International Journal of Data Science and Analysis
2022; 8(2): 30-37
http://www.sciencepublishinggroup.com/j/ijdsa
doi: 10.11648/j.ijdsa.20220802.13
ISSN: 2575-1883 (Print); ISSN: 2575-1891 (Online)

An Application of Multi-label Linear Discriminant Analysis and Binary Relevance K-Nearest Neighbor in Multi-label Classification of Annotated Images

Festus Malombe Mwinzi[1], Thomas Mageto[1], Victor Muthama[2]

[1]Department of Statistics and Actuarial Sciences, Jomo Kenyatta University of Agriculture and Technology, Nairobi, Kenya
[2]School of Pure and Applied Sciences, Kirinyaga University, Kirinyaga, Kenya

Email address:
mwinzimfestus@gmail.com (F. M. Mwinzi), tttmageto@gmail.com (T. Mageto), vmuthama2@gmail.com (V. Muthama)

To cite this article:
Festus Malombe Mwinzi, Thomas Mageto, Victor Muthama. An Application of Multi-label Linear Discriminant Analysis and Binary Relevance K-Nearest Neighbor in Multi-label Classification of Annotated Images. *International Journal of Data Science and Analysis*. Vol. 8, No. 2, 2022, pp. 30-37. doi: 10.11648/j.ijdsa.20220802.13

Received: March 4, 2022; **Accepted**: March 24, 2022; **Published**: March 31, 2022

Abstract: Although Binary Relevance (BR) is an adaptive and conceptually simple multi-label learning technique, its inability to exploit label dependencies and other inherent problems in multi-label examples makes it difficult to generalize well in the classification of real-world multi-label examples like annotated images. Thus, to strengthen the generalization ability of Binary Relevance, this study used Multi-label Linear Discriminant Analysis (MLDA) as a preprocessing technique to take care of the label dependencies, the curse of dimensionality, and label over counting inherent in multi-labeled images. After that, Binary Relevance with K Nearest Neighbor as the base learner was fitted and its classification performance was evaluated on randomly selected 1000 images with a label cardinality of 2.149 of the five most frequent categories, namely; "person", "chair", "bottle", "dining table" and "cup" in the Microsoft Common Objects in Context 2017 (MS COCO 2017) dataset. Experimental results showed that micro averages of precision, recall, and f1-score of Multi-label Linear Discriminant Analysis followed by Binary Relevance K Nearest Neighbor (MLDA-BRKNN) achieved a more than 30% improvement in classification of the 1000 annotated images in the dataset when compared with the micro averages of precision, recall, and f1-score of Binary Relevance K Nearest Neighbor (BRKNN), which was used as the reference classifier method in this study.

Keywords: Binary Relevance, K-Nearest Neighbor, Binary Relevance K-Nearest Neighbor (BRKNN), Multi-label Linear Discriminant Analysis (MLDA)

1. Introduction

Advancements in digital technologies such as high-resolution photography devices and large multimedia databases have seen an enormous increase in the volume of digital images and videos all over the world, making image annotation and classification active research topics [1].

An image in a typical image annotation problem will consist several objects, each associated with multiple different conceptual classes [2]. Such a scenario in machine learning is referred to as a multi-label classification task. Among the many multi-label classification algorithms, Binary Relevance (BR) is considered as an instinctual solution to learn from multi-label instances due to its conceptual simplicity and ability to take any binary learning method as a base learner [3].

However, BR becomes complex and computationally expensive in the case of annotated images which are always highly dimensional [4]. In addition, its inability to exploit label correlation contradicts multi-label learning assumption that labels correlation must be well exploited alongside other inherent properties of multi-label problems so as to build a multi-label prediction model with excellent generalization capability.

Therefore, to enhance the generalization ability of BR in multi-label classification of annotated images, this study uses multi-label linear discriminant analysis (MLDA) which is a development of linear discriminant analysis as a preprocessing step to take care of the curse of dimensionality, label dependencies, and label over-counting inherent in multi-label examples. After which, Binary Relevance with K Nearest Neighbor as the base learner is fitted and its classification performance evaluated on randomly selected 1000 images of the MS COCO 2017 dataset [5].

2. Literature Review

Unlike the single-label problems, multi-label problems are associated with more than one class at a time making classification inevitable [6]. Multi-label classification techniques are divided into ensemble-based methods, algorithm adaptation (AA) and problem transformation (PT) [7]. PT algorithms such as BR break down a multi-label classification task into a sequence of independent single-label classification tasks that can be solved using various binary classification algorithm.

Santos compared classification performance of BR and Label Powerset (LP) on scene categorization using five traditional learning algorithms as the base classifier [8]. Experimental results indicated that LP with support vector machine as the base classifier achieved better classification accuracy than the counterparts. In 2016, Aldrees and Chikh investigated the multi-label classification performance of Binary Relevance One-vs.-one, Label Powerset, and Multi-label K Nearest Neighbors (MLKNN) on music and emotion datasets [9]. MLKNN performed poorly.

In 2018, Zhang examined BR from three perspectives, namely; the primary setting of binary relevance, certain typical correlation-enabling extensions, and associated class-imbalance problems. He suggests that adequate BR learning methods that are in a position of modeling multi-label complexities as well as label-specific features be developed to identify distinct properties of class-wise label [4].

Multi-label learning, like many other machine learning problems, ails from the curse of dimensionality [7]. To reduce the large dimensions of multi-label data, Wang developed a generalized linear discriminant analysis dubbed Multi Label Linear Discriminant Analysis [2]. In 2016, Wang et al. extended Multi Label Linear Discriminant Analysis by normalizing label dependencies to correct over-counting of instances with multiple labels [10]. Extensive experimental evaluations demonstrated promising discriminative capability of MLDA.

As presented in the above reviewed literatures, many studies have investigated binary relevance as a multi-label learning technique, however its weaknesses harbor its generalization ability. Thus, this study utilizes multi-label linear discriminant analysis which has proven to be a good discriminant in multi-label problems as a preprocessing technique to address most of the short coming of binary relevance in multi-label classification.

3. Methodology

3.1. Multi-label Linear Discriminant Analysis

Wang et al. defines Multi-label Linear Discriminant Analysis (MLDA) as a natural extension of standard Linear Discriminant Analysis (LDA) for multi-label problems [2]. In contrast to LDA, MLDA incorporates class label correlations in the definition of scatter matrices which are computed from a class-wise perspective rather than from data point perspective.

3.1.1. Multi-label Data

Let $\{x_i, y_i\}_i^N$ be a multi-label dataset portioned into K classes as $\{\pi_k\}_{k=1}^K$ where; π_k represents the sample set of class k with n_k instances, $x_i \in \mathbb{R}^d$, $y_i \in \{0,1\}^K$, $y_i(k) = 1$ if x_i is a member of k^{th} class, and 0 otherwise. In general, $X = [x_1, \dots, x_N]$ and $\gamma = [y_1, \dots, y_N]^T = [y(1), \dots, y(k)]$ where, $y_i \in \{0,1\}^{n_k}$ is the indication vector for the k^{th} class.

3.1.2. Measures of Multi-label Data

The measures of multi-label data are; the number of instances (N) and the number of features (d) in the input space $\{x_i, y_i\}_i^N$, the number of classes (K) and label distribution which is characterized by label cardinality and label density [11].

Label cardinality denoted as LCard, is a measure of multi-labelled-ness [12]; that is, the how many labels on average are associated with each data point.

$$LCard = \frac{\sum_{i=1}^{N} |y_i|}{N} \tag{1}$$

On the other hand, label density denoted as LDens refers to cardinality divided by the size of label set.

$$LDens = \frac{1}{K} Lcard \tag{2}$$

3.1.3. MLDA Algorithm

Step 1: Multi-Label correlations

Wang et al. defines label correlation between two classes as [2];

$$C_{kl} = \frac{<y_k, y_l>}{||y_k||\ ||y_l||}\ for\ l, k\ \in 1, \dots K \tag{3}$$

Where $C \in \mathbb{R}^{K \times K}$ is a symmetric matrix, y_k and y_l are l^{th} and k^{th} class label vectors, $||.||$ is the l_2 norm and $< y_k, y_l >$ is the inner product between label vector k and l. Then C is normalized so as to correct the number of times an instance x_i with multiple labels is used in the scatter matrices via equation 4

$$p_{ik} = \frac{y_i C}{||y_i||_{l1}} \tag{4}$$

Where, p_{ik} is the correlated normalized weight factor for the i^{th} instance in k^{th} class and $\sum_{k=1}^{K} p_{ik} \geq 1$.

Step 2: d-dimensional mean vectors

Mean vector of k^{th} class denoted as μ_k is computed as

$$\mu_k = \frac{\sum_{i=1}^{n_k} p_{ik} x_i}{\sum_{i=1}^{n_k} p_{ik}} \tag{5}$$

Multi-label global mean vector denoted as μ is computed as

$$\mu = \frac{\sum_{k=1}^{K}\sum_{i=1}^{N} p_{ik}x_i}{\sum_{k=1}^{K}\sum_{i=1}^{N} p_{ik}} \quad (6)$$

Step 3: Class-wise scatter matrices

Class-wise between-class scatter matrix (S_B)

$$S_B = \sum_{k=1}^{K}(\sum_{i=1}^{N} p_{ik})\,(\mu_k - \mu)(\mu_k - \mu)^T \quad (7)$$

Class-wise within-class scatter matrix (S_W)

$$S_W = \sum_{k=1}^{K}\sum_{i=1}^{n_k} p_{ik}\,(x_i - \mu_k)(x_i - \mu_k)^T \quad (8)$$

Step 4: Eigenvalues and Eigenvectors

Compute the eigenvalues and associated eigenvectors of the matrix S

$$S = S_W^{-1}\, S_B \quad (9)$$

Group the realized eigenvectors by decreasing eigenvalues and choose $c\ (K\ -1)$ eigenvectors whose eigenvalues are the largest to form a $(d\ \times\ c)$-dimensional matrix G.

Step 5: New feature subspace

Use matrix G to transform the original feature space $X \in\ \mathbb{R}^{N\times d}$onto a new feature subspace $Q \in \mathbb{R}^{N\times c}$ for $c < d$ as given by equation 10.

$$Q = X \times G \quad (10)$$

3.2. Binary Relevance K-Nearest Neighbor

Binary Relevance K Nearest Neighbor (BRKNN) is a multi-label learning algorithm that adapts the K Nearest Neighbor (KNN) algorithm to categorize multi-label instances [13]. In particular, BRKNN extends the KNN to produce independent predictions after a single search of the nearest neighbors [14]. Furthermore, Zhang et al. claims that BRKNN is theoretically analogous to combining BR with the KNN method [15].

3.2.1. Binary Relevance

Binary Relevance refers to a problem transformation method that breaks down a multi-label learning problem say $Q\ =\ \{x_i, y_i\}_{i=1}^{N}$ into m independent single label problems say Q_j [3]. Where $x_i\ \in\ \mathbb{R}^c,\ y_i$ denotes the corresponding label vector for instance x_i and $j\ = 1,\cdot\cdot\cdot, m$.

Each of these m single label problems correspond an individual class label in the label space. More specifically, for every class label β_j, BR constructs a corresponding single label dataset Q_j from the original multi-label dataset Q by taking into account the relevance of each instance to β_j as follows [16];

$$\mathrm{Q}_j\ =\ \{(x_i, f(y_i,\ \beta_j))|1\ \le j\ \le m\} \quad (11)$$

Where, $f(y_i, \beta_j) = \begin{cases} 1, if\ \beta_j \in y_i \\ 0, otherwise \end{cases}$

Table 1. *Multi-label dataset example.*

Instance	Label Set
1	$\{\beta_2, \beta_4\}$
2	$\{\beta_1, \beta_2, \beta_3\}$
3	$\{\beta_1\}$
4	$\{\beta_2, \beta_3\}$

Table 2. *Single-label dataset produced by BR.*

Instance	Label (β_1)	Instance	Label (β_2)	Instance	Label (β_3)	Instance	Label (β_4)
1	0	1	1	1	0	1	1
2	1	2	1	2	1	2	0
3	1	3	0	3	0	3	0
4	0	4	1	4	1	4	0

Then, on each of the m resultant single-label datasets, a binary classifier is trained. BR has the following properties;

1. Binary relevance is conceptually simple since it is a first-order technique that builds classification models in a label-wise manner ignoring the coexistence between class labels. The difficulty of binary relevance modeling is proportional to the size of label space [17].
2. BR being a problem transformation method, is not limited to any specific binary learning technique. Thus, it can be initiated with any binary classifier with multiple properties [4].
3. BR quickly learns from multi-label cases with labels that are missing as a result of high labeling costs and human labelers' negligence [18].

3.2.2. K-Nearest Neighbors

K Nearest Neighbors (KNN) is a memory-based supervised machine learning method used for classification and/or regression tasks [4]. Wu et al. regards KNN as one of the top data mining algorithms and attributes this fact to the following properties [19];

1. KNN is non-parametric meaning it does not assume any probability distribution on the input data. This comes in handy for inputs whose probability distributions are unknown and thus makes KNN more robust.
2. KNN quickly adapts to changes in the input data by employing lazing learning which generalizes during testing. That is, the model has no learning phase and all work is done when prediction is sought.

i. K-Nearest Neighbor Classifier

As a classifier, KNN saves all the available examples and classifies new cases based on a distance function as follows [20];

Step 1: Compute the distance metric

Compute the distance between the test instance and all the training examples in new subspace Q using various types of distance metrics. The Euclidean distance (equation 13) defined as the square root of the sum of difference between the test instance and the training instance is used in this study.

Step 2: Sort training instances according to evaluated distance

The K-NN classifier then sorts the training instances x_i according to the evaluated Euclidean distance in ascending order and selects $U_{x_i'} \sqsubseteq U$ the set of K closest training instances to the test instance $x_i^{'}$.

Step 3: Classify according to majority vote

Finally, KNN classifies $x_i^{'}$ into to the class with majority of its K x_i as follows;

$$y_i' = argmax\ _v \sum_{(x_i,\pi_i)\in U_{x_i'}} I(v = y_i) \quad (12)$$

However, KNN as a classifier has two open issues to be addressed namely; the similarity measure between the test and training instances, and choice of an optimal K value [21].

ii. Parameter selection

a. Distance metric

Different KNN applications require different distance measures [22]. However, the most commonly used distance metric is Euclidean distance [23]. It is evaluated as in equation 13.

$$d(x_i^{'}, x_i) = \sqrt{\sum_{i=1}(x_i^{'} - x_i)^2} \quad (13)$$

Where, $x_i^{'}$ is the test instance and x_i denotes all the training instances in the feature space. Other distance metrics includes the Manhattan distance, the Lagrange distance and the cosine coefficient [24].

b. Optimal value of K

The parameter K in the KNN refers to the number of nearest neighbors to the instance being categorized. Several approaches have been proposed for finding an appropriate K value. One of the most frequent methods cross validate distinct values of K and keep the K with the lowest classification error rate [24].

3.3. Multi-label Classification Metrics

In multi-label classification the main interest is on how well all the labels can be predicted rather than a single one [25]; that is, how the results of m binary problems can be averaged to a single measurement value based on a contingency table of predicting behavior for binary classification.

Table 3. *Contingency table for binary classification.*

Predicted label

		Positive	Negative
Actual label	Positive	True Positive	False Negative
	Negative	False Positive	True Negative

From the above table the following metrics were evaluated;

$$Micro\ average\ precision = \sum_{i=1}^{m} \frac{TP_i}{TP_i + FP_i} \quad (14)$$

$$Micro\ average\ recall = \sum_{i=1}^{m} \frac{TP_i}{TP_i + FN_i} \quad (15)$$

$$Micro\ average\ F1\ score = \frac{2(Micro\ average\ precision \times Micro\ average\ recall)}{(Micro\ average\ precision + Micro\ average\ recall)} \quad (16)$$

Where, precision and recall refers to the ratio of predicted correct labels to the total number of positive predictions and ratio of predicted correct labels to the total number of actual positives respectively, and F1 score is the harmonic mean of precision and recall.

4. Results

4.1. Label Distribution

This study was experimented on MS COCO 2017 dataset. Table 4 summarizes the label statistics of this dataset.

Table 4. *Dataset label statistics.*

Dataset	Instances	Features	Labels	Cardinality	Density
MS COCO 2017	1000	727	5	2.149	0.4298

As shown in Table 4, the MS COCO 2017 dataset had a total number of 1000 examples, labeled in up to 5 classes and each example was described by 727 features. Each of these 1000 examples was associated with had an average of more than 2 labels (Cardinality = 2.149) and each label had an average occurrence of 42.98% in the dataset (Density = 0.4298).

In addition, the frequency distribution of the five selected labels in the MS COCO 2017 datasets is as shown in Figure 1.

From Figure 1 it is clear that the most frequent label is “Person” and the least frequent label is “Bottle”.

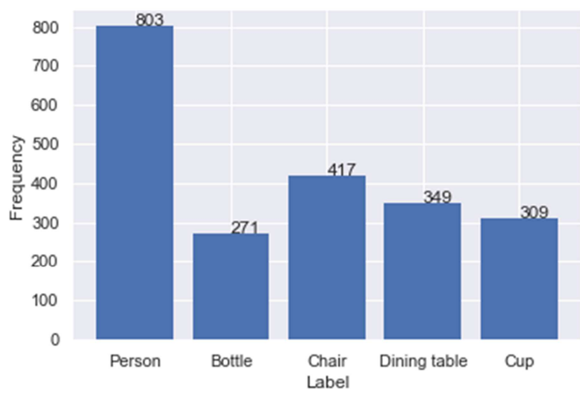

Figure 1. *MS COCO 2017 labels’ frequency distribution.*

4.2. Label Correlations

After examination of the data set' label distribution, the study then explored the label correlations in the dataset using cosine similarity scores. Figure 2 is the cosine similarity score heat map for the labels in the dataset.

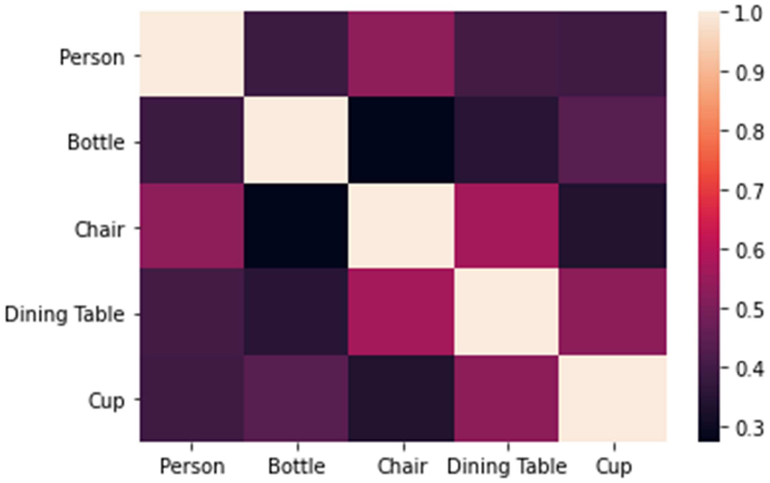

Figure 2. *Cosine similarity heat map of the MS COCO 2017 dataset.*

From Figure 2, the cosine similarity score between "Chair" and "Dining table" is 0.57 which implies that there is a 57% overlapping between these two labels as illustrated below.

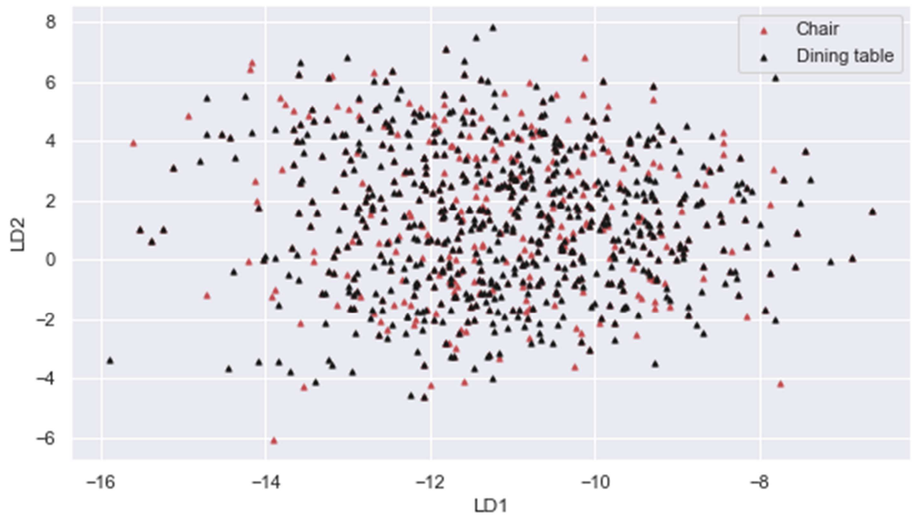

Figure 3. *Scatter plot of simulated labels.*

From Figure 3 above, the data points of "Chair" and "Dining table" are mixed up making it difficult to determine a linear decision boundary between them. Thus, this study incorporated cosine similarity scores in Figure 2 in the calculation of scatter matrices defined in equations 7-8.

4.3. Discriminative Capability of MLDA

After exploration of the label correlations, the study run Multi-label Linear Discriminant Analysis algorithm formulated in section 3.1.3 on MS COCO 2017 dataset. The algorithm transformed the feature space of the dataset from 727 features to a new feature subspace with 4 features as shown by the two tables below.

Table 5. *MS COCO 2017 dataset features space.*

Id	Att1	Att2	Att3	Att725	Att726	Att727
1	0.00627	0.00752	0.00376	0.66291	0.684210	0.68672
2	0.47368	0.47870	0.48622	0.78133	0.781955	0.79449
3	0.00877	0.01003	0.01128	0.89975	0.879700	0.90226
4	0.00000	0.00376	0.01504	0.83208	0.829570	0.85965
5	0.09398	0.09023	0.11278	0.94110	0.923560	0.93734

Table 6. *MS COCO 2017 new features subspace produced by MLDA.*

Id	1	2	3	4
1	0.548931	0.347216	0.592742	0.183433
2	0.482971	0.687382	0.562128	0.316599
3	0.566009	0.534774	0.507913	0.281472
4	0.486544	0.559246	0.333150	0.287352
5	0.479095	0.578148	0.397271	0.393479

Then the study at random selected two classes and visualized their corresponding data points in the new feature subspace onto a 2-dimensional plane of the first two multi-label linear discriminants as shown in Figure 4.

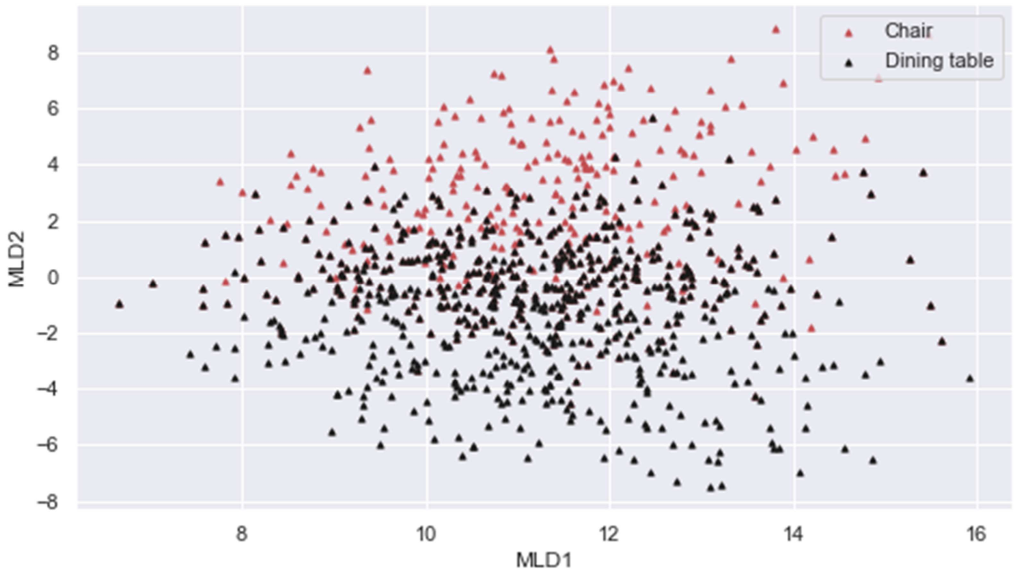

Figure 4. *MS COCO 2017 labels on 2D plane in reduced subspace by MLDA.*

As demonstrated in Figure 4, the data points of the "chair" and "dining table" were clearly segregated with a cosine similarity of 0.21 based on their class membership along the second multi-label linear discriminant.

4.4. Binary Relevance K-Nearest Neighbor

After preprocessing of the dataset, the study normalized the resultant feature subspace so as to give same importance to all features. Then those feature were split at random into training and testing set in the ratio 70:30.

Thereafter, the study employed BR which transformed the MS COCO 2017 datasets in the new feature subspace into 5 independent single label datasets according to equation 11 and as shown in Tables 7 and 8.

Table 7. *MS COCO 2017 dataset labels.*

Image Id	Person	Bottle	Chair	Dining table	Cup
5	1	1	0	0	0
6	1	0	1	0	0
8	1	1	1	1	1
10	1	0	0	0	1
15	1	0	0	0	0

Table 8. *Single label datasets of Scene dataset produced by BR.*

Image Id	Person	Image Id	Bottle	Image Id	Chair	Image Id	Dining table	Image Id	Cup
5	1	5	1	5	0	5	0	5	0
6	1	6	0	6	1	6	0	6	0
8	1	8	1	8	1	8	1	8	1
10	1	10	0	10	0	10	0	10	1
15	1	15	0	15	0	15	0	15	0

Then on each of these 5 independent single label datasets, K Nearest Neighbor was employed as the base classifier with Euclidean distance to decide the neighborhood in the KNN.

4.5. Classification Performance

To evaluate the performance of MLDA-BRKNN on multi-label classification of the 1000 randomly selected annotated images in the MS COCO 2017 dataset, micro averages of precision, recall and f1-score were reported to account for class imbalances as shown in Table 8 with BRKNN as a reference classifier for this study.

Table 9. *Performance evaluation of BRKNN and MLDA-BRKNN.*

Method	Micro avg precision	Micro avg recall	Micro avg f1- score
BRKNN	0.56	0.52	0.54
MLDA-BRKNN	0.87	0.91	0.89

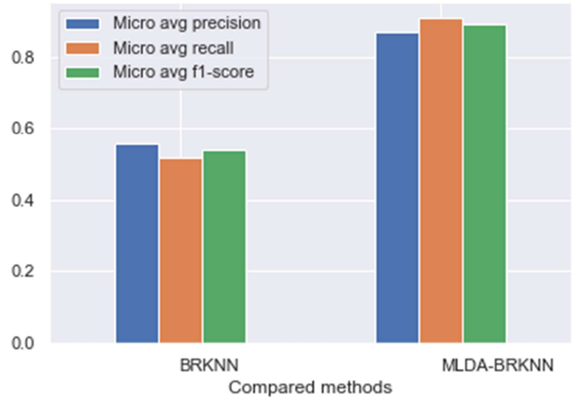

Figure 5. *Classification performance of BRKNN and MLDA-BRKNN.*

From Figure 5, MLDA-BRKNN achieved a more than 30% improvement in classification of MS COCO examples when compared to BRKNN which is used as the reference method in this study.

5. Conclusion

In this study, MLDA is presented as a preprocessing technique to address Binary Relevance short comings of inability to exploit label correlations, computational complexities in case of high dimensional problems and label over counting. Thereafter, a multi-label learning model is built using Binary Relevance with K Nearest Neighbors as the base learner. Experimental evaluations of the Binary Relevance K Nearest Neighbors with prior preprocessing by Multi-label Linear discriminant Analysis (MLDA-BRKNN) on the randomly selected 1000 annotated images of MS COCO 2017 dataset achieved a more than 30% improvement than when classified using only Binary Relevance K Nearest Neighbors (BRKNN).

Future studies should explore the performance of other base classifiers on Binary Relevance with Multi-label Linear Discriminant Analysis as the preprocessing technique in classification of annotated images. Also, other standard and non-standard classification experiments including transfer learning and deep learning should be conducted to explore the label correlations in annotated images.

References

[1] B. M. Brhanie, Multi-Label Classification Methods for Image Annotation, 2016.

[2] H. Wang, C. Ding and H. Huang, "Multi-label linear discriminant analysis," in European conference on computer vision, 2010.

[3] M. S. Sorower, "A literature survey on algorithms for multi-label learning," Oregon State University, Corvallis, vol. 18, p. 1–25, 2010.

[4] M.-L. Zhang, Y.-K. Li, X.-Y. Liu and X. Geng, "Binary relevance for multi-label learning: an overview," Frontiers of Computer Science, vol. 12, p. 191–202, 2018.

[5] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár and C. L. Zitnick, "Microsoft coco: Common objects in context," in European conference on computer vision, 2014.

[6] J. M. Nareshpalsingh and H. N. Modi, "Multi-label classification methods: a comparative study," International Research Journal of Engineering and Technology (IRJET), vol. 4, p. 263–270, 2017.

[7] L. Sun, S. Ji and J. Ye, Multi-label dimensionality reduction, Chapman and Hall/CRC, 2019.

[8] A. Santos, A. Canuto and A. F. Neto, "A comparative analysis of classification methods to multi-label tasks in different application domains," Int. J. Comput. Inform. Syst. Indust. Manag. Appl, vol. 3, p. 218–227, 2011.

[9] A. Aldrees and A. Chikh, "Comparative evaluation of four multi-label classification algorithms in classifying learning objects," Computer Applications in Engineering Education, vol. 24, p. 651–660, 2016.

[10] H. Wang, L. Yan, H. Huang and C. Ding, "From protein sequence to protein function via multi-label linear discriminant analysis," IEEE/ACM transactions on computational biology and bioinformatics, vol. 14, p. 503–513, 2016.

[11] J. Read, B. Pfahringer, G. Holmes and E. Frank, "Classifier chains for multi-label classification," in Joint European Conference on Machine Learning and Knowledge Discovery in Databases, 2009.

[12] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," International Journal of Data Warehousing and Mining (IJDWM), vol. 3, p. 1–13, 2007.

[13] E. Spyromitros, G. Tsoumakas and I. Vlahavas, "An empirical study of lazy multilabel classification algorithms," in Hellenic conference on artificial intelligence, 2008.

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

[15] E. A. Cherman, N. Spolaôr, J. Valverde-Rebaza and M. C. Monard, "Lazy multi-label learning algorithms based on mutuality strategies," Journal of Intelligent & Robotic Systems, vol. 80, p. 261–276, 2015.

[16] M. R. Boutell, J. Luo, X. Shen and C. M. Brown, "Learning multi-label scene classification," Pattern recognition, vol. 37, p. 1757–1771, 2004.

[17] M.-L. Zhang and Z.-H. Zhou, "A review on multi-label learning algorithms," IEEE transactions on knowledge and data engineering, vol. 26, p. 1819–1837, 2013.

[18] R. Cabral, F. De la Torre, J. P. Costeira and A. Bernardino, "Matrix completion for weakly-supervised multi-label image classification," IEEE transactions on pattern analysis and machine intelligence, vol. 37, p. 121–135, 2014.

[19] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, S. Y. Philip and others, "Top 10 algorithms in data mining," Knowledge and information systems, vol. 14, p. 1–37, 2008.

[20] J. L. Villa Medina and others, "Reliability of classification and prediction in k-nearest neighbours," 2013.

[21] 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.

[22] X. Zhu, S. Zhang, Z. Jin, Z. Zhang and Z. Xu, "Missing value estimation for mixed-attribute data sets," IEEE Transactions on Knowledge and Data Engineering, vol. 23, p. 110–121, 2010.

[23] R. Short and K. Fukunaga, "The optimal distance measure for nearest neighbor classification," IEEE transactions on Information Theory, vol. 27, p. 622–627, 1981.

[24] K. Fukunaga and L. Hostetler, "Optimization of k nearest neighbor density estimates," IEEE Transactions on Information Theory, vol. 19, p. 320–326, 1973.

[25] T. Joachims, Learning to classify text using support vector machines, vol. 668, Springer Science & Business Media, 2002.