Scientific Journal Impact Factor (SJIF): 3.518

International Journal of Modern Trends in Engineering and Research

IJMTER

ISSN (ONLINE) : 2349 - 9745

ISSN (PRINT) : 2393 - 8161

Anomaly Detection using Decision Tree based Classifiers

Deepak Sinwar[1]and Manish Kumar[2]

[1]*Department of Computer Science & Engineering, BRCM College of Engg. & Tech., Bahal*
[2]*Department of Computer Science, NIMS University, Jaipur*

Abstract— as we know that with the help of Data mining techniques we can find out knowledge in terms of various characteristics and patterns. In this regard this paper presents finding out of anomalies/ outliers using various decision tree based classifiers viz. Best-first Decision Tree, Functional Tree, Logistic Model Tree, J48 and Random Forest decision tree. Three real world datasets has been used in this study. Theoretical analysis and experimental results shown that the Random Forest decision tree has outperformed other decision tree based classifiers of this study in terms of correct classification rate and kappa statistic.

Keywords- anomaly, outlier, decision tree, classification

I. INTRODUCTION

Anomaly detection is becoming a critical issue now days. There are wider variety of anomaly detection ranging from fraud detection in financial transactions, faulty node detection in computer networks and many more. In decision support systems, we have to be very careful during decision making process; but before proceeding for decision making we should know about the anomalies present in our databases. There may chances that our data may suffer from various kind of anomalies such as missing entries, incorrect entries etc. There may be lots of reasons behind the presence of anomalies in datasets i.e., typing mistakes, improper working or malfunctioning of any component of a machine, erroneous transactions, faulty machines, lack of proper environment during reading taken process of any medical diagnosis etc. Plenty of anomaly detection techniques exist in research literature, but we have to ensure which technique will gives us better results. The performance of anomaly detection algorithms may vary according to various situations i.e., domain of anomaly detection, size of datasets, type of datasets etc. During the layout of this paper we may use the terms interchangeably such as ‘anomaly’ and ‘outlier’; ‘databases’ and ‘datasets’; and ‘data sets’ and ‘datasets’. Decision trees are one of the most powerful techniques of data mining which enable us to find out proper classification of our data items. Han et al. [2], classified data mining techniques into three categories viz. Association Rule Mining, Clustering and Classification/ Prediction. Classification is the method of discovering set of models that describes/ predicts the class of objects whose class is unknown. Classification helps in predicting the class of a unknown tuple by matching the characteristics from the testing database with training database; this overall process is sometimes known as pattern matching. There is one another term used sometimes with classification i.e., prediction. The major difference is that, classification predicts a categorical value, while regression is applied when prediction is required on real-valued or numeric domain. As we are aware that Decision trees are helpful in making effective business decisions due to their predictability properties, that’s why they are categorized in classification algorithms of data mining. In this paper, we have tested five decision tree based classifiers viz. ‘*Best-First Tree*’, ‘*Functional-Tree*’, ‘*Logistic Model Tree*’, *J48* and ‘*Random Forest*’ on three real world datasets obtained from UCI machine learning repository [3].

The rest of the paper is organized as follows, Section-II describes some of the work related with anomaly detection, whereas Section-III elaborates the experimental work with brief discussions; finally conclusions are made in Section-IV.

II. BACKGROUND WORK

Machine learning techniques have been used by researchers to address the problem of anomaly/ outlier detection. Most of the work in the area of anomaly detection has been done using clustering techniques; but in this paper we will review some of the wok related with anomaly detection using classification as well as clustering. Let us review anomaly detection using both techniques as follows:

As mentioned above, most of the work in the area of outlier mining has been done using various clustering techniques. Starting from the very famous algorithm of the decade i.e., BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) by Zhang et al. [6], many other clustering algorithms have been proposed in the research literature. Another method of same kind was proposed by Yogita and Toshniwal [4]. They proposed a clustering based framework for outlier detection in evolving data streams that assigns weights to attributes depending upon their respective relevance. Weighted attributes are helpful to reduce or remove the effect of noisy attributes in mining tasks. Keeping in view the challenges of data stream mining, the proposed framework is incremental and adaptive to concept evolution. On the other hand Carmelo Cassisi et al. [5] presented the enhancements in density based clustering. The method was based on the concept of space stratification, efficiently identifies the different densities in the dataset and, accordingly, ranks the objects of the original space. Next, it exploits such knowledge by projecting the original data into a space with one more dimension. It performs a density based clustering taking into account the reverse-nearest-neighbor of the objects. Their method also reduces the number of input parameters by giving a guideline to set them in a suitable way. Whereas to detect outliers from industrial data Cateni et al. the use of artificial intelligence techniques has been proposed in [7]. Noha A. Yousri et al. [8] proposed a fuzzy approach for integrating results from an outlier detection method and a clustering algorithm. A universal set of clusters is proposed which combines clusters obtained from clustering, and a virtual cluster for the outliers. This approach has two phases; the first computes patterns' initial memberships for the outlier cluster, and the second calculates memberships for the universal clusters, using an iterative membership propagation technique. This approach was general and can combine any outlier detection method with any clustering algorithm.

Another outlier mining approach related with Wireless Sensor Networks was presented by Fuzzy et al. [9]. They proposed a novel in-network knowledge discovery approach that provides outlier detection and data clustering simultaneously. Their approach is capable to distinguish between an error due to faulty sensor and an error due to an event (probably an environmental event) which characterize the spatial and temporal correlations between events observed by sensor nodes in a confined network neighborhood. Experiments on both synthetic and real datasets show that the proposed algorithm outperforms other techniques in both effectiveness and efficiency. On the other hand Xu et al. [10] proposed a hierarchical anomaly detection framework to overcome the challenges of anomaly detection in WSNs. They aim to detect anomalies by the accurate model and the approximated model learned at the remote server and sink nodes, respectively. Besides the framework, they also proposed an approximated local outlier factor algorithm, which can be learned at the sink nodes. Various other approaches [1, 12-18] for finding out anomalies using clustering and other techniques are also exist in research literature.

III. EXPERIMENTAL WORK

In order to evaluate the performances of decision trees (mentioned in Section-I), we performed classification on three real world datasets viz. Horse-colic, Glass and Sonar. All the three datasets has been obtained from UCI machine learning repository [3]. Five different parameters (correct classification rate, incorrect classification rate, kappa statistic, mean absolute error and time taken to build model) have been selected for the overall evaluation.

Waikato Environment for Knowledge Acquisition version 3.7 is being used as simulation tool. The first dataset (*horse-colic*) containes 368 instances and 23 attributes (7 numeric and rest are of nominal type); whereas the second dataset (*Glass*) contains a total of 214 instances and 10 attributes (all are of numeric type except the class attribute, which is of nominal type). The third dataset (*sonar*) contains a total of 208 instances and 61 attributes (all are of numeric type except the class attribute, which is generally of nominal type). The default configuration of WEKA has been used in all the experiments except the 'Test options' option; in the 'Test options' we have selected the first option i.e. 'Use training set' instead of 10-folds cross validation. Before discussing the experimental work, let us discuss in brief about the five decision tree based classifiers (obtained from [11]) which have been used in this study.

Best First Tree (BFT) [19] uses binary split for both nominal and numeric attributes. Haijian Shi first introduces the algorithm for building binary best-first decision trees for classification problems; then an investigation on two new pruning methods which determines an appropriate tree size by combining best-first decision tree growth with cross-validation-based selection of the number of expansions that are performed. The Functional Tree (FT) [20] is a Classifier that could have logistic regression functions at the inner nodes and/ or leaves. The algorithm can handle binary and multi-class target variables, numeric and nominal attributes and missing values. In this work, a unified framework and multivariate models for various types of classification and regression problems with in-depth study of the behavior of functional trees has been proposed. On the other hand J48 [21] has been used which is basically a class of generating pruned or unpruned C4.5 decision tree. The Logistic Model Tree (LMT) [22] is a classification model with an associated supervised training algorithm that combines logistic regression (LR) and decision tree learning. Their experiments show that LMT produces more accurate classification than C4.5, CART, logistic regression, model trees, functional trees, naive Bayes trees and Lotus. Whereas Breiman [23] developed Random Forests (RF), which are a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. The detailed discussion about these five algorithms has been provided in [19-23].

Description of Experimental Results: As mentioned above we have used five different algorithms of decision trees on three real world datasets. Table-1 shows the summary of experiments carried out on these algorithms. In first dataset, we found that RF has classified 99 % of the instances correctly with 0.98 as kappa value. The kappa values lie between 0 and 1. The 1 value of kappa means all the instances of a dataset has been classified correctly without any errors as given in the training dataset. Whereas other parameters i.e. incorrect classification rate, mean absolute error are associated with the correct classification and kappa value. If the correct classification rate is higher mean the less the errors and the less the incorrect classification rate. On the other hand the time parameter gives us the speed of execution of algorithm in building the classification model.

Table 1: Summary of various experiments carried out by 5 Decision Tree based classifiers on 3 real world datasets

Colic Dataset					
Parameters	BFT	FT	J48	LMT	RF
Correct Classification Rate	98.64	93.21	85.87	90.76	***99.18***
Incorrect Classification Rate	1.36	6.79	14.13	9.24	***0.82***
Kappa Statistic	0.97	0.85	0.68	0.80	***0.98***
Mean Absolute Error	***0.05***	0.10	0.24	0.14	0.10
Time Taken to Build Model (in sec.)	0.41	0.47	***0.03***	4.71	0.12
Glass Dataset					

Correct Classification Rate	76.64	80.37	96.26	98.60	***99.53***
Incorrect Classification Rate	23.36	19.63	3.74	1.40	***0.47***
Kappa Statistic	0.67	0.74	0.95	0.98	***0.99***
Mean Absolute Error	0.10	0.07	0.02	***0.01***	0.03
Time Taken to Build Model (in sec.)	0.06	0.31	***0.03***	3.74	0.06
Sonar Dataset					
Correct Classification Rate	92.79	93.75	98.08	88.46	***100.00***
Incorrect Classification Rate	7.21	6.25	1.92	11.54	***0.00***
Kappa Statistic	0.86	0.87	0.96	0.77	***1.00***
Mean Absolute Error	0.13	0.13	0.03	0.19	***0.07***
Time Taken to Build Model (in sec.)	0.23	***0.20***	0.09	5.94	0.06

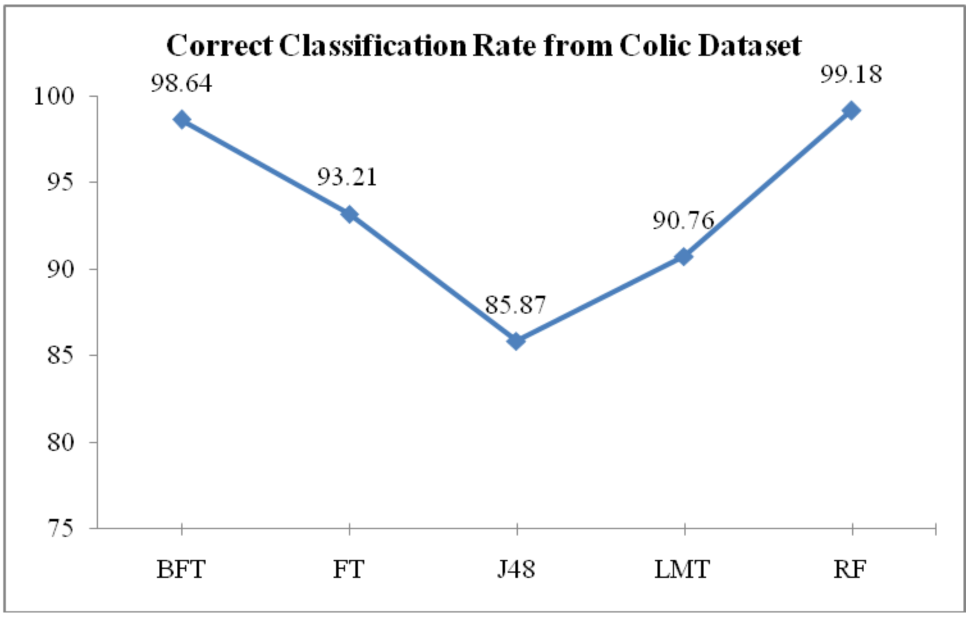

Fig. 1: The correct classificatio rate of various decion tree based classfiers on *‘colic’* dataset

From first dataset we found that J48 has taken least time in building the classification model, whereas BFT has the least error value. But if we talk about the second dataset, the less number of errors has been produced in case of LMT algorithm, the values of other parameters in this experiment are same as of first dataset. Accordingly in third dataset, the same lead has been taken by RF algorithm except the time taken as shown in Table-1.

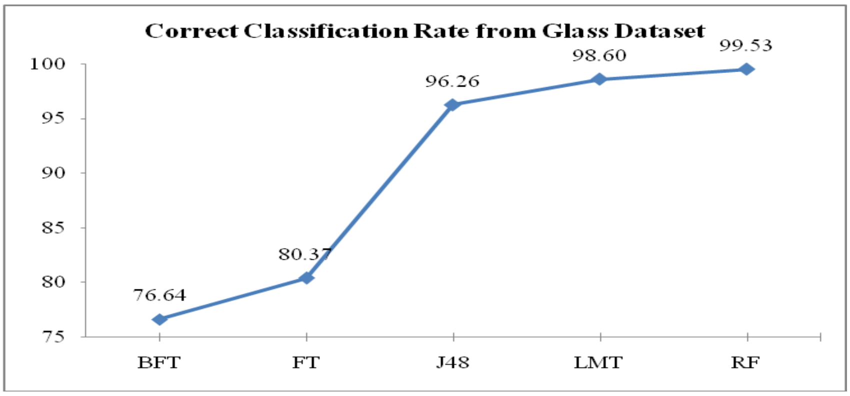

Fig. 2: The correct classificatio rate of various decion tree based classfiers on *‘Glass’* dataset

The correct classfication rates of various decision tree based classifiers from colic, glass and sonar datasets is shown in Figures 1,2 and 3 respectively. Whereas Figure-4 represents the average number of outliers discovered during all experiments. After viewing all the results we can say that the performance of RF algorithm is better as compared to other decions tree based classifiers in terms of correct classification rate and kappa statistic; whereas J48 is able to build the model in less number of time than other algorithms.

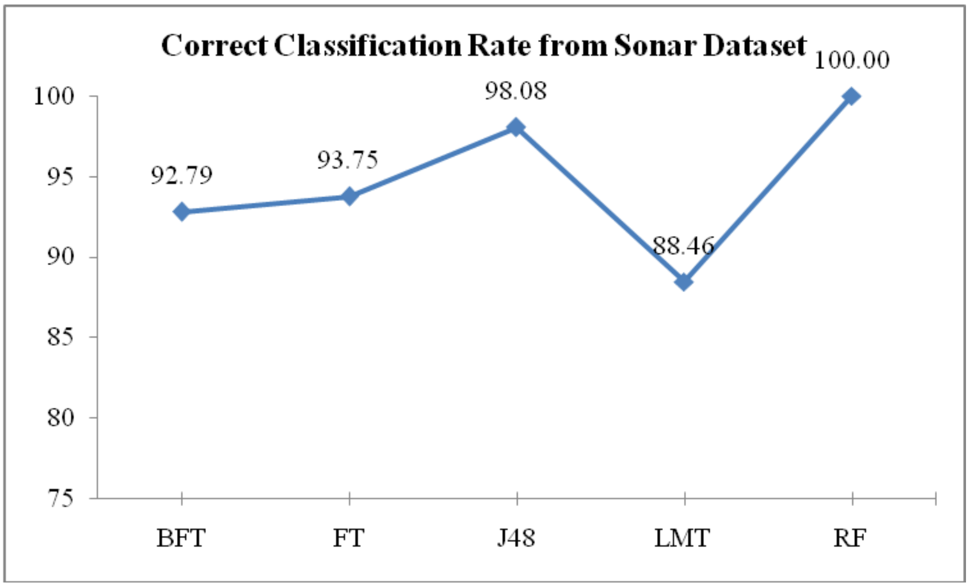

Fig. 3: The correct classificatio rate of various decion tree based classfiers on *‘sonar’* dataset

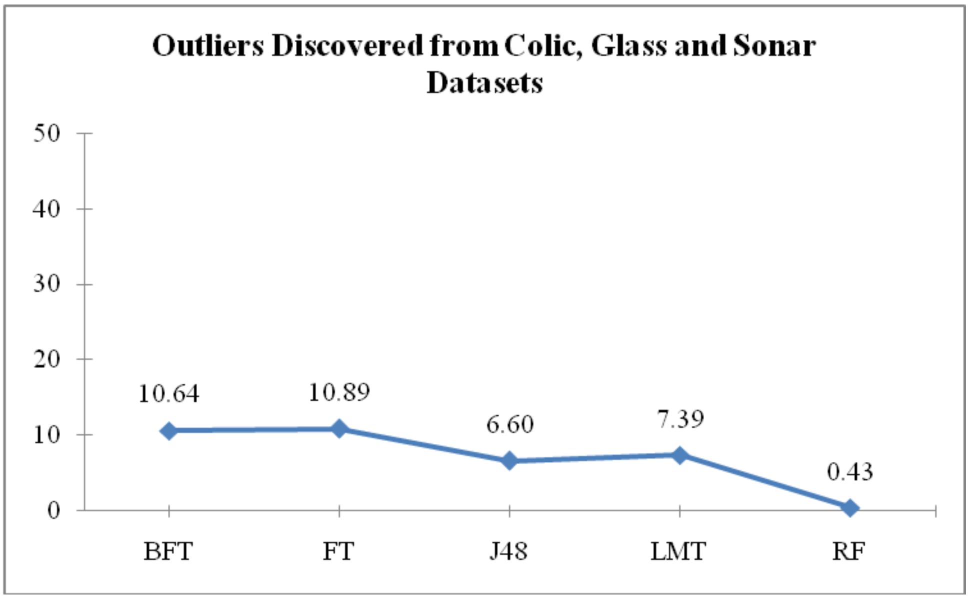

Fig. 4: The average number of outliers discovered from colic, glass and sonar datasets using various decision tree based cassifiers

IV. CONCLUSION

The paper proposes an evaluation of five decision tree based algorithms (BFT, FT, J48, LMT and RF) on three real world datasets (colic, glass and sonar). Theoretical analysis and experimental work shown that Random Forest (RF) decision tree has outperformed other decision tree based classifiers (in this evaluation) in terms of correct classification; whereas J48 is able to build the classification model in lesser time than other algorithms. In future, we may extend this work with the incorporation of some other machine learning algorithms with larger datasets.

REFERENCES

[1] Rahmani A., Afra S., Zarour O., Addam O., Koochakzadeh N., Kianmehr K., Alhajj R. and Rokne J., "Graph-based approach for outlier detection in sequential data and its application on stock market and weather data", Knowledge-Based Systems, Vol. 61, pp. 89-97, 2014

[2] Han J. and Kamber M., Data Mining: Concepts and Techniques, Morgan Kaufmann Publisher, San Francisco, USA, 2001.

[3] UCI Machine Learning Repository, http://archive.ics.uci.edu

[4] Yogita and Toshniwal D., "A Framework for Outlier Detection in Evolving Data Streams by Weighting Attributes in Clustering", 2nd International Conference on Communication, Computing & Security (ICCCS-2012), Procedia Technology, Vol. 6, pp. 214-22, 2012

[5] Cassisi C., Ferro A., Giugno R., Pigola G. and Pulvirenti A., "Enhancing density-based clustering: Parameter reduction and outlier detection", Information Systems, Vol. 38, p. 317-330, 2013

[6] Zhang T., Ramakrishnan R. and Livny M., "BIRCH: A New Data Clustering Algorithm and Its Applications", Data Mining and Knowledge Discovery, Vol. 1, pp. 141-182, 1997

[7] Cateni S., Colla V. and Vannucci M., "Outlier Detection Methods for Industrial Applications", Advances in Robotics, Automation and Control, Jesus Aramburo and Antonio Ramirez Trevino (Ed.), ISBN: 978-953-7619-16-9, InTech, Available from: http://www.intechopen.com/books/advances_in_robotics_automation_and_control/outlier_detection_methods_ for_industrial_applications, 2008

[8] Yousri N. A., Ismail M. A. and Kamel M. S., "Fuzzy Outlier Analysis A Combined Clustering - Outlier Detection Approach", IEEE International Conference on Systems, Man and Cybernetics, pp. 412-18, 2007

[9] Fawzy A., Mokhtar H.M.O. and Hegazy O., "Outliers detection and classification in wireless sensor networks", Egyptian Informatics Journal, Vol. 14, pp. 157-164, 2013

[10] Xu L., Yeh Y.R., Lee Y.J. and Li J., "A Hierarchical Framework Using Approximated Local Outlier Factor for Efficient Anomaly Detection", Procedia Computer Science, Vol. 19, pp. 1174 -1181, 2013

[11] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten (2009); The WEKA Data Mining Software: An Update; SIGKDD Explorations, Vol. 11, Issue 1.

[12] Hassanzadeh R. and Nayak R., "A semi-supervised graph-based algorithm for detecting outliers in online-social-networks", in Proceedings of the 28th Annual ACM Symposium on Applied Computing (SAC '13), ACM, New York, NY, USA, pp. 577-582, 2013.

[13] Daneshpazhouh A. and Sami A., "Entropy-based outlier detection using semi-supervised approach with few positive examples", Pattern Recognition Letters, Vol. 49, pp. 77–84, 2014

[14] Angiulli F. and Fassetti F., "Exploiting domain knowledge to detect outliers", Data Mining & Knowledge Discovery, Vol. 28, Issue 2, pp. 519-568, 2014

[15] Zhang J., Tao X. and Wang H., "Outlier detection from large distributed databases", World Wide Web, Vol. 17, Issue 4, pp. 539-568, 2014

[16] Sevakula R. K., "Clustering based outlier detection in fuzzy SVM", IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), ISBN: 978-1-4799-2073-0, pp. 1172-77, 2014

[17] Ju F., Sun Y., Gao J., Hu Y., and Yin B., "Image Outlier Detection and Feature Extraction via L1-Norm-Based 2D Probabilistic PCA", IEEE Transactions on Image Processing, Vol. 24, Issue 12, pp. 1057-7149, 2015

[18] Bouguessa M., "Clustering categorical data in projected spaces", Data Mining & Knowledge Discovery, Vol. 29, Issue 1, pp. 3-38, 2015

[19] Shi H., "Best-first Decision Tree Learning", Master of Science Thesis at The University of Waikato, 2006

[20] Gama J. , "Functional Trees", Machine Learning, Vol. 55, pp. 219–250, 2004

[21] Quinlan R. (1993), "C4.5: Programs for Machine Learning", Machine Learning, Vol. 16, pp. 235-240, 1994

[22] Landwehr N., Hall M. and Frank E., "Logistic Model Trees", Machine Learning Vol. 59, 161, doi:10.1007/s10994-005-0466-3

[23] Breiman L. "Random Forests", Machine Learning, Vol. 45, 1, pp. 5-32, 2001