International Journal of Electrical and Computer Engineering (IJECE)
Vol. 13, No. 3, June 2023, pp. 3384∼3397
ISSN: 2088-8708, DOI: 10.11591/ijece.v13i3.pp3384-3397

Multi-label learning by extended multi-tier stacked ensemble method with label correlated feature subset augmentation

Hemavati[1], Visweswariah Susheela Devi[2], Ramalingappa Aparna[1]
[1]Department of Information Science and Engineering, Siddaganga Institute of Technology, Tumakuru, India
[2]Department of Computer Science and Automation, Indian Institute of Science, Bengaluru, India

Article Info

Article history:

Received Apr 16, 2022
Revised Sep 23, 2022
Accepted Oct 1, 2022

Keywords:

Classification
Correlation
Ensemble learning
Stacking

ABSTRACT

Classification is one of the basic and most important operations that can be used in data science and machine learning applications. Multi-label classification is an extension of the multi-class problem where a set of class labels are associated with a particular instance at a time. In a multiclass problem, a single class label is associated with an instance at a time. However, there are many different stacked ensemble methods that have been proposed and because of the complexity associated with the multi-label problems, there is still a lot of scope for improving the prediction accuracy. In this paper, we are proposing the novel extended multi-tier stacked ensemble (EMSTE) method with label correlation by feature subset selection technique and then augmenting those feature subsets while constructing the intermediate dataset for improving the prediction accuracy in the generalization phase of the stacking. The performance effect of the proposed method has been compared with existing methods and showed that our proposed method outperforms the other methods.

Corresponding Author:

Hemavati
Department of Information Science and Engineering, Siddaganga Institute of Technology
Tumakuru, India
Email: hemavd@sit.ac.in

1. INTRODUCTION

Classification is one of the basic and most important operations that can be used in data science and machine learning applications. Multi-label classifications are found in the areas of text categorization [1], medical applications [2], movie genres classification [3], protein classification [4] and so on. The classification of multi-label data is entirely different from the binary and multi-class problems, since the classification of multi-label data may get affected by the inherently hidden label correlation [5]. There are a substantial number of proposed algorithms for dealing with these kinds of problems. In [6], these are categorized into two different approaches. They are problem transformation and algorithm adaptation approach. problem transformation (PT)approach, data is adapted according to the algorithm and the problem is decomposed into multiple binary class or multi-class problems. The different transformation methods include: binary relevance (BR), ranking by pairwise comparison (RPC), label powerset (LP) method, and calibrated label ranking (CLR).

BR [7]–[9]: This method converts the original problem of multi-label into m number of binary problems by containing all examples of the original dataset D_j, $j = 1...m$, and learning happens with each label. The final output will be the union of all labels predicted by m classifiers. The main drawback of the BR method

is that it ignores label correlation between the labels. LP [10] method considers the combination of unique label sets present in the original training dataset as a single class label in the transformed dataset. The output for the new instance will be one of these unique label sets with the highest probability or label ranking. LP suffers from the problem of computation complexity, $(n, 2^L)$ where n indicates the total number of data points and L is the number of labels. Another problem is that it can only predict the unique label combinations which are present in the training dataset, which may lead to a problem of class imbalance. The pruned PT method [10] provides the solution for the second problem indicated above by removing the label sets which are occurring rarer than a user-defined threshold, and often changing by replacing them with decomposed subsets of these corresponding set of labels which are present whose value is greater than the peak value in a dataset. Ranking by pairwise comparison (RPC) [11] transforms the dataset of multi-label into $L(L-1)/2$ binary label datasets where L indicates the total number of labels, one for each pair of labels. Each dataset keeps as it is the data points from the original dataset, that may be associated with at least one of the corresponding labels but not both. However, ranking keeps a corresponding order of the labels. In [12], the argument is that such kind of ranking may not hold a usual "zero-point" and hence, does not give any kind of knowledge about the preference or the absolute choice, which can distinguish among all other alternative options. Later, by proposing an extension of RPC, that is, calibrated label ranking (CLR), with an extra label to the original or the base label set, that can be considered as a "neutral breaking point" between related and unrelated class labels. Algorithm adaptation approach adapts the base algorithms to work on multi-label datasets without using any transformation method. Some of the algorithms like AdaBoost.MH [13], MLkNN [14], and random k-labelsets (RAkEL) [15] can be used directly on multi-label data.

2. LITERATURE REVIEW

This section gives the background of different existing methods for multi-label classification. The review of existing methods helps to improve the new methods and the usage of the methodologies provides more clarity on the novelty of new methods. This section also gives a comparative study of existing methods.

2.1. Multi-label learning with label correlation and feature selection method

Exploiting the label correlation is one of the research issues to be considered for improving the multi-label classification performance. In the literature, we can find many related approaches that have been proposed in the past decades for exploiting label correlation and feature selection. There are three categories of label correlation strategies: one is the first-order strategy that ignores the label correlation. One can understand how this strategy is achieved from the BR example considered. But, there are advantages with the first-order strategy concerning implementation simplicity and efficiency, but it will not consider label correlation information. Another strategy is the second-order strategy. Here, the pairwise correlation is considered between the labels. For example, in LR proposed in [16] and an extended LR proposed in [17], a virtual label is added to each instance i.e., CLR. In [18], a conventional support vector machine (SVM) is modified based on ranking loss by considering a new loss term which is called RankSVM. In the work carried out by Zhi-Fen He *et al.* [19], label correlations with missing labels are considered. The last one is the high order strategy. Here, among the labels, the high order correlations will be exploited. For instance, the label powerset (LP) [20] considers the combination of unique label sets present in the original training dataset as a single class label in the transformed dataset. The output for the new instance will be one of these unique label sets with the highest probability or label ranking. The PPT method overcomes the problem of LP by the selection of a small random subset of labels, then trains an ensemble of classifiers by using the LP method. But, it only addresses the implicit correlation among labels which considers the weight vector by learning each classifier depending on the weight vectors of other classifiers, the matrix of correlated labels, and the parameters of the model. As we know, multi-label learning has many problems concerning label correlation, and it may not fit well for all real-time applications where the overall importance of label distribution matters. Geng [21] have proposed label distribution learning (LDL) for such problems. The LDL covers some of the labels, which will represent the degree of the instance. LDL could improve the learning process in case of utilization of the label correlation concerning facial depression recognition as contributed by Zhou *et al.* [22], facial age estimation by Wen *et al.* [23] and many other applications.

2.2. Ensemble learning

Ensemble methods are useful when it requires to take a decision based on collective opinion of different sources. The result of the base classifier can be used in constructing the next-level classifier which are considered as dependent ensemble methods. The AdaBoost algorithm is useful for improving the performance of various machine learning algorithms or weak learners. All data points in the training dataset are weighted. The initial weight is set to: $W(X_i) = 1/n$ where X_i is an example of i^{th} and n indicates the total number of data points with multiple labels in the training dataset. Methods of meta-learning are particularly useful for those which are categorized by always correctly classifying or incorrectly classifying certain conditions. Some of the methods for measuring are distribution weight, majority vote, Bayesian combination, performance weight, and entropy weight. In our paper, we use a meta-learning method called Stacking. This method is used to accomplish greater accuracy with maximal generalization. It is often used to integrate models created by distinct classifiers to form a meta dataset. That is, rather than using input features from the original training dataset, the input features will be the predicted results of its base classifiers and class labels will be considered as such in the original training dataset. Later, the meta classifier will produce the final prediction from the meta dataset. The weighted classifier selection has been proposed by Xia *et al.* [24], with sparsity regularization for ensemble member construction for multi-label classification.

2.3. Research targets

The existing algorithms which are mentioned above are developed to find out the solution for the multi-label classification problems. But still, there are challenges related to label correlation, the curse of dimensionality, and label imbalance. There exist many ensemble techniques which improve the performance of learning through different combination approaches. These combination approaches are mainly used to concentrate on the model overfitting and the sensitivity of initialization. Our contributions are summarized as: i) we propose a novel extended multi-tier stacked ensemble method for multi-label classification, which is useful in constructing the strong meta-learning phase by considering different combination schemes for model setting; ii) we also consider the label correlation feature subset augmentation for improving the classification performance; and iii) to the best of our knowledge, this approach works significantly more fine than the existing systems.

The experiments have been carried out on 10 different multi-label datasets of different domains. The rest of the paper is organized as. In section 2, we describe the proposed method, in section 3 we present the mathematical model, in section 4 we will be discussing evaluation metrics and in section 5 we discuss experiments, followed by the conclusion.

3. PROPOSED METHOD

The main challenge with the stacked ensemble method is that the variance of the base tier may be inherited to level1, i.e., meta learner. One more problem associated with the multi-label data is the ignorance of label correlation information when applied with the binary relevance method. In this paper, we are considering the label correlated feature subset for augmenting with the ensemble tier. In the literature, many different ways have been discussed for stacking and label correlation methods and those approaches have shown improved results for the performance metrics, but still, there is a lot of scope for improving the performance by considering the label correlation with multi-tier stacked ensemble method. Figure 1 shows the extended multi-tier stack method. The base level uses predefined base classifiers for the given problem. The combination schemes are then used to get the combined prediction. So, the bias and variance can be controlled in these two tiers. The novelty we are introducing is that we are augmenting the meta dataset constructed in the generalization tier with the label correlated feature subset. Using the label correlation helps to improve the classification accuracy.

4. MATHEMATICAL MODEL

In this section, we discuss the mathematical model, which gives the process of creating a real-world representation of the proposed system required in this paper. The representation of the proposed model in a mathematical model helps to transform the description into an accurate presentation of the proposed model. Table 1 gives the description of notations used in this paper.

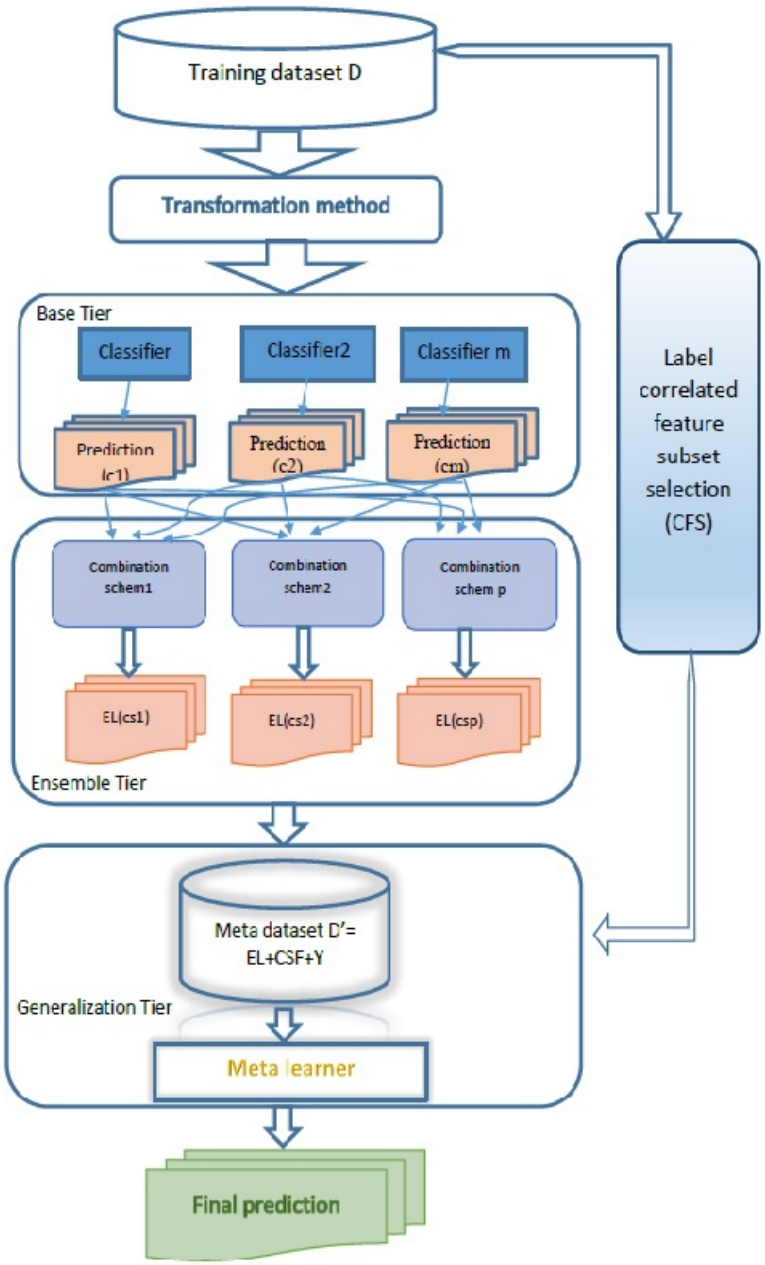

Figure 1. Architecture of multi-tier stacked ensemble with correlated feature subset augmentation method for multi-label data

4.1. Base-tier

Let $\mathfrak{D} = \{x, y\}$ where $x \in X$ and $y \subseteq Y$, $X = \{x_1, x_2, ..., x_n\}$ and $Y = \{y_1, y_2, ..., y_L\}$ and each $y_j = \{0, 1\}$ and $j = 1, 2, ..., L$. Then, train the classifier with $\hbar(X)$ to predict the label subset Y', that is, $Y' \leftarrow \hbar(X)$ where $Y' \in \{0, 1\}^{\mathrm{L}}$. Let $\mathcal{H} = \{\hbar_1, \hbar_2 ..., \hbar_m\}$ be set of base classifiers and each base classifier predicts the set of labels which can be mapped on the label set Y.

$$Y' = \hbar_k(X) \tag{1}$$

where $\hbar_k \in \mathcal{H}$ and $k = \{1, 2, 3, ..., m\}$ let $\mathcal{B}_L = \{\mathcal{B}_{L1}, \mathcal{B}_{L2}, ..., \mathcal{B}_{Lm}\}$ be the set of label sets predicted by each base classifier.

$$\mathcal{B}_{Lk} = Y' \leftarrow \hbar_k(X) \tag{2}$$

Table 1. Notations

Symbols	Descriptions
$\mathfrak{D}$	Training dataset
X	Feature set of training dataset
Y	Label set of training dataset
n	Number of features
x_i	i^{th} instance
y_j	j^{th} label
$h_k(X)$	k^{th} base classifier
Y'	Predicted label set
$\mathcal{H}$	Base classifier set
m	Number of classifiers
$\mathcal{B}_{Lk}$	Set of Predicted label set
$\mathcal{W}_{Lk}$	Set of combination methods
S_i	Semantic label set
F_k	Features of semantic label set
$(F_k; S_i)$	Relatedness among feature of candidate set
$E(F_k)$	Function for evaluation Mx(F$_k$, L)

4.2. Ensemble tier

Let $\mathcal{W}_L = \{\mathcal{W}_{L1}, \mathcal{W}_{L2}, ..., \mathcal{W}_{LN}\}$ be the set of combination methods for combining the labels predicted by the base classifiers to map them with the proper subset of the label set. Consider the following (3):

$$\mathcal{W}_{Lk} : \mathcal{B}_{Lk} \rightarrow Y' \tag{3}$$

and $Y' \in \{0, 1\}^{\mathrm{L}}$

$$\mathcal{W}_{Lk} = Aggr(\hbar_k(X)) \tag{4}$$

where $k = \{1, 2, 3, ..., m\}$

4.3. Generalization tier

In this step, we are selecting maximum label correlated feature subset for augmenting with ensemble dataset. There are many feature selection methods for multi-label dataset such as D2F, SCLS and MDMR methods. These feature selection methods use low order similarities to figure out candidate features [25]–[27]. In this paper, we are making use of considering max-correlation (MCMFS) proposed by Zhang *et al.* [28] algorithm, which uses max correlation between the labels. In this method, the label set is divided into m semantic groups $L' = \{s_1, s_2, ..., s_m\}$, where each $s_i = \{l_{i1}, l_{i2}, ..., l_{iq}\} \in Y$ and $i = 1, 2...m$ and the following condition is satisfied, i.e., $s_1 \cup s_2 \cup .. \cup s_m = L$ and $s_i \cap s_j =$. Then the $(F_k; S_i)$ indicates relatedness among feature of candidate set and the label sets in each semantic group is defined as (5).

$$(F_k; S_i) = max(\mu(F_k; y_j)) \tag{5}$$

In (5) a larger value indicates the candidate feature which is of more relevance in the semantic group and a small value indicates a weak relevance in the semantic group. Then we can obtain the maximum correlation of d dimensional vector of feature F_k and the label set, L i.e.,

$$Cor(fk; L') = [(F_k; s_1), (F_k, s_2)...(F_k, s_m)] \tag{6}$$

then

$$\mathcal{M}x(F_k, L) = \max_{s_i \in L'}(F_k, s_i) \tag{7}$$

$$\mathcal{M}x(F_k, L) = \max_{s_i \in L'}\{\max_{l_j \in s_i} \mu(F_k, l_j)\} \tag{8}$$

$$\mathcal{M}x(F_k, L) = \max_{l_j \in L} \mu(F_k, l_j) \tag{9}$$

where $\mathcal{M}x(F_k, L)$ takes the maximum relevance of the corresponding features effectively with respect to label set, irrespective of any number of labels that are present in the semantic group. $\mathcal{M}x(F_k, L)$ can exactly make a selection of the critical features. Based on this, we can use MCMFS as feature subset selection algorithm for selecting feature subset. The function for evaluation is given as (10) and (11):

$$E(F_k) = \mathcal{M}x(F_k, L) - \frac{1}{|s|}\Sigma\mu(F_k; F_j) \tag{10}$$

$$E(F_k) = \max_{l_j \in L} \mu(F_k, l_j) - \frac{1}{|s|}\Sigma\mu(F_k; F_j) \tag{11}$$

after selecting the feature subset, F_s, it has to be augmented with the predicted result of the ensemble method.

$$D' = \mathcal{W}_L \cup F_s \cup Y \tag{12}$$

where $\mathcal{W}_L$ is the predicted result of the ensemble classifier, F_s is the label correlated feature subset and Y is the label set from the original training dataset. After constructing meta dataset, we apply the meta learner for final model training.

$$fin\_model \leftarrow \hbar(D') \tag{13}$$

Algorithm 1 uses the base classifiers for predicting the first level prediction. We use binary relevance for the problem transformation and then, for the base classification we have chosen naive Bayes, SVM, decision tree classifiers. Algorithm 2 depicts applying the combination schemes for the predicted result from the base classifier. Here we are considering bagging and boosting combination schemes for stacking. In Algorithm 3 we are using the MCMFS algorithm for feature subset selection in our model for selecting the critical features for augmenting in the generalization tier. Algorithm 4 depicts the generalization tier, where the meta dataset is constructed by combining the predicted result, feature subset, and the label set from the original training dataset. Once the meta dataset is constructed, the meta learner is applied for the final prediction.

Algorithm 1 Classification with base classifier by using transformation method

Input: $D = \{X_{\mathrm{i}}, Y_{\mathrm{i}}\}_{i=1}^{n}$, where $X_{\mathrm{i}} \in R^{\mathrm{n}}, Y_{\mathrm{i}} \subseteq Y$ (set of all labels)
Output: Stacked Ensemble classifier
begin
Step 1: Transform the multi-label data to binary class problem by applying binary relevance (BR) method.
    $D_{\mathrm{BR}} \leftarrow BR(D)$
Step 2: Apply the base classifiers on D_{BR}
    for i ← 1 to m
        Train the base classifier $\hbar_i(X)$ based on D_{BR}.
    end for
    $\mathcal{B}_L = \{\hbar_1(X), \hbar_2(X), ..., \hbar_m(X)\}$
Step 3: return $\mathcal{B}_L$
end

Algorithm 2 Applying the combination scheme for the predicted result from the base classifier

Input: The predicted result of the base classifiers $\mathcal{B}_L$
Output: new ensemble dataset $\mathcal{W}_L$
begin
Step 1: apply the combination schemes (CS) to produce ensemble prediction.
    for j ← 1 to p
      for i ← 1 to m
        $\mathcal{W}_{Li} = CS_j(\mathcal{B}_{Li})$
      end for
    end for
Step 2: return $Aggr(\mathcal{W}_L)$
end

Algorithm 3 Selecting the label correlated feature subset by making use of the algorithm MCMFS [28]

Input: Full feature set from the original training dataset $F = \{f_1, f_2...f_d\}$ and label set $L = \{l_1, l_2...l_m\}$
Output: Correlated feature subset F_s
begin
  $F_s \leftarrow Null$
  $p \leftarrow 0$
  for i ← 1 to d
    calculate $\mathcal{M}x(F_i, L)$ using eq(9)
  end for
  while $p < b$ **do**
    if $p == 0$ **then**
      select f_j with highest $\mathcal{M}x(F_i, L)$
      $p = p + 1$
      $F_s = F_s \cup \{f_j\}$
      $F = F - \{f_j\}$
    end if
    for each $f_i \in F$ **do**
      calculate $\mu(f_i; f_j)$
      by using equation (11) calculate $E(F_i)$
    end for
    with the highest $E(F_i)$, select the feature f_j
    $F_s = F_s \cup \{f_j\}$
    $F = F - \{f_j\}$
  end while
end

Algorithm 4 Constructing meta dataset and applying meta classifier for final prediction

Input: The predicted result of the ensemble classifiers $\mathcal{W}_L$ and the label correlated feature subset F_s and Y .
Output: Final prediction
begin
Step1: Construct the meta dataset by augmenting the F_s and Y with $\mathcal{W}_L$.
  $D' = \mathcal{W}_L \cup F_s \cup Y$
Step2: train the model with new meta dataset D' with meta learner
  $fin\_model \leftarrow \hbar(D')$
end

5. EVALUATION METRICS

In the multi-label classification method, the output is a set of labels, so the normal evaluation metrics used for analyzing single-label classification algorithms cannot be applied directly. The following list includes some of the commonly used evaluation metrics. The metrics used for evaluation differs according to the nature of the target problem [29].

$$Precision = \frac{1}{N}\sum_{i=1}^{N}\frac{|f(X_{\mathrm{i}}) \cap Y_{\mathrm{i}}|}{|f(X_{\mathrm{i}})|} \tag{14}$$

$$Recall = \frac{1}{N}\sum_{i=1}^{N}\frac{|f(X_{\mathrm{i}}) \cap Y_{\mathrm{i}}|}{|Y_{\mathrm{i}}|} \tag{15}$$

$$F1 - Measure = \frac{1}{N}\sum_{i=1}^{N}\frac{2 * |f(X_{\mathrm{i}}) \cap Y_{\mathrm{i}}|}{|Y_{\mathrm{i}}| + |f(X_{\mathrm{i}})|} \tag{16}$$

A higher the value indicates better performance of an algorithm.

$$Hamming\ Loss = \frac{1}{N}\sum_{i=1}^{N}\frac{|f(X_{\mathrm{i}}) \oplus Y_{\mathrm{i}}|}{|Y_{\mathrm{i}}| + |f(X_{\mathrm{i}})|} \tag{17}$$

The performance of the algorithm increases with the decrease in value of Hamming loss, i.e., zero indicates a perfect classification. The measures such as precision, recall, F1-measure, or any measure which can be used to evaluate binary classification can be computed on a per-label basis and used for multi-label evaluation after averaging over all possible labels. The difference between macro-averaging and micro-averaging is that macro-averaging gets affected by classes with fewer instances, but micro-averaging is affected by classes with numerous instances.

Macro F1:

$$F_1^l = 2\sum_{i=1}^{N}(Y_i^l * f(X_i)^l)/(\sum_{i=1}^{N} Y_i^l + \sum_{i=1}^{N} f(X_i)^l) \tag{18}$$

$$macroF1 = \sum_{l=1}^{h} F_1^l \tag{19}$$

Where Y_i^l is 1 if instance X_i originally has a label l. $f(X_i)$ is the predicted label of instance X_i, $f(X_i)^l$ is 1 if X_i is predicted to have label l, both Y_i^l and $f(X_i)^l$ will have a value 0 otherwise.

$$MicroF1 = \frac{\sum_{i=1}^{h}\sum_{j=1}^{N} Y_j^i f(X)_j^i}{\sum_{i=1}^{h}\sum_{j=1}^{N} Y_j^i + \sum_{i=1}^{h}\sum_{j=1}^{N} f(X)_j^i} \tag{20}$$

6. RESULTS AND DISCUSSION

6.1. Datasets

For finding out the performance of multi-tier stacked ensemble (EMSTE) we have selected the 10 most common data sets from Mulan and Meka Repository [30] of standard multi-label datasets and each column of Table 2 shows the characteristics of the data sets. Here we tried to cover all datasets from different domain. We have tested whether the proposed method works on all of these different domain datasets.

Table 2. Description of the datasets

Dataset Name	#instances	#Features	#Labels
yeast	2417	103	14
scene	2407	294	6
emotions	593	72	6
music	592	77	6
corel5k	5000	499	374
enron	1702	1001	53
bibtex	7395	1836	159
flags	194	26	12
birds	322	260	20
genbase	662	1186	27

6.2. Baseline methods

The proposed method will be compared with the following baseline ensemble multi-label classification methods which are related to different concepts like ensemble, stacking and multi-label.

a. Ensemble binary relevance (EBR) method [31]: it is an ensemble version of the binary relevance model, and it does not consider the label correlation.
b. Ensemble classifier chain (ECC) method [32]: an ensemble version of classifier chain. In this method, the global label correlation will be considered and the order of chain for each CC will be generated randomly.
c. Ensemble label powerset (EPS) [33] method: it is an improved version of the ensemble Label Power set method, which focuses on the most important correlation of labels by pruning rarely occurring label sets.
d. RAkEL: It is an improved version of the ensemble of label powerset based on a random small subset of k labels and the relationship between this small subset of k labels.
e. ADABOOST.MH: a weighted ensemble version based on BR that not only maintains a set of weights over the instances as AdaBoost does, but also over the labels.

f. MLS [34] it is a stacked multi-label ensemble version based on binary relevance. The global correlation between the labels will be considered at the meta level, but misses the local pairwise relationship between the labels.

g. MLkNN [35] it is a lazy learning multi-label version of kNN algorithm, which is based on statistical information taken from the label sets which are neighboring instances of unseen labels. ML-kNN makes use of maximum a posterior principle to find out the label set for the unseen instances.

6.3. Experimental setup

The experiment has been done by using python3 as well as MEKA [36] tool to verify the performance of our EMTSE scheme with the existing schemes, which provides the multi-label methods. We have tested the proposed system on 10 benchmark datasets of different domains for the effectiveness of the proposed system. In practice, the 10-fold cross-validation is applied for performance evaluation. The random division of all samples is done in 10 equal parts. Each part is held for testing and the remaining part of the data is merged with the training dataset. The validation has run ten times in an iterative manner. Then the average performance of the evaluation metrics is calculated for the given model. Binary relevance decision tree (BRDT) [37] is used for improved generalization capability of the model. It finds one decision tree for each label. For each label associated with label set Y, the binary naive Bayes classifier is learned. This is called binary relevance naive Bayes (BRNB) [38]. This method outputs a set of labels which is the union of different single-label naive bayes classifiers. In SVM kernel [39], [40] the multi-label problem is decomposed into multi-class problem. Then it tries to determine the optimal hyperplane based on training data, which is used to classify test data points or the new data points. In two dimensions, the hyperplane is a simple line. The combination schemes we selected for experimentation are bagging and boosting. The label correlated feature selection is a method used to select the top-ranked features for augmenting in the ensemble tier for improving the prediction performance of the meta learner. We have used MLkNN classifier in the generalization tier as a meta classifier for the final prediction since MLkNN uses the approach of maximum a-posteriori (MAP) combined with k-NN, and it is been proved that MLkNN performed far better than several other algorithms [41]. The experimental results have been recorded in Tables 3–7 with the corresponding rank value among the algorithms and ↑ indicates that the higher the value, better the performance and ↓ indicates lower the value better the performance.

Table 3. Average precision of different multi-label classification algorithms ↑

Dataset	EBR	ECC	EPS	RAkEL	Adaboost.MH	MLkNN	MLS	EMSTE
yeast	0.654(3)	0.562(8)	0.573(7)	0.580(6)	0.638(5)	0.701(2)	0.642(4)	0.772(1)
scene	0.635(8)	0.654(5)	0.729(3)	0.711(4)	0.651(7)	0.738(2)	0.671(6)	0.751(1)
emotions	0.735(4)	0.637(6)	0.571(8)	0.581(7)	0.746(3)	0.708(5)	0.732(2)	0.756(1)
music	0.755(2)	0.615(7)	0.595(8)	0.689(4)	0.623(6)	0.658(5)	0.715(3)	0.801(1)
corel5k	0.174(6)	0.167(7)	0.160(8)	0.279(3)	0.294(2)	0.228(5)	0.342(1)	0.251(4)
enron	0.163(8)	0.187(5)	0.170(6)	0.168(7)	0.191(4)	0.238(1)	0.237(2)	0.208(3)
bibtex	0.143(7)	0.114(8)	0.180(5)	0.175(6)	0.198(4)	0.338(1)	0.250(3)	0.251(2)
flags	0.622(6)	0.774(2)	0.542(8)	0.562(7)	0.701(5)	0.738(3)	0.721(4)	0.851(1)
birds	0.231(5)	0.324(4)	0.218(6)	0.475(2)	0.488(1)	0.210(7)	0.701(2)	0.422(3)
genbase	0.798(2)	0.789(4)	0.799(1)	0.727(7)	0.714(8)	0.738(6)	0.797(3)	0.780(5)
Average score	0.491(6)	0.4823(7)	0.4537(8)	0.4947(5)	0.5244(4)	0.5295(3)	0.5808(2)	0.5843(1)

6.4. Experimental results and analysis

Tables 3-7 show the average precision, F1score, MAcroF1, MicroF1, and Hamming loss respectively of the various methods. It can be seen in many of the cases our method (EMSTE) gives the best result. Among the other cases, mostly EMSTE is in second place. Even if EMSTE is not in second place, its value is close to the best value. We are using the Friedman test F_r [42] to analyze the performance of different algorithms based on the evaluation metrics. The Friedman test F_r and critical value with the significance level $\alpha = 0.05$ have been considered. The rejection or null hypothesis is for the equal performance of each metric. As long as we have two or more populations, and among them, some are not normal, we can make use of the non-parametric Friedman test considering it an omnibus test to find out the existence of any significant differences among the median values of the populations. We are using the post-hoc Nemenyi test [43] to retrieve which differences are significant. If the differences in the mean rank are greater than the critical distance CD=3.32 of the Nemenyi test, then we can say that differences among the populations are significant. The statistical analysis was

conducted for k=7 (No. of algorithms used for comparison), N=10 (no. of datasets) significance level of the tests alpha=0.050 with the family-wise. The algorithms which are connected within the CD line are not statistically significantly better than each other. The results of these tests are shown in Figure 2 as critical distance diagrams. In the diagram, the top-ranked algorithms are on the left side of the CD diagram and the algorithms which are on the right side of the CD line are low ranked algorithms. For example, in Figure 2(a) EMSTE, ECC, EBR, and MLkNN are left side of the CD line and are considered to be the top-ranked algorithms, and RAkEL, EPS, and ADABOOST fall on the right side the of the CD line and are considered to be the low ranking algorithms for the Average Precision performance metrics. Similarly, for the other performance metrics of the CD diagrams which are shown as Figure 2(b) F1, 2(c) macro F1, 2(d) Micro F1, and 2(e) Hamming loss, holds the same.

Table 4. F1-score of different multi-label classification algorithms ↑

Dataset	EBR	ECC	EPS	RAkEL	Adaboost.MH	MLkNN	MLS	EMSTE
yeast	0.385(6)	0.398(5)	0.374(8)	0.383(7)	0.405(3)	0.401(4)	0.455(2)	0.457(1)
scene	0.712(5)	0.728(3)	0.707(6)	0.634(7)	0.051(8)	0.734(2)	0.718(4)	0.752(1)
emotions	0.635(5)	0.637(4)	0.631(6)	0.581(7)	0.046(8)	0.667(1)	0.521(3)	0.651(2)
music	0.705(2)	0.615(7)	0.695(4)	0.689(5)	0.455(8)	0.654(6)	0.702(3)	0.750(1)
corel5k	0.274(4)	0.267(5)	0.260(6)	0.279(3)	0.294(2)	0.112(8)	0.295(1)	0.250(7)
enron	0.219(4)	0.225(2)	0.187(7)	0.168(8)	0.214(5)	0.212(6)	0.220(3)	0.235(1)
bibtex	0.143(7)	0.114(8)	0.180(4)	0.175(5)	0.198(3)	0.210(2)	0.174(6)	0.320(1)
flags	0.654(4)	0.674(2)	0.582(7)	0.652(5)	0.562(8)	0.638(6)	0.667(3)	0.750(1)
birds	0.301(4)	0.324(3)	0.218(7)	0.275(5)	0.388(2)	0.059(8)	0.274(6)	0.451(1)
genbase	0.551(2)	0.498(5)	0.540(3)	0.427(7)	0.414(8)	0.438(6)	0.501(4)	0.582(1)
Average score	0.4579(2)	0.448(4)	0.4374(5)	0.4263(6)	0.2977 (8)	0.4125(7)	0.4527(3)	0.5198(1)

Table 5. Macro F1-score of different multi-label classification algorithms ↑

Dataset	EBR	ECC	EPS	RAkEL	Adaboost.MH	MLkNN	MLS	EMSTE
yeast	0.385(6)	0.398(5)	0.374(8)	0.383(7)	0.405(3)	0.401(4)	0.431(2)	0.455(1)
scene	0.712(4)	0.728(3)	0.707(5)	0.634(6)	0.001(8)	0.734(2)	0.621(7)	0.754(1)
emotions	0.635(5)	0.637(4)	0.631(6)	0.581(7)	0.046(8)	0.667 (1)	0.651(3)	0.652(2)
music	0.705(2)	0.615(6)	0.695(3)	0.689(4)	0.455(8)	0.654(7)	0.686(5)	0.751(1)
corel5k	0.274(3)	0.267(4)	0.260(5)	0.279(2)	0.294(1)	0.112(7)	0.401(3)	0.251(6)
enron	0.219(4)	0.225(2)	0.187(7)	0.168(8)	0.214(5)	0.212(6)	0.222(3)	0.245(1)
bibtex	0.143(7)	0.114(8)	0.180(5)	0.175(6)	0.198(4)	0.210(3)	0.311(2)	0.321(1)
flags	0.652(3)	0.674(2)	0.582(7)	0.650(5)	0.562(8)	0.638(6)	0.655(4)	0.751(1)
birds	0.301(4)	0.388(1)	0.218(6)	0.275(5)	0.358(3)	0.059(7)	0.401(3)	0.361(2)
genbase	0.551(3)	0.498(5)	0.540(4)	0.427(7)	0.414(8)	0.438(6)	0.570(2)	0.581(1)
Average score	0.4577(3)	0.4544(4)	0.4374(5)	0.4261(6)	0.2947(8)	0.4125(7)	0.4949(2)	0.5122(1)

Table 6. Micro F1-score of different multi-label classification algorithms ↑

Dataset	EBR	ECC	EPS	RAkEL	Adaboost.MH	MLkNN	MLS	EMSTE
yeast	0.554(3)	0.532(7)	0.545(5)	0.550(4)	0.559(2)	0.528(8)	0.534(6)	0.564(1)
scene	0.708(7)	0.724(2)	0.722(4)	0.711(6)	0.02(8)	0.712(5)	0.723(3)	0.764(1)
emotions	0.665(3)	0.656(4)	0.671(2)	0.655(5)	0.646(7)	0.638(8)	0.650(6)	0.681(1)
music	0.709(2)	0.615(7)	0.595(8)	0.689(3)	0.668(5)	0.663(6)	0.670(4)	0.788(1)
corel5k	0.390(1)	0.284(8)	0.290(7)	0.320(6)	0.339(5)	0.359(4)	0.361(3)	0.383(2)
enron	0.563(4)	0.584(1)	0.489(7)	0.580(2)	0.561(5)	0.538(6)	0.401(3)	0. 573(3)
bibtex	0.343(6)	0.234(7)	0.432(5)	0.443(4)	0.467(3)	0.232(8)	0.534(2)	0.543(1)
flags	0.622(6)	0.734(2)	0.542(8)	0.562(7)	0.731(3)	0.754(1)	0.688(5)	0.698(4)
birds	0.441(4)	0.424(6)	0.407(7)	0.475(3)	0.488(2)	0.438(5)	0.401(3)	0.491(1)
genbase	0.347(8)	0.387(5)	0.378(6)	0.356(7)	0.456(3)	0.476(2)	0.431(4)	0.565(1)
Average score	0.5342(3)	0.5174(6)	0.5071(7)	0.5341(4)	0.4935(8)	0.5338(5)	0.5393(2)	0.605(1)

Table 7. Hamming loss of different multi-label classification algorithms ↓

Dataset	EBR	ECC	EPS	RAkEL	Adaboost.MH	MLkNN	MLS	EMSTE
yeast	0.206(2)	0.210(4)	0.212(5)	0.247(8)	0.232(6)	0.209(3)	0.238(7)	0.202(1)
scene	0.0942(1)	0.098(4)	0.097(3)	0.138(6)	0.178(8)	0.099(5)	0.143(7)	0.095(2)
emotions	0.197(2)	0.205(3)	0.212(4)	0.265(7)	0.306(8)	0.235(6)	0.254(5)	0.192(1)
music	0.025(2)	0.021(1)	0.125(6)	0.109(5)	0.212(7)	0.238(8)	0.060(4)	0.055(3)
corel5k	0.174(3)	0.167(2)	0.160(1)	0.279(7)	0.294(8)	0.238(4)	0.255(6)	0.251(5)
enron	0.163(3)	0.187(6)	0.170(5)	0.168(4)	0.191(7)	0.218(8)	0.092(2)	0.051(1)
bibtex	0.249(4)	0.245(3)	0.259(6)	0.253(5)	0.278(8)	0.232(1)	0.277(7)	0.236(2)
flags	0.227(5)	0.297(8)	0.098(2)	0.162(4)	0.121(3)	0.228(6)	0.252(7)	0.031(1)
birds	0.043(2)	0.045(3)	0.046(4)	0.075(7)	0.052(5)	0.0.054(6)	0.151(8)	0.041(1)
genbase	0.022(1)	0.102(2)	0.140(3)	0.221(6)	0.314(8)	0.238(7)	0.211(5)	0.151(4)
Average score	0.14002(2)	0.1577(4)	0.1519(3)	0.1917 (5)	0.2178(7)	0.2189 (8)	0.1933(6)	0.1305(1)

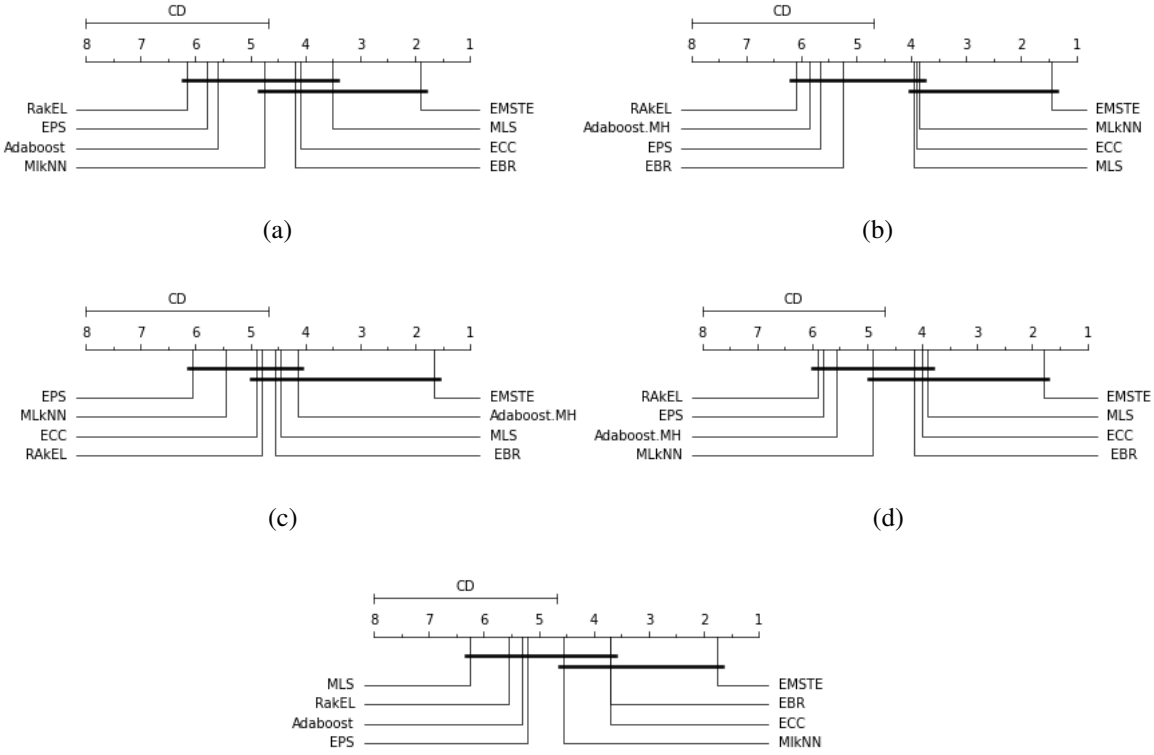

Figure 2. Critical distance diagrams for the comparison algorithms of different performance evaluation metrics (a) average precision, (b) F1, (c) macro F1, (d) micro F1, and (e) Hamming loss

In each CD diagram of Figure 2 any comparison algorithm connected with EMSTE are having a different performance among them significantly. EBR performs well on Hamming loss (HLoss) , because of its first-order label correlation approach that tries to optimize the Hamming loss. We can observe EMSTE outperforms in terms of HLoss. We have used ECC as a comparison algorithm to show how the high order label correlation approach outperforms the other approaches on MicroF1, which considers the global label correlations. But the proposed EMSTE outperforms in terms of these methods. Figure 3 shows the graph that compares the average score of performance evaluation metrics Average Precision and F1 Score of different comparison algorithms and Figure 4 shows the graph that compares the average score of performance evaluation metrics macro F1 and micro F1 Score and Hamming loss of different comparison algorithms. It is clear from the graph that the performance of the proposed method i.e., EMSTE is better in every case. EMSTE outperforms

compared to other methods concerning other evaluation metrics. This indicates the superiority of the proposed method and the effectiveness of augmentation of correlated features in meta-learning.

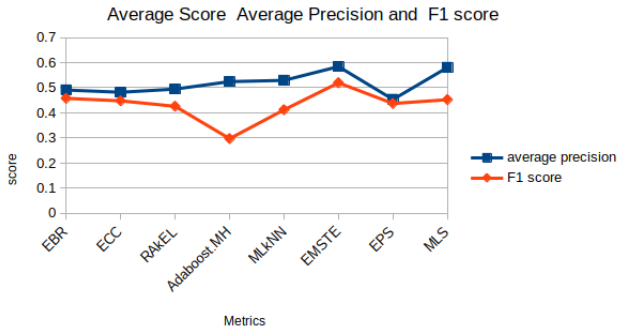

Figure 3. Average score of performance evaluation metrics F1 score

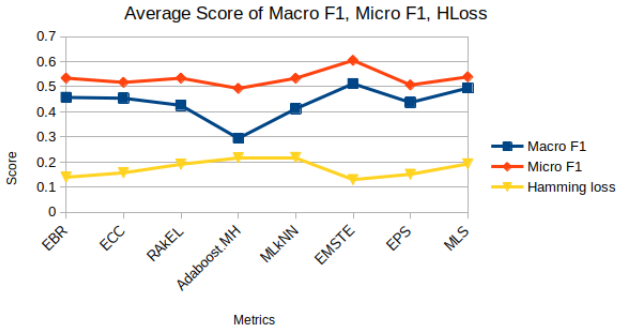

Figure 4. Average score of performance evaluation metrics macro F1, micro F1, Haming loss

7. CONCLUSION

In this paper, we have experimented with the novel extended multi-tier stacked ensemble method (EMSTE) for multi-label classification to enhance the performance of the classification by considering label correlated feature subset selection and by augmenting those features while constructing the meta data set at the generalization tier. In this approach, we are trying to control the bias and the variance at the ensemble tier by making use of combination schemes and improving the performance of classification at the generalization tier. We have tested our method with ten datasets and compared it with a number of different multi-label classification algorithms and our method is giving very good results.

REFERENCES

[1] I. Katakis, G. Tsoumakas, and I. Vlahavas, "Multilabel text classification for automated tag suggestion," in *Proceedings of the ECMLPKDD 2008 Discovery Challenge*, vol. 9, no. 3, pp. 1–9, 2008.

[2] H. Weng *et al.*, "Multi-label symptom analysis and modeling of TCM diagnosis of hypertension," in *Proceedings - 2018 IEEE International Conference on Bioinformatics and Biomedicine*, pp. 1922–1929, 2019, doi: 10.1109/BIBM.2018.8621173.

[3] R. E. Schapire and Y. Singer, "BoosTexter: a boosting-based system for text categorization," *Machine Learning*, vol. 39, no. 2, pp. 135–168, 2000, doi: 10.1023/a:1007649029923.

[4] R. B. Mangolin *et al.*, "A multimodal approach for multi-label movie genre classification," *Multimedia Tools and Applications*, vol. 81, no. 14, pp. 19071–19096, Jun. 2022, doi: 10.1007/s11042-020-10086-2.

[5] Y. Zhang and D. Y. Yeung, "Multilabel relationship learning," *ACM Transactions on Knowledge Discovery from Data*, vol. 7, no. 2, 2013, doi: 10.1145/2499907.2499910.

[6] M. L. Zhang and Z. H. Zhou, "A review on multi-label learning algorithms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 8, pp. 1819–1837, 2014, doi: 10.1109/TKDE.2013.39.
[7] G. Nasierding and A. Z. Kouzani, "Comparative evaluation of multi-label classification methods," in *International Conference on Fuzzy Systems and Knowledge Discovery*, May 2012, pp. 679–683, doi: 10.1109/FSKD.2012.6234347.
[8] L. Rokach, A. Schclar, and E. Itach, "Ensemble methods for multi-label classification," *Expert Systems with Applications*, vol. 41, no. 16, pp. 7507–7523, Jul. 2013, doi: 10.1016/j.eswa.2014.06.015.
[9] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown, "Learning multi-label scene classification," *Pattern Recognition*, vol. 37, no. 9, pp. 1757–1771, 2004, doi: 10.1016/j.patcog.2004.03.009.
[10] J. Read, "A pruned problem transformation method for multi-label classification," in *New Zealand Computer Science Research Student Conference*, 2008, pp. 143–150.
[11] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Random k-labelsets for multilabel classification," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 7, pp. 1079–1089, Jul. 2011, doi: 10.1109/TKDE.2010.164.
[12] G. Chen, Y. Song, F. Wang, and C. Zhang, "Semi-supervised multi-label learning by solving a sylvester equation," in *Proceedings of the 2008 SIAM International Conference on Data Mining*, Apr. 2008, vol. 1, pp. 410–419, doi: 10.1137/1.9781611972788.37.
[13] J. Read, B. Pfahringer, and G. Holmes, "Multi-label classification using ensembles of pruned sets," in *Proceedings - IEEE International Conference on Data Mining*, ICDM, pp. 995–1000, 2008, doi: 10.1109/ICDM.2008.74.
[14] M.-L. Zhang and Z.-H. Zhou, "ML-kNN: a lazy learning approach to multi-label learning," *Pattern Recognition*, vol. 40, no. 7, pp. 2038–2048, Jul. 2007, doi: 10.1016/j.patcog.2006.12.019.
[15] G. Tsoumakas, I. Katakis, and I. Vlahavas, "Mining multi-label data," in *Data Mining and Knowledge Discovery Handbook*, Boston, MA: Springer US, 2009, pp. 667–685.
[16] E. Hüllermeier, J. Fürnkranz, W. Cheng, and K. Brinker, "Label ranking by learning pairwise preferences," *Artificial Intelligence*, vol. 172, no. 16–17, pp. 1897–1916, 2008, doi: 10.1016/j.artint.2008.08.002.
[17] J. Fürnkranz, E. Hüllermeier, E. Loza Mencía, and K. Brinker, "Multilabel classification via calibrated label ranking," *Machine Learning*, vol. 73, no. 2, pp. 133–153, Nov. 2008, doi: 10.1007/s10994-008-5064-8.
[18] A. Elisseeff and J. Weston, "A kernel method for multi-labelled classification," in *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, 2001, pp. 681–687.
[19] Z.-F. He, M. Yang, Y. Gao, H.-D. Liu, and Y. Yin, "Joint multi-label classification and label correlations with missing labels and feature selection," *Knowledge-Based Systems*, vol. 163, pp. 145–158, Jan. 2019, doi: 10.1016/j.knosys.2018.08.018.
[20] Z. Abdallah, A. El-Zaart, and M. Oueidat, "An improvement of label power set method based on priority label transformation," *International Journal of Applied Engineering Research*, vol. 11, no. 16, pp. 9079–9087, 2016.
[21] X. Geng, "Label distribution learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 7, pp. 1734–1748, 2016, doi: 10.1109/TKDE.2016.2545658.
[22] X. Zhou, Z. Wei, M. Xu, S. Qu, and G. Guo, "Facial depression recognition by deep joint label distribution and metric learning," *IEEE Transactions on Affective Computing*, vol. 13, no. 3, pp. 1605–1618, Jul. 2022, doi: 10.1109/TAFFC.2020.3022732.
[23] X. Wen *et al.*, "Adaptive variance based label distribution learning for facial age estimation," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 12368, 2020, pp. 379–395.
[24] Y. Xia, K. Chen, and Y. Yang, "Multi-label classification with weighted classifier selection and stacked ensemble," *Information Sciences*, vol. 557, pp. 421–442, May 2021, doi: 10.1016/j.ins.2020.06.017.
[25] J. Lee and D.-W. Kim, "Mutual Information-based multi-label feature selection using interaction information," *Expert Systems with Applications*, vol. 42, no. 4, pp. 2013–2025, Mar. 2015, doi: 10.1016/j.eswa.2014.09.063.
[26] J. Lee and D. W. Kim, "SCLS: Multi-label feature selection based on scalable criterion for large label set," *Pattern Recognition*, vol. 66, pp. 342–352, 2017, doi: 10.1016/j.patcog.2017.01.014.
[27] 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, doi: 10.1016/j.neucom.2015.06.010.
[28] P. Zhang, W. Gao, J. Hu, and Y. Li, "Multi-label feature selection based on high-order label correlation assumption," *Entropy*, vol. 22, no. 7, Jul. 2020, doi: 10.3390/e22070797.
[29] M. Sorower, "A literature survey on algorithms for multi-label learning," *Oregon State University, Corvallis*. pp. 1–25, 2010.
[30] "Mulan: a java library for multi-label learning." sourceforge, https://mulan.sourceforge.net/datasets-mlc.html (accessed Dec. 1, 2022).
[31] Z. F. He, M. Yang, and H. D. Liu, "Joint learning of multi-label classification and label correlations," *Ruan Jian Xue Bao/Journal of Software*, vol. 25, no. 9, pp. 1967–1981, 2014, doi: 10.13328/j.cnki.jos.004634.
[32] J. M. Moyano, E. L. Gibaja, K. J. Cios, and S. Ventura, "Review of ensembles of multi-label classifiers: Models, experimental study and prospects," *Information Fusion*, vol. 44, pp. 33–45, Nov. 2018,

doi: 10.1016/j.inffus.2017.12.001.
[33] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 5782, no. 2, 2009, pp. 254–269.
[34] M.-L. Zhang and K. Zhang, "Multi-label learning by exploiting label dependency," in *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2010, doi: 10.1145/1835804.1835930.
[35] E. Spyromitros, G. Tsoumakas, and I. Vlahavas, "An empirical study of lazy multilabel classification algorithms," in *Artificial Intelligence: Theories, Models and Applications*, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 401–406, 2008.
[36] "MEKA," *GitHub*. https://waikato.github.io/meka/ (accessed Jan. 1, 2022).
[37] E. A. Tanaka, S. R. Nozawa, A. A. Macedo, and J. A. Baranauskas, "A multi-label approach using binary relevance and decision trees applied to functional genomics," *Journal of Biomedical Informatics*, vol. 54, pp. 85–95, Apr. 2015, doi: 10.1016/j.jbi.2014.12.011.
[38] Z. Wei, H. Zhang, Z. Zhang, W. Li, and D. Miao, "A naive Bayesian multi-label classification algorithm with application to visualize text search results," *International Journal of Advanced Intelligence*, vol. 3, no. 2, pp. 173–188, 2011.
[39] A. Patle and D. S. Chouhan, "SVM kernel functions for classification," in *2013 International Conference on Advances in Technology and Engineering*, 2013, doi: 10.1109/ICAdTE.2013.6524743.
[40] J. Huang, G. Li, Q. Huang, and X. Wu, "Learning label specific features for multi-label classification," in *2015 IEEE International Conference on Data Mining*, Nov. 2015, pp. 181–190, doi: 10.1109/ICDM.2015.67.
[41] A. Pakrashi and B. M. Namee, "Stacked-MLkNN: a stacking based improvement to multi-label k-nearest neighbours," in *Proceedings of Machine Learning Research*, 2017, pp. 51–63.
[42] M. Friedman, "A comparison of alternative tests of significance for the problem of m rankings," *The Annals of Mathematical Statistics*, vol. 11, no. 1, pp. 86–92, Mar. 1940, doi: 10.1214/aoms/1177731944.
[43] J. Demsar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.

BIOGRAPHIES OF AUTHORS

Hemavati received her Bachelor of Engineering degree in Information Science and Engineering, and M.Tech in Computer Science and Engineering from Siddaganga Institute of Technology, Tumakuru, VTU University, Belgaum, India. She is currently a research scholar carrying out her research work in PAMI Lab, IISc Bangalore, Karnataka, India. She is an Assistant Professor at Siddaganga Institute of Technology, Tumakuru, India. Her areas of interests are AI and machine learning. She can be contacted at email: hemavd@sit.ac.in.

Visweswariah Susheela Devi is a Principal Research Scientist in the Department of Computer Science and Automation, Indian Institute of Science. She received her Ph.D. from Indian Institute of Science, Bangalore. Her research topics include pattern recognition, data mining, soft computing, machine learning, Deep learning. She wrote four Books, many book chapters, and published many papers. She is a reviewer for many conferences and journal papers including IEEE. She was session chair for many conferences. She supervises M.Tech., M.Tech. research and Ph.D. Students. She has taught many courses like soft computing, intelligent agents, and machine learning. She has delivered many invited lecture talks. She can be contacted at email: susheela@iisc.ac.in.

Ramalingappa Aparna received her M.S. from BITS Pillani, and Ph.D. from VTU, Belgaum. She has been working as Professor and Head of the department, at Siddaganga Institute of Technology Tumakuru. Her main research interests include cryptography and network security, security in wireless sensor networks, routing issues in ad hoc wireless networks, and machine learning. She has taught many subjects for undergraduate and post graduate students, and she is supervising M.Tech. and Ph.D. students. She can be contacted at email: raparna@sit.ac.in.