An Empirical Comparison of Weighting Functions for Multi-label Distance-weighted k-Nearest Neighbour Method

Jianhua Xu

School of Computer Science and Technology, Nanjing Normal University, Nanjing 210097, China
xujianhua@njnu.edu.cn

Abstract

Multi-label classification is an extension of classical multi-class one, where any instance can be associated with several classes simultaneously and thus the classes are no longer mutually exclusive. It was experimentally shown that the distance-weighted k-nearest neighbour (DWkNN) algorithm is superior to the original kNN rule for multi-class learning. But, it has not been investigated whether the distance-weighted strategy is valid for multi-label learning and which weighting function performs well. In this paper, we provide a concise multi-label DWkNN form (MLC-DWkNN). Furthermore, four weighting functions, Dudani's linear function varying from 1 to 0, Macleod's linear function ranging from 1 to 1/2, Dudani's inverse distance function, and Zavrel's exponential function, are collected and then investigated by detailed experiments on three benchmark data sets with Manhattan distance. Our study demonstrates that Dudani's linear and Zavrel's exponential functions work well, and moreover MLC-DWkNN with such two functions outperforms an existing kNN-based multi-label classifier ML-kNN.

Keywords

Multi-label Classification, k-Nearest Neighbours, Weighting Function, Manhattan Distance

1. Introduction

Multi-label classification is a generalization of traditional multi-class one, in which any instance could belong to several classes (or labels) at the same time and thus the classes are not mutually exclusive [1]. Typical examples involve text categorization, scene classification and functional genomics. In text categorization, each document is usually linked to several different topics, such as music, entertainment and sport. In scene annotation, each picture is often labelled by multiple conceptual types, such as beach, sea and sunset. In gene functional detection, each gene has many effects on a cell, such as energy, metabolism and cellular biogenesis.

Nowadays, the existing discriminative multi-label approaches can be mainly categorized into two groups: algorithm extension and data decomposition. The former considers all instances and all classes at once, resulting into some complicated optimization problems, e.g., large-scale quadratic programming in multi-label support vector machine (Rank-SVM) [2] and unconstrained optimization problem in multi-label neural networks (BP-MLL) [3]. The latter implicitly or explicitly applies some data decomposition trick (e.g., one-versus-rest, one-versus-one, or label powerset) to divide an entire multi-label data set into a series of binary or multi-class subsets which are easy and convenient to be solved by lots of existing single-label classifiers or their modified versions [1]. However, how to design and implement efficient and effective multi-label classifiers is still a challenging issue.

David Bracewell, et al. (Eds): AIAA 2011,CS & IT 03, pp. 13–20, 2011.

DOI : 10.5121/csit.2011.1302

The *k*-nearest neighbour (*k*NN) method [4-6] is one of the oldest and simplest algorithms for classical single-label (binary or multi-class) learning. Despite its simplicity, the asymptotic analysis shows that the error rate will not greater than twice the Bayesian error rate [4, 5]. In essence, the original *k*NN method assumes that each neighbour has the same contribution to a query instance. The distance-weighted *k*NN method (DW*k*NN) was proposed in [7], which weighs the instances nearby more heavily than those farther away. Some analytical weighting functions have been designed in [7-9], most of which monotonously decrease as the distance between the query instance and any neighbour increases. The performance of DW*k*NN has been verified by many experiments [8, 9] for single-label classification.

Due to the success of *k*NN and WD*k*NN in single-label learning, their several multi-label versions have been proposed recently. The simplest form is referred to as MLC-*k*NN simply in this paper, which is directly to assign several labels of each multi-label neighbour to different classes simultaneously, and then decide a query instance to be associated with several classes with more than *k*/2 votes. Note that, in essence, this method can be interpreted and implemented using one-versus-rest decomposition (or binary relevance) and binary *k*NN rule. Therefore it is abbreviated to BR-*k*NN in [10].

Four complicated multi-label classifiers (ML-*k*NN [11], DML-*k*NN [12], Mr.KNN [13] and IBLR-ML [14]) combine *k*NN or WD*k*NN with one or two additional techniques. ML-*k*NN [11] utilizes discrete Bayesian formula for each class independently. DML-*k*NN [12] generalizes ML-*k*NN to further consider label dependencies via linking posterior probability of each class to the occurrence frequencies of labels of all classes. Mr.KNN [13] both adds supervised fuzzy c-mean (FCM) algorithm to characterize soft relevance value for each instance with each label, and integrates exponential weighting function [9, 15] into its discriminant functions. IBLR-ML [14] uses inverse distance weighting function [7] to calculate the weighted sum of positive and negative labels of all classes, and estimates class posterior probabilities by logistic regression (LR). It is worth noting these four multi-label methods all need a training procedure to estimate all quantities for their additional models, which is implemented using leave-one-out procedure. Therefore, strictly speaking, these approaches are not model-free and instance-based yet. Additionally, in Mr.KNN and IBLR-ML, besides distance-weighted strategy, an additional technique was exploited, but it was not identified how one or both of two techniques perform. To the best of our knowledge, no empirical comparison of weighting functions only for multi-label WD*k*NN-type classifiers has been reported so far.

In this paper, we still focus on model-free and instance-based characteristic of WD*k*NN techniques. The DW*k*NN algorithm is generalized to construct its concise multi-label classification form: MLC-DW*k*NN. Four different distance-weighted functions [7-9] are collected and compared by detailed experiments. Our results show that Dudani's linear function [7] and Zavrel's exponential function [9] work well, and further their corresponding MLC-DW*k*NN forms even outperform ML-*k*NN classifier [11].

This paper is organized as follows. In Section 2, we describe MLC-DW*k*NN. Four different distance-weighting functions are summarized in Section 3. Experimental results are reported and analyzed in Section 4. Finally Section 5 ends up this paper with some conclusions.

2. Multi-label Distance-weighted k-Nearest Neighbour Method

Assume a *q*-class training data set of size *l* drawn identically and independently from an unknown probability distribution to be,

$$\{(\boldsymbol{x}_1,\boldsymbol{y}_1),...,(\boldsymbol{x}_i,\boldsymbol{y}_i),...,(\boldsymbol{x}_l,\boldsymbol{y}_l)\}, \tag{1}$$

where $\boldsymbol{x}_i \in R^d$ denotes a *d*-dimensional instance vector, and $\boldsymbol{y}_i = [y_{i1},...,y_{ij},...,y_{iq}]^T$ is its *q*-dimensional binary label vector, in which $y_{ij} = +1$ if $\boldsymbol{x}_i$ belongs to the *j*th class (i.e., a relevant label of $\boldsymbol{x}_i$), otherwise $y_{ij} = -1$ (i.e., an irrelevant label). It is worth noting that here any $\boldsymbol{y}_i$ has many components with +1 possibly. When only one component of $\boldsymbol{y}_i$ is equal to +1, a multi-label problem degrades into a traditional multi-class one.

For a query instance $\boldsymbol{x}$, we search for the first k-nearest neighbours with some distance metric from the training set (1), and sort them in increasing order of distances ($d_1 \le ... \le d_k$) as follows,

$$\{(\overline{\boldsymbol{x}}_1, \overline{\boldsymbol{y}}_1), ..., (\overline{\boldsymbol{x}}_i, \overline{\boldsymbol{y}}_i), ..., (\overline{\boldsymbol{x}}_k, \overline{\boldsymbol{y}}_k)\}. \quad (2)$$

In multi-class classification, the kNN rule sums up the number of instances belonging to each class from (2) as its discriminant functions,

$$f_j(\boldsymbol{x}) = |\{i \mid \overline{y}_{ij} = +1, i = 1, ..., k\}|, j = 1, ..., q, \quad (3)$$

where $\overline{y}_{ij} = \pm 1$ denotes the jth binary label of the ith nearest neighbour. When each nearest neighbour $\overline{\boldsymbol{x}}_i$ is allocated to a non-negative distance-based weight w_i, the DWkNN rule calculates the weight sum over those instances belonging to each class,

$$f_j(\boldsymbol{x}) = \sum_{i=1, \overline{y}_{ij}=+1}^{k} w_i. \quad (4)$$

Regardless of kNN and DWkNN, the majority voting strategy is used to decide one of labels only for $\boldsymbol{x}$,

$$y_j = \begin{cases} +1, if\ f_j(\boldsymbol{x}) \ge f_{j'}(x), j' = 1, ..., q, j' \ne j \\ -1, \quad otherwise \end{cases} \quad (5)$$

In multi-label classification, since a query instance is possibly associated with many classes, the above formulae (3)-(5) have to be extended properly. We consider all positive and negative labels of each class independently in (2). For the jth class, the difference between the number of positive labels and that of negative labels is defined as its discriminant functions,

$$f_j(\boldsymbol{x}) = \sum_{i=1}^{k} \overline{y}_{ij} = \sum_{i=1, \overline{y}_{ij}=+1}^{k} \overline{y}_{ij} - \sum_{i=1, \overline{y}_{ij}=-1}^{k} (-\overline{y}_{ij}). \quad (6)$$

With distance-based weights, the corresponding discriminant functions are considered as the weighted sum of positive and negative labels of each class,

$$f_j(\boldsymbol{x}) = \sum_{i=1}^{k} w_i \overline{y}_{ij} = \sum_{i=1, \overline{y}_{ij}=+1}^{k} w_i \overline{y}_{ij} - \sum_{i=1, \overline{y}_{ij}=-1}^{k} (-w_i \overline{y}_{ij}). \quad (7)$$

According to (6) and (7), the predicted binary label vector $\boldsymbol{y} = [y_1, ..., y_j, ..., y_q]^T$ of $\boldsymbol{x}$ is detected by,

$$y_j = \begin{cases} +1, if\ f_j(\boldsymbol{x}) \ge 0 \\ -1, \ otherwise \end{cases} \quad (8)$$

In this paper, such two kNN-type multi-label methods respectively based on (6) and (7), are simply referred to as MLC-kNN and MLC-DWkNN. Furthermore, we use suffixes to distinguish MLC-DWkNN with different weighting functions. Although these two methods can also be interpreted and implemented using one-versus-rest decomposition trick, and binary kNN or WkNN rule, we give a concise representation here. Note that in [10], our MLC-kNN is referred to as BR-kNN, but no detailed formula is provided.

3. FOUR WEIGHTING FUNCTIONS

For the classical DWkNN for multi-class classification, several analytical distance-based weighting functions have been constructed in [7 - 9], whose basic characteristic is that a neighbour with a smaller distance should be weighted more heavily than one with a greater distance. Therefore any distance-based weighting function is monotonously decreasing with distance. In [7], Dudani proposed and tested a linear function,

$$\text{Dudani1: } w_i = \begin{cases} \frac{(d_k - d_i)}{(d_k - d_1)}, if \;\; d_k \neq d_1, \\ 1, \qquad \text{otherwise,} \end{cases} \tag{9}$$

in which the nearest neighbour gets a weight of 1, the furthest one a weight of 0, and the other weights are scaled linearly to the interval in between. Since the formula (9) essentially removes the *k*th nearest neighbour from participating in the *k*NN rule due to $w_k = 0$, its modified form was introduced in [8],

$$\text{Macleod: } w_i = \begin{cases} \frac{(d_s - d_i) + \alpha(d_s - d_1)}{(1+\alpha)(d_s - d_1)}, if \;\; d_s \neq d_1, \\ 1, \qquad \text{otherwise,} \end{cases} \tag{10}$$

where $s = k, k+1, \ldots$ and α is a positive constant. In [8], several combinations of s and α were verified. In this paper, we only test one combination of $s = k$ and $\alpha = 1$. This means that the furthest neighbour is allocated a weight of 1/2.

Dudani also proposed an inverse distance weighting function, i.e., the reciprocal of distance, in [7]

$$\text{Dudani2: } w_i = \tfrac{1}{d_i}, if \;\; d_i \neq 0. \tag{11}$$

Note that this function was not evaluated in [7] and subsequently was pointed out to work slightly badly in [9]. The formula (11) takes very larger values for distances close to zero and reduces the *k*NN rule in many cases into the simple 1NN one [7]. To avoid division by zero, a small constant (0.01) is added to the denominator in our study, as recommended in [9]. An exponential weighting function based on a universal perceptual law [15] was used in [9],

$$\text{Zavrel: } w_i = \exp(-\alpha d_i^{\beta}). \tag{12}$$

A specific form with $\alpha = \beta = 1$ in [9] is validated in this paper. In the next section, we will test the above four analytical weighting functions for MLC-DW*k*NN, whose curves are shown in Figure 1, where $d_1 = 4$ and $d_k = 10$.

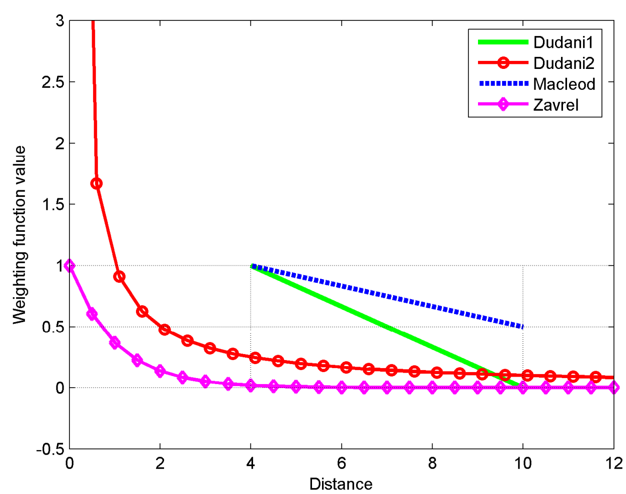

Figure 1. Four weighting functions used in our study

4. EXPERIMENTS

In this section, we test the above four distance-based weighting functions for MLC-DW*k*NN, and compare these MLC-DW*k*NN forms with MLC-*k*NN and ML-*k*NN, on three benchmark data sets, according to five widely-used measures. ML-*k*NN cascades MLC-*k*NN and discrete binary Bayesian formula, where the prior and discrete conditional probabilities are estimated through combining leave-one-out (LOO) procedure with *k*NN rule [11].

4.1. Five Evaluation Measures and Three Data Sets

It is more difficult and complicated to evaluate a multi-label classifier than a single one. In this paper, we utilize five popular and indicative measures in [11], as listed in Table 1, where the up arrow means that the higher a measure is, the better a method performs, and just the reverse for the down arrow. All these measures are calculated for a single instance first and then are averaged over all test instances. Among five measures, only the Hamming loss depends on the predicted label set of a test instance, while the other four are associated with the sorted discriminant function values. On their definitions, please refer to [11].

Table 1.Five measures used in this paper

Measure	Interval	Good performance
Ranking loss	[0, 1]	↓
Coverage	[0, *q*]	↓
One error	[0, 1]	↓
Average precision	[0, 1]	↑
Hamming loss	[0, 1]	↓

Table 2. Statistics for three benchmark data sets used in our experiments

Data set	Domain	Instances	Features	Classes	Average labels
Emotions	Music	593	72	6	1.87
Image	Scene	2000	294	5	1.24
Yeast	Biology	2417	103	14	4.24

On the other hand, we collect three widely-used benchmark data sets: Emotions, Image, and Yeast from [16, 17], as listed in Table 2. Table 2 shows some useful statistics of these data sets, such as, the number of instances, features and classes, and the average labels. These data sets cover three distinct domains: semantic scene, music and biology. For more detailed information and description for these data sets, please refer to their web sites and references therein [16, 17].

2.2. Results and Analysis

In this paper, we utilize 10-fold cross validation to evaluate four different weighting functions for MLC-DW*k*NN, and compare our technique with its original unweighted form MLC-*k*NN and high-performed classifier ML-*k*NN. To reduce the effect of random seeds, three repeats are conducted and thus each measure denotes the average value in this section. Since the Manhattan or absolute distance is tested, there is only a tunable parameter k in all methods.

To begin with, we investigate the ranking loss as a function of k value for three data sets, as shown in Figure 2, where k is varied from 5 to 100 with a step of 5. It is observed that: (a) most of curves share a general tendency that as the k value increases the ranking loss value dramatically decreases first, achieves a optimum then and lastly increases almost monotonically; (b) For all k values, four MLC-DW*k*NN versions are superior to its unweighted form MLC-*k*NN; (c) MLC-DW*k*NN-Dudani2 and -Macleod have a very approximate performance; (d) when $k \geq 20$, MLC-DW*k*NN-Dudani1 and -Zavrel outperform ML-*k*NN consistently.

According to the optimal *k* values in Figure 2, we list all corresponding five measures for three data sets, as shown in Tables 3-5. In order to compare these methods comprehensively, we sort them using each measure first and then calculate the average rank over five measures for each method, as listed in the last rows of Tables 3-5. This comparison strategy was recommended in [18].

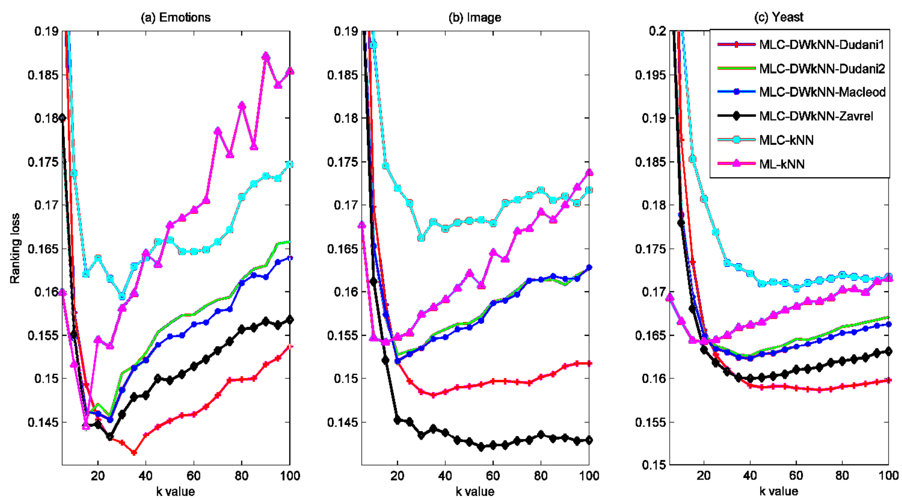

Figure 2. The ranking loss as a function of *k* for Emotions, Image and Yeast

Table 3. Five measure values from Emotions

Measure	MLC-DW*k*NN				MLC-*k*NN	ML-*k*NN
	Dudani1 (*k*=35)	Dudani2 (*k*=15)	Macleod (*k*=25)	Zavrel (*k*=25)	(*k*=30)	(*k*=15)
↓Ranking loss	**0.1414**(1)	0.1456(5)	0.1453(4)	0.1433(2)	0.1595(6)	0.1445(3)
↓Coverage	**1.6869**(1)	1.7032(5)	1.6976(4)	1.6931(3)	1.7589(6)	1.6897(2)
↓One error	**0.2277**(1)	0.2451(4)	0.2400(3)	0.2305(2)	0.2664(6)	0.2490(5)
↑Average precision	**0.8258**(1)	0.8194(4)	0.8199(3)	0.8235(2)	0.8036(6)	0.8180(5)
↓Hamming loss	**0.1756**(1)	0.1806(4)	0.1825(5)	0.1797(3)	0.1793(2)	0.1884(6)
↓Average Rank	**1.00**	4.40	3.80	2.40	5.2	4.2

Table 4. Five measure values from Image

Measure	MLC-DW*k*NN				MLC-*k*NN	ML-*k*NN
	Dudani1 (*k*=35)	Dudani2 (*k*=20)	Macleod (*k*=20)	Zavrel (*k*=55)	(*k*=30)	(*k*=15)
↓Ranking loss	0.1481(2)	0.1527(4)	0.1520(3)	**0.1421**(1)	0.1662(6)	0.1542(5)
↓Coverage	0.8687(2)	0.8872(4)	0.8828(3)	**0.8455**(1)	0.9447(6)	0.8923(5)
↓One error	0.2710(2)	0.2823(4)	0.2830(5)	**0.2565**(1)	0.2898(6)	0.2815(3)
↑Average precision	0.8220(2)	0.8155(4)	0.8160(3)	**0.8299**(1)	0.8035(6)	0.8143(5)
↓Hamming loss	0.1528(2)	0.1555(3)	0.1571(4)	**0.1500**(1)	0.1600(6)	0.1583(5)
↓Average Ranking	2.00	3.80	3.60	**1.00**	6.00	4.60

Table 5. Five measure values from Yeast

Measure	MLC-DW*k*NN				MLC-*k*NN	ML-*k*NN
	Dudani1 (*k*=70)	Dudani2 (*k*=40)	Macleod (*k*=40)	Zavrel (*k*=40)	(*k*=60)	(*k*=20)
↓Ranking loss	**0.1587**(1)	0.1626(4)	0.1623(3)	0.1600(2)	0.1704(6)	0.1643(5)
↓Coverage	**6.0759**(1)	6.1556(4)	6.1512(3)	6.1047(2)	6.2932(6)	6.2169(5)
↓One error	**0.2259**(1)	0.2324(5)	0.2309(4)	0.2277(2)	0.2400(6)	0.2278(3)
↑Average precision	**0.7717**(1)	0.7653(5)	0.7664(4)	0.7702(2)	0.7564(6)	0.7677(3)
↓Hamming loss	**0.1894**(1)	0.1938(5)	0.1933(4)	0.1902(2)	0.1971(6)	0.1930(3)
↓Average Rank	**1.00**	4.60	3.60	2.00	6.00	3.80

In Table 3 for the Emotions data, MLC-DW*k*NN-Dudani1 works the best on all five measures. In terms of the average rank, it is found that four MLC-DW*k*NN forms perform better than MLC-*k*NN, and further MLC-DW*k*NN-Dudani1, -Macleod and -Zavrel are superior to ML-*k*NN.

In Table 4 on the Image data, MLC-DW*k*NN-Zavrel achieves five best measures. According to the average rank, all MLC-DW*k*NN versions outperform its original one MLC-*k*NN and even ML-*k*NN.

In Table 5 for the Yeast data, MLC-DW*k*NN-Dudani1 obtains the top position on all five measures. From the average rank, our all MLC-DW*k*NNs behave better than MLC-*k*NN, and further all these methods but Dudani2 version, outperform ML-*k*NN.

According to Tables 3-5, it is illustrated that the Dudani1's and Zavrel's functions (9) and (12) are two best candidates among four weighting ways, whose MLC-DW*k*NN versions are superior to the high-performed multi-label approach ML-*k*NN [11].

5. CONCLUSIONS

In this paper, we construct a concise multi-label distance-weighted *k*-nearest neighbour algorithm and compared four weighting functions with three benchmark data sets and five evaluation measures. From our study, it can be concluded that, (a) various multi-label distance-weighted *k*-nearest neighbour methods can indeed improve the performance of their original unweighted form; (b) Dudina linear function varying from 1 to 0, and Zarvel exponential function, are two best weighting functions, whose performance for multi-label classification outperforms that of the state-of-the-art technique ML-*k*NN, when the Manhattan distance is used. Since our multi-label distance-weighted *k*-nearest neighbour method is still model-free, instance-based and well-performed, it will be widely used in many applications. In our future work, we will conduct experiments on more benchmark data sets and use different distance metrics to show our above issue further.

ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China under grant no. 60875001.

REFERENCES

[1] Tsoumakas, G., Katakis, I. & Vlahavas I. (2010) "Mining multi-label data", in: Maimon, O., Rokach, L. (eds.), *Data Mining and Knowledge Discovery Handbook*, Second Edition, Springer, New York, pp667-685.

[2] Elisseeff, A. & Weston, J. (2001) "A kernel method for multi-labelled classification", in: *Proceedings of the 14th Conference on Neural Information Processing Systems* (NIPS2001), pp681-687.

[3] Zhang, M.L. & Zhou, Z.H. (2006) "Multi-label neural networks with application to function genomics and text categorization", *IEEE Transactions on Knowledge and Data Engineering*, Vol.18, No.10, pp1338-1351.

[4] Cover, T.M. & Hart, P.E. (1967) "Nearest neighbor pattern classification". *IEEE Transactions on Information Theory*, Vol. 13, No. 1, pp21-27.

[5] Duda, R.O. & Hart, P.E. (1973) *Pattern Classification and Scene Analysis*, Wiley, New York.

[6] Wu, X., Kumar, V., Quinlan, J.R., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G.J., Ng, A., Liu, B., Yu, P.S., Zhou, Z.H., Steinbach, M., Hand, D.J. & Steinberg, D. (2008) "Top 10 algorithms in data mining", *Knowledge and Information Systems*, Vol. 14, No. 1, pp1-37.

[7] Duduni, S.A. (1967) "The distance-weighted k-nearest neighbor rule", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 6, No. 2, pp325-327.

[8] Macleod, J.E.S., Luk, A. & Titterington, D.M. (1987) "A re-examination of the distance-weighted *k*-nearest neighbor classification rule", *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 17, No. 4, pp689-696.

[9] Zavrel, J. (1997) "An empirical re-examination of weighted voting for k-NN", in: Daelemans, W., Flach, P. & van den Bosch, A. (eds.) *Proceedings of the 7th Belgian-Dutch Conference on Machine Learning*, pp139-148.

[10] Spyromittos, E., Tsoumakas, G. &Vlahavas, I. (2008) "An empirical study of lazy multilabel classification algorithms", in: *Proceedings of the 5th Hellenic Conference on Artificial Intelligence* (SETN 2008), pp401-406.

[11] Zhang, M.L. & Zhou, Z.H. (2007) "ML-*k*NN: a lazy learning approach to multilabel learning", *Pattern Recognition*, Vol. 40, No. 7, pp2038-2048.

[12] Younes, Z., Abdallah, F. & Denoeux, T. (2008) "Multi-label classification algorithm derived from k-nearest neighbour rule with label dependencies", in: *Proceedings of the 16th European Signal Processing Conference* (EUSIPCO 2008), pp25-29.

[13] Lin, X.T. & Chen X.W. (2010) "Mr.KNN: soft relevance for multi-label classification", in: *Proceedings of the 19th ACM International Conference on Information and Knowledge Management* (CIKM2010), pp 349 - 358.

[14] Cheng, W.W. & Hullermeier, E. (2009) "Combining instance-based learning and logistic regression for multi-label classification", *Machine Learning*, Vol. 76, No. 2/3, pp211-225.

[15] Stepard, R.N. (1987) "Toward a universal law of generalization for psychological science", *Science*, Vol. 237, No. 2820, pp1317-1323.

[16] Tsoumakas, G. (2010) "Multi-label datasets", http://mulan.sourceforge.net/datasets.html.

[17] Zhang, M.L. (2010) "Image data set", http://cse.seu.edu.cn/people/zhangml.

[18] Brazdil, P.B. & Soares, G. (2000) "A comparison of ranking methods for classification algorithm selection", in: *Proceedings of the 11th European Conference on Machine Learning* (ECML2000), LNCS, Vol. 1810, pp63-74.