RESEARCH ARTICLE OPEN ACCESS

Novel Ensemble Tree for Fast Prediction on Data Streams

Shashi*, Priyanka Paygude**, Snehal Chaudhary**
**(Department of Information Technology, College of Engineering Bharati Vidyapeeth University, Pune*
*** (Department of Information Technology, College of Engineering Bharati Vidyapeeth University, Pune*
**** (Department of Information Technology, College of Engineering Bharati Vidyapeeth University, Pune*

ABSTRACT
Data Streams are sequential set of data records. When data appears at highest speed and constantly, so predicting the class accordingly to the time is very essential. Currently Ensemble modeling techniques are growing speedily in Classification of Data Stream. Ensemble learning will be accepted since its benefit to manage huge amount of data stream, means it will manage the data in a large size and also it will be able to manage concept drifting. Prior learning, mostly focused on accuracy of ensemble model, prediction efficiency has not considered much since existing ensemble model predicts in linear time, which is enough for small applications and accessible models workings on integrating some of the classifier. Although real time application has huge amount of data stream so we required base classifier to recognize dissimilar model and make a high grade ensemble model. To fix these challenges we developed Ensemble tree which is height balanced tree indexing structure of base classifier for quick prediction on data streams by ensemble modeling techniques. Ensemble Tree manages ensembles as geodatabases and it utilizes R tree similar to structure to achieve sub linear time complexity.

Keywords - Data Stream mining, classification, machine learning, spatial indexing, concept drifting.

I. INTRODUCTION

Nowadays data stream are presents at everywhere in information system, thus it's a resilient task to manage such huge information, additionally the storage and analysis of it. Envision of such huge amounts of data to classify as it is additionally resilient. Utilization of constant or earliest algorithms will be resilient because the information stream possibly will show the construct drift [4]. Therefore it turns out to be a resilient or analysis space for data processing is classification of data stream. There are many possible assumptions for such applications. This kind of data stream classification is universally used everywhere when the intrusion detection is needed, additionally to determine the spam mails or websites. For security reason, biosensor measurements system classification and analysis is a significant growing application. To manage the challenges of huge volume information and construct drifting, some machine learning technique i.e. Ensemble construct has been planned by researchers. This method contains weighted classifier machine learning that's nothing but based on weight of node [1] [2], incremental classifier ensembles, classifier and cluster ensembles [3], etc. Basic technique used by ensemble models is to separate the stream information into a small block of data and from every block of data, it makes the bottom classifier, and after that combine these classifier into several way for prediction.

Prior learning, up to this point are mainly focused on accuracy of ensemble model, prediction adequacy did not considered, regarding a lot of as a result of current ensemble model predicts in linear time, that is sufficient for general or small applications and existing models working on integration small collection of classifier. Though real time application have huge amount of data stream thus we require other base classifier to catch totally different patterns and make an ensemble model which is high grade. Like this applications required fast sub linear prediction resolution. At this point basic classification working in Ensemble model is made discrimination call tree without any loss of generalization. At that time, each base classification module enclosed some rules to create a preference. Each call condition covers a few spaces inside the call space that's referred to as spatial object. During this technique each base category will have been modified to a collection of spatial objects, while ensemble model is modified into a geodatabase. Discriminating this methodology, the problem of classification will be subtracted, that's used for a few patterns sharing among all special entities used for base classification within the geodatabase[5][8] working in machine learning. Numbers of various categorization structures for geodatabase are obtainable that use common patterns for all spatial information. This can be used to decrease the quantity of queries and change prices in database. There are several techniques that are used to

associate index in Training ensemble models like M-tree, R+ -tree. There are a several benefits of using ensemble model categorization as compared to standard methodology of spatial categorization: (1) Ensemble model uses decision rules for indexing, these rules contain a lot of prosperous information like label of categories , likelihood distribution and classifier weight. Typical methodology of categorization planned for particular information like multimedia system, road maps, rectangles and pictures etc. (2) Aim of ensemble model is to predict earlier whereas plan of typical methodology is rapid change and retrieval. (3) Recognition of conception drifting technique, stream of data modification very often, therefore call space applied by call order ensemble models might belong to entirely dissimilar classes' category or category labels.

To solve the challenges mentioned above, we are proposing a novel ensemble tree which integrates the base classifier into a height balanced tree structure for fast prediction that is to achieve the sub linear time complexity. E-tree (Ensemble tree) provides an efficient indexing structure. With the help of this approach we attain logarithmic complexity of time for predicting data streams.

Paper includes the following sections: Section 2 contains all research work and literature survey. In section 3 we proposed the system, it describes the architecture and basic operation of tree. Section 4 and 5 includes mathematical model and experimental results respectively and in section 6 conclusion is presented.

II. LITERATURE SURVEY

There are many algorithms and techniques which are proposed for handling large volume of data or data streams based on ensemble learning. In this paper [6] classification and novel Class detection for Concept-Drifting was proposed for data streams under time constraints, this issue has not been focused by majority of the existing classification technique. Existing techniques of classification systems expect that overall number of classes is fixed in the data stream; it would not be able to find out the novel class up to the time models. When the trained dataset with the novel class is not available for classification model, it is not required to identify novel classes. Novel class identification turns out to be additionally major challenging part in the presence of concept-drift. In this approach they find out the intrusion. In concern to figure out if an instance of data stream adjacent to a novel class, the classification technique intermittently needs to look for additional test instances to find out the similarities between those instances. For classification of an instance wait time Tc is applied as time constraints to discover the novel class in absence of training model for that class. The major difficulties in class detection are: 1.efficiently save training data without much memory utilization, 2. must have knowledge about when to detain the classification of stream and when to immediately classify the test instance 3. Delayed instances are classified within Tc unit of time and 4. Prediction of a novel class rapidly and accurately. They apply their technique on two distinct classifiers: first is decision tree classification and second is k-nearest neighbor classification. But this technique may lead to an inefficiency of classification model, in context of memory utilization and execution time. So as to build more effective model and improve efficiency, they used K means clustering with the training data.

In this paper [2] author proposed solution for large scale classification i.e. large scale or data stream classification problem is overcome, to do this they propose algorithm. They build a committee or ensemble classifier which is a combination of many classifiers, each built on a subset of the accessible data points. Lately, a lot of consideration in the machine learning group has been coordinated towards strategies like bagging and boosting. Their approach was based on the merging of classifiers as follows: Individual classifier is constructed from small data set, read data in blocks rather than entire data set at a time, and then component classifiers are inserted into an ensemble with a fixed size. If ensemble is full, new classifier is inserted only if they satisfy the quality criteria so that the performance of ensemble is improved. Performance estimation is done by testing the existing ensemble and the new tree is built on the next set of data. They performed some experiment based on this framework. Their results included: (1) Better generalization will be achieve by increasing the size of the ensemble up to 20 to 25 classifier. But there is interchange between the number of classifiers and number of points per classifier with data sets of limited size, unless sampling is performed again. (2) Even if the accuracy of the trees was increased, accuracy of ensemble is decreased by pruning the individual trees. (3) Ensemble accuracy does not effect by simple little or more variations in majority. The Key performance of this technique is the strategy used to figure out which existing or outdated tree should be deleted and also whether or not a new tree should be added to the group. Their technique shows that classification diversity is also important; accuracy alone is not the best practice. Ensemble is changing continuously and constantly, so collecting the calculation over time is complicated and also estimates are noisy. Another probability is that to support classifiers which do better on points that are misclassified but it leaves data noisy. So they rather support classifiers on

which ensemble are not decided yet and they classify points correctly.

In next paper [7] author had discussed about ADWIN Bagging and Adaptive-size Hoeffding Tree (ASHT). These are two new approaches i.e. ADWIN Bagging and ASHT proposed for concept drift study. Firstly, they proposed a new method of bagging using Hoeffding Trees of various sizes. It is an incremental approach for constructing tree. Decision tree induction algorithm is used in this method. This algorithm is capable for learning from large volume of data streams, with assumption that the distribution generating examples does not change very frequently. With some differences from Hoeffding Tree algorithm the Adaptive-Size Hoeffding Tree (ASHT) is derived. Listed are as: (1) ASHT has a maximal size or number of split nodes. (2) To reduce its size it deletes some nodes, approach used is as, once node splits, and number of split nodes is greater than the maximum value. Using this method we achieve: small size trees accommodate to changes more quickly whereas large size trees build on large data so they do better in one condition when there are small changes or no changes. Secondly, they proposed new method of bagging using ADWIN. ADWIN solves the problem of tracking the average stream of bits or real-valued numbers in a well specified way. It does not look after any program explicitly, but handles it by exponential histogram technique. There is only one online bagging method named as ADWIN Bagging which is used along with the ADWIN algorithm for detecting a change and estimating the weights of the boosting method. When a change is detected, the most outdated classifier is deleted and a new classifier is added to the tree.

In 2010, Classifier and Cluster ensembles [3] were proposed for mining concept drifting data streams. This paper, overcomes the two main challenges of the existing system of ensemble classifiers which are as follows: (1) obtaining an authentic label of class for every unlabeled cluster. (2) in tree as node represent the class need to decide how to allocate weights to all base classifiers and clusters correctly, so that the concept drifting problem handle by ensemble predictor. To handle these challenges weighted tree classifiers and clusters model is used for concept drifting. To overcome the difficulties (1) first construct a graph which constitute of all classifiers and clusters? This graph is used to presents a new label mechanics to newly introduced class, it firstly promote this label data from all classifiers to the clusters, and then at each iteration of insert operation modifies the outcomes by reproducing resemblance between all clusters. (2) A consistency-based weighing technique is also proposed by author in whom it uses assignment of a weighted value to each base class of decision tree depending on their frequencies with reference to the updated base model. After following this method we get a combined result of both classifiers and clusters for accurate prediction through a weighted averaging schema.

Existing system facing two main problems of knowledge discovery that is large volume of incoming data streams and other one is concept drifting. To overcome this they propose a weighted classifier [1] for mining the data streams with concept drifting. For timely prediction of data streams it is important that data streams should take new and updated patterns. There are some challenges like Accuracy, Efficiency and ease of use for maintaining an up to date and accurate classifier works on large volume of data streams (infinite) with concept drifts. (1) Accuracy: High rate will affect the accuracy of the current model i.e. less accurate, low rate will lead to less delicate model. (2) Efficiency: Even a slight use of the base of them may activate a huge modification in the tree, and may drastically result into a undermine efficiency. (3) Ease of Use: In current classification methods consist of decision trees to handle data streams with drifting concepts in an increased manner. Reusability of the approach cannot be used directly as it is limited. To overcome this, they propose an ensemble of weighted classifier for mining the input data streams with concept drift. As in most of qualified classifier there are chances of valuable information wastage which are less accurate and trained before. In order to avoid over fitting and the problems of conflicting concepts, the deletion of old data must be based on data distribution instead of based on their arrival time. Ensemble approach achieves this competence by assigning all classifier a weight, which is based on the expected accuracy of prediction on current data of the model. Other advantages of this approach is its efficiency and easy to use.

Ensemble model indexing mostly preferred for quick prediction or classification of incoming data stream. Data representation Changes: Data stream changes consistently because of concept drifting, and hidden patterns. Therefore a decision region used for decision tree in machine learning method may be of distinct class labels. To overcome the challenges of existing systems, in this paper we propose a novel ensemble (machine learning) tree that integrates the base classifier into a height balanced tree structure for fast prediction that is to achieve the sub linear time complexity, E-tree (Ensemble tree) provides an efficient indexing structure.

Table 1 Data Streams

Name	Attributes	Areas
Intrusion detection	22	Security
Spam detection	35	Security
Malicious URL	12	Security

III. PROPOSED SYSTEM

In this section we discuss about the E-tree, E-tree structure consist of two parts: first is a tree i.e. R-Tree [5] like structure and second part consist of a table where tree stores decision rules and table stores information about the classifier like ID and classifier weight. Both structures are coupled by linking every base classifier of the table to its related decision rules. E-Tree consists of three basic operations as follows: Search Operation: it is traversing operation that is traveling of tree to classify input data. Insertion Operation: it is used for integrating new base classifier into a tree. Deletion Operation: When E-tree is full, delete operation is called and deletes outdated (not used anymore) classifier.

There are two main modules of the system: Training module and Prediction Module shown in Figure 1. Training Module: For each input data stream (unlabeled stream is coming), in training module this stream data record is stored in buffer until buffer is full for labeling. This labeled data is used for creating or introducing a new class to label data stream which is inserted in an E-Tree using Insertion operation. When E-Tree is full then classifier which is outdated will be deleted using Deletion operation. Prediction Module: Prediction modules also maintain the similar copy of E-Tree generated in training module, so when unlabeled data stream comes, Prediction module call search operation for predicting the unlabeled data. The updated tree from Training module will be integrated or synchronized with the tree in prediction module every time when new classifier is added in E-Tree. Buffer: It is used to store data records until it is full. In this module stream will be labeled by experts. Classifier: This module is used for building a new base classifier from all the labeled data records. We use data streams from KDD data sets which are listed in Table 1.

Operations on E-Tree

Ensemble Tree (E-Tree) are similar to all other tree in data structure like R-Tree and it consist of three main operations which are as follows:

3.1. Search Operation: Whenever a new record comes, search operation is called to predict its class label. In this operation we first travel the tree to find the suitable decision rule which covers the record in the leaf node. We perform the depth first search on tree.

3.2. Insert Operation: Insertion operation on Ensemble based Tree are same as insertion operation on other trees that is, it is used for integrating new

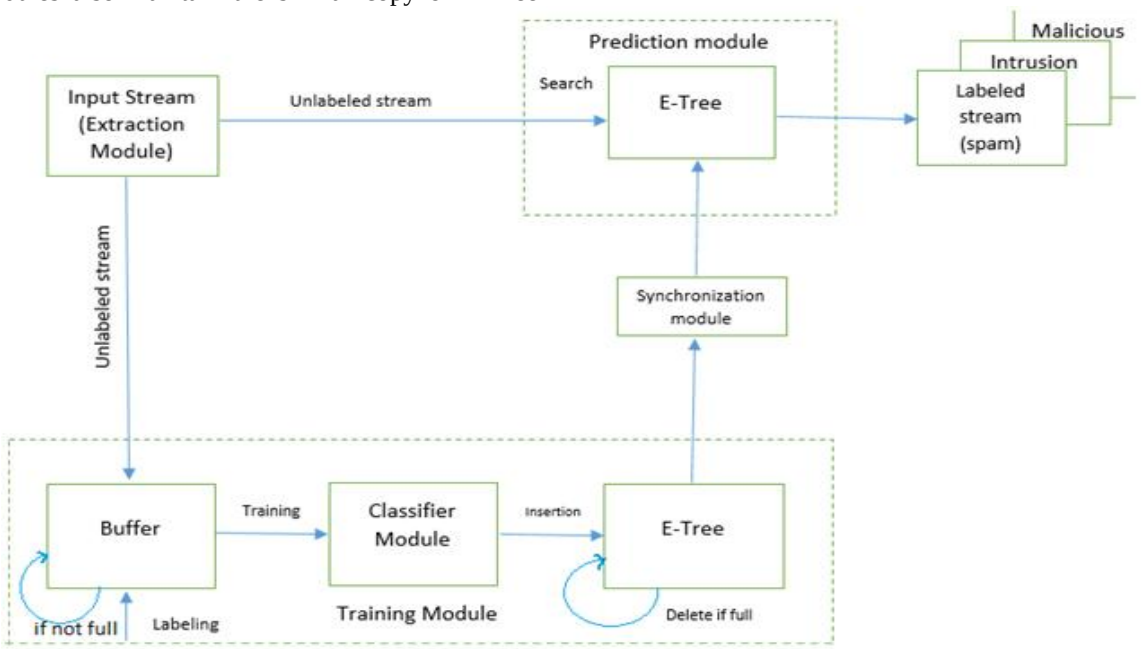

Fig 1. Architecture of Proposed system

base class into E-Tree, this helps to ensemble model to adapt new trends or class of data streams. Insertion operation on Ensemble based Tree are same as insertion operation on other trees that is, It is used for integrating new base class into E-Tree, this helps to ensemble model to adapt new trends or class of data streams.

3.3. Delete Operation: This operation remove or delete the old classifiers which are not in use from long time when the E-Tree reaches its capacity. There are basically two different types of methods for deletion which can be performed. First method is similar to the method used in B tree. Merging operation is performed to all the full nodes to any one sibling that results from the least area increase. The second one appears deletions in R-trees and performs in sequence of delete-then-insert. Firstly it goes to deletion of the under-full node, and then only adds using insert operation on the remaining entries into the tree. This method has more advantageous as: (1)Implementation is easy; and (2)Re-insertion operation will automatically or reproduce the spatial structure of the tree as new class found or some new features are added to existing class training dataset.

3.4 Basic Steps in the J48 Algorithm:
1. The leaf represented by the tree is returned by labeling where the instances belong to the same class.
2. The potential information given by the test done on each attribute is calculated for every attribute.
Then calculated information gain that would result from a test on the attribute.
3. The best attribute is selected for branching and on the basis of present selection criteria the best attribute is founded.

IV. MATHEMATICAL MODEL

Here we consider three class of Data stream DS Consisting of an infinite number of records {(pi, qi)}, Where pi ∈ r^d is a d-dimensional attribute vector, qi ∈ {normal, abnormal} is the class label. Suppose that we have built n base classifiers (a1, a2, a3 …an) from historical stream data using a decision tree algorithm.
Each base classifier Ai (1≤i≥n)
l is decision rules
M=n×l Decision rule ensemble decision rules in the ensemble T.
Sub linear complexity O (log (M)

4.1 Pseudo Code for Search:
Input: E, x, y, where 'E' is e-tree, 'x' is stream record and 'y' is parameter
Output: x'^{s} label of class y_x
Initialize (stack); // S is initialization of stack;
// All rules are recorded by S covering x , Q is E.tree. root;
// calculate tree root

For each entry r ϵ E do
push (stack, r);
While stack is not null do
t← pop (stack)
If entry of leaf is t then
S ← S ∪ t;
else
Q ← t.child;
For each entry t ← Q do
if x ϵ t then
push (stack, t);
For each entry t ϵ S do
Calculate its weight in structure of table;
$y_x \leftarrow$ *to evalute class label* x's
Output is 'y_x';

4.2 Pseudo Code for Insertion:
Input: E, F, n, N
Where 'E' is E-tree, 'F' is classifier and 'n/ N' is parameter
Output: E' as updated E-tree

Q is E.tree.root; // get the tree root
For each r ϵ F do
Where 'r' is decision rule
D ← searchLeaf(r, Q); // D holds r;
If D.size < n then
D ← D ∪ r;
E' ← (D) updatedParentNode;
else
<Q_D, Q_r > ← (D, r) splitNode;
E' ← (E, Q_D, Q_r) adjustTree;
Insert F into the structure of table;
Output: is E' i.e. updated E-tree;

4.3 Pseudo Code for Deletion:
Input: E, F, n, N
Where 'E' is E-tree, 'F' is classifier and 'n/ N' is parameter
Output: is E' as updated E-tree.
Q ← E.tree.root; // obtain the tree root.
G ← E.table.ref; // obtain references
r ← (F, G) searchClassifier; // F is Classifier
Q ← r.pointer;
While Q is not null do
D ← Q.node (); // D is leaf node which contains rule 'Q'
p← Q.sibling;
D' ← deleteEntry (D, Q);
if D.size < n then
E' ← (E, D') deleteNode // deleted;
For each entry t ϵ D' do
E' ← insertRule (t, E);
Q← p;

Output: is E' i.e. updated E-tree;

4.4 Pseudo Code for Random Forest:

To create classifier C:
For i =1 to C do
Training data 'D' is sampled randomly with replacement to produce D_i
Produce a root node, Ni containing D_i
Call BuildTree (N_i)
End for

BuildTree (N):
If M include instances of single class only then
Return
Else
Selects x% of splitting features randomly in M
Select highest information gain feature H to split on
Create h child nodes of M, M2 ,…., Mh , where F has f possible values F1 ,…., Fh
For i = 1 to h do
Put the contents of Mi to Ji
Where Mi and Ji are instances in M that matches to Fi
Call BuildTree (Mi)
end for
end if

V. EXPERIMENTAL RESULTS

For experimental setup we used KDD dataset. We define class label as normal and abnormal class. For this we used three input dataset stream as Intrusion Detection, Spam Detection, Malicious URL detection. Here we used J48 and Random Forest classifier to major the performance of system for above mentioned class. Parameter consider for it accuracy and detection rate (percentage). Fig 2 shows the graphical comparison and As shown in Table2 for Intrusion Detection that Random Forest gives better result but not much different. Same for Spam detection, it does not make major difference. But for Malicious URL detection as shown in Table it makes major difference in accuracy.

Figure 2 shows the classification comparision of the different types of attacks such as normal, U2R, R2L and DOS and Probe. It shows the graph of accuracy vs all these attacks

TABLE 2: Comparison between J48 and Random Forest

Type	Instances		J48		Random Forest	
	J48	Random Forest	Percentage	Accuracy	Percentage	Accuracy
Malicious	4814.0	4887.0	43.55%	73.08%	44.21%	71.16%
Benign	6241.0	6168	56.45%	85.94%	55.79%	84.94%
Spam	2832.0	2769.0	61.55%	98.42%	60.77%	99.77%
Not Spam	1769.0	1805.0	38.45%	97.57%	39.23%	99.56%
Normal	29565.0	29584.0	19.7%	99.85%	19.71%	99.79%
DOS	119014.0	119006.0	79.29%	99.98%	79.28%	99.99%
Probe	1205.0	1208.0	0.8%	98.45%	0.8%	98.69%
R2L	312.0	299.0	0.21%	86.43%	0.2%	82.83%
U2R	12.0	11.0	0.01%	90.91%	0.01%	100.0%

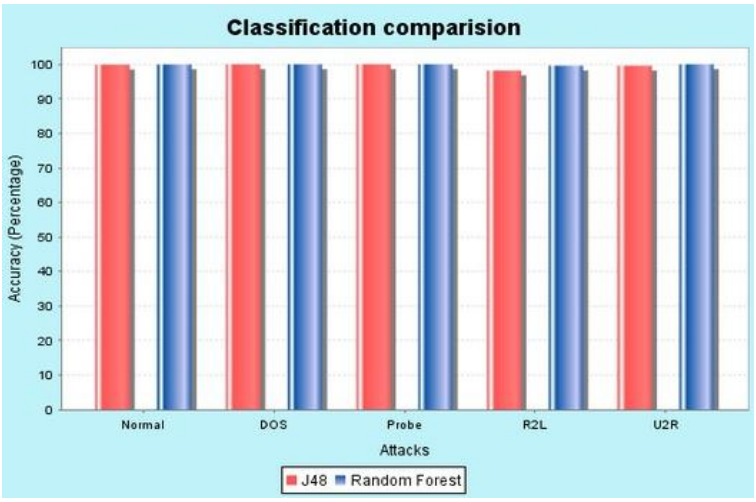

Fig 2 : Intrusion Detection

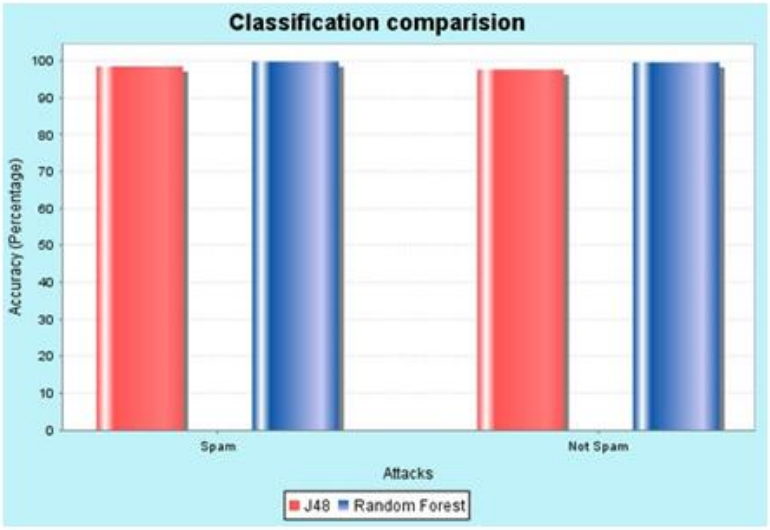

Fig 3: Spam Detection

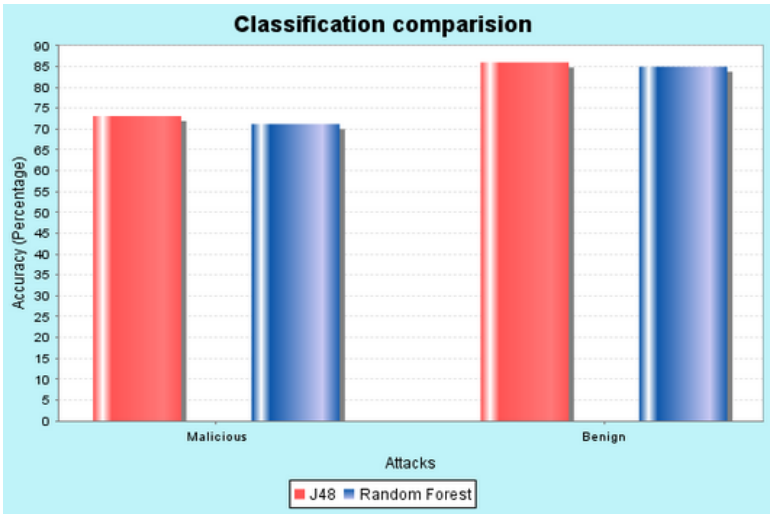

Fig 4: Malicious URL

Figure 3 shows the graph of the classification comparison of the spam and non-spam vs accuracy. Both spam and non-spam has different accuracy.
Figure 4 also shows the graph of classification comparison of the malicious and benign vs accuracy. This figure shows that benign has more accuracy than malicious attacks.

VI. CONCLUSION

In this project, we are proposed a novel Ensemble tree indexing structure for classifying high speed data streams to reduce the expected time complexity for prediction. We are able to reduce the linear time complexity of system to sub linear time complexity of system. Classification in this project is based on the ensemble models. In future we can extend novel Ensemble tree to other classification models of data stream.

REFERENCES

[1]. H. Whang, W. Fan, P.S. Yu, "Mining Concept-Drifting Data Streams using Ensemble Classifier", *ACM SIGKDD International Conference on. Knowledge Discovery and Data Mining (KDD)*, Washington DC, August 24-27, 2003, pp. 226-235.

[2]. W. Street and Y. Kim, "A Streaming Ensemble Algorithm (SEA) for Large-Scale Classification", *ACM SIGKDD international conference on Knowledge discovery and data mining*, San Francisco, CA, August 26-29, 2001, pp. 377-382.

[3]. P. Zhang, X. Zhu, J. Tan and L. Guo, "Classifier and Cluster Ensembles for Mining Concept Drifting Data Streams*", 2010 IEEE International Conference on Data Mining (ICDM)*, Sydney, NSW, December 13-17, 2010, pp. 1175-1180.

[4]. M. Kelly, D. Hand and N. Adams, "The impact of changing populations on classifier performance", *ACM SIGKDD International Conference on. Knowledge Discovery and Data Mining (KDD)*, San Diego, CA, US, August 15-18, 1999, pp. 367-371.

[5]. A. Guttman, "R-Trees: A Dynamic Index Structure for Spatial Searching", *ACM SIGMOD International Conference on Management of Data*, New York, US, June 1984, pp 47-57.

[6]. M. Masud, J. Gao, L. Khan, J. Han and B. Thuraisingham, "Classification and Novel Class Detection in Concept-Drifting Data Streams under Time Constraints", *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no.6, June, 2011, pp. 859-874.

[7]. A. Bifet, G. Holmes, B. Pfahringer, R. Kirkby and R. Gavalda, "New Ensemble Methods for Evolving Data Streams", *15th ACM SIGKDD International Conference Knowledge Discovery and Data Mining (KDD)*, Paris, France, June 28-July 1, 2009, pp. 139-148.

[8]. R. Guting, "An Introduction to Spatial Database Systems", VLDB Journal, 1994 vol.3, no.4, pp.357-399.