Volume 4, Issue 8, August 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Data Mining With Big Data Using C4.5 and Bayesian Classifier

Bharti Thakur
Computer Science Department
LRIET, Solan, H.P, India

Manish Mann
Computer Science Department
LRIET, Solan, H.P, India

Abstract:- ***Data mining is the extraction of useful data from the vast amount of data i-e big data. Data mining algorithms decision tree C4.5 and Bayesian classifier are compared in this paper. We have taken the record of 71,100students and these records are executed on both the algorithm simultaneously to see the variation in there execution time.***

Keywords:-Data mining, Big data, Pattern, C4.5, Bayesian classifier

I. INTRODUCTION

Data mining is the process of analyzing data from different perspectives and summarizing it into useful information. Data are any facts, numbers, or text that can be process by a computer. The patterns, associations, or relationships among all collected data can provide information. Information can be converted into knowledge about historical patterns and future trends. To maximize user access and analysis, there needs to be a centralization of data in a data warehouse. With the help of data mining we can extract the meaningful data from the vast amount of data i-e big data.

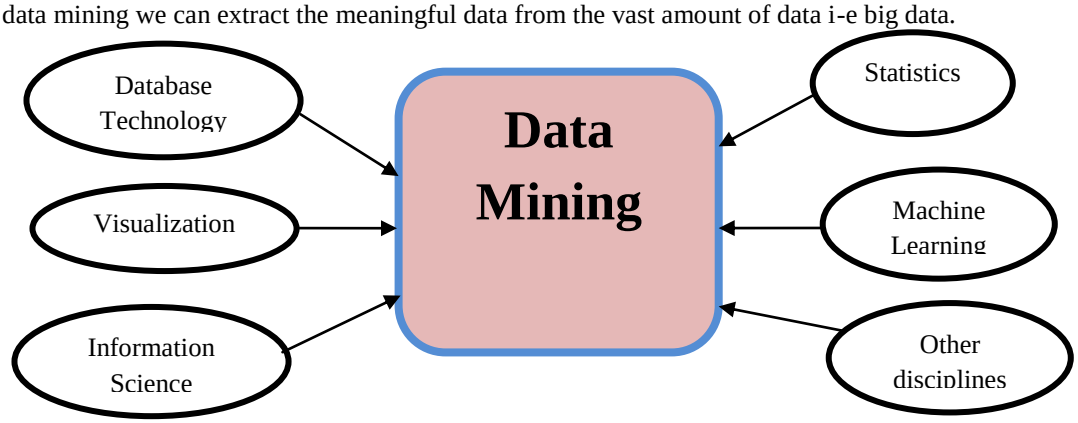


Figure 1.1 showing data mining system

Classification according to the kinds of database mined:A data mining can be classified according to the kinds of databases mined. Database systems can be classified according to different criteria each of which may require its own data mining technique.Data Mining systems can be therefore classified accordingly.
Classification according to the kinds of knowledge mined: Data mining systems can be categorized according the kinds of knowledge they mine that is based on data mining functionalities such as characterization discrimination association classification,prediction analysis.
Classification according to the kinds of technique utilized:-Data mining system can be catergorized according to the underlying data mining techniques employed.These techniques can be described according to the degree of user interaction involved.

II. C4.5 ALGORITHM

Systems that construct classifiers are one of the commonly used tools in data mining. Such systems take as input a collection of cases, each belonging to one of a small number of classes and described by its values for a fixed set of attributes, and output a classifier that can accurately predict the class to which a new case belongs. Like CLS and ID3, C4.5 generates classifiers expressed as decision trees, but it can also construct classifiers in more comprehensible rule set form comprehensible rule set form.
C4.5 is an algorithm used to generate a decision tree developed by Ross Quinlan. C4.5 is an extension of Quinlan's earlier ID3 algorithm. The decision trees generated by C4.5 can be used for classification, and for this reason, C4.5 is often referred to as a statistical classifier.C4.5 builds decision trees from a set of training data using the concept of information entropy. The training data is a set $S = s_1, s_2, \cdots$ of already classified samples. Each sample s_i consists of a p-dimensional vector $(x_{1,i}, x_{2,i}, \ldots, x_{p,i})$, where the x_j represent attributes or features of the sample, as well as the class in which s_i falls. At each node of the tree, C4.5 chooses the attribute of the data that most

effectively splits its set of samples into subsets enriched in one class or the other. The splitting criterion is the normalized information gain (difference in entropy). The attribute with the highest normalized information gain is chosen to make the decision. The C4.5 algorithm then recurs on the smaller sublists.

This algorithm has a few base cases.

1. All the samples in the list belong to the same class. When this happens, it simply creates a leaf node for the decision tree saying to choose that class.

2.None of the features provide any information gain. In this case, C4.5 creates a decision node higher up the tree using the expected value of the class.

3. Instance of previously-unseen class encountered. Again, C4.5 creates a decision node higher up the tree using the expected value.

4.Let the classes be denoted $\{C_1, C_2,\ldots, C_k\}$. There are three possibilities for the content of the set of training samples T in the given node of decision tree:

A. T contains one or more samples, all belonging to a single class C_j. The decision tree for T is a leaf identifying class C_j

B. T contains no samples. The decision tree is again a leaf, but the class to be associated with the leaf must be determined from information other than T, such as the overall majority class in T. C4.5 algorithm uses as a criterion the most frequent class at the parent of the given node.

C. T contains samples that belong to a mixture of classes. In this situation, the idea is to refine T into subsets of samples that are heading towards single-class collections of samples. An appropriate test is chosen, based on single attribute, that has one or more mutually exclusive outcomes $\{O_1,O_2, \ldots,O_n\}$:

(i) T is partitioned into subsets $T_1, T_2, \ldots, T_n$ where T_i contains all the samples in T that have outcome O_i of the chosen test. The decision tree for T consists of a decision node identifying the test and one branch for each possible outcome.

(ii)Test – entropy:

(iii)If *S* is any set of samples, let *freq* (C_i, *S*) stand for the number of samples in S that belong to class C_i (out of *k* possible classes), and $|S|$ denotes the number of samples in the set *S*. Then the entropy of the set *S*:

(iv)After set *T* has been partitioned in accordance with n outcomes of one attribute test *X*:

(a) $\text{Info}_x(T) = \sum((|Ti|/|T|) \cdot \text{Info}(Ti))$

(b) $\text{Gain}(X) = \text{Info}(T) - \text{Info}_x(T)$

(c) Criterion: select an attribute with the highest Gain value.

(d) $\text{Info}(S) = -\sum((\text{freq}(C_i, S)/|S|) \cdot \log_2(\text{freq}(C_i, S)/|S|))$

III. NAÏVE BAYES CLASSIFIERS

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. One very important one is the naïve Bayes method—also called idiot's Bayes, 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.

In machine learning, naive Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive)independence assumptions between the features.Naive Bayes is a popular (baseline) method for text categorization , the problem of judging documents as belonging to one category or the other (such as spam or legitimate, sports or politics, etc.) with word frequencies as the features. With appropriate preprocessing, it is competitive in this domain with more advanced methods including support vector machines.Training naive Bayes can be done by evaluating an approximation algorithm in closed form in linear time,rather than by expensive iterative approximation. In simple terms, a naive Bayes classifier assumes that the value of a particular feature is unrelated to the presence or absence of any other feature, given the class variable. For example, a fruit may be considered to be an apple if it is red, round, and about 3" in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of the presence or absence of the other features.For some types of probability models, naive Bayes classifiers can be trained very efficiently in a supervised learning setting. In many practical applications, parameter estimation for naive Bayes models uses the method of maximum likelihood, in other words, one can work with the naive Bayes model without accepting Bayesian probability or using any Bayesian methods.

A.*Bayes Rule* :-The basic idea of Bayes' rule is that the outcome of a hypothesis or an event (H)can be predicted based on some evidences(E) that can be observed.The Bayes rule is given as

$$P(H|E)=\frac{P(E|H)*P(H)}{P(E)}$$

From Bayes rule we have

A priori probability of H orP(H).This is the probability of an event before the evidence is observed.

A posterior probability of H or P(H|E).This is the probability of an event after the evidence is observed.

Example

To predict the chance of the probability of raining we usually use some evidences such as the amout of dark cloud in that area.

Let H be the event of raining and E be the evidence of dark cloud,then we have

$$P(\text{raining}|\text{dark cloud}) = \frac{P(\text{dark cloud}|\text{raining}) * P(\text{raining})}{P(\text{dark cloud})}$$

Bayes Classifier is mainly suited when the dimenstionality of the inputs is high.In spite of its simplicity the Naive Bayes can frequently do better than more sophisticated classification methods.

We can predict the outcome of some events by observing some evdences.Generally it is better to have more than one evidences to support the prediction of an event. When there are more evidences the accuracy is higher in the classificatio task,But the evidence must be related to the event.

Basic idea of Naive Bayes Classifier

(1) Assume that there are a set of m samples $S=(S_1,S_2,S_3....S_m)$

(2) Where every sample Si is represented as an n-dimensional vector$(x_1,x_2,x_3...x_n)$

(3) Values xi correspond to attributes A1,A2...An respectively.

(4) There are k classes C1,C2,C3 and every sample belong to one of these classes.

(5) Given an additional data sample X(its class is unknown)it is possible to predict the calss for X using the highest probability $P(c_1|X)$where i=1...K

(6)These probabilities are computed using Bayes Theroem$P(C_i|X)$where i=1...K

(7)These probabilites are computed using Bayes Theorem

$P(Ci|X)=\{P(X|C_i).P(C_i)\}/P(X)$

(8)As P(X) is constant for all classes only the product $P(X|C_i)$,P(Ci) needs to be maximized.

(9)We compute the prior probabilities of the class as $P(C_i)$=number of training samples of class C_i/m(m is total number of training samples)

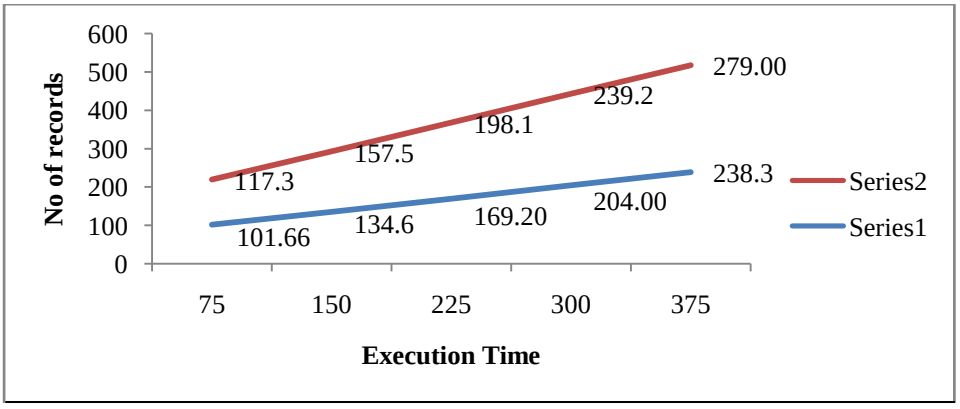
(10)Becausse the computation of P(X|Ci) is extremly complex especially for large data sets the Naive assumption of conditional independence between attributes is made .

IV. OBJECTIVES OF OUR WORK

The objective of our experimental work is to find out the different patterns from the large amount of data and distribute a particular class to new data. Our work is presented on mining frequent patterns from a large scale database. We used AMD Athlon(tm)II* 2 Processor,Memory used 4GB RAM,we compared C4.5 and Bayesian classifiers algorithm to identify the class of a particular data,We have taken the data of 71100 students of a university,the data is divided into different classes and in different phases.We implemented different rules of both the algorithm on same data to find out the differences in the execution time. While comparing both the algorithm on same platform we find out that Bayesian classifier uses the probability factor in it and is more efficient when we have to mine the large amount of dataset of different classes.On the basis of our experiement we have drawn chart given below.

Execution time of C4.5 and Bayesian classifier using different number of records

Sr.No	No of Records	Execution Time Bayesian Classifier	Execution Time C4.5
1	31,100	101.66milliseconds	117.3milliseconds
2	41,100	134.6milliseconds	157.5milliseconds
3	51,100	169.2milliseconds	198.1milliseconds
4	61,100	204.0milliseconds	239.2milliseconds
5	71,100	238.3 milliseconds	279.0milliseconds



In the above graph we have executed both the algorithm decision tree C4.5 and Bayesian Classifier, total no. of records are 71,100(which are same for both the algorithms) these records are divided into the following series 31100,41100,51100,61100,7110(x axis)and execution time is taken on (y axis) as the no.of records are increasing the execution time of both the algorithms are increasing simultaneously but the execution time of Bayesian Classifier is less than the execution time of the Decision tree.Graph is moving towards the upward direction of both the algorithms but the execution time is different.

V. FORECAST TO FUTURE

There are many future important challenges in Big Data management and analytics, that arise from the nature of data:large, diverse, and evolving. These are some of the challenges that researchers and practitioners will have to deal during the next years:

A. Different Algorithms:- Different type of algorithms of data mining are compared with each other to find out the efficiency of an algorithm for pattern recognisition in big data.

***B. Unstructured Data*:-**Large amount of data is unstructured which contains numbers, images different alphabets in it.Lot of work can be done on unstructured data in future because mostly data is unstructured these days.

***C. Analytics Architecture*:-** It is not clear yet how an optimal architecture of an analytics systems should be to deal with historic data and with real-time data at the same time. An interesting proposal is the Lambda architecture of Nathan Marz.The Lambda Architecture solves the problem of computing arbitrary functions on arbitrary data in real time by decomposing the problem into three layers: the batch layer, the serving layer, and the speed layer. It combines in the same system Hadoop for the batch layer, and Storm for the speed layer. The properties of the system are: robust and fault tolerant, scalable, general, extensible, allows ad hoc queries, minimal maintenance, and debuggable.

D. Distributed mining:- Many data mining techniques are not trivial to paralyze. To have distributed versions of some methods, a lot of research is needed with practical and theoretical analysis to provide new methods.

E. Time evolving data:- Data may be evolving over time, so it is important that the Big Data mining techniquesshould be able to adapt and in some cases to detect change first. For example, the data stream mining field has very powerful techniques for this task

REFERENCES

[1] Decision Trees for Business Intelligence and Data Mining: Using SAS Enterprise Miner "Decision Trees-What Are They?"

[2] Weiss, S.H. and Indurkhya, N. (1998), Predictive Data Mining: A Practical Guide, Morgan Kaufmann Publishers, San Francisco, CA.

[3] Minimal MapReduce Algorithms,Yufei Tao,Wenqing Lin, Xiaokui Xiao.

[4] Algorithm and approaches to handle large Data-A Survey, IJCSN International Journal of Computer Science and network, Vol 2, Issue 3, 2013 ISSN (Online) : 2277-5420.

[5] Supervised Learning: K-Nearest Neighbors and Decision Trees Piyush Rai CS5350/6350: Machin learning,August 25, 2011,(CS5350/6350

[6] Alex Berson and Stephen J.Smith Data Warehousing,Data Mining and OLAP edition 2010.

[7] Working with Big Data www.bls.gov/ooq.Fa 2013.

[8] Department of Finance and Deregulation Australian Government Big Data Strategy-Issue Paper March 2013.

[9] Supervised learning with decision tree-based methods in computational and systems biology.

[10] Niuniu and L. Yuxun, "Review of Decision Trees," IEEE, 2010

