Training Data Selection Based on Fuzzy C-means

Donghai Guan, Weiwei Yuan, Young-Koo Lee and Sungyoung Lee

Abstract—**The performance of supervised learning could be improved when valuable data are selected for training. In this paper, we proposed three data selection methods based on fuzzy c-means algorithm. They are: center-based selection, border-based selection and bin-based selection. In center-based selection, the data with high degree of membership in each cluster are selected for training. In border-based selection, the data around the borders between clusters are selected. In bin-based selection, the data in each cluster are sorted based on their membership degrees. Then for each cluster, the sorted data are divided into bins. Finally, there is one data selected from each bin for training. The effects of them are empirically studied on a set of UCI data sets. Experimental results indicate that bin-based selection could effectively improve the performance of learning compared to randomly selecting training samples.**

I. INTRODUCTION

Supervised learning is one primary sub-field of classical machine learning. The design of a supervised learning system consists of the following stages: data collection and annotation, feature selection, model selection and training, and classifier evaluation. In this work, we focus on data collection and annotation.

Data collection and annotation is a crucial part of supervised learning system because it determines the success of later stages. Moreover, data collection and annotation is surprisingly time consuming and costly. For example, it is well known that 80 percent of the effort in data mining projects is spent on data collection, cleaning and annotation [1].

The widely used approach for data collection and annotation is called passive learning, where samples are randomly and independently selected from the underlying distributions. Then human assessors manually annotate these samples. Considering the time and cost associated with annotation, the number of labeled training data is always limited. In many applications, such as speech recognition and web page classification, it is easy enough to collect a large amount of data, while it is tedious to annotate the data. In these situations, training data selection is a suitable approach to minimize the effort of annotation. Training data selection is known as active learning [2][3]. In active learning, the learning process iteratively queries unlabeled samples to select the most informative samples to annotate and update its learned models. Therefore, the unnecessary and redundant annotation is avoided.

This paper proposes three new data selection methods based on fuzzy c-means algorithm. Our methods first partition the given unlabeled samples into clusters and then select the most representative ones from each cluster to label. Our proposed methods are center-based selection (CS), border-based selection (BS) and bin-based selection (BINS). In CS, the data with high degree of membership in each cluster are selected. Center-based selection is named because the selected samples are usually close to the cluster centers. BS selects training samples around the borders between clusters. In BINS, firstly, the data in each cluster are sorted based on their membership degrees. For each cluster, the sorted data are divided into bins. Then there is one sample selected from each bin for training.

Compared with existing works on data selection, the proposed fuzzy c-means based methods do not require much computational effort. Moreover, they can be used with any type of supervised learning method.

We regard the performance of training data random selection (RS) as baseline and compare it with that of CS, BS and BINS. Experimental results indicate that BINS is a robust method that outperforms RS in all the experimental data sets. On the other hand, CS and BS cannot show significant improvement over RS.

The rest of this paper is organized as follows. In Section II, related work in active learning is presented. Section III presents our proposed three data selection mechanisms. Section IV reports on the experimental results. Section V discloses conclusions and future work.

II. RELATED WORK

For a data set $D = \{x_1, x_2, ..., x_n\} \subset R^d$, let D_l represent the labeled set in which every sample is given a label and $D_u = D - D_l$. Most active learning systems comprise two parts: a learning engine and a selection engine. The learning engine uses a supervised learning algorithm to train a classifier on D_l at every iteration. The selection engine then selects a sample from D_u and requests a human expert to label the sample before passing it to the learning engine. The goal of active learning is to achieve the best possible classifier with a reasonable number of calls for labeling by human help.

Existing work on active learning can be characterized by the learning algorithms used by the learning engine, which include multilayer perceptron [4], combination of naïve bayes and logistic regression [2], support vector machine [5][6][7]

Donghai Guan is with the Computer Engineering Department, Kyung Hee University, Korea (e-mail: donghai@oslab.khu.ac.kr).

Weiwei Yuan is with the Computer Engineering Department, Kyung Hee University, Korea (e-mail: weiwei@oslab.khu.ac.kr).

Young-Koo Lee is with the Computer Engineering Department, Kyung Hee University, Korea.(e-mail: yklee@ khu.ac.kr).

Sungyoung Lee (Corresponding author) is with the Computer Engineering Department, Kyung Hee University, Korea (e-mail: sylee@oslab.khu.ac.kr).

978-1-4244-1819-0/08/$25.00©2008 IEEE

and so on.

The central part in active learning is data selection strategy. Most existing work has concentrated on two strategies: certainty-based and committee-based selection. In the certainty-based strategy, an initial system is trained using D_l [8][9][10][11]. Then the system labels the samples in D_u and determines the certainties of its predictions of them. The sample with the lowest certainty is then selected and presented to the experts for annotation. In the committee-based methods, a distinct set of classifiers is created using D_l [12][13][14][15]. The sample in D_u, whose label differs most when presented to different classifiers are presented to the experts for annotation. In both paradigms, a new system is trained using the new set of annotated samples, and this process is repeated until it reaches the predefined rounds or some stopping criteria are satisfied.

Different with above methods which using supervised methods for data selection, we propose three new strategies based on fuzzy c-means algorithm that is unsupervised method. Our methods first partition the given unlabeled samples into clusters and then select the most representative ones from each cluster to label.

Note that existing data selection methods require much computational effort. The reason is that many iterations are needed. And one iteration requires one training and classification (for certainty-based selection) or multiple training and classification processes (for committee-based selection). For our proposed methods, the only computational effort is clustering. As we know, clustering requires much less computational effort than training. Even our methods might not provide so nice performance as existing methods as much less computational effort is required, they are still useful if they outperform random selection (passive learning).

III. Our Proposed Data Selection Methods

Because our proposed methods are based on fuzzy c-means algorithm, a brief introduction to fuzzy c-means is given in Section A firstly. Then we present our methods in Section B.

A. Fuzzy C-means

Fuzzy C-means clustering (FCM) [16] is a popular data clustering algorithm and it combines K-means clustering with fuzzy logic. As with fuzzy sets [17], using FCM, each data point can be a member of more than one cluster with different degrees of membership function between 0 and 1. FCM is an objective function based clustering method. Here objective function measures the overall dissimilarity within clusters. By minimizing the objective function we can obtain the optimal partition. Let $X = \{x_1, x_2, ..., x_n\}$ denote the measured data set. The FCM objective function J is defined as:

$$J = \sum_{i=1}^{n} \sum_{j=1}^{c} (u_{ij})^m \left\| x_i - v_j \right\|^2 \quad (1)$$

Clustering of FCM is carried out through an iterative minimization of J according to the following steps:

S1: Choose fuzzy factor (m), number of clusters (c) and c initial cluster centers v_j.

REPEAT

S2: At iteration t, compute u_{ij} with v_j by

$$u_{ij} = \left(\sum_{k=1}^{c} \frac{\left\| x_i - v_j \right\|^{2/(m-1)}}{\left\| x_i - v_k \right\|^{2/(m-1)}} \right)^{-1} \quad (2)$$

S3: Update v_j with u_{ij}, by

$$v_j = \frac{\sum_{i=1}^{n} u_{ij}^m \cdot x_i}{\sum_{i=1}^{n} u_{ij}^m} \quad (3)$$

UNTIL ($\left\| V_t - V_{t-1} \right\| \le \varepsilon$, V_t and V_{t-1} denote the vector of clusters centers at iteration t and $t-1$ respectively, ε is convergence criterion with $0 < \varepsilon < 1$)

Here u_{ij} is the degree of membership of x_i in cluster j and m is the fuzzy factor that determines the degree of fuzziness ($m > 1$). As m approaches one, fuzziness degrades and the FCM algorithm approaches to the standard K-means algorithm. $V = \{v_1, v_2, ..., v_c\}$ is the vector of cluster centers. $\left\| x_i - v_j \right\|^2$ is any norm expressing the similarity between the measured data x_i and the center v_j.

B. Our Methods

Fuzzy c-means computes the cluster centers and generates the class membership matrix. In this work, center-based selection, border-based selection and bin-based selection mechanisms are implemented through analyzing membership matrix. Our proposed methods include:

Center-based selection: This selection method selects the samples with high degree of membership in each cluster. We extract these samples through analyzing membership matrix $U_{(n \times m)}$. Here n is the number of samples partitioned and m is the number of clusters. u_{ij} is the element at the i^{th} row and j^{th} column in U, which denotes the degree of membership of sample x_i in cluster j. In each cluster j ($j = 1:m$), if $i^* = \arg\max_{i=1:n} u_{ij}$, then sample x_{i^*} is regarded as the most representative sample in this cluster to be selected. The next selected sample is $x_{i^{**}}$ with $i^{**} = \arg\max_{i=1:n, i \neq i^*} u_{ij}$. In turn, other samples in cluster j will be selected, until the number of data equals to k_j (the number of training data allocated to cluster j). Usually in active learning, we are

given the total number of training data K ($K = \sum_{j=1}^{m} k_j$) instead of k_j, so how to determine k_j with the knowledge of K is another issue in center-based selection. A simple way we used is to select same or similar number of samples from each cluster ($k_j \cong \frac{K}{m}$).

Border-based selection: This selection strategy selects the samples around borders of clusters. Here we say a sample is located at the border between clusters when its two high degrees of membership are very similar. For example, a data set comprises three clusters. For a sample of it, when its degrees of membership in each cluster is [0.5, 0.49, 0.01], its two high membership degree (0.5 & 0.49) are very similar. In this case, we can say that this sample is located at the border between cluster 1 and 2. Membership matrix $U_{(n\times m)}$ is also used in this part. Here n is the number of samples partitioned and m is the number of clusters. For each sample x_i ($i = 1:n$), if $j^* = \arg\max_{j=1:m} u_{ij}$ and $j^{**} = \arg\max_{j=1:m, j\neq j^*} u_{ij}$, then T_i ($T_i = u_{ij^*} - u_{ij^{**}}$) is calculated. Finally sample x_{i^*} with $i^* = \arg\min_{i=1:n} T_i$ is regarded as the most representative sample. In turn $x_{i^{**}}$ with $i^{**} = \arg\min_{i=1:n, i\neq i^*} T_i$ is the next valuable data to be selected. Other samples will be selected using this way until the number of selected reaches the limitation.

Bin-based selection: This selection strategy assumes that the samples from center-based selection or border-based selection alone cannot provide comprehensive view for learning. Hence it selects the samples uniformly from the whole input vector space. In detail, if we want to select k_{j^*} samples from cluster j^*, the procedures are:

1) Extract the samples which belong to cluster j^*. Suppose n is the number of samples partitioned and m is the number of clusters. For each sample x_i ($i = 1:n$), if $j^* = \arg\max_{j=1:m} u_{ij}$, then sample x_i belongs to cluster j^*.

2) Sort the samples obtained from procedure 1 based on their membership degrees in cluster j^*. In this work, this sorting is from low to high.

3) Dividing the sorted samples obtained from procedure 2 into k_{j^*} bins. Assuming the number of samples belonging to cluster j^* is n_{j^*}. Then the number of samples in bin_1, bin_2 ,… $bin_{(k_{j^*}-1)}$ is $\left\lfloor n_{j^*} / k_{j^*} \right\rfloor$. The number of samples in $bin_{k_{j^*}}$ is $n_{j^*} - \left\lfloor n_{j^*} / k_{j^*} \right\rfloor \times (k_{j^*} - 1)$.

4) Data are selected from each bin obtained from procedure 3. In bin_1, the leftmost sample, which has the lowest membership degree in cluster j^*, is selected. In $bin_{k_{j^*}}$, the rightmost sample, which has the highest membership degree in cluster j^*, is selected. Then one sample will be randomly selected from each other bins.

When k_{j^*} equals to one, we will select the rightmost sample for training. In this case, it is same with center-based-selection mechanism. Moreover, for bin-based selection, we will select the same or similar number of samples from each cluster.

IV. EXPERIMENTAL RESULTS

In this work, training samples are selected by analyzing membership matrix computed by fuzzy c-means. Fuzzy c-means is configured as follows: Fuzzy factor (m , in Section III.A) is set to 2. Convergence criterion (ε , in Section III.A) is set to 0.00001. Maximum iteration is set to 100. Euclidean distance is used as the similarity measure.

To test the effect of our proposed methods, a classifier is needed. In this empirical study, Multilayer Perceptron neural network with back propagation (BP) training algorithm is used. In all the experiments, the network with one hidden layer is adopted. TANSIG, LOGSIG activation functions are used in the hidden layer and output layer respectively. Let n_1, n_2, n_3 denote the number of input nodes, hidden nodes and output nodes respectively. In our experiments, n_1 is the number of attributes in each sample, $n_2 = 2\times n_1 + 1$, n_3 is the number of classes. Consider the example of the iris data set. It contains four attributes and classifies them into three classes. In this case, a 4-9-3 network is used. Each network is trained to 100 epochs. Note that, since the relative instead of absolute performance of the proposed methods are concerned, the architecture and training process of the neural networks have not been finely tuned.

Four well-known data sets are used in our study. The first data set used is the well-known iris dataset [18]. It contains of four characteristics of iris plants and classifies them into three classes of iris with 50 exemplars in each class. One class is linearly separable from the other two which are not linearly separable from each other. In our experiment, 75 samples in iris are randomly selected as test data. Training data are selected from the other 75 samples. Different numbers of training data are used, which include 6, 9, 15, 21 and 27. The learning performance on iris is shown in Table I. In Table I (and Table II, III, IV), T denotes the number of training data. RS, CS, BS and BINS represent random selection, center-based selection, border-based selection and bin-based selection respectively. The value following "±" gives the standard deviation and the best result on each training data number is shown in bold face.

The second data set is Dr. William W. Wolberg's Wisconsin Breast Cancer Dataset [18]. Originally this dataset

contains 699 samples with 458 samples in the class Benign and 241 samples in the class Malignant. Each sample has 9 input features. There are 16 samples with incomplete features. After filtering out those samples, 683 samples are used in our experiment. 100 samples are randomly selected as test data. Training samples are selected from the other 583 samples. Training data numbers include 6, 10, 14, 20 and 26. The learning performance on this dataset is shown in Table II.

TABLE I
ACCURACY COMPARISON ON IRIS

T	Dataset: iris			
	Accuracy Comparison			
	RS	CS	BS	BINS
6	0.720 ± 0.139	0.775 ± 0.075	0.487 ± 0.153	**0.845 ± 0.076**
9	0.807 ± 0.109	0.794 ± 0.063	0.527 ± 0.118	**0.865 ± 0.066**
15	0.867 ± 0.080	0.816 ± 0.075	0.544 ± 0.119	**0.903 ± 0.055**
21	0.910 ± 0.054	0.839 ± 0.065	0.598 ± 0.079	**0.924 ± 0.050**
27	0.913 ± 0.061	0.860 ± 0.059	0.670 ± 0.146	**0.932 ± 0.039**
Ave.	0.843	0.817	0.565	**0.894**

TABLE II
ACCURACY COMPARISON ON BREAST CANCER

T	Dataset: breast cancer			
	Accuracy Comparison			
	RS	CS	BS	BINS
6	0.860 ± 0.116	**0.926 ± 0.042**	0.302 ± 0.183	0.923 ± 0.040
10	0.910 ± 0.070	0.923 ± 0.046	0.328 ± 0.155	**0.953 ± 0.028**
14	0.926 ± 0.052	0.924 ± 0.038	0.306 ± 0.180	**0.933 ± 0.039**
20	0.930 ± 0.046	0.930 ± 0.035	0.334 ± 0.230	**0.957 ± 0.026**
26	0.946 ± 0.031	0.934 ± 0.041	0.584 ± 0.295	**0.956 ± 0.034**
Ave.	0.914	0.927	0.371	**0.944**

The third data set used is the wine dataset [18]. It contains 178 samples with 59, 71 and 48 in three classes respectively. Each sample has 13 input features. 50 samples are randomly selected as test data. Training samples are selected from the other 128 samples. The number of training data includes 6,9,12,15,18,21 and 30. The learning performance on wine dataset is shown in Table III.

The last data set is german dataset [18]. It contains 1000 samples with 700 and 300 in two classes respectively. Each sample has 24 input features. 100 samples are randomly selected as test data. Then training samples are selected from the other 900 samples. The number of training data includes 10,20,30 and 40. The learning performance on german dataset is shown in Table IV.

TABLE III
ACCURACY COMPARISON ON WINE

T	Dataset: wine			
	Accuracy Comparison			
	RS	CS	BS	BINS
6	0.698 ± 0.149	**0.838 ± 0.064**	0.760 ± 0.123	**0.838 ± 0.079**
9	0.779 ± 0.110	0.842 ± 0.068	0.805 ± 0.100	**0.858 ± 0.068**
12	0.822 ± 0.089	0.807 ± 0.069	0.845 ± 0.081	**0.864 ± 0.066**
15	0.842 ± 0.073	0.815 ± 0.069	0.843 ± 0.071	**0.861 ± 0.058**
18	0.857 ± 0.061	0.829 ± 0.052	0.851 ± 0.060	**0.867 ± 0.060**
21	0.850 ± 0.067	0.843 ± 0.056	0.858 ± 0.074	**0.878 ± 0.051**
30	0.883 ± 0.049	0.880 ± 0.052	0.885 ± 0.051	**0.889 ± 0.051**
Ave.	0.819	0.836	0.835	**0.865**

TABLE IV
ACCURACY COMPARISON ON GERMAN

T	Dataset: german			
	Accuracy Comparison			
	RS	CS	BS	BINS
10	0.627 ± 0.116	**0.659 ± 0.051**	0.516 ± 0.105	0.658 ± 0.073
20	0.651 ± 0.070	0.660 ± 0.066	0.618 ± 0.079	**0.661 ± 0.061**
30	**0.670 ± 0.058**	0.647 ± 0.052	0.547 ± 0.090	0.667 ± 0.068
40	0.662 ± 0.053	0.639 ± 0.051	0.578 ± 0.078	**0.675 ± 0.057**
Ave.	0.653	0.651	0.565	**0.665**

Let us see the experimental results given in above tables. In Table I, center-based selection is better than random selection only when the number of training data is small (T=6). For border-based selection, it is always worse than random selection. The promising results come from bin-based selection. It provides better learning performance at all the different training numbers we used. In term of classification error rate, its average improvement on random-selection is 32.48 percent.

In Table II, center-based selection is better than random-selection when the number of training data is small (T=6 and 10). For border-based selection, it is always worse than random-selection. For bin-based selection, it is best in almost all (except T=6) the different training numbers for this dataset. In term of classification error rate, its average improvement on random-selection is 34.88 percent.

In Table III, center-based selection is better than random selection when the number of training data is small (T=6 and 9). Different with last two experiments, for this dataset, border-based selection is better than random selection almost in all the cases. Same with previous result, bin-based selection is best in all the cases. In term of classification error rate, its average improvement on random-selection is 25.41

percent.

In Table IV, center-based selection is better than random selection when the number of training data is small (T=10 and 20). For border-based selection, it is worse than random selection in all the cases. For bin-based selection, it is better than random selection in most cases. In term of classification error rate, its average improvement on random-selection is 3.46 percent.

In summary, the observations reported in this section indicate that:

1) Center-based selection is better than random selection when the number of training data is quite small.

2) Border-based selection is worst among these four methods on most of data sets.

3) Bin-based selection is best among these four methods on all the four data sets.

V. CONCLUSIONS AND FUTURE WORK

Training data is indispensable for supervised learning. In some cases, we must limit the number of training data, because annotating data is time-consuming and tedious. Therefore, how to achieve a good classifier as best as possible with a reasonable number of labeled data is an important issue.

Training data selection is a one of the topics that can be used to solve this issue. In this work, we propose to use fuzzy c-means for data selection. Three data selection mechanisms are proposed: center-based selection (CS), border-based selection (BS) and bin-based selection (BINS).

Experimental results show that among these three mechanisms, BINS can consistently provide significant improvement compared to random selection method. On the other hand, CS and BS cannot achieve significant improvement over random selection. In current work, we do not consider the distribution of samples in each cluster. In the future, we will utilize this information to refine the selection of BINS. This refinement is expected to generate better result.

ACKNOWLEDGMENT

This work is financially supported by the Ministry of Education and Human Resources Development (MOE), the Ministry of Commerce, Industry and Energy (MOCIE) and the Ministry of Labor (MOLAB), Korea, through the fostering project of the Lab of Excellency.

REFERENCES

[1] I.K. Sethi, "Data Mining: An Introduction", *Data Mining for Design and Manufacturing*, D. Braha, ed., pp. 1-40, Kluwer Academic, 2001.

[2] D.D. Lewis and W.A. Gale, "A sequential algorithm for training text classifiers", Proceedings of 17th ACM International Conference on Research and Development in Information Retrieval, 1994, pp. 3-12.

[3] D. Mackay, "Information-based objective functions for active data selection", *Neural Computation,* 4, 4, 1992, pp. 305-318.

[4] K. Fukumizu, "Statistical active learning in multilayer perceptrons", *IEEE Transaction on Neural Networks*, 11, 1, 2000, pp. 17-26.

[5] C. Campbell, N. Cristianini and A. Smola, "Query learning with large margin classifiers", *Proceedings of 17th International Conference on Machine Learning*, 2000, pp. 111-118.

[6] S. Tong and D. Koller, "Support vector machine active learning with application to text classification", *Journal of Machine Learning Research*, 2, 2001, pp. 45-66.

[7] G. Schohn and D. Cohn, "Less is more: active learning with support vector machines", *Proceedings of 17th International Conference on Machine Learning*, 2000, pp. 839-846.

[8] D.D. Lewis and J. Catlett, "Heterogeneous uncertainty sampling for supervised learning", *Proceedings of 11th International Conference on Machine Learning*, 1994, pp. 148-156.

[9] C. Thompson, M.E. Califf and R.J. Mooney, "Active learning for natual language parsing and information extraction", *Proceeding of 16th International Conference on Machine Learning*, 1999, pp. 406-414.

[10] M. Tang, X. Luo and S. Roukos, "Active learning for statistical natural language parsing", *Proceeding of 40th Anniversary Meeting of Association for Computational Linguistics*, 2002, pp. 120-127.

[11] R. Hwa, "On minimizing training corpus for parser acquisition", *Proceeding of 5th Computational Natural Language Learning Workshop*, 2001, pp. 84-89.

[12] D. Cohn, L. Atlas and R. Ladner, "Improving generalization with active learning", *Machine Learning Journal,* 15, 1994, pp. 210-221.

[13] I. Dagan and S.P. Engelson, "Committee-based sampling for training probabilistic classifiers", *Proceeding of 12th International Conference on Machine Learning*, 1995, pp. 150-157.

[14] I.A. Muslea, "Active learning with multiple views", *Ph.D. dissertation*, Univ. Southern California, 2000.

[15] R. Liere, "Active learning with committees: an approach to efficient learning in text categorization using linear threshold algorithms", *Ph.D. dissertation*, Oregon State Univ. 2000.

[16] J.C. Bezdek, *Pattern recognition with fuzzy objective function algorithms*, Plenum Press, New York, 1981.

[17] L.A. Zadeh, "Fuzzy sets as a basis for a theory of possibility", *Fuzzy Sets and Systems,* 1, 1, 1978, pp. 3-28.

[18] UCI Machine Learning Repository, http://www.ics.uci.edu/~mlearn/MLRepository.html.