International Journal of Database Theory and Application
Vol. 11, No.1 (2018), pp. 1-8
hhtp://dx.doi.org/10.14257/ijdta.2018.11.1.01

Decision Tree Algorithms C4.5 and C5.0 in Data Mining: A Review

Muhammad Arif[1]

[1]*Faculty of Computer Science and Information Technology, University of Malaya 50603 Kuala Lumpur, Malaysia*

Abstract

The purpose of this paper is to briefly define and describe the key concept of Machine Learning Decision Tree having a comparison between C4.5 and C5 and its usage in Data Mining. Decision Tree uses different Algorithms for the Classification of data in Data Mining. Out of these many, we have taken two, C4.5 and C5 for our review. We have taken the data from different well known references, after a deep search and study process, compile a good comparison between these two Algorithms in different aspects.

Keywords: *Data Mining, Machine Learning, Classification Algorithms, Data Disorder*

1. Introduction

A decision tree gives a set of rules to divide data in different groups, to make any kind of decision on them. These rules are applied to data in Data Mining and data warehousing [2, 5, 11, 12, 13] in a specific order. Rule 1 breaks the data in a number of pieces. On this divided data set, Rule 2 is applied to further classify it. As we go further, following these rules, the data becomes more and more refined. Thus, the above mentioned procedure gives us a classified data, depending on which one can make decision.

The decision tree, as clear from its name, follows the structure of a tree, but it is drawn upside down. Root is at the top of tree. Then it gets divided into branches, after applying Rule 1. The branches have leaves at their ends. It is quite possible that after applying the first rule, a leave may get into its final shape and is not divided further. The process continues until the whole data set is drawn in a tree form. All the leave nodes are the decisions with respect to some target measure.

By this point, it is very clear that decision tree is used to make some sort of prediction. But it is not that simple as one might expect. It includes a number of steps before reaching a point where some kind of prediction can be made [14,15,16]. There are a lot of hindrances in the process. We will not get into the detail of those; some of them are, Variable selection, Variable importance, Interaction detection, Stratified modeling, missing value imputation, model interpretation and Predictive modeling.

There are a number of Algorithms for implementing decision tree, *e.g.*, AID, SEARCH, CHAID and so on but we will discuss here C4.5 [9] and C5 which were presented by Ross Quinlan, from Stanford University. ID3 was also written by Ross. ID3 and C4.5 remained the primary algorithms for AI and Machine Learning [8, 10,17,18]. An example of might be the simplest problem is shown in figure 1. Here we take the example of a data set belongs to the size to trousers. If the size K is smaller than 20, it will move to the Left Hand Side and gives the result as, 'tiny'. If the size is greater than 20, then it will move downward in the tree on Right Hand Side. Now, if the size is less than 80, it gives 'normal' by moving to Left Hand Side of the tree, otherwise it will give us 'Large' by moving on the Right Hand Side. We observed that the value was 40, which is greater than 20 but less than 80, so the result is 'Normal'. So, we can say that decision

Received (March 19, 2017), Review Result (May 15, 2017), Accepted (March 8, 2018)

ISSN: 2005-4270 IJDTA

tree is a fast mechanism of learning, gives quick results on a data set and is easy to implement. But the limitation is that, due to the use of one variable at a time, it results in limited types of trees. So, to handle large trees, it requires batch method.

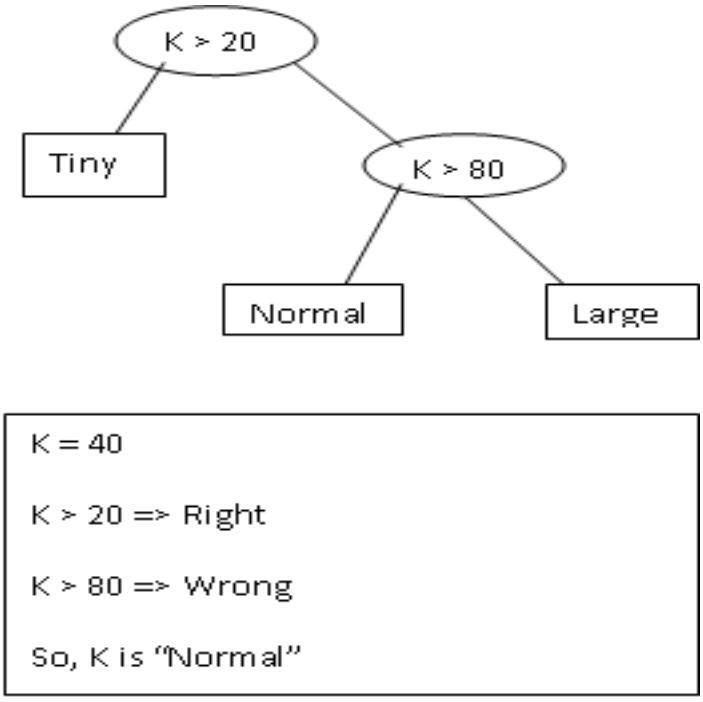

Figure 1. A Simple Classification Process

2. Related Work

It was shown in [1] that all P2P programs Has a similar behavior, so statistical analysis may be used To identify unknown apps even. Several attempts were made To categorize accurate P2P and Skype traffic using the older one Implement MLAs, such as REPTree, C4.5 or J48. that in [1,18], the authors proposed a few simple algorithms based on REPTree and C4.5 that can categorize P2P traffic Using the first five packets of a stream. Their method is based on C4.5 Performed very accurately (97% of P2P traffic was classified True), but at the start, the accuracy was not tested The packet was lost. In addition, the set of features used to classify the source and destination port numbers, which could link the classifier to a close relationship with assigning port numbers to specific programs in educational data. Another approach to categorizing P2P applications in [3] was to use the Java C4.5 implementation called J48 to distinguish between 5 different applications. The authors tried to reject a number of packets at the beginning of the 10 to 1000 queues, with only a slight change in performance, they obtained a classification accuracy of over 96%. It has been shown in [10,17,18] that the original C4.5 and J48 are very different in relatively small and noisy collections (the accuracy of J48 and C5.0 is similar in cases tested and worse than C4.5). J48 processing using statistics based on. To measure BitTorrent and FTP traffic, measurements were performed in [11.17] and reached an accuracy of 98%. This publication showed that the behavior of the data parameters contained in encrypted and non-encrypted traffic generated by the same The program is almost the same. Additionally, it has been shown that ACKs can distort statistics by size. In [12], many different mechanisms for classifying network traffic were investigated, including C5.0. The accuracy obtained for traffic belonging to 14 different functional classes was about 88-97%. This classification accuracy was very high, which was partly due to the provision of educational and experimental records in which the decision attribute (program name) was obtained by the

DPI (PACE, OpenDPI and L7-filter). These DPI solutions use multiple algorithms (including statistical analysis) to obtain the application name. Therefore, both training and testing data were somewhat flawed, resulting in more errors than C5.0.

3. C4.5 Algorithm

In this section, we will describe a well-known decision tree algorithm, *i.e.*, C4.5. It is based on the classical ID3 algorithm, which usually tries to find easy and trouble free decision trees. Some of the guidelines for this algorithm are,

- If all the cases taken belong to a single class, the tree will limit to a leaf, which will in result be named after that class.
- For each element, the provided information will be calculated applying a test on that element. The gain from information will also be calculated resulting from the test upon a particular element or attribute.
- After applying the above characteristics, find the best element to the appropriate branch after the selection.

3.1. Test Criteria

According to maximum learning systems, the shorter the tree, the higher would be the predictiveness. But in general, it is too difficult to assure the minimalistic tree. For this, C4.5 depends upon greedy search, which gives surety of a particular test selection that gives maximum heuristic splitting criteria [19].

The Algorithm exercise two criteria for this, first is the information [4] gain and the second one is gain ratio. Let RF be the relative frequency of cases in set M which belongs to class Ki, makes it, RF(Ki,M). The information content of message that describes the class of a case in M will be,

$$C(M)= -\sum RF\ (Ki,M)\ \log\ (RF(Ki,M))$$

After M is broken into M1, M2, M3……. By test T, information gain will be,

$$G(M,T)= C(M) - \sum |Mj|/|M|\ C\ (Mj)$$

So, the choosen test by gain criteria is T which capitalize on G(M,T). There is a limitation of this criteria and that is, it chooses the test that have too many consequences. The gain ratio resolves this issue as well, because it takes the possible information from division itself:

$$P(M,T) = -\sum |Mi|/|M|\ \log\ |Mi|/|M|$$

By this, we can address the problems of those data sets which have unknown occurrences or are not well defined according to a particular case. Another problem arrives when there is a condition or occurrence upon which no rule applies. This type of problem is resolved using metaconditions where no other rule applies.

3.2. Missing Values

It is very common that in data, there is some ratio of missing values, and it occasionally becomes very significant as well. This can be due to different reasons. For example, it can be due to the incorrect recording of the data or some real values were thought inappropriate to a particular problem. This problem can also be resolved using different techniques in C4.5.

The above problem is also defined comprehensively in the concept of pruning by many experts. This concept is very important concept because almost every data set has some values which are not well defined. Some values also differ from the neighboring set of

values. After the completion of decision tree, all the occurrences must be classified and distinct.

Both these problems discussed above addresses a particular setback and that is disorder. The minimum the disorder, the higher will be the correctness of results and reliability of the predictiveness. This is done to make the decision tree more universal.

Figure 2 describes a simple example of disorder. The data set is shown by the mathematical symbols of + and -. After applying the selected algorithm, if we get the data in some ordered format, then we can call it a good split otherwise it is bad split.

Although C4.5 had a lot of problems in it yet it was a good advancement from ID3, as it addresses most of the problems which were being faced in ID3. For example, C4.5 can hold both type of data whether it is continuous or discrete. It uses a threshold for this purpose, and splits the values above or below, according to that threshold.

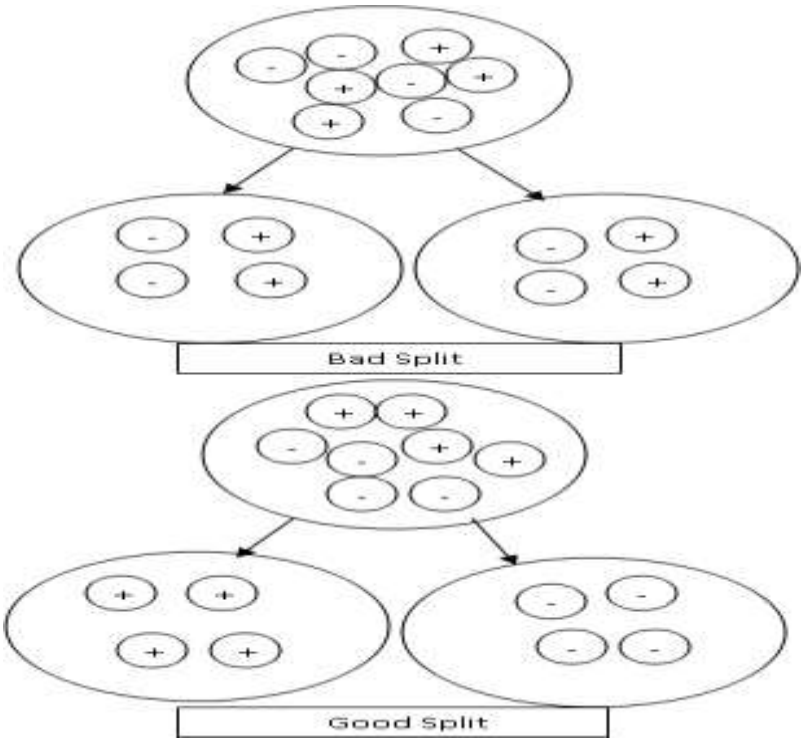

Figure 2. Information Gain Example

4. C5 Algorithm

C5 algorithm is an extension of C4.5. In C4.5, all the errors were taken equally. There was no segregation of the errors based on their importance or significance. A clear improvement in C5 over C4.5 is that it treats all the errors with individual classification depending on the magnitude of their impact on the system. It builds classifiers which helps reducing the misclassification cost instead of the high error toll. This characteristic of C5 is known as variable misclassification costs. Until now we discussed about the errors costs for individual attributes of a data set. There can also be the situations where the cases differ itself. Suppose an Application which classifies a group of individuals to churn. In this example, every case be of diverse importance due to the size of account. This problem is also very well addressed in C5 by applying a characteristic called case weight attribute. Using this feature, C5 reduces the biased predictive miscalculation cost.

C5 also have much more amount of data types as compared to C4.5 or the previous algorithms. This includes date, timestamp case labels *etc*. Another limitation was of missing value, *i.e.*, if some value does not fir according to the given dataset, or due to

some other reasons it gets into that account, increases the error ratio and thus reduces the predictiveness of the result. C5 defines a new data type for this, named 'not applicable'. It also makes easy the inclusion of a new feature as a function of some other feature.

In recent work, there are many applications of data mining [2, 5, 11] that consists of thousands of attributes or features. Most of them are obviously not required as we get specific to a classification. This algorithm minnow the features if these are not significantly relevant to a particular classification. This approach saves us a lot of computing [1] time for many applications and even get more accurate and to the point results. In addition, many of the different features of C4.5 have been merged together in C5, for example, cross validation and sampling, which makes this algorithm more easy and efficient. It also gives efficient generation of rule sets and decision trees.

This algorithm has two versions, one for UNIX named as C5 and the second for Windows named as See5. The Windows version is easier as it gives GUI. It has got many interesting and useful features such as "Cross Reference Window". This makes linking of the relevant cases for a classifier very easy. A problem with this algorithm was, as it is with every new technique, that it was publically not available. But now the code is freely available and it can be used for customized application of classification.

A case from RULEQUEST RESEARCH [7, 11, 14,15,20] has been taken for the validation of our above comparison between C4.5 and C5. Complete data set and the results are the propriety of RULEQUEST. We have just included these in our paper with slight changes, to elaborate the assessment for the two algorithms in various aspects. There is only one data set used here and forest with 572 cases.

4.1. Rule Set

From Figure 3, it is clear that C5 is much faster as compared to C4.5 in computation. Rule sets require memory and normally C5 requires much less space for rule set construction.

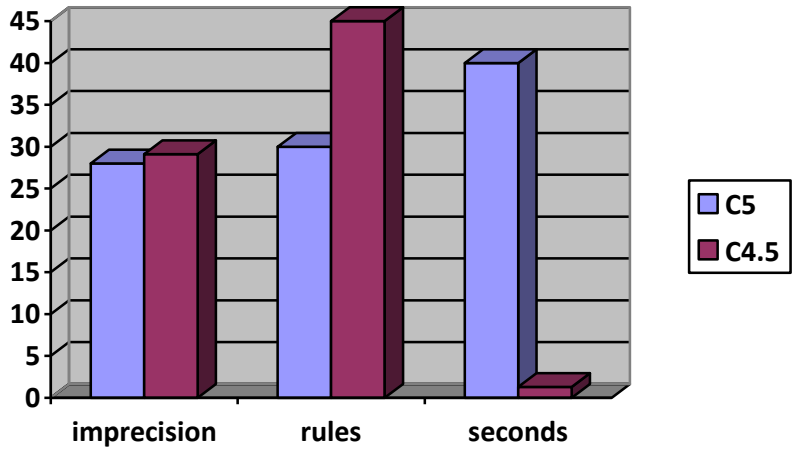

Figure 3. Accuracy, Speed and Memory

4.2. Decision Tree

From Figure 4, it is clear that C5 is much faster as compared to C4.5 in computation. Decision tree comparison depicts the graph in imprecision, leaves and seconds' context.

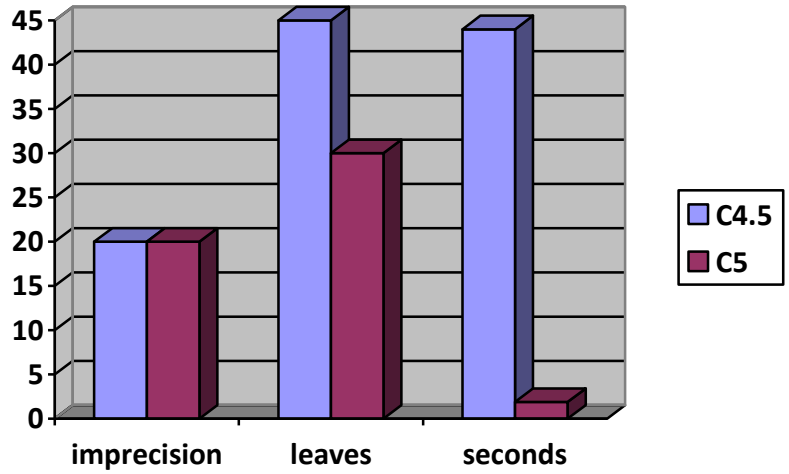

Figure 4. Decision Tree Comparison

5. Conclusion

C5 has been designed to handle the large database in Data Mining which consists of hundreds of thousands of records. It can have any type of values, *i.e.*, numeric, timestamps or any other type. It maximizes the interpretability by generating the rules in if-then form rather than difficult approaches used for Neural Networks. It is easily available for different operating systems. Another feature of these algorithms is that it does not require any of machine learning [4, 11, 15] knowledge type of other difficult algorithm, it's quite easy to implement. A lot of free software's are also available for implementing these algorithms. Its code is also available. So, these algorithms make data mining a lot easier.

References

[1] W. H. Au, K. C. Chan and X. Yao, "A novel evolutionary data mining algorithm with applications to churn prediction", Evolutionary Computation, IEEE Transactions, vol. 7, no. 6, **(2003)**, pp. 532-545.
[2] X. Li and A. Gar-On Yeh, "Data mining of cellular automata's transition rules", International Journal of Geographical Information Science, vol. 18, no. 8, **(2004)**, pp. 723-744.
[3] R. Kimball and M. Ross, "The data warehouse toolkit: the complete guide to dimensional modeling", John Wiley & Sons, **(2011)**.
[4] X. Wu, V. Kumar, J. R. Quinlan, J. Ghosh, Q. Yang, H. Motoda and D. Steinberg, "Top 10 algorithms in data mining", Knowledge and Information Systems, vol. 14, no. 1, **(2008)**, pp. 1-37.
[5] D. P. Foster and R. A. Stine, "Variable selection in data mining: Building a predictive model for bankruptcy", Journal of the American Statistical Association, vol. 99, no. 466, **(2004)**, pp. 303-313.
[6] R. Kimball, Margy RossGAO Yi-yang (School of Economic, Huazhong University of Science and Technology,Wuhan 470024, China).
[7] RULEQUEST research 2009 and 2011.
[8] L. A. Zadeh, W. J. Freeman, M. M. Gupta, M. Jamshidi, E. Sanchez, H. Szu and N. Ishihara, "Plenary Lecture", In IEEE International Conference on Neural Networks, San Francisco, CA, **(1993)**.
[9] T. S. Korting, "C4. 5 algorithm and multivariate decision trees", Image Processing Division, National Institute for Space Research–INPE Sao Jose dos Campos–SP, Brazil, **(2006)**.
[10] Decision Tree Discovery, by Ross Quinlan and Ron Kohavi, Computer Science and Engineering Dept, University of South Wales, USA.
[11] T. M. Mitchell, "Machine learning and data mining", Communications of the ACM, vol. 42, no. 11, **(1999)**, pp. 30-36.
[12] M. Arif, "A survey on data warehouse Construction, Processes and Architecture", International Journal of u- and e- Service, Science and Technology, vol. 8, no. 4, **(2015)**, pp. 9-16.
[13] M. Arif and F. Zaffar, "Challenges in efficient Data warehousing", International Journal of Grid and Distributed Computing, vol. 8, no. 2, **(2015)**.

[14] M. Arif and A. Roohani Dar, "Survey on Fraud Detection Techniques Using Data Mining", International Journal of u-and e-Service, Science and Technology, vol. 8, no. 3, **(2015)**, pp. 165-170.
[15] M. Arif and T. Mahmood, "Cloud Computing and its Environmental Effects", International Journal of Grid and Distributed Computing, vol. 8, no. 1, **(2015)**, pp. 279-286.
[16] M. Arif, K. Amjad Alam and M. Hussain, "Crime Mining: A Comprehensive Survey", International Journal of u-and e-Service, Science and Technology, vol. 8, no. 2, **(2015)**, pp. 357-364.
[17] M. Arif and H. Shakeel, "Virtualization Security: Analysis and Open Challenges", International Journal of Hybrid Information Technology, vol. 8, no. 2, **(2015)**, pp. 237-246.
[18] M. Arif, K. Amjad Alam and M. Hussain, "Application of data mining using artificial neural network: Survey", International Journal of Database Theory and Application, vol. 8, no. 1, **(2015)**, pp. 245-270.
[19] Z. Ahmed, "A Comparative Study for Ontology and Software Design Patterns", International Workshop Soft Computing Applications, Springer, Cham, **(2016)**.
[20] A. Ahmed, "MainIndex Sorting Algorithm", International Workshop Soft Computing Applications, Springer, Cham, **(2016)**.
[21] A. Ahmed, "A Smart Way to Improve the Printing Capability of Operating System", International Workshop Soft Computing Applications, Springer, Cham, **(2016)**.

Author

Muhammad Arif is a PhD student at Faculty of CS and IT, University of Malaya. Currently he is working on Medical image Processing. His research interests include image processing, E learning, Artificial intelligence and data mining. He joined UM as a Bright Spark Scholar in September 2013 for the period of 3 years. Before this he completed masters and bachelor degrees in Pakistan. He received his BS degree in Computer Science from University of Sargodha, Pakistan in 2011. He obtained his MS degree in Computer Science from COMSATS Islamabad 2013 Pakistan.

