A survey on factors affecting the accuracy of ENSEMBLE BASED classification approaches for mining data streams

[1]Monika Arya, [2]Chaitali Choudhary
[1, 2]Bhilai Institute of Technology,Durg, Bhilai, CG, India

Abstract : Classification and analysis of data streams are the most promising fields of research and development in Data stream mining. Ensemble based classification approach is one the most challenging flavor of developing an efficient classifier due to large number available base classifiers and increase in the computational time required for training and classification. This paper emphasizes on various factors which affects the accuracy of an ensemble based classifier.

Keywords: Data stream mining, Classification, Ensemble based classification.

I. INTRODUCTION

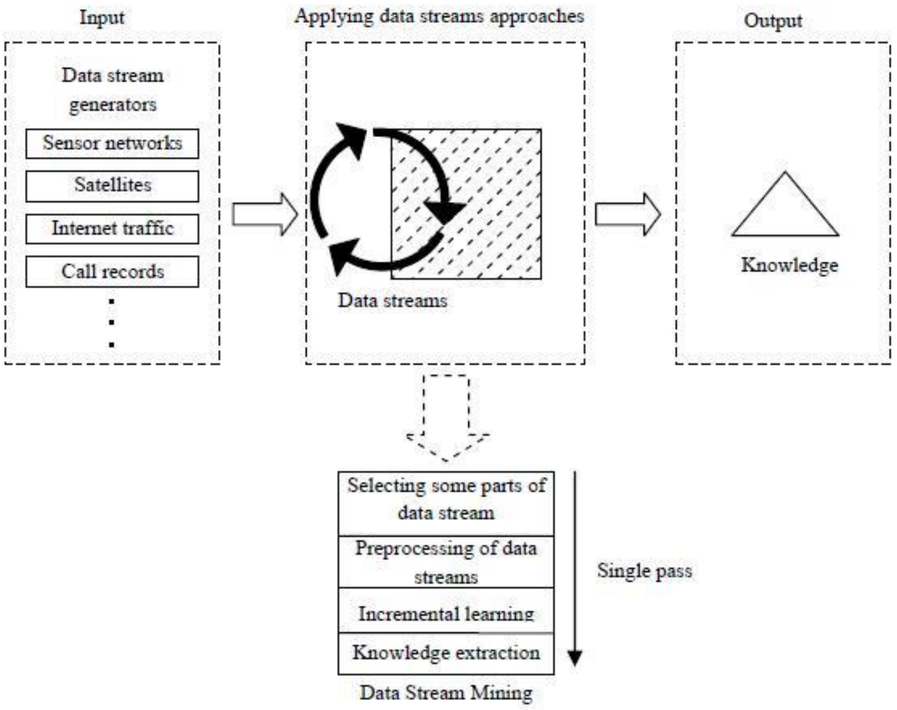
An ensemble classifier is a group of classifiers whose individual decisions are merged in some manner to provide, as an output, a consensus decision [1]. The main idea of ensemble methodology is to combine a set of classifiers in order to obtain more accurate estimations than can be achieved by using a single classifier [2]. Broadly speaking, the ensemble methodology attempts to learn from the errors of the base classifiers with the aim of achieving a more accurate final classifier [3].Ensemble techniques increase classification accuracy with the trade-off of increasing computation time [4]. Training a large number of learners can be time-consuming, especially when the dimensionality of the training data is high. Ensemble approaches are best suited to domains where computational complexity is relatively unimportant or where the highest possible classification accuracy is desired. A number of different approaches to building ensemble learners have been proposed. There are numerous ways to build ensemble systems and there are several decisions to be made which affect the performance of the final model [5]:

- How are subsets of the training data chosen for each individual learner? Training subsets can be chosen by random selection, by examining which training patterns are difficult to classify and focusing on those, or by other means.
- How are classifications made by the different individual learners combined to form the final prediction? They can be combined by averaging, majority vote, weighted majority vote, etc.
- What types of learners are used to form the ensemble? Do all the learners use the same basic learning mechanism or are there differences? If the learners use the same learning algorithm, then do they use the same initialization parameters?
- What should be the size of ensemble i.e How many component classifier should be used to design the ensemble model.

The only drawback of ensemble methods is that they increase the computational time required for training and classification.[4] From literature survey it has been concluded that When constructing an ensemble, the ensemble size affects the accuracy of the ensemble. If there are a smaller number of individual classifiers, then the ensemble will not perform properly, whereas if there is a large number of individual classifier, the ensemble accuracy improves but will lead to increase of storage space and computational time.

The volumes of automatically generated data are constantly increasing. According to the Digital Universe Study, over 2.8ZB of data were created and processed in 2012, with a projected increase of 15 times by 2020[6]. This growth in the production of digital data results from our surrounding environment which is being equipped with more and more sensors. This data can be the source of valuable information which can be used in trend analysis for taking strategic decisions and variety of other business and industrial applications. Examples of data streams include computer network traffic, phone conversations, ATM transactions, web searches, and sensor data. A significant part of such data is volatile, which means it needs to be analyzed in real time as it arrives otherwise it is lost forever. Data stream mining is a research field that studies methods and algorithms for extracting knowledge from volatile streaming data.

The general process of data stream mining is depicted in Fig. 1[7].



There are several applications of data stream classification. For example:

1. Critical astronomical applications.
2. Real time decision support systems in business and industrial applications.
3. Classification and analysis of biosensor measurements around a city for security reasons.
4. Analysis of simulation results for scientific applications.
5. Classification of web log for e-commerce.
6. Classification of data stream for stock markets.

As distinguished from general individual classification methods, including naïve Bayes [23], decision tree [24], and svm [325], the most important idea behind the ensemble methods [26] is the use of a set of base classifiers and combining their predictive capabilities into a single classification task. Through the combination of multiple base classifiers, a more accurate and stronger prediction can be obtained. In recent decades, many researchers have investigated ensemble technology, resulting in a number of outstanding algorithms proposed in the literature, such as bagging [27], adaboost [28], mixture-of-experts [29], and random forest [30]. Nevertheless, there are two primary shortcomings in generic ensemble methods: efficiency and redundant classifiers. According to the survey results reported by Tsoumakas et al. [31], a large-scale ensemble learning task can easily create thousands of base classifiers, or even more.

- Having such a large number of classifiers in an ensemble requires large memory and computational overhead.
- This in turn leads to an increase in the training cost, storage demands, and prediction time.
- In addition, an ensemble with a large number of classifiers does not always generate better prediction results. This is because an ensemble tends to contain redundant classifiers in addition to high-quality ones.

There are several factors that differentiate between the various ensembles methods. The main factors are:

1. Inter-classifiers relationship— How does each classifier affect the other classifiers? The ensemble methods can be divided into two main types: sequential and concurrent.

2. Combining method — The strategy of combining the classifiers generated by an induction algorithm. The simplest combiner determines the output solely from the outputs of the individual inducers. Ali and
Pazzani (1996) have compared several combination methods: uniform voting, Bayesian combination, distribution summation and likelihood combination. Moreover, theoretical analysis has been developed for estimating the classification improvement (Tumer and Ghosh, 1999). Along with simple combiners there are other more sophisticated methods, such as stacking (Wolpert, 1992) and arbitration (Chan and Stolfo, 1995).

3. Diversity generator — In order to make the ensemble efficient, there should be some sort of diversity between the classifiers. Diversity may be obtained through different presentations of the input data, as in bagging, variations in learner design, or by adding a penalty to the outputs to encourage diversity.

4. Ensemble size— the number of classifiers in the ensemble.

All these factors negatively affect the overall ensemble predictive performance. The research focuses on the issue of improving the efficiency of the classifier in terms of performance and accuracy. From the review of the existing research works we can identify Ensemble selection (i.e., ensemble pruning, ensemble thinning, or classifier selection) is regarded as a type of effective technique to improve the efficiency of the classifier. The goal in ensemble selection is to reduce the memory requirement and accelerate the classification process while preserving or improving the predictive ability [10].

Ensemble selection is the process of choosing a subset of all available classifiers that perform well together, since including every classifier may decrease performance. Testing all possible classifier combinations quickly becomes infeasible for ensembles of any practical size and so heuristics are used to approximate the optimal subset. The performance of the ensemble can only improve upon that of the best base classifier if the ensemble has a sufficient pool of accurate and diverse classifiers, and so successful selection methods must balance these two requirements. Just as the name implies, ensemble selection refers to the approaches that address the selection of a subset of optimal classifiers from the original ensemble prior to prediction combination. Given an original ensemble with m base classifiers E = {C1, C2,…, Cm} and a validation (evaluation, pruning, or selection) dataset with k samples D = {(x1, y1), (x2, y2),…, (xk, yk)}, the objective is to form an optimal subensemble E′ = {C1, C2,…, Cn}, where the size of the optimal subensemble, n, is less than or equal to the size of the original ensemble, m (n ≤ m).

Table: A comparison of investigated research paper

Author(s)	Year	Case	Description	CD?	CE?	Tech	Approach
Aggarwal CC et al.[11]	2006	KDD 99'	Considering only labeled instances of data and Building the classifier through an on-demand classifica-tion process which can dynamically select the appropriate window of past training data.	Y	N	I	Supervised micro-clustering, Cluster-based, Sliding window
Peng Zhang et al.[12]	2010	The Malicious URLs Detection dataset. The Intrusion detection Dataset	Accumulating labeled records and combine them to create a classi-fier according to threshold.	Y	N	E	Semi-supervised, Label propagation in clusters and weighting in updat-ing ensemble frame-work.
Masud et al. [13]	2010	Twitter, ASRS, KDD99', Forest	Considering dynamic feature space and classification and addressing feature-evolution	Y	Y	I	Semi-Supervised, Lossless Homogenizing Conversion for feature-evolution
Xingquan Zhu et al. [14]	2010	Data stream generated by Hyperplane-based synthetic data stream generator	Selecting best instanc-es to determine labels by foreign agent by the purpose of decreasing classifier ensemble variance.	Y	N	E	Minimum Variance (MV), optimal weighting
Masud et al. [15]	2011	SynD, SynDE, KDD 99', ASRS	Utilizing both labeled and unlabeled instanc-es to train and update classification model	Y	Y	E	Semi-supervised clustering + Label propagation
Abdulsalam et al.[16]	2011	Synthetic dataset, Sloan, Digital Sky Survey, (SDSS)	Considering multiple target class labels.	Y	N	E	entropy-based concept drift detection, Random Forest
Hosseini et al. [17]	2011	Data stream generated by Hyperplane-based synthetic data stream generator, and Emailing list dataset	Construct a pool of classifier and updating the pool according to new classifier created from new arrived data stream to improve accuracy of the en-semble.	Y	N	E	Bayesian formulation, heuristic methods
Xindong Wua et al.[18]	2012	SEA, STAGGER, KDD’99,Yahoo shopping data,LED	Handling both challenges of concept drifting and unlabeled data streams	Y	N	I	Semi-supervised, k-modes based clustering, statistical approach in detecting concept drifts
Yunyun Wang et al. [19]	2012	Some datasets from UCI repository[31].	Enhancing classifica-tion reliability by consistency check between predictions of two functions. Each instance has likelihood to class labels instead of belonging to only one class.	N	N	I	Semi-supervised, Label membership function, decision function
Dewan Md. Farid et al. [20]	2013	Dewan Md. Farid et al. [9]	Handling concept evolution by consider-ing inter-class distance and intra-class dis-tance.	Y	Y	E	Decision Tree Learning, Similarity Based Clustering

Dariusz Brzezinski[21]	2014	Some datasets from UCI repository	Reacting to different types of concept drift .	Y	N	E	Accuracy-based weighting, Hoeffding Trees
LIU Jing et al. [22]	2014	SynCN, KDD 99'	Data streams classifi-cation with ensemble model based on deci-ion-feedback	Y	Y	E	Novel class label detec-tion, feedback from unsupervised mecha-nisms

II. REVIEW OF LITERATURE

Generally, classification algorithms are grouped in two groups of single and ensemble models. Single models learn incrementally and need new data for updating. The updating process in single models is complicated. Additionally, classification of the data by single models is not one hundred percent reliable. On the other hand, ensemble models are constituted of several single models. In data stream classification, ensemble learning methods enjoy several advantages over other models such as being easily scalable, having parallel function compatibility, and fast change adaptability through pruning of low performance sections. Ensemble classifications also feature high accuracy (Kuncheva, 2004). Under data stream classification, the unlabeled data is fed to the classification system and then assigned with the correct labels. The labeled data can be used to update the classifier model. Data stream classification is always performed at the training stage, as the feedback is the only way to determine concept drift, to adapt, and to update the classification model. An ideal classifier for data streams needs to meet specific features, including: (1) high accuracy, (2)fast change adaptation, (3) low computation and storage load, (4) minimum number of parameters, (5) noise tolerance, and (6) compatibility with new concepts, recursiveness , and optimum use of the past data. Some of these features such as low storage load and recursiveness are controversial and not all these features can be collected in a single system. Due to the specific nature of the data stream, the following requirements must be considered for a data stream classifier (Street and Kim, 2001): (1) all samples are processed only once, (2) limited storage is needed, (3) there is limited data processing time, and (4) the model should provide the best prediction if it stops before the conclusion. The concept adapting very fast decision tree (CVFDT) (Hulten et al., 2001) is an extension to the very fast decision tree algorithm. The algorithm is known for high accuracy and a fast decision tree, showing the capability to detect and respond to change in the process of data sample generation. In fact, CVFDT is capable of detecting and dealing with concept drift. Dynamic weighted majority (DWM) (Kolter and Maloof, 2007) is an ensemble algorithm, which does not use any internal explicit detection method. Concept drift is detected by weighting on the performances of base classifiers.

At first, a classifier is assigned a fixed weight; then, the weight of the classifier is increased/decreased based on its performance and parameter ρ (a factor set by the operator for increasing/decreasing weights). When the classifier error exceeds a threshold level, one of the base classifiers is dropped and replaced by another classifier. Eventually, the majority voting is done by implementing a weighting function on the classifiers. OZAboost is a kind of online boosting algorithm (Oza, 2005). OZAboost updates the weight with a Poisson distribution and is a parallel boosting method that follows Adaboost. OZAboost-Adwin is an extension to OZAboost in which drifts are detected using the ADWIN method. Like the DWM method, an accuracy weighted ensemble (AWE) (Wang et al., 2003) generates variation by weighting base classifiers and employs Hoeffding trees for classification. Instead of using a mechanism to detect drift, the method employs a function for weighting. Regardless of drift, the proposed classifier drops classifications with minimum efficiency and generates a new classification based on the data from the last training step. Standard datasets given in the table 1 below will be used here. The data is available in MOA, which enables us to exactly set the point and place of drift. Therefore, the accuracy and error level of the model can be measured when drift is induced. Different types of data were tested.

Table1: Characteristics of Datasets

Dataset	No.of Drifts	No.of Lables	No.of attributes
SEAS	3	4	3
SEAG	9	4	3
HYPERS	4	2	10
HYPERG	8	2	10
LEDM	4	4	5
WAVE	0	3	40
WAVEM	9	3	40

*S:sudden drift; G: gradual drift; M:combination of drifts including sudden and gradual drifts. For all the datasets, the number of instances is 1*10^5*

We have compared the above discussed algorithms like CVFDT, DWM, OZA, and AWE models regarding accuracy, average required memory, prediction precision, and classification time. Accuracy comparison methods is given in Table 2.

Table 2: Average classification Accuracies (%)

Dataset	DWM	OZA	CVFDT	AWE
SEAS	84.82	82.64	87.71	87.19
SEAG	84.91	83.37	85.00	85.10
HYPERS	88.70	71.65	82.40	87.30
HYPERG	76.84	71.79	71.36	72.17
LEDM	73.95	71.63	68.11	73.58
WAVE	83.82	83.37	83.90	81.57
WAVEM	83.75	83.22	82.71	81.31

Table 3 lists the memory usage. With only one decision tree, the classifier CVFDT needs smaller memory, which is not listed in the table.

Precision

There are different definitions for reliability of an algorithm; here we used precision of classifier, which is composed of smaller elements.

Time

The average run time of the algorithm for 1000 test samples was obtained (Table 5). Due to pre-processing, OZA has a high volume of computation and memory space.

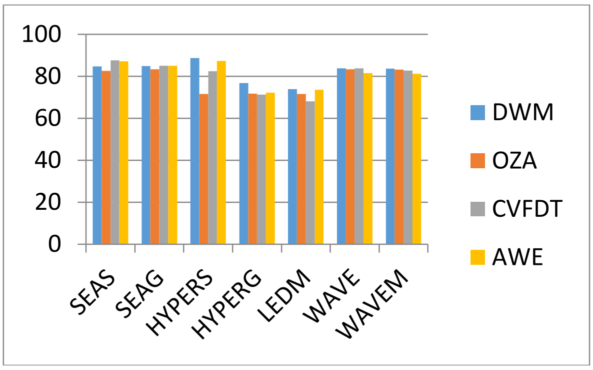


Table 3: Average memory usage

Memory (MB)			
Dataset	DWM	OZA	AWE
SEAS	1.32	6.23	2.66
SEAG	1.73	4.82	1.93
HYPERS	4.24	11.31	3.41
HYPERG	4.37	10.19	3.71
LEDM	0.61	2.56	0.32
WAVE	6.18	69.73	50.63
WAVEM	6.42	26.16	12.29

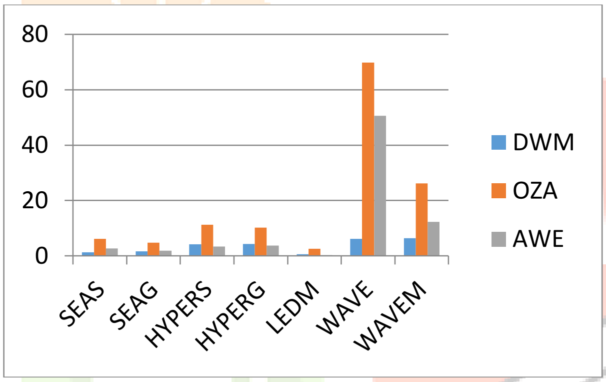


Table 4: Average classification precision

Classification precision(%)			
Dataset	DWM	OZA	AWE
SEAS	65.59	60.92	69.96
SEAG	66.15	62.70	68.72
HYPERS	76.82	41.58	74.59
HYPERG	53.11	40.81	43.84
LEDM	70.33	67.78	69.92
WAVE	75.63	74.95	72.26
WAVEM	73.86	74.57	71.70

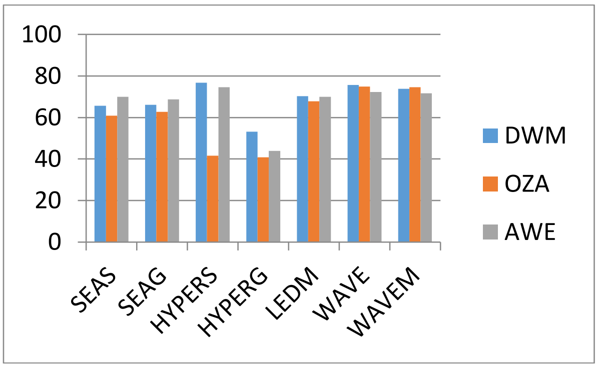
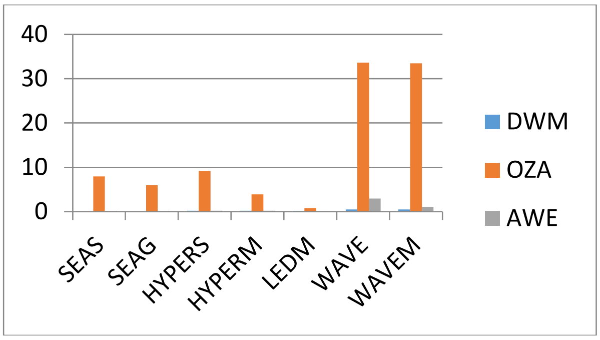


Table 5: Average of time consumption for 1000 test example

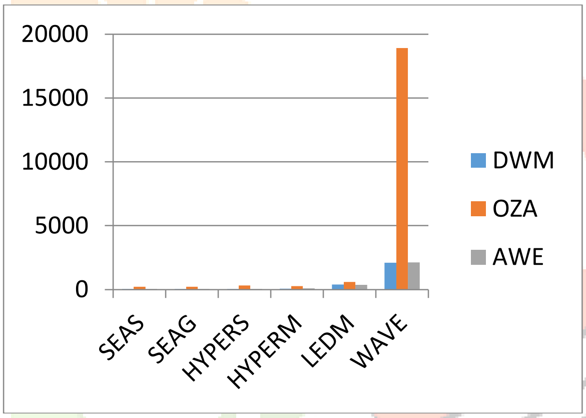
Time (s)			
Dataset	DWM	OZA	AWE
SEAS	0.07	7.94	0.14
SEAG	0.06	5.97	0.09

HYPERS	0.20	9.16	0.20
HYPERM	0.21	3.90	0.22
LEDM	0.15	0.75	0.15
WAVE	0.48	33.65	2.97
WAVEM	0.46	33.46	1.05



In Table 6, we compare the three ensemble methods in terms of execution time.

Time (s)			
Dataset	DWM	OZA	AWE
SEAS	41.22	221.09	43.22
SEAG	35.93	206.11	38.66
HYPERS	47.08	327.40	48.73
HYPERM	74.09	260.55	79.09
LEDM	384.77	597.00	369.56
WAVE	2111.20	18932.00	2120.96



III. CONCLUSION

A three dimensional matrix is being constructed for Stephenson's chain showing the interconnection between all the Here we can conclude that among various ensemble based approaches compared here, some algorithm perform better in one criteria while some perform better in other criteria. The performance can be further improved using different methods, databases settings and parameter tuning that can improve the efficiency of the classifier. Parameters like ensemble selection, ensemble size ,type of dataset ,number of attributes in dataset, number of labels etc are the parameters which affect the overall efficiency of the ensemble classifier. Efficiency in terms of accuracy, precision, memory usage, execution time etc. This survey aims to examine what and how big impact tuning the methods have on the accuracy and what should be studied and developed to achieve greater accuracy in Data stream classification.

REFERENCES

[1] Melville, Wojciech Gryc (2009).Sentiment Analysis of Blogs by Combining Lexical Knowledge with Text Classification.ACM
[2] Rokach, L., & Maimon, O. (2005). Clustering methods. In Data mining and knowledge discovery handbook (pp. 321-352). Springer US.
[3] Martınez-Cámara, E., Gutiérrez-Vázquez, Y., Fernández, J., Montejo-Ráez, A., & Munoz-Guillena, R. (2013). Ensemble classifier for Twitter Sentiment Analysis
[4] Whitehead, M., & Yaeger, L. (2010). Sentiment mining using ensemble classification models. In Innovations and advances in computer sciences and engineering (pp. 509-514). Springer Netherlands
[5] Polikar, R. (2006). Ensemble based systems in decision making. IEEE Circuits and systems magazine, 6(3), 21-45.
[6] Madan, M., & Malhotra, R. (2016). A Descriptive Study on Sentiment Analysis Measures and Methods. Journal of Network Communications and Emerging Technologies (JNCET) www. jncet. org, 6(5).
[7] Mate, C. (2016). Product Aspect Ranking using Sentiment Analysis: A Survey.
[8] website::http://publications.drdo.gov.in/ojs/index.php/dsj/article/view/1088/4752
[9] Liu, H., & Cocea, M. (2016). Granular computing-based approach for classification towards reduction of bias in ensemble learning. Granular Computing, 1-9.
[10] Homayoun, S., & Ahmadzadeh, M. (2016). A review on data stream classification approaches. Journal of Advanced Computer Science & Technology, 5(1), 8.

[11] J. H. Charu C. Aggarwal, Jianyong Wang & Philip S. Yu, "A Fremework for On-Demand Classification of Evolving Data Streams," IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, vol. 18, p. 13, 2006, http://dx.doi.org/10.1109/TKDE.2006.69.

[12] X. Z. Peng Zhang, Jianlong Tan & Li Guo, "Classifier and Cluster Ensembles for Mining Concept Drifting Data Streams," presented at the 2010 IEEE 10th International Conference on Data Mining (ICDM), Sydney, NSW, 2010, http://dx.doi.org/10.1109/ICDM.2010.125.

[13] Q. C. Mohammad M. Masud, Jing Gao, Latifur Khan, Jiawei Han & Bhavani Thuraisingham, "Classification and Novel Class Detection of Data Streams in a Dynamic Feature Space," presented at the European conference on Machine learning and knowledge discovery in databases, Berlin, 2010, http://dx.doi.org/10.1007/978-3-642-15883-4_22.

[14] P. Z. Xingquan Zhu, Xiaodong Lin & Yong Shi, "Active Learning From Stream Data Using Optimal Weight Classifier Ensemble," IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS-PART B: CYBERNETICS, vol. 40, p. 15, 2010, http://dx.doi.org/10.1109/TSMCB.2010.2042445.

[15] C. W. Mohammad M. Masud, Jing Gao, Latifur Khan, Jiawei Han, Kevin W. Hamlen & Nikunj C. Oza, "Facing the reality of data stream classification: coping with scarcity of labeled data," Knowl Inf Syst, vol. 33, p. 32, 2011, http://dx.doi.org/10.1007/s10115-011-0447-8

[16] D. B. S. P. M. Hanady Abdulsalam, "Classification Using Streaming Random Forests," IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, vol. 23, p. 15, 2011, http://dx.doi.org/10.1109/TKDE.2010.36.

[17] Z. A. H. B. Mohammad Javad Hosseini, "Pool and Accuracy Based Stream Classification: A new ensemble algorithm on data stream classification using recurring concept detection," presented at the 11th IEEE International Conference on Data Mining Workshops, 2011, http://dx.doi.org/10.1109/ICDMW.2011.137.

[18] P. L. X. H. Xindong Wua, "Learning from concept drifting data streams with unlabeled data," Neurocomputing, vol. 92, p. 11, 2012, http://dx.doi.org/10.1016/j.neucom.2011.08.041.

[19] S. C. Z.-H. Z. Yunyun Wang, "New Semi-Supervised Classification Method Based on Modified Cluster Assumption," IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, vol. 23, p. 14, 2012, http://dx.doi.org/10.1109/TNNLS.2012.2186825.

[20] L. Z. Dewan Md. Farid, Alamgir Hossain, Chowdhury Mofizur Rahman, Rebecca Strachan, Graham Sexton, Keshav DahalDewan Md. Farid, Li Zhang, Alamgir Hossain, Chowdhury Mofizur Rahman, Rebecca Strachan, Graham Sexton & Keshav Dahal, "An adaptive ensemble classifier for mining concept drifting data streams," Expert Systems with Applications, vol. 40, p. 12, 2013,

[21] D. B. J. Stefanowski, "Reacting to Different Types of Concept Drift: The Accuracy Updated Ensemble Algorithm," IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, vol. 25, p. 14, 2014, http://dx.doi.org/10.1109/TNNLS.2013.2251352.

[22] X. G.-s. LIU Jing, ZHENG Shi-hui, XIAO Da & GU Li-ze, "Data streams classification with ensemble model based on decision feedback," The Journal of China Universities of Posts and Telecommunications, vol. 21, p. 7, 2014, http://dx.doi.org/10.1016/S1005-8885(14)60272-7.

[23] John Holland H. Adaptation in Natural and Artificial Systems. MIT Press; 1992.

[24]Quinlan JR. C4.5: Programs for Machine Learning. 1993;1

[25] Cortes C, Vapnik V. Support-vector networks. Machine Learning. 1995;20(3):273–297.

[26]Dietterich T. Ensemble methods in machine learning. Multiple Classifier Systems. 2000:1–15.

[27] Breiman L. Bagging predictors. Machine Learning. 1996;24(2):123–140.

[28] Freund Y, Schapire RE. A decision-theoretic generalization of on-line learning and an application to boosting. Computational Learning Theory. 1995;55(1):23–37.

[29] Schapire RE. The Boosting Approach to Machine Learning an Overview. New York, NY, USA: Springer; 2003. (Lecture Notes in Statistics).

[30] Svetnik V, Liaw A, Tong C, Christopher Culberson J, Sheridan RP, Feuston BP. Random forest: a classification and regression tool for compound classification and QSAR modeling. Journal of Chemical Information and Computer Sciences. 2003;43(6):1947–1958. [PubMed]

[31]Tsoumakas G, Partalas I, Vlahavas I. An ensemble pruning primer. Studies in Computational Intelligence. 2009;245:1–13.

[32] Zhang Y, Burer S, Street WN. Ensemble pruning via semi-definite programming. The Journal of Machine Learning Research. 2006;7:1315–1338.