International Journal of Multimedia and Ubiquitous Engineering
Vol. 10, No. 3 (2015), pp. 231-244
http://dx.doi.org/10.14257/ijmue.2015.10.3.22

Meta-Ensemble Classification Modeling for Concept Drift

Joung Woo Ryu[1] and Jin-Hee Song[2]

[1]Technical Research Center, Safetia Ltd. Co., South Korea
[2]School of IT Convergence Engineering, Shinhan University, South Korea
ryu0914@gmail.com, jhsong@shinhan.ac.kr

Abstract

We propose ensemble-based modeling for classifying streaming data with concept drift. The concept drift is a phenomenon in which the distribution of streaming data changes. In this paper, the types of the concept drift are categorized into the change of data distribution and the change of class distribution. The proposed ensemble modeling generates a meta-ensemble which consists of ensembles of classifiers. Whenever a change of class distribution occurs in streaming data, our modeling builds a new classifier of an existing ensemble and whenever a change of data distribution occurs, it builds a new ensemble which consists of an only one classifier. In our approach, new classifiers of a meta-ensemble on streaming data will be generated dynamically according to the estimated distribution of streaming data. We compared the results of our approach and of the chunk-based ensemble approach, which builds new classifiers of an ensemble periodically. In experiments with 13 benchmark data sets, our approach produced an average of 21.95% higher classification accuracy generating an average of 61.7% fewer new classifiers of an ensemble than the chunk-based ensemble method using partially labeled samples. We also examine that the time points when our approach builds new classifiers are appropriate for maintaining performance of an ensemble.

Keywords: *modeling, classification, streaming data, ensemble, concept drift*

1. Introduction

Many companies which deal with customer's preferences or life styles provide reliable services using a prediction/classification model management method. Those models have to predict/classify something from streaming data in real time [1]. A major characteristic of streaming data is to be changed in data distribution according to data generation status. Usually prediction/classification models are periodically updated, and those methods have the following disadvantages [2]. First, if the accuracy of a classifier is high, the classifier does not need to be updated. However, even in this case, the classifier is updated because of the update period. Second, if the current time doesn't reaches a period setting value, the classifier would not be updated even though the distribution of streaming data is changed within the update period. Third, human experts should label all samples, and then evaluate or refine the current classifier using them. In a real-world application, it is very impractical process that a human expert gives the classifier feedback on its decision for every single sample.

The conventional ensemble approach works on the assumption that all streaming data have correct labels. It usually adds new classifiers to an ensemble based on regular time intervals or fixed number of streaming samples called chunk. To deal with concept drift the conventional ensemble approach assigns a new weight to each classifier of an ensemble [2, 3] or builds new classifiers for an ensemble using cross-validation [4, 5] whenever a new chunk is coming. Chu *et al.*, [6] and Zhang *et al.*, [3] used the weighted samples when a new classifier for an ensemble are built from a chunk. For tracking a recurrent concept drift, Katakis *et al.*, [7] transforms a chunk as a "conceptual vector"

ISSN: 1975-0080 IJMUE

when correct classes for each sample within the chunk are available. Zhang *et al.*, [8] proposed an aggregate ensemble framework where different classifiers are built by each different learning algorithm from a chunk including noisy data. Wei *et al.*, [9] proposed an ensemble approach for multi-label classification problems in streaming data where each sample can be classified into more than one category (*e.g.*, multiple illnesses and interesting topics at the same time). However, it is impractical in the real-world applications that human experts product correct labels on all streaming data. Recently, some researchers have recognized that it is not reasonable in practice to manually label all samples within a chunk for building a new classifier of an ensemble [10, 11].

We propose an efficient ensemble-based modeling approach for classifying data streams with concept drift. Our approach is able to dynamically generate new classifiers for an ensemble on streaming data. It decides if streaming samples should be selected for building new classifiers not according to a time interval, but according to a change in the estimated distribution of streaming data. In addition, our ensemble approach can handle concept drift in an online process.

This paper is organized as follows. Section 2 introduces the categorization of concept drift to be used in this paper. The proposed meta-ensemble modeling is described in Section 3. We report experimental results in Section 4 where our approach is compared with traditional methodologies using real data sets, while conclusions and future works are presented in Section 5.

2. Changes in Distribution of Streaming Data

A data stream is a continuous and infinite sequence of data, which makes either storing or scanning all the historical data nearly impossible [12]. Moreover, streaming data often evolve considerably over time. The change in streaming data distribution is referred to as concept drift. Types of the concept drift are categorized into (a) *the change of data distribution* or (b) *the change of class distribution* according to change in streaming data distribution. The numbers in Figure 1 represent sequence of streaming samples. The four samples belonging to a class distribution different from the previous one occurred after the 11st sample. Figure 1 also shows that the current data distribution is changing from the 15th sample.

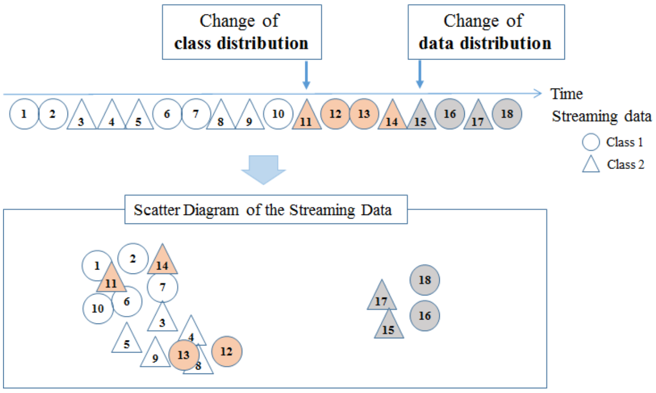


Figure 1. Categorization of Concept Drifts According to Change in Streaming Data Distribution

3. Ensemble-based Modeling for Data Streams with Concept Drift

We propose a more flexible approach which does not build a new classifier periodically in a fixed interval of time. Our ensemble approach decides dynamically when to build a new classifier and which samples should be used as a training data set according to changes in the distribution of streaming data.

Our ensemble modeling method adopts the concept of training data areas in order to estimate the current data distribution. A training data area is defined the mean vector (center) and the standard deviation of training data from which a classifier is built. If the coming streaming data do not belong to any of training data areas and the streaming data are near to each other, we can assume that a change in the current data distribution occurs. Using such an assumption, a methodology of maintaining the performance of an ensemble classifier in streaming data was proposed and its validity was showed by experiments with ten benchmark data sets [13]. However, the previous our methodology proposed by [13] suffers from changes of class distribution occurring within training data areas. We complement the previous our methodology to overcome its shortcoming.

3.1. Building New Classifiers of a Meta-ensemble

By the method proposed in [13] a meta-ensemble builds a new classifier whenever a change of the estimated data distribution occurs. The meta-ensemble also estimates the current class distribution in order to deal with changes of class distribution in streaming data. In other words, the meta-ensemble also builds a new classifier whenever a change of the estimated class distribution occurs.

The class distribution of training data is represented by a classifier which is generated from them. If new streaming data do not belong to the class distribution of training data, a classifier generated from the training data will incorrectly predict their class labels. Therefore, the meta-ensemble modeling compares the accuracy of a classifier on training data with its accuracy on test data. If the difference between the two accuracies is statistically significant, we believe that a change of class distribution occurs. The test data set consists of data which randomly selected from new streaming data belonging to the same training data area.

Figure 2 shows the meta-ensemble modeling process. In Figure 2, σ_i denotes the radius of the i-th training data area (the standard deviation of the training data). α is the probability of selecting a sample as a test datum from streaming samples which belong to the same training data area. If a random value is less than α, the input streaming sample generated at the time is selected as a test datum. θ_k denote the radius of the k-th neighbor area where the user-defined labels of all suspicious samples are required to build a new classifier. Streaming data which do not belong to any of training data areas are called suspicious samples. The suspicious samples belonging to a neighbor area are close each other. The meta-ensemble modeling uses the minimum number (θ_m) of neighboring suspicious samples as a trigger for finding out whether the change of class distribution inside the training data area occurs or not. In our experiments, θ_k was defined as $2\times\sigma_0$(the standard deviation of the initial training data) and θ_m the number of initial training data.

Our approach is divided into an off-line process and an on-line process. In the off-line process, we label the selected streaming data manually and build new classifiers of a meta-ensemble from the labeled data. In the on-line process, a meta-ensemble selects useful streaming samples for maintaining its performance from streaming data. The suspicious samples are useful samples for dealing with changes of data distribution. Useful samples for dealing with changes of class distribution are samples belonging to the same training data area when the difference of the two accuracies on training data and on the samples is statistically significant. The statistical hypothesis testing is applied in order to decide whether the comparison result is statistically significant or not. We use the Fisher's Exact Test or the chi-square test as the method of statistical hypothesis testing. The Fisher's Exact Test is used when the size of a training data set is small, the chi-square test when the size of a training data set is large. In our experiments, the Fisher's Exact Test was used because the size of a training data set was defined as 300.

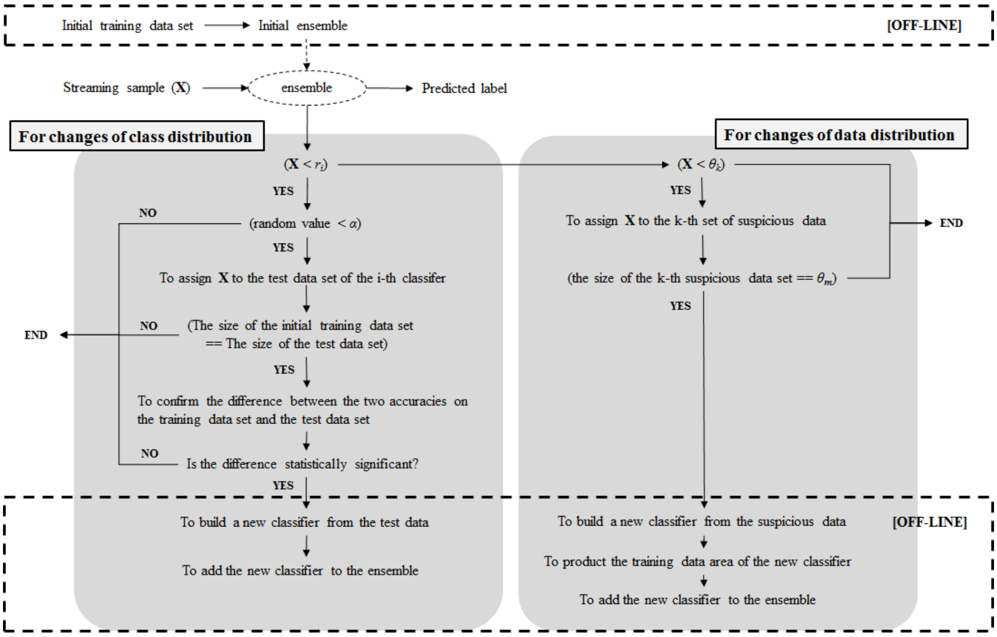

Initial training data set
Initial ensemble
[OFF-LINE]
Streaming sample (X)
ensemble
Predicted label
For changes of class distribution
For changes of data distribution
$(X < r_i)$
YES
NO
(random value $< \alpha$)
YES
To assign X to the test data set of the i-th classifer
END
NO
(The size of the initial training data set == The size of the test data set)
YES
To confirm the difference between the two accuracies on the training data set and the test data set
NO
Is the difference statistically significant?
YES
$(X < \theta_k)$
YES
To assign X to the k-th set of suspicious data
END
(the size of the k-th suspicious data set $== \theta_m$)
YES
To build a new classifier from the test data
To add the new classifier to the ensemble
To build a new classifier from the suspicious data
[OFF-LINE]
To product the training data area of the new classifier
To add the new classifier to the ensemble

Figure 2. Meta-ensemble Modeling Process

Figure 3 shows the structure of a meta-ensemble. The meta-ensemble builds an ensemble with an only one classifier whenever the change of the estimated data distribution occurs, and a new classifier of the ensemble corresponding to a training data area whenever the change of the estimated class distribution occurs within the training data area. The change of streaming data distribution in Figure 3 has happened four times so far. The change of data distribution has happened one time and the change of class distribution three times.

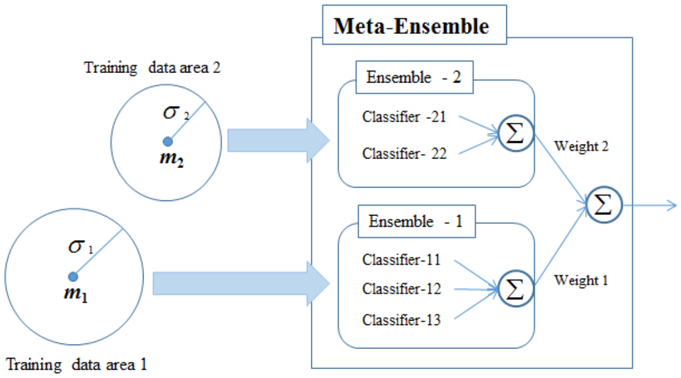


Figure 3. Structure of a Meta-ensemble Generated by the Proposed Approach

3.2. Classifying Streaming Data in a Meta-ensemble

When the meta-ensemble classifies a coming streaming datum, if the datum belongs to a training data area as shown in (a) of Figure 4, it is classified by the only ensemble corresponding to the training data area. The final output value of an ensemble is decided by the simple majority voting method. If the datum does not belong to any of training data areas as shown in (b) of Figure 4, the meta-ensemble classifies the datum using all ensembles. The final output value of the meta-ensemble is decided by the weighted majority voting method as shown in equation (1).

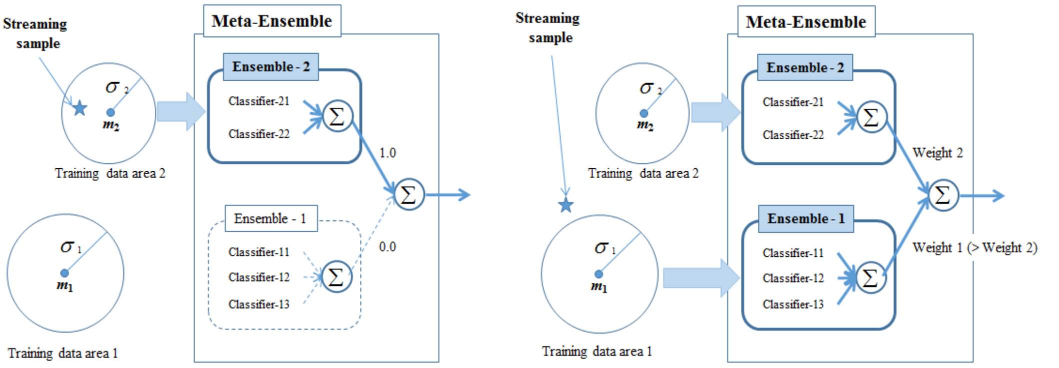
$$y_t = argmax_{y_k \in CLASS}\left(\sum_{j=1}^{n} w_{tj} P_j(y_k|\boldsymbol{S}_t)\right) \qquad (1)$$

- $P_j(y_k|\boldsymbol{S}_t)$ is the probability of the class $\boldsymbol{y}_k$ predicted by the j-th ensemble of a meta- ensemble.
- w_{tj} is a weight value of the *j*-th ensemble on the new samples $\boldsymbol{S}_t$.
- *n* is the total number of ensembles in a meta-ensemble, and
- *CLASS* denotes the set of classes.

The weight values of each ensemble in a meta-ensemble are calculated by the membership function used in Fuzzy C-means as shown in equation (2).

$$w_{tj} = \frac{1}{\sum_{k=1}^{n}\left(\left(\frac{dist(\boldsymbol{S}_t,\boldsymbol{M}_j)}{\sigma_j}\right)/\left(\frac{dist(\boldsymbol{S}_t,\boldsymbol{M}_k)}{\sigma_k}\right)\right)^2} \qquad (2)$$

- w_{tj} is a weight value of the *j*-th ensemble of a meta-ensemble on a new sample $\boldsymbol{S}_t$. $\boldsymbol{M}_j$ and σ_j denote the mean vector and the standard deviation of the training data area representing *j*-th ensemble of a meta-ensemble.
- *n* is the total number of ensembles of a meta-ensemble.
- $dist(\boldsymbol{S}_t, \boldsymbol{M}_j)$ denotes the distance between a new sample vector $\boldsymbol{S}_t$ and the mean vector $\boldsymbol{M}_j$ of the training data area of the *j*-th ensemble.



(a) The case that a streaming sample belongs to a training data area

(b) The case that a streaming sample DOES NOT belongs to any of training data areas

Figure 4. Mechanism of the Meta-ensemble for Classifying Streaming Data

4. Experiments

We evaluated the proposed ensemble approach using real data sets from the UCI data repository, three streaming data sets from Wikipedia under the keyword "concept drift", and the click data set which is generated by the click fraud detection system, NetMosaics, as shown in Table 1. These sets have various numbers of classes and various types of attributes. In particular, some of the data sets have the class distribution which is not uniform among the classes.

The original "Electricity market" data set has samples with missing values for some numerical attributes. We removed those samples from the original data sets. The "Mushroom" and "Adult" data sets also have missing values for some categorical attributes. We replaced these missing values in each sample with the new, categorical "NULL" value.

We divided each data set into an initial training data set and a streaming data set. The initial training data set was used for building an initial classifier of an ensemble, and the streaming data set was used as a test data set for evaluating ensemble approaches. The first 300 samples were used as the initial training data. The remaining samples were used as a streaming data.

Table 1. Real Data Sets Used in Experiments

Data	#data	#attributes	#classes	Rate of each class (%)
Landsat Satellite	6,435	37	6	C1(23.8), C2(10.9), C3(21.1), C4(9.7), C5(11.0), C6(23.4)
Mushroom	8,124	23	2	C1(51.8), C2(48.2)
Nursery	12,960	10	5	C1(33.3), C2(0.015), C3(2.5),C4(32.9),C5(31.2)
MAGIC	19,020	11	2	C1(64.8), C2(35.2)
Click Data	24,537	14	2	C1(63.6),C2(36.4)
EM	27,549	7	2	C1(41.5), C2(58.5)
Adult	48,842	15	2	C1(23.9), C2(76.1)
PAKDD2009	50,000	24	2	C1(80.3), C2(19.7)
Shuttle	57,800	9	7	C1(78.60), C2(0.09), C3(0.29), C4(15.35), C5(5.63), C6(0.02), C7(0.02)
MiniBoone	129,596	51	2	C1(28.2), C2(71.8)
Census	299,285	37	2	C1(93.8), C2(6.2)
KDDCup1999	494,021	42	2	C1(19.7), C2(80.3)
Covertype	581,012	55	7	C1(36.5), C2(48.8), C3(6.2), C4(0.5), C5(1.6), C6(3.0), C7(3.5)

We used the following three performance measures for evaluation of approaches of modeling for classifying data streams

The total number of new classifiers (TC) is a count of generated new classifiers for an ensemble over streaming data (after manually labeling samples in an offline process). Suppose that two ensemble methods use the same total number of labeled samples for building each classifiers over a stream data, and their classification accuracies on the data stream are the same. If one ensemble method builds more new classifiers than the other, that method requires more interactions with a human expert because of more intensive labeling process. Accordingly, an ensemble method with less new classifiers is more efficient methodology for real world problems where systematic human labeling is not feasible.

The labeled sample rate (LR) is the proportion of the labeled samples used for building new classifiers for an ensemble in a data stream. Suppose that two ensemble methods built the same number of new classifiers from a data stream and they produced the same classification accuracy on the data stream. If one ensemble method uses fewer labeled samples than the other, that ensemble method is more efficient because human labeling is less required.

The weighted sum of F-measures for all classes (WSF) is an appropriate measure for an ensemble accuracy applied for streaming data. The ordinary classification accuracy, defined as the rate of correctly classified samples, is inadequate in our experiments because most of the real data sets have large differences among the numbers of samples belonging to each class. Such a skewed class distribution means that the samples in the majority class dominate the results when using the ordinary classification accuracy measure. For example, if an ensemble with an initial classifier predicts the classes of all streaming data as class 1(C1) in the “Census” data set, then the classification accuracy of the ensemble becomes 93.8%. Classifiers dealing with the skewed class distribution have been evaluated using a cost matrix or a confusion matrix. In the cost matrix, a cost is the penalty for incorrectly classifying a sample. Each cost of classes is predefined by a domain expert according to the significance of a class in the domain. If such costs are unavailable, performance measures such as precision, recall, etc. derived from the confusion matrix can be used. However, those measures have been used to evaluate the classifiers for a binary classification problem with the skewed class distribution. Since some of our real data sets have more than two classes, we define the weighted sum of f-measure $F(c_i)$ for all classes as follows:

$$WSF = \sum_{c_i \in CLASS} w_i F(c_i) \tag{3}$$

$$F(c_i) = \frac{2 \times Precision(c_i) \times Recall(c_i)}{Precesion(c_i) + Recall(c_i)} \tag{4}$$

where *CLASS* denotes the set of classes, and $F(c_i)$ is the harmonic mean of the precision and the recall of c_i class. Each weight w_i of classes is attached according to the proportion of the corresponding class c_i in a streaming data set. If the proportion of a class is large, a small w_i value is assigned to the class's weight maintaining the balance of influence on the final WSF parameter. Conversely, if the proportion is small, the class's weight becomes a large value.

Formally, we determined the weight w_i of a class c_i using equation (5), where N is the number of streaming data, and n_i is the number of samples belonging to c_i class in the streaming data.

A weight value according to the proportion of a class was divided by |*CLASS*|-1 so that the sum of weights of the classes becomes normalized to 1.

$$w_i = \frac{1}{|CLASS| - 1} \times \left(1 - \frac{n_i}{N}\right) \tag{5}$$

4.1. Comparison with the Chunk-based Ensemble Approaches using Partially Labeled Sample

We implemented simple voting ensemble (SVE-P) and weighted ensemble (WE-P) methods as the chunk-based ensemble approach to show that our approach efficiently maintains performance of an ensemble. Both the SVE-P and WE-P methods periodically build new classifiers for an ensemble using samples which are randomly selected within each chunk. The SVE-P method combined results of classifiers in an ensemble by the majority voting method. WE-P used the weighted majority voting method as the combining method for classification.

Each classifier weight in the SVE-P method was predefined as equal to 1.0. In the weighted ensemble method WE-P, each classifier weight was determined using the most recent chunk according to the method presented by Wang et al. [2]. If a classifier in an ensemble provides the highest accuracy on the samples of the most recent chunk, the largest value will be attached to the classifier weight. Classifiers' weights are maintained until the next new classifier for the ensemble is built.

Table 2. Comparison with TC and LR of SVE-P and WE-P

Data	Meta-ensemble (θ_m =300, α =0.04)		SVE-P (chunk size = 3000)		WE-P (chunk size = 3000)	
	TC	RL(%)	TC	RL(%)	TC	RL(%)
Landsat Satellite	5	19.5	3	9.7	3	9.7
Mushroom	11	38.3	3	7.6	3	7.6
Nursery	5	9.4	5	9.4	5	9.4
MAGIC	7	9.6	7	9.6	7	9.6
Click Data	8	8.6	9	9.9	9	9.9
EM	9	8.8	10	9.9	10	9.9
Adult	8	6.1	17	9.8	17	9.8
PAKDD2009	13	8.4	17	9.7	17	9.7
Shuttle	2	3.6	20	9.8	20	9.8
MiniBoone	21	4.8	44	9.9	44	9.9
Census	16	4.6	100	9.9	100	9.9
KDDCup1999	28	5.1	165	10.0	165	10.0
Covertype	94	4.8	194	10.0	194	10.0
Average	**17.46**	**10.12**	**45.69**	**9.63**	**45.69**	**9.63**

Table 3. Comparison with WSF of SVE-P and WE-P

Data	Meta-ensemble (θ_m =300, α =0.04)	SVE-P (chunk size = 3000)	WE-P (chunk size = 3000)
Landsat Satellite	0.591	0.440±0.005	0.516±0.040
Mushroom	0.898	0.532±0.128	0.816±0.095
Nursery	0.515	0.441±0.008	0.473±0.020
MAGIC	0.742	0.712±0.004	0.722±0.007
Click Data	0.688	0.643±0.007	0.675±0.009
EM	0.670	0.641±0.010	0.608±0.015
Adult	0.532	0.599±0.036	0.527±0.031
PAKDD2009	0.275	0.181±0.005	0.367±0.017
Shuttle	0.129	0.031±0.000	0.161±0.050
MiniBoone	0.817	0.822±0.003	0.589±0.018
Census	0.215	0.061±0.004	0.162±0.054
KDDCup1999	0.982	0.666±0.009	0.547±0.033
Covertype	0.386	0.182±0.014	0.115±0.018

We defined the chunk size as 3,000 samples and used 10% labeled samples that were randomly selected in each chunk. Table 2 shows the total number of new classifiers generated by each method, and the rate of labeled samples used by each method. Table 3 shows WSF values of Meta-ensemble, SVE-P and WE-P. A set of ten experiments was performed for both SVE-P and WE-P with different random seeds for selecting 300 samples to be labeled in each chunk. The meta-ensemble generated 61.7% fewer new classifiers than the chunk-based ensemble approach using partially labeled samples, and used an average of 10% labeled samples for the 13 data sets. The meta-ensemble produced an average of 0.572. This average is 25.2% higher than the average WSF of SVE-P and 18.7% higher than the average WSF of WE-P. Each of these differences is statistically significant (Wilcoxon's test, significant level=0.05).

4.2. Comparison with the Initial Classifier

We look into the accuracy of the initial classifier which was built from the initial training data. The initial classifier predicted the labels of streaming data without the process of relearning it. We can evaluate efficiency of the proposed approach through comparing it's accuracy with the accuracy of the initial classifier. Table 4 shows the classification accuracies of the initial classifier and of the meta-ensemble generated by our approach. The numbers in Table 4 represent values of WSF. The meta-ensemble produced an average of 41.6% higher WSF measure value than the initial classifier. In particular, its WSF value is bigger than one of the initial classifier for all data sets.

Table 4. Comparison with WSF of the One Single Classifier

Data	Meta-ensemble (θ_m=300, α=0.04)	Initial classifier
Landsat Satellite	0.591	0.476
Mushroom	0.898	0.393
Nursery	0.515	0.445
MAGIC	0.742	0.725
Click Data	0.688	0.669
EM	0.670	0.633
Adult	0.532	0.508
PAKDD2009	0.275	0.175
Shuttle	0.129	0.031
MiniBoone	0.817	0.758
Census	0.215	0.060
KDDCup1999	0.982	0.263
Covertype	0.386	0.126
Average	**0.572**	**0.404**

To verify that the meta-ensemble is built new classifiers at reasonable time points, we included some specific analyses and interpretations of the classification in a time domain. Figure 4 and Figure 5 show variations in prequencial errors of the meta-ensemble and the initial classifier. As in work by Gama et al.[14], the prequencial error is calculated by a forgetting mechanism using fading factors(= 0.975). The vertical lines in Figure 4 and Figure 5 shows points in time where a new classifier for a meta-ensemble is built.

In Figure 4, the error rate of the meta-ensemble is lower than the initial classifier after a new classifier is added to the meta-ensemble at the time when the 1125th sample is generated. The error rate of the meta-ensemble in Figure 6 decreased after the 3602nd sample and the 3738th sample, whereas one of the initial classifier increased. This analysis in time shows how real time adjustments in the ensemble influence the quality of classification results for streaming data.

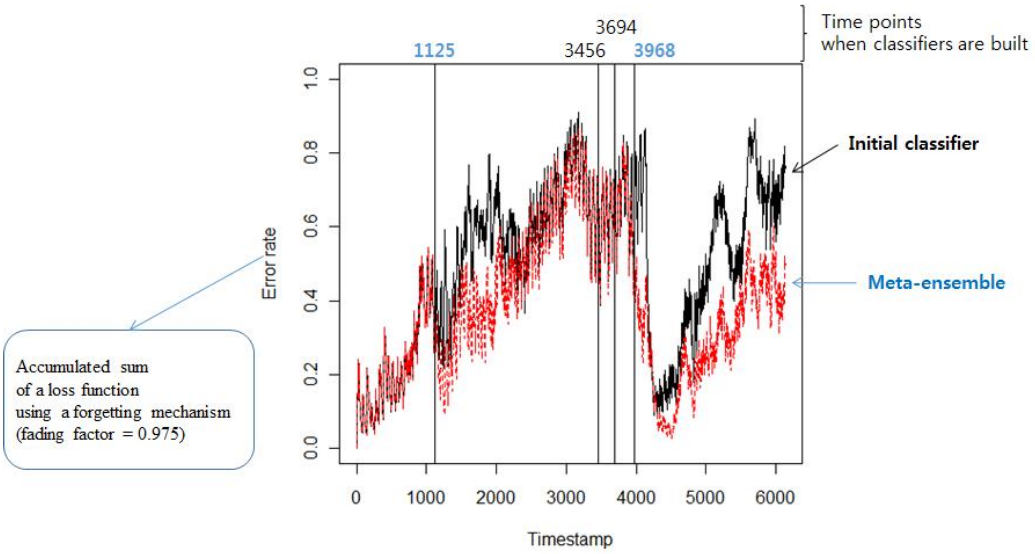


Figure 5. Variations in Errors, and Time Points when Classifiers are Built in the Landsat Satellite Dataset

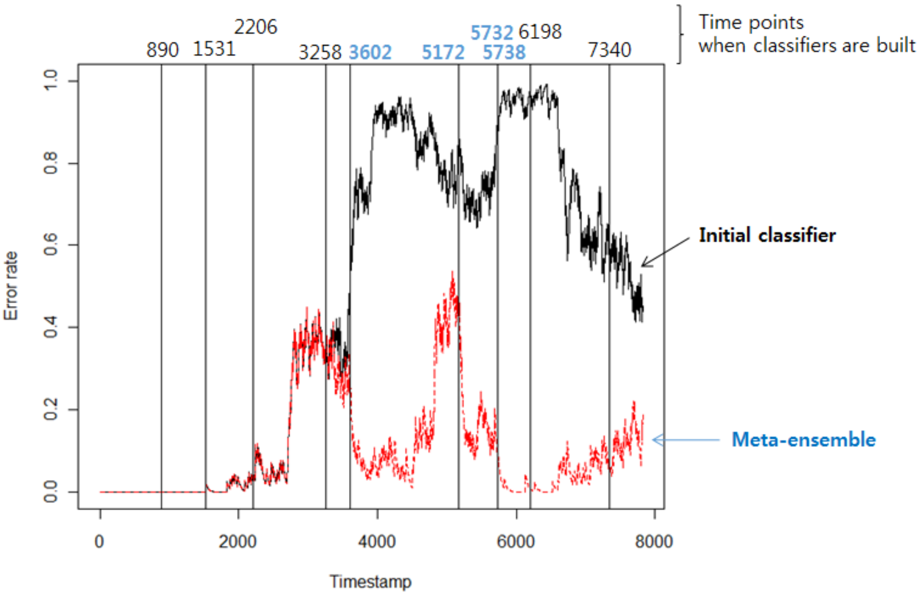


Figure 6. Variations in Errors, and Time Points when Classifiers are Built in the Mushroom Dataset

4.3. Comparison among Proposed Approaches using Different Classification Algorithms

We used decision trees as classifiers of an ensemble for the experiments so far. The decision tree was generated with J48 decision tree (C4.5 algorithm) from Weka (http://www.cs.waikato.ac.nz/ml/weka/). However, our ensemble approach does not depend on a specific classification algorithm for building a classifier of an ensemble. To prove that, we carried out experiments on 13 real data sets with four other classification algorithms: SVM(Support Vector Machine), MLP(Multilayer Perceptron), NB(Naïve Bayesian), and LR(Logistic Regression). The four kinds of classifier were also built with each algorithm provided by Weka.

Table 5 shows WSF values produced when each algorithm was used. To use the Friedman Test as in Demšar [15] the algorithms achieved their ranks according to WSF values for each data set separately. Numbers in parentheses in Table 5 denote ranks of algorithms. In this test, the null-hypothesis is that all the classifiers perform the same and the observed differences are merely random. With five algorithms and 13 data sets, F_F=2.11 is distributed according to the F distribution with 5-1=4 and (5-1)×(13-1)=48 degree of freedom. The critical value of F(4,48) for α=0.05 is 2.57, so we accept the null-hypothesis.

Table 5. Comparison of WSF for Meta-ensembles with Each of Five Classification Algorithms; Numbers in the Parentheses Denote Ranks of the Algorithms

Data	Meta-ensemble				
	with DT	with SVM	with NN	with NB	with LR
Landsat Satellite	0.591 (4)	0.608 (3)	0.612 (2)	0.628 (1)	0.544 (5)
Mushroom	0.898 (1)	0.843 (4)	0.842 (5)	0.858 (3)	0.870 (2)
Nursery	0.515 (4)	0.595 (2)	0.615 (1)	0.480 (5)	0.561 (3)
MAGIC	0.742 (2)	0.718 (4)	0.778 (1)	0.635 (5)	0.720 (3)
Click Data	0.688 (4)	0.717 (2)	0.752 (1)	0.644 (5)	0.705 (3)
EM	0.670 (2)	0.650 (5)	0.669 (3)	0.662 (4)	0.674 (1)
Adult	0.532 (5)	0.617 (4)	0.635 (2)	0.627 (3)	0.638 (1)
PAKDD2009	0.275 (3)	0.175 (5)	0.343 (2)	0.403 (1)	0.180 (4)
Shuttle	0.129 (1)	0.033 (5)	0.127 (2)	0.118 (4)	0.123 (3)
MiniBoone	0.817 (3)	0.841 (2)	0.856 (1)	0.786 (5)	0.816 (4)
Census	0.215 (5)	0.386 (2)	0.288 (4)	0.435 (1)	0.364 (3)
KDDCup1999	0.982 (2)	0.961 (5)	0.985 (1)	0.979 (3)	0.977 (4)
Covertype	0.386 (2)	0.326 (4)	0.414 (1)	0.358 (3)	0.310 (5)

5. Conclusions

This paper presents a new ensemble-based modeling approach for classifying data streams with concept drift. The methodology is based dynamic extension of the ensemble of classifiers according to changes in streaming data distribution. We categorize the concept drift into the change of data distribution and the change of class distribution. In order to deal with concept drift the data distribution is estimated using training data and the class distribution is estimated using classifiers. Our approach generates a meta-ensemble where new ensembles, each of which consists of an only one classifier, are built whenever changes of the estimated data distribution occur and new classifiers of an ensemble are built whenever changes of the estimated class distribution occur. The proposed ensemble-based modeling has the following main characteristics: (1) Our ensemble approach is able to select the most promising samples in an online process which should be labeled; (2) Our approach is able to build a smaller number of classifiers than the chunk-based ensemble approaches; (3) Our approach is able to build new classifiers for an ensemble when the new classifier is necessary, not systematically in time intervals for a fixed number of streaming samples; (4) Our approach is able to dynamically accommodate each ensemble weight for every new sample to be classified, unlike the existing methods where an ensemble keeps classifier weights fixed until the next new classifier is built. (5) We confirmed that our approach is independent of a specific classification algorithm for building new classifiers of an ensemble.

Our ensemble approach was compared with the chunk-based ensemble approach using partially labeled samples. On 13 real data sets, our approach generated an average of

61.7% fewer new classifier of an ensemble than the chunk-based ensemble approach. We implemented two types (SVE-P and WE-P) of the chunk-based ensemble approach using partially labeled samples. SVE-P used the simple majority voting method as the combining method of an ensemble for classification. WE-P used the weighted majority voting method where the each weight of classifiers was calculated according to the method presented by Wang et al. [2]. Our approach produced an average of 25.2% higher classification accuracy than SVE-P and an average of 18.7% higher classification accuracy than WE-P. We showed that these two differences are statistically significant through Wilcoxon's test. We also showed that our approach can build new classifiers at reasonable time points through comparison of the classification accuracy for our meta-ensemble and a single classifier in a time domain.

We are planning to address the problem of maintaining reasonable number of classifiers, including a process to delete classifiers in an ensemble over streaming data. We assume that the deleting mechanism should be designed according to the characteristics of an application. However, Most of the existing ensemble methods delete the oldest classifier in an ensemble when the number of classifiers in the ensemble is larger than the predefined maximum number. We believe that this approach is oversimplified and not appropriate for many real world applications. We are also planning to apply our ensemble-based modeling method to the system recognizing human activities in a smartphone using the smartphone's accelerometer.

Acknowledgement

This research was supported by the 2014 Research Promotion Program sponsored by Shinhan University.

This paper is a revised and expanded version of a paper entitled "Modeling for Classifying Data Streams with Concept Drift" presented at NGCIT2014, Hochiminh (Vietnam), 24-26, Oct., 2014.

References

[1] W.-Y. Kim and S. G. Kim, "The Implementation of the Smart Festival Management Model using Streaming Service based on the Complex Recognition", International Journal of Multimedia and Ubiquitous Engineering, vol.8, no.2, **(2013)**, pp.57-68.

[2] H. Wang, W. Fan, P. S. Yu and J. Han, "Mining concept-drifting data streams using ensemble classifiers", Proceeding of the 9th ACM SIGKDD international conference on knowledge discovery and data mining, pp.226-235, **(2003)** August 24-27; Washington, DC, USA.

[3] P. Zhang, X. Zhu and Y. Shi, "Categorizing and Mining Concept Drifting Data Streams", Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, **(2008)**; Las Vegas, NV, USA.

[4] W. Fan, "StreamMiner: A Classifier Ensemble-based Engine to Mine Concept-drifting Data Streams", Proceeding of the 30th International Conference on Very Large Data Bases, **(2004)** August 29-September 3; Toronto, Canada.

[5] M. M. Masud, J. Gao, L. Khan, J. Han and B. Thuraisingham, "A Multi-partition Multi-chunk Ensemble Technique to Classify Concept-Drifting Data Streams", Lecture Notes in Artificial Intelligence 5476, **(2009)**, pp.363-375

[6] F. Chu, C. Zaniolo, "Fast and Light Boosting for Adaptive Mining of Data Streams", Lecture Notes in Computer Science, **(2004)**, pp.282-292

[7] I. Katakis, G. Tsoumakas and I. Vlahavas, "Tracking recurring contexts using ensemble classifiers: an application to email filtering", Knowledge and Information Systems, vol.22, no. 3, **(2010)**, pp.371-391.

[8] P. Zhang, X. Zhu, Y. Shi and X. Wu, "An Aggregate Ensemble for Mining Concept Drifting Data Streams with Noise", Lecture Notes in Artificial Intelligence, **(2009)**, pp.1021-1029.

[9] Q. Wei, Z. Yang, Z. Junping and W. Youg, "Mining Multi-Label Concept-Drifting Data Streams Using Ensemble Classifiers", Proceeding of the 6th International Conference on Fuzzy Systems and Knowledge Discovery, **(2009)**; Tiangin, China.

[10] G. Song, Y. Li, C. Li, J. Chen and Y. Ye, "Mining Textual Stream with Partial Labeled Instances Using Ensemble Framework", International Journal of Database Theory and Application, vol.7, no. 4, **(2014)**, pp.47-58

[11] C. Woolam, M. M. Masud and L. Khan, "Lacking Labels in the Stream: Classifying Evolving Stream Data with Few Labels", Lecture Notes in Computer Science, **(2009)**, pp.552-562.

[12] L. Xiaofeng and G. Weiwei, "Study on a Classification Model of Data Stream based on Concept Drift", International Journal of Multimedia and Ubiquitous Engineering, vol. 9, no.5, **(2014)**, pp.363-372.

[13] J. W. Ryu, M. M. Kantardzic and M.-W. Kim, "Efficiently Maintaining the Performance of an Ensemble Classifier in Streaming Data", Lecture Notes in Computer Science, **(2012)**, pp.533-540.

[14] J. Gama, R. Sebastião and P. P. Rodrigues, "Issues in Evaluation of Stream Learning Algorithms", Proceeding of the 15th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, **(2009)**; Paris France

[15] J. Demšar, "Statistical Comparisons of Classifiers over Multiple Data Sets", Journal of Machine Learning Research, vol. 7, **(2006)**, pp.1-30.

Authors

Joung Woo Ryu, he received B.S., M.S., and Ph.D. degrees in computer science from Soongsil University, South Korea. Currently, he is a senior research engineer at the technical research center in Safetia Ltd. Co., South Korea. He is also an adjunct professor at School of IT Convergence Engineering, Shinhan University, South Korea. His research interests include data mining & knowledge discovery, machine learning, soft computing, pattern recognition, and robotics.

Jin-Hee Song, she received B.S. degree in computer science from Seoul National University of Science & Technology, South Korea, M.S. degree in computer science from Hankuk University of Foreign Studies, South Korea, and Ph.D. degree in computer science from Soongsil University, South Korea. Currently, she is a professor at School of IT Convergence Engineering, Shinhan University, South Korea. Her research interests include parallel algorithms, distributed systems, embedded system, and data mining.