Pattern Recognition 43 (2010) 3113–3119

Contents lists available at ScienceDirect

Pattern Recognition

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

Multiple-view multiple-learner active learning

Qingjiu Zhang, Shiliang Sun *

Department of Computer Science and Technology, East China Normal University, 500 Dongchuan Road, Shanghai 200241, PR China

ARTICLE INFO

Article history:
Received 13 June 2009
Received in revised form
3 February 2010
Accepted 3 April 2010

Keywords:
Multiple-view learning
Multiple-learner learning
Active learning
Artificial neural network

ABSTRACT

Generally, collecting a large quantity of unlabeled examples is feasible, but labeling them all is not. Active learning can reduce the number of labeled examples needed to train a good classifier. Existing active learning algorithms can be roughly divided into three categories: single-view single-learner (SVSL) active learning, multiple-view single-learner (MVSL) active learning and single-view multiple-learner (SVML) active learning. In this paper, a new approach that incorporates multiple views and multiple learners (MVML) into active learning is proposed. Multiple artificial neural networks are used as learners in each view, and they are set with different numbers of hidden neurons and weights to ensure each of them has a different bias. The selective sampling of our proposed method is implemented in three different ways. For comparative purpose, the traditional methods MVSL and SVML active learning as well as bagging active learning and adaboost active learning are also implemented together with MVML active learning in our experiments. The empirical results indicate that the MVML active learning outperforms the other traditional methods.

1. Introduction

Unlabeled examples, which can provide additional information, are useful in machine learning. In many scenarios, training a good learner needs a lot of labeled examples. However, in many real-world problems, although people can get lots of unlabeled examples easily, it usually needs much human power to label them, which can be costly and time-consuming. Since labeling requires expensive human labor, whereas unlabeled data is far easier to obtain, learning from labeled and unlabeled data has drawn significant attentions. Semi-supervised learning and active learning can solve this kind of problems from different perspectives.

The main goal of semi-supervised learning is to train better learners or to generate better models by combining the abundant unlabeled data with a small quantity of labeled data. Semi-supervised learning is halfway between unsupervised learning and supervised learning [1]. Unsupervised learning is a class of problems in which only unlabeled data are given, while supervised learning is another class of problems in which only labeled data is used. However, under suitable assumptions, unlabeled data can be very useful to help supervised learning tasks [1]. Semi-supervised learning aims to exploit the abundant unlabeled data to learn an inner data structure, based on which a few labeled data can be used to accomplish classification or regression.

Active learning is closely related to semi-supervised learning. The essence of active learning is to reduce the number of examples needed to be labeled or generated by investigating different values of unlabeled examples. Active learning also assumes that unlabeled data are easy to collect but labeling is expensive. By interactive queries, active learning can operate with significantly fewer labels than that would be needed to achieve a certain performance in a regular supervised learning framework [2]. An active learner differs from a passive learner in that the latter simply receives a data set from the world and then outputs a classifier or model [2]. This paper is mainly focused on active learning.

There have been many studies conducted on active learning [3–7]. Past theoretical and experimental active learning work can be roughly divided into three categories: single-view single-learner (SVSL) active learning, multiple-view single-learner (MVSL) active learning and single-view multiple-learner (SVML) active learning. MVSL active learning solves problems with information from multiple views, which is usually superior to SVSL algorithms. But its setting with a single learner in each view may not result in good results if the learner is not very effective. SVML active learning adopts ensemble techniques to establish a new framework for the considered problems. Ensemble methods can effectively improve the accuracy of learners. Such as bagging [8] and boosting [9,10]. But it is clear that it does not fully use the information, especially when multiple views are available. For the above reasons, a novel approach, multiple-view multiple-learner (MVML) active learning, is proposed in this paper.

MVML active learning benefits from the advantages of both multiple views and multiple learners. In real world domains,

* Corresponding author. Tel.: +86 21 54345186; fax: +86 21 54345119.
E-mail address: slsun@cs.ecnu.edu.cn (S. Sun).

0031-3203/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2010.04.004

some problems have more than one natural feature sets. For example, in webpage classification, people can judge a webpage belonging to which class by the words occurring in that page, or by the words occurring in the hyperlinks pointing to that page. It is very likely to get better learners when employing both of the two views, if the problem assumes that each of the views can train good learners separately. Much research has shown that multiple-view is superior to single-view in solving machine learning problems [11,12]. Ensemble learning considers how to generate or combine an ensemble of learners for classification or regression. The effectiveness of ensemble learning has also been shown by many studies, such as the work in [8,13–15].

Two real-world problems are used in the experiments: webpage classification and web advertisement classification. Both of them have multiple distinct views. All of the empirical results indicate that MVML active learning outperforms MVSL and SVML active learning.

The rest of the paper is organized as follows. Section 2 briefly reviews some related work on active learning. Section 3 thoroughly describes the proposed MVML active learning approach. Section 4 presents our experiments on real-world problems. At last, conclusions and future work are presented in Section 5.

2. Related work

In general, active learning problems have been studied from two aspects: multiple views or single view, multiple learners or single learner. Therefore, there are four combinatorial ways to investigate them: SVSL, MVSL, SVML and MVML. In this section, existing methods of the former three are briefly introduced and remarked, from which we reach our MVML approach.

2.1. Single view single learner

In learning problems, SVSL active learning gets its hypotheses just from one single learner of a single view. SVSL active learning is probably the most original method that people utilized in active learning problems. It assumes there is only one view attainable for learning. If there is more than one view, SVSL active learning solves this problem by using only one of them or the merger of them. Then, a hypothesis can be formed from the selected view or the merged view. Moreover, only one learner is employed in the whole process. In this situation, if the learner is not very effective (including not stable), the final result may be suboptimal.

There are some algorithms belonging to this category. The uncertainty sampling method repeatedly chooses to label an unlabeled example which seems as the most uncertain one [16]. The selected example will be regarded as the most informative one and moved to the training set after giving its authentic label. It is worth noticing that the uncertainty is measured just by one learner. Furthermore, it cannot be applied to multiple-view problems directly. The expectation-maximization (EM) algorithm [17] can also be considered as a SVSL algorithm. When the assumed mode is not consonant with the distribution of the data, EM may not turn to be a good learner [18]. Thus, it has certain limitations in its application fields.

SVSL active learning has its inherent limitations. It is obvious that the learner used in SVSL active learning should be effective. If the learner has a strong inductive bias, it can be hard to predict well. In pattern recognition, people should adopt the best way to utilize the information as much as possible. When there are multiple views available and all of them are adequate to infer the prediction relationship, people may try to combine all the available views to one. But it is better to utilize all these views in parallel with interactions than the all-in-one manner [12]. Hence, SVSL active learning is not suitable for solving multiple-view problems.

2.2. Multiple views single learner

MVSL active learning, originating from the co-training technique [11], extends the SVSL active learning into the multiple-view situation. The major difference between MVSL and SVSL active learning is that the former infers the prediction relationship of a problem from multiple views. For example, an advertisement image can be judged by the surrounding text or by its visual information. If each view is capable of inferring the relationship, it is doubtless that a better learner can be obtained when all the views are appropriately used. MVSL active learning does not merge the multiple views into one view, and instead it uses each view to infer the relationship. MVSL active learning still employs only one learner in each view. The learner running on each view will label data for the other and they will be boosted by this cooperative work.

There are lots of machine learning algorithms based on multiple views. The representative method is co-training for semi-supervised learning [11]. It assumes the problem has two sufficient and conditionally independent views, which means that every view is capable to train a good learner and that the two views are independent given the label information. Two learners, respectively, select unlabeled examples for each other from their corresponding views. Co-test (co-EM) is an active learning approach that incorporates the co-training technique [19]. It has been successfully applied to multiple-view problems, and indicates that this method is not as sensitive as the co-training to the correlation between views. Similar to co-test (co-EM), co-testing is also an active learning algorithm that belongs to MVSL active learning [12].

Although MVSL outperforms SVSL by exploiting multiple views, it still has much space to improve. We can understand this by the fact that recent success on ensemble learning has already shown that almost every kind of classifiers can be further improved by a certain kind of ensembles. For ensemble learning, especially when individual classifiers have a good diversity and accuracy, it is more likely to get an improvement [8–10,13].

2.3. Single view multiple learners

SVML active learning, benefiting from the ensemble technique, integrates multiple learners into one-view problems. Different from MVSL active learning which trains a single learner on different feature sets of the labeled data, SVML active learning runs different learners on the same feature set. SVML active learning mainly leverages the fact that different learners have different biases. When SVML is used for active learning, committees would predict different hypotheses and have different confidence on the unlabeled data. The unlabeled examples on which the committee members have the most disagreement will be selected. The disagreement can be measured by statistical methods, such as voting, entropy, etc.

Query-by-committee (QBC) [16,20] measures the degree to which a group of learners disagree rather than using a single learner to measure the uncertainty of its prediction. Zhou and Goldman presented the democratic co-learning in which learners from multiple algorithms instead of multiple views are used to label data for each other [21]. Both of these two methods assume that there is only one feature set in the learning problem. The final prediction on the unlabeled data is the combined results of all the learners.

The limitation of the learners in the SVML active learning is not so much as that of the MVSL and SVSL active learning due to the adoption of multiple learners which can usually output a more accurate hypothesis. The learners used in SVML active learning can be very diverse. They can be any kind of useful learners with different biases to ensure that they will give different hypotheses on the unlabeled data. If all the learners present the same estimation, it will be a time and cost waste to use multiple learners. But when several views are available, it is no longer sensible to solve the problem by using only one of the views or the merger of them.

2.4. Multiple views multiple learners

MVML active learning, proposed in this paper, is different from the above methods. It is a multi-view approach which integrates a group of learners in each view to promote the overall accuracy. Unlike the bagging [8] and the sampling version of boosting [5] which boost themselves by creating random subsets or purposely biased distributions from the training set, all of the learners in each view run on the whole training data. In the next section, MVML active learning will be described in detail. From existing evidence, people can conclude that both SVML active learning and MVSL active learning outperform SVSL active learning [11,21]. Therefore, the SVSL active learning is not implemented in our experiments; instead, the comparison between MVML active learning and the other two traditional methods will be mainly focused on.

3. MVML active learning

In this section, the MVML active learning approach is presented in detail. As in semi-supervised learning and other active learning, MVML active learning exploits both labeled and unlabeled examples. Moreover, it solves our considered classification problems using both multiple views and multiple learners. In other words, the MVML framework incorporates the co-testing and democratic co-learning techniques.

3.1. Selective sampling

Selective sampling is a typical method for choosing informative examples to be labeled manually in active learning. The most valuable examples should be selected while the ones which cannot provide much information would be discarded. In other words, all the selected examples are regarded as the most informative ones for the learning problem. Because of this, the result is most likely to be superior to that of randomly sampling. In active learning, the example with the largest disagreement is chosen as the most useful one. Many active learning algorithms choose the contention points with maximum entropy or least confidence between learners for human labeling [22].

In MVML active learning, if disagreement is computed between multiple views, entropy is not a good measure for selecting samples. Suppose a binary classification problem contains two views V_1 and V_2, and k learners are combined in each view. m_i^j denotes the number of learners which belong to V_i and classify the predicted example as class j. Consequently, V_1 and V_2 will give their confidences as follows:

$$\begin{cases} P_1^1=(r/2+m_1^1)/(k+r), \\ P_1^2=(r/2+m_1^2)/(k+r), \end{cases} \quad \begin{cases} P_2^1=(r/2+m_2^1)/(k+r), \\ P_2^2=(r/2+m_2^2)/(k+r), \end{cases} \tag{1}$$

where P_i^j denotes the confidence that V_i considers the predicted example as class j. r is a small positive constant to make sure the confidences do not equal zero. However, if the entropy is used to calculate the disagreement between the two views, the paradigm would be like this

$$\begin{cases} P^1=(r/2+m_1^1+m_2^1)/(2k+r), \\ P^2=(r/2+m_1^2+m_2^2)/(2k+r), \end{cases} \tag{2}$$

$$entropy(x)=-P^1\log_2(P^1)-P^2\log_2(P^2). \tag{3}$$

Suppose an example x_1 is predicted by learners from the two views and the results are $(P_1^1, P_1^2)=(0.8, 0.2)$ and $(P_2^1, P_2^2)=(0.2, 0.8)$, respectively, while another example x_2 is predicted with the results (0.6, 0.4) and (0.4, 0.6) corresponding to V_1 and V_2. In this situation, if all the learners of the two views are put together to calculate the entropy, the P^1 and P^2 would be the average of predictions of the two views. It is easy to figure out that both x_1 and x_2 have the same final result (P^1,P^2), which means that it is hard to tell the differences between them by using entropy. Actually, x_1 is more suitable to be selected for MVML active learning algorithms. It can be concluded that this kind of selective sampling not really reflects the disagreement between the two views.

To resolve the above problem, we define a kind of measurement *ambiguity* as follows. *Ambiguity* completely depends on the disagreement between the views, instead of putting all the learners of the two views together. Its formulation is

$$ambiguity(x)=\begin{cases} 1+P_1^j\log_2(P_1^j)+P_2^k\log_2(P_2^k), & H_1\neq H_2, \\ -1-P_1^j\log_2(P_1^j)-P_2^k\log_2(P_2^k), & H_1=H_2, \end{cases} \tag{4}$$

where P_1^j and P_2^k denote the confidences of the suggested classes of V_1 and V_2. For example, if an example x_1 is regarded as $(P_1^1, P_1^2)=(0.8, 0.2)$ and $(P_2^1, P_2^2)=(0.2, 0.8)$, then $P_1^j=P_1^1=0.8$ and $P_2^k=P_2^2=0.8$. Similarly, if an example x_2 is regarded as $(P_1^1, P_1^2)=(0.9, 0.1)$ and $(P_2^1, P_2^2)=(0.7, 0.3)$, then $P_1^j=P_1^1=0.9$ and $P_2^k=P_2^1=0.7$. $H_1=H_2$ means the two views suggest the same label, while $H_1\neq H_2$ means the two views disagreeing on their hypotheses. Below we characterize the properties of the proposed *ambiguity* measurement.

Let $a(x)$ be the *ambiguity* for example x. Suppose P_1^j and P_2^k are the confidence of the two views in Eq. (4), respectively. Then, $0.5\leq P_1^j<1$ and $0.5\leq P_2^k<1$. Consequently, we have $-0.5\leq P_1^j\log_2(P_1^j)<0$ and $-0.5\leq P_2^k\log_2(P_2^k)<0$ as a result of the increased property of function $p\log_2(p)$ for $p\in[0.5,1)$. When $H_1\neq H_2$, $a(x)\geq 0$. When $H_1=H_2$, $a(x)\leq 0$. Thus, the example which is regarded as different classes by the two views will have a larger value and will be selected firstly. This is what entropy cannot do when the disagreement is calculated between the views. Moreover, when the situation belongs to $H_1\neq H_2$, *ambiguity* is an increasing function. In this scenario, the example which has a larger confidence will have a larger value, and it will be regarded as the more informative one to be selected. However, when the situation belongs to $H_1=H_2$, *ambiguity* is a decreasing function. The example which has a smaller confidence will have a larger value, and it still has the priority to be selected. This validates the feasibility of the *ambiguity* for selecting informative examples for active learning.

3.2. Algorithm

The MVML active learning algorithm uses multiple views and multiple learners when learning from the labeled and unlabeled data. It assumes that a small quantity of labeled examples and large quantity of unlabeled examples are available. The algorithm aims to provide good accuracy by incrementally enlarging the labeled set and boosting the learners. Different from conventional

co-training [11] and co-testing [12], multiple learners are used in each view. At the beginning, the learners of the two views are initialized by the small quantity of labeled examples on hand. At every round of the training process every learner will estimate the labels of a pool of unlabeled examples. The algorithm will randomly select several examples on which the two views have the largest disagreement. Then, the selected examples are given correct labels and moved from the unlabeled set to the labeled set. The learners will be rebuilt again based on the enlarged labeled set. At the classification stage, the final hypothesis is calculated by the combined results of the learners of the two views. Our algorithm is illustrated as follows:

```
Given:
  L: labeled set
  U: unlabeled set
Create a pool P by moving u examples from U to P
 randomly
Run for k iterations:
  Use L to train a group of classifiers H1 from view V1
  Use L to train a group of classifiers H2 from view V2
  Both H1 and H2 predict the P by voting measurement
  Choose n examples randomly from the w most
  ambiguous ones
  Label the n examples manually
  Move the newly labeled points from U to L
  Replenish the P by choosing n examples from U at
  random.
```

It is clear that the selected examples are useful for the two views, which means every selected example will significantly boost the learners of the two views. As selective sampling depends on the disagreement between views, each view in some sense helps others in the process of learning. Like co-training [11] and co-testing [12], the algorithm also needs the assumption that all the views should be sufficient to train a good classifier. If the assumption does not hold, this algorithm will probably lose its effectiveness, as a result of inaccurate confidence computation for unlabeled examples. Moreover, when examples are selected the class ratio should be taken into consideration if possible, because suboptimal results would be obtained if added examples for different classes are heavily unbalanced.

4. Experiments

In our experiments, artificial neural network (ANN) is employed as the learner. All the ANNs have the same number of layers but different numbers of hidden neurons, and they are set with different weights randomly at the beginning. Consequently, the learners have different inductive biases and will output different hypotheses on the test data. In our experiments, every network has three layers, and the number of hidden neurons is set according to

$$Neuron\_number = \sqrt{n+m}+a, \tag{5}$$

where n and m denote the number of input and output neurons of the networks, respectively, and a is a constant ranging from 1 to 10. In single-learner active learning, the number of hidden neurons is $\sqrt{n+m}+5$, while in multiple-learner active learning the average number of hidden neurons is equal to that of single-learner active learning algorithms.

Ten-fold cross validation is used in every experiment. At every round of validation one-tenth of the whole data are used as the validation set to stop the training of neural networks while another one-tenth are used as the test data. The final result is the average outcome of the 10 accuracies on the test data. In some figures, not only the final results are shown, but also the standard variances of the results are illustrated. In order to roughly keep the balance of different classes, the selected examples are randomly chosen from a small group of the most informative points instead of being chosen directly from the most informative ones. Moreover, the number of selected examples is approximately 1/4 of the size of the candidate group, and the size of the group is approximately 1/10 of the size of the pool. In this paper, the pool is set around 1/5 of the size of the whole dataset. The dimensionality reduction method PCA is adopted to problems with high dimensions. In the experiments of this paper, more than 90% energy is reserved to ensure that good learners can be trained.

The rest of this section is organized as follows. Initially, several methods for comparison are listed. Then, two experiments are described in detail. At last the comparison of characteristics of the selected examples of those methods is given.

4.1. Methods for comparison

To demonstrate the effect of MVML active learning, other well-known algorithms are also evaluated. They are MVSL active learning, SVML active learning, bagging and adaboost. In the experiments, SVML active learning not only runs on each view, but also runs on the view concatenated by the multiple views. When it runs on the concatenated view, the number of learners is double that of running on single views, and is the same as that of running for the MVML active learning.

MVSL active learning: MVSL active learning is one of the traditional multiple-view active learning methods. In the experiments, the selective sampling is based on the disagreement between views. Therefore, *ambiguity* is suitable for sample selecting. But the confidence of each view is calculated by the individual learner's confidence instead of by committee-based measurement, because there is only one learner running on each view.

SVML active learning: SVML active learning is also a conventional approach. Different from the MVSL active learning, multiple learners are used in each view. However, the algorithm assumes there is only one view. In this scenario, *entropy* which reflects the disagreement between learners is competent to select samples.

MVML active learning: In Section 3, MVML active learning evaluates the disagreement between the two views. However, other aspects of selective sampling is also possible. For example, selective sampling can be based on the disagreement between learners within each view, which means each view selects the example for itself without considering the other views. In this scenario, entropy is qualified to select samples. Actually, combining the two methods is possible to select samples, which means the selective sampling paradigm would be

$$ambiguity' = ambiguity + entropy_1 + entrop_2, \tag{6}$$

where *ambiguity* is the disagreement between the two views, while $entropy_1$ and $entropy_2$ are the within-view disagreements of V_1 and V_2.

Bagging active learning: Bagging is commonly regarded as a standard ensemble method. In order to compare with the above MVML active learning, bagging active learning is also implemented in the experiments. Bagging active learning employs 25 learners in each view and makes up its initial labeled set by randomly sampling with replacement from the labeled set. In each iteration, learners still randomly sample with replacement from the n candidates. Therefore, the difference between bagging

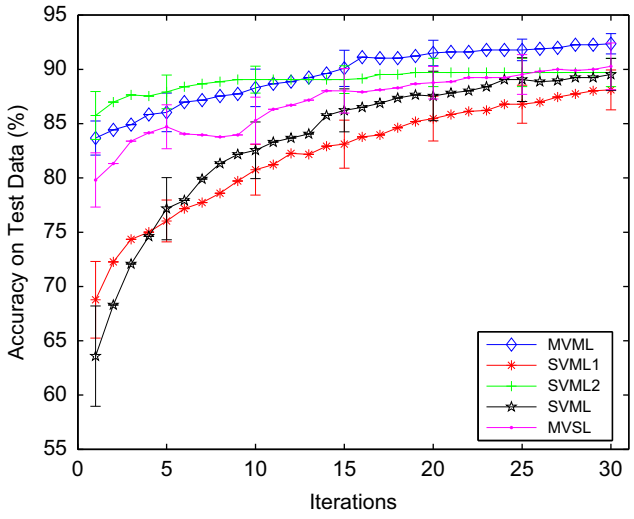

Fig. 1. Webpage classification performance, respectively, launched by MVSL active learning, SVML active learning and MVML active learning. SVML1 and SVML2 represent the results of SVML active learning running on V_1 and V_2, respectively, while SVML represents the algorithm running on the concatenated view of V_1 and V_2.

active learning and MVML active learning is that the former samples the training set. In the experiments of bagging active learning, the *ambiguity* is also evaluated between the multiple views. Since bagging active learning takes advantage of multiple views and multiple learners, it is also a kind of MVML active learning.

Adaboost active learning: Adaboost is a member of the boosting algorithm family. In this paper, adaboost is combined with active learning and co-training to compare with MVML active learning. Adaboost active learning generates 20 learners in each view. Some important parameters are set according to [23]. In the related experiments, the *ambiguity* is still evaluated between the multiple views. The difference between adaboost active learning and MVML active learning is that the learners are combined by different ensemble methods, and adaboost has a different manner to generate individual learners. Note that the adaboost active learning is also a kind of MVML active learning.

4.2. Webpage classification

This data set has two natural views, one over the words occurring in the page and the other over the words appearing in the links pointing to that page [11,24]. The dimensionality of the two views is reduced from 2333 and 87 to 197 and 23, respectively. The data set consists of 230 course pages and 821 non-course pages. Initially, only nine course examples and 36 non-course examples form the labeled set. For calculating purpose, not all unlabeled examples participate in evaluation, and instead a pool consisted of 350 unlabeled ones is evaluated. At every round of iteration, eight samples are randomly selected from 30 of the most informative ones of the pool, and they are given correct labels and moved to the labeled set.

Fig. 1 demonstrates that MVML active learning is superior to MVSL active learning and SVML active learning. Although MVML active learning is not the best one at the first several iterations, it exceeds the others soon as the algorithms proceed. This phenomenon is quite reasonable. As active learning methods do not have a reliable method to keep the class ratio, it is normal to observe the low performance in the beginning as a result of the mismatch between the distribution of the labeled data and the overall distribution. However, as they can select genuinely informative examples, they tend to outperform the baselines after several rounds of queries when the imbalance problem relieves. Fig. 2 shows each view's performance in the process of MVML active learning as well as the performance of bagging and adaboost active learning. It would be easy to come to this conclusion: it is better to use the combined result than just using one view's results. Note that selective sampling of the MVML active learning both in Figs. 1 and 2 is based on the disagreement between the two views.

In order to thoroughly evaluate MVML active learning, selective sampling is also implemented by two other manners within-view sampling and the combination of within-view and between-view sampling. Their effects can be seen from Fig. 3. It is hard to tell which one is the best, but compared with the results in Fig. 1 all of them outperform MVSL active learning and SVML active learning.

4.3. Advertisement classification

This is also a bench mark which has been used in other people's research [25]. This data set has two natural views and two classes consisting of 459 advertisements and 2820 non-advertisements. PCA is used to preprocess the data set before training. The dimensionality of one view drops from 587 to 167, while the dimensionality of the other view falls from 967 to 160. Nine ads and 54 non-ads form the initial labeled set. The size of the pool is 600. At every round of training 14 samples are selected randomly from 56 of the most informative ones of the pool. Then they are manually labeled and moved to the labeled set.

From Fig. 4, it is obviously that MVML active learning outperforms MVSL active learning and SVML active learning during the whole process. Fig. 5 shows each view's performance in the process of MVML active learning as well as the performance of bagging and adaboost active learning. It indicates that the combined result is better than the uncombined results. The MVML active learning both in Figs. 4 and 5 selects examples according to the disagreement between the two views.

To thoroughly evaluate the MVML active learning, the within-view sampling and the combination of within-view and between-view sampling are also implemented. Their effects are shown in Fig. 6 where the combination performs the best, but all of the three results outperform the traditional methods.

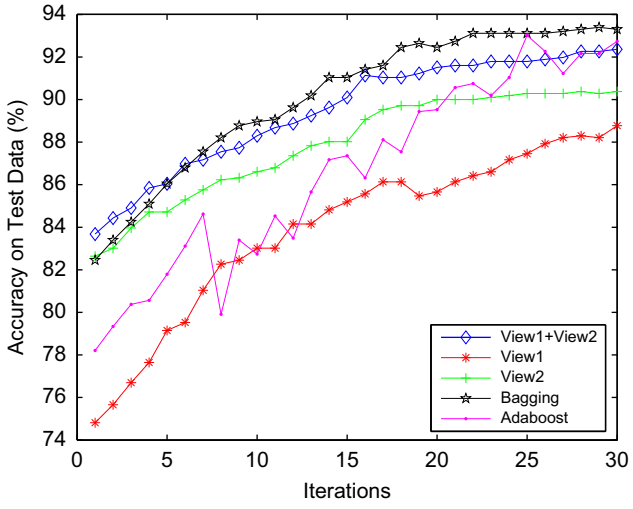

Fig. 2. Bagging active learning performance, adaboost active learning performance and each view's performance in the process of MVML active learning of webpage classification.

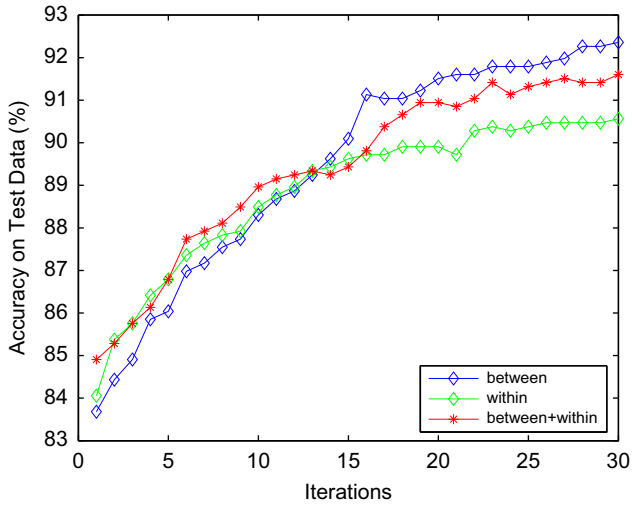

Fig. 3. Webpage classification lunched by MVML active learning. The disagreement of the contention point is measured in three ways: between views, within view and both of them are considered.

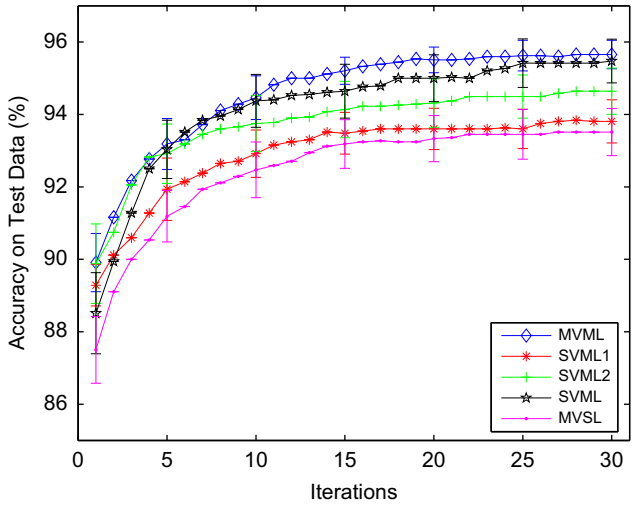

Fig. 4. Advertisement classification performance, respectively, launched by MVSL active learning, SVML active learning and MVML active learning. SVML1 and SVML2 represent the results of SVML active learning running on V_1 and V_2, respectively, while SVML represents the algorithm running on the concatenated view of V_1 and V_2.

4.4. Characteristics of the selected examples

In order to explore the characteristics of the examples selected by those different methods, the average distance between the network outputs and the objective outputs is calculated in order to provide a tentative explanation on the observed performance. It is calculated according to

$$Distance = (\Sigma_1^d \Sigma_1^m \Sigma_1^n |real\_output - objective\_output|)/(d \times m \times n), \quad (7)$$

where d and m denote the times of validations and the number of networks, respectively, and n denotes the number of selected examples in each iteration. Intuitively, this kind of distance may reflect how close the selected example is to the separating hyperplane. The larger the distance is, the closer the example would be to the separating hyperplane. Therefore, if an example has a larger distance, it would be more useful for the learner to further improve its accuracy. MVML method is compared with the other methods at this aspect. The result of the first iteration is presented in Table 1. From this table, we see that the MVML method corresponds to comparatively large distances and thus has the potential to select informative examples from the unlabeled set.

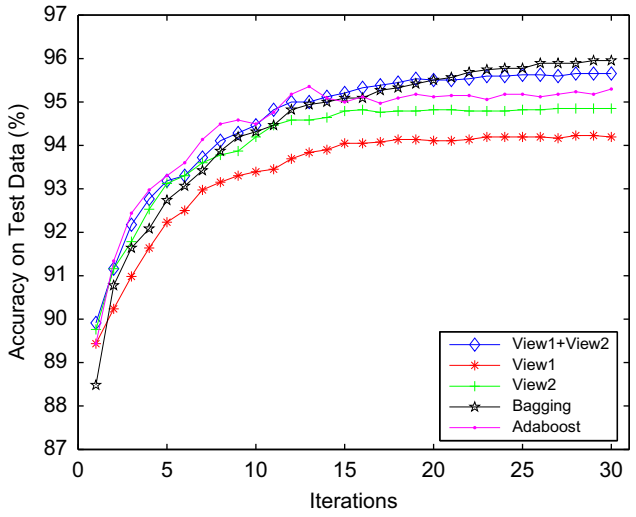

Fig. 5. Bagging active learning performance, adaboost active learning performance and each view's performance in the process of MVML active learning of the advertisement classification.

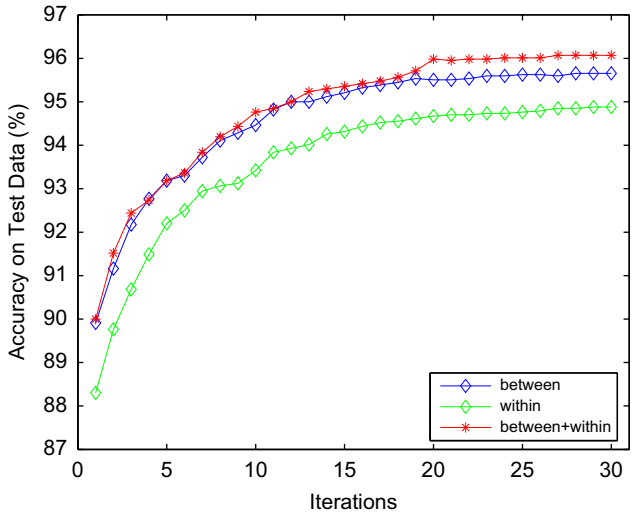

Fig. 6. Advertisement classification lunched by MVML active learning. The disagreement of the contention point is measured in three ways: between views, within view and both of them are considered.

Table 1
The average distance from the real outputs to the objective outputs of the first iteration.

Experiments	MVSL	SVML1	SVML2	SVML	MVML
Webpage	0.6805	0.7115	0.6390	0.6826	0.7172
Advertisement	0.5030	0.5557	0.4892	0.6697	0.5623

5. Conclusions

In this paper a new ensemble-styled approach, the MVML active learning approach, is presented. It benefits from the advantages of both multiple-view learning and ensemble

learning. Two real-world problems webpage classification and advertisement classification are adopted to validate the method. The empirical results show that the existing active learning methods are not as good as our proposed MVML active learning. In the experiments, selective sampling is implemented by three ways:

- choose samples just considering the disagreement between the two views;
- choose samples just considering the disagreement within each view;
- choose samples considering both within-view disagreement and between-view disagreement.

It is hard to tell which one is best, but all of these manners perform well.

MVML active learning is also compared with the bagging active learning and adaboost active learning methods in this paper. The difference between MVML active learning and bagging active learning is that the later combines learners which run on the sampled training sets while MVML active learning runs on the original training set. Research has shown that if sampling the training set can cause significant changes in the learner constructed, the bagging method can improve the accuracy [8]. Moreover, the difference between MVML active learning and adaboost active learning is that the learners are generated and combined in different ways. In the experiments, both bagging active learning and adaboost active learning exhibit comparable performances with MVML active learning. Moreover, since bagging active learning and adaboost active learning still employ multiple learners running on multiple views, they are in fact two special MVML active learning methods. All this indicates that MVML active learning is superior to the traditional MVSL and SVML methods.

We will continue our study on multiple-view semi-supervised learning and active learning. Since only ANNs are used as learners in the current experiments, we plan to use different kinds of classifiers in each view. We will also investigate other ways to promote learning when both labeled and unlabeled data are available.

Acknowledgments

We would like to express our sincere appreciation to the anonymous reviewers for their insightful comments, which have greatly aided us in improving the quality of the paper. This work is supported in part by the National Natural Science Foundation of China under Project 60703005, and by Shanghai Educational Development Foundation under Project 2007*CG* 30.

References

[1] O. Chapelle, B. Schölkopf, A. Zien, Semi-supervised Learning, MIT Press, Cambridge, MA, 2006.
[2] S. Tong, Active learning: theory and applications, Ph.D. Thesis, Stanford University, Stanford, CA, 2001.
[3] D. Cohn, Z. Ghahramani, M. Jordan, Active learning with statistical models, Journal of Artificial Intelligence Research 4 (1996) 129–145.
[4] S. Hoi, R. Jin, J. Zhu, M. Lyu, Batch mode active learning and its application to medical image classification, in: Proceedings of the 23th International Conference on Machine Learning, 2006, pp. 417–424.
[5] Y. Freund, H. Seung, E. Shamir, N. Tishby, Selective sampling using the query by committee algorithm, Machine Learning 28 (2–3) (1997) 133–168.
[6] S. Sun, Active learning with extremely sparse labeled examples, in: Proceedings of Neural Information Processing Systems Workshop on Learning from Multiple Sources, 2008.
[7] D. Zhang, F. Wang, Z. Shi, C. Zhang, Interactive localized content based image retrieval with multiple-instance active learning, Pattern Recognition, 43 (2) (2009) 478–484.
[8] L. Breiman, Bagging predictors, Machine Learning 24 (2) (1996) 123–140.
[9] Y. Freund, R.E. Schapire, A short introduction to boosting, Journal of Japanese Society for Artificial Intelligence 14 (5) (1999) 771–780.
[10] Y. Freund, R.E. Schapire, Experiments with a new boosting algorithm, in: International Conference on Machine Learning, 1996, pp. 148–156.
[11] A. Blum, T. Mitchell, Combining labeled and unlabeled data with co-training, in: Proceedings of the 11th Annual Conference on Computational Learning Theory, 1998, pp. 92–100.
[12] I. Muslea, S. Minton, C. Knoblock, Selective sampling with redundant views, in: Proceedings of Association for the Advancement of Artificial Intelligence, 2000, pp. 621–626.
[13] S. Sun, C. Zhang, D. Zhang, An experimental evaluation of ensemble methods for EEG signal classification, Pattern Recognition Letters 28 (15) (2007) 2157–2163.
[14] S. Sun, C. Zhang, Y. Lu, The random electrode selection ensemble for EEG signal classification, Pattern Recognition 41 (5) (2008) 1663–1675.
[15] Y. Freund, R.E. Shapire, A decision-theoretic generalization of online learning and an application to boosting, Journal of Computer System Science 55 (1) (1997) 119–139.
[16] D.D. Lewis, A.W. Gale, A sequential algorithm for training text classifiers, in: Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1994, pp. 3–12.
[17] A. Dempster, N. Laird, D. Rubin, Maximum likelihood from incomplete data via the EM algorithm, Journal of the Royal Statistical Society, Series B 39 (1977) 1–38.
[18] K. Nigam, A.K. McCallum, S. Thrun, T. Mitchell, Text classification from labeled and unlabeled documents using EM, Machine Learning 39 (2000) 103–134.
[19] I. Muslea, S. Minton, C. Knoblock, Selective sampling + semi-supervised learning = robust multi-view learning, in: Proceedings of the International Joint Conference on Artificial Intelligence, 2001, pp. 435–442.
[20] H.S. Seung, M. Opper, H. Sompolinsky, Query by committee, in: Proceedings of the ACM Workshop on Computational Learning Theory, 1992, pp. 287–294.
[21] Y. Zhou, S. Goldman, Democratic co-learning, in: Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence, 2004, pp. 594–602.
[22] X. Zhu, J. Lafferty, Z. Ghahramani, Combining active learning and semi-supervised learning using Gaussian fields and harmonic functions, in: Proceedings of the International Conference on Machine Learning Workshop on the Continuum from Labeled to Unlabeled Data in Machine Learning and Data Mining, 2003.
[23] P. Viola, M.J. Jones, Robust real-time face detection, International Journal of Computer Vision 57 (2) (2004) 137–154.
[24] V. Sindhwani, P. Niyogi, M. Belkin, A co-regularization approach to semi-supervised learning with multiple views, in: International Conference on Machine Learning Workshop on Learning with Multiple Views, 2005.
[25] Z. Zhou, D. Zhan, Q. Yang, Semi-supervised learning with very few labeled training examples, in: Proceedings of the 22nd AAAI Conference on Artificial Intelligence, 2007, pp. 675–680.

About the Author—QINGJIU ZHANG is a master student in the Pattern Recognition and Machine Learning Research Group, Department of Computer Science and Technology, East China Normal University. His research interests include machine learning, pattern recognition, etc.

About the Author—SHILIANG SUN received the B.E. degree in Automatic Control from Beijing University of Aeronautics and Astronautics and Ph.D. degree in Pattern Recognition and Intelligent Systems from Tsinghua University, respectively, in 2002 and 2007. Now he is an associate professor and the director of the Pattern Recognition and Machine Learning Research Group, Department of Computer Science and Technology, East China Normal University. He is on the editorial boards of several international journals and a referee of some top journals. His research interests include machine learning, pattern recognition, brain–computer interfaces and intelligent transportation systems, etc.