Learning of DDD

Virendra C. Patil[1] and Sharmila M. Shinde[2]

[1] *Student of ME Computer Engineering, Jayawantrao Sawant College of Engineering, Hadapsar, Pune University, India*

[2] *Assit. Professor, Head of Computer Engineering Department, Jayawantrao Sawant College of Engineering, Hadapsar, Pune University, India.*

Abstract:- Online learning algorithmes often have to operate in the presence of concept drifts. A recent study revealed that different diversity levels in an ensemble of learning machines are required in order to maintain high generalization on both old and new concepts. Inspired by this study and based on a further study of diversity with different strategies to deal with drifts, so propose a new online ensemble learning approach called Diversity for Dealing with Drifts (DDD).DDD maintains ensembles with different diversity levels and is able to attain better accuracy than other approaches. Furthermore, it is very robust, outperforming other drift handling approaches in terms of accuracy when there are false positive drift detections. It is always performed at least as well as other drift handling approaches under various conditions, with very few exceptions. Presents an analysis of low and high diversity ensembles combined with different strategies to deal with concept drift and proposes a new approach (DDD) to handle drifts.

Keywords:- Concept drift, DDD, diversity, online learning.

I. INTRODUCTION

Online learning has been showing to be very useful for a growing number of applications in which training data are available continuously in time (streams of data) and or there are time and space constraints [4].

The work describe is motivated by the belief that successfully being able to capture a new ensemble approach of learning diversity for dealing with concept drift effectively [8],[9],[15][16]. This will lead to increase scope of problem to which random direction mobility model is applied. Examples of such applications are industrial process control, computer security, intelligent user interfaces, market-basket analysis, information filtering, and prediction of conditional branch outcomes in microprocessors. Several definitions of online learning can be found in the literature. In this work, adopt the definition that online learning algorithms process each training example once “on arrival,” without the need for storage or reprocessing [2].

In this way, they take as input a single training example as well as a hypothesis and output an updated hypothesis [3]. Now consider online learning as a particular case of incremental learning. The latter term refers to learning machines that are also used to model continuous processes, but process incoming data in chunks, instead of having to process each training example separately. Ensembles of classifiers have been successfully used to improve the accuracy of single classifiers in online and incremental learning However, online environments are often nonstationary and the variables to be predicted by the learning machine may change with time (concept drift) [1], [2], [3], [4], [5], [6]. For example, in an information filtering system, the users may change their subjects of interest with time. So, learning machines used to model these environments should be able to adapt quickly and accurately to possible changes.

The study of reveals that different levels of ensemble diversity are required before and after a drift in order to obtain better generalization on the new concept[10]. However, even though diversity by itself can help to improve generalization right after the beginning of a drift, it does not provide a faster recovery from drift in long term. So, additional strategies with different levels of diversity are necessary to better handle drifts. This paper provides an analysis of different strategies to be used with diversity, which are then combined into a new approach to deal with drifts. In all the experimental comparisons we have carried out, the proposed approach always performed at least as well as other drift handling approaches under various conditions, with very few exceptions.

A recent study revealed that different diversity levels in an ensemble of learning machines are required in order to maintain high generalization on both old and new concepts. Inspired by this study and based on a further study of diversity with different strategies to deal with drifts, we propose a new online ensemble learning approach called

Diversity for Dealing with Drifts (DDD) [8]. DDD maintains ensembles with different diversity levels and is able to attain better accuracy than other approaches. Furthermore, it is very robust, outperforming other drift handling approaches in terms of accuracy when there are false positive drift detections [8]

II. RELATED WORK

There are many approaches proposed to handle concept drift. So most of these incremental learning approaches [9],.The online learning algorithms which handle concept drift can be divided into two groups: approaches which use a mechanism to detect drifts [8], [13]. Both of them handle drifts based directly or indirectly on the accuracy of the current classifiers. The former approaches use some measure related to the accuracy to detect drifts. They usually discard the current system and create a new one after a drift is detected and/or confirmed.

In this way, they can have quick response to drifts, as long as the drifts are detected early. It is based on the idea that the distance between two consecutive errors increases when a stable concept is being learned. An example of drift detection method is the one used by the approach presented in [13]. So, the distance is monitored and, if it reduces considerably according to a predefined constant which we call γ in this paper, it is considered that there is a concept drift. The approach presented to handle concept drift in [13] is called Early Drift Detection Method.

That different diversity levels are required before and after a drift in order to improve generalization on the old or new concepts and concludes that diversity by itself can help to reduce the initial increase in error caused by a drift, but does not provide a faster recovery from drifts in long term. In online learning, an example of how to explicitly encourage more or less diversity in an ensemble is by using a modified version of online bagging [2], [10]. The original online bagging is based on the fact that, when the number of training examples tends to 1 in offline bagging, each base learner hm contains K copies of each original training example d, where the distribution of K tends to a Poisson (1) distribution. [3] So, in online bagging, whenever a training example is available, it is presented K times for each base learner hm, where K is drawn from a Poisson (1) distribution. The classification is done by unweighted majority vote, as in offline bagging.

III. IMPLEMENTATION

3.1 Proposed Framework

The proposed system framework is an enhancement to techniques introduced in [1].The main motive of proposed system is to improve learning of diversity for dealing with concept drift. This helps user to make DDD is accurate both in the presence and in the absence of drift. The main advantage of proposed system is its user-friendly interface.

Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

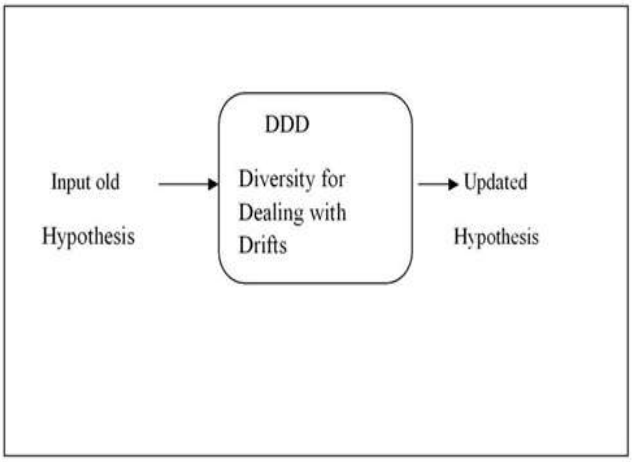

Fig 1. Block Diagram

When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow.

3.2 Detail of System Mode

In this section we will study details of proposed methods or techniques. A new ensemble approach for dealing with concept drift consists of the following modules.

1) Data Conversion:

Data Conversion is the process of converting the dataset of to our database using specialized splitting process.The Dataset is entirely converted as fields in our database in order to freely access the information regarding the database. Whereas the database contains information on required parameters. Such as: Duration, Protocol Type, Service, Flag, Source Bytes, and Destination Bytes.

2) Protocol View:

The protocol view module can be used to view the protocol and its total records from the KDD99Cup.This dataset already extracted and stored in the database. Here 3 protocols are available. Such as, TCP / IP, UDP and IMCP

3) Prequential Accuracy:

The prequential accuray module consists of the following sub modules.Such as, Online Bagging R2L, U2R and Probe

Online Bagging is a well-known ensemble learning method which is used to improving generalization performance.R2L is anunautherized access from a remote machine. U2R is an unauthorized access from a local super user. Probe is used to learning something about the state of the network.

4) DDD Description:

DDD operates in two modes: prior to drift detection and after drift detection. We chose to use a drift detection method, instead of treating drifts implicitly, because it allows immediate treatment of drifts once they are detected. So, if the parameters of the drift detection method are tuned to detect drifts the earliest possible and the approach is designed to be robust to false alarms, we can obtain fast adaptation to new concepts.

3.3 Algorithm:

DDD (Algorithm 1):

It's operates in two modes: prior to drift detection and after drift detection. We chose to use a drift detection method, instead of treating drifts implicitly, because it allows immediate treatment of drifts once they are detected [1]. So, if the parameters of the drift detection method are tuned to detect drifts the earliest possible and the approach is designed to be robust to false alarms, we can obtain fast adaptation to new concepts.

Inputs:

- multiplier constant W for the weight of the old low diversity ensemble;
- online ensemble learning algorithm Ensemble Learning;
- parameters for ensemble learning with low diversity pl and high diversity ph;
- drift detection method Detect Drift;
- parameters for drift detection method pd;
- data stream D;

```
Mode ← before drift,
$h_{nl}$ then new ensemble  / * new low diversity * /
$h_{nh}$ then new ensemble  / * new high diversity * /
$h_{ol}$ tends to $h_{oh}$ tend to  null  / * old low and high diversity * / $acc_{ol}$ tends to $acc_{oh}$  tends to $acc_{nl}$ tends to $acc_{nh}$ tends to 0
 / * accuracies * /
$std_{ol}$ tends to $std_{oh}$ tends to $std_{nl}$ tends to $std_{nh}$tends to 0
/* standard deviations */
While true do d then next example from D
if mode == before drift then prediction tends to  $h_{nl}$(d).
$sum_{acc}$ tends to $acc_{nl}$ + $acc_{ol}$ * W + $acc_{oh,}w_{nl}$ equal to $acc_{nl}/sum_{acc.}$

$w_{ol}$ equal  to $acc_{ol}$ * W/$sum_{acc}$,
$w_{oh}$ tends to $acc_{oh}/sum_{acc,}$
```

```
Prediction tends to WeightedMajority($h_{nl}$(d),
$h_{ol}$(d),$h_{oh}$(d), $w_{nl}$, $w_{ol}$, $w_{oh}$).

Update ($acc_{nl}$, $std_{nl}$, $h_{nl}$, d)
Update ($acc_{ol}$, $std_{ol}$, $h_{ol}$, d)
Update ($acc_{oh}$, $std_{oh}$, $h_{oh}$, d)

End if  drift  tends to  DetectDrift($h_{nl}$, d, $p_d$)
if drift == true then
if mode == before drift OR (mode == after drift AND $acc_{nl}$>$acc_{oh}$) then,

$h_{ol}$ tends to  $h_{nl}$ else $h_{ol}$ tends to $h_{oh}$  end if
 $h_{oh}$ tends to $h_{nh}$,  $h_{nl}$ tends to new ensemble,
$h_{nh}$ tends to new ensemble,

$acc_{ol}$ tends to $acc_{oh}$ tends to  $acc_{nl}$ tends to  $acc_{nh}$  tends to 0
$std_{ol}$ tends to  $std_{oh}$ tends to $std_{nl}$  tends to  $std_{nh}$  tends to  0, mode  tends to after drift end if.
If mode == after drift then

If $acc_{nl}$>$acc_{oh}$ AND $acc_{nl}$>$acc_{ol}$ then mode tends to before drift
else if $acc_{oh}$ - $std_{oh}$>$acc_{nl}$ + $std_{nl}$ AND $acc_{oh}$ - $std_{oh}$>$acc_{ol}$ +   $std_{ol}$ then
 $h_{nl}$ tends to $h_{oh}acc_{nl}$  tends to $acc_{oh}$ mode tends to before drift end if

EnsembleLearning($h_{nl}$; d; $p_l$)
EnsembleLearning($h_{nh}$; d; $p_h$)
if mode == after drift then
EnsembleLearning($h_{ol}$; d; $p_l$)
EnsembleLearning($h_{oh}$; d; $p_l$)
end if
Output: $h_{nl}$; $h_{ol}$; $h_{oh}$; $w_{nl}$; $w_{ol}$; $w_{oh}$, prediction
```

IV. RESULTS

Data Conversion is the process of converting the dataset of to our database using specialized splitting process. The Dataset is entirely converted as fields in our database in order to freely access the information regarding the database. Whereas the database contains information on required parameters required.

The protocol view module can be used to view the protocol and its total records from the Dataset. This dataset already extracted and stored in the database.

V. CONCLUSION

It is presents an analysis of low and high diversity ensembles combined with different strategies to deal with concept drift and proposes a new approach (DDD) to handle drifts. The analysis shows that different diversity levels obtain the best prequential accuracy depending on the type of drift. It also shows that it is possible to use information learned from the old concept in order to aid the learning of the new concept, by training ensembles which learned the old concept with high diversity, using low diversity on the new concept. Such ensembles are able to outperform new ensembles created from scratch after the beginning of the drift, especially when the drift has low severity and high speed, and soon after the beginning of medium or low speed drifts. DDD maintains ensembles with different diversity levels, exploiting the advantages of diversity to handle drifts and using information from the old concept to aid the learning of the new concept. It has better accuracy than EDDM mainly when the drifts have low severity or low speed, due to the use of ensembles with different diversity levels. DDD has also considerably good robustness to false alarms. When they occur, its accuracy is better than EDDM's also during stable concepts due to the use of old ensembles. Besides, DDD's accuracy is almost always higher than DWM's, both during stable concept and after drifts. So, DDD is accurate both in the presence and in the absence of drifts

ACKNOWLEDGEMENTS

I express true sense of gratitude towards my project guide Prof.S.M. Shinde, head of computer department for his invaluable co-operation and guidance that she gave me throughout my project. I specially thank our P.G coordinator Prof. M.D.Ingle for inspiring me and providing me all the lab facilities, which made this project work very convenient and easy. I would also like to express my appreciation and thanks to JSCOE principal Dr. M.G. Jadhav and all my friends who knowingly or unknowingly have assisted me throughout my hard work.

REFERENCES

Journal Papers:

[1]. Leandro L. Minku, Member, IEEE, and Xin Yao, Fellow,"DDD: A New Ensemble Appproach for Dealing with Concept Drift". IEEE Transactions on Knowledge and Data Engineering, Vol. 24, NO. 4, APRIL 2012

[2]. N.C. Oza and S. Russell, "Experimental Comparisons of Online and Batch Versions of Bagging and Boosting," Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining

[3]. Fern and R. Givan, "Online Ensemble Learning: An Empirical Study," Machine Learning,vol.53, pp. 71-109, 2003

[4]. R. Polikar, L. Udpa, S.S. Udpa, and V. Honavar, "Learn++: An Incremental Learning Algorithm for Supervised Neural Networks," IEEE Trans. Systems, Man, and Cybernetics - Part C, vol. 31, no. 4, pp. 497-508, Nov. 2001.

[5]. F.L. Minku, H. Inoue, and X. Yao, "Negative Correlation inIncremental Learning," Natural Computing J., Special Issue on nature-Inspired Learning and Adaptive Systems, vol. 8, no. 2, pp. 289-320, 2009.

[6]. Bock R K and Krisher W. The data analysis brief book, Springer-Verlag, New York, 1998.

[7]. H. Abdulsalam, D.B.Skillicorn, and P. Martin, "Streaming Random Forests," Proc. Int'l Database Eng. and Applications Symp. (IDEAS), pp. 225-232, 2007

[8]. Narasimhamurthy and L.I. Kuncheva, "A Framework for Generating Data to Simulate Changing Environments," Proc. 25th IASTED Int'l Multi-Conf.: Artificial Intelligence and Applications, pp. 384-389, 2007.

[9]. J. Gama, P. Medas, G. Castillo, and P. Rodrigues, "Learning with Drift Detection," Proc. Seventh Brazilian Symp. Artificial Intelligence (SBIA '04), pp. 286-295, 2004.

[10]. J. Gao, W. Fan, and J. Han, "On Appropriate Assumptions to Mine Data Streams: Analysis and Practice," Proc. IEEE Int'l Conf. Data Mining (ICDM), pp. 143-152, 2007.

[11]. F.L. Minku, A. White, and X. Yao, "The Impact of Diversity on On-Line Ensemble Learning in the Presence of Concept Drift," IEEE Trans. Knowledge and Data Eng., vol. 22, no. 5, pp. 730-742, http://dx.doi.org/10.1109/TKDE.2009.156, May 2010.