IJCTA, 9(22), 2016, pp. 317-324

A Hybrid Approach Based on Decision Tree for Data Stream Classification

Pramod Patil* Priyanka Abhang Parag Kulkarni*** Bhagyashree Bhoyar******

***Abstract* :** Due to increasing use of internet and smart devices, there is large increase in network traffic. Therefore, there is need of effective intrusion detection in network traffic. It is significant to propose accurate and efficient decision tree classification model for new incoming data along with selecting only useful high ranked features. Decision tree learning method is used to gather knowledge and for better decision making purpose. In a proposed method, hybrid approach of splitting functions *i.e*. Misclassification error and Gini index along with McDiarmid bound criterion for Gini index is used.McDiarmid bound criterion for Gini index is used by replacing Hoeffding bound in Concept adapting very fast decision treeto split attribute node and to improve performance of tree creation. This proposed hybrid approach outperforms than existing state of art methods. The classification accuracy is increased with increasing number of data elements as well as with selecting only high ranked useful features and proposed method is also efficient in processing large amount of data.

Keywords : Classifier, Data stream mining, Decision tree, Hoeffding bound, Mcdiarmid bound, Gini Index.

1. INTRODUCTION

As the advancement of technology, individuals have acquired capacity of gathering and utilizing data which is unique. Data stream mining is the procedure of mining helpful data from fast flow of data. This is growing quick because of this boundless number of data. To classify stream data and improve its performance is a big challenge.This bulk of data cannot be stored due to memory limitations of storing these huge data. It is needed to mine useful information by scanning data only one time. Thus, in literature various methods have been proposed to handle these data[9-10].

Decision tree[8] is selected as best among many classification processes to work on stream data effectively. In decision tree development process, node is divided into its children node by utilizing some split measures, for example, Information Entropy, Gini Index, Misclassification error. Leaves are utilized to label class to unlabeled information components. Misclassification error has never been utilized for data stream mining [1]. In any case, as of late it has been watched that it is extremely gainful at the start of tree development and Gini Index and Information Entropy perform well at the end of tree. Beforehand the ID3 and the CART algorithm are utilized for preparing information set of fixed size. Those algorithms are not suitable for handling continuous coming data.Some are applicable to handle only concept drifts in which predicted values can be changed as time passes. But this loses prediction accuracy.Many approaches also used labeled as well as unlabeled data for training. Decision tree algorithm such asID3, C4.5 and CART are applicable only for data set of fixed size[5]. To be able to keep classifier model updated, incremental decision tree algorithms performs well[11-12]. Afterwards, algorithms such as(VFDT)Very fast decision tree[7], Concept adaptive very fast decision tree(CVFDT), improved C4.5 are developed to classify stream data. Objectives of proposed system are:

* Department of Computer Engineering DYPIET ,Pimpri, Pune, Maharashtra, India Email- pdpatiljune@gmail.com
** Department of Computer Engineering DYPIET, Pimpri, Pune, Maharashtra, India Email- pabhang4@gmail.com
*** Department of Computer Engineering DYPIET, Pimpri, Pune, Maharashtra, India Email- paragindia@gmail.com
**** Department of Computer Engineering DYPIET, Pimpri, Pune, Maharashtra, India Email- bhagyashree05bhoyar@gmail.com

1. To implement better splitting criteria for nodes of tree.
2. To obtain hybrid approach of split measures for classification.
3. To select top high ranked features.
4. To improve efficiency and accuracy of classification.

The rest of content of paper is given below. In section II, Literature work is given. Section III describes Classifier algorithm and feature selection algorithms. In section IV, Final Results are given. Section V presents conclusion with future work.

2. RELATED WORK

This section discusses existing work done by the researchers for data stream classification.

dsCART algorithm [2] has been proposed for steam data. This algorithm is derived from Classification and regression tree (CART) decision tree algorithm. In this algorithm, criterion to choose best attribute for splitting current node into its further children node, Gaussian approximation is applied. High accuracy of classification is achieved in less processing time. The proposed algorithm requires that for two class problem, in current node smaller number of examples are required for splitting node than required in McDiarmid tree algorithm. Performance parameters as accuracy and tree size are given to compare proposed method, Gaussian tree and McDiarmid tree algorithm.

A method of Gaussian tree classification [3]which gives better results of classification than McDiarmid tree algorithm. This is applicable for stream data. Best attribute selected from finite number of examples is also the best when continuous data is coming. Binary tree is constructed in this method. Previous methods have wrong mathematical justification for splitting criterion and they are very time consuming. But proposed method does not solve concept drift problem. ID3 also takes more time to process time than this method.Experimental results have shown that this method is better in terms of accuracy than McDiarmid tree algorithm.

Splitting criterion of McDiarmid bound [4] was applied for both Information Gain and Gini Index. This method has advantages of high accuracy, low memory consumption and high speed processing. CART and proposed algorithms are compared. Accuracy also increases as data set size increases.

Very fast decision tree(VFDT) algorithm [7] was proposed for stream data in which Hoeffding bound criterion was used to choose smallest examples. This takes very less time to process examples. It takes space only for size of decision tree. It is very adaptable to handle new incoming data. But, memory of old leaf nodes is always released for new leaf nodes whenever it reaches to maximum level.

New data stream classification method is based on decision tree [6].To overcome problem of handling large data set, proposed algorithm is extended version of very fast decision tree (VFDT). It is also able to handle time changing data. Alternative new sub tree is created for each node. It is done by using windowing concept in which old data is removed and new one is updated continuously. When old data become useless then it should be replaced by its new alternative sub tree. This process goes on recursively along with rechecking splitting criterion at current node. This gives an advantage of keeping decision tree up-to-date by using windowing concept and requires small amount of time to process each new example. This algorithm provides higher accuracy than VFDT. But use of Hoeffding bound incorrectly in algorithm makes tree generation ineffective.

An algorithm of hybrid approach [8] of constructing decisiontree was used for incremental learning of classification. Incremental decision tree training Nested generalized exemplar is proposed by modifying Nested generalized exemplar (NGE). To present knowledge in the form of hyper rectangles, set of rules and training data setare used simultaneously. There are two phases in proposed algorithm. In first phase, decision tree is constructed by using C4.5 algorithm. Then it is converted into set of rules. After that, hyper rectangles are established corresponding to each rule from set of rules. In second phase, using original Nested generalized exemplar algorithm, set of hyper rectangles are again rebuilt. When new data comes, then this procedure will be started again for each new incoming data. Prepruning has also been done to remove repeated conditions and because of this performance of system is improved by increasing accuracy. Computational cost of proposed method is low for classification.

Working of classification algorithms such as Hoeffding tree, VFDT (Very Fast Decision Tree), Naive Bayesian and CVFDT (Concept Adapting Very Fast Decision Tree)[5] has been compared along with its advantages and disadvantages. Among these algorithms, only CFVDT algorithm handles concept drift and performs better than Hoeffding tree and VFDT in terms of accuracy.

3. PROPOSED ALGORITHM

3.1. System overview

The following Figure.1 shows system architecture. The description of the system is as follows:

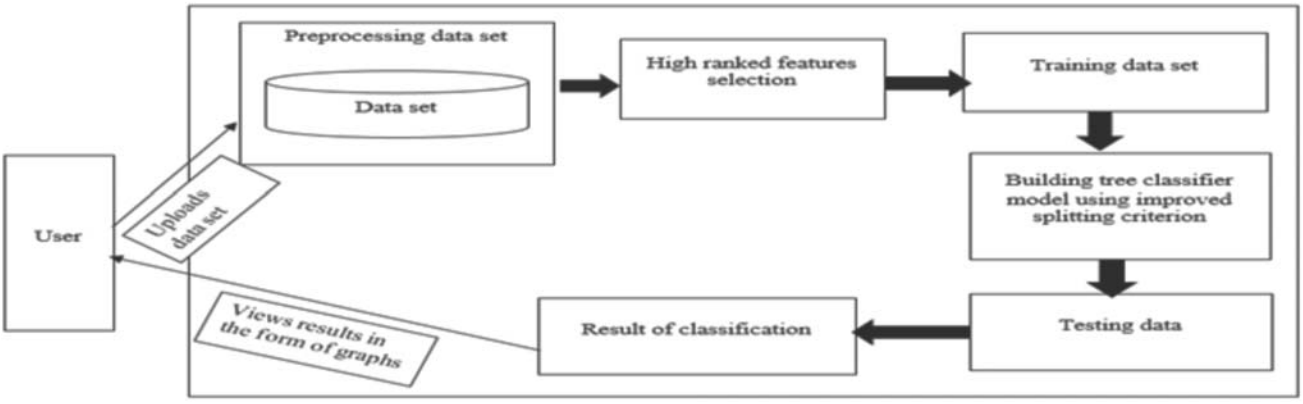

Fig. 1. System Architecture

Steps

1. **Upload Dataset :** Data set of any size is loaded. Arff which is attribute relation file format of data set is selected. Total 33,076 instances to learn classifier model have been taken from KDD CUP 1999 data set.
2. **Preprocessing :** Preprocessing is done to convert data into compatible format for next processing. For proposed system, string values are converted into numerical format.
3. **Feature selection :** Top high ranked 20 features are selected from available features. These features are selected using Chi square method increasing classification accuracy. In this case, processing time is also reduced.
4. **Training data set :** This step consists of training data set and building tree classifier model which will be used to classify further unlabeled data elements.
5. **Testing data set :** Trained data set is used to classify test data in which class labels are assigned to each records.
6. **Result :** Results of processing time, classified data, misclassified data, number of leaf nodes, number of levels, accuracy, false positive rate, selected high ranked features are displayed.

3.2. Proposed Algorithms

This section discusses an algorithm of enhancement of CVFDT by replacing Hoeffding bound with proposed splitting criterion of McDiarmid bound for Gini Index along with Misclassification split measure and feature selection algorithm.

Classifier algorithm

Steps

1. Initially window set and alternate sub tree of each node is empty.
2. Do processing on each incoming instances.
3. Add each instance and its ID to the very beginning of window set.
4. Whenever new instance comes, it is then added to window.

5. Whenever node is formed, ID is given to every node.
6. Each example is added to starting of window.
7. Example is removed if its ID is less than stored value of ID.
8. If overflow of window occurs then oldest instances are removed.
9. Compute entropy.
10. Find two attributes having highest values using Gini index in which McDiarmid bound criterion for Gini index is also checked.
11. Misclassification value of each attribute is calculated and checked.
12. If misclassification value is true and splitting criterion is positive then split node on first highest attribute.
13. Replace each leaf node recursively by an internal node which is ready for splitting.
14. Scanning of every node is done recursively to replace current splitting attribute when new attribute has higher value of split measure.
15. Add new leaves to the branch of node and its alternate sub tree is set to empty.
16. This procedure does scanning of each internal node recursively to make its alternate sub tree.
17. Creation of alternate tree is started when new attribute having highest value of split measure than current one is found
18. Recursively start process of creating alternate sub tree from step 2 to 15.

Feature selection algorithm

Steps

1. Get input dataset having total n features.
2. Expected values of eachattribute for each class is calculated.
3. Observed values of each attribute for each class is calculated.
4. Chi squared statistics value of all attributes is calculated.
5. Based on confidence value, values of all features are compared.
6. Top features having high rank are selected based on user's support value.

3.3. Mathematical Model

$$S = \{I,F,O\}$$

Where S = System, I = Input set, F = Functions set, O = Outputs set.

1. For decision tree classification, input data is in the form of :

$$I = (p,Q) = \{p1, p2, p3,\ldots pn,Q\}$$

Where $p1, p2, p3,.. pn$ are set of features and Q is target variable which is to be classified.

2. To choose root and to split current node into its children nodes, split measures should satisfy following conditions:
   (*a*) Set should contain elements of only one class if any impurity measure value is zero.
   (*b*) If each class has same number of elements, then impurity measure value is maximum.
3. **Gini index :**

$$\text{Gini(D)} = 1 - \sum p_t^2 \quad (3)$$

Where p_t is the relative frequency of class *t*.

4. At each node before splitting, misclassification error value is calculated.

$$g(S) = 1 - \max_{t\in(1,\ldots,Y)} \{p^t(S)\} \quad (2)$$

Use of max function is to give total number of correctly classified data.

Where Y = total number of class.

$g(S)$ is misclassification error of set S.

5. McDiarmid bound for Gini index replacing Hoeffding bound

$$\varepsilon = \sqrt{\frac{\ln \frac{1}{\delta}}{2N}} \quad (3)$$

Where N = number of data elements

6. Chi square statistics for feature selection:

$$X^2 = \sum \frac{(O - E)^2}{E} \quad (4)$$

Where O = Observed frequency of values for one class and

E = Expected frequency of values for one class.

7. O = {O1, O2, O3, O4}

Where O = Set of outputs.

O1 = Classified data and misclassified data without feature selection using Hoeffding bound and Mcdiarmid bound.

O2 = Classified data and misclassified data with feature selection using Hoeffding bound and Mcdiarmid bound.

O3 = Comparison between existing system and proposed system in terms of Accuracy, Time to process data and False positive rate.

O4 = Total no. of leaves and total no. of levels.

4. EXPERIMENT AND RESULT

4.1. Data Set

For evaluating performance of system, KDD CUP 99 data set is used which has been taken from UCI Machine Learning Repository. This is widely used data set for anomaly detection. 33,076 instances have been taken for learning model classifier. Data set includes 41 features of which 7 features are nominal and 34 features are numerical in which class label is normal or anomaly. Table 1 shows example overview of KDD CUP 99 data set.

Table 1. Overview of KDD CUP 1999 data set

duration	*protocol type*	*service*	*src_byte*	*dst_bytes*	*count*	*class*
0	tcp	time	0	0	9	Normal
0	udp	finger	12	1	3	Anomaly
1	icmp	telnet	234	5	7	Anomaly
7	tcp	finger	6	86	4	Normal

4.2. Feature selection

From 41 features of KDD CUP 1999 data set, only top 20 features are selected. These are selected using chi square statistics for all features. Based on confidence values, top 20 high scored features are selected. Selected top 20 features are given in following Table 2.

Table 2. High ranked top 20 features selected

src_byte	dst_bytes	service	flag	same_srv_rate
diff_srv_rate	dst_host_srv_count	dst_host_same_srv_rate	dst_host_diff_srv_rate	logged_jn
count	dst_host_serror_rate	serror_rate	dst_host_srv_serror_rate	srv_serror_rate
dst_host_srv_diff_ host_rate	dst_host_count	dst_host_same_src _port_rate	Srv_diff_host_rate	srv_count

4.3. Performance evaluation

To evaluate performance of system, four parameters are chosen which are given below :

1. **True positive (TP) :** It is normal data elements correctly classified as normal.
2. **False positive (FP) :** It is normal data elements incorrectly classified as anomaly.
3. **True negative (TN) :** It is anomalydata elementscorrectly classified as anomaly.
4. **False negative (FN) :** It is anomalydata elements incorrectly classified as normal.

Using all above four parameters, classification accuracy and False positive rate(FPR) are calculated using below formulae for existing and proposed system :

$$\text{Classification accuracy} : \frac{TP + TN}{TP + FP + FN + TN}$$

$$\text{False positive rate} : \frac{FP}{TN + FP}$$

4.4. Comparison between existing and proposed system

Following result Table 3 shows improved classification performance on 79 test instances of KDD CUP 1999 dataset along with feature selection.Table 4 shows observed values of decision tree statistics on training 33,000 data set.

Table 3. Experiment result

Property	***Existing system (Hybrid algorithm using Hoeffding bound)***	***Proposed system(Hybrid algorithm using Mcdiarmid bound)***
Correctly classified instances	68	73
Incorrectly classified instances	11	6
False positive rate	0.032	0.028

Table 4. decision tree statistics

Property	***Existing system (Hybrid algorithm using Hoeffding bound)***	***Proposed system (Hybrid algorithm using Mcdiarmid bound)***
Training time in milliseconds	1675	697
Number of levels	5	5
Number of leaves	10	10

Following Figure.2 and Figure. 3 shows observed experimental results of accuracy and false positive rate(FPR) between Existing Hybrid algorithm in which Hoeffding bound is used and proposed Hybrid algorithm along with feature selection in which improved splitting criterion McDiarmid bound replacing Hoeffding bound is used. Using McDiarmidbound criterion for Gini Indexreplacing Hoeffding bound in Concept adapting very fast decision tree, classification accuracy is improved.

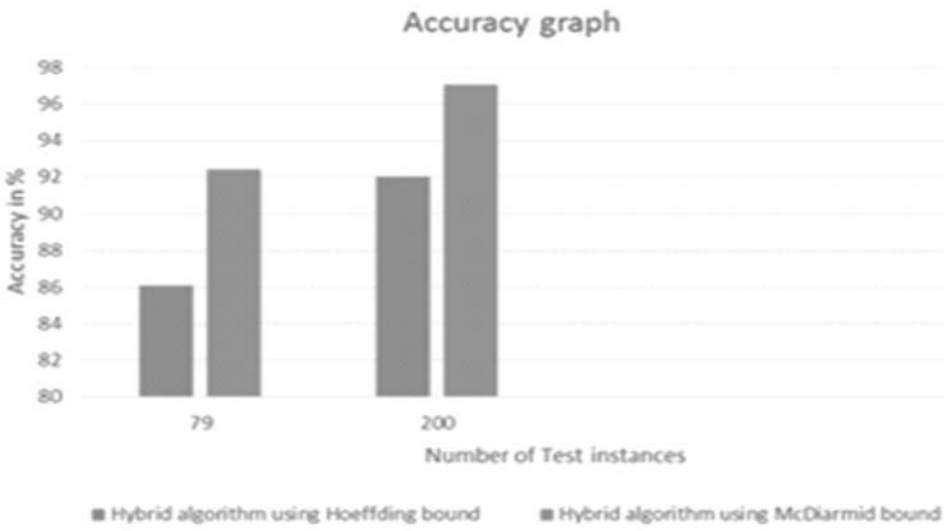

Fig. 2. Accuracy graph

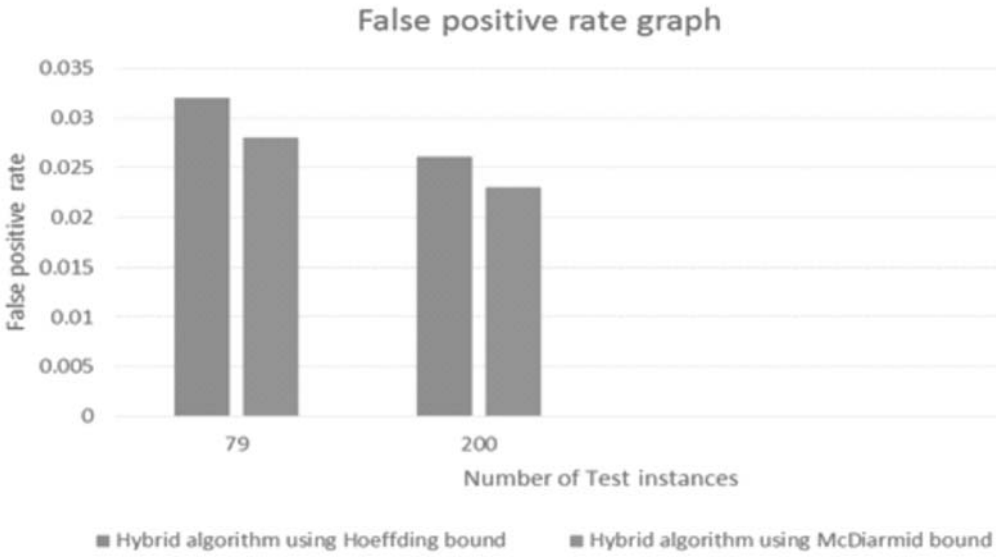

Fig. 3. False positive rate graph

In proposed system, as shown in Figure.4 time to process the data is also minimized by selecting only top useful and high ranked 20 features which reduces time to process the data as well as improves classification performance.

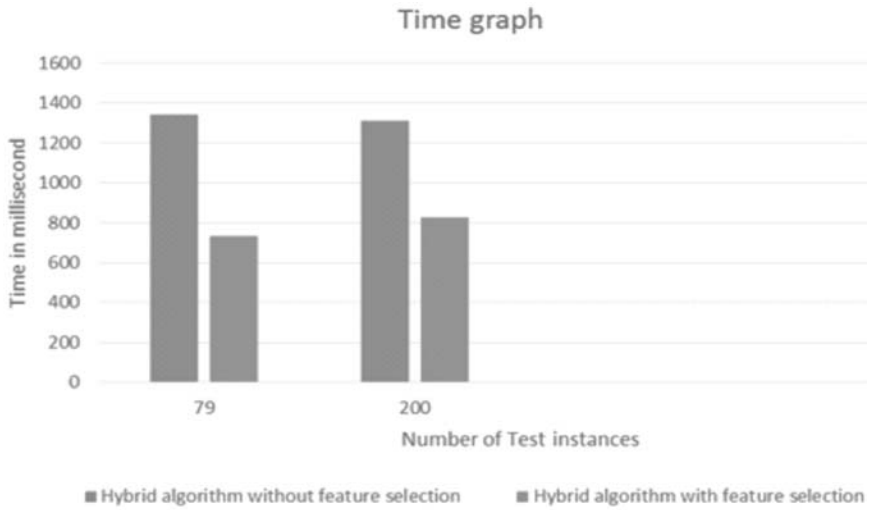

Fig. 4. Time graph

5. CONCLUSION AND FUTURE WORK

Decision tree performs well for classifying stream data. Previous research into this area gives rise to reduce problems occurring during classification. In proposed system, tree construction is based on hybrid approach of best splitting methods to split node into leaves along with improved splitting criterion. This system has been considered for two class problems. And it totally removes sub trees which are not useful. Instead of removing these sub trees, future study can include to keep all sub trees which can be useful again. This system is applicable for intrusion detection, spam detection. Proposed system increases accuracy than previous methods, minimizes time to process the data by selecting high ranked attribute.

6. REFERENCES

1. LeszekRutkowski, Fellow, MaciejJaworski, Lena Pietruczuk, and PiotrDuda, "A New Method for Data Stream Mining Based on the Misclassification Error", IEEE transactions on neural networks and learning systems, vol. 26, no. 5, May 2015
2. L. Rutkowski, M. Jaworski, L. Pietruczuk, and P. Duda, The CART decision tree for mining data streams, Int. J. Inform. Sci., vol. 266, pp. 115, May 2014.
3. L. Rutkowski, L. Pietruczuk, P. Duda, and M. Jaworski, Decision trees for mining data streams based on the Gaussian approximation, IEEE Trans. Knowl. Data Eng., vol. 26, no. 1, pp. 108119, Jan. 2014.
4. L. Rutkowski, L. Pietruczuk, P. Duda, and M. Jaworski, Decision trees for mining data streams based on the McDiarmids bound, IEEE Trans. Knowl. Data Eng., vol. 25, no. 6, pp. 12721279, Jun. 2013.
5. Shukla, MsMadhu S., and MrKirit R. Rathod. "Stream Data Mining and Comparative Study of Classification Algorithms." Algorithms 3.1 (2013).
6. Hulten, Geoff, Laurie Spencer, and Pedro Domingos. "Mining time-changing data streams." Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2001.
7. Domingos, Pedro, and Geoff Hulten. "Mining high-speed data streams."Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2000.
8. M. Wozniak, "A hybrid decision tree training method using data streams," Knowl. Inform. Syst., vol. 29, no. 2, pp. 335–347, 2011.
9. I. Zliobaite, A. Bifet, B. Pfahringer, and G. Holmes, "Active learning with drifting streaming data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 1, pp. 27–39, Jan. 2014.
10. V. Grossi and F. Turini, "Stream mining: A novel architecture for ensemble-based classification," Knowl. Inform. Syst., vol. 30, no. 2,pp. 247–281, Feb. 2012.
11. Yang, Hang. "Solving problems of imperfect data streams by incremental decision trees." Journal of Emerging Technologies in Web Intelligence 5.3 (2013): 322-331.
12. Parita, Ponkiya, and Purnima Singh. "A Review on Tree Based Incremental Classification."