A K-NEAREST NEIGHBOR MULTILABEL RANKING ALGORITHM WITH APPLICATION TO CONTENT-BASED IMAGE RETRIEVAL

Honglei Zhang[*], *Serkan Kiranyaz*[*†], *Moncef Gabbouj*[*]

[*] Signal Processing
Tampere University of Technology

[†] Electrical Engineering
Qatar University

ABSTRACT

Multilabel ranking is an important machine learning task with many applications, such as content-based image retrieval (CBIR). However, when the number of labels is large, traditional algorithms are either infeasible or show poor performance. In this paper, we propose a simple yet effective multilabel ranking algorithm that is based on k-nearest neighbor paradigm. The proposed algorithm ranks labels according to the probabilities of the label association using the neighboring samples around a query sample. Different from traditional approaches, we take only positive samples into consideration and determine the model parameters by directly optimizing ranking loss measures. We evaluated the proposed algorithm using four popular multilabel datasets. The proposed algorithm achieves equivalent or better performance than other instance-based learning algorithms. When applied to a CBIR system with a dataset of 1 million samples and over 190 thousand labels, which is much larger than any other multilabel datasets used earlier, the proposed algorithm clearly outperforms the competing algorithms.

Index Terms— Multilabel Learning, k-Nearest Neighbor, Content-Based Image Retrieval

1. INTRODUCTION

Multilabel ranking algorithms deal with the problems that each sample can be assigned to multiple labels [1, 2]. Labels used in multilabel learning are not mutually exclusive. This is different from the classes used in traditional multiclass classification where a sample can only be assigned to one class. Multilabel data is very common in many applications such as text categorization, bioinformatics and multimedia content retrieval. For example, an image may be labeled by keywords "cat", "animal" and "funny". Given a query sample, multilabel ranking algorithms give scores to each label and sort them from the most relevant to the least relevant.

Different approaches have been developed to solve the multilabel learning problems. Tsoumakas et al. categorized the algorithms into three groups: problem transformation methods, algorithm adaptation methods and ensemble methods [2]. The problem transformation methods take the binary classification methods as basis and use either one-against-all or one-against-one strategy to get the classification results. The algorithm adaptation methods modify the existing binary classification algorithms to handle multiple labels. The ensemble algorithms apply a set of basic classifiers to subsets of samples and labels and the results are aggregated using sum, voting or other appropriate rules [3]. However, when the number of labels is large, previous algorithms are either infeasible or perform poorly.

In this paper, we propose an instance-based learning algorithm that can effectively handle the problem of the large number of labels. We compared the proposed algorithm with other instance-based multilabel ranking algorithms on four popular benchmark datasets. More importantly, we evaluated the performance of the proposed algorithm using a real-world content-based image retrieval (CBIR) system with a dataset of one million samples and over 190 thousand labels. To our best knowledge, this dataset is much larger than any other multilabel datasets used earlier [1, 2, 4].

2. BACKGROUND AND RELATED WORKS

2.1. Notations

Let $L = \{L_1, L_2, \cdots L_q\}$ be the set of labels, where q is the number of labels. Let $x_i \in R^d$, $i = 1, 2, \cdots, n$ be the feature vector of the samples, where R is the field of real numbers, d is the dimension of the feature vector and n is the number of samples. Let $y_i = \left(y_i^{(1)}, y_i^{(2)}, \cdots y_i^{(q)}\right) \in \{0, 1\}^q$, $i = 1, 2, \cdots, n$ be the set of labels to which sample x_i is assigned, where $y_i^{(l)} = 1$ if x_i is assigned to label L_l. We call $Y_i = \left\{l | y_i^{(l)} = 1\right\}$ the relevant label set, and $\bar{Y_i} = \left\{l | y_i^{(l)} = 0\right\}$ the irrelevant label set. Let $T = \{(x_i, y_i) | i = 1, 2, \cdots, m\}$ be the training set.

The score function for label L_k is defined as $f_k(x) : R^d \rightarrow R$, $k = 1, 2, \cdots q$. The labels are ranked according to these scores such that $rank(x, L_i) < rank(x, L_j)$ if $f_i(x) > f_j(x)$, where $rank(L_i)$ is the rank of label L_i. We aim to learn a set of score functions $\mathcal{F} = \{f_1, f_2, \cdots, f_q\}$ that optimize a predefined objective function.

2.2. Evaluation measures

Different measures have been proposed to evaluate the performance of multilabel ranking algorithms.

Ranking loss evaluates the fraction of label pairs that have been ranked in a wrong order. The evaluation function is defined as

$$ranking\ loss = \frac{1}{n}\sum_{i=1}^{n}\frac{|D_i|}{|Y_i|\left|\bar{Y}_i\right|}, \quad (1)$$

where and $D_i = \{(k,l)\,|f_k(x) > f_l(x), rank(L_k) > rank(L_l)\}$ is the set of labels pairs that have been ranked in a wrong order.

Average precision evaluates average fraction of the labels that are ranked above a true label that is actually in the relevant label set. The metric is defined as

$$average\ precision = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{|Y_i|}\sum_{l\in Y_i}\frac{|B_{i,l}|}{rank(x_i, L_l)}, \quad (2)$$

where $B_{i,l} = \{k|rank(x_i, L_k) > rank(x_i, L_l), k \in Y_i\}$.

In this paper we also use one error and coverage as evaluation measures. Details can be found in [2].

Note, smaller values indicate better performance for all measures except average precision.

2.3. Related work

Binary Relevance (BR) methods apply one against all strategy and learn a binary classifier for each label [2]. For prediction, the binary classifiers for each label are applied independently. Read et al. further developed Classifier Chain (CC) [5] and Classifier Trellis (CT) [6] such that binary classifiers are connected by extending the feature space with the output of other classifiers. Label Power-set (LP) methods learn binary classifiers for sets of labels with different combinations. These methods can effectively deal with the correlation between labels [2]. However, the number of classifiers explodes as the number of labels increases. Most machine learning algorithms for binary classes have been adapted for the multilabel learning problems, for example ML-C4.5 [7], RFML-4.5 [8], and rank-SVM [9]. All of these algorithms learn certain number of classifiers or models from the training set. They show difficulties to handle the problems of large number of labels, large dataset or changes in the training set. For this reason, instance-based learning algorithms are more appropriate for some applications.

Zhang et al. developed the Multilabel k-NN (MLkNN) algorithm from the traditional k-nearest neighbor (k-NN) classification method [10]. MLkNN gather statistical information (the counts of the labels around a sample) for each label from the training set. For prediction, the maximum a posteriori (MAP) approach is applied to determine the set of labels. DMLkNN extends MLkNN method by using not only the statistical information from positive samples, but also negative samples [11]. However neither MLkNN nor DMLKNN perform well when the number of labels is large, since there is insufficient training data to achieve reliable statistical information. Cheng and Hullermerier developed IBLR-ML method by combining linear regression and k-NN algorithms [12]. IBLR-ML method contains q classifiers, similar to BR methods, and thus suffers from the big label set problem too.

Spyromitros et al. combined BR methods with k-NN method (BRkNN) by using the count of label L_l in the set of neighbors as the confidence score [4]. MLC-WkNN improves BRkNN by giving weights to each sample according to its distance to the query sample [13]. The weights are the coefficients of a linear model learned by approximating the query sample from its k nearest neighbors. BR-kNN and MLC-WkNN can handle the large label problem. But their performance suffers due to the simple models they have applied.

3. K-NEAREST NEIGHBOR MULTILABEL RANKING ALGORITHM

3.1. Positive sample model

In our approach, we treat labels as the properties of a sample. The closer two samples are, the more likely they share same labels. In an extreme case, if two samples are identical, they will have the same set of labels.

Let i be the query sample, t be a sample that has label L_l and $d(i,t)$ be the distance between sample i and t. Let $E_i^{(l)}$ denote the event that sample i has label L_l. We model the probability of $E_i^{(l)}$ as a function of the distance between sample i and t, and define it by

$$P\left(E_i^{(l)}|E_t^{(l)}, d(i,t)\right) = \exp\left(-z\cdot d(i,t)\right), \quad (3)$$

where z is a constant number. When $d(i,t) = 0$ (sample i and t are identical), the probability of sample i to have label L_l is 1. In such a case, the prediction of label L_l is determined. Note, when $d(i,t) \mapsto \infty$, the probability function in Eq. 3 returns 0, which shall not be interpreted as the probability of $E_i^{(l)}$ is 0. It actually indicates that sample t does not give any information to infer the association of label L_l.

Taking all positive samples into consideration, we make each of them contribute to the association of label L_l. A sample is not associated with label L_l if none of the positive sample is in favor of it. Thus we derive the probability of $E_i^{(l)}$ given the training set T as

$$P\left(E_i^{(l)}|T\right) = 1 - \prod_{j\in T^{(l)}}\left(1-\exp\left(-z\cdot d(i,t)\right)\right), \quad (4)$$

where T is the training set and $T^{(l)} = \{j|(x_j, y_j) \in T, l \in Y_j\}$.

Because the samples located far from the query sample do little contribution to the probability function in Eq. 4, we can

apply the k-nearest neighbor paradigm. Let $N_k(i)$ be the set of k neighboring samples, we have

$$P\left(E_i^{(l)}|N_k(i)\right) = 1 - \prod_{j \in N_k^{(l)}(i)} \left(1 - \exp\left(-z \cdot d(i,t)\right)\right), \tag{5}$$

where $N_k^{(l)}(i)$ is the set of samples that are associated with label L_l in $N_k(i)$, which is the set of k nearest neighbors of sample i.

Both our approach and IBLR-ML algorithm use exponential function to model the label association. The fundamental difference between our approach and IBLR-ML (and other similar algorithms) is the way we treat negative samples. IBLR-ML algorithm uses both positive and negative samples to learn binary classifiers or regressors; whereas in our approach, only positive samples contribute to the label association. When the number of labels is large, the training set can only be labeled loosely such that the relevant label set for a sample is incomplete even if the labels are accurate. Thus the models using both positive and negative samples often get confused because of the wrong negative samples in training set.

3.2. Model fitting

Since only positive samples are used for prediction, we cannot use maximum likelihood (ML), maximum a posteriori (MAP), or other classification techniques to fit the parameters. However, we can directly optimize any ranking measure defined in Section 2.2. In this paper, we choose the ranking loss as our objective function since it gives a complete evaluation of a ranking result.

Although an analytical solution is hard to find and the objective function is not convex, we can obtain a suboptimal solution using grid search or stochastic gradient descent method because of the model we incorporate. Subsampling techniques can be applied when the size of the training data is large.

Note that the parameter z acts like the smoothing parameter (bandwidth) used in kernel density estimation method. When z is small, a positive sample has a wide impact to the feature space around it; and when z is large, the impact of a positive sample is small. Considering the model in Eq. 4, when $z \to 0$, the proposed algorithm assign the labels according to their empirical prior probabilities. When $z \to +\infty$, the proposed algorithm is equivalent to BRkNN algorithm.

4. EXPERIMENTAL RESULTS ON BENCHMARK DATASETS

In this section, we evaluate the performance of the proposed algorithm on four popular benchmark datasets: Emotions, Scene, Yeast and Mediamill. The datasets are collected from [14] and have been widely used to evaluate the performance of multilabel learning algorithms [1, 6, 10, 15]. Each of the datasets is divided into training and test set by the data providers [14].

We first show the impact of the parameter z using the Yeast dataset. We fix the number of neighbors to be 40 and vary z from 0.01 to 1. The values of the ranking loss against the parameter z are shown in Fig. 1.

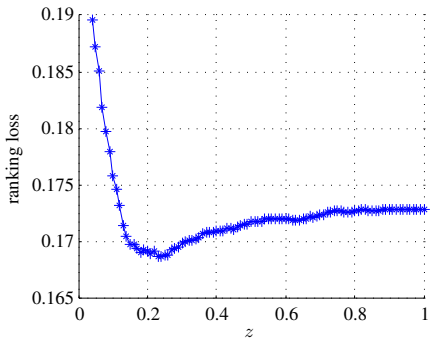

Fig. 1. Graph of ranking loss against the parameter z

The result shows that even though the objective function of the proposed algorithm is not convex, the problem is not ill-posed. Using grid search or stochastic gradient descent method, a good solution can be found. The result also illustrates that the ranking loss converge to a certain value as z increases. According to Section 3.2, the proposed method converges to BRkNN algorithm.

Next, we compare the results of the proposed algorithm with other instance-based methods. Results of MLkNN, IBLR-ML, BRkNN and DMLkNN are obtained using Mulan multilabel learning toolbox [14] and parameter k is selected by optimizing the ranking loss on the test sets. The results are shown in Table 1, where r.l. stands for ranking loss, o.e. stands for one error, cov. stands for coverage and a.p. stands for average precision. The best values are marked using bold font.

Except the Scene dataset, the proposed method performs better than most of the other competing algorithms. Next, we will show the performance of the proposed algorithm in a real-world application with significant higher number of samples and labels.

5. APPLICATION TO CONTENT-BASED IMAGE RETRIEVAL

Given a query image, a CBIR system tries to find "similar" images in a large dataset and present the retrieved images in the order of relevance [16]. However, the concept of "similarity" is ambiguous since it totally depends on the user and the purpose of the query. A query result might be totally irrelevant if the purpose is not correctly justified. One aspect

Table 1. Comparison with other instance-based algorithms

	r.l.	o.e.	cov.	a.p.
Emotions				
MLkNN	**0.145**	**0.252**	1.787	0.818
IBLR-ML	**0.145**	0.262	**1.752**	**0.821**
BRkNN	0.150	0.262	1.792	0.818
DMLkNN	0.148	0.252	1.802	0.817
Ours	**0.145**	0.257	1.772	0.817
Scene				
MLkNN	0.082	0.252	0.512	0.852
IBLR-ML	**0.081**	**0.237**	**0.508**	**0.859**
BRkNN	0.101	0.278	0.607	0.826
DMLkNN	0.083	0.242	0.513	0.855
Ours	0.093	0.259	0.564	0.843
Yeast				
MLkNN	0.170	0.243	6.336	0.753
IBLR-ML	0.166	0.233	6.338	0.763
BRkNN	0.169	0.237	6.314	0.757
DMLkNN	0.169	0.249	6.371	0.757
Ours	**0.160**	**0.226**	**6.116**	**0.771**
Mediamill				
MLkNN	0.051	0.181	18.277	0.716
IBLR-ML	**0.050**	0.181	17.988	0.718
BRkNN	0.054	0.197	19.047	0.709
DMLkNN	**0.050**	0.178	17.924	0.719
Ours	0.051	**0.169**	**16.963**	**0.739**

of this problem has been coined as "semantic gap", which describes the mismatching of the visual feature and the semantic intention of the query [17–19]. Since an image can be described by multiple labels and the purpose of the query is unknown, the CBIR system we used first rank the labels for the query image and then group the retrieved images accordingly. This requires a large number of labels that can describe an unknown image well and a large amount of images that have been labeled.

In this experiment, we converted the MSR-bing challenge dataset into a multilabel ranking dataset [20]. The VGG deep neural network is used to extract features [21].

Because neither the training set nor the test set are fully labeled, we are not able to use the measures defined in Section 2.2. We propose to use the mean inverse rank (MIR) as the evaluation metric. MIR is defined as

$$MIR = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{\left|\tilde{Y}_i\right|}\sum_{l\in\tilde{Y}_i}\frac{1}{rank\left(x_i, L_l\right)}, \qquad (6)$$

where $\tilde{Y}_i$ is the label set given in the test set for sample x_i. Note, $\tilde{Y}_i \subseteq Y_i$, where Y_i is the set of ground truth labels. MIR is similar to average precision defined in Eq. 2. But it does not require the full set of ground truth labels. The larger a MIR value is, the better the algorithm performs.

Other than MIR, we also use success rate as a measure. A query is marked as success if the rank of a true label is less than a predefined value. Let SR@r denote a success rate at position r. Large success rate value indicates better performance of an algorithm.

In this experiment, we compared the proposed algorithm with BRkNN, BRkNN-w and MLkNN algorithms. BRkNN-w algorithm differs from standard BRkNN method by applying a weight to each sample in the nearest neighbor set. In our experiments, the inverse of the distance is used as the weight. For all algorithms, k is set to be 100. The experiment result is shown in Table 2.

Table 2. Experiment results on MSR-bing multilabel dataset

	BRkNN	BRkNN-w	MLkNN	Ours(z=10)
MIR	0.0807	0.0930	0.0551	**0.123**
SR@5	0.103	0.117	0.0736	**0.159**
SR@10	0.152	0.176	0.1075	**0.216**
SR@100	0.327	0.328	0.2160	**0.333**

Table 2 shows that the proposed algorithm significantly outperforms the competing algorithms. The results of BRkNN-w and BRkNN method show that using the distance information between the query sample and the neighboring sample can clearly improve the performance. MLkNN method does not incorporate this information. Since of the number of labels is large, there is not enough samples to collect statistical information for each label. That also accounts for the poor performance of MLkNN algorithm.

6. CONCLUSIONS

In this paper, we have proposed a simple yet effective instance-based multilabel ranking algorithm to tackle the problem of the large number of labels for content-base image retrieval in large scale. We treat labels as the properties of samples and model the probability of having a certain property as an exponential function of the distance. Taking all the positive samples around a query sample into consideration, we calculate the probability of not having the property using a product rule. Unlike traditional methods, we use only positive samples in our method. For this reason, we choose to optimize the ranking loss function directly. Because of the simplicity of our model, grid search or stochastic gradient descent method can effectively find a suboptimal solution.

We compared the performance of the proposed algorithm with other instance-based algorithms on four benchmark datasets. The proposed algorithm is either the best or close to the best on three datasets. We used the proposed algorithm in a CBIR system with the MSR-bing challenge multilabel dataset, which contains 1M samples and over 190 thousand labels. The proposed algorithm clearly outperforms other methods by a significant margin.

7. REFERENCES

[1] E. Gibaja and S. Ventura, "A Tutorial on Multilabel Learning," *ACM Comput. Surv.*, vol. 47, no. 3, pp. 52:1–52:38, Apr. 2015.

[2] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *Dept. of Informatics, Aristotle University of Thessaloniki, Greece*, 2006.

[3] J. Read, L. Martino, and D. Luengo, "Efficient monte carlo methods for multi-dimensional learning with classifier chains," *Pattern Recognition*, vol. 47, no. 3, pp. 1535–1546, Mar. 2014.

[4] E. Spyromitros, G. Tsoumakas, and I. Vlahavas, "An empirical study of lazy multilabel classification algorithms," in *Hellenic conference on Artificial Intelligence*. Springer, 2008, pp. 401–406.

[5] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," *Machine Learning*, vol. 85, no. 3, pp. 333–359, Jun. 2011.

[6] J. Read, L. Martino, P. M. Olmos, and D. Luengo, "Scalable multi-output label prediction: From classifier chains to classifier trellises," *Pattern Recognition*, vol. 48, no. 6, pp. 2096–2109, Jun. 2015.

[7] A. Clare and R. D. King, "Knowledge discovery in multi-label phenotype data," in *European Conference on Principles of Data Mining and Knowledge Discovery*. Springer, 2001, pp. 42–53.

[8] D. Kocev, C. Vens, J. Struyf, and S. Džeroski, "Ensembles of multi-objective decision trees," in *European Conference on Machine Learning*. Springer, 2007, pp. 624–631.

[9] A. Elisseeff and J. Weston, "A kernel method for multi-labelled classification," in *Advances in neural information processing systems*, 2001, pp. 681–687.

[10] 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.

[11] Z. Younes, F. Abdallah, T. Denoeux, and H. Snoussi, "A dependent multilabel classification method derived from the k-nearest neighbor rule," *EURASIP Journal on Advances in Signal Processing*, vol. 2011, no. 1, pp. 1–14, 2011.

[12] W. Cheng and E. Hüllermeier, "Combining instance-based learning and logistic regression for multilabel classification," *Machine Learning*, vol. 76, no. 2-3, pp. 211–225, 2009.

[13] J. Xu, "Multi-Label Weighted k-Nearest Neighbor Classifier with Adaptive Weight Estimation," in *Neural Information Processing*, ser. Lecture Notes in Computer Science, B.-L. Lu, L. Zhang, and J. Kwok, Eds. Springer Berlin Heidelberg, Nov. 2011, no. 7063, pp. 79–88.

[14] G. Tsoumakas, E. Spyromitros-Xioufis, J. Vilcek, and I. Vlahavas, "Mulan: A java library for multi-label learning," *Journal of Machine Learning Research*, vol. 12, no. Jul, pp. 2411–2414, 2011.

[15] 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, Sep. 2012.

[16] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349–1380, Dec. 2000.

[17] Y. Liu, D. Zhang, G. Lu, and W.-Y. Ma, "A survey of content-based image retrieval with high-level semantics," *Pattern Recognition*, vol. 40, no. 1, pp. 262–282, Jan. 2007.

[18] M. K. Kundu, M. Chowdhury, and S. Rota Bulò, "A graph-based relevance feedback mechanism in content-based image retrieval," *Knowledge-Based Systems*, vol. 73, pp. 254–264, Jan. 2015.

[19] B. Xu, J. Bu, C. Chen, C. Wang, D. Cai, and X. He, "EMR: A Scalable Graph-Based Ranking Model for Content-Based Image Retrieval," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 1, pp. 102–114, Jan. 2015.

[20] X.-S. Hua, M. Ye, and J. Li, "Mining knowledge from clicks: MSR-Bing image retrieval challenge," in *Multimedia and Expo Workshops (ICMEW), 2014 IEEE International Conference on*. IEEE, 2014, pp. 1–4.

[21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.