Multi-label Sparse Representation based Classification

Journal:	*Transactions on Knowledge and Data Engineering*
Manuscript ID	TKDE-2018-11-1141
Manuscript Type:	Concise
Keywords:	H.2.8.b Clustering, classification, and association rules < H.2.8 Database Applications < H.2 Database Management < H Informatio, I.4.7.a Feature representation < I.4.7 Feature Measurement < I.4 Image Processing and Computer Vision < I Computing Methodologie

Multi-label Sparse Representation based Classification

Akshay Sethi, Angshul Majumdar, Mayank Vatsa and Richa Singh

Abstract—In this paper we propose a novel algorithm for multi-label classification based on the sparse representation based classification (SRC) paradigm. SRC was originally developed for single label classification where each class is supposed to lie on a separate sub-space. In the original SRC algorithm for single label, classification happens by identifying the correct subspace corresponding to the test sample. For multi-label classification, since each sample is associated with multiple classes, one needs a union of subspaces model for representation. Therefore, during classification of a test sample, the objective is to identify the union of subspaces that represents the sample; this reveals the identities / class labels of the test sample. Here we propose the use of l_1-norm minimization algorithm to retrieve the union of subspaces. Experiments have been carried out on five benchmark datasets, and compared against four algorithms. The proposed method yields among the best results on these databases.

Index Terms—Multi-label classification, Sparse representation classification, kernel methods, classification.

1 Introduction

MAJORITY of studies on predictive analytics concentrate on the problem of single label classification. In this problem, each sample has a unique class label. Classical examples of such problems include face recognition, fingerprint recognition, speaker identification, and object categorization. In these problems, even though each of the problems may be multi-class classification problems, each sample belongs to one class only.

In several applications, it is possible that the samples cannot be crisply classified as a single class. The intrinsic nature of the problem dictates that one sample can be associated with multiple classes / identities. For example consider the problem of emotion recognition from music [1], [2] or text [3], [4]. It would not be correct to assume that, each song or tweet will reflect only a single emotion; being associated with multiple emotions is more likely.

Similarly, consider the problem of scene understanding. Traditionally the problem was handled by segmenting different objects in the scene and applying single label classification algorithms on the segmented objects to detect its type. However, a more natural way to analyse scenes is via a multi-label classification approach [5], [6]. Every scene is likely to be consisting of multiple classes, e.g. a natural scene on a beach may have the sky, clouds, sea, people etc; or an indoor scene in an office may have desks, work stations, and people.

Another real world application is in power engineering, which is dubbed as 'Non-intrusive load monitoring' (NILM). It attempts to detect what are the appliances which are ON / ACTIVE during a particular period, given the total reading from the smart-meter [7], [8]. In any period of time, more than one appliance is usually ON. Therefore, each set of readings from the smart-meter is likely to be associated with multiple appliance identities / classes. The aforesaid problems are examples of multi-label classification. In such problems, each sample can be associated with multiple classes. We have given a few examples here, but the application of multi-label classification extends to many other areas of applied data analytics.

Many multi-label classification algorithms have been proposed in the past. They are largely classified into three categories:

1. Algorithm adaptation
2. Problem transformation
3. Ensemble methods

We first briefly discuss them in the literature review section. In this research, we propose a novel algorithm for multi-label classification based on the sparse representation based classification (SRC) paradigm [9]. We demonstrate how the SRC model can be easily extended to accommodate multi-label classification. The proposed approach falls under the category of algorithm adaptation. We also propose to modify the kernelized version of SRC [10], [11] to solve multi-label classification. Experiments on multiple databases showcase the efficacy of the proposed algorithm.

2 Literature Review

The slightly dated review article [12] categorized multi-label classification into algorithm adaptation approaches and problem transformation approaches. Since the publication of [12] (over a decade back), many studies were published that used ensemble methods for the said task. This led to the addition of the third category in [13]. This section provides a brief description of each of these approaches below.

- *A. Sethi, A. Majumdar, M. Vatsa and R. Singh are with Indraprastha Institute of Information Technology, New Delhi, India, 110020. E-mail: {akshay14133, angshul, mayank and rsingh}@iiitd.ac.in.*

- *The authors are partially supported by the Infosys Center for Artificial Intelligence @ IIITD.*

2.1 Algorithm adaptation

Nearly all popular single label classification techniques have been modified to address the multi-label classification problem. For example, in [14] the popular ADA-BOOST algorithm was modified to address the said problem. In one variant, it was designed to minimize the Hamming loss and in another, a ranking method was proposed so that correct classes end up at the top.

One of the most popular techniques for multi-label classification ML-KNN [15] is based on the K nearest neighbor algorithm. In simple terms, instead of looking at only one class with the shortest aggregate distance, multiple classes with short distances are considered.

In [16], neural networks were modified to address the problem of multi-label classification. The main idea was to modify the cost function at the output layer so as to handle multiple classes. Neural networks for multi-label classification is regaining momentum with the success of deep learning. In [17], [18] autoencoders were modified to address the said problem. Similarly in [19], [20] deep convolutional neural networks were modified for multi-label classification (arising in scene understanding).

There have been studies that directly proposed to adopt decision trees [21] and support vector machines [22] for multi-label classification. However, these algorithms were found to be more suited for the category of problem transformation or ensemble approaches.

2.2 Problem transformation

In this class of techniques, the multi-label classification problem is recasted as single label classification problem(s). Problem transformation can be further subdivided into three branches.

The label power-set approach [23], [24] considers combination of multiple labels into single super labels. This expands the number of classes to a large extent, but allows standard single label classification algorithms to be directly applied. Studies like [25] proposed techniques to prune the number of such super labels.

The second branch is called the binary relevance method [26]. It is one of the simplest approaches for multi-label classification where one-against-all strategy is employed to convert multi-label classification into a series of binary classification problems. Scalable implementations for this approach are available [27]. As mentioned before, support vector machines are suitable for this approach [28].

The third method falling under this category adopts a pair-wise classification approach. For C classes, C(C-1)/2 classifiers are trained to distinguish between as many pair of classes. The output from all the classifiers are fused to arrive at the final decision by majority voting [29], [30]. More sophisticated voting algorithms have also been proposed for this approach [31].

2.3 Ensemble methods

Ensemble methods exist on top of problem transformation or algorithm adaptation approaches. The Random K label-sets (RaKEL) [32] approach is a classical example of ensemble methods. Instead of exactly solving the label power-set problem with many super labels, the problem is split into random sub-sets and each of them are used for training a classifier. The output from the ensemble of classifiers is fused to draw inference. Classifier chains [33] under binary relevance fall under this category as well. They sum the total number of times each label appears from the one-against-all classifiers and use a threshold to determine the multiple labels.

In algorithm adaptation approaches, we mentioned that although decision trees have been used for multi-label classification, they are more attuned for ensemble approaches. An ensemble of decision trees leading to decision forest has been proposed recently for multi-label classification [34].

3 Multi-label Sparse Representation based Classification

Before we can go into our proposed modifications, we need to understand sparse representation based classification (SRC). We will discuss the base algorithm first followed by the proposed adaptation for multi-label problems. We will start with the standard SRC, following which we will discuss dictionary learnt SRC and finally kernel SRC.

3.1 Sparse Representation based Classification

Sparse Representation based Classification (SRC) [9] assumes that each class can be represented by a subspace, i.e. the training samples and any new test sample should belong to the same sub-space. Another assumption is that there are sufficient training samples in each class to form a basis for representing the class specific sub-space. This allows the test sample v_{test} to be represented in terms of training samples (basis) of the k^{th} class. Formally, this is expressed as,

$$v_{test} = \alpha_{k,1}v_{k,1} + \alpha_{k,2}v_{k,2} + \ldots + \alpha_{k,n_k}v_{k,n_k} + \varepsilon \qquad (1)$$

where, the i^{th} training sample for the k^{th} class is denoted as $v_{k,i}$ and the approximation error as ε.

In single label classification, the training samples and their corresponding identities / labels are known; the task is to assign a (correct) label to the new test sample. In the context of SRC, this requires finding the coefficients $\alpha_{k,i}$ in equation (1). However, since the correct class is not known, we can at best represent the test sample as a linear combination of samples from all classes.

$$v_{test} = V\alpha + \varepsilon \qquad (2)$$

where, $V = \left[\underbrace{v_{1,1}|\ldots|v_{1,n_1}}_{v_1} \underbrace{v_{2,1}|\ldots|v_{2,n_2}}_{v_2} \ldots \underbrace{v_{C,1}|\ldots|v_{C,n_C}}_{v_C} \right]$ and

$$\alpha = \left[\underbrace{\alpha_{1,1},\ldots,\alpha_{1,n_1}}_{\alpha_1} \underbrace{\alpha_{2,1},\ldots,\alpha_{2,n_2}}_{\alpha_2} \ldots \underbrace{\alpha_{C,1},\ldots,\alpha_{C,n_C}}_{\alpha_C} \right]^T$$

The formulations (1) and (2) are equivalent when the α_k's corresponding to the incorrect class are zeroes. While solving for α such a sparsity constraint can be explicitly imposed by regularizing the inverse problem (2) by an l_1-norm [9]; note that l_0-norm can also be used alternately.

$$\min_{\alpha} \left\| v_{test} - V\alpha \right\|_2^2 + \lambda \left\| \alpha \right\|_1 \qquad (3)$$

Once a sparse solution of α is obtained, one needs to use it to determine the identity of the test sample. Various strategies can be proposed here. One can be to evaluate the norm of the α_k's and assign the test sample to the class having the highest norm. But such a strategy is not known to yield the best results. In [9], an analysis via synthesis approach is proposed instead. It synthesizes a representative sample of each class by multiplying V_k with α_k's and then computes the distance between the class-wise representative sample and the test sample. It is likely that for the correct class, the distance will be minimum. Therefore, it is reasonable to assign the test sample to the class having the minimum distance.

```
SRC Algorithm
1. Solve the optimization problem expressed in (3).
2. For each class k repeat the following two steps:
    a. Synthesize representative sample for each class by a linear combination of the training samples belonging to that class by the equation $v_{rep}(k) = V_k \alpha_k$ .
    b. Find the distance between the synthesized class specific samples and the given test sample by $d_k = \left\| v_{test} - v_{rep(k)} \right\|$
3. After computing the class-wise distances, the test sample is assigned to the class having the minimum one.
```

3.2 ML-SRC

We modify SRC to solve multi-label classification problems. The basic assumption remains the same – each class forms a separate sub-space. In the previous single label case, samples could belong to one subspace. However, for the multi-label case, each sample belongs to multiple classes, each sample is ideally represented by a union of subspaces.

The question now is how do we solve the inverse problem (2) so that the inversion allows the test sample to be represented as a union of subspaces? Note that, even though it is a multi-label classification problem, α is sparse since sample typically belongs to only a few classes. Fortunately in this scenario, the l_1-norm minimization can optimally invert (2) preserving the union of subspaces model. Therefore, we do not have to make any change to the basic SRC algorithm so far.

Once the α is solved from (2), we need to assign identities to the test sample. Unlike the single label case, vtest will belong to multiple classes. As before, we follow an analysis via synthesis approach. The representative sample for each class 'k' is obtained by $v_{rep}(k) = V_k \alpha_k$. After that the distances of the test sample from the representative samples will be obtained by $d_k = \left\| v_{test} - v_{rep(k)} \right\|_2^2$. Since, it is a multi-label problem, we have to realize that one test sample can belong to multiple classes. Therefore instead of assigning the test sample to the class having the minimum distance we need to threshold the class-wise distances and assign the test sample to all classes falling under the threshold. For our experiments, we have used a threshold of 2 x min(d_k) and considered all classes falling in this set.

```
ML-SRC Algorithm
1. Solve the optimization problem expressed in (3).
2. For each class k repeat the following two steps:
    a. Compute class-wise representative sample: $v_{rep}(k) = V_k \alpha_k$ .
    b. Find class-wise distance:
       $d_k = \left\| v_{test} - v_{rep(k)} \right\|$
3. Assign test sample to all classes whose distance is less than $2 \times \min(d_k)$ .
```

3.2 ML-KSRC

The basic SRC assumes that the training samples form a linear basis for the test sample. What if the relationship is non-linear? The oldest trick in the book to handle non-linearity is via kernelization.

Several studies independently proposed the Kernel Sparse Representation based Classification (KSRC) approach [10, 11] for single label classification. It generalizes the linear model by assuming that a non-linear projection of the test sample can be expressed as a linear combination of non-linear combination of training samples.

$$\phi(v_{test}) = \phi(V)\alpha + \varepsilon \qquad (4)$$

Here, $\phi(.)$ represents a non-linear transformation (unknown). The simplest way to apply the Kernel trick is to pre-multiply by $\phi(V)^T$ [45]. This leads to:

$$\phi(V)^T \phi(v_{test}) = \phi(V)^T \phi(V)\alpha + \varepsilon \qquad (5)$$

The new form (5) is amenable to the kernel trick as it expresses the problem in terms of inner-products; the trick is to replace the inner-products by Mercer kernels. The kernelized version is represented as,

$$\kappa(V, v_{test}) = \kappa(V, V)\alpha + \varepsilon \qquad (6)$$

where, $\kappa(\cdot) = \langle \cdot, \cdot \rangle$ is the kernel. This encapsulates to the standard SRC form (2). One now employs l_1-norm minimization to solve α. After that one needs to compute the kernel distance between the test sample and the class representations. This is represented as follows,

$$\begin{aligned} d_k &= \left\| \phi(v_{test}) - \phi(V_k)\alpha_k \right\|_2^2 \qquad (7) \\ &= \left(\phi(v_{test}) - \phi(V_k)\alpha_k \right)^T \left(\phi(v_{test}) - \phi(V_k)\alpha_k \right) \\ &= \phi(v_{test})^T \phi(v_{test}) - 2\alpha_k^T \phi(V_k)^T \phi(v_{test}) + \alpha_k^T \phi(V_k)^T \phi(V_k)\alpha_k \\ &= \kappa(v_{test}, v_{test}) - 2\alpha_k^T \kappa(V_k, v_{test}) + \alpha_k^T \kappa(V_k, V_k)\alpha_k \end{aligned}$$

For single label classification problems, the test sample is assigned to the class having the minimum kernel distance d_k.

It is straightforward to modify KSRC to solve multi-label classification problems. We start with the assumption that a non-linear transformed version of the samples from each class lie on a different subspace. Since we do not assume the nature of the non-linearity, we kernelize the test and training samples as in (6). This modifies our original assumption; the kernelized version of samples for each class lie on a separate subspace. Therefore, the kernelized version of test sample, which in our case can belong to multiple classes, can be expressed by the union of subspaces model – each subspace corresponds to a different class.

Since l_1-minimization preserves the union of subspaces structure, we use it to solve (6) to obtain α. After that we compute the kernel distances between the test sample and the class-wise representative samples. The rest of the algorithm remains the same as the multi-label SRC.

```
ML-KSRC Algorithm
1. Solve the optimization problem expressed in (3).
2. For each class k repeat the following two steps:
    a. Compute class-wise representative sample: $v_{rep}(k) = V_k \alpha_k$ .
    b. Find kernelized class-wise distance:
       $d_k = \left\| \phi(v_{test}) - \phi(V_k)\alpha_k \right\|_2^2$ using (7).
3. Assign test sample to all classes whose distance is less than $2 \times \min(d_k)$ .
```

3.4. Dictionary learnt ML-SRC (ML-DSRC)

The SRC approach can yield poor results if there is noise in the training samples, in that case the basis for representing each class will be noisy. To ameliorate this issues, dictionary learning was used to replace the raw training samples by a learnt basis [35], [36]. This was done for each class by the K-SVD algorithm [37]. This is represented as follows,

$$D_k \leftarrow \min_{D_k, Z_k} \left\| V_k - D_k Z_k \right\|_F^2 \text{ such that } \left\| Z_k \right\|_0 \le \tau \ \forall k \qquad (7)$$

During dictionary learning, the noise is removed and the learnt dictionaries D_k's are clean basis for representing the test sample.

$$v_{test} = D\alpha + \varepsilon \text{ where } D = \left[D_1 \,|\, D_2 \,|\, \ldots \,|\, D_C \right] \qquad (8)$$

The rest of the algorithm remains the same as SRC. The only difference is that, the classwise representations are computed as $v_{rep}(k) = D_k \alpha_k$. The dictionary learning SRC (DLSRC) was proposed for single label classification problems. As we have been doing so far, we can adopt it for multi-label classification problems using the same strategy as before.

4 Experimental Evaluation

We carry out extensive evaluation on several benchmark multi-label classification datasets.

- Yeast – This biological dataset [22] is concerned with protein function classification.
- Scene – This image dataset [5] is concerned with semantic indexing of still scenes.
- TMC2007 – The textual dataset [38] concerns aviation safety reports.
- Mediamill – This challenge dataset [39] was created for automatic annotation of semantic concepts in multimedia.
- Bibtex – This dataset [4] concerns text classification for automatic tagging.

We have benchmarked our technique with the two standard – MLKNN [15] and RaKEL [32]; and three state-of-the-art – generalized K label-sets (GeKEL) [24] PruDent [27], and ML-Forest [34] techniques.

The performance is compared in terms of three measures – hamming loss (HL), one error (OE) and average precision (AP). The mathematical definitions are standard in this area [5], [6], [25]-[27] etc. and hence we are not repeating them. We only discuss their intuitive meanings.

- Hamming loss: evaluates how many times an instance–label pair is misclassified, i.e. a label not belonging to the instance is predicted or a label belonging to the instance is not predicted. The performance is perfect when the loss is 0; the smaller the value of the loss, the better the performance.
- One-error: evaluates how many times the top-ranked label is not in the set of proper labels of the instance. The performance is perfect when the error is 0; the smaller the value of error, the better the performance.
- Average precision: evaluates the average fraction of labels ranked above a particular label which should actually be there. The performance is perfect when precision is 1; the bigger the value of precision, the better the performance.

TABLE 1
Results on Yeast Dataset

	ML-SRC	ML-KSRC	ML-DSRC	ML-KNN	RaKEL	GeKEL	PruDent	ML-Forest
HL	.189	.151	.178	.195	.195	.212	.243	.199
OE	.218	.166	.192	.235	.234	.253	.216	.241
AP	.781	.857	.804	.762	.762	.741	.720	.754

TABLE 2
Results on Scene Dataset

	ML-SRC	ML-KSRC	ML-DSRC	ML-KNN	RaKEL	GeKEL	PruDent	ML-Forest
HL	.161	.154	.158	.169	.195	.199	.174	.097
OE	.192	.189	.192	.235	.234	.246	.239	.248
AP	.804	.802	.804	.762	.762	.759	.685	.835

TABLE 3
Results on TMC-2007 Dataset

	ML-SRC	ML-KSRC	ML-DSRC	ML-KNN	RaKEL	GeKEL	PruDent	ML-Forest
HL	.182	.182	.186	.201	.193	.217	.184	.060
OE	.212	.207	.218	.236	.225	.232	.232	.117
AP	.794	.796	.790	.781	.786	.763	.708	.879

TABLE 4
Results on Mediamill

	ML-SRC	ML-KSRC	ML-DSRC	ML-KNN	RaKEL	GeKEL	PruDent	ML-Forest
HL	.177	.172	.175	.182	.178	.180	.173	-
OE	.254	.249	.247	.264	.253	.261	.268	-
AP	.764	.781	.791	.751	.763	.758	.741	-

TABLE 5
Results on Bibtex

	ML-SRC	ML-KSRC	ML-DSRC	ML-KNN	RaKEL	GeKEL	PruDent	ML-Forest
HL	.164	.161	.155	.176	.153	.159	.154	.012
OE	.213	.208	.208	.232	.204	.204	.223	.346
AP	.812	.822	.818	.792	.821	.819	.799	.607

Tables 1 to 5 summarize the results of the proposed approaches (ML-SRC, ML-KSRC, and ML-DSRC) and show the comparison with existing algorithms. Generally, performance of the proposed algorithms is among the top three results. For the large databases such as Mediamill and TMC2007, the proposed algorithms yield low Hamming loss and high average precision. Further, compared to some published results, the proposed algorithm yields improved results. For example, on the Yeast database, the proposed algorithm shows superior performance compared to [41], [42], [43] both in terms of Hamming loss and average precision. Additionally, we have experimented with Yahoo database [44], which consists of 11 sub databases. For all the sub databases, we obtained state of the art performance with lowest Hamming distance. The average Hamming distance obtained from the proposed algorithm across all the sub databases is 0.0418 and the next best average of an existing algorithm is 0.0429.

The proposed approaches require solving an l_1-minimization problem. This is achieved via the SPGLI solver [40]. This is fairly non-parametric and it only requires specification of λ. We found that the proposed algorithm is resilient to the value of this parameter. The strategy for fixing the threshold has already been discussed while describing the algorithms. For the kernelized version, radial basis function (RBF) yields be best results. Computationally, during training, separate dictionary is trained for each label and based on the reconstruction from each label, we predict the presence or absence of a label. Therefore, the proposed algorithm is fast and does not require heavy computational resources such as GPU.

7 Conclusion

In this paper, we have presented a novel algorithm that modifies the sparse representation based classification approach to address multi-label classification problems. Experiments on standard datasets, with benchmark techniques show that the proposed method, at an average, performs better than others.

There are some recent deep learning based techniques that have been proposed for multi-label classification. We have not compared against them. This is largely because such techniques only work on problems with large scale training databases; on the other hand, for many applications and databases, the availability of large training data is a challenge and we have to train the models with limited training samples. Therefore, it would not be fair to compare with deep learning algorithms. In the future, we intend to perform evaluation on large scale datasets and compare with modern deep learning based multi-label classification techniques.

REFERENCES

[1] K. Trohidis, G. Tsoumakas, G. Kalliris and I. P. Vlahavas. "Multi-label classification of music into emotions," International Society for Music Information Retrieval Conference, vol. 8, pp. 325-330, 2008.

[2] K. Trohidis, G. Tsoumakas, G. Kalliris and I. P. Vlahavas. "Multi-label classification of music by emotion," EURASIP Journal on Audio, Speech, and Music Processing, vol. 1, p.4, 2011.

[3] S.M. Liu and J.H. Chen, "A multi-label classification based approach for sentiment classification," Expert Systems with Applications, vol. 42, no.3, pp.1083-1093, 2015.

[4] I. Katakis, G. Tsoumakas and I. Vlahavas, "Multilabel text classification for automated tag suggestion," European Conference on Machine Learning/ The Pacific-Asia Conference on Knowledge Discovery and Data Mining, vol. 18, 2008.

[5] 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.

[6] Z.H. Zhou and M.L. Zhang, "Multi-instance multi-label learning with application to scene classification," Neural Inforamtion Processing Systems, pp. 1609-1616, 2007.

[7] K. Basu, V. Debusschere, S. Bacha, U. Maulik and S. Bondyopadhyay, "Nonintrusive Load Monitoring: A Temporal Multilabel Classification Approach," IEEE Transactions on Industrial Informatics, vol. 11, no. 1, pp. 262-270, 2015.

[8] S. M. Tabatabaei, S. Dick and W. Xu, "Toward Non-Intrusive Load Monitoring via Multi-Label Classification," IEEE Transactions on Smart Grid, vol. 8, no. 1, pp. 26-40, 2017.

[9] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry and Y. Ma, "Robust Face Recognition via Sparse Representation," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 31, no. 2, pp. 210-227, 2009.

[10] L. Zhang et al., "Kernel Sparse Representation-Based Classifier," in IEEE Transactions on Signal Processing, vol. 60, no. 4, pp. 1684-1695, April 2012.

[11] J. Yin, Z. Liu, Z. Jin and W. Yang, "Kernel sparse representation based classification," Neurocomputing," vol. 77, no. 1, pp. 120-128, 2012.

[12] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," International Journal of Data Warehousing and Mining, vol. 3, no. 3, pp. 1-13, 2007.

[13] G. Madjarov, D. Kocev, D. Gjorgjevikj and S. Džeroski, "An extensive experimental comparison of methods for multi-label learning," Pattern Recognition, vol. 45, no. 9, pp. 3084-3104, 2012.

[14] R.E. Schapir and Y. Singer, "Boostexter: a boosting-based system for text categorization," Machine Learning, vol. 39, pp. 135–168, 2000.

[15] M.L. Zhang and Z.H. Zhou, "ML-kNN: a lazy learning approach to multi-label learning," Pattern Recognition, vol. 40 (7), pp. 2038–2048, 2007.

[16] M.-L. Zhang and Z.H. Zhou, "Multilabel Neural Networks with Applications to Functional Genomics and Text Categorization," IEEE Transactions on Knowledge and Data Engineering, vol. 18, no. 10, pp. 1338-1351, 2006.

[17] C.K. Yeh, W.C. Wu, W.J. Ko and Y.C.F. Wang, "Learning Deep Latent Space for Multi-Label Classification," AAAI Conference on Artificial Intelligence, pp. 2838-2844, 2017.

[18] J. Wicker, A. Tyukin and S. Kramer, "A nonlinear label compression and transformation method for multi-label classification using autoencoders," The Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 328-340, 2016.

[19] Y. Wei, W. Xia, M. Lin, J. Huang, B. Ni, J. Dong, Y. Zhao and S. Yan, "HCP: A Flexible CNN Framework for Multi-Label Image Classification," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 9, pp. 1901-1907, 2016.

[20] F. Wu, Z. Wang, Z. Zhang, Y. Yang, J. Luo, W. Zhu, Y. Zhuang, "Weakly Semi-Supervised Deep Learning for Multi-Label Image Annotation," IEEE Transactions on Big Data, vol. 1, no. 3, pp. 109-122, 2015.

[21] A. Clare and R.D. King, "Knowledge discovery in multi-label phenotype data," European Conference on Machine Learning/ The Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 42–53, 2001.

[22] A. Elisseeff and J. Weston, "A Kernel method for multi-labelled classification," International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 274–281, 2005.

[23] J. Read, B. Pfahringer and G. Holmes, "Multi-label classification using ensembles of pruned sets," International Conference on Data Mining, pp. 995–1000, 2008.

[24] H. Lo, S. Lin and H. Wang, "Generalizedk-Labelsets Ensemble for Multi-Label and Cost-Sensitive Classification," IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 7, pp. 1679-1691, 2014.

[25] G. Tsoumakas, I. Katakis and I. Vlahavas, "Effective and efficient multilabel classification in domains with large number of labels," ECML/PKDD Workshop on Mining Multidimensional Data, pp. 30–44, 2008.

[26] M. Zhang and Z. Zhou, "A Review on Multi-Label Learning Algorithms," IEEE Transactions on Knowledge and Data Engineering, vol. 26, no. 8, pp. 1819-1837, 2014.

[27] A. Alali and M. Kubat, "PruDent: A Pruned and Confident Stacking Approach for Multi-Label Classification," IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 9, pp. 2480-2493, 2015.

[28] X. Li and Y. Guo, "Active Learning with Multi-Label SVM Classification," Intenatinal Joint Conference on Aritificial Intelligence, pp. 1479-1485, 2013.

[29] J. Furnkranz, "Round robin classification," Journal of Machine Learning Research, vol. 2, pp. 721–747, 2002.

[30] T.-F. Wu, C.-J. Lin and R.C. Weng, "Probability estimates for multiclass classification by pairwise coupling," Journal of Machine Learning Research, vol. 5, 975–1005, 2005.

[31] S.-H. Park and J. Furnkranz, "Efficient pairwise classification," European Conference on Machine Learning/ The Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 658–665, 2007.

[32] 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, 2011.

[33] J. Read, B. Pfahringer, G. Holmes and E. Frank, "Classifier chains for multi-label classification," European Conference on Machine Learning/ The Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 254–269, 2009.

[34] Q. Wu, M. Tan, H. Song, J. Chen and M. K. Ng, "ML-FOREST: A Multi-Label Tree Ensemble Method for Multi-Label Classification," IEEE Transactions on Knowledge and Data Engineering, vol. 28, no. 10, pp. 2665-2680, 2016.

[35] M. Yang, L. Zhang, J. Yang and D. Zhang, "Metaface learning for sparse representation based face recognition," Intenrational Conference on Image Processing, pp. 1601-1604, 2010.

[36] T. Guha and R. K. Ward, "Learning Sparse Representations for Human Action Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 8, pp. 1576-1588, 2012.

[37] M. Aharon, M. Elad and A. Bruckstein, "*rmK*-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation," IEEE Transactions on Signal Processing, vol. 54, no. 11, pp. 4311-4322, 2006.

[38] N. Ghamrawi and A. McCallum, "Collective multi-label classification," Conference on Information and Knowledge Management, pp. 195-200, 2005.

[39] C.G.M. Snoek, M. Worring, J.C. van Gemert, J.-M. Geusebroek, and A.W.M. Smeulders, "The Challenge Problem for Automated Detection of 101 Semantic Concepts in Multimedia," ACM Multimedia, pp. 421-430, 2006.

[40] M. Friedlander and E. Van den Berg, "SPGL1, a solver for large scale sparse reconstruction," SIAM Journal on Scientific Computing, vol. 31, 2, pp.890-912, 2008.

[41] 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.

[42] G. Wu, Y. Tian, and D. Liu, "Cost-sensitive multi-label learning with positive and negative label pairwise correlations," Neural Networks, vol. 108, pp. 411-423, 2018.

[43] W. Weng, Y. Lin, S. Wu, Y. Li, and Y. Kang, "Multi-label learning based on label-specific features and local pairwise label correlation," Neurocomputing, vol. 273, pp. 385-394, 2018.

[44] N. Ueda and K. Saito, "Parametric mixture models for multi-labeled text," Neural information processing systems, pp. 737-744, 2002.

[45] G. Goswami, P. Mittal, A. Majumdar, M. Vatsa, and R. Singh,"Group sparse representation based classification for multi-feature multimodal biometrics," Information Fusion, vol 32, Part B, pp. 3-12, 2016,