Knowledge-Based Systems 190 (2020) 105137

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

An efficient active learning method for multi-task learning

Yanshan Xiao [a], Zheng Chang [a], Bo Liu [b,*]

[a] Faculty of Computer, Guangdong University of Technology, China
[b] Faculty of Automation, Guangdong University of Technology, China

ARTICLE INFO

Article history:
Received 29 January 2019
Received in revised form 17 October 2019
Accepted 19 October 2019
Available online 23 October 2019

Keywords:
Multi-task classification
Active learning
Support vector machine

ABSTRACT

In multi-task learning, the sharing of information between related tasks affects and promotes the learning of each task. However, the traditional multi-task learning techniques always require sufficient labeled data to improve the learning of each task, and labeling samples is always expensive in practice. In this paper, we propose two variants of active learning methods for multi-task classification. In the uncertainty step, we propose the support vector preservation criterion that evaluates uncertainty at the level of classifier, which is called classifier-level uncertainty (CLU). In the diversity step, we propose two diversity criteria that evaluate diversity by the clustering method and the partition method respectively, which are called clustering-based diversity (CBD) and partition-based diversity (PBD) respectively. Each diversity criterion together with the uncertainty criterion is to form an active learning method for multi-task learning. In addition, the proposed support vector preservation criterion selects local informative samples which determine the hyperplane for each task. Furthermore, in order to maintain the distribution structure of the samples, we put forward the micro-kernel k-means clustering method and partition-based method to select global informative samples from the non-support vectors. By incorporating the local and global informative samples into active learning, we propose the two active learning methods for multi-task problems. We evaluate the effectiveness of the proposed methods by conducting experiments with other active learning methods. The experimental results show that the proposed two methods perform better than other active learning methods.

1. Introduction

Traditional learning focuses on a single task, in which a single task model is built for learning. Multi-task learning (MTL) is different from the traditional learning, by sharing the characterization between the related tasks, the model can be better generalized. Since the shared representation and multiple tasks are performed simultaneously, the number of the training samples and the overall model parameters are reduced, which makes the task execution more efficient. In recent years, support vector machines [1] (SVM) have been successfully used in multi-task classification. The SVM-based methods [2] are designed based on a general assumption that the classifier model parameters of different tasks are close to each other [3]. Sharing parameters and representations in SVM improve the performance of each classifier. Multi-task learning is widely used in various fields, such as drug interaction extraction [4], person identification [5–7] and object tracking [8,9].

Traditional supervised learning methods are dependent on adequate labeled samples to improve the generalization ability of classification models. However, in real-world applications, the number of labeled samples is always limited and it is time consuming and expensive to obtain large amounts of labeled samples. Active learning is one of the most effective methods to solve such problem. Different from traditional supervised learning methods, active learning [10,11] is an iterative process, and the goal of active learning is to obtain a well learning model with a small number of labeled samples. During each iteration, an initial classification model is created on a small number of labeled samples. Then, the most informative samples are selected from the unlabeled pool based on a selection strategy. Newly selected samples are labeled and added to the labeled set, which is used to create a new classification model. In this way, active learning can obtain high classification accuracy with few initially labeled samples, which reduces the cost of obtaining labeled data. In the past few years, active learning methods are exploited in many research areas. For supervised classification, CSAL [12] is proposed to define the reliable training sets for the classification of remote sensing images with support vector machines. For multi-class classification, MC_SVMA [13] is proposed to allow active learning to participate in the initial pattern classes mining and

[*] Corresponding author.
E-mail addresses: xiaoyanshan@189.cn (Y. Xiao), 854494092qq@sina.com (Z. Chang), csboliu@163.com (B. Liu).

https://doi.org/10.1016/j.knosys.2019.105137
0950-7051/© 2019 Elsevier B.V. All rights reserved.

the subsequent SVM training. Therefore, it is necessary to study the active learning method for multi-task classification problems.

As for the problem of multi-task classification, the conventional single-task active learning methods are not suitable for the problem of multi-task classification directly. The main challenge is how to select the informative data from the unlabeled samples based on the characteristic of the multi-task model. Therefore, we propose two novel variants of active learning methods for the SVM-based multi-task classification. Both active learning methods consist of two steps: the uncertainty step and the diversity step. In the first step, we put forward a support vectors preservation criterion to select the most informative samples around the classifier for each task. In the second step, we propose two variants of diverse samples selection methods to select the samples which can describe the structure of the data distribution. In all, the main contributions of the paper are as follows:

1. In the uncertainty step, we propose the support vectors preservation method to select the samples which have most effect on the classifier for each task. In addition, this is called classifier-level uncertainty criterion (CLU), which evaluates uncertainty at the level of SVM classifier. In this step, we first analyze the objective model of SVM-based multi-task learning, and then select the support vectors which can form the classifier for each task. To preserve the structure of the data distribution, another part of the most uncertain samples are selected in the diversity step.
2. In the diversity step, we propose two different diverse samples selection criteria: (1) clustering-based diversity (CBD) and (2) partition-based diversity (PBD). In CBD, the non-support vectors are divided into different clusters based on the micro-kernel k-means clustering method for each task. For each micro-cluster, we select only one representative sample to form the new training set, which can vaguely maintain the structure of the samples except for support vectors. In addition, to reduce time consumption of the diversity step, we further propose a partition-based diversity criterion (PBD). In PBD, we first divide the non-support vectors into different partitions in the feature space based on their distances to the hyperplane. We then select one representative sample from each non-empty partition to form a new training set, which can be used in the subsequent active learning. In addition, both diverse samples selection methods can be together with the uncertainty criterion to form a novel multi-task active learning method. In this way, we propose two variants of active learning methods, which are called CLU–CBD and CLU–PBD, for SVM-based multi-task learning.
3. In order to evaluate the effectiveness of the proposed methods, we perform extensive experiments on multiple datasets. The results show that the proposed methods can result in better accuracy with respect to the other active learning methods.

The rest of the paper is organized as follows. Section 2 discusses the related work. The proposed active learning methods for multi-task SVM are presented in Section 3. Section 4 describes the experimental setup and the experimental results. The conclusion and future work are presented in Section 5.

2. Related work

In this section, we briefly review previous work related to our research. In Section 2.1, we review the previous work on active learning. Then, we review the previous work on multi-task learning in Section 2.2.

2.1. Active learning

In the field of machine learning, high-quality samples can make the training process more efficient and less expensive. Active learning [14,15] is an effective learning method that enables high-quality samples to be selected during the training process for reducing sample redundancy and improving the performance of classification model with a small number of labeled samples. Active learning is an iterative process, in which the most informative samples are selected from the unlabeled pool, labeled and added to the training set during each iteration. Because the selection strategy directly affects learning efficiency and generalization performance, designing the selection strategies is the crucial step for active learning. According to the query strategies used, active learning can be divided into two categories: uncertainty sampling methods [16,17] and query by committee methods [18–20]. Next, we review them as follows.

For uncertainty sampling-based strategies, they focus on selecting the samples that have most influence on the current classifier, i.e. the sample which the classifier has low confidence in [21], or the samples which are easily mispredicted by the classifier [22]. Chen et al. [16] develop an active learning method based on uncertainty and complexity that guarantees diagnosis accuracy and improves fault pattern classification robustness. In the method, uncertainty is to describe the confusion degree of the samples, and complexity is to express the ambiguity of samples. Wang et al. [17] integrate uncertainty and diversity into one formula by multi-class settings. Uncertainty is measured by the margin minimum while diversity is measured by the maximum mean discrepancy. In another study, Tran et al. [23] propose a novel sampling strategy based on the similarity between the unlabeled instances and the labeled instances. Also, the self-learning method, in which highly reliable instance is selected based on the probability of the predicted label sequence, is adopted to further reduce the labeling effort.

For query by committee-based strategies, the Query by Committee algorithm (QBC) uses the examples, whose expected information gain is high, as queries examples. And it filters the informative examples from the unlabeled examples. In [18], QBC is utilized for uncertainty and demonstrates that its performance can be improved by introducing diversity and density in instance utility. Majidi and Crane [19] provide a novel framework in which a committee of parsers is used to generate separate models. These separate models predict the head nodes and the relation in the unlabeled set. Then, for each sentence with most entropy value, the tokens with the highest entropy are annotated by the expert.

2.2. Multi-task learning

The traditional single task learning methods only focus on the information of the learning task itself and pay little attention to the information of the related tasks. In fact, by learning related tasks simultaneously, the objective of getting better generalization accuracy can be achieved [24,25]. Therefore, multi-task learning [26] is studied to address problems of multiple tasks by sharing useful information, such as a common representation or some model parameters between related tasks. According to the learning algorithms used in multi-task learning, multi-task learning can be divided into three categories: SVM-based multi-task learning [27,28], neural networks-based multi-task learning [29,30], and Bayesian multi-task learning [31].

For SVM-based multi-task learning, support vector machine is used as the classification model and the multi-task problem is transformed into a quadratic optimization problem by parameters sharing and common feature representation [32]. Thus, performance of the classification model can be improved by jointly performing multiple learning tasks. Lu et al. [27] propose two multi-task learning methods, named as MTL-aLS-SVM I and MTL-aLS-SVM II respectively, for binary classification. MTL-aLS-SVM I seeks for a trade-off between the maximal expectile distance for each task model and the closeness of each task model to the averaged model. MTL-aLS-SVM II can use different kernel functions for different tasks, and it is an extension of the MTL-aLS-SVM I. In [33], Liang et al. first propose a novel multi-task ranking SVM model which incorporates multi-task learning and learning to rank into a unified framework. The proposed model is trained using the relative order information between the cosaliency score of pixel pairs.

For neural networks-based multi-task learning, the algorithms are usually designed to reduce the risk of over-fitting or achieve similarity of parameters by different parameters sharing mechanisms. Liu et al. [30] introduce a gating mechanism, which can better control the information passed by the neuron in the shared layers. In addition, they integrate RNN into the multi-learning framework for text classification to map arbitrary text into semantic vector representations. They also propose three models with different shared mechanisms, called uniform-layer architecture, coupled-layer architecture, and shared-layer architecture. Liao et al. [34] propose a novel multi-task deep learning (MTDL) method to solve the data insufficiency problem. MTDL is inspired by multi-task learning and deep learning. Specifically, MTDL can not only classify the small-scale datasets from different cancers simultaneously but also employ closely related datasets to help learning a better representation and boosting the classification performance.

For Bayesian multi-task learning, Bayesian approaches are adopted to recognize parallel tasks and learn their underlying structure from the data. In addition, some model parameters are shared and others are soft-shared through a prior distribution in some Bayesian-based multi-task learning settings [31]. In [35], Greenlaw et al. propose a framework for the analysis of data arising in the study of imaging genomics that extends a previously developed regularization approach in order to allow for the quantification of estimation (posterior) uncertainty in multi-task regression. In [36], Marquand et al. construct MTL models, in which each subject is modeled by a separate task, and use a flexible covariance structure to model the relationships between tasks and induce coupling between them using Gaussian process priors.

In addition, multi-task clustering has been studied to improve the performance of single clustering task by extracting knowledge among related tasks. Yi et al. [37] propose a multi-task multi-view algorithm based on Locally Linear Embedding (LLE) and Laplacian Eigenmaps (LE) methods. In the first transformation step, the samples of multiple views from each task in the original space are transformed into the common view space. In the second step, the samples are mapped from view space to the task space. Finally, the K-Means clustering algorithm is used to achieve multi-task multi-view clustering. Also, a novel graph-based multi-view clustering method that works in the GBS framework without additional clustering steps is proposed [38]. It produces the final clusters on the graph matrix of the data by imposing a rank constraint on the graph Laplacian matrix.

3. The proposed active learning methods for SVM-based multi-task learning

In Section 3.1, the multi-task support vector machine is first presented. In Section 3.2, the proposed active learning methods are presented.

3.1. Multi-task SVM classification

We denote the dataset in the kth task as $X_k = L_k \cup U_k$, where L_k and U_k represent the labeled dataset and the unlabeled dataset in the kth task, respectively. Besides, we have all these tasks on the same space χ, with $\chi \subseteq \mathbb{R}^d$. The goal is to learn n decision functions $f_1(x), f_2(x), \ldots, f_n(x)$, one for each task.

The ith sample in the kth task is denoted as x_{ik}, the decision function is $f_k(x_{ik}) = w_k \cdot \phi(x_{ik}) + b_k$, in which $\phi(x)$ is a non-linear feature mapping and b_k is the offset vector. If $f_k(x_{ik}) \geq 0, x_{ik}$ is labeled as positive; otherwise, it is labeled as negative. w_k is the normal vector to the decision hyperplane and consists of two parts. The first part is the common mean vector shared by each task, which is denoted as w_0, and the second part is the specific vector v_k for a specific task. Here, w_k for each task is expressed as:

$$w_k = w_0 + v_k, \tag{1}$$

Following the above assumption, SVM can be generalized to multi-task learning. The primal optimization problem can be written as follows:

$$\begin{aligned} &\min_{w_0, b_k, v_k, \xi_{ik}} \frac{1}{2}\|w_0\|^2 + \sum_{k=1}^{n} \lambda_k \|v_k\|^2 + \mathcal{C} \sum_{k=1}^{n} \sum_{i=1}^{n_k} \xi_{ik} \\ &s.t. \quad y_{ik}(w_k^{\mathrm{T}} \cdot \phi(x_{ik}) + b_k) \geq 1 - \xi_{ik}, \xi_{ik} \geq 0 \end{aligned} \tag{2}$$

where n_k is the number of data in kth task, $\mathcal{C}$ is penalty parameter that balances the margin and errors. In addition, parameter λ_k is used to control the preference of the tasks. In other words, the larger the value of the parameter λ_k, the higher the preference of the kth task. ξ_{ik} is the corresponding slack variable.

By introducing Lagrangian multipliers α_{ik} for the samples in each task, the solution of Problem (2) is to resolve the dual problem:

$$\begin{aligned} &\max_{\alpha_{ik}} \sum_{k=1}^{n} \sum_{i=1}^{n_k} \alpha_{ik} - \frac{1}{2} \sum_{k=1}^{n} \sum_{t=1}^{n} \sum_{i=1}^{n_k} \sum_{j=1}^{n_t} \alpha_{ik} \alpha_{jt} \langle \phi(x_{ik}), \phi(x_{jt}) \rangle \\ &\quad s.t. \quad 0 \leq \alpha_{ik} \leq \mathcal{C} \end{aligned} \tag{3}$$

Suppose that we define a kernel function as $k(x_{ik}, x_{jt}) = \langle \phi(x_{ik}), \phi(x_{jt}) \rangle$, where k and t are the task index associated to each sample. Thus, the classifier of each task is obtained by solving the above optimization problem using the kernel function $k(x_{ik}, x_{jt})$, and then the decision function for each task is given by:

$$f_k(x) = \sum_{k=1}^{n} \sum_{i=1}^{n_k} \alpha_{ik} k(x_{ik}, x) + b_k \tag{4}$$

3.2. The proposed methods

In this section, the proposed active learning strategies are presented by dividing the strategy into two consecutive steps: the uncertainty step and the diversity step. In the uncertainty step, the uncertain samples around the hyperplane are extracted. In the diversity step, the uncertain samples outside the margins are selected.

3.2.1. Uncertainty step

The uncertainty criterion identifies samples with the lowest classification confidence as the most uncertain samples, since the samples, which the classifier is most uncertain with, always have useful information for the classifier. We propose a novel uncertainty criterion, called support vectors preservation criterion, which selects the samples around the classifier for each task in the SVM-based multi-task learning. Since the proposed uncertainty criterion evaluates uncertainty at the level of

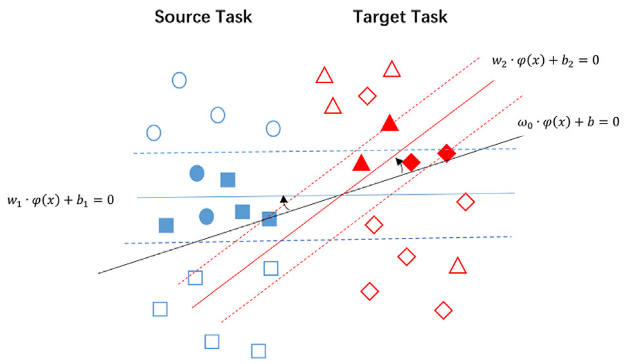

Source Task
Target Task
$w_2 \cdot \varphi(x) + b_2 = 0$
$\omega_0 \cdot \varphi(x) + b = 0$
$w_1 \cdot \varphi(x) + b_1 = 0$

Fig. 1. Extraction of the uncertain samples. Color-filled samples are support vectors which determine the classifiers. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

SVM classifier, it can be abbreviated as CLU. The details are as follows.

Take the case of two-task, that is transfer learning, as an example, we present the uncertainty criterion for two-task learning. Fig. 1 shows an intuitive example in which the uncertain samples are selected based on the uncertain criterion. For the source task in the left of Fig. 1, circles and squares represent positive and negative samples, respectively. Shapes filled with blue represent the selected samples for the source task. The solid blue line represents the classification hyperplane and the equation $w_1 \cdot \phi(x) + b_1 = 0$ represents the classifier for the source task. The dotted blue lines represent the margin boundaries, $w_1 \cdot \phi(x) + b_1 = 1$ and $w_1 \cdot \phi(x) + b_1 = -1$. For the target task in the right of Fig. 1, triangles and diamonds represent positive and negative samples, respectively. Shapes filled with red represent the selected samples for the target task. The solid red line represents the classification hyperplane and the equation $w_2 \cdot \phi(x) + b_2 = 0$ represents the classifier for the target task. The dotted red lines represent the margin boundaries, $w_2 \cdot \phi(x) + b_2 = 1$ and $w_2 \cdot \phi(x) + b_2 = -1$. It can be seen that the solid red and blue samples support the transfer learning classifiers.

For SVM-based multi-task learning, we have the analysis as follows. According to the decision function (4), the samples, which have non-zero α_{ik}, determine the classification hyperplane. Such samples are known as support vectors (SVs). Other samples whose Lagrangian coefficient α_{ik} is zero, do not affect the classification hyperplane. The Karush–Kuhn–Tucker Conditions of transfer learning can be written as:

$$\alpha_{ik} = 0 \Leftrightarrow y_{ik}(w_k \cdot \phi(x_{ik}) + b_k) \geq 1 \tag{5}$$

$$0 < \alpha_{ik} < C \Leftrightarrow y_{ik}(w_k \cdot \phi(x_{ik}) + b_k) = 1 \tag{6}$$

$$\alpha_{ik} = C \Leftrightarrow y_{ik}(w_k \cdot \phi(x_{ik}) + b_k) \leq 1 \tag{7}$$

Furthermore, for the formula (5), the equation $y_{ik}(w_k \cdot \phi(x_{ik}) + b_k) \geq 1$ means the sample x_{ik} lies outside the margin boundaries. The α_{ik} of samples that lie outside the margin boundaries are zero. For the formula (6)(7), the equation $y_{ik}(w_k \cdot \phi(x_{ik}) + b_k) = 1$ means the sample x_{ik} resides on the margin boundaries, the equation $y_{ik}(w_k \cdot \phi(x_{ik}) + b_k) \leq 1$ means the sample x_{ik} resides inside the margin boundaries. Thus, the samples that lie on or inside the margin boundaries have non-zero α_{ik}. Based on the above analysis, we draw a conclusion that the support vectors (SVs), which fall within the margin boundaries of the source task, determine the classifier of the source task and are easily misclassified by the classifier. The analysis is similar for the target task, the support vectors (SVs), which fall within the margin boundaries of the target task, determine the classifier of the target task and are easily misclassified by the classifier. Adding such samples into the training set to retrain the classifiers will improve the performance of both classifiers.

For all $x_{ik} \in U_k$, the uncertain samples, which are also called support vectors (SVs) here, are selected as

$$SV_k = \{x_{ik} | f_k(x_{ik}) \in [-1, +1]\} \tag{8}$$

where SV_k is the set of samples selected from the unlabeled set of kth task in the uncertainty step. Then, the labels are assigned to the selected samples, the unlabeled set is updated and the selected samples are added to the training set.

3.2.2. Diversity step

In order to vaguely preserve the sample distribution and avoid degradation of the generalization performance in classifying test unlabeled samples, it is necessary to select a part of samples outside the margin boundaries to maintain the basic structure of the data. Next, we propose two different diversity criteria to select the samples outside the margin boundaries in the following diversity step.

A. clustering-based diversity

Clustering techniques evaluate the distribution of the samples in a feature space and group the similar samples into the same clusters. Since the samples within the same cluster are correlated and provide similar information, selecting a representative sample from each cluster prevents sample redundancy. In addition, a cluster-based active learning approach is previously proposed in [39], which is based on both center-based selection and border-based selection. The former selection criterion is to select the samples which are close to the cluster centers, on the grounds that these samples usually characterize the intrinsic location and the core informative parts of the classes. The latter is to select samples which are close to the border of the clusters, on the grounds that these samples characterize the border-line between the classes and the expected location of the decision boundary. However, the proposed clustering-based criterion in this paper is different from the methods mentioned above. We focus on the diversity and the uncertainty of the samples to be selected. On the one hand, only one sample in each micro-cluster is selected to ensure that the information contained in the selected samples are highly variable. On the other hand, considering the characteristics of the support vector machine, that is, the samples close to the decision boundary usually characterize the expected location of the decision boundary, the samples closest to the decision boundary should be selected. Here we conduct the micro-kernel k-means clustering algorithm to divide the unlabeled samples outside the margin boundaries into different clusters. Specific steps are as follows:

- For each task, the unlabeled samples U_k are divided into two parts: a positive sample set U_k^+ and a negative sample set U_k^- $(k = 1, 2, \ldots, n)$, which are denoted as

$$U_k^+ = \{x_{ik} \mid f_k(x_{ik}) > 1, x_{ik} \in U_k\} \tag{9}$$

$$U_k^- = \{x_{ik} \mid f_k(x_{ik}) < -1, x_{ik} \in U_k\} \tag{10}$$

- Apply the micro-kernel k-means clustering algorithm to U_k^+ and U_k^- with $K = h$, respectively. Then, U_k^+ is divided into $h = \frac{|U_k|}{a}$ different clusters $C_1^+, C_2^+, \ldots, C_h^+$. Similarly, U_k^- is divided into h different clusters $C_1^-, C_2^-, \ldots, C_h^-$.
- After $2h$ clusters of each task are obtained, we choose the most uncertain sample, which has minimum $|f(x)|$ value, as the representative sample from each cluster. Then, the $2h$ uncertain samples are selected as

$$x_{tk}^+ = \arg\min_{x_{ik} \in C_t^+} |f(x_{ik})| \tag{11}$$

$$x_{tk}^{-} = \arg\min_{x_{ik}\epsilon C_t^{-}} |f(x_{ik})| \tag{12}$$

where $t = 1, 2, \ldots, h$, x_{tk}^{+} is the uncertain sample selected from cluster C_t^{+} of kth task, x_{tk}^{-} is the uncertain sample selected from cluster C_t^{-} of kth task.

- After obtaining $2h$ selected samples from the unlabeled samples outside the margin boundaries in each task, assign the labels to the selected samples and add them to the training set. At last, the classifiers get retrained with the updated training set.

B. partition-based diversity

In order to select unlabeled samples diverse and preserve the basic data structure of the data, we propose a partition-based diversity method that divides the unlabeled samples outside the margin boundaries into different partitions in the feature space. Firstly, the unlabeled set updated in the uncertainty step for each task, such as kth task, is divided into two subsets U_k^{+} and U_k^{-} using (9) and (10). The maximum decision value $f_k^{max}(\cdot)$ of samples in U_k^{+} and the minimum decision value $f_k^{min}(\cdot)$ of samples in U_k^{-} are obtained. Then, each of the two subsets is divided into $m = \frac{|U_k|}{a}$ partitions, the width of each partition in the subsets U_k^{+} and U_k^{-} is computed as

$$W_k^{+} = \frac{f_k^{max} - 1}{m} \tag{13}$$

$$W_k^{-} = \frac{-1 - f_k^{min}}{m} \tag{14}$$

For W_k^{+}, $f_k^{max} - 1$ means the distance between the hyperplane $w_k \cdot \phi(x) + b_k = 1$ and the sample with the largest decision value. For W_k^{-}, $-1 - f_k^{min}$ means the distance between the hyperplane $w_k \cdot \phi(x) + b_k = -1$ and the sample with the smallest decision value. Thus, the lower bounds and upper bounds of each partition can be denoted below:

$$L_k^{+}(i) = 1 + (i-1)W_k^{+} \quad and \quad H_k^{+}(i) = 1 + (i)W_k^{+} \tag{15}$$

$$L_k^{-}(i) = -1 + (i-1)W_k^{-} \quad and \quad H_k^{-}(i) = -1 + (i)W_k^{-} \tag{16}$$

where $i = 1, 2, \ldots, m$, $L_k^{+}(i)$ and $H_k^{+}(i)$ represent the lower and upper bounds of *ith* partition of subset U_k^{+}, $L_k^{-}(i)$ and $H_k^{-}(i)$ represent the lower and upper bounds of *ith* partition of subset U_k^{-}. Let P_k be the set of non-empty partitions of two subsets U_k^{+} and U_k^{-}, $P_k(j)$ is *jth* partition in P_k. Then, we select one sample as representative sample from each non-empty partition as follows:

$$x_{jk} = \arg\min_{x_{ik}\in P_k(j)} |f_k(x_{ik})|, j = 1, 2, \ldots, |P_k| \tag{17}$$

After obtaining a batch of uncertain samples from the unlabeled samples outside the margin boundaries in each task, they are labeled and added to the training set. Then, the classification model is retrained with the updated training set.

Each diversity technique can be together with the uncertainty technique to form an active learning method for SVM-based multi-task learning. The two active learning methods are called as: CLU with CBD (denoted by CLU–CBD) and CLU with PBD (denoted by CLU–PBD). The details of CLU–CBD and CLU–PBD algorithms for multi-task classification are presented in Algorithms 1 and 2 respectively.

In each iteration of the algorithm CLU–CBD, we first create the initial classifiers with the labeled samples, then we select the uncertain samples inside the margin boundaries using CLU and select the uncertain samples outside the margin boundaries using CBD. The selected uncertain samples are labeled and added into the labeled set. The classifiers are retrained with the new labeled set. The process of the algorithm CLU–PBD is similar except for

Algorithm 1 CLU–CBD

```
Input: Labeled set $L_k$, unlabeled set $U_k$ for multi-task learning, parameter $a$, the number of tasks $n$.
Output: Classifiers for each task $f_0, f_1, \ldots, f_n$.
Train the classifiers $f_0, f_1, \ldots, f_n$ for each task on the initial labeled set based on the optimization (2).
repeat
  for $k = 1$ to $n$ do
    Select the set of support vectors using (8) and assign labels to them.
    Add the selected samples into the training set and update the unlabeled set.
    Divide the unlabeled set into two sets $U_k^{+}$ and $U_k^{-}$ by (9) and (10).
    Apply micro-kernel k-means clustering algorithm to $U_k^{+}$ and $U_k^{-}$ to divide them into $h = \frac{|U_k|}{a}$ different clusters, respectively.
    Choose the representative samples from each non-empty partition in $P_k$ by (11)(12) and assign labels to them.
    Add the selected samples into the training set and update the unlabeled set.
    Retrain the classifiers with the updated training set using optimization (2).
  end for
until stop criterion is satisfied
```

Algorithm 2 CLU–PBD

```
Input: Labeled set $L_k$, unlabeled set $U_k$ for multi-task learning, parameter $a$, the number of tasks $n$.
Output: Classifiers for each task $f_0, f_1, \ldots, f_n$.
Train the classifiers $f_0, f_1, \ldots, f_n$ for each task on the initial labeled set using the optimization (2).
repeat
  for $k = 1$ to $n$ do
    Select the set of support vectors using (8) and assign labels to them.
    Add the selected samples into the training set and update the unlabeled set.
    Divide the unlabeled set into two sets $U_k^{+}$ and $U_k^{-}$ by (9) and (10).
    Compute the maximum decision value $f_k^{max}(\cdot)$ of samples in $U_k^{+}$ and the minimum decision value $f_k^{min}(\cdot)$ of samples in $U_k^{-}$.
    Compute the width ($W_k^{+}$) of each partition of samples in the subsets $U_k^{+}$ and the width ($W_k^{-}$) of each partition of samples in the subsets $U_k^{-}$ by (13).
    Compute the bounds ($H_k$ and $L_k$) of each partition in $U_k^{+}$ and $U_k^{-}$ by (15).
    Use $H_k, L_k, W_k$ to partition $U_k^{+}$ and $U_k^{-}$ into $m = \frac{|U_k|}{a}$ partitions, respectively.
    Compute the set $P_k$ of non-empty partitions of two subsets $U_k^{+}$ and $U_k^{-}$.
    Choose the representative samples from each cluster by (17) and assign labels to them.
    Add the selected samples into the training set and update the unlabeled set.
    Retrain the classifiers with the updated training set using (2).
  end for
until stop criterion is satisfied
```

Table 1
Description of the datasets.

	Dataset	Task number	Positive sub-dataset	Negative sub-dataset
1	comp vs. rec	4	comp.graphics	rec.autos
			comp.os.ms-windows.misc	rec.motorcycles
			comp.sys.ibm.pc.hardware	rec.sport.baseball
			comp.sys.mac.hardware	rec.sport.hockey
2	rec vs. sci	4	rec.autos	sci.crypt
			rec.motorcycles	sci.electronics
			rec.sport.baseball	sci.med
			rec.sport.hockey	sci.space
3	comp vs. sci	4	comp.os.ms-windows.misc	sci.med
			comp.sys.ibm.pc.hardware	sci.crypt
			comp.graphics	sci.space
			comp.sys.mac.hardware	sci.electronics
4	sci vs. talk	4	sci.crypt	talk.politics.guns
			sci.electronics	talk.politics.mideast
			sci.med	talk.politics.misc
			sci.space	talk.religion.misc
5	talk vs. comp	4	talk.politics.mideast	comp.sys.ibm.pc.hardware
			talk.religion.misc	comp.graphics
			talk.politics.guns	comp.sys.mac.hardware
			talk.politics.misc	comp.os.ms-windows.misc
6	talk vs. rec	4	talk.religion.misc	rec.sport.hockey
			talk.politics.misc	rec.sport.baseball
			talk.politics.guns	rec.autos
			talk.politics.mideast	rec.motorcycles
7	orgs vs. people	2	part of samples in orgs	part of samples in people
			another part of samples in orgs	another part of samples in people
8	people vs. places	2	part of samples in people	part of samples in places
			another part of samples in people	another part of samples in places
9	orgs vs. places	2	part of samples in places	part of samples in orgs
			another part of samples in places	another part of samples in orgs
10	Dermatology	5	psoriasis	lichen planus
			lichen planus	seboreic dermatitis
			cronic dermatitis	pityriasis rosea
			seboreic dermatitis	pityriasis rubra pilaris
			pityriasis rosea	cronic dermatitis

the step of selecting the informative samples outside the margin boundaries. As for the stop criterion, taking SVMs as the base model, we follow the model-specific stopping criterion [40] that the active learning process should stop when there are no unlabeled samples lying within the margin boundaries of classifiers, $w_k \cdot \phi(x) + b_k = 1$ and $w_k \cdot \phi(x) + b_k = -1$, since the samples (support vectors) within the boundaries support the classifier and the samples outside the boundaries will not alter the hyperplane when the obtained classifier is stable after iterations.

4. Experiment

4.1. Baselines and metrics

In this section, we investigate the effectiveness of the proposed approaches (CLU–CBD and CLU–PBD) empirically. For comparison, four other active learning methods (QDD, MS-RP, VOI and Random) are used as baselines. The details of the baselines are as follows.

- **QDD:** The batch active learning with query-by-committee, diversity and density [18] (QDD) method incorporates uncertainty, diversity and density to measure the utility of samples. Then, samples of the highest utility $u(x)$ are queried as the following form

  $$u(x) = (1 - \lambda - \beta)f(x) + \lambda d(x) + \beta h(x) \tag{18}$$

  where $f(x)$, $d(x)$ and $h(x)$ are the uncertainty, diversity and density functions, respectively, and $0 \leq \lambda, \beta \leq 1$ control the relative importance. $f(x)$, $d(x)$ and $h(x)$ are scaled to between zero and one by subtracting the minimum and dividing the range to prevent misleading results that may arise from their different ranges. Referring to the settings in QDD, vote entropy [41], RF dissimilarity and a $k - nearest$ neighbors (kNN)-based density measure [42] are utilized as the uncertainty function, the distance measure to derive the diversity function and the density function, respectively.
- **MS-RP:** The region-partitioning margin sampling algorithm [43] (MS-RP) first selects informative samples based on the adopted uncertainty criterion. Then a constraint that the region label of the newly selected sample must be different from those of already selected samples at each iteration is imposed to the uncertainty criterion. And candidate samples are extracted from the informative samples based on the uncertainty criterion with constraints.
- **VOI:** The value of information algorithm [44] utilizes a cross-task value of information criteria, in which the reward of a labeling assignment is propagated and measured over all relevant tasks.

  $$VOI(Y, x) = \sum_{y} p(Y = y|x)R(p, Y = y, x), \tag{19}$$

  where R is the rewards function and we use $R(p, Y = y, x) = -\log_2 p(Y = y|x)$. This strategy is to select the sample which is the most uncertain over all tasks;
- **Random Method:**This method always randomly selects samples from the data, we then utilize random strategy to select samples from each task.

To exhibit the effectiveness of the proposed methods, three metrics have been considered: average overall classification accuracy($\overline{OA}$), standard deviation(s), and average kappa accuracy ($\overline{k}$). Overall classification accuracy, which is a typical metric used in active learning, is defined as the ratio of the number of samples correctly classified to the total number of samples in the test data. Standard deviation is to measure the stability of the method

Table 2
The results of average overall classification accuracy ($\overline{OA}$), standard deviation (s) and average kappa accuracy ($\bar{k}$) produced by the investigated methods for the datasets in Table 1 (best results are reported in bold-face).

Dataset	CLU–CBD			CLU–PBD			QDD			MS-RP			VOI			Random		
	$\overline{OA}$	s	$\bar{k}$	$\overline{OA}$	s	$\bar{k}$	$\overline{OA}$	s	$\bar{k}$	$\overline{OA}$	s	$\bar{k}$	$\overline{OA}$	s	$\bar{k}$	$\overline{OA}$	s	$\bar{k}$
comp vs. rec	**85.45**	**0.08**	**0.91**	85.12	0.24	0.83	84.25	0.28	0.77	84.03	0.35	0.73	83.81	0.55	0.88	82.15	2.57	0.75
rec vs. sci	**83.92**	**1.04**	**0.76**	83.54	1.32	0.73	82.07	1.42	0.73	81.92	1.57	0.72	82.85	1.95	0.74	80.27	2.35	0.69
comp vs. sci	**87.68**	2.13	**0.87**	87.15	2.62	0.85	85.26	**2.10**	0.80	85.97	2.43	0.84	83.75	2.72	0.79	82.38	3.08	0.67
sci vs. talk	85.74	**0.97**	**0.72**	**85.83**	1.22	0.70	83.94	1.55	0.68	84.23	1.64	0.70	81.78	2.35	0.62	79.37	3.26	0.69
talk vs. comp	**84.89**	1.89	**0.73**	84.35	**1.58**	0.65	83.12	1.86	0.62	83.05	1.78	0.65	82.53	1.83	0.67	76.42	2.82	0.61
talk vs. rec	**87.55**	**2.05**	**0.89**	86.78	2.54	0.85	85.67	2.56	0.87	85.45	2.33	0.82	86.25	2.76	0.78	83.24	2.96	0.72
orgs vs. people	85.19	**1.22**	**0.85**	**85.77**	2.53	0.75	84.18	1.96	0.77	84.09	1.93	0.75	83.62	2.37	0.64	81.26	2.47	0.72
orgs vs. places	**84.72**	**0.27**	**0.82**	84.31	1.26	0.75	82.56	1.19	0.75	83.54	1.73	0.72	83.26	2.38	0.73	81.43	3.23	0.68
people vs. places	**86.52**	**0.82**	**0.92**	85.26	1.35	0.84	84.32	1.46	0.79	83.32	1.77	0.76	81.23	2.19	0.65	79.24	2.18	0.65
dermatology	**72.38**	1.77	0.62	71.86	2.82	0.56	71.22	1.84	**0.65**	70.89	**1.75**	0.59	70.34	2.67	0.50	68.35	3.14	0.45

Table 3
The average performance (mean performance of 24 corresponding problems for each pair of top categories) of average overall classification accuracy ($\overline{OA}$), standard deviation (s) and average kappa accuracy ($\bar{k}$) produced by the investigated methods for each pair of top categories (best results are reported in bold-face).

Dataset	CLU–CBD			CLU–PBD			QDD			MS-RP			VOI			Random		
	$\overline{OA}$	s	$\bar{k}$	$\overline{OA}$	s	$\bar{k}$	$\overline{OA}$	s	$\bar{k}$	$\overline{OA}$	s	$\bar{k}$	$\overline{OA}$	s	$\bar{k}$	$\overline{OA}$	s	$\bar{k}$
comp vs. rec	**85.53**	**0.10**	**0.92**	85.35	0.27	0.79	84.37	0.27	0.74	83.84	0.31	0.75	83.86	0.58	0.83	82.15	2.46	0.73
rec vs. sci	**83.84**	**1.09**	**0.78**	83.62	1.35	0.74	82.25	1.39	0.73	82.08	1.54	0.74	82.75	1.96	0.73	80.36	2.25	0.70
comp vs. sci	**87.72**	2.10	**0.88**	87.32	2.57	0.86	85.35	**2.08**	0.82	85.92	2.42	0.78	83.72	2.68	0.78	82.43	2.95	0.71
sci vs. talk	85.75	**0.97**	**0.74**	**85.81**	1.20	0.73	83.77	1.51	0.68	84.32	1.65	0.72	81.73	2.39	0.61	79.54	3.22	0.68
talk vs. comp	**84.82**	1.85	**0.72**	84.41	**1.62**	0.68	83.15	1.90	0.65	83.04	1.74	0.69	82.62	1.84	0.67	76.63	2.88	0.60
talk vs. rec	**87.60**	**2.03**	**0.88**	86.75	2.54	0.83	85.73	2.55	0.89	85.42	2.35	0.81	86.37	2.74	0.81	83.09	2.99	0.69

with respect to the initial available labeled sample. The kappa accuracy [45] is another measure to compute the classification accuracy in the test data. Let $M = m_{ij}$, $1 \leq i \leq c$, $1 \leq j \leq c$ be the generated confusion matrix for a task containing n samples with c classes in the test set, where m_{ij} indicates the number of samples of class i that are labeled as class j. Then kappa accuracy is computed as follows:

$$k = \frac{n \times \sum_{i=1}^{c} m_{ii} - \sum_{i=1}^{c}(\sum_{j=1}^{c} m_{ij} \times \sum_{j=1}^{c} m_{ji})}{n^2 - \sum_{i=1}^{c}(\sum_{j=1}^{c} m_{ij} \times \sum_{j=1}^{c} m_{ji})} \tag{20}$$

4.2. Datasets and settings

We evaluate the proposed multi-task active learning approaches by conducting experiments on 20 Newsgroups[1] dataset which contains about 20,000 documents taken from 20 newsgroups, Reuters-21578[2] which contains about 21578 documents from the Reuters newswire, and the Dermatology data[3] which is from the UCI datasets. Since these datasets are not originally designed for multi-task learning, referring to the operation in [46–48], we reorganize the datasets as follows.

20 newsgroups. There are seven top categories in the 20 Newsgroups dataset: “alt”, “comp”, “misc”, “rec”, “sci”, “soc” and “talk”. These top categories are further divided into 20 subcategories where each categories has 1000 samples. We define the tasks as top-category-classification problems. Referring to the operation in [48], we remove three categories “alt”, “soc” and “misc”, since they are too small. The remaining four top categories(“comp”, “rec”, “sci”, “talk”) are used to construct six multi-task learning sub-datasets by combining two of them as illustrated in Table 1. For example, to construct the multi-task dataset comp vs sci, we select one sub-category from “comp” as positive sub-dataset and select one sub-category from “sci” as negative sub-dataset for each task. In addition, each document is represented as a binary vector consisting of the 200 most discriminating words determined by Weka’s info-gain filter [49]. Using this split strategy ensures that tasks are relevant because they are under the same top categories. At the same time, the tasks are ensured to be different because they are drawn from different sub-categories. We can randomly select subcategories to construct 4-task classification problems from any two top categories. In fact, there are totally 24 randomly constructed 4-task problems for each pair of top categories. We will first present the results on the six orderly constructed 4-task problems in Table 1, then present the average results over all randomly constructed problems.

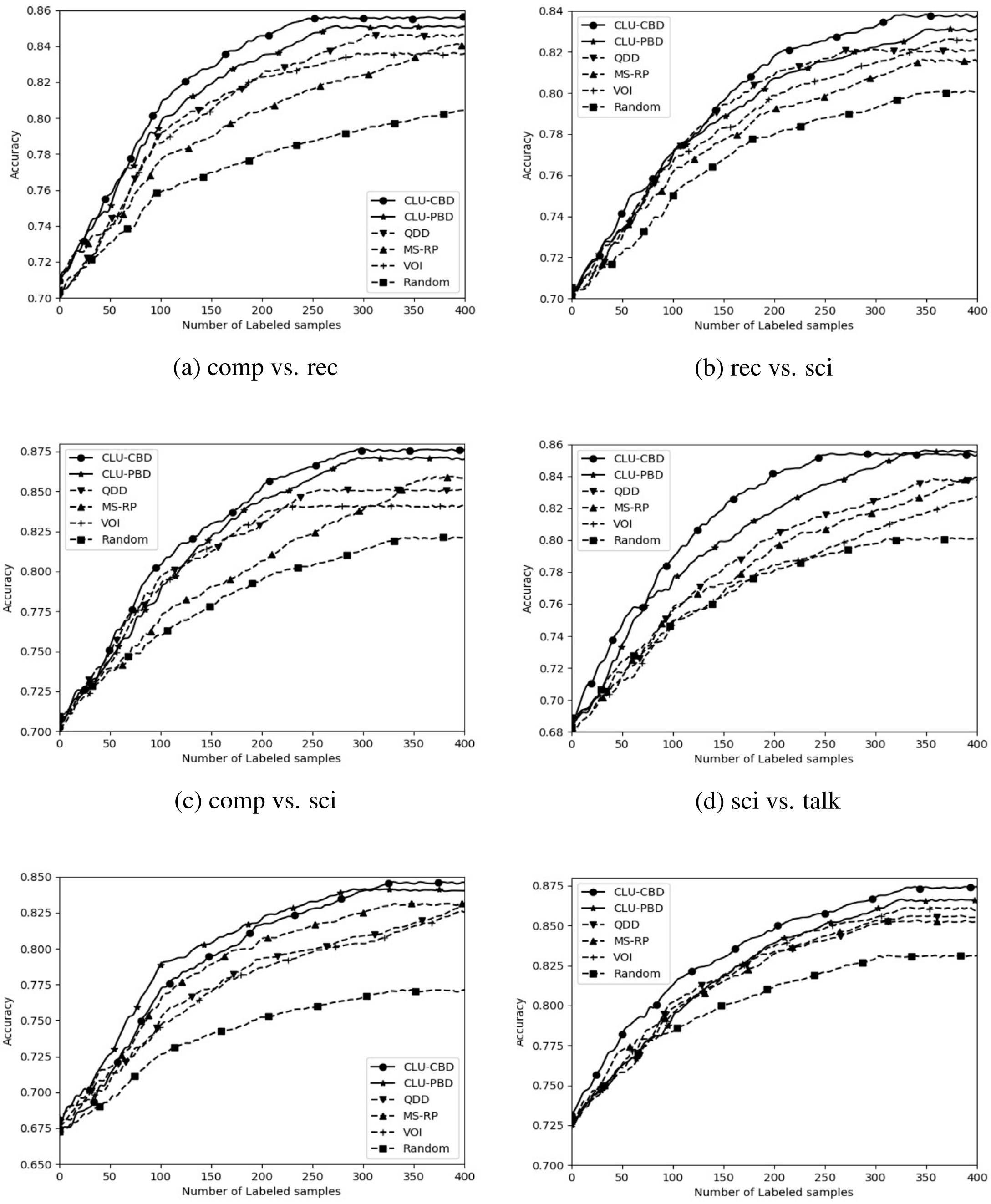
Reuters-21578. There are five top categories, among which “orgs”, “people” and “places” are three big ones, and these three top categories are used in the experiments. Similar to the setting in the 20 Newsgroups, three top categories are used to construct three datasets orgs vs people, orgs vs places and people vs places. Further, three datasets are reorganized into three multi-task sub-datasets with two tasks by dividing the sub-categories in each top categories. For example, the people vs places dataset consists of two tasks. We randomly divide “people” which has 267 sub-categories into two parts. Similarly, “places” is also divided randomly into two parts. Documents in “people” are considered as positive and documents in “places” are considered as negative. Therefore, the two tasks are related since the positive classes and the negative classes belong to the same top categories, respectively. As done in the 20 Newsgroups, each document is represented as a binary vector consisting of the 200 most discriminating words.

Dermatology data. The Dermatology data which is used for the differential diagnosis of erythematous-squamous diseases, consists of 6 diseases with very little differences: psoriasis, seboreic dermatitis, lichen planus, pityriasis rosea, chronic dermatitis, and pityriasis rubra pilaris. The 6 diseases are considered as six categories, which contain 112, 61, 72, 49, 52, 20 samples, respectively. This dataset contains 34 attributes, 33 of which are linear valued and one of them is nominal. To divide the dataset to a multi-task dataset, five classification tasks get constructed. The purpose of

[1] http://qwone.com/~jason/20Newsgroups/.

[2] http://kdd.ics.uci.edu/databases/reuters21578/.

[3] http://archive.ics.uci.edu/ml/datasets/Dermatology.



(a) comp vs. rec

(b) rec vs. sci

(c) comp vs. sci

(d) sci vs. talk

(e) talk vs. comp

(f) talk vs. rec

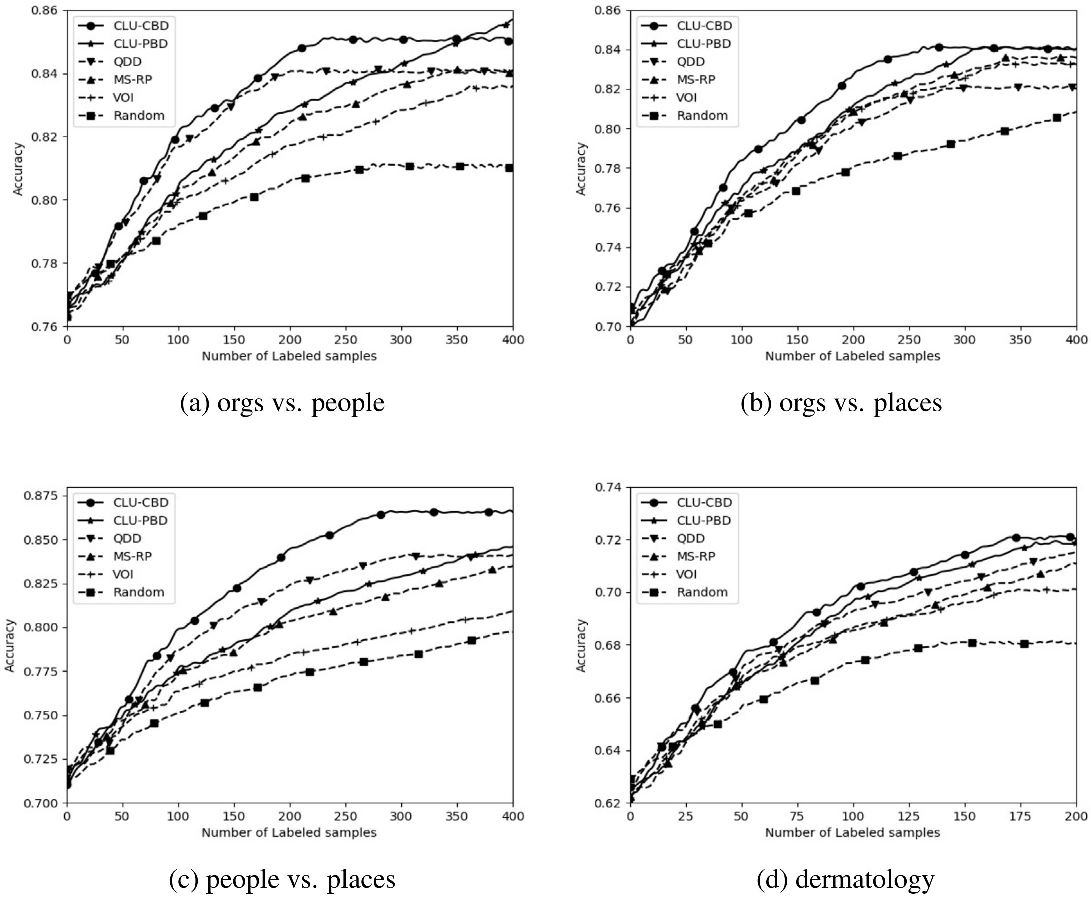
Fig. 2. Comparative performance on the six constructed 4-task problems from 20 Newsgroups.

each task is to diagnose one of six dermatological disease. For example, in the first classification task, the positive samples are taken from psoriasis and the negative samples are taken from lichen planus.

Each sub-dataset is randomly divided into two parts: the unlabeled training set(65%) and the labeled testing set(35%). 5% samples are randomly selected from the training set as an initial labeled set. The performance of each classifier is measured based on classification accuracy on the test set at the end of each iteration. For the parameter settings of the proposed methods, the value of clustering parameter h is set as $h = \frac{|U_k|}{a}$, where $|U_k|$ is the number of unlabeled samples in the training set for task k. The value of the partition parameter m is set as $m = \frac{|U_k|}{a}$. To void the bias, the active learning process in the experiment is repeated for 10 runs with different randomly generated initial training sets.

4.3. Performance comparison

Table 2 presents the results of average overall classification accuracy ($\overline{OA}$), standard deviation (s) and the average kappa accuracy ($\overline{k}$) produced by the investigated methods for the datasets in Table 1. It can be observed that the proposed CLU–CBD and CLU–PBD methods outperform other methods on all metrics for most datasets, this is because the proposed methods extract the informative samples around the classifier and the representative examples, which can maintain the basic structure of the data, while the other methods do not explicitly extract these structure



(a) orgs vs. people

(b) orgs vs. places

(c) people vs. places

(d) dermatology

Fig. 3. Comparative performance on Reuters-21578 data and Dermatology data.

data. As for the standard deviation, the proposed CLU–CBD and PBD methods obtain less variation in all since their standard deviations are comparatively lower with respect to other methods. At the same time, the kappa coefficient of the proposed CLU–CBD and CLU–PBD methods are higher than other methods, which also verifies the performance of the proposed methods in terms of accuracy. As for the dermatology dataset, the stability of accuracy for MS-RP method and the kappa accuracy of QDD are comparable, but their overall classification accuracies are relatively lower compared with the proposed CLU–CBD and CLU–PBD methods. In addition, QDD method chooses more representative samples from dense regions that can better exploit the feature space of unlabeled data. MS-RP selects queries based on the XOR-based partition to ensure the diversity of samples. VOI method only extracts samples based on their total information value for all tasks. However, a sample with maximum total information value for all tasks does not mean that it is valuable for each task. Random method always selects samples randomly from the data of each task, the selected samples are not always informative. So, the two proposed methods, QDD and MS-RP methods perform better than VOI and Random methods for most datasets. We also find that the proposed CLU–CBD method performs better than CLU–PBD method for most datasets, this occurs since the micro-clustering method can always cluster the non-support vector data into a number of condense clusters, and the clusters can cover most of the data; however, the partition-based method splits the non-support vector data according to the distance to the hyperplane, and selects samples from the non-empty partitions. Thus, the proposed CLU–CBD can select more representative samples than CLU–PBD method and perform better than the latter one. However, the CLU–PBD method has advantages in time consumption, which we will discuss in Section 4.5.

Except for the experiments on the combinations of each pair of top categories from 20 Newsgroups listed in Table 1, we conduct comparison experiments on all the $A_4^4 = 24$ randomly constructed 4-task learning problems for each pair of top categories from 20 Newsgroups. And there exists totally 144 4-task learning problems for the six pairs of top categories. According to the operation in [48], we present the average performance over all 24 problems for each pair of top categories in Table 3. From the table, we can observe that the average performance of the proposed CLU–CBD and CLU–PBD methods is better than other investigated methods in all.

4.4. Performance variation as labeled samples increase

In Fig. 2–3, we present the variation of overall average classification accuracy on the ten datasets in Table 1 as the number of labeled samples increases. Fig. 2 illustrates the results for the dataset 1 to dataset 6, and Fig. 3 shows the results for dataset 7 to dataset 10. From the figures, we observe that as the number of labeled samples increases, the performance of all the methods increases in all, since more labeled samples can contain more data information and build more accurate multi-task classifiers. However, we can still find that the overall accuracy of the proposed CLU–CBD and CLU–PBD methods can always yield higher performance than QDD, MS-RP, VOI and Random methods, since the proposed methods select local and global informative samples for each task at the same time, and this results in higher performance compared with other methods.

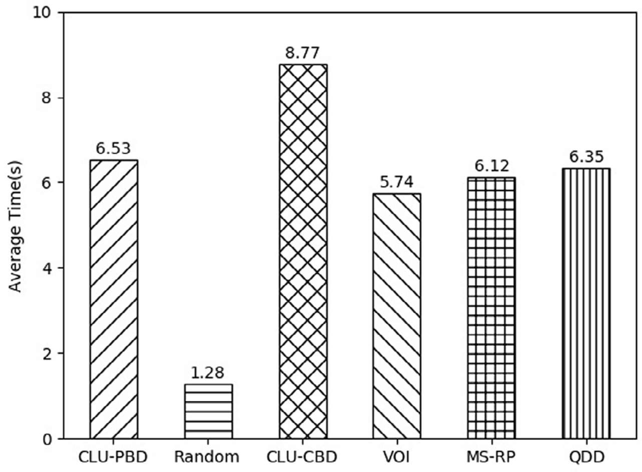


Fig. 4. Average computational time.

4.5. Computational time comparison

Fig. 4 presents the average computational time in seconds required by the investigated methods. From this figure, one can see that the computational time required by Random method is less than other methods, because it selects samples randomly rather than strategically. The proposed CLU–CBD method requires more computational time than other methods, since it adopts the clustering method to avoid sample redundancy. As the comparison of the proposed CLU–CBD and CLU–PBD methods, the CLU–CBD method costs more time than the CLU–PBD method, since the CLU–CBD method utilizes micro-cluster method to separate the data, and requires more time than the partition-based CLU–PBD method. The computational time required by the proposed CLU–PBD is similar to the computational time taken by QDD, MS-RP and VOI, however, it obtains better classification accuracy than QDD, MS-RP and VOI methods.

5. Conclusion

In this paper, we propose two different active learning methods and incorporate them with multi-task SVM to cope with multi-task classification problems. We generalize the proposed methods (CLU–CBD and CLU–PBD) based on CLU in the uncertainty step, and CBD and PBD in the diversity step to multi-task problems. CLU–CBD strategy exploits the support vectors of each hyperplane for queries in the uncertainty step. Furthermore, by means of micro-kernel k-means clustering, it mines the informative samples based on the distribution of the unlabeled samples in the diversity step. CLU–PBD method also exploits the support vectors of each hyperplane for queries in the uncertainty step, and mines the informative samples based on the distribution of the unlabeled samples by means of partition in the diversity step. We carry out the process of active learning simultaneously in each task to ensure that the classifier for each task is optimized. In the experimental analysis, the proposed techniques get compared with other active learning methods adopted in multi-task classification on three datasets. The experimental results show that the proposed methods can take advantage of the informative samples to improve the classification accuracy compared with other active learning methods. In addition, considering that the increase of sample size leads to the explosion of the overall kernel matrix, we deal with the problem by non-negative matrix factorization on kernels.

In the future, we plan to extend the proposed methods to the multi-class classification, in which the correlations between classes and the relevance between samples in different classes will be considered in the selection criteria to collect informative samples for each class.

Acknowledgments

This work was supported in part by the Natural Science Foundation of China under Grant 61876044 and Grant 61672169, in part by the NSFC-Guangdong Joint Found U1501254, in part by the Guangdong Natural Science Funds for Distinguished Young Scholar under Grant S2013050014133.

References

[1] You Ji, Shiliang Sun, Multitask multiclass support vector machines: Model and experiments, Pattern Recognit. 46 (3) (2013) 914–924.

[2] Jiang Zhao, Yitian Xu, Hamido Fujita, An improved non-parallel universum support vector machine and its safe sample screening rule, Knowl. Based Syst. 170 (2019) 79–88.

[3] Xiyan He, Gilles Mourot, Didier Maquin, José Ragot, Pierre Beauseroy, André Smolarz, Edith Grall-Maës, Multi-task learning with one-class SVM, Neurocomputing 133 (2014) 416–426.

[4] Deyu Zhou, Lei Miao, Yulan He, Position-aware deep multi-task learning for drug-drug interaction extraction, Artif. Intell. Med. 87 (2018) 1–8.

[5] Chi Su, Fan Yang, Shiliang Zhang, Qi Tian, Larry S. Davis, Wen Gao, Multi-task learning with low rank attribute embedding for multi-camera person re-identification, IEEE Trans. Pattern Anal. Mach. Intell. 40 (5) (2018) 1167–1181.

[6] Weihua Chen, Xiaotang Chen, Jianguo Zhang, Kaiqi Huang, A multi-task deep network for person re-identification, 2016, CoRR, arXiv:abs/160705369.

[7] Lianyang Ma, Xiaokang Yang, Dacheng Tao, Person re-identification over camera networks using multi-task distance metric learning, IEEE Trans. Image Process. 23 (8) (2014) 3656–3670.

[8] X. Cheng, Nijun Li, Tongchi Zhou, Zhenyang Wu, Lin Zhou, Multi-task object tracking with feature selection, IEICE Trans. 98-A (6) (2015) 1351–1354.

[9] Zhangjian Ji, Weiqiang Wang, Robust object tracking via multi-task dynamic sparse model, in: 2014 IEEE International Conference on Image Processing, ICIP 2014, Paris, France, October 27–30, 2014, 2014, pp. 393–397.

[10] Jian Wu, Anqian Guo, Victor S. Sheng, Pengpeng Zhao, Zhiming Cui, An active learning approach for multi-label image classification with sample noise, IJPRAI 32 (3) (2018) 1–23.

[11] Min Wang, Fan Min, Zhi-Heng Zhang, Yan-Xue Wu, Active learning through density clustering, Expert Syst. Appl. 85 (2017) 305–317.

[12] Begüm Demir, Luca Minello, Lorenzo Bruzzone, Definition of effective training sets for supervised classification of remote sensing images by a novel cost-sensitive active learning method, IEEE Trans. Geosci. Remote Sens. 52 (2) (2014) 1272–1284.

[13] Husheng Guo, Wenjian Wang, An active learning-based SVM multi-class classification model, Pattern Recognit. 48 (5) (2015) 1577–1597.

[14] Saad Mohamad, Moamar Sayed Mouchaweh, Abdelhamid Bouchachia, Active learning for classifying data streams with unknown number of classes, Neural Netw. 98 (2018) 1–15.

[15] Erelcan Yanik, Tevfik Metin Sezgin, Active learning for sketch recognition, Comput. Graph. 52 (2015) 93–105.

[16] Jiayu Chen, Dong Zhou, Ziyue Guo, Jing Lin, Chuan Lyu, Chen Lu, An active learning method based on uncertainty and complexity for gearbox fault diagnosis, IEEE Access 7 (2019) 9022–9031.

[17] Zengmao Wang, Xi Fang, Xinyao Tang, Chen Wu, Multi-class active learning by integrating uncertainty and diversity, IEEE Access 6 (2018) 22794–22803.

[18] Seho Kee, Enrique del Castillo, George Runger, Query-by-committee improvement with diversity and density in batch active learning, Inform. Sci. 454–455 (2018) 401–418.

[19] Saeed Majidi, Gregory R. Crane, Committee-based active learning for dependency parsing, in: Research and Advanced Technology for Digital Libraries - International Conference on Theory and Practice of Digital Libraries, TPDL 2013, Valletta, Malta, September 22–26, 2013. Proceedings, 2013, pp. 442–445.

[20] Handing Wang, Yaochu Jin, John Doherty, Committee-based active learning for surrogate-assisted particle swarm optimization of expensive problems, IEEE Trans. Cybern. 47 (9) (2017) 2664–2677.

[21] Hande Özgür Alemdar, T.L.M. van Kasteren, Cem Ersoy, Active learning with uncertainty sampling for large scale activity recognition in smart homes, JAISE 9 (2) (2017) 209–223.

[22] Oscar Gabriel Reyes Pupo, Carlos Morell, Sebastián Ventura, Effective active learning strategy for multi-label learning, Neurocomputing 273 (2018) 494–508.
[23] Van Cuong Tran, Ngoc Thanh Nguyen, Hamido Fujita, Dinh Tuyen Hoang, Dosam Hwang, A combination of active learning and self-learning for named entity recognition on twitter using conditional random fields, Knowl. Based Syst. 132 (2017) 179–187.
[24] Rohitash Chandra, Yew-Soon Ong, Chi-Keong Goh, Co-evolutionary multi-task learning for dynamic time series prediction, Appl. Soft Comput. 70 (2018) 576–589.
[25] Zihan Liu, Bo Huang, Yuqi Cui, Yifan Xu, Bo Zhang, Lixia Zhu, Yang Wang, Lei Jin, Dongrui Wu, Multi-task deep learning with dynamic programming for embryo early development stage classification from time-lapse videos, IEEE Access 7 (2019) 122153–122163.
[26] Sahil Sharma, Balaraman Ravindran, Online multi-task learning using active sampling, 2017, CoRR, arXiv:abs/170206053.
[27] Liyun Lu, Qiang Lin, Huimin Pei, Ping Zhong, The als-svm based multi-task learning classifiers, Appl. Intell. 48 (8) (2018) 2393–2407.
[28] Han-Tai Shiao, Vladimir Cherkassky, Implementation and comparison of svm-based multi-task learning methods, in: The 2012 International Joint Conference on Neural Networks, IJCNN, Brisbane, Australia, June 10–15, 2012, 2012, pp. 1–7.
[29] Yaran Chen, Dongbin Zhao, Le Lv, Qichao Zhang, Multi-task learning for dangerous object detection in autonomous driving, Inform. Sci. 432 (2018) 559–571.
[30] Pengfei Liu, Xipeng Qiu, Xuanjing Huang, Recurrent neural network for text classification with multi-task learning, in: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9–15 July 2016, 2016, pp. 2873–2879.
[31] Michael Pearce, Jürgen Branke, Continuous multi-task bayesian optimisation with correlation, European J. Oper. Res. 270 (3) (2018) 1074–1085.
[32] Haiqin Yang, Irwin King, Michael R. Lyu, Multi-task learning for one-class classification, in: International Joint Conference on Neural Networks, IJCNN 2010, Barcelona, Spain, 18–23 July, 2010, 2010, pp. 1–8.
[33] Xianpeng Liang, Lin Zhu, De-Shuang Huang, Multi-task ranking SVM for image cosegmentation, Neurocomputing 247 (2017) 126–136.
[34] Qing Liao, Ye Ding, Zoe L. Jiang, Xuan Wang, Chunkai Zhang, Qian Zhang, Multi-task deep convolutional neural network for cancer diagnosis, Neurocomputing 348 (2019) 66–73.
[35] Keelin Greenlaw, Elena Szefer, Jinko Graham, Mary Lesperance, Farouk S. Nathoo, A bayesian group sparse multi-task regression model for imaging genetics, Bioinformatics 33 (16) (2017) 2513–2522.
[36] Andre F. Marquand, Michael J. Brammer, Steven C.R. Williams, Orla M. Doyle, Bayesian multi-task learning for decoding multi-subject neuroimaging data, Neuroimage 92 (2014) 298–311.
[37] Zhang Yi, Yan Yang, Tianrui Li, Hamido Fujita, A multitask multiview clustering algorithm in heterogeneous situations based on LLE and LE, Knowl. Based Syst. 163 (2019) 776–786.
[38] Hao Wang, Yan Yang, Bing Liu, Hamido Fujita, A study of graph-based system for multi-view clustering, Knowl. Based Syst. 163 (2019) 1009–1019.
[39] Edwin Lughofer, Hybrid active learning for reducing the annotation effort of operators in classification systems, Pattern Recognit. 45 (2) (2012) 884–896.
[40] Greg Schohn, David Cohn, Less is more: Active learning with support vector machines, in: Proceedings of the Seventeenth International Conference on Machine Learning, ICML 2000, Stanford University, Stanford, CA, USA, June 29 - July 2, 2000, 2000, pp. 839–846.
[41] Ido Dagan, Sean P. Engelson, Committee-based sampling for training probabilistic classifiers, in: Machine Learning, Proceedings of the Twelfth International Conference on Machine Learning, Tahoe City, California, USA, July 9–12, 1995, 1995, pp. 150–157.
[42] Jingbo Zhu, Huizhen Wang, Tianshun Yao, Benjamin K. Tsou, Active learning with sampling by uncertainty and density for word sense disambiguation and text classification, in: COLING 2008, 22nd International Conference on Computational Linguistics, Proceedings of the Conference, 18–22 August 2008, Manchester, UK, 2008, pp. 1137–1144.
[43] Lian-Zhi Huo, Ping Tang, A batch-mode active learning algorithm using region-partitioning diversity for svm classifier, IEEE J Sel. Top. Appl. Earth Observ. Remote Sens. 7 (4) (2014) 1036–1046.
[44] Yi Zhang, Multi-task active learning with output constraints, in: Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11–15, 2010, 2010.
[45] Giles M. Foody, Thematic map comparison: Evaluating the statistical significance of differences in classification accuracy, Photogramm. Eng. Remote Sens. 70 (5) (2004) 627–634.
[46] Samir Al-Stouhi, Chandan K. Reddy, Multi-task clustering using constrained symmetric non-negative matrix factorization, in: Proceedings of the 2014 SIAM International Conference on Data Mining, Philadelphia, Pennsylvania, USA, April 24–26, 2014, 2014, pp. 785–793.
[47] Rita Chattopadhyay, Wei Fan, Ian Davidson, Sethuraman Panchanathan, Jieping Ye, Joint transfer and batch-mode active learning, in: Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16–21 June 2013, 2013, pp. 253–261.
[48] Changying Du, Fuzhen Zhuang, Qing He, Zhongzhi Shi, Multi-task semi-supervised semantic feature learning for classification, in: 12th IEEE International Conference on Data Mining, ICDM 2012, Brussels, Belgium, December 10–13, 2012, 2012, pp. 191–200.
[49] Ian H. Witten, Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations, Morgan Kaufmann, 1999.

Yanshan Xiao received the Ph.D. degree in computer science from the Faculty of Engineering and In formation Technology, University of Technology, Sydney, Australia, in 2011. She is with the Faculty of Computer, Guangdong University of Technology. Her research interests include data mining and machine learning. She has published papers on IEEE Transactions on Neural Networks and Learning Systems, IEEE Transactions on Cybernetics, Knowledge and Information Systems, and International Joint Conferences on Artificial Intelligence (IJCAI).

Zheng Chang is pursuing a master's degree at the school of Computers, Guangdong University of Technology, China. His research interests include machine learning and data mining.

Bo Liu is with the Faculty of Automation, Guangdong University of Technology. His research interests include machine learning and data mining. He has published papers on IEEE Transactions on Neural Networks, IEEE Transactions on Knowledge and Data Engineering, Knowledge and Information Systems, IEEE International Conference on Data Mining (ICDM), SIAM International Conference on Data Mining (SDM) and ACM International Conference on Information and Knowledge Management (CIKM). His homepage is at https://www.researchgate.net/profile/Bo_Liu144.