Effective Handling of Recurring Concept Drifts in Data Streams

Parneeta Dhaliwal[1]* and M. P. S. Bhatia[2]

[1]Department of CSE, School of Engineering and Technology, K. R. Mangalam University, Gurgaon – 122103, Haryana, India; parneeta07@gmail.com
[2]Division of CoE, Netaji Subhas Institute of Technology, Dwarka, New Delhi - 110078, India; bhatia.mps@gmail.com

Abstract

Background: Nowadays, many applications involve huge amounts of data with variations in underlying concept. This large data needs to be handled with high accuracy, even in a resource-constrained environment. **Objectives**: In order to achieve better generalization accuracy while handling data with drifting concepts mainly recurrent drifts, we proposed an ensemble system called Recurring Dynamic Weighted Majority (RDWM). **Methods**: Our system maintains a primary online ensemble consisting of experts that represent the present concepts and a secondary ensemble that maintains experts representing the old concepts, since the beginning of learning. An effective pruning methodology helps to remove redundant and old classifiers from the system. **Findings:** Experimental analysis using Stagger dataset shows that our system proves to be the best system for handling dataset containing abrupt as well as recurrent drifts, achieving the best prequential accuracy using an optimal window size. RDWM proves to be highly resource effective as compared to EDDM approach. Experimental evaluation using a real world electricity pricing dataset proves RDWM to be the best system, performing very accurately even in a resource-constrained environment. **Improvements:** We can further enhance our system to handle novelty detection in data streams.

Keywords: Concept Drift, Data Streams, Recurring, Recurring Concept

1. Introduction

Data stream mining is the process of analyzing the concepts and the drift in concepts, underlying the data instances[1–3] so as to classify the newly arriving instances with higher accuracy. A concept drift could be a change in features, a change in class label or both. Further, the drift can be sudden, gradual, incremental or recurring. Sudden drift results in the conceptual change within one time step. A gradual drift occurs when the new concept emerges gradually over time. A change is recurrent if the same old data concept reappears. The drift is incremental if at any time the two consecutive concepts are almost similar, but the change is felt after a longer time. Various applications of drifts are Market-Basket analysis[4], computer security, medical diagnosis, etc. Further, a concept drift is measured by its severity and speed. Severity rep-

resents the amount of changes caused by a new concept. Speed is the inverse of the total time taken for a new concept to completely replace the old existing concept.

Online learning approaches[1,2,4–10] for handling concept drift, process each data instance only "once" on arrival without storing it for any further processing. They maintain a current hypothesis that represents the concepts that have arrived so far[11]. The online approaches can be categorized as approaches that explicitly use a mechanism to handle drifts[1,2,5,7] and those that do not explicitly use a mechanism for drift detection[4,6,8]. The former set of approaches use some measure related to their classification accuracy and rebuilds the system upon drift detection. The latter category maintains a set of base learners and updates them, when drift is detected.

An ensemble model maintains a set of experts and trains them as per the newly arriving instances. The classi-

fication of an instance is the combined classification result of the existing learners. The existing literature suggests that an ensemble model always has higher generalization accuracy[12,13] than a single expert model.

In this paper, we propose a new ensemble approach, called Recurring Dynamic Weighted Majority (RDWM) to accurately classify new instances with drifting concepts, mainly recurrent drifts. The primary online ensemble of experts represents the present concepts, with experts being trained and updated as per their classification accuracy in handling the new instances. The secondary ensemble maintains old experts that are neither trained nor updated. In all the experimental evaluations, RDWM performed better or at least similar as the other existing approaches for handling concept drifts.

In the following section, we would be discussing the related work for handling drifts. Section 3 gives the description of various datasets used for evaluation of our approach. In Section 4, we will explain our proposed algorithm in detail. In Section 5, we will present a detailed analysis and evaluation of our approach. In the last section, we summarize our paper and discuss the scope for future research.

Weighted Majority (WM)[14,15] maintains experts and believes that all features are not necessary for making a prediction. Drift Detection Method (DDM)[1] detects concept drift by monitoring the online error-rate generated during prediction whereas Early Drift Detection Method (EDDM)[5] detects drift by monitoring the distances between prediction errors. Adaptive Windowing (ADWIN)[16] uses sliding windows with variable sizes, which are recomputed online as per the rate of change in data.

Dynamic Weighted Majority (DWM)[4,17] is an extended version of WM[14]. It dynamically creates new experts and removes an expert if its weight reaches a given threshold value. Two Online Classifiers for Learning and Detecting Concept Drift (Todi)[18] reduced the impact of false alarms on the classification accuracy. Diversity for Dealing with Drifts (DDD)[19] used the concept of varying diversity levels between ensembles. In Diversified Dynamic Weighted Majority (DDWM)[12], the classification result is the class with the maximum support considering both the low diversity and the high diversity ensembles.

Adaptive Classifier Ensemble (ACE)[20] handles drifts mainly recurrent drifts by using an online classifier, a set of batch classifiers and a drift detection mechanism. An enhanced version of ACE[7] used a pruning strategy so as to remove old redundant learners from the system. Pool and Accuracy based Stream Classification (PASC)[21] handles recurrent concepts by maintaining a pool of experts and classifies instances using two approaches: An active classifier approach and weighted classifier approach. A novel Just-In-Time (JIT) classifier[22] handles recurrent concept drifts by means of a practical formalization of the concept representation and the definition of a set of operators working on such representations. A context-aware data stream learning system[23] exploits available context information to improve existing ensemble approaches for handling recurring concept drifts.

2. Performance Metrics

-- **Prequential Accuracy (%):** It is the average accuracy calculated online by classification of every instance to be learned, prior to its learning.

-- **Kappa Statistic (%)**: It gives a score of homogeneity among the experts[12].

-- **Model cost (RAM-Hours):** One RAM-Hour is equivalent to one GB of RAM being deployed for one hour, giving a measure of resource-efficiency.

-- **Time (CPU-seconds)**: It is the total runtime involved for training of the experts and testing of the system.

-- **Memory (bytes)**: It measures total memory used to store the running statistics and the online model.

3. Concept Drifting Data Streams

3.1 Stagger Dataset

A concept in Stagger dataset[9,10] has 3 features: shape ∈ {triangle, circle, rectangle}, size ∈ {small, medium, large} and color ∈ {blue, green, red}. The dataset consists of 240 instances, with a new instance at each time step. A learner is evaluated based on maximum of a pair of features with at least one of the features being irrelevant. The dataset has been used for evaluating RDWM while handling abrupt drifts as well as recurrent drifts.

3.2 Electricity Pricing Domain

This dataset[24] was obtained from the electricity supplier TransGrid, New South Wales, Australia. It contains 45,312

instances collected at 30-minute intervals between 7 May 1996 and 5 December 1998. Each instance consists of five features and a class label of either up or down. The prediction task is to predict the price of electricity and is affected by demand and supply. As all the real-world drifting datasets represent a real–world phenomenon, we could not predict when and how many times drift has occurred.

4. Recurring Dynamic Weighted Majority (RDWM) Approach

Our proposed system RDWM maintains a primary online ensemble (**EO**), developed using modified version of online bagging[11], containing m experts each with an accuracy weight of one, both pruning weight and accuracy value of zero. The accuracy weight helps in deciding the classification result and the pruning weight helps us to decide the pruning order[18]. The accuracy value measures the classification accuracy of the expert for the most recent window of W (window size) instances. The primary ensemble maintains online experts, which are highly accurate in classifying the present instances. Our approach also maintains a Secondary Ensemble (SE), consisting of the most accurate expert being copied from primary ensemble at times of drift and represent the old concepts that have already arrived in the data stream[18]. Input to the system is n data instances, each consisting of a feature vector and its corresponding class label.

The experts in the primary ensemble are trained, updated as per their accuracy in classifying the new instances[4]. The accuracy weight of an expert in OE is reduced by the multiplicative constant (β)[4], when the local prediction is incorrect. However, when the local prediction is correct, the accuracy value is increased by one, at each time step. After every W instance, the accuracy of each expert is set to zero, to have a comparative analysis of the experts in terms of their classification accuracy on the most recent W instances. Upon drift detection, the primary ensemble is re-initialized. On the other hand, the secondary ensemble does not update nor train its experts. The global prediction by an ensemble for an instance is the Weighted Majority vote of its experts' predictions and is the class with the maximum support[4,18]. After each accuracy weight update, the weights of all the experts in primary ensemble are normalized so that, the maximum value of weight is one. Upon drift detection, the most accurate expert from OE is copied into SE and a new expert is created in OE.

The pruning weights of all the experts in both the ensembles are reduced by one at each time step, except for the most accurate expert on the most recent W instances[18]. Further, if this expert has pruning weight less than zero, we set its weight to zero else increase it by one, at every time step[18]. If the pruning weights of an expert in primary ensemble reaches the threshold value θ, that expert is removed. However, the update of accuracy weights and removal of experts in OE is controlled by the window size (W).

When SE contains at least one expert, the final prediction by our algorithm (G) is the class with the maximum support, involving the Weighted Majority vote of the experts' predictions from both the ensembles. However, when the secondary ensemble is empty, the final global classification result is the class as predicted by the global prediction from primary ensemble.

If the global prediction by both **EO** and **EB** or only by **EO** is incorrect, **EO** is re-initialized to learn the next concept from scratch. If the global prediction by only **EB** is incorrect, a new expert is created in **EO**. The detection and handling of drift by RDWM occurs only after the first $2W$ instances have arrived in the data stream. Training of the experts in **EO** is a continuous process happening at each time step and one could use any base learner considering the various parameters of the base learner.

5. Experimental Evaluation

The various experimental evaluations were done using Massive Online Analysis (MOA)[25], a tool developed for analyzing online data streams. In EDDM, the parameter drift level (β') and the parameter warning level (α) has been set to 0.90 and of 0.95, respectively. For RDWM and DWM, the value of multiplicative factor (β) was set to 0.5 and threshold value (θ) was set to 0.01. For DDM, the minimum number of instances before permitting, detecting a change has been set to 30. The base learner used for the various learning systems was naive bayes, assuming feature independence. Further, DDM and EDDM were used along-with OzaBag[26] so as to have a fair comparison among the various approaches. The ensemble size in DWM is set to double the ensemble size (m) of RDWM.

RDWM was evaluated using Stagger concepts with 240 instances ($n = 240$). The size of ensemble and the window size is set to 3 ($m = 3$) and 10 ($W = 10$), respectively. For DWM, the period value (p) and ensemble size (m') have been set to 10 and 6, respectively. EDDM along-with

OzaBag contains 6 experts. The three concepts are 1. color = green orshape = circle, 2. size = large or size = medium, 3. size = small and color = red. The target concept changes in the order: (1)-(2)-(3), every 80 data instances.

RDWM was evaluated while handling recurrent concept drifts, using Stagger concepts containing 720 instances ($n = 720$). The size of ensemble is set to 3 ($m = 3$) and the window size to set to 10 ($W = 10$). For DWM, the period value (p) is 10 and ensemble size (m') is 6. EDDM was used with OzaBag to contain 6 experts in its system. As per the concepts described earlier, the target concept changes every 80 instances in the order (1)-(2)-(3)-(1)-(2)-(3)-(1)-(2)-(3). For both variations of Stagger concepts, we randomly generate 80 examples of the current target concept, with one example arriving at each time step.

Our proposed approach was evaluated using electricity pricing dataset containing 45,312 instances ($n = 45,312$). The ensemble size is set to 15 ($m = 15$) and the window size is set to 30 ($W = 30$). For DWM, both the period and the ensemble size have been set to 30 each. EDDM was used with OzaBag to contain 30 experts in its system. The instances were processed in the same temporal order as they appear in the dataset, with one example at each time step.

In RDWM, the pruning weight of all the experts is reduced by one at each time step. The pruning methodology helps it to remove old and redundant experts, which may otherwise cause interference in learning the new concepts.

5.1 Experimental Evaluation on Stagger Concepts

On the first target concept, RDWM with naive bayes as base classifier i.e. RDWM-NB has almost similar prequential accuracy as DWM-NB (DWM with NB as base learner), EDDM-NB (EDDM with NB as base learner) and NB classifier, as seen in Figure 1. However, our proposed system achieves the best accuracy on the second and third target concepts as compared to all the approaches. Its accuracy graph has a better slope and asymptote as compared to other systems, reaching the target concepts very quickly. The better accuracy performance of RDWM is solely because of its inherent methodology and not because of the choice of the base classifier.

From the analysis of the results in Table 1 we can state that RDWM performed very poorly in terms of accuracy and kappa statistic, when we measured performance based on only the recent instance i.e. window size (W) of 1. RDWM with window size of 10 performed better as compared to system with window size of 100, achieving an accuracy of nearly 91.40%. Therefore, we can state that for best performance, RDWM should have an optimal window size. If W is too large, our window consists of instances belonging to different concepts and hence our system does not accurately classify any given concept. If W is too small, then there are not enough instances to develop and train a highly accurate model.

While handling Stagger concepts with recurrent drifts our system achieves similar accuracy as DWM-NB, EDDM-NB and the standard NB classifier on the first target concept as shown in Figure 2. However, with progress in learning our system performs the best, achieving the highest accuracy among all the approaches and having quick convergence to the new target concepts. The time and model-cost of RDWM is lower as compared to EDDM, proving RDWM to be a highly resource efficient system, as summarized in Table 2. RDWM is better than DWM, achieving higher accuracy levels with a slight increase in time and cost. Hence, RDWM proves to be the best system for handling recurrent drifts present in the dataset.
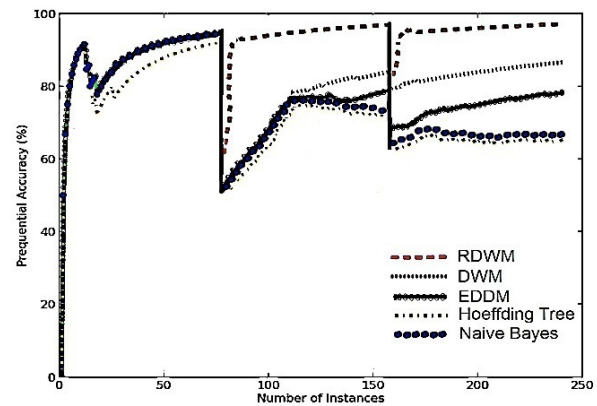

Figure 1. Prequential accuracy performance of RDWM-NB on Stagger concepts.

Table 1. Results of evaluation of RDWM-NB on Stagger concepts with variation in window size (W)

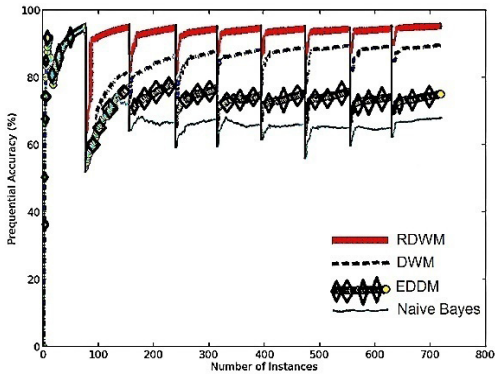|	$W = 10$	$W = 1$	$W = 100$
Accuracy	91.40	66.26	77.83
kappa statistic	71.68	14.97	32.78
model cost (*exp. -9)	0.28	0.28	0.28
Time	0.10	0.10	0.10
Memory	0.01	0.01	0.01

Figure 2. Performance of RDWM-NB on Stagger concepts containing recurrent drifts.

Table 2. Average results of evaluation of RDWM-NB on Stagger concepts with recurrent drifts

	RDWM	**EDDM**	**DWM**	**NB**
Accuracy	93.05	76.51	86.87	69.65
kappa statistic	81.92	44.10	67.11	22.79
model cost (*exp. -9)	0.78	0.84	0.72	0.14
Time	0.32	0.34	0.26	0.12
Memory	0.01	0.00	0.01	0.00

5.2 Experimental Evaluation on Electricity Pricing Domain

RDWM-NB performs the best, achieving the highest prequential accuracy among all the approaches as seen in Figure 3. Overall, NB averaged 73.40% average accuracy, EDDM has accuracy of 84.82%, DWM has 84.18% accuracy whereas RDWM-NB performed with the best average accuracy of 85.80%. RDWM has high adaptability to the new concepts, as compared to the other systems. Further, RDWM provides a highly stable system as compared to EDDM showing a slight variation of nearly 2% in its accuracy levels as compared to a huge variation of nearly 7.5% for EDDM, between time steps, 22,000 and 24,500. RDWM-NB provides better accuracy than DWM-NB and EDDM-NB with a slightly higher evaluation time as analyzed from Table 3.

Table 3. Average results for RDWM on electricity pricing domain, with NB as the base learners

	NB	**RDWM-NB**	**DWM-NB**	**EDDM-NB**
Accuracy (%)	73.40	85.80	84.18	84.82
kappa statistic (%)	39.95	70.80	66.99	68.38

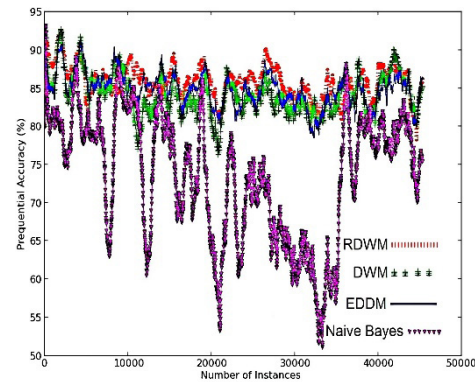| **Time (CPU-seconds)**	1.66	4.72	4.44	3.30
Memory (bytes)	0.01	0.01	0.01	0.00

Figure 3. Accuracy of RDWM on electricity pricing domain using NB as base classifier.

6. Conclusions

In our research paper, we provide an empirical evaluation of our proposed learning system, RDWM for handling drifting concepts, mainly recurrent concept drifts. The detailed analysis of the results using Stagger concepts state that RDWM achieves the best average prequential accuracy among all the approaches, using an optimal window size. RDWM when evaluated using real-time drifting datasets proves to be the best system, performing very accurately even in a resource-constrained environment. For future work, we plan to include fuzzy logic and try to make our system more effective for handling drifting datasets with weights assigned to instances.

7. References

1. Gama J, Medas P, Castillo G, Rodrigues P. Learning with Drift Detection. Proceedings of Advances in Artificial Intelligence – SBIA Brazil; 2004. p. 286–95.
2. Gao J, Fan W, Han J. On appropriate assumptions to mine data streams analysis and practice. Proceedings of 7th IEEE International Conference on Data Mining; USA. 2007. p. 143–52. Crossref
3. Minku FL, White A, Yao X. The impact of diversity on on-line ensemble learn-ing in the presence of concept drift. IEEE Transactions on Knowledge and Data Engineering. 2009 Jul; 22(5):730–42.
4. Kolter JZ, Maloof MA. Dynamic weighted majority. A new ensemble method for tracking concept drift. Proceedings of the 3rd ICDM; USA. 2003. p. 123–30. Crossref

5. Baena-Garcıa M, Campo-Avila JD, Fidalgo R, Bifet A. Early Drift Detection Method. Proceedings of Fourth ECML PKDD Int'l Workshop Knowledge Discovery from Data Streams; Germany. 2006. p. 77–86.

6. Dawid A, Vovk V. Prequential probability principles and proper ties. Bernoulli. 1999; 5(1):125–62. Crossref

7. Nishida K, Yamauchi K. Adaptive classifiers-ensemble system for tracking concept drift. Proceedings of IEEE International Conference on Machine Learning and Cybernetics; 2007. p. 3607–12.

8. Stanley KO. Learning concept drift with a committee of decision trees. Technical Report AI-TR-03-302 Department of Computer Sciences University of Texas. 2003.

9. Sidhu P, Bhatia MPS. An online ensembles approach for handling concept drift in data streams diversified online ensembles detection. International Journal of Machine Learning and Cybernetics. 2015 Apr; 6(6):883–909. Crossref

10. Schlimmer J, Granger R. Beyond incremental processing tracking concept drift. Proceedings of 5th National Conference on Artificial Intelligence; 1986 Aug. p. 502–7.

11. Oza NC, Russell S. Experimental comparisons of online and batch versions of bagging and boosting. Proceedings of Seventh ACM SIGKDD; 2001 Aug. p. 359–64. Crossref

12. Wang XZ, Wang R, Feng HM, Wang H. A new approach to classifier fusion based on upper integral. IEEE Transactions on Cybernetics. 2014 May; 44(5):620–35. PMid: 23782843. Crossref

13. Sidhu P, Bhatia MPS. A novel online ensemble approach to handle concept drifting data streams: Diversified dynamic weighted majority. IJMLC. 2015 Jan; 6(6):883–909. Crossref

14. Littlestone N, Warmuth M. The Weighted Majority algorithm. Information and Computation. 1994 May; 108(1):212–61. Crossref

15. Blum A. Empirical support for winnow and Weighted Majority algorithms re-sults on a calendar scheduling domain. Machine Learning. 1997Jan; 26(1):5–23. Crossref

16. Bifet A, Gavalda R. Learning from time-changing data with adaptive windowing. Proceedings of SIAM International Conference on Data Mining; 2007. p. 443–8. Crossref

17. Kolter JZ, Maloof MA. Dynamic weighted majority. An Ensemble Method for Drifting Concepts. 2007; 8 (1):2755–90.

18. Nishida K. Learning and detecting concept drift. [PhD dissertation]. Hokkaido University; 2008.

19. Minku LL, Yao X. DDD - A new ensemble approach for dealing with concept drift. ITKDE. 2012 Apr; 24(4):619–33. Crossref

20. Nishida K, Yamauchi K, Omori T. ACE Adaptive classifiers-ensemble system for concept-drifting environments. Proceedings of 6th Int'l Workshop on Multiple Classifier Systems; CA USA. 2005. p. 176–85. Crossref

21. Hosseini M, Ahmadi Z, Beigy H. Using a classifier pool in accuracy based tracking of recurring concepts in data stream classification. Evolving Systems. 2013 Mar; 4(1):43–60. Crossref

22. Alippi C, Boracchi G, Roveri M. Just-In-Time classifiers for recurrent concepts. IEEE Transactions on Neural Networks and Learning Systems. 2013 Apr; 24(4):620–34. PMid: 24808382. Crossref

23. Gomes, J, Menasalvas, E, Sousa, P. Learning recurring concepts from data streams with a context-aware ensemble. Proceedings of the ACM Symposium on Applied Computing; 201. p. 994–9. Crossref

24. Harries M. Splice-2 comparative evaluation electricity pricing. Technical Report. Australia: University of New South Wales; 1999.

25. Bifet A, Holmes G, Kirkby R, Pfahringer B. MOA Massive Online Analysis a framework for stream classification and clustering. Proceedings of JMLR Workshop on Applications of Pattern Analysis; Spain. 2010. p. 44–50.

26. Oza N, Russell S. Online bagging and boosting. Proceedings of the Eighth Interna-tional Workshop on Artificial Intelligence and Statistics; Florida. 2001. p. 105–12.