Performance Evaluation and Estimation for Concept Drifting Data Stream Mining

Veena Mittal
CSE, FET, MRIIRS
Faridabad, Haryana, India

Indu Kashyap
CSE, FET, MRIIRS
Faridabad, Haryana, India

ABSTRACT

In Machine learning tasks, mainly in classification problems, the performance measures are of major concern in order to determine and compare the performance of classification methods. In classification problem the accuracy of the classifier is one of the most important performance measures commonly used. However, computing performance in the dynamic environment learning offered for concept drifting data streams also requires some performance considerations as compared to classification tasks in static environment. Furthermore, the learning and testing strategies widely used for training and testing of classifiers of static environments cannot be utilized efficiently to meet the requirements of concept drifting data stream mining as the main requirement of online leaning is to perform one pass incremental learning on large datasets conversely to the allowable iterative learning in static environment with small datasets. This paper describes some important performance measures and learning and testing strategies pertaining to online and incremental learning in the presence of concept drifting data streams. Furthermore, this paper also presents performance measures of drift detection methods widely used as an explicit component in many concept drifting data stream mining algorithms.

Keywords

Data Streams, dynamic environments, classifiers, ensemble learning, online methods.

1. INTRODUCTION

Performance evaluation of classifier is of high concern in decision making. Many performance evaluation measures for classification methods in static environment have been proposed and practiced. Contrary to the evaluation methods of data mining in stationary environment, the non-stationary concept drifting data streaming data mining approaches imposes mandatory requirements of new performance measures other than then the conventional performance measures utilized for stationary environment so far.The new performance measures for the data mining in concept drifting data streaming scenario is required to satisfy the characteristics requirements and constrains of online and incremental data streams mining methods with non-uniform underlying data distribution. While evaluating predictive ability of classifiers it is very crucial to consider the incremental and online learning constraints. The online learning algorithms and incremental learning algorithms are characterized by following properties.as below:

- **Support One Pass Learning:** Unlike to the often iterative methods of learning by static environment learner, the concept drifting data streams classification methods should be one-pass, i.e., the learning should be online and incremental.
- **Low Memory Requirement:** The learning algorithm of drifting data streams should be low on memory requirement but still should be able to contain all possible concepts.
- **Adaptive Towards Changes:** In order to maintain the consistent accuracy of the classifier, the learning method should be able to update the classifier model whenever change in the underlying concept of the data stream appeared.
- **Order Independent:** The learning should not be order dependent.

In concept drifting data streaming environment, the new classes may evolve very frequently, which tends to the change in decision rules of the trained model with respect to time as shown in Fig. 1. In order to achieve high and consistent performance, the trained model required to be updated whenever, the underlying concept of the data stream changes.

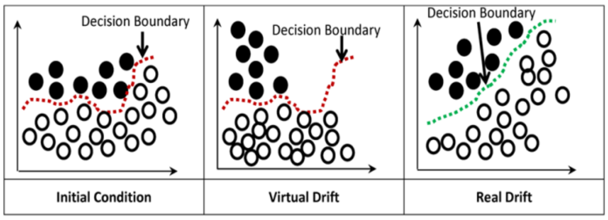

Fig 1:Change in decision boundary due to the occurrence of concept drift in data stream

Moreover, in general, the non-stationary data generating source generates concept drifting data stream at very high speed, therefore it very crucial for data mining methods in such dynamic environment to adopt these changes very fast.

While measuring the performance of the classifiers in concept drifting data stream environment, instead of only considering the overall average performance measures of the classification model, the run-time performance evaluation is of more interest. Such run-time analysis time analysis can facilitate in-depth observation of performance measures of classifiers when concept drifts occur.

To measure the run-time performance of classifiers for the classification of concept drifting data streams, the graphical plots are commonly used. In such depiction of the run-time analysis, the Y-axis represents the performance measure and the X-axis represents the number of instances as shown in Fig 2. for performance measures like accuracy, memory consumption, testing and learning time, etc.

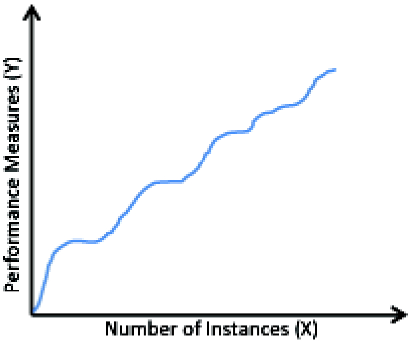

Fig 2: Graphical representation for run-time performance analysis of classification algorithms for concept drifting data streams

The remaining part of this paper is organized as follows; the section 2 formally defines concept drift and explain online and incremental learning methods for data streams. Section 3 presents various methods for mining data streams with concept drifts. Section 4 presents various performance evaluation measures for concept drifting data stream mining. Section 5 explains the estimation methods used in data stream mining.

2. DATA STREAM AND CONCEPT DRIFTS

Classification algorithms are generally characterized by theiraccuracy of classification, their speed of classification and their memory requirement for performing classification [1]. During last few years, the problem of classification of data generated from stationary or static sources has matured enough and providing very accurate results. The stationary data generators generate data continuously with respect to time with uniform data distribution. A data that is generated continuously with time is termed as *data stream* [2-5].

The data stream with uniform data distribution can be learned very easily by utilizing a small set of labeled data belongs to the stream. Such type of learning is called offline learning. The offline can be performed by utilizing *classical batch learning* setting [6].

However, the sensors or data generators of the real world applications are the major sources of data streams with varying underlying data distribution. Continuously generated data streams whose underlying data distribution changes with time are commonly referred as concept drifting data streams[7-11] or dynamic (non-stationary) data streams. There are many real world examples where concept drifting data stream are generated and required to be classified for real-time decision-making. For example, market-basket analysis [12], web page classification [7], sentiment analysis [5, 13-18]. Unlike to the learning algorithms of stationary data streams where the learning is performed offline using batch setting, the concept drifting data streams learning algorithms required to be adaptive with the change in the data distribution. The *Online Learning* and *Incremental Learning* strategies [6] are generally used in such changing environment. Figure 1.2 depicts the incremental learning process in the presence of concept drifts.

The concept drifts in data stream can be described by the help of distribution probability data stream as follows. Let $P(X)$ represents data distribution of X and $P(c/X)$ represents the class conditional probability of class c, the prior probability of the class c is denoted by $P(c)$, then formally the change in $P(c/X)$ is called concept drift. [10].

3. CONCEPT DRIFTING DATA STREAM MINING APPROACHES

The concept drifting data stream mining methods can be broadly categorized in four categories as shown in Figure 4.

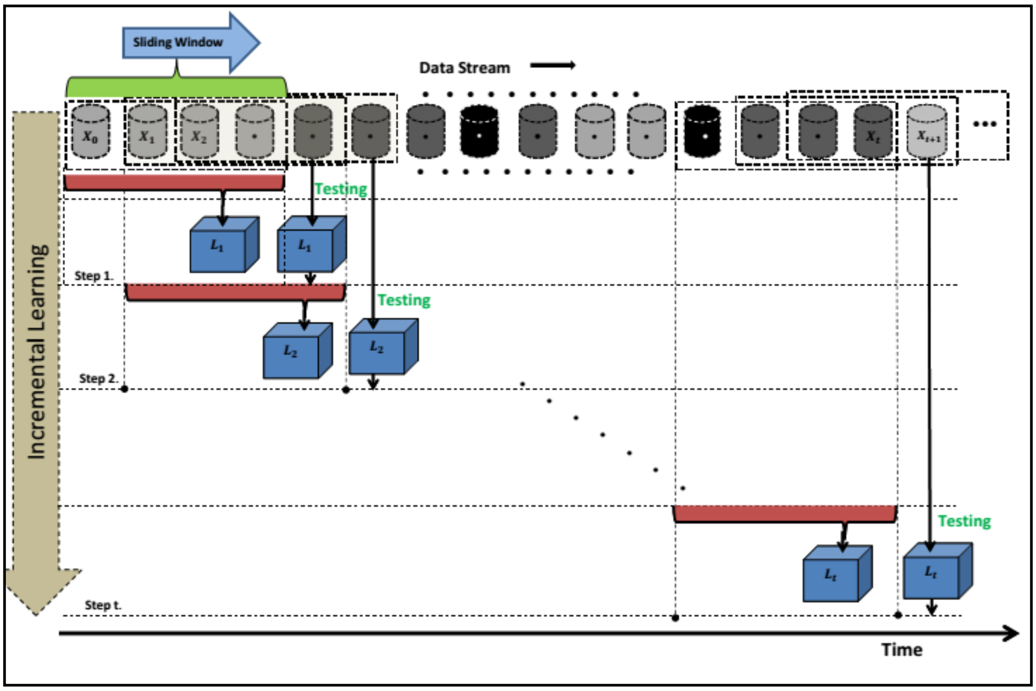

Fig 3.Incremental Learning Process

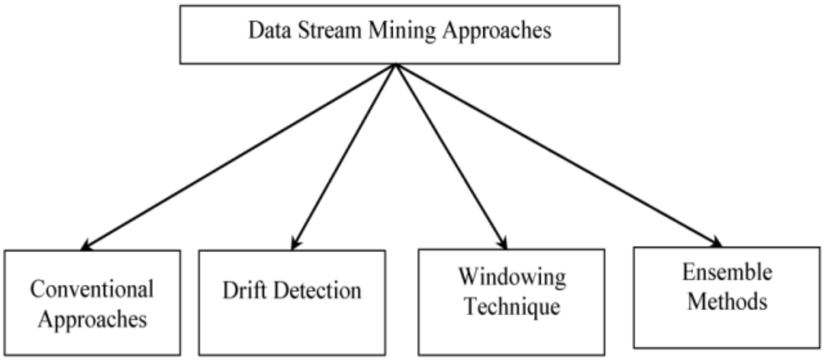

Fig 4: Data stream mining approaches

(i) Classical Single Classifier Approaches
(ii) Explicit Drift Detection Methods
(iii) Window Based Methods
(iv) Ensemble Methods

3.1 Classical Single Classifier Approaches

The single classification methods are the conventional methods used for the classification of stationary data. The most commonly used methods are Naïve Bayesian (NB) [19] and Support Vector Machine (SVM) [20]. The Naïve Bayesian (NB) method and constructed on conditional probability as sown in Equation (1), where C is class labels, x denotes an instance. The NB based on the assumption of independent features.

$$P(C/x) = \frac{P(C)P(x/C)}{P(x)} \quad (1)$$

SVM is another popular method for classification. SVM is based on discriminative model that is defined by separating hyper plane. SVMs are basically linear classifiers that learn decision boundary as given in Equation (2), where is weight vector and b is threshold. SVM try to select the hyper-plan with maximum margin.

$$f(x) = sign\,(w^T x + b) = \begin{cases} +1\ if\ w^T x + b > 0 \\ -1\ otherwise \end{cases} \quad (2)$$

3.2 Explicit Drift Detection Methods

There are certain classification models which exclusively employ an explicit drift detection method to detect for detecting drifts [21-23] as shown in Figure 5. After the detection of driftsthe structure of the classification model rebuilt itself in order to high accuracy by performing learning on the new distribution of the data stream. Two commonly used drift detection methods include drift detection method DDM [24] and Early Drift Detection Method (EDDM) [22].

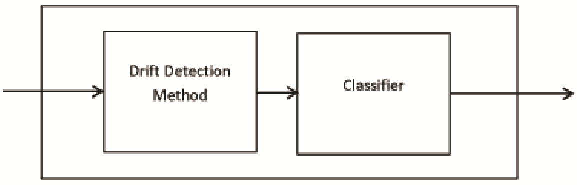

Fig 5.Explicit Drift Detection Method

3.3 Window Based Methods

The window based method is very common in data stream mining. In this method, the data stream is observed for a fixed size chunk of instances. Such methods can also be considered as a process of incremental learning process. Some methods in this category also offered adaptive window size to reduce the delay introduced by fixed window size[25].

3.4 Ensemble Methods

Ensemble learning methods utilized several classifiers in order obtain better accuracy than a single classifier could have[26]. In this method all existing member classifiers of the ensemble make prediction for an instance and finally the decision of all the classifiers are aggregate using some integration, rule, like voted majority.

4. Performance Measures for Concept Drifting Data Stream Mining Techniques

Similar to the performance measures used so far for evaluating the performance measures of stationary or static environment, the performance measures of classification models for the non-stationary concept drifting data stream are also consider the commonly used measures as listed below:-

4.1 Accuracy Based Performance Measures

- **Accuracy:** The accuracy of a classifier **is defined as the ratio** of correctly classified instances with respect to the total number of instances present in the dataset. The accuracy of a classifier can be calculated using Equation (3).

$$Accuracy\ (A) = \frac{Correctly\ Classified\ Instances}{Total\ Number\ of\ Instances} \quad (3)$$

- **Sensitivity & Specificity:** The sensitivity and specificity performance measures the performance of binary classification test. The specificity is called the recall or true positive rate (TPR) or some time also called probability of detection as given in Equation (4). It is the measure of true positive instanced detected by the classifiers. Contrary to sensitivity, the specificity is called true negative rate (TNR) as given in Equation (4). The specificity measures the truly classified negative instances.

$$Sensitivity\ or\ TNR = \frac{True\ Positive\ Instances}{Positive\ Instances}$$

$$= \frac{True\ Positive\ Instances}{True\ Positive + False\ Negative} \quad (4)$$

$$Specificity = \frac{True\ Negative}{Negative\ Instances}$$

$$= \frac{True\ Negative}{True\ Negatice + False\ Positive} \quad (5)$$

- **G- Mean:**The G-Mean (Geometric mean) of a classifier is used to compute the mean of specificity and sensitivity of classifier. The G-mean is commonly used for class imbalanced data streams.
- **Kappa Statistic (K):** Kappa Statistics can be computed as using Equation (6),

$$K = \frac{P_0 - P_c}{1 - P_c} \quad (6)$$

Where, P_0 is the probability that any random classifier makes the correct prediction and P_c is the accuracy of the classifiers currently used for the classification.

- **Generalized Kappa Statistics:** The generalized kappa statistics is more accurate in handling class imbalanced data streams as compared to G-Mean and standard kappa statistics.
- **Receiver Operating Characteristics (ROC)/ AUC**: ROC is a commonly used method for evaluating classifiers. However, because of the involvement of iterative computations, the direct implementation of ROC is not very efficient for large datasets and data streams. To meet such requirements, the ROC can be implemented with the use of periodic hold-out evaluation.
- **Prequential Accuracy:** Prequential accuracy is use to test the new incoming instances one-by-one or in small chunk of instances. This method is highly suited for online learning for concept drifting data streams.

4.2 Performance Measures Other than Accuracy

Although the accuracy of any classifier, but there are many other crucial measures on which the performance of concept drifting data stream mining methods are discussed. Some of them are discussed in this section.

- **Memory Requirement:** The average memory consumption by an algorithm is a major concern for every algorithm. Moreover, the tracking of run-time change in memory requirement is also very crucial concept drifting data stream mining in order to get in-depth run-time observation.
- **Update/ Rebuilt Time:** The update/ rebuilt time in online learning refers to the time that an algorithm needs to adjust the changes in the underlying data distribution whenever concept drifts occur in the data stream. Such kind of action is required to balance the accuracy in drifting data streams.
- **Classification Time:**The classification time is the time required to take decision on incoming instances. The large classification time is discoursed. Most of the iterative methods are having large classification time as compared to one pass learning.

5. Performance Estimation Techniques

The performance estimation techniques are used to estimate the any decided performance measures. In machine learning such estimation techniques are studied as the method of learning and testing. The learning and testing techniques can be broadly classified into two categories for better understanding as given below:

(i) Learning and testing strategies for static data of small size.
(ii) Learning and testing strategies for non-stationary data of large size.

5.1 Learning and testing strategies for static data of small size

In supervised learning, before training the classifier, the labeled dataset is divided into training data and test data. After training, the accuracy of the learned classifier is measured on test data. Such practice of training and testing is termed as ***Batch Setting***. Usually the Batch Setting is preferred in situations where the underlying data distribution of the training dataset remains static. The two well-established methods used in batch settings are:

(i) **Holdout:** In holdout, the labeled dataset in divided into two parts, one is training data and other is test data.
(ii) **K-fold Cross-Validation:** The K-fold Cross-Validation is a variant of holdout, where the labeled dataset is partitioned into k equal samples. Out of the k samples, one sample is used as the validation data for testing and the remaining $k - 1$ samples are used for training purpose. In K- fold cross-validation, this process is repeated for all samples by selecting one sample as validating data and rest as training data.

Although the holdout method has comparatively low learning and testing time complexity, but it has various issues such as:

(i) It reduces the size of training data.
(ii) Unbalanced training and testing.
(iii) Inconsistent in predicting the presence of concept drifts.

The *K-fold cross-validation* method of batch setting guards the issues of training data size and unbalanced training and testing, however it has high time complexity.

5.2 Learning and testing strategies for non-stationary data stream

The high speed concept drifting data stream poses challenge in completely storing the data in memory, due to which the iterative learning is not suitable for such a large data. Furthermore, the necessity of fast decision making also limits the possibility of iterative processing capability offer by batch setting. Therefore, the concept drifting data streams some new one pass approach suitable in this scenario.

5.2.2 *Interleaved Test-Then-Train or Prequential*

This approach is very suitable for learning concept drifting data streams. In this method, each newly arrived instance are first tested before it is used for training. If the instance is classified wrongly by the current model then only the instance is considered for updating the model, otherwise, the instance is dropped. This model satisfies the one pass learning requirement of the classifier in dynamic data streams. Furthermore, it does not require any hold-out.

6. PERFORMANCE MEASURES FOR EXPLICIT DRIFT DETECTION METHODS

This section exclusively presents discusses performance measures of explicit drift detection based methods for classification of concept drifting data stream mining.

- **Probability of True Change Detection:** It is the probability of the learning system for the true detection of the drifts and reacts accordingly. It is the measure that specifies the capacity of the system for true detection.
- **Probability of false alarms:** It is used to measure the resilience to false alarms in absence of drift. It detects the drift even when there is no change in the target concept. The computation of false alarms can be done by computing the detections on the data in absence of

concept drifts where all detections will be considered as false but it is not the case of gradual drifts as there can be drift detection that may occur for more than one time.

- **Delay in or time lag of detection:** It is used to measure the detection of change which is measured after its actual detection. It can use the elapsed time to measure the time lag. It gives the estimation of new instances which are taken into account after the actual drift detection to measure the time lag.
- **Detection point accuracy:** The accuracy of change localization is also important in addition to delay in detection. It can be estimated by making count to the number of instances measured from the change point to that of true change point.

7. CONCLUSION

Learning in non-stationary environment poses challenges in practicing various established performance evaluation and estimation methods used so far for static environment. Due the presence of concept drifts such challenges become more strengthen. Because of these challenges the direct implementation it very necessary to understand new performance measures and performance estimation methods that should meet the requirement of concept drifting data streams. This paper described all major performance evaluation measures and also discussed the estimation methods suitable for concept drifting data mining methods. Furthermore, this paper also discusses the performance evaluation measures for drift detection methods, which are utilized as an explicit component of the complete classification model.

8. REFERENCES

[1] J. Han and M. Kamber, "Classification and prediction," Data mining: Concepts and techniques, pp. 347-350, 2006.

[2] M. M. Gaber, A. Zaslavsky, and S. Krishnaswamy, "Mining data streams: a review," ACM Sigmod Record, vol. 34, no. 2, pp. 18-26, 2005.

[3] V. Mittal and I. Kashyap, "Online methods of learning in occurrence of concept drift," International Journal of Computer Applications, vol. 117, no. 13, 2015.

[4] V. Mittal and I. Kashyap, "Empirical Study of Impact of Various Concept Drifts in Data Stream Mining Methods," International Journal of Intelligent Systems and Applications, vol. 8, no. 12, p. 65, 2016.

[5] R. Srivastava and M. Bhatia, "Quantifying modified opinion strength: A fuzzy inference system for sentiment analysis," presented at the International Conference on Advances in Computing, Communications and Informatics (ICACCI), 2013 2013.

[6] Gepperth and B. Hammer, "Incremental learning algorithms and applications," in European Symposium on Artificial Neural Networks (ESANN), 2016.

[7] K. Nishida, "Learning and detecting concept drift," Graduate School of Information Science and Technology, Hokkaido University, 2008.

[8] Bifet and R. Gavalda, "Learning from time-changing data with adaptive windowing," in Proceedings of the 2007 SIAM International Conference on Data Mining, 2007, pp. 443-448: SIAM.

[9] G. Ditzler, M. Roveri, C. Alippi, and R. Polikar, "Learning in nonstationary environments: A survey," IEEE Computational Intelligence Magazine, vol. 10, no. 4, pp. 12-25, 2015.

[10] Žliobaitė, "Learning under concept drift: an overview," arXiv preprint arXiv:1010.4784, 2010.

[11] D. Brzeziński, "Mining data streams with concept drift," Master’s thesis, Poznan University of Technology, 2010.

[12] P. Angelov, "Nature-inspired methods for knowledge generation from data in real-time," URL: http://www. nisis. risk-technologies. com/popup/Mallorca2006 Papers A, vol. 333, p. 13774, 2006.

[13] R. Srivastava and M. Bhatia, "Real-Time Unspecified Major Sub-Events Detection in the Twitter Data Stream That Cause the Change in the Sentiment Score of the Targeted Event," International Journal of Information Technology and Web Engineering (IJITWE), vol. 12, no. 4, pp. 1-21, 2017.

[14] R. Srivastava and M. Bhatia, "Challenges with Sentiment Analysis of On-line Micro-texts," International Journal of Intelligent Systems and Applications, vol. 9, no. 7, p. 31, 2017.

[15] R. Srivastava and M. Bhatia, "Ensemble methods for sentiment analysis of on-line micro-texts," presented at the International Conference on Recent Advances and Innovations in Engineering (ICRAIE), 2016 2016.

[16] R. Srivastava and M. Bhatia, "Offline vs. Online Sentiment Analysis: Issues With Sentiment Analysis of Online Micro-Texts," International Journal of Information Retrieval Research (IJIRR), vol. 7, no. 4, pp. 1-18, 2017.

[17] R. Srivastava, M. Bhatia, H. K. Srivastava, and C. Sahu, "Exploiting grammatical dependencies for fine-grained opinion mining," in Computer and communication technology (iccct), 2010 international conference on, 2010, pp. 768-775: IEEE.

[18] R. Srivastava, H. Kumar, M. Bhatia, and S. Jain, "Analyzing Delhi Assembly Election 2015 Using Textual Content of Social Network," in Proceedings of the Sixth International Conference on Computer and Communication Technology 2015, 2015, pp. 78-85: ACM.

[19] P. Langley, W. Iba, and K. Thompson, "An analysis of Bayesian classifiers," in Aaai, 1992, vol. 90, pp. 223-228.

[20] C. Cortes and V. Vapnik, "Support-vector networks," Machine learning, vol. 20, no. 3, pp. 273-297, 1995.

[21] D. Kifer, S. Ben-David, and J. Gehrke, "Detecting change in data streams," in Proceedings of the Thirtieth international conference on Very large data bases-Volume 30, 2004, pp. 180-191: VLDB Endowment.

[22] M. Baena-Garcıa, J. del Campo-Ávila, R. Fidalgo, A. Bifet, R. Gavalda, and R. Morales-Bueno, "Early drift detection method," in Fourth international workshop on knowledge discovery from data streams, 2006, vol. 6, pp. 77-86.

[23] W. Raza and K. Ahmad, "A highly selective Fe@ ZnO modified disposable screen printed electrode based non-

enzymatic glucose sensor (SPE/Fe@ ZnO)," Materials Letters, vol. 212, pp. 231-234, 2018.

[24] J. Gama, P. Medas, G. Castillo, and P. Rodrigues, "Learning with drift detection," in Brazilian Symposium on Artificial Intelligence, 2004, pp. 286-295: Springer.

[25] Bifet and R. Gavalda, "Learning from Time-Changing Data with Adaptive Windowing," in SDM, 2007, vol. 7, p. 2007: SIAM.

[26] L. Rokach, "Ensemble-based classifiers," Artificial Intelligence Review, vol. 33, no. 1, pp. 1-39, 2010.

[27]