Feature Reduction Using Ensemble Approach

Yingju Xia Cuiqin Hou Zhuoran Xu Jun Sun
Fujitsu Research & Development Center Co.,LTD.
355Unit 3F, Gate 6，Space 8,Pacific Century Place,
No.2A Gong Ti Bei Lu, Chaoyang District, Beijing 100027
{yjxia, houcuiqin, xuzhuoran, sunjun}@cn.fujitsu.com

Abstract

The performance of many content analysis methods heavily dependent on the features they are applied. A fundamental problem that makes the content analysis difficult is the curse of dimensionality. In this study, we propose a novel feature reduction method which adopts ensemble approach to measure the divergence between the training set and test set and use the divergence to supervise the feature reduction procedure. The proposed method uses pairwise measure to get the diversity between classifiers and selects the complementary classifiers to get the pseudo labels on test set. The pseudo labels are used to measure the divergence between training set and test set. The feature reduction algorithm merges the adjacent feature space according to the divergence, such reduce the feature number. We evaluated the proposed method on several standard datasets. Experiment results shown the efficiency of the proposed feature reduction method.

1 Introduction

A large number of electronic textual documentations are generated everyday on webs and the Internet. For example: e-books, e-newspapers, e-magazines, and essays in blogs. It is difficult for web administrators to manage and classify numerous electronic documentations manually (Ng et al. 1997; Combarro et al. 2005; Gao and Chien, 2012; Robati et al., 2015). It makes the content analysis tools more and more important. A main problem is the high dimensions of features which not only increase the processing time but also decrease the performance of analysis tools. Automatic feature reduction or selection methods are usually used to reduce the number of features (Reif and Shafait 2014). Removing irrelevant or redundant features not only improves performance, but also reduces the dimensionality of the data thereby shortening the training and application time of the learning scheme, building better generalizable models, and decreasing required storage. Furthermore, shorter feature vectors help the content analysis tools in better coping with the curse of dimensionality.

There is a vast literature on the feature reduction (How and Kiong, 2005; Garcia et al., 2013; Choudhary and Saraswat, 2014). When dealing with the features with continuous (real) values, the feature reduction can be regarded as discretization procedure which aim at finding a representation of each feature that contains enough information for the learning task at hand, while ignoring minor fluctuations that maybe irrelevant for that task (Ferreira and Figueiredo, 2012). In practice, discretization can be viewed as a feature reduction method since it maps data from a huge spectrum of numeric values to a greatly reduced subset of discrete values (Garcia et al., 2013).

Actually, the techniques in Garcia et al.(2013) can also be adopted to discrete values. The feature reduction task can be defined as following:

Assuming a data set consisting of N examples and C target classes, for a feature A in this data set with

29th Pacific Asia Conference on Language, Information and Computation: Posters, pages 309 - 318
Shanghai, China, October 30 - November 1, 2015

continuous values which has the range [d_0, d_m], or a set of discrete values (d_0, d_1, …, d_m). The feature reduction algorithms aim to put these values into several bins or intervals: D = {[d_0, d_1], [d_1, d_2], …, [d_{m-1}, d_m]}. Each feature value is then mapped into the bin or interval in which it falls. By tuning the number of the bins, the feature space can be reduced.

Two major categories of feature reduction techniques include unsupervised and supervised methods. Unsupervised methods (Bay, 2001; Li and Wang, 2002; Yang and Webb, 2009) do not consider the class label whereas supervised ones do. (Wu, 1996; Kerber, 1992; Zighed et al., 1998; Singh and Minz, 2007; Jin et al., 2009; Jiang et al., 2010) Comprehensive listings of these techniques can be found in the works of Garcia et al. (2013). The main drawback of all the previous work is the difficulty to accurately handle the gap between the training set and test set. Once the test set changes, the previous trained model cannot catch the property of the new test set.

In this study, we propose a novel feature reduction method which adopts ensemble approach to evaluate the difference/divergence between training set and test set. The divergence is used to merge and modify the feature space, such reduce the feature number. The remaining sections of the paper are organized as follows. Section 2 presents our methods for feature reduction. Section 3 reports experimental results on standard datasets. Section 4 presents concluding remarks and future work.

2 Method

2.1 Related work

As shown by Dougherty el al. (1995), the unsupervised methods and supervised methods are different in the way they use the instance labels. The unsupervised methods do not make use of the instance labels. In contrast, supervised methods utilize the class labels of instances. The representative unsupervised method are Equal Width and Equal Frequency. The Equal Width method divides the range of observed values for a feature into k equal sized bins, where k is a user-supplied parameter. Equal Frequency method divides a continuous variable into k bins where (given m instances) each bin contains m/k (possibly duplicated) adjacent values. Take a feature which is observed to have values bounded by d_0 and d_m ([d_0, d_m]), the Equal Width method computes the bin width:

$$\delta = \frac{d_m - d_0}{k}$$

The bin boundaries are constructed at $d_0 + i\,\delta$, where i = 1, …, k-1, thus the intervals will be { [d_0, $d_0+\delta$], ($d_0+\delta$, $d_0+2\delta$], …, ($d_0+(k-1)\delta$, $d_0+k\delta$] }

The method is applied to each feature independently. It makes no use of instance class information. Since these unsupervised methods do not utilize instance labels in setting partition boundaries, it is likely that classification information will be lost by binning as a result of combing values that are strongly associated with different classes into the same bin (Kerber, 1992). In some cases this could make effective classification much more difficult.

As mentioned above, the supervised methods utilize the instances labels to adjust the bin/interval borders. The simplest way may be to place interval borders between each adjacent pair of examples that are not classified into the same class. Suppose the pair of adjacent values on feature A are x_1 and x_2, $x=(x_1+x_2)/2$ can be taken as an interval border. If the feature A is very informative, which means that positive and negative examples take different value intervals on the attribute, this method is very efficient and useful. However, this method tends to produce too many intervals on those attributes which are not very informative. Such many other supervised methods have been proposed. The representative method is Bayesian method (Wu, 1996).

According to Bayes formula,

$$P(c_j|x) = \frac{P(x|c_j)P(c_j)}{\sum_{k=1}^{m} P(x|c_k)P(c_k)} \quad (1)$$

Where $P(c_j|x)$ is the probability of an example belonging to class c_j if the example takes value x. $P(x|c_j)$ is the probability of the example taking value x on the feature if it is classified in the class c_j.

Given $P(c_j)$ and $P(c_j|x)$, we can construct a probability curve for each class c_j:

$$B_j(x) = P(x|c_j)P(c_j) \quad (2)$$

When the curves for every class have been constructed, interval/bin borders are placed on each of those points where the leading curves are different on its two sides. Between each pair of those points including the two open ends, the learning curve is the same.

2.2 Motivation

From the description in Section 2.1, we known that the supervised methods consider the class attribute depends on the interaction between input features and class labels. It depends on the stationary assumption. Actually, the stationary assumption does not always hold in the real applications (Bai et al., 2014; Gama et al. 2014). For many learning tasks where data is collected over an extended period of time, its underlying distribution is likely to change. The drift in the underling distribution may result in a change in the learning problem.

If we can get the real labels in the test set, we should utilize these labels to supervise the feature reduction. But actually, we can't get the real labels. Consider that there is always a pool of classifiers such as Random Forest, Gradient Boosting, Maximum Entropy and Naïve Bayes. Each classifier has its own advantage. The ensemble learning (Dietterich, 2000; Wozniak et al., 2014) is such a technique focus on the combination of classifiers from heterogeneous or homogeneous modeling background to give the final decision. It is primarily used to improve the classification performance of a model, or reduce the likelihood of an unfortunate selection of a poor one. Dietterich (2000a) summarized the benefits:

(a) Allowing to filter out hypothesis that, though accurate, might be incorrect due to a small training set.

(b) Combining classifiers trained starting from different initial conditions could overcome the local optima problem.

(c) The true function may be impossible to be modeled by any single hypothesis, but combinations of hypotheses may expand the space of representable functions.

In this study, we adopt the ensemble learning method to the feature reduction. We employ ensemble classifiers to process the test set and get classification labels. We call the labels gotten from this procedure the pseudo labels since they are not the real labels in the test set. The pseudo labels are utilized to measure the difference/divergence between the training set and test set. The difference is been used to modify the feature space. More concretely, for each adjacent interval, the proposed method calculates the divergence between the labeled examples in training set and the pseudo labeled examples in test set and decide whether merge these intervals or not.

2.3 Method

To simplify, we take the two-class classification as example. The task of feature reduction is to put the feature values into several bins. The feature number will be reduced since the number of bins is generally less than the feature value number.

The typical unsupervised method such as the Equal Width method, do not make use of instance labels. The feature values are put into several equal sized bins. The supervised methods try to utilize the distribution of the classes in the training set to supervise the feature merge procedure. The equal-width method has the risk that merges values that are strongly associated with different classes into the same bin. The representative supervised method such as Bayesian avoids this problem by estimating the condition probability in the training set. The basic assumption is that the training set and test set has the same distribution, but it does not always holds. When distribution of training and test set are difference, the typical supervised method will fail.

The ensemble learning approach is adopted in this study, we get the pseudo labels of every instance in the test set by using other classifiers. Then, we use the *KL* divergence to measure the difference between training set and test set.

$$\mathrm{D}(P_{tr} \parallel P_{ts}) = \sum_y \sum_i P_{tr}(y|f_i) log \frac{P_{tr}(y|f_i)}{P_{ts}(y|f_i)} \quad (3)$$

Here the f_i denote the feature i, the $P_{tr}(y|fi)$ and $P_{ts}(y|fi)$ are the probability of the output label under the condition f_i in the training set and test set respectively, the $D(P_{tr} \parallel P_{ts})$ is the divergence between the training set and test set in the given interval.

Since the pseudo labels are the crucial to the feature reduction, how to select the candidate classifiers for getting the pseudo labels is also the key point. The intuition is that the mutually complementary classifiers which are characterized by high diversity and accuracy should be selected to get the pseudo labels for each other. Actually, the diversity has been recognized as a very important characteristic in classifier combination. Empirical results have illustrated that there exists positive correlation between accuracy of the ensemble and diversity among the base cassifiers (Dietterich,

2000b; Kuncheva and Whitaker, 2003; Tang *et al.*, 2006). Further, most of the existing ensemble learning algorithms (Brieman, 1996; Liu *et al*. 2000) can be interpreted as building diverse base classifiers implicitly. However, the problem of measuring classifier diversity and so using it effectively for building better classifier ensembles is still an open topic. Most researchers discuss the concept of diversity in terms of correct/incorrect outputs (Brown *et al*., 2005; Kuncheva and Whitaker, 2003; Tang *et al*., 2006). Kuncheva and Whitaker (2003) divide the diversity measures into pairwise diversity measures and non-pairwise diversity measures. For pairwise diversity measure, the *Q* statistics, the correlation coefficient, the disagreement measure and the double-fault measure are most commonly used. The previous experimental studies have shown that most diversity measures perform similarly (Kuncheva and Whitaker, 2003; Tang *et al*., 2006). In this study, we adopt the disagreement measure (Ho, 1998; Skalak, 1996) to select the classifiers for getting pseudo labels.

The disagreement measure of classifier *i* and *k* is defined as :

$$Dis_{i,k} = \frac{N^{01} + N^{10}}{N^{00} + N^{01} + N^{10} + N^{11}} \tag{4}$$

Where N^{00}, N^{01}, N^{10} and N^{11} are derived from the below table:

	D_k correct(1)	D_k wrong(0)
D_i correct(1)	N^{11}	N^{10}
D_i wrong(0)	N^{01}	N^{00}

Table 1: A 2*2 table of the relationship between a pair of classifiers

Support we have gotten the *L* classifiers which have high diversity with the target classifier for feature space reduction. The straightforward way is to use the classifier with highest diversity to get the pseudo labels. However, this method does not consider the accuracy of the classifier been selected. How about the result if the classifier with the highest diversity does not performance well? Actually, beside the diversity, the accuracy of the classifier and the classification confidence are also key factors for the pseudo labels getting. The accuracy of classifier can be explicitly expressed by the weight of classifier. The classification confidence , which was theoretically proved to be a key factor on the generalization performance (Shawe-Taylor and Cristianini, 1999), has been utilized in certain ensemble learning algorithms (Freund and Schapire, 1997; Li *et al*., 2014; Quinlan, 1996; Schapire and Singer, 1999).

In this study, we extract the pseudo labels by combining the ensemble margin (Schapire *et al*., 1998) and classification confidence (Li *et al*., 2014).

Let:

h_j ($j=1,2, \ldots, L$): the selected classifiers with high diversity.

$X=\{(x_i, y_i), i=1,2, \ldots, n\}$: the data set

y_i: the class label of the sample x_i

$\bar{y}_{ij}$: the classification decision of x_i estimated by the classifier h_j

c_{ij}: the classification confidence of x_i estimated by the classifier h_j

define the margin as:

$$\mathrm{m}(x_i) = \sum_{j=1}^{L} w_j \gamma_{ij} c_{ij}$$

$$\text{s.t. } w_j \geq 0, \quad \sum_{j=1}^{L} w_j = 1 \tag{5}$$

where the w_j is the weight of the classifier h_j and

$$\gamma_{ij} = \begin{cases} 1 & if\ y_i = \bar{y}_{ij} \\ -1 & if\ y_i \neq \bar{y}_{ij} \end{cases} \tag{6}$$

We can get the optimal $\mathrm{W} = [w_1, \ldots, w_L]_{L*1}^T$ by minizing the objective function below:

$$\mathrm{W} = \underset{W}{\operatorname{argmin}} \|U - \mathrm{T}W\|_2^2 + \lambda \|W\|_2 \tag{7}$$

Where $\mathrm{U} = [1, \ldots, 1]_{n*1}^T$, $\mathrm{T} = [\gamma_{ij} c_{ij}]_{n*L}$

$$\|U - \mathrm{T}W\|_2^2 = \sum_{i=1}^{n} (1 - m(x_i))^2 \tag{8}$$

λ is a Lagrange multiplier

The optimal W is utilized to get the finnal pseudo labels by combine the *L* classifiers with high diversity.

Once we got the pseudo labels, we will use these labels to supervise the feature reduction procedure. The distribution difference between the training set and test set can be measured.

The proposed feature reduction method searches the whole feature space by a fixed step. For each adjacent interval, the proposed method calculates the divergence between the labeled examples in training set and the pseudo labeled examples in test set and decides whether merge these intervals or not.

The adjacent intervals which have small change in the distribution will be merged. By elaborately selected moving step and the distribution distance threshold, the feature space will finnally partitioned into several sub-space which will reduce the original feature space.

The algorithm is shown below:

BEGIN

For each classifier *i*:

Select the *L* classifiers with high disagreement with the classifier *i* in the classifier pool

Optimize the weight *W* of the selected *L* classifiers

Make an ensemble model form the selected *L* classifiers and the optimal weight *W*

Get the pseudo labels in the test set using the ensemble model

For each feature f_i:

Set the interval merge step: *T*

For each adjacent *T*:

Get the Bayesian measure B_T using the formula (2)

Get *KL* Divergence D_p using the formula (3)

IF $B_T < \theta_b$ and $D_p < \theta_d$

Merge the adjacent intervals

ELSE

Go to next interval *T*

END

Here, the θ_b and θ_d are the threshold for Bayesian-measure and *KL* divergence respectively.

3 Experimental Results

The performance of the proposed method is evaluated on 20 UCI datasets (Frank and Asuncion, 2010). The detailed information of these datasets are shown in Table 2.

In the table 2, '#I' denotes the number of instances, '#F' denotes the feature number and '#C' denotes the classess number. These datasets cover some high-dimensional sets, some large sets, some small sets and some typical/balanced sets. More detailed information can be found on the UCI website.

The classifier pool includes Random Forest, Decision Tree, Gradient boosting, Maximum Entropy and Naïve Bayes. Every model uses the pseudo labels gotten from others to make the feature reduction.

A set of experiments are conducted in the multiple classifier system to show the performance of the proposed ensemble featrure reduction method. The conventional weighted majority voting approach is adopted as the fusion method for multiply classifier. Some analysis (Kuncheva, 2004; Wozniak and Jackowski, 2009) shown that it is an effective way for fusion of multiply classifier. The algorithm begins by creating a set of experts and assigning a weight to each. When a new instance arrives, the algorithm passes it to and receives a prediction from each expert. The algorithm predicts based on a weighted majority vote of the expert predictions.

The data sets considered are partitioned using the 10-fold cross-validation procedure. The 'Accuracy' is used as the performance measures. The 'Accuracy' is the number of successful hits relative to the total number of classification. It has been by far the most commonly used metric for assessing the performance of classifiers for years (Prati et al., 2011; Witten et al., 2011).

Dataset	#I	#F	#C
Abalone	4177	8	28
Audiology	226	69	23
Breast Cancer	286	9	2
Car Evaluation	1728	6	4
Census	199523	40	2
Ecoli	336	8	8
Internet Advertisements	3279	1558	2
Iris	150	4	3
Letter Recognition	20000	16	26
Magic Gamma Telescope	19020	11	2
Mammographic Mass	961	6	2
Molecular Biology	3190	61	3
Musk	476	168	2
Nursery	12960	8	5
Ozone Level Detection	2536	73	2
Page Blocks Classification	5473	10	5
Pima Indians Diabetes	768	8	2
Spectf Heart	267	44	2
Statlog (Vehicle Silhouettes)	946	18	4
Yeast	1484	8	10

Table 2. The datasets description

The experimental results on very data set are shown on Table 3. Here, the proposed ensemble method is compared with the typical unsupervised method EW (Equal Width) and the typical supervised method Bayes (Bayesian). The experimental results show that the proposed ensemble method outperform the conventional method (Equal Width and Bayesian) on almost all data set except the 'Iris' data set.

By analysis of the size of dataset, we found that the dataset size will impact the performance. Take the 'Iris' as example, there are only 150 instances in this dataset which lead to a small feature space (only 22 unique values for the first feature). There is little hint to make the feature reduction. It is very difficult to put them into several bins.

Dataset	EW	Bayes	Ensemble
Abalone	87.86	88.62	**89.58**
Audiolog	59.13	59.6	**60.06**
Breast Cancer	90.6	91.65	**92.21**
Car Evaluation	84.24	85.12	**86.19**
Census	84.04	84.39	**86.53**
Ecoli	77.5	78.35	**78.89**
Internet	64.09	64.64	**65.85**
Iris	**95.5**	94.25	94.25
Letter	87.59	88.21	**90.26**
Magic	86.72	87.82	**90.09**
Mammographic	67.6	68.05	**69.01**
Molecular	70.89	71.64	**72.83**
Musk	84.42	84.61	**85.35**
Nursery	83.59	84.29	**86.05**
Ozone	73.02	73.26	**74.95**
Page	83.72	84.16	**85.63**
Pima	69.07	69.52	**70.61**
Spectf Heart	80.57	80.72	**81.19**
Statlog	89.05	89.35	**90.61**
Yeast	60.99	61.84	**62.65**

Table 3. The experimental results

Since the feature space reduction is conducted on the feature space for each classifier. To further investigate the performance of the proposed feature reduction method, the compared experiments on each single classifier are also conducted to show the effect of the proposed method. Here, we take the Equal Width as the baseline method and the relative difference is taken as the evaluation measure.

The relative difference is calculated as:

$$\frac{Accuracy_{ref} - Accuracy_{baseline}}{Accuracy_{baseline}} \quad (9)$$

Here, the $Accuracy_{baseline}$ is the accuracy of EW on each dataset. The $Accuracy_{ref}$ is the accuracy of Bayesian and the proposed ensemble method.

Figure 1 ~ 6 show the experimental results on each individual classifier (Random Forest, Decision Tree, Gradient boosting, Maximum Entropy and Na ïve Bayes). Here, the baseline method is Equal Width. The blue line is the relative difference of Bayesian method comparing with the baseline. The red line is the relative difference of the Ensemble method. The x-axis shows the name of the selected datasets which are sorted by the size. The smallest dataset is 'Iris' which only has 150 instances while the largest dataset is the ' Census' dataset which has 199,523 instances.

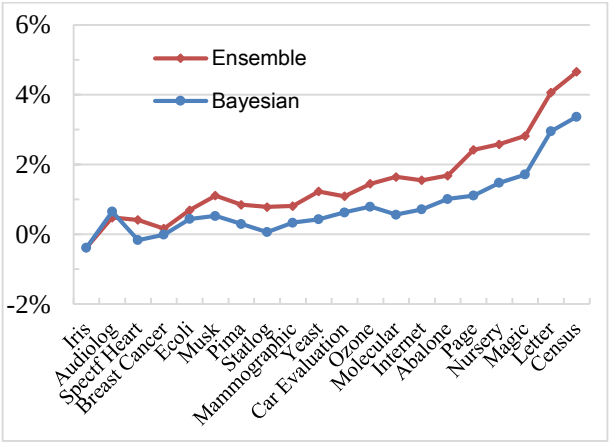

Figure 1. The experimental results on Random Forest

From Figure 1, we see that for Random Forest classifiers, the more data, the better performance. More than 4% enhancement has been achived on the 'Census' dataset which has 199,523 instances. In most dataset, the proposed ensemble method and Bayesian method are better than the unpervised method Equal Width. When the dataset is small, the performance is not so satisfied. For example, the ensemlbe method and Bayesian method worse than

the Equal Width method on the 'Iris' dataset. Also we can see that, when the dataset is small, the ensemble method can not beat the Bayesian method (' Audiolog ': 226 instances).

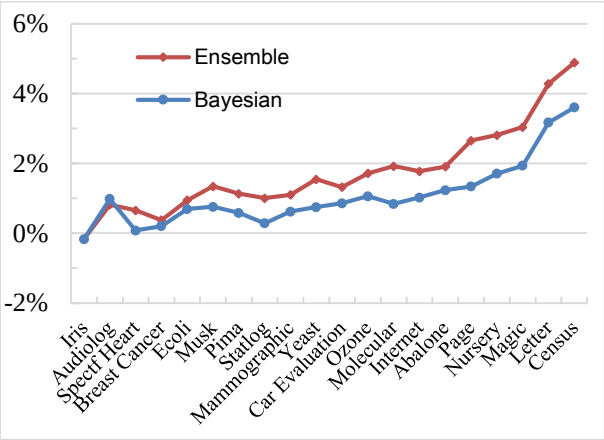

Figure 2. The experimental results on Decision Tree

Figure 2 shows the experimental results using Decision Tree classifier. We can see that the same trend as shown on the Random Forest. The highest enhancement is about 5% which is a little high than Random Forest. It is also gotten from the 'Census' dataset.

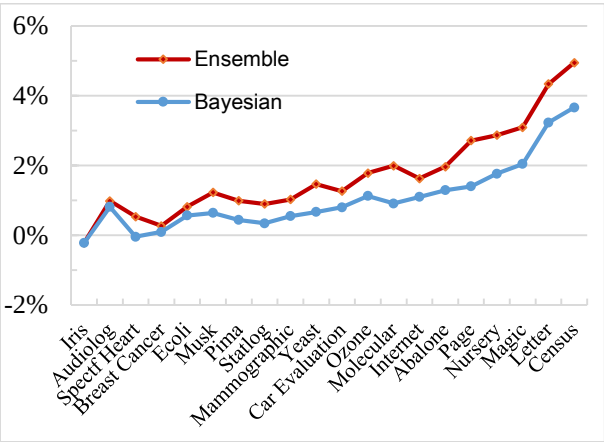

Figure 3. The experimental results on Gradient Boosting

Figure 3 shows the experimental results using Gradient Boosting classifiers. It's similar with the Rodom Forest and Decision Tree. For Gradient Boosting classifier, the ensemble method also does not performance well on the small datasets.

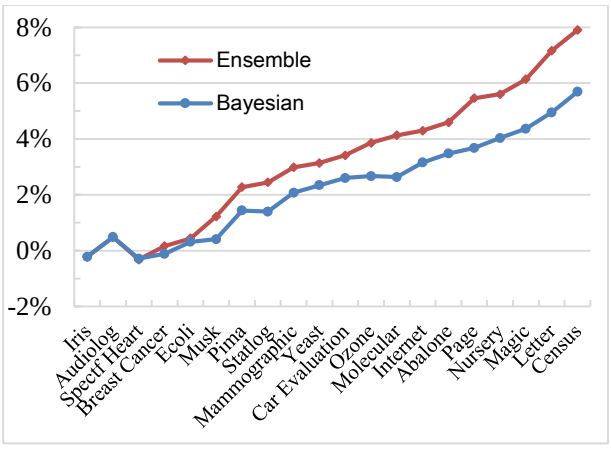

Figure 4. The experimental results on Maximum Entropy

Figure 4 shows the experimental results using Maximum Entropy classifier. The proposed ensemble method achived about 8% enhancement when the dataset is large ('Census': 199,523 instances). However, the performance also fluctuant when the dataset is small. It become stable when the dataset size is larger than 500. This may because the ensemble method need more data to measure the distribution divergence between training set and test set.

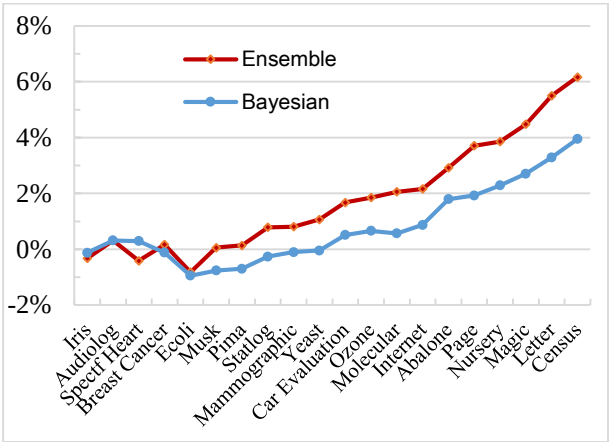

Figure 5. The experimental results on Na ïve Bayes

Figure 5 shows the experimental results using Na ïve Bayes classifier. The enhancement is also great (more than 6%). It is more fluctuanct than the Maximum Entroyp classifier when the dataset is small.

To further investigate the performance on different data size. A set of experiments on 'Census'

dataset are conducted. The sub-datasets range from 50 to 190,000 are extracted from the whole dataset. The experiments are intend to compare the performance of EW, Bayesian and the proposed ensemble method. The experimental results are shown as the relative difference with the baseline method (Equal Width method).

Figure 6 shows the experimental results. The x-axis shows the size of each sub-datasets. The y-axis shows the relative difference. The expeirments are conducted in the multiply classifier scenario, that is the finnal predciton is made by the ensemble classifier. We can see that the total enhancement is not higher than the Maximum Entropy or the Naïve Bayes classifier. This is because the fusion procedure highly depends on the diversity among the classifiers. It can't get the highest enhancement as the single classifier.

When the data size is small, both the proposed ensemble and Bayesian method cannot get good performance. For example, when the data size is less than 100, the ensemble and Bayesian methods are worse than EW. It is because that the Bayesian method needs to make statistic on the training set. The ensemble method need more data to calculate the distribution difference between training set and test set. From the Figure 6, we can see that, even there are about 1,000 samples, the ensemble method cannot get great enhancement in comparison with the Bayesian method. The ensemble method is worse than Bayesian method when the data size is small than 200. With the bigger dataset, the ensemble method performance better, about 4% enhancement can be achieved.

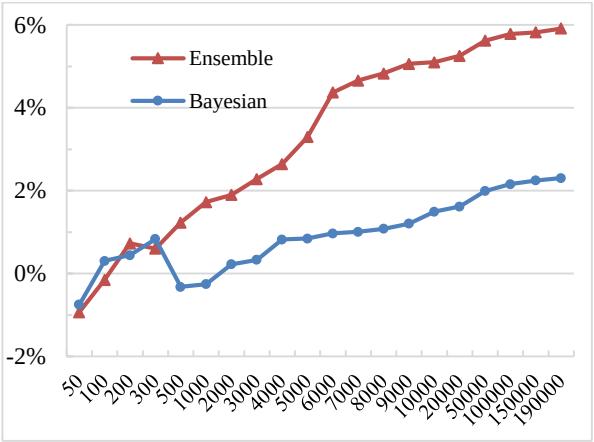

Figure 6. The experimental results on dataset size

4 Conclusions and Future Work

In this study, we propose a feature reduction method which uses ensemble approach to get the pseudo labels and utilize the pseudo labels to supervise the feature reduction procedure. The experiments conducted on different type of datasets compared the proposed method with the conventional feature reduction methods. The experimental results shown the effectiveness and efficiency of the proposed method.

The future work includes the scheme on selecting the candidate models for getting the pseudo labels. The measurement on distribution difference between training set and test set also need to be explored. How to improve the performance on small datasets is also research topic.

References

Bai Q. X., Lam H. and Sclaroff S. 2014. A Bayesian Framework for Online Classifier Ensemble. The 31st International Conference on Machine Learning, pages 1584-1592, Beijing, China, 2014.

Bay S. D. 2001. Multivariate Discretization for set Mining. Knowledge information Systems, Vol. 3, pp 491-512

Breiman L. 1996. Bagging predictors. Machine Learning, 24(2), 1996, 123-140

Brown G., Wyatt J., Harris R. and Yao X. 2005. Diversity creation methods: a survey and categorization. Journal of Information Fusion 6(1), 2005, 5–20.

Choudhary A., and Saraswat J. K. 2014. Survey on Hybrid Approach for Feature Selection. International Journal of Science and Research, 3(4), 438-439.

Combarro E. F., Montan E., D′ Iaz I., Ranilla J., and Mones R. 2005. Introducing a Family of Linear Measures for Feature Selection in Text Categorization. IEEE Transactions on Knowledge and Data Engineering, Vol. 17, No. 9, pp. 1223-1232

Dietterich T. 2000a. Ensemble methods in machine learning, in: Multiple Classifier Systems. Lecture Notes in Computer Science, vol. 1857, Springer, Berlin, Heidelberg, 2000, 1–15.

Dietterich T. 2000b. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting and randomization. Machine Learning, 40(1), 2000, 1-22.

Dougherty J., Kohavi R., and Sahami M. 1995. Supervised and unsupervised discretization of

continuous features. In Machine learning: proceedings of the twelfth international conference, Vol. 12, pp 194-202

Ferreira A. J. and Figueiredo M. A. T. 2012. An unsupervised approach to feature discretization and selection. Pattern Recognition 45(2012), pp. 3048–3060

Frank A. and Asuncion A. 2010. UCI machine learning repository, http://archive.ics.uci.edu/ml.

Freund Y. and Schapire R. E. 1997. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of computer and system sciences, 55(1), 1997, 119-139.

Gama J., zliobaite I., Bifet A., Pechenizkiy M. and Bouchachia A. 2014. A survey on concept drift adaptation. ACM Computing Surveys 46.4 (2014): 44.

Gao L. J. and Chien B. C. 2012. Feature Reduction for Text Categorization Using Cluster-Based Discriminant Coefficient. In Technologies and Applications of Artificial Intelligence (TAAI), 2012 Conference on (pp. 137-142). IEEE.

Garcia S., Luengo J., Sez J., Lpez V. and Herrera F. 2013. A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. IEEE Transactions on Knowledge and Data Engineering 25, pp. 734–750.

Ho T. 1998. The random space method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20:8, 1998, 832–844.

How B. C. and Kiong W. T. 2005. An examination of feature selection frameworks in text categorization. In AIRS'05: Proceedings of 2nd Asia information retrieval symposium, PP 558–564.

Kerber R. 1992. ChiMerge: Discretization of Numeric Attributes. Proc. Nat'l Conf. Artifical Intelligence Am. Assoc. for Artificial intelligence, pp 123-128

Kuncheva L. 2004. Combining Pattern Classifiers: Method and Algorithms, Wiley Interscience, 2004

Kuncheva L. and Whitaker C. 2003. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. Machine Learning, 51:181-207, 2003.

Jiang F., Zhao Z., and Ge Y. 2010. A Supervised and Multivariate Discretization Algorithm for Rough Sets. Proc. Fifth Int'l Conf. Rough Set and Knowledge Technology (RSKT), pp. 596-603

Li L., Hu Q., Wu X. and Yu D. 2014. Exploration of classification confidence in ensemble learning. Pattern Recognition, 47(9), 2014, 3120-3131.

Li R. P. and Wang Z. O. 2002. An Entropy-based Discretization Method for Classification Rules with Inconsistency Checking. Proc. First Int'l Conf. Machine Learning and Cybernetics. pp 243-246

Liu H. and Setiono R. 1997. Feature selection via discretization. IEEE transactions on knowledge and data engineering, Vol 9, No. 4, 642-645

Liu Y. Yao X. and Higuchi T. 2000. Evolutionary ensembles with negative correlation learning. IEEE Transactions on Evolutionary Computation, 4, 2000, 380-387

Jin R., Breitbart Y. and Muoh C. 2009. Data Discretization Unification. Knowledge and Information Systems, Vol. 19, pp 1-29

Ng H. T., Goh W. B., and Low K. L. 1997. Feature selection, perceptron learning, and a usability case study for text categorization. In SIGIR '97: Proceedings of the 20th annual international ACM SIGIR conference on Research and development in information retrieval, PP 67–73

Prati R.C., Batista G.E.A.P.A., and Monard M.C. 2011. A Survey on Graphical Methods for Classification Predictive Performance Evaluation, IEEE Trans. Knowledge and Data Eng., Vol. 23, No. 11, pp. 1601-1618, Nov. 2011, doi: 10.1109/TKDE.2011.59.

Quinlan J. R. 1996. Bagging, boosting, and C4. 5. In AAAI/IAAI, Vol. 1, 1996, 725-730

Reif M. and Shafait F. 2014. Efficient feature size reduction via predictive forward selection, Pattern Recognition(2014), Vol. 47, PP 1664-1673

Robati, Z., Zahedi, M., and Fayazi Far, N. 2015. Feature Selection and Reduction for Persian Text Classification. International Journal of Computer Applications, 109(17), 1-5.

Schapire R. E., Freund Y., Bartlett P. and Lee W. S. 1998. Boosting the margin: A new explanation for the effectiveness of voting methods. Annals of statistics, 1998, 1651-1686.

Schapire R. E. and Singer Y. 1999. Improved boosting algorithms using confidence-rated predictions. Machine learning, 37(3), 1999, 297-336.

Shawe-Taylor J. and Cristianini N. 1999. Robust bounds on generalization from the margin distribution. The 4th European Conference on Computational Learning Theory, 1999.

Singh G. K. and Minz S. 2007. Discretization Using Clustering and Rough Set Theory. Proc. 17th int'l Conf. Computer Theory and Applications, pp 330-336

Skalak D. 1996. The sources of increased accuracy for two proposed boosting algorithms. In Proc. American Association for Artificial Intelligence, AAAI-96, Integrating Multiple Learned Models Workshop

Tang E. K., Suganthan P. N. and Yao X. 2006. An analysis of diversity measures. Mach. Learn. 65(2006)247–271.

Witten I.H., Frank E., and Hall M.A.2011. Data Mining: Practical Machine Learning Tools and Techniques, third ed. Morgan Kaufmann, 2011.

Wozniak M., Grana M. and Corchado E. 2014. A survey of multiple classifier systems as hybrid systems. Information Fusion, 16:3-17, 2014.

Wozniak M. and Jackowski K. 2009. Some remarks on chosen methods of classifier fusion based on weighted voting, in: E. Corchado, X. Wu, E. Oja, A. Herrero, B. Baruque (Eds.), Hybrid Artificial Intelligence Systems, Lecture Notes in Computer Science, vol. 5572, Springer, Berlin/Heidelberg, 2009, pp. 541–548

Wu X. 1996. A Bayesian discretizer for real-valued attributes. The Computer Journal, 39(8), 688-691.

Yang Y. and Webb G. I. 2009. Discretization for Naive-Bayes Learning: Managing Discratization bias and Variance. Machine Learning. vol. 74, No. 1, pp 39-74

Zighed D. A., Rabaseda S. and Rakotomalala R. 1998. FUSINTER: A method for discretization of continuous Attributes. Int'l J. Uncertainty, Fuzziness Knowledge-based Systems, Vol. 6, pp 307-326