

Purifying training data to improve performance of multi-label classification algorithms

Sawsan Kanj

HEUDIASYC, UMR CNRS 7253
Université de Technologie de Compiègne
Compiègne - France
sawsan.kanj@hds.utc.fr

Fahed Abdallah

HEUDIASYC, UMR CNRS 7253
Université de Technologie de Compiègne
Compiègne - France
fahed.abdallah@hds.utc.fr

Thierry Denceux

HEUDIASYC, UMR CNRS 7253
Université de Technologie de Compiègne
Compiègne - France
thierry.denoix@hds.utc.fr

Abstract—Multi-label classification assumes that each object in the training set is associated with a set of labels, and the goal is to assign labels to unseen instances. k -nearest neighbors based algorithms answer the multi-label problem by using inherent information given by the neighbors of the observation to classify. Due to several problems, like errors in the input vectors, or in their labels, this information may be wrong and might lead the multi-label algorithm to fail. In this paper, we propose a simple algorithm for editing out some training instances by voting of some metrics in order to purify the existing training sample. This purifying approach is adapted on the recently proposed evidential k -nearest neighbors for multi-label classification. Comparative experimental results on various data sets demonstrate the usefulness and effectiveness of our approach.

I. INTRODUCTION

In multi-label classification, each instance in the training set belongs to several classes from a set of predefined labels. Recently, multi-label classification methods have received increasing attention in machine learning. In text categorization, an electronic document can be referred to sport topics as to politics and society. In semantic scene classification, an image may contain multiple objects. In video annotation, a film can be annotated with several labels or tags.

The Nearest neighbor (NN) rule is one of the simplest and most popular methods for statistical learning [1]. This is an instance-based classifier that has been shown to be very effective in many classification problems [2]. The intuition is simple. Giving a single-labeled training set, the classification of a query instance \mathbf{x} is performed by assigning it the label of the least distant training pattern according to some distance measure. The voting k -nearest neighbor rule, with $k \geq 1$, is a generalization of the NN approach where the most frequent class occurring in the k neighbors of \mathbf{x} is predicted.

The evidential multi-label k -nearest neighbor (EML k NN) method [3], [4], [5] is a recent algorithm which answers the multi-label problem under the belief function framework. It considers each neighbor as an item of evidence described by two-valued mass function, and combines these evidence using an adapted rule. A decision rule is then introduced to associate labels to unseen instances. Due to the high degree of local sensitivity of the nearest neighbors based methods [6], the performance of this rule is decreasing with the presence of noisy examples in the training set. Noisy examples include

those with errors in the input vector, and/or in their label assignment.

Several works under the single-label classification problem have been proposed in order to offer an edited and clean data set in order to improve methods' performances [6], [7], [8]. In this paper, we propose a simple algorithm to purify training data set and eliminate erroneous instances using the hamming loss metric evaluation [9] calculated under the EML k NN framework. The effectiveness of the proposed algorithm is shown on several real world data sets widely used in the literature of multi-label classification. More than increasing classification performance, the new method has the advantage of needing less of storage requirements and decreasing the running time of the initial EML k NN algorithm.

The remainder of this paper is organized as follows. Section II describes the principle of multi-label classification and recalls the EML k NN method. Section III introduces the purified EML k NN and its implementation. Section IV presents results of some experiments. Finally, section V concludes this paper.

II. MULTI-LABEL CLASSIFICATION

A. Principle

Let \mathcal{X} denote an instance space, and let $\Omega = \{\omega_1, \dots, \omega_Q\}$ be a finite set of class labels. We can state the multi-label classification problem as follows. Given a training set of observations $\mathcal{T} = \{(\mathbf{x}_1, \mathbf{Y}_1), \dots, (\mathbf{x}_n, \mathbf{Y}_n)\}$, where \mathbf{x}_i is a feature vector describing instance i , and $\mathbf{Y}_i \in 2^\Omega$ is the set of labels for that instance, the goal is to learn a multi-label classifier $\mathcal{H} : \mathcal{X} \rightarrow 2^\Omega$ that assigns a predicted label subset to each unseen instance.

Several methods have been proposed in the literature for multi-label learning. These methods can be categorized into two main groups: problem transformation and problem adaptation methods [9]. The former group transforms the multi-label classification problem into binary classification problems (a binary classifier for each class), while the latter adapts specific algorithms to manipulate multi-label data directly. In this paper, we mainly focus on the EML k NN method specially developed to handle multi-label data.

B. Evidential multi-label k -NN classification

The evidential k -NN method introduced in [3] for multi-label classification problems can be summarized as follows. Let $\mathcal{T} = \{(\mathbf{x}_1, A_1, B_1), \dots, (\mathbf{x}_n, A_n, B_n)\}$ be the learning set, where $A_i \subseteq \Omega = \{\omega_1, \dots, \omega_Q\}$ denotes a set of classes that surely apply to the instance \mathbf{x}_i , and $B_i \subseteq \Omega$ a set of classes that surely do not apply to the same instance. Ω is known as the *frame of discernment* of the problem.

Given \mathbf{x} an unseen instance that we look for estimating its set of labels \mathbf{Y} . Let $\mathcal{N}_{\mathbf{x}}$ be its k nearest neighbors in \mathcal{T} based on a certain distance function. Each element \mathbf{x}_i in $\mathcal{N}_{\mathbf{x}}$ constitutes an item of evidence regarding the label set of \mathbf{x} . This item of evidence can be described by the following simple two-valued mass function:

$$\begin{aligned} m_i(A_i, B_i) &= \alpha \exp(-\gamma d(\mathbf{x}, \mathbf{x}_i)), \\ m_i(\emptyset, \emptyset) &= 1 - \alpha \exp(-\gamma d(\mathbf{x}, \mathbf{x}_i)), \end{aligned} \quad (1)$$

where $d(\mathbf{x}, \mathbf{x}_i)$ is the distance between \mathbf{x} and \mathbf{x}_i , α and γ are two parameters, such that $0 < \alpha < 1$ and $\gamma > 0$. Parameter α is usually fixed to a value close to 1 such as 0.95 [10], whereas γ can be optimized or fixed heuristically [11]. The resulting k mass functions are then combined using the conjunctive sum:

$$m = \odot_{i: \mathbf{x}_i \in \mathcal{N}_{\mathbf{x}}} m_i. \quad (2)$$

To calculate the set of predicted labels for instance \mathbf{x} , two quantities are computed separately for each label $\omega \in \Omega$: the degree of belief $bel(\{\omega\}, \emptyset)$ that the true label set \mathbf{Y} contains ω , and the degree of belief $bel(\emptyset, \{\omega\})$ that it does not contain ω . Then, the multi-label classifier \mathcal{H} is defined as:

$$\mathcal{H}(\mathbf{x}) = \{\omega \in \Omega / bel(\{\omega\}, \emptyset) \geq bel(\emptyset, \{\omega\})\}, \quad (3)$$

where \emptyset denotes the empty set of Ω .

III. PURIFYING DATA SET FOR EMLkNN

A. Motivation

The k -nearest neighbor based algorithms are very popular in Machine Learning domain due to their simplicity and effectiveness at the same time [1]. Several algorithms based on the k -NN rule were developed for multi-label classification. Due to the high degree of local sensitivity of the k -NN rule, these algorithms are susceptible to erroneous instances in the training set [6]. For example, a noisy input vector \mathbf{x} with correct set of labels will probably be in a region of the hyperspace where no similar labeled examples. If the number of noisy vectors increases, the performance of the multi-label classifier will naturally decrease. Furthermore, in many applications, in addition to errors on the features of the training instances, many examples may have an erroneous set of labels due to an experimental assignment problem or even a human annotation error. This will affect the multi-label classifier performance. The aim of our work is to increase the classifier's generalization ability by removing noisy instances from the training set.

B. Performance evaluation for multi-label classifiers

For the traditional single-label classification task, the target classes are disjoint and exclusive and each example belong to one and only one class, while for the multi-label classification task, the target classes are not exclusive and an example may belong to an unrestricted set of classes instead of exactly one class. The evaluation of multi-label learning systems is more complex from that of single-label learning systems. A result can be fully correct, partially correct or fully wrong. There exist a number of evaluation criteria that evaluate the performance of a multi-label learning system, given a set $D = \{(\mathbf{x}_1, \mathbf{Y}_1), \dots, (\mathbf{x}_m, \mathbf{Y}_m)\}$ of test examples. The evaluation metrics can be divided into two groups: *prediction-based* and *ranking-based* metrics. Prediction-based metrics assess the *correctness* of the label sets predicted by the multi-label classifier \mathcal{H} , while ranking-based metrics evaluate the label ranking quality depending on a scoring function that attributes a score to each class in Ω [12]. As a scoring function is not computed by all multi-label classification methods, the former category of metrics is of more general use. We will focus in this paper on the hamming loss which is a prediction-based metric regarded as an average of the error rate of the classifier on the Q binary problems where the decision on each label is performed separately [13]. It is defined by:

$$HLoss = \frac{1}{m} \sum_{i=1}^m \frac{|\mathbf{Y}_i \Delta \widehat{\mathbf{Y}}_i|}{Q}, \quad (4)$$

where \mathbf{Y}_i is the ground truth label set for the pattern \mathbf{x}_i , $\widehat{\mathbf{Y}}_i$ is the predicted label set for \mathbf{x}_i and Δ denotes the symmetric difference between two sets. In other words, the Hamming Loss is based on counting prediction errors (an incorrect label is predicted) and missing errors (a true label is not predicted). Note that the value of the Hamming Loss evaluation criterion is in the interval $[0, 1]$. Smaller values of this metric correspond to higher classification quality. We will present hereafter a simple method using this metric conjointly with the EMLkNN in order to purify the training data set.

C. Method description

Let \mathbf{x} denote the feature vector for a new object with unknown set of labels \mathbf{Y} . Let $\mathcal{T} = \{(\mathbf{x}_1, \mathbf{Y}_1), \dots, (\mathbf{x}_n, \mathbf{Y}_n)\}$ be the given training set where $\mathbf{Y}_i \subseteq \Omega = \{\omega_1, \dots, \omega_Q\}$. Another representation of \mathcal{T} suitable to the EMLkNN is $\mathcal{T} = \{(\mathbf{x}_1, A_1, B_1), \dots, (\mathbf{x}_n, A_n, B_n)\}$ where A_i is the set of labels that surely apply to instance i which can be taken as existing labels in \mathbf{Y}_i , and B_i is the set of labels that surely do not apply to the same instance and can be taken as $\Omega \setminus \mathbf{Y}_i$.

We summarize the proposed method to purify the training data set by the following algorithm:

- 1) For each training instance \mathbf{x}_i in \mathcal{T} , find the set $\mathcal{N}_{\mathbf{x}_i}$ of the k nearest neighbors to \mathbf{x}_i ;
- 2) Apply a k -NN based multi-label classifier and calculate a predicted set of labels $\widehat{\mathbf{Y}}_i$ for \mathbf{x}_i ;
- 3) Evaluate the general Hamming Loss $HLoss$:

- if $HLoss$ is less than a predefined threshold, then stop the algorithm,
- else, for each training instance in \mathcal{T} , calculate the associated Hamming Loss given by:

$$HLoss_i = \frac{|\mathbf{Y}_i \Delta \widehat{\mathbf{Y}}_i|}{Q}; \quad (5)$$

- 4) Rank the instances in \mathcal{T} with respect to their $HLoss_i$ and select a subset \mathcal{E}^l containing l instances with the higher Hamming Loss $HLoss_i$;
- 5) Update the training set by deleting those in \mathcal{E}^l : $\mathcal{T} \leftarrow \mathcal{T} \setminus \mathcal{E}^l$;
- 6) Return to step 1.

Note that any k -NN based multi-label classifier may be applied in step 2. In this paper, we only demonstrate results on the EML k NN method introduced in Section II. In this method we associate, according to (1), a two-valued mass function to each of the k neighbors in \mathcal{N}_{x_i} considered as items of evidence. These items of evidence are combined using the conjunctive rule of equation (2), producing a global mass function. This mass function is used according to equation (3) in order to estimate the label set for x_i denoted by $\widehat{\mathbf{Y}}_i$. Intuitively, k should be set to a small value because if k is high, undesirable instances elimination will be occurred on the boundary between different classes. If k is equal to 1, the EML k NN algorithm is reduced to the NN algorithm, and the set of labels to be assigned to an example will be the same as that of his neighbor. In step 3, one can use other stopping criterion than the general $HLoss$. For example, we can stop purifying if the hamming loss associated to each instance is less than a predefined threshold t . We can also substitute the Hamming Loss by another multi-label metric evaluation. In steps 4 and 5, we delete instances with high value of $HLoss_i$ which means deleting the worse instances with respect to a local EML k NN rule. One can add a condition to not delete instances where they belong to classes with low occurrence.

D. Applying EML k NN on purified data set

An independent EML k NN method can be applied using the purified training data set in order to classify unseen instances. The number k of neighbors to be used is not necessarily the same of that used in the purifying algorithm presented in Section III-C. To avoid confusion, the number of neighbors used in the purifying algorithm will be noted by k' if needed. It is easy to see that by using the purified training data set, the number of neighbors to be used by the EML k NN method can be less than that used by the same method on the original training data set before purification. This will decrease the running time of the algorithm since less distances and masses combination have to be calculated. Thus, the objective of using a purified training data set is three-fold: improving the performance of multi-label classifiers, less memory to use for storing training instances, and providing faster decision rule.

IV. EXPERIMENTS

A. Datasets

Our method is experimented with real-world datasets coming from different domains ¹.

- *Emotions dataset*. This dataset consists of 593 examples each one represented by 72 features [14]. Each instance represents a song and is labelled according to the emotions generated.
- *Yeast dataset*. The learning task in the yeast dataset is to identify genes with 14 functional classes describing the yeast *Saccharomyces cerevisiae*. The database contains several hundred functional classes, whose definitions come from biochemical and genetic studies of gene function [15].
- *Medical dataset*. This dataset [16] contains 978 documents for patient symptom histories collected from the Computational Medicine Center concerning a challenge task on the automated processing of clinical free text. Each document is represented by a vector of 1449 features.
- *Corel5k dataset*. This dataset contains images associated with several linguistic terms and widely used for both content based image retrieval and image classification [17]. It includes 5000 Corel images with 374 distinct keywords were used for image annotation. Most of the images have 4 word annotations. This data set is split between 4500 training images and 263 test images.
- *Webpage categorization dataset*. The dataset were collected from the "yahoo.com" domain [18]. Eleven different webpage categorization subproblems are considered, corresponding to 11 independent multi-label categories. Each webpage was represented as a bag of words and normalized to the unit length.

Table I summarizes the characteristics of the emotions, yeast, medical and corel5k datasets, while Table II shows the statistics of the different subproblems within the webpage dataset. The *label cardinality* is the average number of labels per instance, while the *label density* is the label cardinality divided by Q .

B. Evaluation Metrics

Multi-label classification requires different metrics than those used in traditional single-label classification [9]. Even if we used the Hamming Loss metric to purify our training sample, we will show the effect of our method on two other metrics used in the literature to evaluate performance of a multi-label learning classifier. These metrics are introduced hereafter.

Accuracy: The accuracy metric gives an average degree of similarity between the predicted and the ground truth label sets:

$$\text{Accuracy} = \frac{1}{m} \sum_{i=1}^m \frac{|\mathbf{Y}_i \cap \widehat{\mathbf{Y}}_i|}{|\mathbf{Y}_i \cup \widehat{\mathbf{Y}}_i|}. \quad (6)$$

¹Dataset available at <http://mulan.sourceforge.net/datasets.html>.

	Number of instances	Feature vector dimension	Number of labels	Training instances	Test instances	Label cardinality	Label density
emotion	593	72	6	391	202	1.868	0.311
yeast	2417	103	14	1500	917	4.237	0.303
medical	978	1449	45	645	333	1.245	0.028
corel5k	5000	499	374	4500	263	3.522	0.009

TABLE I
CHARACTERISTICS OF EMOTION, YEAST, MEDICAL AND COREL5K DATASETS.

	Number of instances	Feature vector dimension	Number of labels	Label cardinality	Label density
Arts and Humanities	5000	462	26	1.627	0.063
Business and Economy	5000	438	30	1.590	0.053
Computers and Internet	5000	681	33	1.487	0.046
Education	5000	550	33	1.465	0.044
Entertainment	5000	640	21	1.426	0.068
Health	5000	612	32	1.667	0.052
Recreation and Sports	5000	606	22	1.414	0.065
Reference	5000	793	33	1.159	0.035
Science	5000	743	40	1.489	0.036
Social and Science	5000	1047	39	1.274	0.033
Society and Culture	5000	636	27	1.705	0.063

TABLE II
CHARACTERISTICS OF THE WEBPAGE CATEGORIZATION DATASET.

F1-measure: F_1 measure is defined as the harmonic mean of two other metrics called precision and recall. It is defined by:

$$\mathcal{F}_1 = \frac{1}{m} \sum_{i=1}^m \frac{2|\mathbf{Y}_i \cap \hat{\mathbf{Y}}_i|}{|\mathbf{Y}_i| + |\hat{\mathbf{Y}}_i|}. \quad (7)$$

Note that the smaller the value of the *Hamming Loss*, the better the performance. For the other metrics, higher values correspond to better classification quality.

C. Method parameters

The model parameters for the purified algorithm presented in Section III-C are:

- γ - Parameter used in equation 1 to scale the distance to each neighbor. It was fixed at the best value obtained by cross validation using the EMLkNN on the initial training data set.
- k' - Number of neighbors used in the purifying algorithm.
- t - Threshold used to determine the number l of instances to delete. We use in the simulation a Hamming Loss calculated on each instance as explained in Section III-C. This Hamming Loss calculated on only one instance will have a value equals q/Q , where $q \in \{0, \dots, Q\}$. Note that the value of the parameter t to be taken should depend on the global Hamming Loss calculated on the training data set.

The model parameters for the EMLkNN used on test data set are k and γ and are determined using cross validation as explained in [3]. Note that the algorithm presented in Section III-C was repeated only one time by taking a value of t

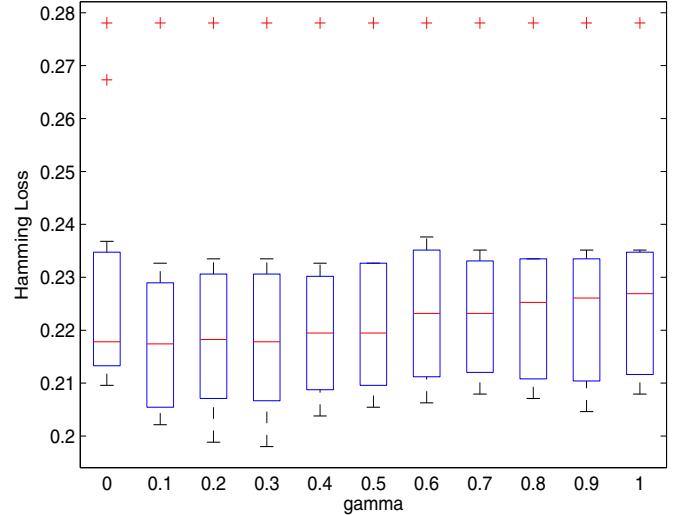


Fig. 1. Hamming Loss for EMLkNN on the initial training set of the emotion data for different values of γ .

sufficiently low to eliminate an important number of erroneous instances at once. We will show the individual influence of each parameter on the *emotion* data set considering other parameters as fixed.

Figure 1 shows the box plot for the Hamming Loss metric obtained by the EMLkNN on the initial data set before purifying for different values of γ , where k was varied from

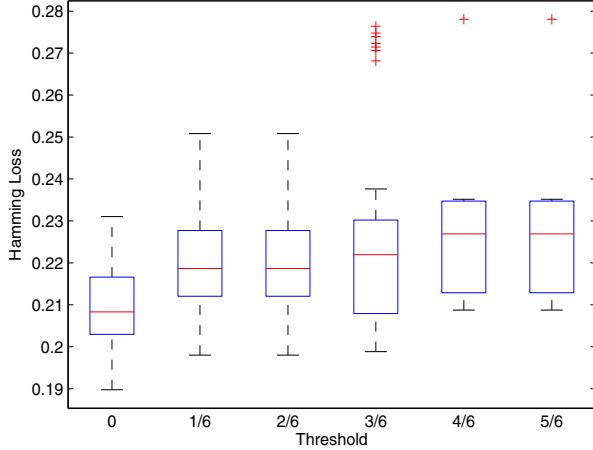


Fig. 2. Hamming Loss measure on the initial training set of the emotion data set as a function of t .

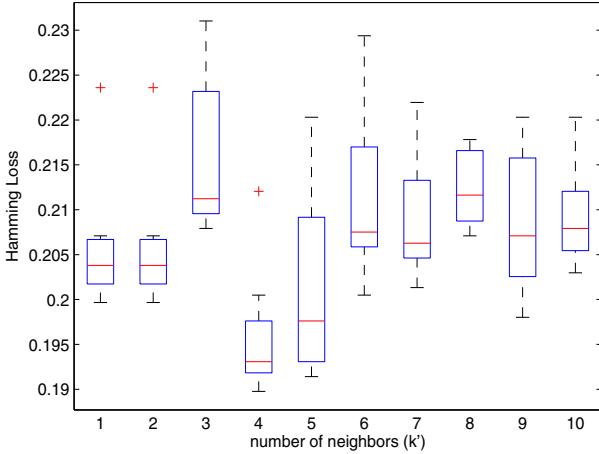


Fig. 3. Hamming Loss measure on the initial training set of the emotion data set as a function of k' .

1 to 12 (thus each box plot corresponds to 12 values of the Hamming Loss obtained for a given γ). Figure 2 shows the Hamming Loss measure obtained as a function of t , where k and k' were varied from 1 to 12 and γ was fixed to 0.1. The box plot in figure 3 shows the Hamming Loss criterion with respect to the number of neighbors k' . k was varied from 1 to 12.

D. Results and Discussion

We plot hereafter the metric measures as a function of the number of neighbors k before and after purifying. k is going from 1 to 12. Figures 4 to 8 show the performance of our method on the emotion, yeast, medical, corel5k and webpage data sets, respectively. For the webpage data set, the average

performance out of the 11 different categorization problems is reported. Based on three evaluation criteria, the purified EML k NN performs better than the classical EML k NN on the different data sets. Even if we use the Hamming Loss criterion for purifying the training data sets, we get better performance on the two other metrics.

Table III shows the number of instances of different training data sets before and after purifying using the proposed algorithm. Table IV shows the best performances for the EML k NN method on the training data set before and after purification. It is clear from this table that the purifying strategy allows us to use smaller number of neighbors with fulfilling in general better performances. Smaller value of k naturally indicates less running time in k -NN based methods as it can be seen also in the same Table.

	Number of training data	
	Before Purifying	After Purifying
emotions	593	113
yeast	2417	832
medical	645	624
corel5k	5000	882
webpage	22000	18678

TABLE III
NUMBER OF TRAINING INSTANCES BEFORE AND AFTER PURIFYING ON
EMOTION, YEAST, MEDICAL, COREL5K AND WEBPAGE DATA SETS.

V. CONCLUSION

In this paper, we propose a simple and original approach to purify training multi-labeled data. The idea remains in deleting erroneous instances by using the Hamming Loss metric. Experimental results on real world data show the effectiveness of this technique. In addition to the performance superiority of the proposed method, other improvements can be observed concerning the running time of the classification algorithm and the memory space to store the training data. Editing the training data set constitutes an original idea in the multi-label classification field. In view of the obtained results, one of our future goals is to apply this strategy conjointly with other multi-label algorithms.

REFERENCES

- [1] T. M. Cover and H. P. E., "Nearest neighbor pattern classification," *IEEE Transactions on Information Theory*, vol. IT-13 (1), pp. 21–27, 1967.
- [2] R. Duda and P. Hart, *Pattern Classification and Scene Analysis*. John Wiley and Sons, New York, 1973.
- [3] T. Denoeux, Z. Younes, and F. Abdallah, "Representing uncertainty on set-valued variables using belief functions," *Artificial Intelligence*, vol. 174, pp. 479–499, 2010.
- [4] Z. Younes, F. Abdallah, and T. Denoeux, "An evidence-theoretic k -nearest neighbor rule for multi-label classification," in *Proc. of the 3rd International Conference on Scalable Uncertainty Management (SUM2009)*, September 2009, Washington, DC, USA, LNAI-5785, pp. 297–308, Springer-Verlag.
- [5] Z. Younes, F. Abdallah, and T. Denoeux, "Evidence multi-label classification approach to learning from data with imprecise labels," in *Proc. of the 13th Int. Conf. on Information Processing and Management of Uncertainty (IPMU 2010)*, June 28–July 2, 2010, Dortmund, Germany, LNAI-6178, pp. 119–128, Springer-Verlag.

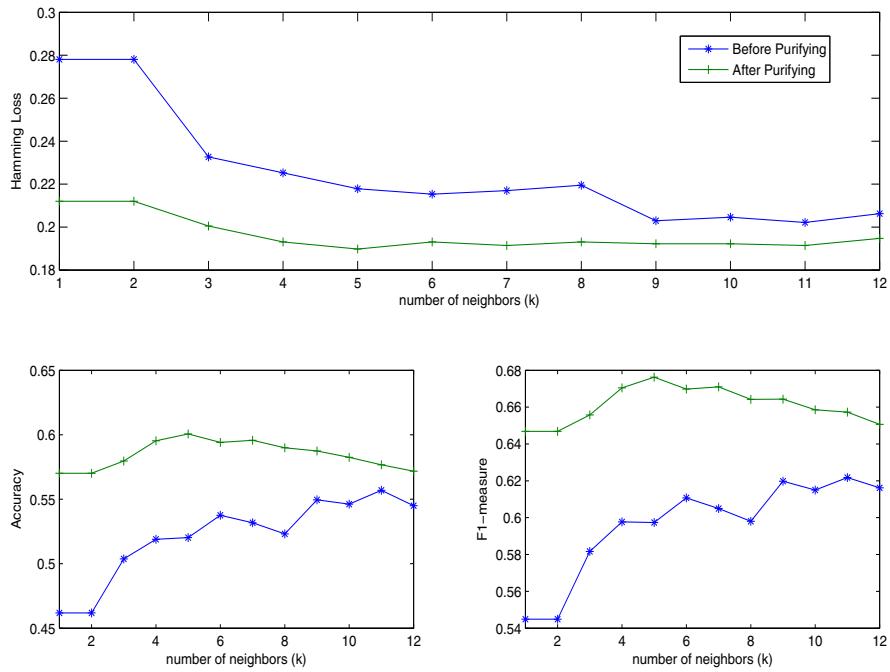


Fig. 4. *Hamming Loss, Accuracy and F1 measures for EMLkNN before and after purifying on the emotion data set.*

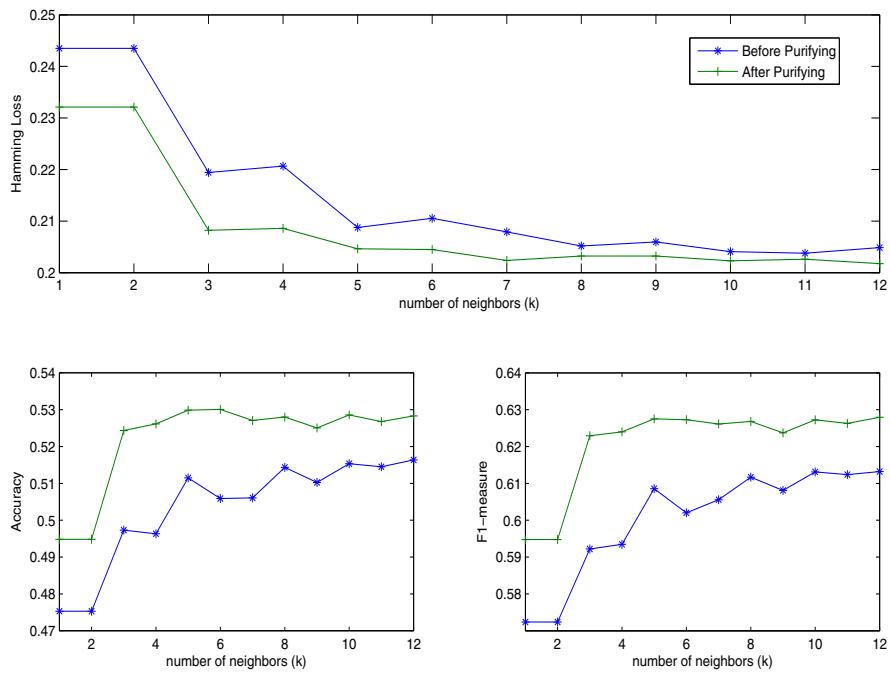


Fig. 5. *Hamming Loss, Accuracy and F1 measures for EMLkNN before and after purifying on the yeast data set.*

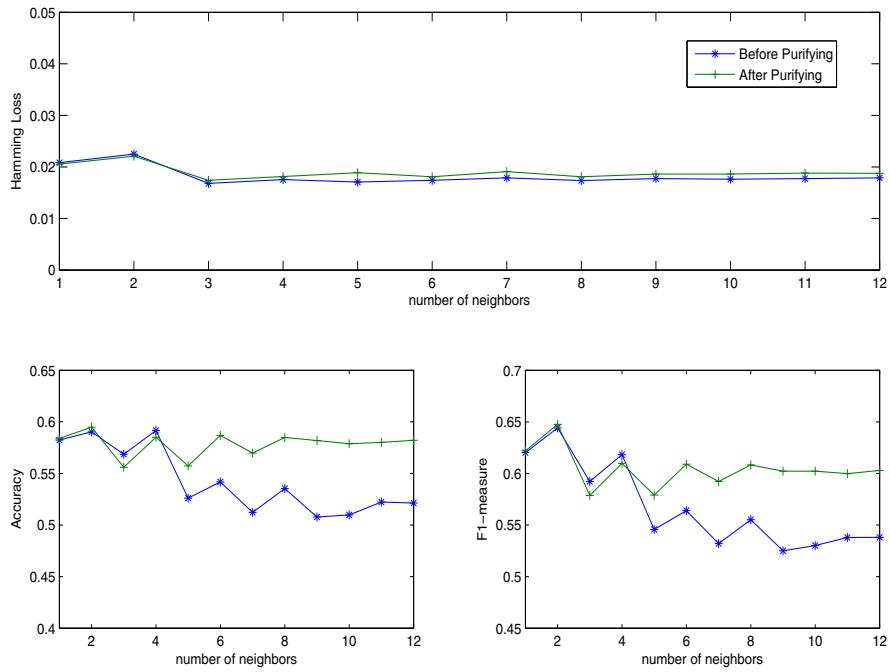


Fig. 6. Hamming Loss, Accuracy and $F1$ measures for EMLkNN before and after purifying on the medical data set.

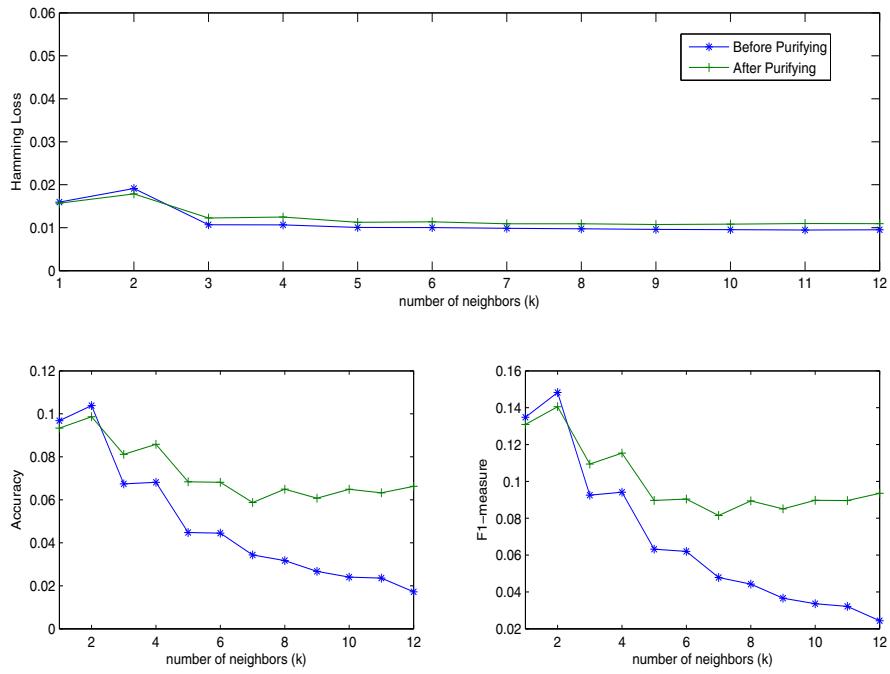
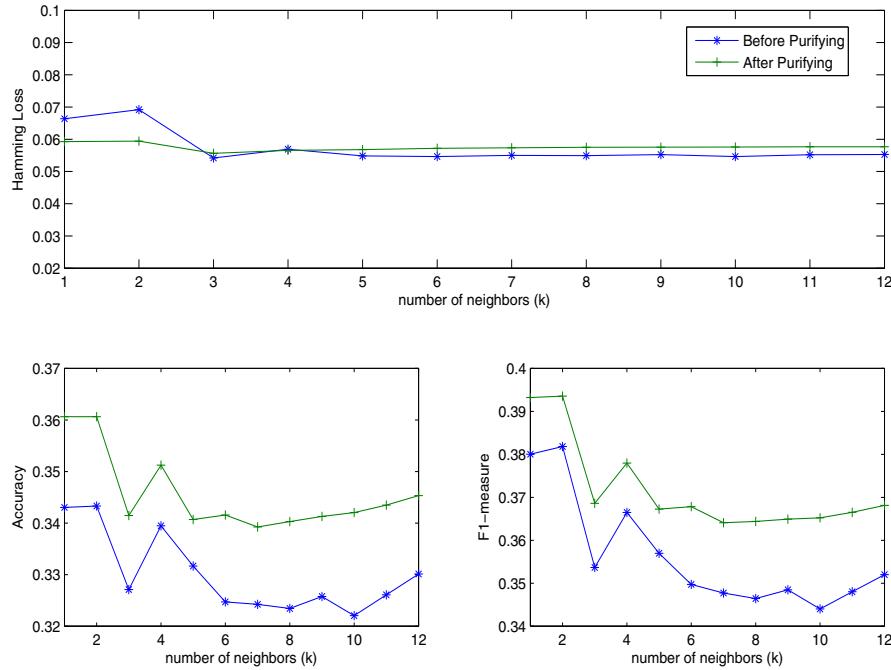


Fig. 7. Hamming Loss, Accuracy and $F1$ measures for EMLkNN before and after purifying on the corel5k data set.

	Before Purifying			After Purifying		
	number of neighbors (k)	Hamming Loss	running time (sec)	number of neighbors (k)	Hamming Loss	running time (sec)
emotions	9	0.203	0.913	5	0.189	0.222
yeast	10	0.211	7.681	7	0.205	3.73
medical	3	0.017	2.891	3	0.017	2.848
corel5k	5	0.010	31.132	5	0.011	23.711
webpage	3	0.056	778.285	3	0.056	606.899

TABLE IV

RUNNING TIME BEFORE AND AFTER PURIFYING FOR THE BEST RESULT ON THE EMOTION, YEAST, MEDICAL AND COREL5K DATA SETS.

Fig. 8. Hamming Loss, Accuracy and F_1 measures for EML k NN before and after purifying on the webpage data set.

- [6] G. D., Y. W., L. Y.K., and L. S., "Nearest neighbor editing aided by unlabeled data," *Information Sciences (SCI), ISSN: 0020-0255*, vol. 179(13), pp. 2273–2282, June 2009.
- [7] J. Koplowitz and T. A. Brown, "On the relation of performance to editing in nearest neighbor rules," *Pattern Recognition*, vol. 13(3), pp. 251–255, 1981.
- [8] D. L. Wilson, "Asymptotic properties of nearest neighbor rules using edited data," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-2 (3), pp. 408–421, 1972.
- [9] G. Tsoumakas and I. Katakis, "Multi-label classification: An overview," *International Journal of Data Warehousing and Mining*, vol. 3(3), pp. 1–13, 2007.
- [10] T. Denoeux, "A k -nearest neighbor classification rule based on dempstershafer theory," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 25 (05), pp. 804–813, 1995.
- [11] L. M. Zouhal and T. Denoeux, "An evidence-theoretic k -nn rule with parameter optimization," *IEEE Transactions on Systems, Man and Cybernetics - Part C*, vol. 28(2), pp. 263–271, 1998.
- [12] 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*, pp. 1–14, 2011.
- [13] R. Schapire and Y. Singer, "Improved boosting algorithms using confidence-rated predictions," *Machine Learning*, vol. 37(3), pp. 297–336, December 1999.
- [14] K. Trohidis, G. Tsoumakas, G. Kalliris, and I. Vlahavas, "Multilabel classification of music into emotions," in *Proc. 9th International Conference on Music Information Retrieval (ISMIR 2008)*, pp. 325–330, 2008.
- [15] P. Pavlidis, J. Weston, J. Cai, and W. N. Grundy, "Combining microarray expression data and phylogenetic profiles to learn gene functional categories using support vector machines," in *Proceedings of the 5th Annual International Conference on Computational Biology*, 2001.
- [16] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," in *Proc. of the 20th European Conference on Machine Learning (ECML 2009)*, 2009.
- [17] C. H. Wei and Y. Li, *Machine learning techniques for adaptive multimedia retrieval: technologies, applications, and perspectives*, C. Wei and Y. Li, Eds. Idea Group Inc (IGI), 2010.
- [18] L. Tang, S. Rajan, and V. Narayanan, "Large scale multi-label classification via metalabeler," in *Proceedings of the 18th international conference on World wide web WWW 09*, 2009.