Implemented Classification for Stock Market Prediction using CART with B+ Tree

Kalpna Singh[1], U.Datta[1],K.K.Joshi[1]
[1] Deptt. of Computer Science &Engg., MPCT Gwalior.

Abstract

Data mining used many techniques to analyze an enormous set of data and get hidden data for analyzing the information for the current and future movement of price in stock market. I started design a system that would allow researcher to chart (Indian) stock market variation nicely and simply, and in a user-friendly environment. Now we decided clients need to be able to look at their charts without having to download large amounts of data every time they open a window. The objective of classification algorithms is to generate meaningful and fruitful results. In this paper, i have implemented the data mining technique in stock market prediction using CART with B+ tree. Why B-Tree is useful? Well, since b tree allow storage of large amounts of data on disk, indexed by key, with records accessible in $O(log\ n)$ time where n is the number of nodes in the tree.

Keywords—Stock market, CART, B+ tree

I. INTRODUCTION

 80% of Intraday Trades are go in lose , Intraday investment without a proper plan and trading system is equivalent to play gambling which most of traders get plunge into. Before betting your hard earned money to earn quick returns its always better to know and understand the fundamentals of intraday trading, if you hope to make money consistently in the market. in my research I will focus on essentials traits required to do Intraday trading successfully.

Always invest in liquid stocks like TCS, Wipro, Infosys which will make sure you are never stuck up with your positions, try ignore trading in illiquid mid cap stocks in which just a volume of 5000 shares can swing prices wildly.
A liquid stock is one, which return high average trading volumes, so that it can be bought or sold in large quantities without causing much impact on the prices.

The importance of set your stop loss cannot be over-emphasized for day trading. Many investors find it extremely difficult to book a loss and thus take delivery of a stock if its rates moves down. What happens in such a condition is that you have gone against the basic rule of day trading by not keeping it an intraday trade. So now investor do not know when the stock will regain and if you will be able to exit. By doing so, many investors to keep the stock for the long time and often get stuck with it. Therefore you should always have a stop loss and stick to it.

There are a so many of stocks in the indian market which have a great correlation to the fluctuation of the premium stock. For example if the Nifty is increasing these stocks will also grow to move up and vice versa. This makes it easier to find and predict the situation of the stock and thus increase your chances of making a profit. So if you ignore trading in Nifty trade in Stocks like ONGC ,Axis,HDFC and TCS, which are having premium stock in Index and they will not rise and fall 8-10% on any given day and hence giving you volatility in your trading capital.
You have to Hard Work and if you want profitable result in intraday trader you need to do lot of Research & study before taking the trade.
Technical analysis **is the study & research of market action primarily through the use of charts & graphs, for the purpose of forecasting future rate or price trends. My work helps out clients to make decision while they will do intraday trading.**

II. RELATED WORK

[1]Sumit Garg   et al In his research he describe that single algorithm may not help to find the suitable data in data mining. His paper focuses on comparative analysis of various data mining techniques and algorithms.
[2] Shashikumar G. Totad , Geeta R. B. et al In this research paper they discuss about the issues that is being carried out on parallel and distributed data mining. They find out some core data mining algorithms such as decision trees, discovery of frequent patterns, clustering,

International Journal of Scientific Engineering and Applied Science (IJSEAS) – Volume-2, Issue-2, February 2016
ISSN: 2395-3470
www.ijseas.com

etc., to find quick result for parallel processing in data mining.

[3]**Mohd Mahmood Ali1, Lakshmi Rajamani et** al In this paper they proposed a data classification method using AVL trees enhances the quality, accuracy and stability of data mining problems. But this research is useful only in memory based

Searching. But when used this technique in disk based serach the this technique slow down the process.

[4] **Sneha Soni Samrat et al.** In this paper, they made combination of three supervised machine learning algorithms, classification and regression tree (CART) , linear discriminant analysis (LDA) and quadratic discriminant analysis (QDA) are proposed for classification of Indian stock market data, which gives simple interpretation of stock market data in the form of binary tree, linear surface and quadratic surface respectively. But as we discuss earlier binary tree & AVL tree is best when we uses this techniques in memory based problems.

[5] **Garima Saini et al** In her paper she found that the classification techniques are best known for producing accurate, rapid and straight forward results. She implemented data mining technique using CART with AVL tree to find out the stock market prediction.

III. PROPOSED WORK

I have implemented CART with B+ tree using C language and Windows programming in C# to prepare Graphs to analyzing the result of stock market selected. CART (Classification and Regression Tree) algorithm is the best technique to find out the quick and accurate result. Many researcher have modified this technique to increase the efficiency of CART algorithm. Some researchers have implemented with decision tree and some with AVL. AVL tree is far much better the decision tree and its work faster than decision tree because we maintain the height of tree at each insertion step so it find out the result faster. .

A. CLASSIFICATION AND REGRESSION TREE (CART)

CART technique was developed by Leo Breiman, Jerome Friedman, Richard Olsen and Charles Stone in 1984. CART is one of the best methods of building decision trees in the machine learning community CART create a binary decision tree by splitting the values/rate at each node, according to a function of single attribute. CART uses the gini index to find out the best split. CART follows the following principle of constructing the decision tree. Binary tree that all elements in the left sub-tree of a node n are less than the contents of n, and all elements in the right sub-tree of n are greater than or equal to the contents of n. Binary tree is useful data structure when two-way decisions must be made at each point in a process. For example search the all duplicates values in a list of numbers. One way doing this is each number compare with given node and precede it. However, it involves a large number of comparisons. The number of comparisons can be reduced by using a binary tree. in this work terminal node is companies of the stock market dataset [2*].The major disadvantage of a binary search tree is that the height of tree must be maximum N-1. This means that the time required to perform deletion as well as insertion and for the many other operations should be O(N) in the worst case . if one can want a tree with a minimum height .Thus, our goal is to keep the height of a binary search tree O(log N) .This can be accomplished* when we implement CART Using AVL tree algorithm.

Why CART using B+ Tree is useful? Using AVL Tree we found that its useful in memory based search but in case of disk based search B+ tree allow storage of large amounts of data on disk, indexed by key, with records accessible in $O(log\ n)$ time where n is the number of nodes in the tree. Here I am implementing stock market analysis for intraday trade and find out the least price, max price and average price movement according to time, this helps client to find out the nature of particular share in market.

struct BPTreeNode
*{char *ti[5];*
  *int *data;*
    *BPTreeNode **child_ptr;*
    int leaf;
    int n;
*}*root = NULL, *np = NULL, *x = NULL;*
Here ti reperesnt time and data represent rate of share.

Search.

To find all records having a search key value data:

- *Begin at the root node.*

  - *scan the current node for lowest key entry value K_i > data*

  - *if K_i exists, retrieve the node from pointer np_i.*

  - *if no such K_i exists in the node, then retrieve the node from pointer np_n.*

- *Repeat the above procedure until a leaf node is reached.*

- *When a leaf node is reached, scan the record for K_i = data. If found, then retrieve the value record using np_i. Else, no key with data exists.*

Insert.

Find the node np_i where data belongs to rate (use above search procedure to find leaf node).

- * *If space for data exists in np_i, add record to data file page and add new entry (data,np) to leaf node. Insert is complete.*

- * *If no space in np_i for new entry, then split: allocate a new node page np_n and move the upper half of the entries in the existing node to the new node page and to npn. Insert data in either np_i or np_n, as appropriate.*

- * *Let k be the lowest value $data_k$ in np_n. Insert k into the parent node of np_i*

  ### Algorithm

Sorting of Nodes
Sort()
Step1:[initialization]
       Set i=0
Step2: Repeat step3 While(l>n)
Step3: set j=i
Step4: Repeat step5 While (j<n)
Step5: [Compare the price]
```
        if (p[i] > p[j])
        {
          temp = p[i];
          p[i] = p[j];
```
End of if statement
Step6 : j++
     [end of step 5 loop]
Step7 : i++
     [end of step 2 loop]
Step8: return

Splitting Nodes

split_child (x, i)
X represent current node data represent price , ti represent price
Here we are using node pointer one *np1 and node pointer two *np3
I represent the index of particular node if I is -1 then node contain 5 item
Step1: [Create new node]
       np3 = create a new node consist of time and price store
Step2: set `np3->leaf = 1;`
```
      if (i == -1) then

      mid = x->data[2];
      strcpy(nmid,x->ti[2]);
      x->data[2] = 0;
      x->ti[2]=NULL;
      x->n--;
      np1 = init();
      np1->leaf = 0;
      x->leaf = 1;
```
`Step3: set j=3`
`Step4: Repeat Step5: While (j<5)`
`Step5: np3->data[j - 3] = x->data[j];`
```
      strcpy(np3->ti[j-3],x->ti[j]);
      np3->child_ptr[j - 3] = x->child_ptr[j];
      np3->n++;
      x->data[j] = 0;
    x->ti[j]=NULL;
      x->n--; j=j+1
```
`[end of step4 while loop]`
`Step6: Set j=0`
`Step7: Repeat step8 While(j<6)`
`Step8:  x->child_ptr[j] = NULL;`
```
        j=j+1
      [end while loop ]
```
`Step9: set np1->data[0] = mid;`
```
        np1->ti[0],nmid);
        np1->child_ptr[np1->n] = x;
      np1->child_ptr[np1->n + 1] = np3;
        np1->n++;
        root = np1;
else   y = x->child->ptr[i] ,
      mid = y->data[2];
    nmid,y->ti[2]);
```

International Journal of Scientific Engineering and Applied Science (IJSEAS) – Volume-2, Issue-2, February 2016
ISSN: 2395-3470
www.ijseas.com

```
        y->data[2] = 0;
        y->ti[2]=NULL;
          y->n--;
Step10: set j=3
Repeat step11 While (j<5)
Step11:
      np3->data[j - 3] = y->data[j];
        (np3->ti[j-3],y->ti[j]);
         np3->n++;
         y->data[j] = 0;
         y->ti[j]=NULL;
         y->n--;
        j=j+1
      [end of step 10 loop]
Step12: x->child_ptr[i + 1] = y;
        x->child_ptr[i + 1] = np3;
Step13: return (mid)
```

Insert Node in a List
```
Insert (ti,p)
```
ti represent time and p represent root is
the price node

```
Step1: [initialization]
      Set x=root,
Step2: if x==null then
      Root = create a new node
      Set = root
      else
    if (x->leaf == 1 && x->n == 5)
       set temp = split_child(x, -1);
      set  x = root;
Set i=0
Repeat step (a) to while (i < (x->n))
```
 (a) ` if ((a > x->data[i]) && (a < x-`
```
      >data[i + 1]))
                i++;
            break;
          else if (a < x->data[0])
            break;
          else
              continue;
      [end of if statement]
      Set x = x->child_ptr[i];
      else
      Repeat step (b) to (c) while
              (x->leaf == 0)
```
 (b) `     Set i=0`
 (c) `     Repeat step (i) while i<x->n`

 (i) `     if ((a > x->data[i]) && (a < x-`
```
        >data[i + 1]))
                i++;
              break;
          else if (a < x->data[0])
              break;
            else
            continue;
      [end of if statement]
      [end of step (c) while ]

    if ((x->child_ptr[i])->n == 5)
              temp = split_child(x, i);
              x->data[x->n] = temp;
            x->n++;
            continue;
          else
          x = x->child_ptr[i];

      [end of if statement]
      [end of while loop]
      [end of step (a) if]
      [end of while loop]
      [end of step (2) if]

Step3: set
      x->ti[x->n],na;
      x->data[x->n] = a;
      sort(x->data, x->n);
      x->n++;

Step4: Return
```

Result


```
traversal of constructed tree


 9AM[10] 11AM[17]
 12PM[20]
 1PM[21] 2PM[22] 10AM[25]

Least Rate:10
Max Rate:25
Mean Value:22
Total Transactions:6
Total Times  Share  Grow Above 22 is 2
Total Times  Share  Go Down 22 is 4
```

Conclusion

The previous researches have been done there work using CART ,CART with Decision Tree & AVL Tree but this work only produce result when data stored and arranged in memory. To Access stored data on disk as well as in memory above algorithm will work fast because all nodes stored in leaf. Thus CART is better implementation using B+ tree algorithm. It is accurate, fast and precise

Reference

1]. Sumit Garg   Comparative Analysis of Data Mining Techniques on Educational Dataset
M.Tech Scholar Dept. of Computer Science Shekhawati Engineering College Dundlod, Rajasthan, India

2]. Shashikumar G. Totad , Geeta R. B. , Chennupati R Prasanna , N Krishna Santhosh , PVGD Prasad Reddy
Scaling Data Mining Algorithms to Large and Distributed Datasets

3]. Mohd Mahmood Ali1, Lakshmi Rajamani2
1Assistant Professor, Dept of CSE, Muffakham Jah
Decision Tree Induction: Data Classification using Height-Balanced Tree,College of Engineering & Technology, Hyderabad, India 2Professor, Department of CSE, University College of engineering, Osmania University, India

4]. Sneha Soni, Classification of Indian Stock Market Data Using Machine Learning Algorithms
Samrat Ashok Technological Institute VIDISHA, M.P., India
5]. Garima Saini, Discovering Approach of Classification for stock market prediction using Cart with AVL. Birla Institute of Technology Mesra India.