Decision Trees for Analyzing Different Versions of KDD Cup Datasets

[1]Prof. Pravin Adivarekar, [2]Vikas Kaushik, [3]Vinay Khair, [4]Tejas Zambre

[1]Asst. Professor, [2,3,4]BE Student, [1,2,3,4]Comp. Engg. Dept, SSJCET, Asangaon, India.

[1]engineerpravin2008@gmail.com, [2]er.vickzvk360@gmail.com, [3]vinay.khair95@gmail.com, [4]tejaszambre@gmail.com

Abstract - **Many Organizations release standard datasets for researchers to work. One amongst it is KDD cup dataset and its versions. Different datasets gives different theory, however this paper only talks about KDD Cup 1999 and KDD cup 2015. KDD Cup 2015 provides a dataset of user's behavior while watching online videos provided by MOOC (Massive Open Online Courses), it provides information about user's behavior – whether user is interested in a particular video or not. Such information can be analyzed so as to help an organization to suggest videos to user in which user may be more interested. Thus ultimately in a long run can save time of both user and organization and keep user with Organization. In KDD cup 1999, studies of similar dataset of network traffic and can train algorithm for various attacks. In proposed system correct classes are predicted by applying C4.5 algorithm and is compared to other algorithms using parameters of confusion matrix.**

***Keywords —C4.5, datasets, entropy, string_tokenizer, decision tree, MOOC, KDD 99, KDD 15, ID3, C4.5,* confusion matrix, KDD cup datasets, training algorithm.**

I. INTRODUCTION

Data Mining (also known as Knowledge Discovery) is a process which does analysis of pre-captured data and extracts information which may be used for business intelligence; example helps for making accurate decision support system. Knowledge Discovery in databases also referred as KDD holds the competition once in a year in which all over the world researchers can participate, and they are made available with certain datasets along with a challenge which they have to determine by using dataset[3].

1.1 KDD CUP 2015

In KDD Cup 2015 the challenge is given to participants is to predict dropouts of MOOC by using raw dataset. MOOC's are mostly made free to all the internet users. Here all the users have remote locations all over the world which make the MOOC organizations difficult to understand the user's interests in particular area. Capturing the user's behavior while watching videos by MOOC's these organizations can try to understand user's lists of interests and guide user with similar set of informative videos.[2]

This data in captured based on the click stream event of the mouse. In general, it fetches the data every time the user starts stops or restarts the video. This is time based that is, ever time user hits an event the corresponding time is captured against that event.

An Oracle database can be used here to store this data. Flow of the proposed system begins with cleaning dataset using various data cleaning algorithms further followed by using cleaned data for training classifier (algorithm). Classifier is expected to give optimal decision tree depending on the entropy of each attribute. Further decision tree will be tested using unlabelled dataset and hence we will be able to get parameters for confusion matrix.[1]

This complete procedure will be able to provide answers like - Whether the user has stopped watching particular video?

1. Is he still watching it?
2. Did he complete the entire video?
3. Number of pause while watching video?
4. Number of other URL visited while watching live video? Etc.

Answers to these kinds of questions will help any MOOC organizations to suggest videos to users in which he/she may be more interested to watch[2].

1.2 KDD CUP 1999

In KDD Cup 1999, algorithm is getting trained for different network attacks and hence will be able to predict any unknown attacks which will help in network intrusion detection system. For both the above datasets C4.5 can help to trace the decision tree on the basis of entropy[2].

II. LITERATURE SURVEY

Let's take analysis of different proposed methodologies for efficient class detection system and our proposed method for class detection. Different data mining approaches are applicable for efficient prediction of class. Various popular methods are:-

- k-means algorithm is a simple iterative method to partition a given dataset into a user specified number of clusters, k.[5]

- In today's machine learning applications, support vector machines (SVM) are considered a must try—it offers one of the most robust and accurate methods among all well-known algorithms.[7]

- One of the most popular data mining approaches is to find frequent item sets from a transaction dataset and derive association rules. Apriori is a seminal algorithm for finding frequent item sets using candidate generation.[8]

- Ensemble learning [5] deals with methods which employ multiple learners to solve a problem. The generalization ability of an ensemble is usually significantly better than that of a single learner, so ensemble methods are very attractive.

- The AdaBoost algorithm proposed by Yoav Freund and Robert Schapire is one of the most important ensemble methods. Given a set of objects, each of which belongs to a known class, and each of which has a known vector of variables, our aim is to construct a rule which will allow us to assign future objects to a class, given only the vectors of variables describing the future objects. Problems of this kind, called problems of supervised classification, are ubiquitous, and many methods for constructing such rules have been developed. [8]

- One very important one is the naiveBayes method—also called idiot's Bayes,[10] simple Bayes, and independence Bayes. This method is important for several reasons. It is very easy to construct, not needing any complicated iterative parameter estimation schemes. This means it may be readily applied to huge data sets. It is easy to interpret, so users unskilled in classifier technology can understand why it is making the classification it makes. And finally, it often does surprisingly well: it may not be the best possible classifier in any particular application, but it can usually be relied on to be robust and to do quite well.[9]

- In our system we will use C4.5, a descendant of CLS and ID3. Like CLS and ID3, C4.5 generates classifiers expressed as decision trees, but it can also construct classifiers in more comprehensible rule set form.[1][5][9]

2.1 AIM AND OBJECTIVE

Aim is to use optimal features given by such outstanding non-trivial method and to increase detection rate for accurate decisions. So in proposed system, aim is to increase correct detection of attacks and decrease false positive rate and false negative rate.

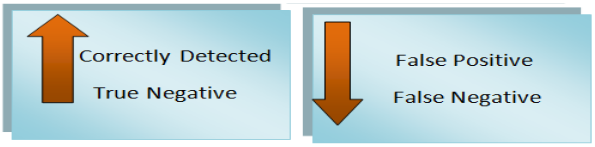


Fig 1: Parameters in confusion matrix

1. More accurate the data more accurate the result.

2. Automation using data mining is most widely used

3. Objective in proposed is to provide efficient training data to algorithm & hence get algorithm trained for different number of distinct patterns which will lead to detecting more efficient way of the unknown patterns & can support the decision

4. KDD CUP 99' expects correct detection of attacks & KDD CUP 15' expects correct detection of student dropout rate for online videos

5. Objective here is not only to have correct prediction for above two datasets but to make algorithm generalized for any version KDD CUP datasets & to contribute Decision Support System (DSS)

6. Correctly detected True negative & false positive will give us the correct prediction while testing our algorithm.

III. STAGES IN PROPOSED SYSTEM

- Extract data from various sources
- Clean data (using various cleansing algorithms)
- Use cleaned data for training algorithm
- Check decision tress
- Test decision tress using testing data
- Note confusion matrix parameter

Above steps can be shown as follows:

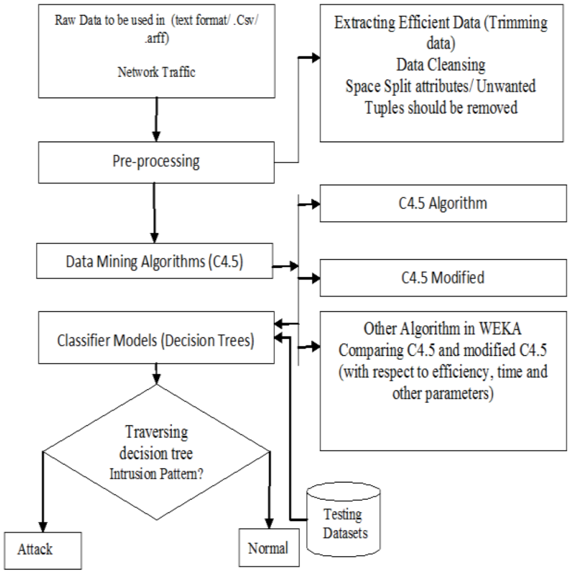

Raw Data to be used in (text format/ .Csv/ .arff)
Network Traffic
Extracting Efficient Data (Trimming data)
Data Cleansing
Space Split attributes/ Unwanted Tuples should be removed
Pre-processing
C4.5 Algorithm
Data Mining Algorithms (C4.5)
C4.5 Modified
Classifier Models (Decision Trees)
Other Algorithm in WEKA
Comparing C4.5 and modified C4.5 (with respect to efficiency, time and other parameters)
Traversing decision tree
Intrusion Pattern?
Testing Datasets
Attack
Normal

Fig 1: Proposed System

3.1 *EFFICIENT DATASET*

Decision support system provides decision which an algorithm learned from the training data.
- If training data is accurate then learning is more accurate
- Hence prime concern is to talk about how we can make training data more efficient
- There are several cleaning algorithms available for removing noisy data , unwanted values
- Entropy can also be used for finding out missing values
Example: we can learn values from remaining tuples and and predict value for unknown tuple.

Data used for training is the vital data and hence its cleaning should be done properly. More accurate the data more accurate the training and more efficient testing can be obtained.

3.2 *PREPROCESSING DATASET*

In below dataset (KDD cup 1999) we may not need to teach all types of attacks to algorithm. There are total 24 attacks. If we use the entire set of data then algorithm will consume much long time to generate the decision tree and if we are not interested for all 24 attacks then it is not worth to use such big dataset. Hence following methods can be used to to find the desired data.

3.2.1DATA TRIMMING

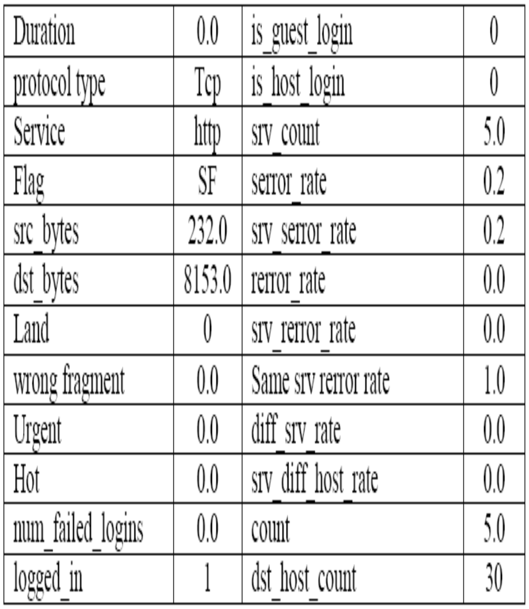
In data trimming we horizontally fragment the data for required classes and then we shuffle all tuples . we should make sure that training data should not be baised to one class. It is expected to have variety of data in training set

Training set: Training set comprises of all columns with label

Example: KDD cup 2015 dataset **log_train: :**Columns: enrollment_id, tstamp, source, logged_event, object, dropped_out.
Log_test: Columns: enrollment_id,tstamp, source, logged_event, object.
Enrollment_id: Columns: enrollment_id, username, course_id.
Object:Columns:course_id, module_id, children, start.
Enrollment_train:Columns:enrollment_id, username, course_id.

Below is the set of columns in KDD cup 1999 dataset with label class.

Table 1: KDD CUP 1999 dataset



Duration	0.0	is_guest_login	0
protocol_type	Tcp	is_host_login	0
Service	http	srv_count	5.0
Flag	SF	serror_rate	0.2
src_bytes	232.0	srv_serror_rate	0.2
dst_bytes	8153.0	rerror_rate	0.0
Land	0	srv_rerror_rate	0.0
wrong_fragment	0.0	Same_srv_rerror_rate	1.0
Urgent	0.0	diff_srv_rate	0.0
Hot	0.0	srv_diff_host_rate	0.0
num_failed_logins	0.0	count	5.0
logged_in	1	dst_host_count	30

IV. ALGORITHMS IMPLEMENTATION

4.1 DECISION TREE ALGORITHMS:

```
Algorithm (C4.5) [7]
→Step (i)
 Read data from file
   Store the data in array of Vector
  Domains[i]← [i]
  //domains[i] is an array of vector (Vector for every attribute)
Step (ii)
Decompose Node
For (i=0; i< numattributes; i++)
(
No. of distinct values of particular column (num values) = domain[i].size
//[num value = duration (distinct values)]
    For (j=0; i< numvalues; j++)
        If node is already used then continue another node
        getsubset of the first distinct value (j)(getrows)
        subset[]← j (no.of rows having value = j)
Calculate Entropy of each subset then
take average and find total entropy of that attribute.
Entropy of a decision=
$-P_1 \times \log P_1 - P_2 \times \log P_2 - \cdots - P_n \times \log P_n$

Where P1, P2, …, Pn are the probabilities of the n possible outcomes.
        Note logarithm is to the base 2.

        If entropy < best entropy
        Then
        (best entropy = entropy)
If Information gain of that attributes > all other attributes
Then that respective attribute is root attribute
 Decision will be taken with respect to its (best attribute) values.
```

4.2 PROPOSED ALGORITHM: EXTENSION (MORE PRUNED C4.5)

```
Step I
Read data from file
Store the data in array of Vector
domains[i]← [i]
//domains[i] is an array of vector (Vector for every attribute)
Step II
Decompose Node [09, 11, 15]
For (i=0; i< numattributes; i++)
(
No. of distinct values of particular column (num values) = domain[i].size
//[num value = duration (distinct values)]
    For (j=0; i< numvalues; j++)
        If node is already used then continue another node
        getsubset of the first distinct value (j)(getrows)
        subset[ ]← j (no.of rows having value = j)
        compliment[ ]← all the remaining rows (!= j)
        calculate entropy of the subset & compliment
        calculate entropy:
        Entropy of a decision =
$-P_1 \times \log P_1 - P_2 \times \log P_2 - \cdots - P_n \times \log P_n$

Where P1, P2, …, Pn are the probabilities of the n possible outcomes.
        Note logarithm is to the base 2.
        If entropy < best entropy
        Then
        (best entropy = entropy)
        Selected attribute = i→ attribute
        Selected value= j→ value
    )
)

Step III
Root node:
Assign the selected attribute & selected value to decomposition attribute and decompostion value respectively of root node
Root node decomposition
Attribute← i
Value←  j
Allocate two child nodes to the root node
Store the best attribute
Select rows for a single distinct best values in thr 1st child node
And remaining compliment distinct value rows in 2nd child node
Repeat decompose node for 1st child, repeat decompose node for 2nd child
Finally binary tree is generated…..finish.
```

→Flow of making binary tree can be shown as follows and it is explained in detailed in algorithm shown in Fig. 5.2: Tree will be starting with the root node; root node is the node giving highest information gain. Highest information gain = minimum entropy then as the procedure explain in algorithm 5.2 binary tree will be generated.

V. RESULTS AND DISCUSSION

This paper concentrates on only training part. For training we have fragmented datasets in two parts. Traffic 1 with more no of instances and traffic 2 with less no of instances. Below table shows how above mentioned algorithms work on it.

Table 2: time consumed to generate decision tree

	C4.5	**Pruned C4.5**
Traffic / dataset 1 → time More number of instances	Less time	More time
Traffic / dataset 2 → time Less number of instances	0 sec	0 sec
Decision tree		
Traffic / dataset 1 → time	Works only for one best attribute	Gives combination of multiple attributes on the basis of which decision is predicted
Traffic / dataset 2 → time		

VI. CONCLUSIONS

It is been observed that when we increase the size of the dataset more time gets consumed to generate decision tree by pruned C4.5 algorithm.

But pruned C4.5 algorithm takes decision by making permutation and combination of multiple attributes together and hence it is expected that during testing it gives more efficient results.

REFERENCES

[1] James Cannady Jay Harrell —A Comparative Analysis of Current Intrusion Detection Technologies.

[2] Mrudula Gudadhe, Prakash Prasad, Kapil Wankhade, Lecturer, —a new data mining based network intrusion detection model‖iccct'10.

[3] N.s.chandolikar & v.d.nandavadekar,International Journal of Computer Science and Engineering (IJCSE),Vol.1, Issue 1 Aug 2012 81-88 —comparative analysis of two algorithms for intrusion attack classification using KDD cup dataset.

[4] Dai Hong Li Haibo,2009 15th IEEE Pacific Rim International Symposium on Dependable Computing, A Lightweight Network Intrusion Detection Model Based on Feature Selection.

[5] Bhavani Thuraisingham, Latifur Khan, Mohammad M. Masud, Kevin W.Hamlen The University of Texas at Dallas 2008 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing,‖ Data Mining for Security Applications‖

[6] Radhika Goel, Anjali Sardana, and Ramesh C. Joshi —Parallel Misuse andAnomaly Detection Model‖ International Journal of Network Security, Vol.14, No.4, PP.211-222, July 2012

[7] Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu, and Ali A. GhorbaniA Detailed Analysis of the KDD CUP 99 Data Set.

[8] Thales Sehn Korting,‖ C4.5 algorithm and Multivariate Decision Trees.

[9] P Amudha, H Abdul Rauf, —Performance Analysis of Data Mining Approaches in Intrusion Detection‖

[10] Knowl Inf Syst (2008) 14:1–37 DOI 10.1007/s10115-007-0114-2 SURVEYPAPER, Top 10 algorithms in data mining XindongWu · Vipin Kumar · J.Ross Quinlan · Joydeep Ghosh · Qiang Yang ·Hiroshi Motoda · Geoffrey J. McLachlan · Angus Ng · Bing Liu · Philip S. Yu ·Zhi-Hua Zhou · Michael Steinbach · David J. Hand · Dan Steinberg

[11] Mrutyunjaya Panda1 and Manas Ranjan Patra2, IJCSNS International Journal of Computer Science and Network Security,VOL.7 No.12, December 2007‖ Network Intrusion Detection Using Naïve Bayes‖

[12] Shaik Akbar, Dr.K.Nageswara Rao, Dr.J.A.Chandulal, International Journal of Computer Applications (0975 –8887) Intrusion Detection System Methodologies Based on Data Analysis

[13] Nathan Einwechter, —An Introduction To Distributed Intrusion Detection Systems‖

[14] Payam Emami Khoonsari and AhmadReza Motie, —A Comparison of Efficiency and Robustness of ID3 and C4.5,Algorithms Using Dynamic Test and Training Data Sets‖, International Journal of Machine Learning and Computing, Vol. 2, No. 5, October 2012

[15] http://KDD.ics.uci.edu/databases/KDDcup99/task.html

[16] http://nsl.cs.unb.ca/NSL-KDD/