Volume 5, Issue 1, January 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

A Dynamic Classifier using Decision Tree Algorithm

Sagar Manohar, Abhishika Mittal, Shivam Naik, Amruta Ambre
Dept. of Information Technology, Rajiv Gandhi Institute of Technology,
Mumbai, Maharashtra, India

Abstract—Every organization handles a vast amount of data. This data can be of different types such as sales, employee, customer, historical etc. The data often contains undiscovered patterns which can be useful if discovered and can be applied to future data for better efficiency. For the same, a dynamic classifier can be developed which can predict a certain attribute of future data by studying past data using concepts of Data Mining under Classification. This paper proposes the idea of such a classifier which can be built independently and without Bulky Business Intelligence software to effectively forecast future occurrences of any phenomena.

Keywords— Data Mining; Machine Learning; Classification; Knowledge discovery; Decision Tre; ID3 Algorithm.

I. INTRODUCTION

Data Mining is a non-trivial extraction of implicit, previously unknown, and imaginable useful information from data. It is a technique used to drill database for giving meaning to the approachable data. It involves systematic analysis of large data sets to find hidden patterns and consistencies. Data Mining is an interdisciplinary field involving: Databases, Statistics, and Machine Learning. [1]

Classification is an approach used in Data Mining which has many applications in real world, such as stock planning of large superstores, medical diagnosis, stock market predictions etc. There are various classification techniques i.e. Decision tree, K-nearest neighbour, Naïve Bayes classifier, Neural Network. [3]

II. LITERATURE SURVEY

Classification is a technique used in Data Mining that maps data into predefined groups or classes. It is a type of supervised learning which requires labelled training data to generate rules for classifying test data into predefined classes of a single goal attribute. Classification is a two-step process namely learning phase and classification phase. The learning phase consists of the classifier or model 'learning' from the training data. Training data is generally historical data whose goal attribute is classified beforehand. Output of the learning phase is a set of rules which can be illustrated in the form of a decision tree. These rules can then be applied to future data which is known as classification phase. [2]

In modern world, huge amount of information is kept in the databases. Thus data-mining can be very effective for extracting knowledge from such huge amount of data. There are various issues that need attention while doing the same.

- Data with uncertainty: Uncertain values have to be handled cautiously, or else the mining results could be unreliable or even wrong.
- High speed input data: The classifier should have the ability to process huge volume of data which arrive at high speed.
- Concept-drift detecting: The classifier should be capable of detecting and responding to changes in the example-generating process.
- Limited memory space: Only limited memory space is available to the classifier, which means that the classifier has to scan the input samples for only once. [4]

In the systems referred, the rule sets are hard coded and thus cannot be changed at run time. Also the type of data allowed is limited and narrowed to a particular field such as Healthcare or Sales data. The rule sets are often not displayed to the user in an easy to comprehend manner. These systems have many issues such as lack of dynamic operations, bulky systems, narrow scope etc. [5]

III. PROPOSED SYSTEM

A. Decision Tree

A Decision tree is a graph like tree-structure which is used to learn a classification function which concludes the value of dependent variable (output) given the value of independent variables (inputs). A Decision Tree is typically characterized by a set of nodes. It is traversed from the root node to a leaf node with each path generating a unique rule. Decision tree can produce a model with rules that are human-readable and interpretable. The classification task using decision tree technique can be performed without complicated computations and the technique can be used for both continuous and categorical variables. The technique is more suitable for predicting categorical outcomes. Fig. 1 shows a typical decision tree where 'Humidity' is the root node and YES or NO are the predefined classes.

B. Decision Tree Algorithm

A Decision Tree can be implemented by two approaches as follows:

1) Univariate Decision Tree: In this technique, splitting is performed by considering a single attribute and defining a predicate with the help of the same. For ex. X>3, Y<12 etc.

2) Multivariate Decision Tree: In this technique, two or more attributes are considered for splitting the nodes at internal leaves. Test predicate in this approach may be of the form X+Y>7.[3]

For the sake of simplicity at start, consider a Univariate Decision Tree for implementing the dynamic classifier. An example of such algorithm is Iterative Dichotomiser 3 (ID3) which is a widely used decision tree algorithm. It uses the concept of Information Gain and Entropy to accurately split the nodes by selecting best-possible attribute at each step.

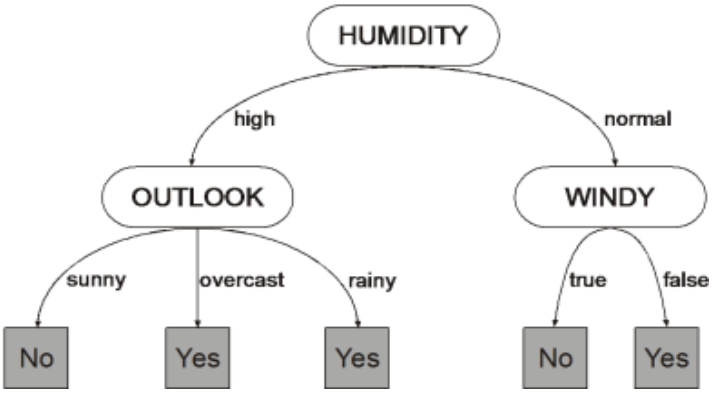

Fig. 1 Decision Tree

C. ID3 Algorithm

Iterative Dichotomiser 3 is typically used in the machine learning and natural language processing domains. The decision tree technique involves constructing a tree to model the classification process. Once a tree is built, it is applied to each tuple in the database and results in classification for that tuple.[2] The following issues are faced by most decision tree algorithms:

- Choosing splitting attributes
- Ordering of splitting attributes
- Number of splits to take
- Balance of tree structure and pruning

1) Entropy: Given a set of probabilities $p_1, p_2, \ldots p_n$ where $\Sigma p_i = 1$, Entropy is defined as

$$H(p_1, p_2 \ldots p_n) = \Sigma - (p_i \log p_i)$$

Entropy finds the amount of order in a given database. A value of H = 0 identifies a perfectly classified set. It means that higher the entropy, higher is the room to improve the classification process.

2) Information Gain: ID3 chooses the splitting attribute with the highest gain in information, where gain is defined as difference between how much information is needed after the split. This is calculated by determining the differences between the entropies of the original dataset and the weighted sum of entropies from each of the subdivided datasets.

$$G(D, S) = H(D) - \Sigma P(D_i) H(D_i)$$

A flow chart of the proposed system can be described as shown in Fig. 2.

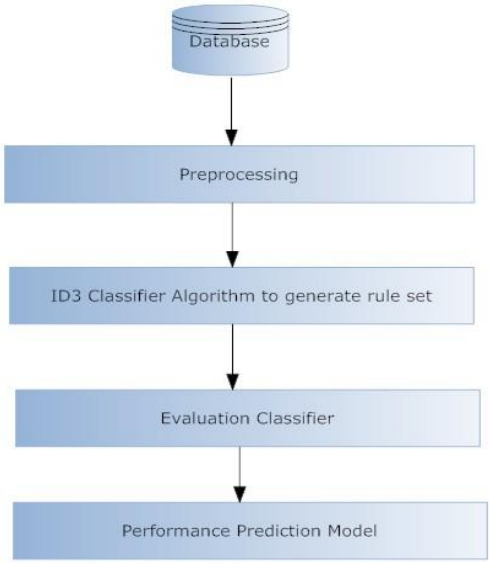

Fig. 2 Flow chart of Proposed System

IV. EXAMPLE

Consider the following example. A dataset is given which consists of independent attributes such as Outlook, Temp, Humidity, and Wind which describe the possible states of weather. 'Playball' is the dependent attribute (goal attribute) which has two values – YES or NO. Depending on the values of independent attributes, the classifier will classify Playball attribute of an unknown tuple as YES or NO. Following is the training data where each line defines a unique tuple that is used to train the classifier which is part of the learning phase as shown in Fig. 3.

```
Outlook Temp Humidity Wind Playball
//*********************************************
sunny hot high weak no
sunny hot high strong no
overcast hot high weak yes
rain mild high weak yes
rain cool normal weak yes
rain cool normal strong no
overcast cool normal strong yes
sunny mild high weak no
sunny cool normal weak yes
rain mild normal weak yes
sunny mild normal strong yes
overcast mild high strong yes
overcast hot normal weak yes
rain mild high strong no
```

Fig. 3 Input: Training Dataset

Once the classifier reads the training data, it applies ID3 Algorithm to the same to generate a set of rules as shown in Fig. 4. The rules are displayed in the form of if-else conditions but can also be represented more sophisticatedly by using a decision tree. These rules can then be applied to test data whose Playball tuple will be blank and will be assigned by the classifier.

```
if(Outlook== "overcast") {
                Playball="yes";
} else {
        if(Humidity== "high") {
                if(Outlook== "sunny") {
                                Playball="no";
                } else {
                        if(Wind== "weak") {
                                        Playball="yes";
                        } else {
                                        Playball="no";
                        }
                }
        } else {
                if(Wind== "weak") {
                                Playball="yes";
                } else {
                        if(Outlook== "sunny") {
                                        Playball="yes";
                        } else {
                                        Playball="no";
                        }
                }
        }
```

Fig. 4 Output: Rule set

V. CONCLUSIONS

In this research paper, we have proposed the idea of a dynamic classifier which can give results instantly. The classifier uses ID3 Algorithm for effective classification. The implementation can also be refined using C4.5 Algorithm which is an extension of ID3 Algorithm and provides added benefits such as support for missing values, pruning, handling large amount of data, and reduced execution time. The classifier can be hosted as a web application to increase its processing capabilities cumulatively. Also a web application will enable support to a diverse set of databases such as MySQL, Oracle, and Microsoft SQL Server etc. There can be an added support for a variety of file formats such as .txt, .csv, .arff etc. Such a classifier can be of great help in variety of fields such as Healthcare, Weather forecasting, Stock market predictions, Sales and Marketing etc.

REFERENCES

[1] Han, J., Kamber, M. (2001) Data Mining: Concepts and Techniques, Morgan Kaufmann.

[2] K. Adhatrao, A. Gaykar, A. Dhawan, R. Jha, V. Honrao, "Predicting students' performance using ID3 and C4.5 classification algorithms", *International Journal of Data Mining and Knowledge Management Process (IJDKP)* September 2013.

[3] H. Jantan, A.R. Hamdan, Z.A. Othman, "Human Talent Prediction in HRM using C4.5 classification algorithm*"*, *International Journal of Computer Science and Engineering,* 2010.

[4] N. Bhargava, G. Sharma, R. Bhargava, M. Mathuria, "Decision Tree Analysis on J48 algorithm for Data Mining*", International Journal of Advanced Research in Computer Science and Software Engineering,* June 2013.

[5] Veronica S. Moertini, "Towards the use of C4.5 algorithm for classifying banking dataset", *Integral Vol.8 No. 2,* October 2003.

