A *k*-Nearest Neighbor Based Algorithm for Multi-Instance Multi-Label Active Learning

Adrian T. Ruiz(✉), Patrick Thiam, Friedhelm Schwenker, and Günther Palm

Institute of Neural Information Processing, Ulm University,
James-Franck-Ring, 89081 Ulm, Germany
adrian.ruiz@uni-ulm.de

Abstract. Multi-instance multi-label learning (MIML) is a framework in machine learning in which each object is represented by multiple instances and associated with multiple labels. This relatively new approach has achieved success in various applications, particularly those involving learning from complex objects. Because of the complexity of MIML, the cost of data labeling increases drastically along with the improvement of the model performance. In this paper, we introduce a MIML active learning approach to reduce the labeling costs of MIML data without compromising the model performance. Based on a query strategy, we select and request from the Oracle the label set of the most informative object. Our approach is formulated in a pool-based scenario and uses Miml-*k*nn as the base classifier. This classifier for MIML is based on the *k*-Nearest Neighbor algorithm and has achieved superior performance in different data domains. We proposed novel query strategies and also implemented previously used query strategies for MIML learning. Finally, we conducted an experimental evaluation on various benchmark datasets. We demonstrate that these approaches can achieve significantly improved results than without active selection for all datasets on various evaluation criteria.

Keywords: Multi-instance · Multi-label · Active learning
k nearest neighbors · Partially supervised learning
Acoustic classification of birds · Text categorization
Scene classification

1 Introduction

In standard supervised learning, an object consists of a single instance, represented by a feature vector, and is associated with a single class label. This framework is known as single-instance single-label (SISL) learning. The goal of SISL learning is to train a classifier model which learns from training instances how to assign a class label to any feature vector. However, in many real applications, such a learning framework is less convenient to model complex objects,

L. Pancioni et al. (Eds.): ANNPR 2018, LNAI 11081, pp. 139–151, 2018.
https://doi.org/10.1007/978-3-319-99978-4_11

which intrinsic representation is a collection of instances. Likewise, these complex objects may also be associated simultaneously with multiple class labels. For example, a scene image may comprise images of mountains, lakes, and trees, and we may associate it with the labels *Landscape* and *Summer* at the same time. If we extract a single instance to represent it, some useful information may get lost. In another approach, we can segment the image into multiple regions and extract one instance from each region of interest. Another example could be in text categorization tasks where a document may be annotated with multiple labels. To fully exploit the content with multiple topics, it would be more advantageous if we represent each paragraph with one instance. Zhou and Zhang [22] introduced multi-instance multi-label (MIML) learning, where each object is represented by a *bag of multiple instances* (feature vectors with fixed-length), and each object is associated with a *set of class labels*. Several algorithms for MIML have been proposed and achieved better performance in image and text classification, in comparison to conventional methods adapted for MIML classification. Other successful applications include genome protein function prediction [18], gene expression patterns annotation [20], relationship extraction [15], video understanding [19], classification of bird species [1,2], and predicting tags for web pages [14].

In most cases of supervised learning, it is necessary to use large amounts of training examples to obtain accurate models. Nevertheless, it is a typical situation that the costs of manually labeled data are expensive or time-consuming. Active learning is an approach of a partially-supervised learning algorithm [3,4,10] that reduces the required amount of training data without compromising the model performance. This goal is accomplished by selecting the most informative examples from the unlabeled examples and query their label from an oracle (expert). Pool-based sampling is the most common scenario in active learning in which queries are drawn from a static or closed pool of unlabeled examples. Many active learning strategies have been proposed to estimate the informativeness of unlabeled samples [13,17]. These query strategies are based on different measures, e.g., uncertainty, expected error reduction and information density. A comprehensive literature survey on query strategies is provided by Settles [12].

For MIML datasets, the cost of labeled data depends on the maximum amount of possible labels for a bag of instances. In some applications, MIML provides a major advantage because it is easier or less costly to obtain labels at the bag-level than at instance-level. Nevertheless, because of their multiplicity in the input and output spaces, the required amount of training data to improve the accuracy model increases dramatically. For this reason, it is of great interest to implement active learning algorithms in a MIML framework. Currently, few studies have proposed active learning methods for MIML. Retz and Schwenker [9] use MimlSvm [23] as the base classifier in which the MIML data is reduced to a bag-level output vector. This representation is later used to formulate an active learning strategy. Another proposed method uses MimlFast as base classifiers and the approach actively queries the most valuable information by exploiting diversity and uncertainty in both the input and output spaces [5].

The efficiency of an active learning algorithm relies not only on the query strategy design but also on the selection of the base classifier. Two of the most commonly used classifiers are MimlBoost and MimlSvm [22,23]. Nevertheless, MimlBoost can handle only small datasets and does not yield good performance in general [6]. MimlSvm reaches a satisfying classification accuracy for text and image, but usually not for other types of data sets [1,6]. A better alternative is Miml-$k$nn[21] (Multi-Instance Multi-Label k-Nearest Neighbor) which combines the well-known k-Nearest Neighbor technique with MIML. Given a test example, Miml-$k$nn not only considers its κ neighbors but also considers its κ' citers, i.e., examples that consider the test example within their κ' nearest neighbors. The identification of neighbors and citers relies on the Hausdorff distance which is an estimation of the distances between bags. One advantage of using Miml-$k$nn with pool-based sampling is that the distance between all bags (i.e., labeled and unlabeled bags) can be precomputed and stored for later use in any model learning or prediction. Beside this, Miml-$k$nn classifiers have achieve a superior performance than the MimlSvm and MimlBoost for different types of data such as text [11,21], image [21,22], and bio-acoustic data [1].

In this paper, we introduce an active multi-instance multi-label learning approach within a pool-based scenario and use Miml-$k$nn as the base classifier. This method aims to reduce the amount of training MIML data needed to achieve the highest possible classification performance. This paper presents two major contributions to active learning and MIML learning. First, we motivate and introduce several new query strategies within the MIML framework. Later we conduct an empirical study of our proposed active learning methods on a variety of benchmark MIML data.

The remainder of this paper is organized as follows. Section 2 describes in detail the proposed approach. Section 3 describes the experiments and presents their results, followed by conclusions in Sect. 4.

2 Method

2.1 MIML Framework

In a MIML framework, an example X consists of a *bag* of instances $X = \{\mathbf{x}_j\}_{j=1}^{m}$ where m is the number of instances and each instance $\mathbf{x}_j = [x_1, \ldots, x_D]$ is a D-dimensional feature vector. The number of instances m can variate among bags. In this framework, each bag X can be associated to one or more labels and they are represented by a *label set* $Y = \{y_k\}$ where $k \in \{1, \ldots, K\}$. For our purposes, Y is represented by a *label indicator vector* $\mathbf{I} = [I_1, \ldots, I_K]$ where the entry $I_k = 1$ if $y_k \in Y$ and $I_k = 0$ otherwise. Given a fully labeled training set $\mathcal{L} = \{(X_l, Y_l)\}_{l=1}^{L}$, the learning task in a MIML framework is to train a classification model which is a function $h : 2^{\mathcal{X}} \rightarrow 2^{\mathcal{Y}}$ that maps a set of instances $X \in \mathcal{X}$ to a set of labels $Y \in \mathcal{Y}$.

MIML algorithms such as MimlSvm, MimlRbf and Miml-$k$nn reduce the MIML problem to a single-instance multi-label problem by associating each bag X with a bag-level feature vector $\mathbf{z}(X) \in \mathbb{R}^K$ which combines information from the instances in the bag. Each algorithm uses different approaches to compute a bag-level feature vector. Nevertheless all these methods heavily depend on the use of some form of bag-level distance measure. The most common choice is the Hausdorff distance $D_H(X, X')$. Retz and Schwenker [9] examined several variations of this distance. For this paper we consider the maximum D_H^{max}, median D_H^{med} and average D_H^{avg} Hausdorff distances defined as:

$$D_H^{max}(X, X') = \max\left\{ \max_{\mathbf{x}\in X} \min_{\mathbf{x}'\in X'} d(\mathbf{x}, \mathbf{x}'), \max_{\mathbf{x}'\in X'} \min_{\mathbf{x}\in X} d(\mathbf{x}, \mathbf{x}') \right\} \tag{1a}$$

$$D_H^{med}(X, X') = \frac{1}{2}\left(\operatorname*{median}_{\mathbf{x}\in X} \min_{\mathbf{x}'\in X'} d(\mathbf{x}, \mathbf{x}'), \operatorname*{median}_{\mathbf{x}'\in X'} \min_{\mathbf{x}\in X} d(\mathbf{x}, \mathbf{x}') \right) \tag{1b}$$

$$D_H^{avg}(X, X') = \frac{1}{|X|+|X'|}\left(\sum_{\mathbf{x}\in X} \min_{\mathbf{x}'\in X'} d(\mathbf{x}, \mathbf{x}') + \sum_{\mathbf{x}'\in X'} \min_{\mathbf{x}\in X} d(\mathbf{x}, \mathbf{x}') \right) \tag{1c}$$

where $d(\mathbf{x}, \mathbf{x}') = \|\mathbf{x} - \mathbf{x}'\|$ is the Euclidean distance between instances.

2.2 MIML-kNN

In the following we describe Miml-$k$nn algorithm [21]. Given an example bag X and a training set $\mathcal{L} = \{(X_l, Y_l)\}$, first we identify in the training bags $\mathcal{X}_L = \{X_l\}$, the κ *nearest neighbors*, and the κ' *citers* of X by employing the Hausdorff metric $D_H(X, X')$. This means that we have to identify the neighbors set $\mathcal{N}_\kappa(X)$ and the citers set $\mathcal{C}_{\kappa'}(X)$. These sets are defined as follows

$$\mathcal{N}_\kappa(X) = \{A | A \text{ is one of } X\text{'s}\, \kappa \text{ nearest neighbors in } \mathcal{X}_\mathcal{L}\} \tag{2a}$$

$$\mathcal{C}_{\kappa'}(X) = \{B | X \text{ is one of } B\text{' s}\kappa' \text{ nearest neighbors in } \mathcal{X}_\mathcal{L} \cup \{X\}\} \tag{2b}$$

The citers bags are the bags that consider X to be one of their κ' nearest neighbors. After the computation of $\mathcal{N}_\kappa(X)$ and $\mathcal{C}_{\kappa'}(X)$, we defined a *labeling counter vector* $\mathbf{z}(X) = [z_1(X), \ldots, z_K(X)]$ where the entry $z_k(X)$ is the number of bags in $\mathcal{Z}(X) = \mathcal{N}_\kappa(X) \cup \mathcal{C}_{\kappa'}(X)$ that include label y_k in their label set. Using the binary label vector $\mathbf{I}(X)$, $\mathbf{z}(X)$ is defined as

$$\mathbf{z}(X) = \sum_{X'\in\mathcal{Z}(X)} \mathbf{I}(X') \tag{3}$$

Later, the information contained in $\mathbf{z}(X)$ is used to obtain the *predicted label set* $\hat{Y}$ associated to X by employing a *prediction function* $\mathbf{f}(X) = [f_1(X), \ldots, f_K(X)]$ such that

$$f_k(X) = \mathbf{w}_k^\top \cdot \mathbf{z}(X) \tag{4}$$

where $\mathbf{w}_k^\top$ is the kth transposed column of the *weight matrix* $\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_K]$. The classification rule is that the label $\hat{y}_k$ belongs to the *predicted label set*

$\hat{Y}(X) = \{\hat{y}_k\}$ only if $f_k(X) > 0$. Hence, for the *predicted indicator vector* $\hat{\mathbf{I}}(X) = \left[\hat{I}_1, \ldots, \hat{I}_K\right]$ the entry $\hat{I}_k = 1$ if $f_k(X) > 0$ and $\hat{I}_k = 0$ otherwise. The values of $\mathbf{W}$ are computed using a linear classification approach by minimizing the following sum-of-squares error function

$$E = \frac{1}{2}\sum_{l=1}^{L}\sum_{k=1}^{K}\left(w_K^\top \cdot \mathbf{z}(X_l) - y_k(X_l)\right)^2 \tag{5}$$

This error minimization implies to solve the weight matrix $\mathbf{W}$ as in a least sum-of-squares problem of the form $\left(\mathbf{Z}^\top\mathbf{Z}\right)\mathbf{W} = \mathbf{Z}^\top\mathbf{Y}$. In this case, the matrix $\mathbf{W}$ is computed using a linear matrix inversion technique of singular value decomposition.

2.3 Active Learning

In this part, we present the strategies of active learning for a multi-instance multi-label data set using Miml-$k$nn as the base classifier. Initially we have a set of *labeled data* $\mathcal{L} = \{(X_l, Y_l)\}$ with L labeled bags and a set of *unlabeled data* $\mathcal{U} = \{X_u\}$ with U unlabeled bags. In an active learning scenario, usually the amount of unlabeled data is much larger than the amount of labeled data, i.e. $U \gg L$. The main task of an active learning algorithm is to select the *most informative bag* X^* according to some *query strategy* $\phi(X)$, which is a function evaluated on each example X from the pool $\mathcal{U}$. In this work, the selection of the bag X^* is done according to

$$X^* = \underset{X \in \mathcal{U}}{\operatorname{argmax}}\, \phi(X) \tag{6}$$

Algorithm 1 describes the pool-based active learning algorithm for training a Miml-$k$nn model. One advantage of using Miml-$k$nn with pool-based sampling, is that, the distance between all bags (i.e. labeled and unlabeled bags) can be precomputed and stored for later use in any model learning or prediction task. As in Algorithm 1, first we calculated the *bag distance matrix* $\mathbf{D}$ such that $d_{ij} = D_H(X_i, X_j)$ for all bags X_i, X_j. Then from this matrix we can extract the distance submatrix $\mathbf{D}_{\mathcal{L}}$ of the labeled bags and use it in the training of a Miml-$k$nn model (see Eq. 5). For classification of the bag X, we have to feed the trained Miml-$k$nn model with the subtracted matrix $\mathbf{D}_{\mathcal{L}\cup\{X\}}$ (see Eq. 2). In the following, we describe in detail the query strategies we proposed which will be later compared in an empirical study.

Uncertainty Sampling (Unc). This approach is one of the most common in SISL framework. Here a learner queries the instance that is most uncertain how to label. For a muti-label problem we define the uncertainty as $\phi(X) = 1 - P(\hat{Y}|X)$ where $P(\hat{Y}|X)$ is the *bag posterior probability* for the predicted label set $\hat{Y}$ given the bag X. We calculate $P(\hat{Y}|X)$ as the probability given the

Algorithm 1. Active *k*MIML

```
input:
  $\mathcal{L}$: Labeled data set $\{(X_l, Y_l)\}$
  $\mathcal{U}$: Unlabeled data set $\{X_u\}$
  $\kappa$ : Neighbors parameter
  $\kappa'$: Citers parameter
output:
  $h$ : Miml-$k$nn model
begin
    Calculate the distance matrix $\mathbf{D}$ using $D_H(X_i, X_j)$ for all bags $X_i, X_j \in \{X_u, X_l\}$
    Train a Miml-$k$nn model $h$ on $\mathcal{L}$ using $\kappa, \kappa'$ and $\mathbf{D}_{\mathcal{L}}$
repeat
    Classify each bag $X \in \mathcal{U}$ with trained Miml-$k$nn model $h$ using $\kappa, \kappa'$,$\mathbf{D}_{\mathcal{L} \cup \{X\}}$
    Calculate $\phi(X)$ for all $X$
    Select the *most informative* bag $X^*$ with $\arg\max \phi(X)$
    Request the label set $Y^*$ for $X^*$
    Remove $X^*$ from $\mathcal{U}$
    Add $(X^*, Y^*)$ to $\mathcal{L}$
    Train a Miml-$k$nn model on $\mathcal{L}$ using $\kappa, \kappa'$ and $\mathbf{D}_{\mathcal{L}}$
until *stop criterion reached*
```

combination of labels $\hat{y}_k$ founded in $\hat{Y}(X)$. For this we use a *single-label posterior probability* $P(\hat{y}_k|X)$ to estimate the uncertainty $\phi(X)$ as

$$\phi(X) = 1 - \prod_{\hat{y}_k \in \hat{Y}} P(\hat{y}_k|X) \tag{7}$$

The Miml-$k$nn classifier output for the kth label is a prediction function $f_k(X)$. This function outputs higher positive or lower negative values for very certain positive or negative predictions respectively. Considering Eq. 4, this means that when $|f_k(X)| \gg 0$ the vectors $\mathbf{w}_k^\top$ and $\mathbf{z}$ are linearly codependent. For the most uncertain label prediction then $|f_k(X)| \approx 0$ which means that $\mathbf{w}_k^\top$ and $\mathbf{z}$ are linearly independent. Based on this, we estimate $P(\hat{y}_k|X)$ using a normalization on $f_k(X)$ using the Cauchy–Schwarz inequality as follows

$$P(\hat{y}_k|X) = \frac{1}{2}\left(\frac{\mathbf{w}_k^\top \cdot \mathbf{z}(X)}{\|\mathbf{w}_k^\top\|\|\mathbf{z}(X)\|} + 1\right) \tag{8}$$

Diversity (Div). This method is based on the multi-label active learning method proposed by Huang et al. [5,6]. This method considers that the most informative bags are those where the number of predictions are inconsistent with the average of predicted labels in the training set. Using the indicator vector $\hat{\mathbf{I}}\left(X\right)$, $\phi\left(X\right)$ is formulated as follows

$$\phi\left(X\right)=\left|\frac{1}{K}\sum_{k=1}^{K}\hat{I}_k\left(X\right)-\rho_{\mathcal{L}}\right| \tag{9}$$

where

$$\rho_{\mathcal{L}}=\frac{1}{LK}\sum_{l=1}^{L}\sum_{k=1}^{K}I_k\left(X_l\right) \tag{10}$$

Margin (Mrg). A high positive (or low negative) value of $f_k(X)$ means that the model has a high certainty that X positively (or negatively) belongs to the kth class. Meanwhile lower absolute values in $f_k(X)$ indicate a high uncertainty. This strategy chooses the bag which average output values are the nearest to zero. This means

$$\phi\left(X\right)=-\frac{1}{K}\sum_{k=1}^{K}\left|f_k\left(X\right)\right| \tag{11}$$

Range (Rng). This method is similar to the *margin* query strategy. In this case is considered that lower range of output values $f_k\left(X\right)$ indicates higher uncertainty. This strategy is defined as

$$\phi\left(X\right)=-\left(\max_k f_k\left(X\right)-\min_k f_k\left(X\right)\right) \tag{12}$$

Percentile (Prc). This approach is related to ExtMidSelect used by Retz und Schwenker [9]. This method measures the distance between the upper and lower values of $\mathbf{f}\left(X\right)=\left[f_1,\ldots,f_K\right]$ delimited by the percentile value $F_p\left(X\right)=\text{percentile}(\mathbf{f}\left(X\right),p)$ at the percentage $p=100\left(1-\rho_{\mathcal{L}}\right)\%$, see Eq. 10. The strategy is defined as

$$\phi\left(X\right)=-\left|F_{\uparrow}\left(X\right)-F_{\downarrow}\left(X\right)\right| \tag{13}$$

where $F_{\uparrow}\left(X\right)$ and $F_{\downarrow}\left(X\right)$ are respectively the conditional means of the upper and lower values, this means $F_{\uparrow}\left(X\right)=E\left[\mathbf{f}\left(X\right)|f_k\geq F_p\right]$ and $F_{\downarrow}\left(X\right)=E\left[\mathbf{f}\left(X\right)|f_k<F_p\right]$.

Information Density (IDC & IDH). It has been suggested that uncertainty based strategies for SISL are prone to querying outliers. To address this problem, Settles et al. [13] proposed a strategy that favors uncertain samples nearest to clusters of unlabeled samples. This strategy uses a similarity measure $S\left(X\right)$ and an uncertainty sampling $\phi_u\left(X\right)$ such that

$$\phi\left(X\right)=\phi_u\left(X\right)\cdot S\left(X\right) \tag{14}$$

Table 1. Statistics on data sets used in experiments

					Instances per bag			Labels per bag		
Dataset	Bags	Labels	Inst.	Feat.	*min*	*max*	*mean* ± *std.*	*min*	*max*	*mean* ± *std.*
Birds	548	13	10,232	38	2	43	8.7 ± 7.9	1	5	2.1 ± 1.0
Scene	2,000	5	18,000	15	9	9	9.0 ± 0.0	1	3	1.2 ± 0.4
Reuters	2,000	7	7,119	243	2	26	3.6 ± 2.7	1	3	1.2 ± 0.4
CK+	430	79	7,915	4,391	4	66	18.4 ± 7.6	2	9	4.0 ± 1.5
UnitPro(G.s.)	379	340	1,250	216	2	8	3.1 ± 1.2	1	69	4.0 ± 7.0

The uncertainty factor $\phi_u(X)$ is formulated as in Eq. 7. We defined two types of similarity measures. The first approach (IDC) is based on a *cosine distance* using the formula

$$\cos(X, X') = \frac{\tilde{\mathbf{x}} \cdot \tilde{\mathbf{x}}'}{\|\tilde{\mathbf{x}}\| \|\tilde{\mathbf{x}}'\|} \tag{15}$$

where $\tilde{\mathbf{x}}$ is a bag-level vector that is the mean of features over all instances $\mathbf{x}_j \in X$, this is $\tilde{\mathbf{x}} = (1/m)\sum_{j=1}^{m} \mathbf{x}_j$ where $m = |X|$. The similarity measure based on cosine distance is defined as

$$S(X) = \frac{1}{U} \sum_{X' \in \mathcal{U}} \cos(X, X') \tag{16}$$

The second approach (IDH) is based on the *Hausdorff distance* from Eq. 1. The similarity measure is defined as

$$S(X) = 1 - \frac{\exp\left(\bar{D}_U(X)\right)}{\sum_{X' \in \mathcal{U}} \exp\left(\bar{D}_U(X')\right)} \tag{17}$$

where $\bar{D}_{\mathcal{U}}(X)$ is the mean distance between the bag X and the unlabeled bags, this is $\bar{D}_{\mathcal{U}}(X) = (1/U)\sum_{u=1}^{U} D_H(X, X_u)$. In order to have comparable measures we applied on $\bar{D}_{\mathcal{U}}(X)$ a softmax averaging.

3 Experiments

We conduct a series of experiments to compare the performance of each of the query strategies presented in this work. We employed five MIML benchmark datasets including *Birds* [1,2], *Reuters* [11], *Scene* [22], *CK+* [7,8] and *Unit-Pro(G.s.)* [16,18]. A summary of the datasets is presented in Table 1. All data sets are publicly available and prepared as MIML datasets except for the *CK+* dataset. We extracted this last one from the *Cohn-Kanade* dataset and the labels correspond to action units categories. A bag represents an image sequence and

Table 2. MIML-kNN parameters

	Parameters			Performance							
Dataset	D_H	κ	κ'	*h.l.* ↓	*r.l.* ↓	*o.e.* ↓	*co.* ↓	*a.a.* ↑	*a.p.* ↑	*a.r.* ↑	*a.f_1* ↑
Birds	*med*	5	15	0.100	0.080	0.138	2.633	0.431	0.764	0.780	0.781
Scene	*med*	1	9	0.171	0.182	0.340	0.975	0.463	0.620	0.575	0.597
Reuters	*max*	5	17	0.037	0.031	0.078	0.355	0.820	0.895	0.910	0.903
CK+	*max*	43	19	0.034	0.124	0.198	28.14	0.163	0.757	0.544	0.633
UnitPro(G.s.)	*avg*	43	11	0.025	0.356	0.653	175.9	0.267	0.237	0.297	0.263

we extracted appearance based (local binary patterns) and shape based (histogram of oriented gradients) features at each image. *UnitPro(G.s.)* dataset is a complete proteome of the bacteria *Geobacter sulfurreducens* downloaded from the UniProt databank [16].

For each dataset, we randomly sample 20% of bags as the test data, and the rest as the unlabeled pool for active learning. Before the active learning tasks, 5% of the unlabeled pool is randomly labeled to train an initial MIML-kNN model. After each query, we train a MIML-kNN model with the extended labeled data and we test the performance of this model on the test set. Additionally, we run an experiment with a bag random sampling and use it as a reference. We run each experiment until we label 50% of the original unlabeled pool. In the experiments, a simulated Oracle provides the labels requested. We repeat the experiment 30 times for each of the datasets. The performance of the MIML-kNN models using active learning was estimated with eight measures: *hamming loss, ranking loss, coverage, one error, average accuracy, average precision, average recall* and *average f_1-measure* (see [1,22,23]). These measures are common performance metrics for evaluation in MIML framework. Lower values for *hamming loss, ranking loss, coverage* and *one error* imply a better performance and vice-versa for the other four measures.

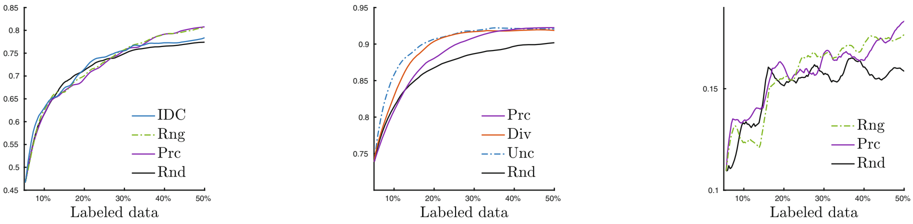
For each data set we tuned the number of neighbors κ, the number of citers κ' and the type of Hausdorff distance D_H to obtain a maximum model performance. We perform a cross-validation test over all combinations of $(\kappa, \kappa') \in \{1, 3, 5, \ldots, 75\}^2$ with $D_H \in \{D_H^{max}, D_H^{avg}, D_H^{med}\}$. For each combination we tested 30 replicas with 20% and 80% of the data randomly selected as testing and training set respectively. At last, we selected the parameters setting that maximizes the *average f_1-measure*. The results of the parameter tuning are reported in Table 2.

The results of the performance experiments are shown in Table 3. The black dot (●) indicates that the performance is significantly better than the bag random sampling (Rnd). The white dot (○) indicates the opposite case. Regarding the query strategy, we observe that among all datasets several strategies have superior performance than Rnd. The information density based approaches (IDD & IDH) in *UnitPro(G.s.)* and *Scene* have significantly worse performance. In contrast, these strategies performed better using the *CK+* and *Birds* dataset. The

Table 3. Comparison of query strategies at 50% of data labeled. ↑ (↓) indicate that higher (lower) values imply a better performance. • (◦) indicate that the query strategy is significantly better (worse) than a random bag sampling (Rnd) based on a paired t-test at the 5% significance level ($p < 0.05$).

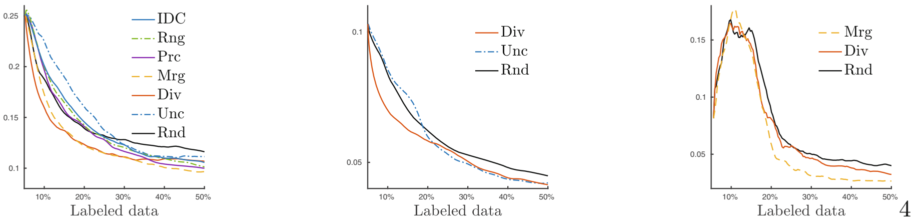
	Rnd	Unc	Div	Mrg	Prc	Rng	IDC	IDH
Birds								
h.l. ↓	0.116	0.111•	0.107•	**0.097•**	0.100•	0.101•	0.106•	0.116
r.l. ↓	0.099	0.093•	0.089•	**0.077•**	0.077•	0.079•	0.086•	0.091•
o.e. ↓	0.188	0.183	0.173•	0.163•	**0.157•**	0.158•	0.178	0.189
co. ↓	2.889	2.804	2.752•	2.559•	**2.552•**	2.584•	2.702•	2.761•
a.a. ↑	0.730	0.720	0.724	0.718	0.767•	**0.768•**	0.738	0.731
a.p. ↑	0.821	0.826	0.835•	0.850•	**0.852•**	0.848•	0.835•	0.822
a.r. ↑	0.730	0.720	0.724	0.718	0.767•	**0.768•**	0.738	0.731
$a.f_1$ ↑	0.773	0.769	0.775	0.778	**0.807•**	0.806•	0.783•	0.774
Scene								
h.l. ↓	0.196	0.204◦	0.200◦	**0.187•**	0.190•	0.191•	0.205◦	0.209◦
r.l. ↓	0.210	0.221◦	0.213	**0.191•**	0.193•	0.195•	0.221◦	0.226◦
o.e. ↓	0.380	0.396◦	0.383	**0.352•**	0.362•	0.363•	0.396◦	0.404◦
co. ↓	1.100	1.140◦	1.110	**1.036•**	1.039•	1.046•	1.140◦	1.160◦
a.a. ↑	0.493	0.492	0.496	0.470◦	0.496	**0.506•**	0.487	0.494
a.p. ↑	0.754	0.744◦	0.752	**0.771•**	0.767•	0.766•	0.744◦	0.739◦
a.r. ↑	0.493	0.492	0.496	0.470◦	0.496	**0.506•**	0.487	0.494
$a.f_1$ ↑	0.596	0.592	0.597	0.584◦	0.603	**0.609•**	0.588	0.592
Reuters								
h.l. ↓	0.045	0.042•	**0.041•**	0.050◦	0.048◦	0.051◦	0.104	0.104
r.l. ↓	0.039	0.035•	0.034•	0.044◦	**0.033•**	0.039	0.121	0.121
o.e. ↓	0.100	0.087•	**0.085•**	0.120◦	0.090•	0.106◦	0.274	0.274
co. ↓	0.409	0.387•	0.381•	0.436◦	**0.374•**	0.407	0.916	0.916
a.a. ↑	0.872	0.901•	0.896•	0.826◦	**0.905•**	0.883•	0.675	0.675
a.p. ↑	0.934	0.941•	**0.943•**	0.923◦	0.941•	0.931	0.816	0.816
a.r. ↑	0.872	0.901•	0.896•	0.826◦	**0.905•**	0.883•	0.675	0.675
$a.f_1$ ↑	0.902	0.921•	0.919•	0.871◦	**0.922•**	0.906	0.738	0.738
CK+								
h.l. ↓	0.041	0.040	0.040	0.040	0.043◦	0.042◦	**0.039•**	0.040•
r.l. ↓	0.163	0.152•	0.150•	0.157	0.157	0.157	**0.146•**	0.149•
o.e. ↓	0.270	0.246•	0.247•	0.268	0.264	0.263	0.250	**0.240•**
co. ↓	32.84	31.98	31.00•	32.19	32.31	32.08	**30.54•**	30.97•
a.a. ↑	0.492	0.514•	0.520•	0.514•	0.524•	**0.526•**	0.511•	0.500
a.p. ↑	0.599	0.615•	0.617•	0.609•	0.605	0.607	0.622•	**0.622•**
a.r. ↑	0.492	0.514•	0.520•	0.514•	0.524•	**0.526•**	0.511•	0.500
$a.f_1$ ↑	0.540	0.560•	**0.564•**	0.557•	0.561•	0.563•	0.561•	0.554•
UnitPro(G.s.)								
h.l. ↓	0.040	0.043	0.032•	**0.027•**	0.064◦	0.061◦	0.076◦	0.086◦
r.l. ↓	0.503	0.496	**0.494**	0.498	0.501	0.514	0.531◦	0.519
o.e. ↓	0.834	0.826	0.819	**0.811•**	0.824	0.828	0.865◦	0.866◦
co. ↓	196.9	192.3	192.5	**187.6•**	189.9•	192.6	212.5◦	201.7
a.a. ↑	0.180	0.202•	0.181	0.170	**0.221•**	0.202•	0.185	0.206•
a.p. ↑	0.141	0.148	0.154	**0.168•**	0.158•	0.153	0.101◦	0.108◦
a.r. ↑	0.180	0.202•	0.181	0.170	**0.221•**	0.202•	0.185	0.206•
$a.f_1$ ↑	0.157	0.170	0.166	0.168	**0.183•**	0.173•	0.129◦	0.141◦

best performance among all datasets is achieved by the percentile strategy (Prc) followed by margin (Mrg) and diversity (Div) strategies. Regarding the dataset, in the *Reuters* and *UnitPro(G.s.)* dataset we observe in general a remarkable performance of the strategies. In the *Reuters* dataset, uncertainty (Unc) and diversity (Div) strategies are significantly better for all metrics.



(a) *Birds* (b) *Reuters* (c) *UnitPro(G.s.)*

Fig. 1. Example of query strategies performance based on the *average* f_1*-measure*



(a) *Birds* (b) *Reuters* (c) *UnitPro(G.s.)*

Fig. 2. Example of query strategies performance based on the *hamming loss*

Figures 1 and 2 shows the performance curves as the number of labeled data increases until the stop criterion is reached (50% labeled). We show a selection of the most representative curves based on the *avg.* f_1*-measure* and *hamming loss* metrics. We observe in Fig. 1b that the MIML-kNN model can reach its best performance with much less labeled data (∼25%) using uncertainty (Unc) or percentile (Prc) query strategies. A similar situation can be observed in Fig. 2c where the MIML-kNN reaches nearly the lowest *hamming loss* at approx. 35% of labeled data using the margin (Mrg) query strategy.

4 Conclusion

In this paper we proposed an active learning approach to reduce the labeling cost of the MIML dataset using MIML-kNN as base classifier. We introduced novel query strategies and also implemented previously used query strategies for MIML learning. Finally, we conducted an experimental evaluation on various benchmark datasets. We demonstrated that these approaches can achieve significantly improved results than no active selection for all datasets on various evaluation criteria.

References

1. Briggs, F., Fern, X.Z., Raich, R.: Rank-loss support instance machines for MIML instance annotation. In: Proceedings of 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2012, p. 534. ACM Press, New York (2012)
2. Briggs, F., et al.: Acoustic classification of multiple simultaneous bird species: a multi-instance multi-label approach. J. Acoust. Soc. Am. **131**(6), 4640–4650 (2012)
3. Hady, M.F.A., Schwenker, F.: Semi-supervised learning. In: Bianchini, M., Maggini, M., Jain, L. (eds.) Handbook on Neural Information Processing, pp. 215–239. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36657-4_7
4. Hady, M.F.A., Schwenker, F., Palm, G.: Semi-supervised learning for tree-structured ensembles of RBF networks with co-training. Neural Netw. **23**(4), 497–509 (2010)
5. Huang, S.J., Gao, N., Chen, S.: Multi-instance multi-label active learning. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, pp. 1886–1892. International Joint Conferences on Artificial Intelligence Organization, California, August 2017
6. Huang, S.J., Zhou, Z.H., Gao, W., Zhou, Z.H.: Fast multi-instance multi-label learning. In: AAAI (61321491), pp. 1868–1874, October 2014
7. Kanade, T., Cohn, J., Tian, Y.: Comprehensive database for facial expression analysis. In: Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580), pp. 46–53. IEEE Computer Society (2000)
8. Lucey, P., Cohn, J.F., Kanade, T., Saragih, J., Ambadar, Z., Matthews, I.: The extended Cohn-Kanade dataset (CK+): a complete dataset for action unit and emotion-specified expression. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pp. 94–101. IEEE, June 2010
9. Retz, R., Schwenker, F.: Active multi-instance multi-label learning. In: Wilhelm, A.F.X., Kestler, H.A. (eds.) Analysis of Large and Complex Data. SCDAKO, pp. 91–101. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-25226-1_8
10. Schwenker, F., Trentin, E.: Pattern classification and clustering: a review of partially supervised learning approaches. Pattern Recogn. Lett. **37**(1), 4–14 (2014)
11. Sebastiani, F.: Machine learning in automated text categorization. ACM Comput. Surv. **34**(1), 1–47 (2002)
12. Settles, B.: Active learning literature survey. Mach. Learn. **15**(2), 201–221 (1994)
13. Settles, B., Craven, M.: An analysis of active learning strategies for sequence labeling tasks. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 1070–1079. Association for Computational Linguistics (2008)
14. Shen, C., Jiao, J., Yang, Y., Wang, B.: Multi-instance multi-label learning for automatic tag recommendation. In: IEEE International Conference on Systems, Man, and Cybernetics, pp. 4910–4914. IEEE, October 2009
15. Surdeanu, M., Tibshirani, J., Nallapati, R., Manning, C.D.: Multi-instance multi-label learning for relation extraction. In: Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 455–465. Association for Computational Linguistics (2012)
16. The UniProt Consortium: UniProt: the universal protein knowledgebase. Nucleic Acids Res. **45**(D1), D158–D169 (2017)

17. Thiam, P., Meudt, S., Palm, G., Schwenker, F.: A temporal dependency based multi-modal active learning approach for audiovisual event detection. Neural Process. Lett. 1–24 (2017)
18. Wu, J.S., Huang, S.J., Zhou, Z.H.: Genome-wide protein function prediction through multi-instance multi-label learning. IEEE/ACM Trans. Comput. Biol. Bioinforma. **11**(5), 891–902 (2014)
19. Xu, X.S., Xue, X., Zhou, Z.H.: Ensemble multi-instance multi-label learning approach for video annotation task. In: Proceedings of the 19th ACM International Conference on Multimedia, MM 2011, p. 1153. ACM Press, New York (2011)
20. Li, Y.-X., Ji, S., Kumar, S., Ye, J., Zhou, Z.-H.: Drosophila gene expression pattern annotation through multi-instance multi-label learning. IEEE/ACM Trans. Comput. Biol. Bioinforma. **9**(1), 98–112 (2012)
21. Zhang, M.L.: A k-nearest neighbor based multi-instance multi-label learning algorithm. In: 2010 22nd IEEE International Conference on Tools with Artificial Intelligence, pp. 207–212. IEEE, October 2010
22. Zhou, Z.H., Zhang, M.l.: Multi-instance multi-label learning with application to scene classification. In: Advances in Neural Information Processing Systems, pp. 1609–1616 (2007)
23. Zhou, Z.H., Zhang, M.L., Huang, S.J., Li, Y.F.: Multi-instance multi-label learning. Artif. Intell. **176**(1), 2291–2320 (2012)