A Process for Online Dynamic Learning With Cost Sensitivity in Data Mining

Mr. Prashant Mahakal, Prof. Pritesh Jain
[1]PG Student, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal
[2]Assistant Professor, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal

ABSTRACT:- In general, performance of the classifier is measure using accuracy i.e. on the basis of number of incorrectly predicted instances in testing phase. Cost of what is misclassified is not considered for the measuring performance in general approaches; cost sensitive classification considers cost of the misclassified label. In online learning, prediction model is updated is predicted label and actual label are not same in each round, but in real applications every time getting the actual class is not possible so there come concept of online dynamic learning. Current online dynamic learning systems not consider cost of the misclassification. The systems propose online dynamic learning system which considers the cost of the misclassification. Malicious uniform resource locator (URL) detection is one of the applications where getting actual label of the instance is not possible and class distribution of malicious and normal URL is unbalanced. To evaluate proposed system implemented the Malicious URL detection system using real world dataset which outperforms that existing Malicious URL detection system.

Keywords- Cost-sensitive classification, online anomaly detection, online learning.

I. INTRODUCTION:

cost-sensitive classification and online learning have been researched broadly in data mining and machine learning communities, respectively, there were very few inclusive studies on "Cost-Sensitive Online Classification" in both data mining and machine learning literature. Today, a critical need in data mining and machine learning is to implement proficient and versatile algorithms for mining substantial fast developing data. A hopeful approach is to explore Online Learning, a family of proficient and adaptable machine learning techniques, which have been keenly, examined in literature [1][2].

IN the era of big data, an urgent need in data mining and machine learning is to develop efficient and scalable Algorithms for mining massive rapidly growing data. A promising direction is to investigate *Online Learning*, a Family of efficient and scalable machine learning methods, which has been actively, studied in literature [6], [20], [30].

In general, the goal of online learning is to incrementally learn some prediction models to make correct predictions on a stream of examples that arrive sequentially. Online learning is advantageous for its high efficiency and scalability for large-scale applications, and has been applied to solve online classification tasks in a variety of real-world data mining applications. Different online learning systems have been proposed in literature. For example: the very popular Perception algorithm [15], Passive-Aggressive (PA) learning [1], and numerous other as of late proposed algorithms [11] [12] [13].

Even though being studied widely, most of existing algorithms failed to handle the cost-sensitive classification tasks. The important issue in data mining which should be explored is misclassification costs. The current online learning methods are not successful enough due to the fact that most of existing online learning research worry about the performance of an online classification algorithm regarding prediction mistake rate, which is clearly taken a cost-insensitive and hence improper for a lot of real applications in data mining, particularly for cost-sensitive classification where datasets are frequently class-imbalanced and the misclassification costs of cases from distinctive classes can be extremely different. Researchers have presented many metrics to deal with problem of cost sensitive classification.

Form last decade, significant research has been done in order to develop batch classification algorithms for enhancing the cost sensitive measures. But these algorithms suffer from inefficiency and poor scalability. In communities of Data Mining and Machine Learning, cost-sensitive classification and online learning have been widely examined. Even though these topics are getting more and more attention, very few studies are based on an important concern of Cost-Sensitive Online Classification.

1.1 RELATED WORK:

Our work is mainly related to four groups of research in data mining and machine learning:
(i) Cost-sensitive classification in data mining literature,
(ii) Online learning in machine learning literature,
(iii) Anomaly detection in both data mining and machine learning literature
(iv) Malicious URL Detection:

The work is mainly related to four groups in data mining and machine learning:

(i).Cost-sensitive classification:

Cost-sensitive classification considers the differing costs of distinctive misclassification Types. A cost matrix encodes the punishment of ordering examples from one class as an alternate. Let C (i, j) indicate the cost of predicting an instance from class i as class j. With this documentation (+ ;-) is the cost of misclassifying a positive instance as the negative instance and C(-; +) is the cost of the opposite case. the recognizable proof essentialness of positive instances is higher than that of negative instances. Consequently, the cost of misclassifying a positive instance exceeds the cost of misclassifying a negative one (i.e., C(+, -)>c(-, +)); making a right classification generally shows 0 punishment (i.e., C(+; +) = C(-;-) = 0). The cost-sensitive learning process then tries to minimize the quantity of high cost lapses and the aggregate misclassification cost A cost-sensitive classification strategy considers the cost matrix in the midst of model building and produces a model that has the most decreased cost. Reported works in cost-sensitive learning fall into three categories:

1. Weighting the data space:

The distribution of the training set is modified with regards to misclassification costs, such that the modified distribution is biased towards the costly classes. Against the normal space without considering the cost item, let us call a data space with domain X Y C as the cost-space, in that X is an the input space, Y is an the output space and C is the cost associated with mislabeling that example. If there have examples this proposed system have A Process for Online Dynamic Learning with Cost Sensitivity in Data Mining drawn from a distribution D in the cost-space, then this proposed system can have another distribution D in the normal space that D(X; Y) ^ (C=EXY C _ D[C])D(X; Y ;C) where EXY C _ D[C]is the expectation of cost values. As per the translation theorem, those ideal blunder rate classifiers for D^ will be ideal cost minimizes for D. Hence, when this system update sample weights integrating the cost items, choosing a hypothesis to minimize the rate of error sunder ^D is equivalent to choosing the hypothesis to minimize the expected cost under D.

2. Making a specific classifier learning algorithm cost-sensitive:

For example, in the context of decision tree induction, the tree-building strategies are adapted to minimize the misclassification costs. The cost information is used to: (1) choose the best attribute to split the data [1],[2]; and **(2)** determine whether a subtree should be pruned [3].

3. By use of Bayes risk theory to assign each sample to its lowest risk class:

For instance, an average decision tree for a binary classification issue allots the class label of a leaf node relying upon the greater part class of the training examples that achieve the node. A cost-sensitive algorithm allocates the class label to the node that minimizes the classification cost [4],[5]. Techniques in the first gathering, changing over specimen ward costs into example weights, are otherwise called cost-sensitive adapting by sample weighting [6].

(ii).Online Learning:

Recently a lot of new online learning algorithms have been developed based on the criterion of maximum margin [8],[17],[18],[10],[19]. One notable technique is the Passive-Aggressive (PA) method [10], which updates the classification function when a new example is misclassified or its classification score does not exceed some predefined margin. In the proposed system PA algorithm is apply to solve the online learning task. Different from the regular PA learning setting which assumes class label of every online incoming instance will be revealed, this proposed system approach queries the class labels of only a limited amount of online incoming instances through active learning. In addition to regular online learning techniques, this proposed system work is also closely related to another online learning topic in machine learning, that is, selective sampling [20],[21] or label efficient learning [22],[23], which also queries class labels of a subset of online received instances by developing appropriate sampling strategies. However, conventional label efficient learning approaches often aim to optimize the mistake rate (or equivalently the classification accuracy), which is clearly inappropriate for malicious URL detection tasks. In contrast, this proposed system approach addresses the challenge of online malicious URL detection by attempting to optimize cost-sensitive metrics (either weighted sum of sensitivity and specificity or weighted cost) [24]. Finally, this proposed system work generally belongs to the category of "online" active learning, which differs from a large family of "batch" active learning studies in literature.

3. Anomaly Detection:

Anomaly detection is also can say that outlier detection or novelty detection. The Aim of anomaly detection is to discover unusual data patterns which do not relate to normal patterns. Anomaly detection has been studied widely from last few years In past works , novelty detection in semisupervised setting is automatically solved by reducing to a binary classification problem. A detector which has desired false positive rate can be accomplished by reduction in to Neyman-Pearson classification. In contrast of inductive method, semi-supervised novelty detection (SSND) defers detectors that are optimal despite of the distribution on novelties. In novelty detection, there is a substantial impaction the theoretical properties of the decision rule of unlabeled data.

4. Malicious URL Detection:

In the Malicious URL detection this is related to how to detect malicious URLs automatically or semi-automatically, which has been extensively studied in web and data mining communities for years In general, which is divide the existing work into two categories: (i) non-machine learning methods, such as blacklisting [25] or rule-based approaches ; and(ii) machine learning methods. The nonmachine learning approaches generally suffer from poor generalization to new malicious URLs and unseen malicious patterns. In the following, this is focus on reviewing important related work using achine learning methods. In literature, a variety of machine learning schemes have been proposed for malicious URL detection, which can be grouped into two categories: (i) regular batch machine learning methods [26], and (ii) online learning methods [27]. Most of the existing malicious URL detection methods employ regular batch classification techniques to learn a classification model (classifier) from a training data set of labeled instances and then applies the model to classify a test/unseen instance. In general, the classification problem can be formulated as either binary classification (normal vs.abnormal) [26] or multi-class classification (assuming normal patterns come from multiple classes). In literature, a variety of classification techniques have been applied, such as Support Vector Machines (SVM) [26], Logistic Regression[26], maximum entropy Naive Bayes [26], and so on. However, these algorithms typically require to collect and store all the training instances in advance and build the models in a batch learning fashion, which is both time and memory inefficient and suffers from very expensive retraining cost whenever any new training data arrives.

However, most of the previous online learning algorithms were designed to optimize the classification accuracy, typically by assuming the underlying training data distribution is class balanced explicitly or implicitly. This is clearly inappropriate for online malicious URL detection tasks since the real world URL data distribution is often highly class-imbalanced, i.e, the number of malicious URLs is usually significantly smaller than the number of benign URLs on the WWW. Therefore, it is very important to take this issue into consideration when designing a machine learning and data mining algorithm for solving a practical URL detection task.Finally, all the existing learning approaches usually have to label a fairly large amount of training instances in order to build a sufficiently good classification model, which is impractical as the labeling cost is often expensive in a real word application. This thus motivates us to study a unified learning scheme, which not only is able to minimize the labeling cost, but also maximize the predictive performance with the given amount of labeled training instances.

IMPLEMENTATION DETAILS

1. System Overview:

Primary target of this paper is to add to a framework which will manage the way that every time getting real class of the example is impractical and will consider the expense of the misclassification to upgrade the classifier in the event of endure misfortune. In proposed the online dynamic learning with cost sensitivity (ODLCS) which will primary target of proposed framework, which is expressed previously. The target of directed malicious URL discovery is to manufacture a prescient model that can unequivocally predict if an approaching URL sample is noxious or not. If all else fails, this can be portrayed as a binary classification errand where malicious URL examples are from positive class ("+1") and typical URL occurrences are from negative ("-1"). For an online pernicious URL recognition responsibility, the objective is to make an online learner to incrementally assemble a arrangement model from a gathering of URL preparing information occasions by method for a online learning fashion. In particular, for every one adapting round, the learner first gets another approaching URL event for location; it then applies the classification model to anticipate in case it is malicious then again not; around the end of the adapting round, if reality class name of the sample can be revealed from the earth, the learner will make usage of the checked case to redesign the characterization model at whatever point the order is erroneous generally speaking, it is normal to apply web figuring out how to comprehend online malicious URL detection. In any case, it is unfeasible to explicitly apply a current online learning framework to settle these issues. This is by virtue of a schedule online classification undertaking typically acknowledge the class label of every approaching event will be revealed keeping in mind the end goal to be used to upgrade the classification model toward the end of every learning round. Plainly it is unfathomable or exceedingly rich if the learner queries the class name of every approaching event in an online malicious URL detection assignment. To address this test, in the proposed framework to research a novel system of ODLCS as demonstrated in Figure 1. Generally speaking, the proposed ODLCS system tries to address two key troubles in a systematic and synergic learning philosophy:

(i) the learner must choose when it ought to query the class label of an approaching URL case; likewise
(ii) how to update the classifier in the best path where there is another marked URL event.

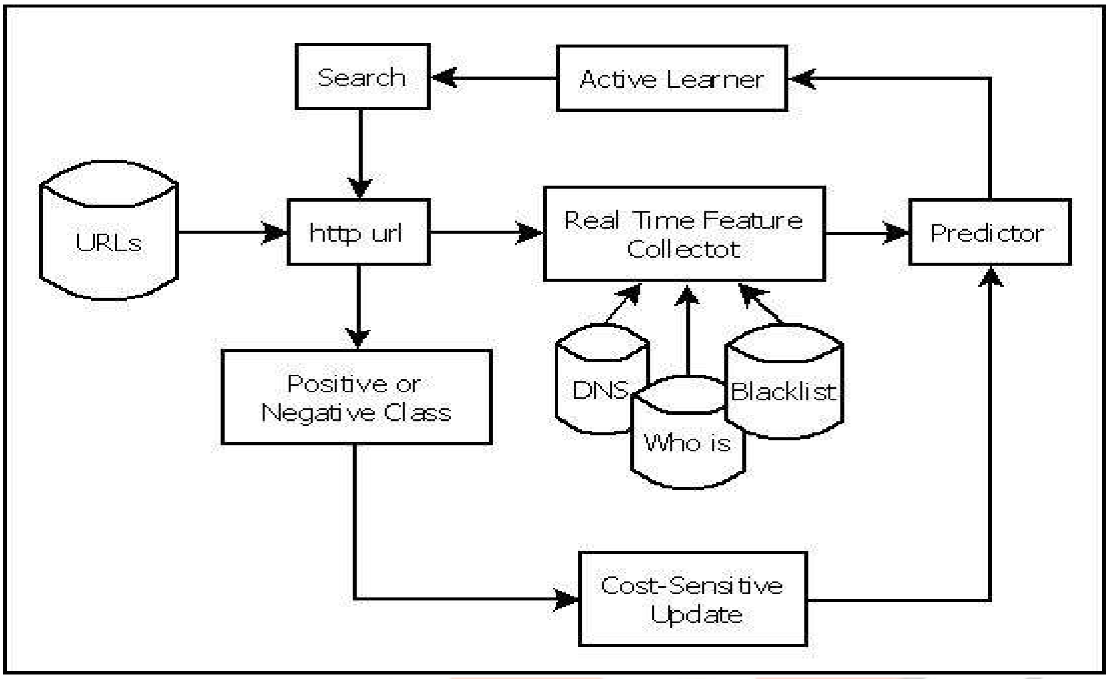

Figure1: System Architecture

2. Mathematical Model for Proposed Work:
Sensitivity = (Tp - Mp) / Tp
Specificity = (Tn - Mn) / Tn
Specificity = (TM) / T
Where, M = denote the number of mistakes
Mp = denote the number of false negatives,
Mn = denote the number of false positives
T = to denote the set of indexes of negative examples,
Tp = denote the number of positive examples,
Tn = denote the number of negative examples.
sum of weighted sensitivity and specificity:
• sum = ηp × sensitivity + ηn × specificity
Where, 0≤ ηp,ηn ≤1 and ηp+ ηn=1:
When ηp=ηn=1/2
sum is the well-known balanced accuracy.
total cost suffered by the algorithm:
• cost = cp ×Mp + cn ×Mn
Where, Mp and Mn are the number of false negatives and false positives respectively,
0≤ cp,cn ≤1 are the cost parameters for positive and negative classes, respectively
URL Detection:
F_p^b (w)=1/2‖w‖2+C_(t=1)^T lt(w)
Where
regularization parameter C > 0.
loss function lt(w).

3. Algorithm:

A) CSOGD algorithm

Step 1: INPUT: penalty parameter C, bias parameter ρ and smooth parameter δ.
Step 2: INITIALIZATION: w1 = 0.
Step 3: for t = 1, . . . , T do
Step 4: receive an incoming instance xt ∈ Rd;
Step 5: predict label ^yt = sign(pt), where pt = wt · xt;
Step 6: draw a Bernoulli random variable Zt ∈ 0, 1 of parameter$\delta/(\delta+|pt|)$; end

B) .ODLCS algorithm

Step 1: INPUT: penalty parameter, bias parameter, smooths parameter
Step 2: INITILIZATION: classifier as zero
Step 3: For every incoming instance
Step 4: receiving incoming instance
Step 5: predicting label of each instance by using classifier
Step 6: draw a Bernoulli random variable of parameter
Step 7: if a Bernoulli random variable is 1 and then suffer loss occur in instance then update classifier unless not update
Step8: end

4. Experimental Setup:

The system is built using Java framework (version jdk 6) on Windows platform. The Netbeans (version 6.9) is used as a development tool. The system doesn't require any specific hardware to run; any standard machine is capable of running the application.

RESULTS AND DISCUSSION

1. Dataset:

To examine the performance, in the proposed system test all the algorithms on a large-scale benchmark dataset for malicious URL detection, which can be downloaded from http://sysnet.ucsd.edu/projects/url/. The original data set was created in purpose to make it somehow class-balanced. In suggested system to produce a separation by sampling from the original data set to make it close to a more realistic distribution scenario where the number of normal URLs is significantly larger than the number of malicious URLs.

2. Results:

In This experiment will evaluate the performance of the proposed algorithms by varying the ratios of queries for comparing different online malicious URL detection algorithms. Figure 2 and Figure 3 shows the online average sum performance and the online average cost performance under varied query ratios, resp. From the tentative outcome, several observations can be drawn as follows.

Table I. Evaluation of The Malicious URL Detection Performance In Terms Of The Cumulative Sum Measure.

Measures	LEPE	ODLCS
Sum (%)	79.162	92.697
Sensitivity (%)	58.492	88.156
Specificity (%)	99.833	97.237
Accuracy (%)	99.419	97.146

In the following graph System compare the proposed novel class detection method with the existing method. In X axis represent different methods for comparison and Y axis the values.

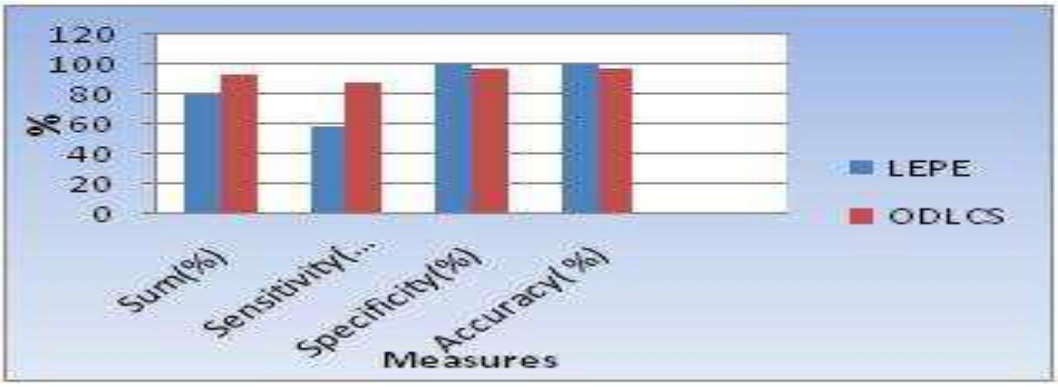

Figure.2: Evaluation of the online cumulative average sum performance with respect to varied ratios

Table II. Evaluation of the Malicious URL Detection Performance.

In the following graph System compare the proposed novel class detection method with the existing method. In X axis represent different methods for comparison and Y axis the values.

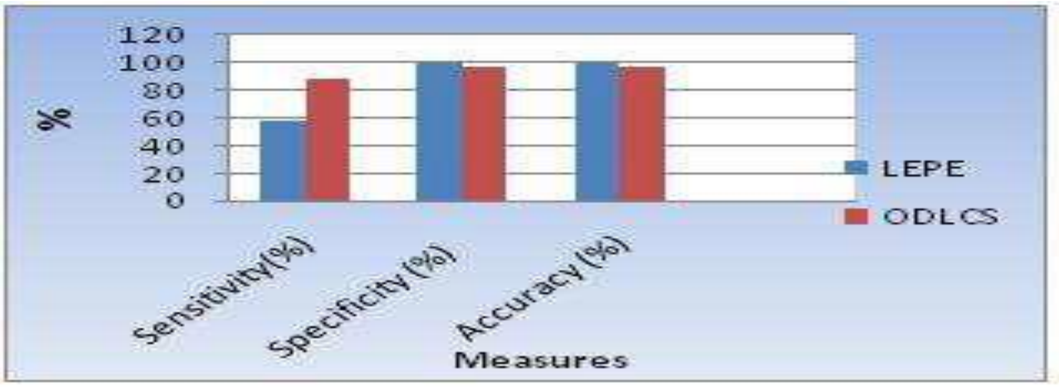

Figure 3: Evaluation of the online cumulative average Cost performance with respect to varied ratios.

Table II. Evaluation of the Malicious URL Detection Performance.

Measures	**LEPE**	**ODLCS**
Sum (%)	57.592	87.742
Sensitivity (%)	99.832	97.285

CONCLUSION

In this paper proposed a novel system of Online Dynamic Learning with Cost Sensitivity (ODLCS) to taking care of real-world applications in the classification domain like online malicious URL recognition undertaking. Paper demonstrates the ODLCS algorithms to push cost-sensitive measures and hypothetically analyze the breaking points of the proposed algorithms. in suggested structure result shows:(i) the proposed ODLCS technique has the capacity consider ablyout perform various directed cost-sensitive alternately cost-insensitive online learning algorithms for malicious URL recognition undertakings (ii)the proposed ODLCS algorithms are very proficient and adaptable for web-scale applications.