

Many-Objective Ensemble-Based Multilabel Classification

Marcos M. Raimundo and Fernando J. Von Zuben^(✉)

LBiC/DCA/FEEC - University of Campinas, Campinas, SP, Brazil
{marcos,vonzuben}@dca.fee.unicamp.br

Abstract. This paper proposes a many-objective ensemble-based algorithm to explore the relations among the labels on multilabel classification problems. This proposal consists in two phases. In the first one, a many-objective optimization method generates a set of candidate components exploring the relations among the labels, and the second one uses a stacking method to aggregate the components for each label. By balancing or not the relevance of each label, two versions were conceived for the proposal. The balanced one presented a good performance for recall and F1 metrics, and the unbalanced one for 1-Hamming loss and precision metrics.

Keywords: Multilabel classification · Many-objective optimization
Multiobjective optimization · Ensemble of classifiers · Stacking

1 Introduction

Multilabel classification is a generalization of the conventional classification problem in machine learning when, instead of assigning a unique, relevant label for each object, it is possible to assign more than one label per object. A straightforward approach, called Binary Relevance (BR), ignores any possible relationship among the labels and learns one classifier per label, for example, using kNN with Bayesian inference [23]. BR is computationally efficient, but it is not capable of exploring the relations among the labels to improve generalization. The main proposals devoted to promoting task relationship rely on Label Powerset, Classification Chains, and Multitask Learning.

Label powerset consists in transforming the multilabel problem into a multiclass one by creating a class for each combination of original labels. Despite exploring the relationship of labels, this proposal promotes an exponential growth of classes in the multiclass equivalent problem. Some solutions for this issue were proposed: converting the powerset process in random subsets of labels which are aggregated by simple voting [21]; excluding the labels on the multiclass equivalent problem characterized by few objects [12]; heuristically subsampling to overcome unbalanced data [2].

Considering an ordered sequence of labels, **Classification Chains** create a sequence of classifiers, each one taking the predicted relevance of the labels provided by classifiers previously trained. The considered sequence can be nominal

or random [13], and the architecture can be a tree instead of a sequence [10], so that the prediction depends on the parents of the label. Also, the classification can be based on the relevance probability [5].

Multitask learning creates binary relevance classifiers jointly exploring the relation of labels by structure learning [1]. This can be done by modeling the dependence among the labels using Ising-Markov Random Fields, further applied to constrain the flexibility of the task parameters adjustment [6], or using a multi-target regression proposal that explores multiple output relations in data streams [7].

Other methods were considered to extend these main proposals. **Ensembles** were proposed to increase robustness by resampling [12]; generating ensemble components using powersets in random sets of labels [21], and filtering then using genetic algorithms and rank-based proposals [4]; generating multiple classifiers by changing the label order in classification chains [13], and using many state-of-the-art multilabel classifiers to compose ensembles with different aggregation methods [19]. **Meta-learning** methods, instead of predicting the relevant labels found by binary relevance, predict the labels with higher membership degree, such that the number of predicted labels are estimated by a previously trained cardinality classifier [15, 20], or by a fixed optimal number of labels [11]. **Multi-objective optimization** was used to: create ensembles by optimizing a novel accuracy metric that takes into account the correlation of the labels and a diversity ensemble metric using evolutionary multiobjective optimization [16]; train an RBF network considering different sets of performance metrics as conflicting objectives [17, 18]; and make feature selection in ML-kNN classifiers [22].

In this paper, we propose a novel ensemble method that uses a many-objective optimization approach to generate components exploring the relations among the labels (by weighted averaging the loss on each label), followed by a stacking method to aggregate the components for each label.

2 Multiobjective Optimization

Based on the Noninferior Set Estimation (NISE) method [3], the multiobjective optimization used in this work is an adaptive process that iteratively calculates a parameter vector \mathbf{w} of the weighted method (Definition 1) using the previously found solutions and its related parameter vectors. The applied method, called MONISE, differs from NISE by being capable of dealing with a high number of objectives (many-objective problems) [9].

Definition 1. *The definition of the weighted method is as follows:*

$$\begin{aligned}
 & \underset{\mathbf{x}}{\text{minimize}} \quad \mathbf{w}^\top \mathbf{f}(\mathbf{x}) \\
 & \text{subject to} \quad \mathbf{x} \in \Omega, \\
 & \quad \mathbf{f}(\mathbf{x}) : \Omega \rightarrow \Psi, \Omega \subset \mathbb{R}^n, \Psi \subset \mathbb{R}^m \\
 & \quad \mathbf{w} \in \mathbb{R}^m, \mathbf{w}_i \geq 0 \quad \forall i \in \{1, 2, \dots, m\}.
 \end{aligned} \tag{1}$$

Parameter finding. Let us consider a utopian solution $\mathbf{z}^{utopian}$, composed of the infimum of all objectives, as well as a set $P, |P| \geq 1$ of solutions described by the objective vector $\mathbf{f}(\mathbf{x}^i)$, $i \in P$, and the weights \mathbf{w}^i , $i \in P$, used to find the solution based on the weighted method (Definition 1). We can then iteratively determine the approximation $\bar{\mathbf{r}}$ and the relaxation $\underline{\mathbf{r}}$ of the current frontier.

The **frontier relaxation** is given by the optimal solution of the weighted method. So, it is possible to conclude that $\mathbf{w}^i \top \underline{\mathbf{r}} \geq \mathbf{w}^i \top \mathbf{f}(\mathbf{x}^i)$, $\forall \underline{\mathbf{r}} \in \Psi$. The **frontier approximation** is also given by the optimality of the weighted method. Given a weight vector \mathbf{w} , the efficient solution must satisfy $\mathbf{w} \top \bar{\mathbf{r}} \leq \mathbf{w} \top \mathbf{f}(\mathbf{x}^i)$, $\forall i \in P$.

These concepts act as constraints to the best and worst possible solutions for the weighted method. The parameter selection procedure must find a new parameter vector \mathbf{w} that leads to the maximum gap between $\underline{\mathbf{r}}$ and $\bar{\mathbf{r}}$. The problem formalized in Definition 2 is used to achieve this goal [9].

Definition 2

$$\begin{aligned}
 & \underset{\mathbf{w}, \underline{\mathbf{r}}, \bar{\mathbf{r}}}{\text{minimize}} && -\mu = \mathbf{w} \top \underline{\mathbf{r}} - \mathbf{w} \top \bar{\mathbf{r}} \\
 & \text{subject to} && \mathbf{w}^i \underline{\mathbf{r}} \geq \mathbf{w}^i \mathbf{f}(\mathbf{x}^i) \quad \forall i \in P \\
 & && \mathbf{w} \top \bar{\mathbf{r}} \leq \mathbf{w} \top \mathbf{f}(\mathbf{x}^i) \quad \forall i \in P \\
 & && \underline{\mathbf{r}} \geq \mathbf{z}^{utopian}, \underline{\mathbf{r}} \leq \bar{\mathbf{r}}, \mathbf{w} \geq \mathbf{0}, \mathbf{w} \top \mathbf{1} = 1.
 \end{aligned}
 \tag{2}$$

The **initialization** of the algorithm consists of finding $\mathbf{z}^{utopian}$, by optimizing each objective separately (finding the infimum of each objective), and finding a solution with an arbitrary \mathbf{w} . Each **iteration** is composed of: 1. Finding a new parameter \mathbf{w}^k using Definition 2; 2. Using this parameter to find a new solution $\mathbf{f}(\mathbf{x}^k)$; 3. Queue up the new solution k in the set P . The **stop criterion** is achieved when the number of solutions found achieves a limit R .

3 Statistical Model

The multilabel classification problem involves a set of N samples, also called objects, where $\mathbf{x}_i \in \mathbb{R}^d : i \in \{1, \dots, N\}$ corresponds to the input feature vector and $\mathbf{y}_i \in \{0, 1\}^L : i \in \{1, \dots, N\}$ is the output feature vector we are willing to predict. In multilabel classification any output vector value \mathbf{y}_i^l indicates the membership degree of label l to sample i , and any sample can have 0 to L labels assigned.

A binary relevance approach using logistic regression creates a classifier for each label l choosing the Bernoulli distribution as the predictive distribution $p(y|z) = z^y(1 - z)^{1-y}$ and the softmax function $f(\mathbf{x}, \theta) = \frac{e^{\theta_1 \top \mathbf{x}}}{e^{\theta_0 \top \mathbf{x}} + e^{\theta_1 \top \mathbf{x}}} \in [0, 1]$ as the input-output model. The optimization problem applied to find the optimal parameter vector θ^l is given by:

$$\min_{\theta^l} l(\theta^l, \mathbf{x}, \mathbf{y}^l) = - \sum_{i=1}^N \left[\frac{1}{n_1^l} \mathbf{y}_i^l \ln (f(\mathbf{x}_i, \theta^l)) + \frac{1}{n_0^l} (1 - \mathbf{y}_i^l) \ln (1 - f(\mathbf{x}_i, \theta^l)) \right] \quad (3)$$

where n_1^l is the number of 1s in label l and n_0^l is the number of 0s in label l .

Notice that, in this approach, the aim is to find the optimal parameters θ^l for specific label l using the general input vectors $\mathbf{x}_i \in \mathbb{R}^d : i \in \{1, \dots, N\}$ and specific output values $\mathbf{y}_i \in \{0, 1\} : i \in \{1, \dots, N\}$. In our approach, instead of finding a parameter vector (and a classifier) for each label, given a weighting factor \mathbf{v}^l for each label, we use a joint optimization to find a single classifier, thus guiding to:

$$\min_{\theta} \sum_{l=1}^L \mathbf{v}_l l(\mathbf{x}, \mathbf{y}^l, \theta) + \lambda \|\theta\|^2 \equiv \sum_{l=1}^L \mathbf{w}_l l(\mathbf{x}, \mathbf{y}^l, \theta) + \mathbf{w}_{L+1} \|\theta\|^2. \quad (4)$$

where $\|\theta\|^2$ is the regularization component, λ is the regularization parameter, $\mathbf{w}_i = \frac{\mathbf{v}_i}{\sum_{k=1}^L \mathbf{v}_k + \lambda} \forall i \in \{1, \dots, L\}$, and $\mathbf{w}_{L+1} = \frac{\lambda}{\sum_{k=1}^L \mathbf{v}_k + \lambda}$.

4 Stacking and Proposed Methodology

Stacking is an ensemble methodology that uses the outcome of the ensemble components (learning machines trained by an ensemble generation methodology, represented in Step 1 of Fig. 1a) to train another classifier (Step 2 of Fig. 1a) which will be responsible for making the prediction. In our proposal, the many-objective optimization method is responsible for generating R classifiers to compose the ensemble, and we create a stack classifier to predict each label l using logistic regression as the classification model:

$$\min_{\hat{\theta}^l} - \sum_{i=1}^N \left[\frac{1}{k_1^l} \mathbf{y}_i^l \ln (f(\mathbf{z}_i, \hat{\theta}^l)) + \frac{1}{k_0^l} (1 - \mathbf{y}_i^l) \ln (1 - f(\mathbf{z}_i, \hat{\theta}^l)) \right] \quad (5)$$

where \mathbf{z}_i^j is the degree of membership predicted by the j -th ensemble component with relation to the i -th sample.

In **Step 1**, ensemble components ($\{\theta_1, \theta_2, \dots, \theta_l\}$) are generated by finding a set of efficient solutions, to the formulation of Eq. 4 using the methodology described in Sect. 2 (Fig. 1b). This generator of many-objective ensemble components can be seen as a **feature vector mapping** ($fm(\theta, x)$), so that each mapped feature is a classifier (Fig. 1c) associated with a distinct weight vector. From Eq. 4 we can realize that each classifier will give a distinct weight to the loss at each label and also to the regularization term. In **Step 2** (represented in Fig. 1d), the output of all efficient solutions are aggregated using a stacking approach [14], which can be seen as a cross-validation procedure in the mapped feature space \hat{x} using the model of Eq. 5. The training procedure in the second step is done for each label l employing the same feature vector \hat{x} , but adopting the output \hat{y}^l of the worked label.

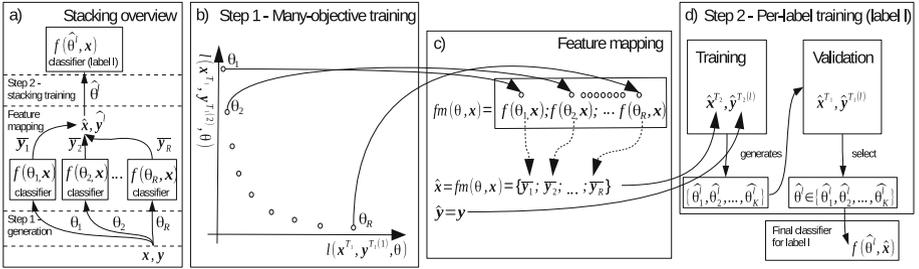


Fig. 1. Many-objective training followed by a stacking aggregation representation.

The set of classifiers were generated by a weighted average of the label losses. For this reason, not all classifiers will have a good performance for a specific label, thus requiring a more flexible aggregation, such as stacking, to create a final classifier.

5 Experimental Setting

5.1 Definition of Datasets and Evaluation Metrics

To evaluate the potential of the proposed multiobjective ensemble-based methodology we consider 6 datasets¹. Table 1 provides the main aspects of these datasets. Aiming at obtaining better statistical results, we used a 10-fold split to create 10 independent test sets with 10% of the samples, and the remaining samples are randomly divided into 75% for training and 25% for validation for the baseline algorithms and 50% for T_1 set and 50% for T_2 set for the proposed method. T_1 set was used to create the ensemble components (Fig. 1b), and T_2 set to train the stacked classifiers (Fig. 1d). T_1 set was used again to select the model in the stacking training procedure (Fig. 1d).

The used evaluation metrics were: 1-Hamming Loss (1-hl), precision, accuracy, recall, F1 and Macro-F1 [6, 8], all of those metrics associated with a quality measure in the interval [0, 1] so that higher values indicate a better method.

5.2 Contenders for a Comparative Analysis and Versions of the Proposed Method

To create a solid baseline we compared our method with 5 other approaches: Binary Relevance, Classification Chains [13], RakelD [21], Label Powerset². All of those methods were implemented using Logistic Regression³ as the base classifier

¹ Available at mulan.sourceforge.net/datasets-mlc.html.

² Implementations available at <http://scikit.ml/>.

³ Available at <http://scikit-learn.org>.

Table 1. Description of the benchmark datasets.

Name	Instances	Attributes	Labels	Cardinality	Density of 1s
Emotions	593	72	6	1.869	0.311
Scene	2407	294	6	1.074	0.179
Flags	194	19	7	3.392	0.485
Yeast	2417	103	14	4.237	0.303
Birds	645	260	19	1.014	0.053
Genbase	662	1186	27	1.252	0.046

and had its parameters selected using hyperopt⁴ with 50 evaluations to tune regularization strength and 50 more evaluations if the method involves another parameter (RakelD). Since the proposed and baseline algorithms use logistic regression as the base classifier, the attributes are considered as a vector of real numerical values.

In our proposal, we generate $10 * (L + 1)$ ensemble components, and the parameter selection on the stacking phase was implemented by cross-validation with 50 evaluations. We developed two versions of our proposal. These versions were created by balancing or not the importance of a label according to the stacking by changing the constants k_1^l and k_0^l on Eq. 5. In the unbalanced approach (described as MOn) k_1^l and k_0^l were set to 1, and in the balanced approach (described as MOb) k_1^l was set to the number of 1s for this labels and k_0^l for the specific number of 0s.

6 Results

To promote an extensive comparison we presented the results from two perspectives. Figure 2 presents the average performance, calculated over the 10-folds, for all evaluated metrics for each dataset. And to make a more incisive evaluation, we used a Friedman paired test with $p = 0.01$ comparing all folders for all datasets, followed by a Finner post-hoc test with the same p , if Friedman test were rejected. Table 2 contains the evaluated method in the rows, and, for each performance metric, the Friedman ranking in the first column, how many methods are worse than the evaluated metric in the second column, and how many methods are better in the third column. This ordering relation (better and worse) is accounted only if there is a statistical significance according to the Finner post-hoc test. Looking to RakelD for the precision score, it is statistically significantly better than the worst ranked method: MOb (4.92), and statistically significantly worse than the three better-ranked methods: MOn (1.99), BR (2.74) and CC (2.94).

⁴ Available at <https://github.com/hyperopt/hyperopt>.

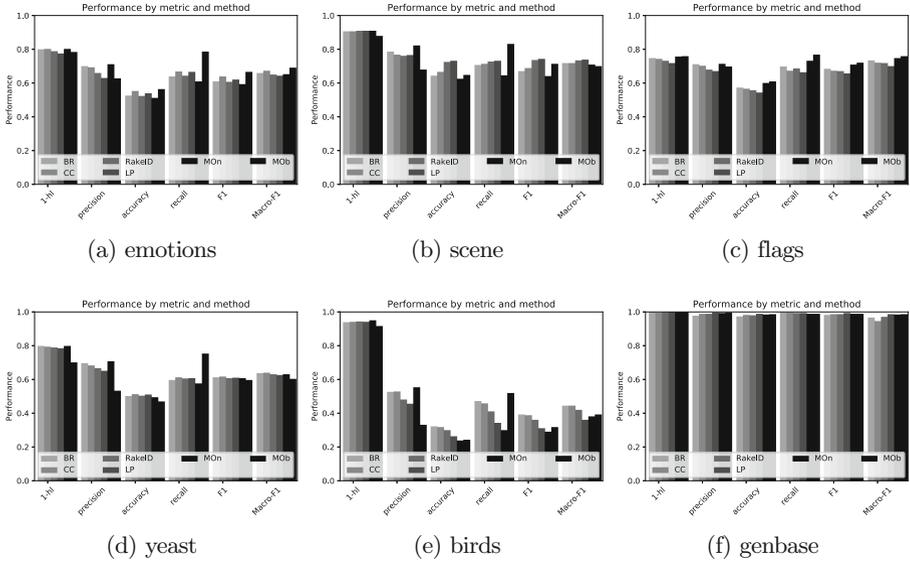


Fig. 2. Average performance of the evaluated methods for each metric in each dataset.

Table 2. Average ranking and statistical comparisons for each metric.

Method	1-hl		Precision			Accuracy			Recall			F1			Macro-F1			
	Rank	>	<	Rank	>	<	Rank	>	<	Rank	>	<	Rank	>	<	Rank	>	<
BR	3.29	1	1	2.74	3	0	3.79	0	1	3.73	1	1	3.68	0	0	3.22	0	0
CC	3.29	1	1	2.94	3	1	2.64	3	0	3.16	1	1	2.9	1	0	2.87	0	0
RakeID	3.64	1	1	3.95	1	3	3.35	0	0	3.59	1	1	3.3	1	0	3.45	0	0
LP	3.89	0	1	4.43	0	3	3.12	1	0	3.85	1	1	3.54	0	0	4.01	0	0
MOn	2.23	5	0	1.99	4	0	4.31	0	2	4.96	0	5	4.37	0	3	3.75	0	0
MOB	4.63	0	4	4.92	0	4	3.75	0	1	1.69	5	0	3.17	1	0	3.66	0	0

7 Concluding Remarks

In this work, we successfully proposed a many-objective ensemble-based classifier to multilabel classification. Analyzing both Fig. 2 and Table 2, it is possible to see that MOn is the best-ranked classifier on 1-hl and precision but falling away on recall, accuracy, and F1. MOB is one of the best-ranked classifiers on recall and F1, but has difficulties on 1-hl and precision. These findings indicate that these two classifiers are biased for some metrics, exhibiting complementary performance. This behavior is due to the low density of the datasets, and to the fact that the non-balanced stacked model focuses the prediction on the 0s, thus producing high precision, as long as the balanced approach is predicting 1s more frequently, explaining the high recall score.

This scenario where an approach has a good performance on specific metrics at the expense of performance loss for other metrics can be useful in some applications. Given the absence of a dominant method for all metrics, our proposals can be seen as valuable choices in metric-driven scenarios. Also, since the complementary behavior was generated changing parameters, further exploration using ensembles of many-objective trained classifiers can promote good classifiers with different performance profiles.

Acknowledgments. This research was supported by grants from FAPESP, process #14/13533-0, and CNPq, process #309115/2014-0.

References

1. Caruana, R.: Multitask learning. *Mach. Learn.* **75**(1), 41–75 (1997)
2. Charte, F., Rivera, A.J., del Jesus, M.J., Herrera, F.: MLeNN: a first approach to heuristic multilabel undersampling. In: Corchado, E., Lozano, J.A., Quintián, H., Yin, H. (eds.) IDEAL 2014. LNCS, vol. 8669, pp. 1–9. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10840-7_1
3. Cohon, J.L., Church, R.L., Sheer, D.P.: Generating multiobjective trade-offs: an algorithm for bicriterion problems. *Water Resour. Res.* **15**(5), 1001–1010 (1979)
4. Costa, N., Coelho, A.L.V.: Genetic and ranking-based selection of components for multilabel classifier ensembles. In: Proceedings of the 2011 11th International Conference on Hybrid Intelligent Systems, HIS 2011, pp. 311–317 (2011)
5. Dembczy, K.: Bayes optimal multilabel classification via probabilistic classifier chains. In: Proceedings of the 27th International Conference on Machine Learning, pp. 279–286 (2010)
6. Gonçalves, A.R., Von Zuben, F.J., Banerjee, A.: Multi-label structure learning with Ising model selection. In: Proceedings of 24th International Joint Conference on Artificial Intelligence, pp. 3525–3531 (2015)
7. Osojnik, A., Panov, P., Džeroski, S.: Multi-label classification via multi-target regression on data streams. In: Japkowicz, N., Matwin, S. (eds.) DS 2015. LNCS (LNAI), vol. 9356, pp. 170–185. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24282-8_15
8. Madjarov, G., Kocev, D., Gjorgjevikj, D., Džeroski, S.: An extensive experimental comparison of methods for multi-label learning. *Pattern Recogn.* **45**(9), 3084–3104 (2012)
9. Raimundo, M.M., Von Zuben, F.J.: MONISE - many objective non-inferior set estimation, pp. 1–39 (2017). [arXiv:1709.00797](https://arxiv.org/abs/1709.00797)
10. Ramírez-Corona, M., Sucar, L.E., Morales, E.F.: Hierarchical multilabel classification based on path evaluation. *Int. J. Approx. Reason.* **68**, 179–193 (2016)
11. Ramón Quevedo, J., Luaces, O., Bahamonde, A.: Multilabel classifiers with a probabilistic thresholding strategy. *Pattern Recogn.* **45**(2), 876–883 (2012)
12. Read, J., Pfahringer, B., Holmes, G.: Multi-label classification using ensembles of pruned sets. In: Proceedings - IEEE International Conference on Data Mining, ICDM, pp. 995–1000 (2008)
13. Read, J., Pfahringer, B., Holmes, G., Frank, E.: Classifier chains for multi-label classification. *Mach. Learn.* **85**(3), 333–359 (2011)
14. Rokach, L.: Ensemble-based classifiers. *Artif. Intell. Rev.* **33**(1–2), 1–39 (2010)

15. Satapathy, S.C., Govardhan, A., Raju, K.S., Mandal, J.K.: Emerging ICT for Bridging the Future - Proceedings of the 49th Annual Convention of the Computer Society of India (CSI) Volume 2. *Advances in Intelligent Systems and Computing*, vol. 338, pp. 1–4. Springer, Heidelberg (2015). <https://doi.org/10.1007/978-3-319-13731-5>
16. Shi, C., Kong, X., Yu, P.S., Wang, B.: Multi-label ensemble learning. In: Gunopulos, D., Hofmann, T., Malerba, D., Vazirgiannis, M. (eds.) *ECML PKDD 2011*. LNCS (LNAI), vol. 6913, pp. 223–239. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23808-6_15
17. Shi, C., Kong, X., Fu, D., Yu, P.S., Wu, B.: Multi-label classification based on multi-objective optimization. *ACM Trans. Intell. Syst. Technol.* **5**(2), 1–22 (2014)
18. Shi, C., Kong, X., Yu, P.S., Wang, B.: Multi-objective multi-label classification. In: *Proceedings of the 12th SIAM International Conference on Data Mining, SDM 2012*, pp. 355–366 (2012)
19. Tahir, M.A., Kittler, J., Bouridane, A.: Multilabel classification using heterogeneous ensemble of multi-label classifiers. *Pattern Recogn. Lett.* **33**(5), 513–523 (2012)
20. Tang, L., Rajan, S., Narayanan, V.K.: Large scale multi-label classification via metalabeler. In: *Proceedings of the 18th International Conference on World Wide Web*, p. 211 (2009)
21. Tsoumakas, G., Katakis, I., Vlahavas, I.: Random k-labelsets for multilabel classification. *IEEE Trans. Knowl. Data Eng.* **23**(7), 1079–1089 (2011)
22. Yin, J., Tao, T., Xu, J.: A multi-label feature selection algorithm based on multi-objective optimization (2015)
23. Zhang, M.L., Zhou, Z.H.: ML-KNN: a lazy learning approach to multi-label learning. *Pattern Recogn.* **40**(7), 2038–2048 (2007)