This criterion estimates the average fraction of miss ordered category pairs.

$$\mathrm{RL} = \frac{1}{\mathrm{p}}\sum_{\mathrm{i}=1}^{\mathrm{p}}\frac{1}{|\mathrm{Y_i}||\overline{\mathrm{Y}}_\mathrm{i}|}\,|\{(y'.y'')|\mathrm{f}(\mathrm{x_i}.y') \le \mathrm{f}(\mathrm{x_i}.y''),(y',y'') \in \mathrm{Y_i}\times\overline{\mathrm{Y}}_\mathrm{i})\}| \tag{9}$$

where $\overline{\mathrm{Y}}_\mathrm{i}$ complementary set of Y in label space.

- One-error (OE): One error computes how many times that the top-ranked label is not relevant.

$$OE = \frac{1}{p}\sum_{i=1}^{p}[[\ [argmax_{y\in Y} f(x_i\,.y)] \notin Y_i]] \tag{10}$$

- Coverage (CO): It computes the average number of steps to go down the list of labels to cover all the relevant labels of the example.

$$CO = \frac{1}{p}\sum_{i=1}^{p} max_{y\in Y_i} rank_f(x_i.y) - 1 \tag{11}$$

- Average Precision (AP): This criterion estimates the average percentage of relevant labels ranked higher than a particular label $y \in Y_i$.

$$AP = \frac{1}{\mathrm{p}}\sum_{\mathrm{i}=1}^{\mathrm{p}}\frac{1}{|\mathrm{Y_i}|}\sum_{\mathrm{y}\in\mathrm{Y_i}}\frac{|\{y'|\mathrm{rank_f}(\mathrm{x}.y') \le \mathrm{rank_f}(\mathrm{x_i}.y)|}{\mathrm{rank_f}(\mathrm{x_i}.y)} \tag{12}$$

- Accuracy (AC): It calculates the correctly predicted labels among all true predicted labels, is defined as:

$$AC = \frac{1}{\mathrm{p}}\sum_{\mathrm{i}=1}^{\mathrm{p}}\frac{|Y_i \cap Z_i|}{|Y_i \cup Z_i|} \tag{13}$$

Moreover, the Wilcoxon test is used to compare the efficiency. This method is a statistical inference test used to evaluate the similarity of two dependent samples with a rating scale. It computes the difference between each pair of objects with a p-value and analyses the

[1] http://mulan.sourceforge.net/datasets.html
[2] https://cse.seu.edu.cn/people/zhangml/Resources.psp#data

differences. In the case of comparing feature selection methods, the null hypothesis shows no difference in the performance of the two feature selection methods. If the value of p is less or equal to a significant level ($\alpha = 0.05$), the null hypothesis is rejected, and it can be inferred that there is a significant difference between the two methods[48].

4.3. Results and Discussion

Several experiments have been carried out to evaluate the efficiency of the proposed method and compare it with the MPSOFS[10], Pareto-FS [49], MLCSO [50], PMU[51], ELA-CHI[33], PPT-CHI[36], PPT-MI[35] methods. In all the methods is used the ML-KNN [47] (Multi-Label k-Nearest Neighbour) classifier (k=10). Besides the MPSOFS method and the proposed method (MPSOLS), wrappers, are other filter-type methods. Data sets such as Yeast, Scene, Emotions and Birds have been used in the experiments. The comparison table shows the performance of the proposed algorithms in terms of hamming loss, ranking loss, one-error, coverage, and average precision. For each evaluation criterion, the sign " in the form of a higher value " is better, and the sign is defined as " the value of the value." The best results are highlighted. The last row of each table results from a statistical comparison of the proposed method with other methods. The sign (+) indicates the superiority of the proposed method against other feature selection methods, and the negative sign indicates that the proposed method is not superior. In contrast, the sign (=) shows no significant difference between the two feature selection methods.

From Table 3, it can be concluded that the proposed method can achieve better classification performance than other selection methods. This table shows that the proposed method achieves the highest average precision and minimum hamming loss and ranking loss in all data sets compared to all methods. Due to the Hamming loss measure results, it is possible to conclude that the proposed method in all data sets has been better than other methods. Also, the average ratio to the strongest competitor, namely MPSOFS on the Emotions, Birds, Yeast, Scene, and Image data sets, has improved 0.015, 0.0023, 0.002, 0.0093, and 0.07, respectively.

As you see, in all criteria except coverage, our proposed method has had the best performance. In the coverage criterion, the proposed method is second place after the MPSOFS method. Also, in some data sets is very significant with other methods. For instance, the coverage in the Birds data set is 2.51 for the proposed method. The variation of coverage in this method is almost equal to 0.3. Also, one - error criterion for the proposed method in the Emotion dataset is 0.22, whose dispute is 0.1 by Pareto-FS method, 0.13 and with ELA - chi method. While considering this criterion in other data sets, the proposed method has achieved better results. The proposed method is ranked first regarding the hamming loss in some data sets with a high variation compared to other methods. The proposed method measures the criterion in this dataset in terms of the PPT - MI method is 0.15, or in the Scene, the dataset relative to the Pareto-FS method is 0.28. The table shows that the proposed method has the best performance among all selection methods. As observed in Table 3, the results show a better performance of the proposed method than the other methods. Note that the proposed method uses a classifier to evaluate feature subsets through the search process and thus it is classified as a wrapper method. Compared to the filter methods our method obtained higher accuracy. Moreover, in comparison with the other wrapper methods such as MLCSO and MPSOFS, the proposed method uses a local search strategy which in fixed number of iterations, it converged to near optimal results and obtained higher accuracy. Considering the last row of Table 3 it can be concluded that the proposed method has a statistical superiority in all cases and fails all other methods in terms of statistics. According to to the obtained results, the best results are obtained from the proposed method's subset of the selected features and the classifier performance. The proposed method is best compared to other methods, and the Wilcoxon test results demonstrate that the proposed method is statistically superior to other methods.

Table 4 shows the average execution time of the feature selection methods in 100 independent runs. As expected, the ELA-CHI and PPT-MI methods have lower execution times than those of the evolutionary algorithms. This is due to using a learning model to evaluate the feature subsets of the solutions. Since these methods have higher accuracy than the other methods and considering the fact that the feature selection process is an offline task, there is trade-off between the accuracy and the executing time criteria in choosing feature selection methods.

In Figures 3 - 5, all criteria have been checked for the Scene, Yeast and Birds data sets. The horizontal and vertical axes in these shapes show the number of selected features and the evaluation criterion. Eight graphs are presented in each of the figures, each representing a feature selection method. These methods show the performance of the test algorithms in terms of hamming loss, ranking loss, one-error, coverage, and average precision. The number of features selected by the user is defined, and each method has 200 independent iterations.

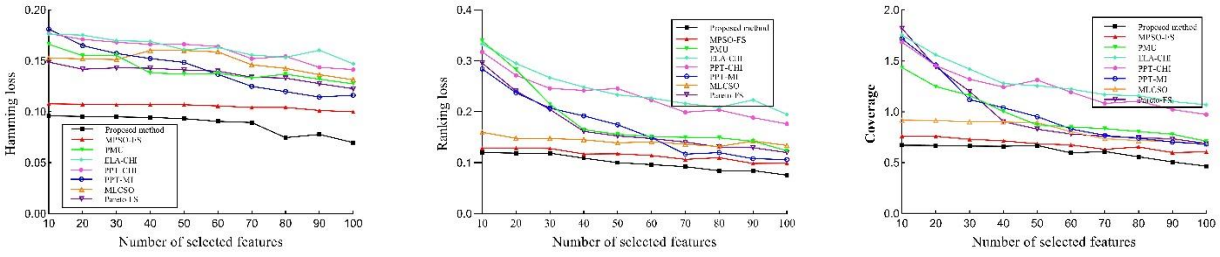
In Figures. 3 - 5, it is clear that the proposed method has the least Hemming loss in the Scene, Birds and Yeast data sets. In these data sets, the proposed method has achieved better results by increasing the number of selected features. It can conclude that the power of predictive features has increased significantly. The coverage criterion in data sets of Birds, and Yeast has the lowest values. The criteria in the Scene, Birds and Yeast data sets are more descending, which means that our proposed method significantly impacts selecting desirable features. Figures 3 - 5 demonstrate that the proposed method produces better property subsets than the other methods for all measurement criteria. In order to measure the effect of local search on the performance of the proposed method, we compare the results obtained by the proposed method using local search and without local search for the accuracy criterion on the *Scene*, *Birds*, and *Image* datasets and the obtained results are reported in Figure 6. It can be seen from the results that the proposed local search strategy results in improving the performance.

Table 3. The results of the experiment are comparing the proposed method with other methods on 100 independent runs on five multi-label datasets.

Metric	Data set	Algorithm								
		MPSOFS	MLCSO	PPT-CHI	PPT- MI	ELA- CHI	PMU	Pareto-FS	Proposed method	Wilcoxon
HL ↓	Emotions	0.177393	0.211221	0.375801	0.313816	0.307125	0.330747	0.300018	**0.162244**	+
	Birds	0.04367	0.093042	0.109038	0.190302	0.145968	0.101452	0.124785	**0.041382**	+
	Yeast	0.193722	0.214052	0.355700	0.415478	0.378954	0.344785	0.504541	**0.191715**	+
	Scene	0.087375	0.13364	0.261158	0.336425	0.254487	0.350564	0.358810	**0.078027**	+
	Image	0.780230	0.87112	0.923874	0.905892	0.93658	0.914875	0.903458	**0.71485**	+
RL ↓	Emotions	0.148927	0.164411	0.721681	0.692034	0.69350	0.575211	0.69297	**0.122035**	+
	Birds	0.211628	0.222522	0.21857	0.26698	0.26361	0.24897	0.287933	**0.194046**	+
	Yeast	0.169225	0.181467	0.70684	0.7312	0.673214	0.70245	0.76045	**0.164586**	+
	Scene	**0.094273**	0.123244	0.5602	0.61245	0.56024	0.53423	0.74125	0.121480	+
	Image	0.20071	0.25094	0.35564	0.32817	0.28305	0.35978	0.21648	**0.18462**	+
OE ↓	Emotions	0.232673	0.287129	0.269852	0.325547	0.326854	0.299421	0.355164	**0.222475**	+
	Birds	**0.517442**	0.593023	0.6014	0.589005	0.61342	0.603845	0.57548	0.564651	+
	Yeast	0.248637	0.245365	0.83710	0.682711	0.812034	0.9037	0.3553	**0.23627**	+
	Scene	**0.231605**	0.31689	1.74254	1.39449	1.82461	1.92906	0.86785	0.261672	+
	Image	0.30268	0.38025	0.44587	0.39847	0.447825	0.40758	0.37845	**0.28547**	+
CO ↓	Emotions	**1.816832**	1.945545	2.266198	2.283012	2.281654	2.320789	2.280235	1.841287	+
	Birds	2.557276	2.662539	2.75142	2.94756	2.66578	2.5357	2.562456	**2.512712**	+
	Yeast	**6.348964**	6.589967	7.230945	7.219387	7.227398	7.23247	7.196205	6.644842	+
	Scene	**0.58194**	0.714883	1.17624	1.169502	1.177431	1.178477	1.14350	0.62505	+
	Image	1.01244	1.130214	1.25647	1.16434	1.23084	1.18556	1.13387	**0.90125**	+
AP ↑	Emotions	0.820215	0.796205	0.39268	0.440422	0.43084	0.57412	0.421978	**0.84942**	+
	Birds	0.534813	0.487686	0.52465	0.51426	0.51022	0.5352	0.547029	**0.55527**	+
	Yeast	0.756879	0.744186	0.255597	0.242264	0.253145	0.255897	0.236478	**0.769049**	+
	Scene	0.82347	0.803491	0.658241	0.56854	0.66458	0.67895	0.42998	**0.825125**	+
	Image	0.7941	0.7629	0.7124	0.7355	0.7115	0.7334	0.7516	**0.8107**	+

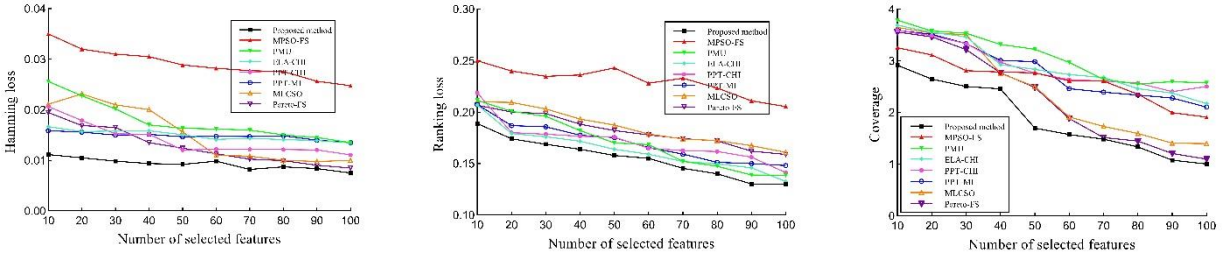
Table 4. The performance of the multi-label feature selection methods in terms of the time.

↓	Proposed method	MPSO-FS	PMU	PPT-CHI	PPT-MI	ELA-CHI	MLCSO	Pareto-FS
Image	8298.4156	8005.8712	54.84652	23.5412	1.98548	**0.09451**	36.54789	4.18974
Yeast	376.2487	310.5874	51.24587	20.1832	**0.25845**	15.3214	11.71061	3.93288
Scene	902.1248	839.2631	162.0287	44.1395	**0.085795**	27.0984	184.2542	6.95182
Birds	1910.217	1182.206	109.2157	3.24857	**0.394874**	1.89475	76.25485	0.45879
Emotions	81.16734	73.89132	10.182243	1.61996	**0.043728**	0.791846	11.77039	0.183892



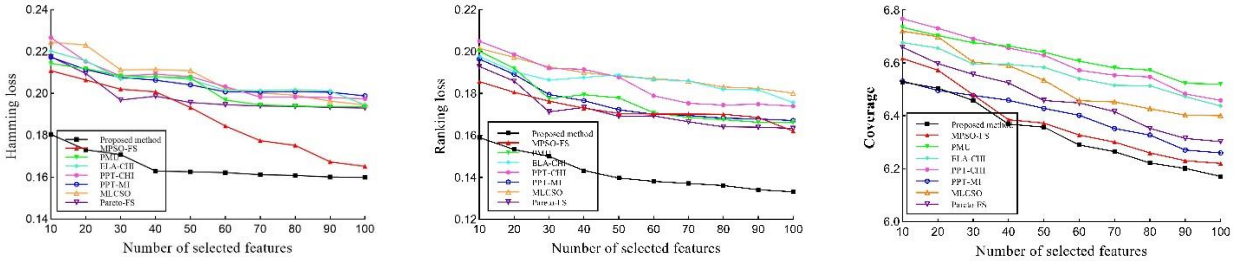
(a) (b) (c)

Fig. 3. Experimental results on Scene data set. (a) Hamming loss (b) Ranking loss (c) Coverage.



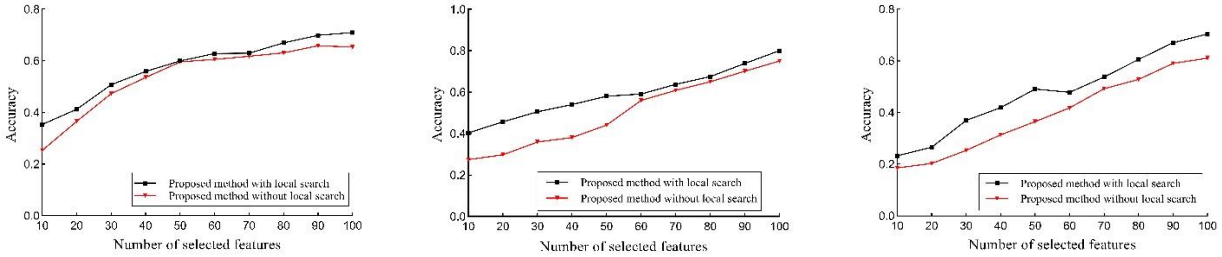
(a) (b) (c)

Fig. 4. Experimental results on Birds data set. (a) Hamming loss (b) Ranking loss (c) Coverage.



(a) (b) (c)

Fig. 5. Experimental results on Yeast data set. (a) Hamming loss (b) Ranking loss (c) Coverage.



(a) (b) (c)

Fig. 6. Evaluating the effects of using the local search strategy on the performance of the proposed feature selection method over (a) Scene (b) Birds, and (c) Image datasets

5. Conclusion

This paper proposes a multi-label feature selection method based on swarm intelligence method integrated with a novel local search strategy. The proposed method first estimates the subset size of features. In generally, the proposed method aims to improve the efficiency of the particle swarm optimization for large-scale problems in the feature selection task. To integrate the local search strategy with the search process of the PSO, a new operator is introduced. The work of the local search operator in the solution space is to increase the convergence speed and prevent falls at the local optimal.To apply the local search operator, the features were classified into two categories. At each step, the local search operator tries to add the features associated with each member of the population and remove the non - related characteristics. The interactive information-based theory is used to recognize the related and unrelated features. Multi-criteria assessment metrics have been used to evaluate the proposed approach. The search algorithm has a random feature; statistical tests such as the t-test and the Wilcoxon test have been used to prove the validity of the results. This work uses different evaluation criteria and five datasets to compare the proposed method with other methods. The results showed that the proposed method has a better performance in most samples than other methods.

There are several idea as future directions to extend the proposed method. A possible idea is using clustering algorithms to classify features into different categories instead of using two categories. This idea leads to further increasing the convergence speed and accuracy of the algorithm since each cluster tries to select a relevant feature and avoids the selection of redundant features. Another idea is the extension of the proposed method for streaming feature selection. In these types of problems, the features are not static, and at each step, some new features are appeared.

References

[1] Y. Lin, Q. Hu, J. Liu, J. Chen, and J. Duan, "Multi-label feature selection based on neighborhood mutual information," Applied Soft Computing, vol. 38, pp. 244-256, 2016.

[2] O. Reyes, C. Morell, and S. Ventura, "Scalable extensions of the ReliefF algorithm for weighting and selecting features on the multi-label learning context," Neurocomputing, vol. 161, pp. 168-182, 2015.

[3] L. Li, H. Liu, Z. Ma, Y. Mo, Z. Duan, J. Zhou, et al., "Multi-label feature selection via information gain," in International Conference on Advanced Data Mining and Applications, 2014, pp. 345-355.

[4] Y. Lin, Q. Hu, J. Liu, and J. Duan, "Multi-label feature selection based on max-dependency and min-redundancy," Neurocomputing, vol. 168, pp. 92-103, 2015.

[5] S. Tabakhi and P. Moradi, "Relevance–redundancy feature selection based on ant colony optimization," Pattern recognition, vol. 48, pp. 2798-2811, 2015.

[6] P. Moradi and M. Rostami, "Integration of graph clustering with ant colony optimization for feature selection," Knowledge-Based Systems, vol. 84, pp. 144-161, 2015.

[7] مریم رحمانی نیا, پرهام مرادی و م. جلیلی، "یک راهکار انتخاب ویژگی چندهدفه بر اساس اطلاعات متقابل شرطی و نظریه مجموعه پارتو،" مجله مهندسی برق دانشگاه تبریز، دوره ۵۰، شماره ۳، صفحات ۱۲۲۵ تا ۱۲۳۷، پاییز ۱۳۹۹.

[8] J. Lee and D.-W. Kim, "Memetic feature selection algorithm for multi-label classification," Information Sciences, vol. 293, pp. 80-96, 2015.

[9]Y. Yu and Y. Wang, "Feature selection for multi-label learning using mutual information and GA," in International Conference on Rough Sets and Knowledge Technology, 2014, pp. 454-463.

[10] Y. Zhang, D.-w. Gong, X.-y. Sun, and Y.-n. Guo, "A PSO-based multi-objective multi-label feature selection method in classification," Scientific reports, vol. 7, p. 376, 2017.

[11] M.-L. Zhang, J. M. Peña, and V. Robles, "Feature selection for multi-label naive Bayes classification," Information Sciences, vol. 179, pp. 3218-3229, 2009.

[12] M. A. Khan, A. Ekbal, E. L. Mencía, and J. Fürnkranz, "Multi-objective optimisation-based feature selection for multi-label classification," in International Conference on Applications of Natural Language to Information Systems, 2017, pp. 38-41.

[13] M. You, J. Liu, G.-Z. Li, and Y. Chen, "Embedded feature selection for multi-label classification of music emotions," International Journal of Computational Intelligence Systems, vol. 5, pp. 668-678, 2012.

[14] P. Zhu, Q. Xu, Q. Hu, C. Zhang, and H. Zhao, "Multi-label feature selection with missing labels," Pattern Recognition, vol. 74, pp. 488-502, 2018.

[15] شیما کاشف و حسین نظام آبادی پور، "یک روش ترکیبی برای یافتن زیرمجموعه ویژگی موثر در داده های چند برچسبی،"مجله مهندسی برق دانشگاه تبریز، دوره ۴۸، شماره ۳، صفحات ۱۳۲۷ تا ۱۳۳۸، پاییز ۱۳۹۷.

[16] I. A. Gheyas and L. S. Smith, "Feature subset selection in large dimensionality domains," Pattern recognition, vol. 43, pp. 5-13, 2010.

[17] Y. Saeys, I. Inza, and P. Larrañaga, "A review of feature selection techniques in bioinformatics," bioinformatics, vol. 23, pp. 2507-2517, 2007.

[18] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," IEEE Transactions on Knowledge & Data Engineering, pp. 491-502, 2005.

[19] S. Tabakhi, A. Najafi, R. Ranjbar, and P. Moradi, "Gene selection for microarray data classification using a novel ant colony optimization," Neurocomputing, vol. 168, pp. 1024-1036, 2015.

[20] R. K. Sivagaminathan and S. Ramakrishnan, "A hybrid approach for feature subset selection using neural networks and ant colony optimization," Expert systems with applications, vol. 33, pp. 49-60, 2007.

[21] M. H. Aghdam, N. Ghasem-Aghaee, and M. E. Basiri, "Text feature selection using ant colony

optimization," Expert systems with applications, vol. 36, pp. 6843-6853, 2009.

[22] J. Yang and V. Honavar, "Feature subset selection using a genetic algorithm," in Feature extraction, construction and selection, ed: Springer, 1998, pp. 117-136.

[23] M. Rostami and P. Moradi, "A clustering based genetic algorithm for feature selection," in 2014 6th Conference on Information and Knowledge Technology (IKT), 2014, pp. 112-116.

[24] T. M. Hamdani, J.-M. Won, A. M. Alimi, and F. Karray, "Hierarchical genetic algorithm with new evaluation function and bi-coded representation for the selection of features considering their confidence rate," Applied Soft Computing, vol. 11, pp. 2501-2509, 2011.

[25] S.-W. Lin, T.-Y. Tseng, S.-Y. Chou, and S.-C. Chen, "A simulated-annealing-based approach for simultaneous parameter optimization and feature selection of back-propagation networks," Expert Systems with Applications, vol. 34, pp. 1491-1499, 2008.

[26] S.-W. Lin, Z.-J. Lee, S.-C. Chen, and T.-Y. Tseng, "Parameter determination of support vector machine and feature selection using simulated annealing approach," Applied soft computing, vol. 8, pp. 1505-1512, 2008.

[27] L.-Y. Chuang, S.-W. Tsai, and C.-H. Yang, "Improved binary particle swarm optimization using catfish effect for feature selection," Expert Systems with Applications, vol. 38, pp. 12699-12707, 2011.

[28] Y. Liu, G. Wang, H. Chen, H. Dong, X. Zhu, and S. Wang, "An improved particle swarm optimization for feature selection," Journal of Bionic Engineering, vol. 8, pp. 191-200, 2011.

[29] B. Xue, M. Zhang, and W. N. Browne, "Particle swarm optimisation for feature selection in classification: Novel initialisation and updating mechanisms," Applied soft computing, vol. 18, pp. 261-276, 2014.

[30] H. M. Abdelsalam and A. M. Mohamed, "Optimal sequencing of design projects' activities using discrete particle swarm optimisation," International Journal of Bio-Inspired Computation, vol. 4, pp. 100-110, 2012.

[31] سمیرا حیدری مقدم بجستانی, سعید شعرباف تبریزی و ع. قاضی خانی، "ارائه ی یک روش انتخاب ویژگی جدید مبتنی بر بهینه سازی ازدحام ذرات با استفاده از به روزرسانی فازی،" مجله مهندسی برق دانشگاه تبریز، دوره ۵۰، شماره ۴، صفحات۱۵۵۷ تا ۱۵۶۷، زمستان ۱۳۹۹.

[32] J. Lee and D.-W. Kim, "Mutual information-based multi-label feature selection using interaction information," Expert Systems with Applications, vol. 42, pp. 2013-2025, 2015.

[33] W. Chen, J. Yan, B. Zhang, Z. Chen, and Q. Yang, "Document transformation for multi-label feature selection in text categorization," in Seventh IEEE International Conference on Data Mining (ICDM 2007), 2007, pp. 451-456.

[34] N. SpolaôR, E. A. Cherman, M. C. Monard, and H. D. Lee, "A comparison of multi-label feature selection methods using the problem transformation approach," Electronic Notes in Theoretical Computer Science, vol. 292, pp. 135-151, 2013.

[35] G. Doquire and M. Verleysen, "Feature selection for multi-label classification problems," in International work-conference on artificial neural networks, 2011, pp. 9-16.

[36] J. Read, B. Pfahringer, and G. Holmes, "Multi-label classification using ensembles of pruned sets," in 8th IEEE international conference on data mining, 2008, pp. 995-1000.

[37] . Doquire and M. Verleysen, "Mutual information-based feature selection for multilabel classification," Neurocomputing, vol. 122, pp. 148-155, 2013.

[38] J. Lee and D.-W. Kim, "Fast multi-label feature selection based on information-theoretic feature ranking," Pattern Recognition, vol. 48, pp. 2761-2771, 2015.

[39] J. Yin, T. Tao, and J. Xu, "A multi-label feature selection algorithm based on multi-objective optimization," in 2015 International Joint Conference on Neural Networks (IJCNN), 2015, pp. 1-7.

[40] J. Kennedy and R. Eberhart, "Particle swarm optimization," in Proceedings of ICNN'95-international conference on neural networks, 1995, pp. 1942-1948.

[41] R. B. Pereira, A. P. d. Carvalho, B. Zadrozny, and L. H. d. C. Merschmann, "Information gain feature selection for multi-label classification," 2015.

[42] M. M. Kabir, M. Shahjahan, and K. Murase, "A new local search based hybrid genetic algorithm for feature selection," Neurocomputing, vol. 74, pp. 2914-2928, 2011.

[43] D. P. Muni, N. R. Pal, and J. Das, "Genetic programming for simultaneous feature selection and classifier design," 2006.

[44] M. M. Kabir, M. M. Islam, and K. Murase, "A new wrapper feature selection approach using neural network," Neurocomputing, vol. 73, pp. 3273-3283, 2010.

[45] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: an open architecture for collaborative filtering of netnews," in Proceedings of the 1994 ACM conference on Computer supported cooperative work, 1994, pp. 175-186.

[46] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in Advances in neural information processing systems, 2006, pp. 507-514.

[47] M.-L. Zhang and Z.-H. Zhou, "ML-KNN: A lazy learning approach to multi-label learning," Pattern recognition, vol. 40, pp. 2038-2048, 2007.

[48] H. O. Parametric, "Handbook Of Parametric And Nonparametric Statistical Procedures."

[49] S. Kashef and H. Nezamabadi-pour, "A label-specific multi-label feature selection algorithm based on the Pareto dominance concept," Pattern Recognition, vol. 88, pp. 654-667, 2019.

[50] H. Bayati, M. B. Dowlatshahi, and M. Paniri, "Multi-label feature selection based on competitive

swarm optimization," Journal of Soft Computing and Information Technology, vol. 9, pp. 56-69, 2020.

[51] J. Lee and D.-W. Kim, "Feature selection for multi-label classification using multivariate mutual information," Pattern Recognition Letters, vol. 34, pp. 349-357, 2013.