A Novel Hybrid Method for Generating Association Rules for Stock Market Data

Pandya Jalpa P.[1], Morena Rustom D.[2]

[1]*Asst. Professor, UCCC & SPBCBA & SDHG College of BCA & IT, Udhna, Surat, India*
[2]*Professor, Veer Narmad South Gujarat University, Department of Computer Science, Surat, India*

Abstract **- The aim of the current research is to extract the knowledge from stock market data to help investors make more profit. We have used NSE (National Stock Exchange) historical data for six years. We have also pre-processed stock market data. We have used hybrid method which consists of two widely used algorithms FP-growth to find frequent patterns and Discovery rule algorithm proposed by Agrawal'94 to get association rules for two different steps of association rule mining. The main focus in our research has been the accuracy of the rules. The goal of the research is to find dependencies among different stock companies in the stock market and generate rules from inter-day transactions that would benefit stock market traders.**

Keywords: **Stock market, NSE, Association Rule Mining, FP-growth**

I. INTRODUCTION

Now days, utilisation of large amount of data from stored databases and to analyse it is one popular trend prevalent amongst researchers. Extraction of hidden patterns or co-relation among related attributes from stock market analysis is one of the major research areas. The stock market is the place where investors purchase stocks become shareholders for financial achievements of the companies. If the price of stocks goes down then the investor losses money and if stock prices goes up the investor makes profit. So, Investors trade on shares for making profit. It is desirable to generate interesting rules or patterns to help the investor make profit.

Stock Market Analysis and prediction is done by lots of researchers to extract associated and co-related rules or patterns to increase the profit in stock market. Stock market data is available in different forms like stock indices [7], inter-day-transaction [8][9][17][18][4], intra-day-transaction [11] and many more forms. Stock market has the number of product types for trading like equity, mutual funds, derivatives etc. Since last many years, researchers have taken keen interest and worked in this area. The stock market fluctuations and unpredictable changes require discovering some concrete and interesting rules from it, so that investors can be guided to make safe decisions for investment to get maximum profit.

The National Stock Exchange of India Limited (NSE) is the leading stock exchange of India, located in Mumbai. NSE was established in 1992 as the first demutualized electronic exchange in the country. It is world's 12th largest stock exchange as of March 2016. NSE offers trading, clearing and settlement services in equity, equity derivatives, and debt and currency derivatives segments. NSE is closed on Saturdays [33][35].

Data mining is the process, which extracts hidden and interesting patterns and rules from large databases. Data Mining plays vital role in organizing and analysing data. Data Mining is also known as "Knowledge Mining", which is used to mine knowledge from massive database. Knowledge in the form of patterns and rules is analysed from data mining. Data mining has different techniques like association rule mining, clustering, classification and etc. to find the meaningful and useful data and rules [26][27][28].

Association Rule Mining (ARM) is one of the major techniques, which is used to discover association between related items. It can also be used to find co-relation between two or more attributes from large databases. In 1993, Agrawal [28] presented the problem of mining association rules between sets of items in a large database of customer transactions. An itemset that occurs in the database for a minimum number of times is said to be a frequent itemset. First time they introduced the concept of minimum transactional supports, and minimum confidence of transactions in the database. They divided the problem in two sub problems.

1) Finding all itemsets called large itemsets that are present in at least s% of transactions
2) Generating rules from each large itemset that use items from the large itemsets.

There are three basic algorithms to find frequent patterns, Apriori, Eclat and FP-growth. Among these three algorithm FP-growth is the most efficient and fastest algorithm [21][6][34].

II. LITERATURE REVIEW

Agrawal and R. Srikant in [2] presented problem of discovering association rules between items in a large database of sales transactions and suggested two new algorithms to solve the problem. Researcher also invented AprioriHybrid algorithm by combining two new algorithms

Apriori and Apriori-TID and showed AprioriHybrid has excellent scale-up properties which scales linearly with the number of transactions.

Krittithee Utthammajai et al. [7] proposed theory of rough set to find the hidden relationship among the indicators which affects the market price. This paper has achieved relation, which does not occur frequently but happens when every time the same cause is raised. The association results show that the set of indicators affects the price change. When the indicators are changed substantially, the price will also change that is most important thing. The researchers used transactional data SET50 index from the Stock Exchange of Thailand (SET) with every half an hour in the period of April 10, 2013 to September 5, 2014.

Anantaporn Srisawat [8] proposed an application of ARM for discovering the relationships between individual stocks based on Thai stock market. The researcher used 242 days transaction data from historical Thai market. Each transaction represents a daily trading information which contains all individual stocks rise or fall more than or equal to a particular percentage over the previous day's close. An individual stock is encoded by adding a letter "i" or "d" in front of a name of that stock. The technique used by this paper is setting of minsup and minconf and also the concept of lift. The results from the discovered association rules can indicate the trends of related individual stocks.

Asadullah Al Galib et al. [9] proposed the STRDTM (Stock Trading Rule Discovery by Temporal Mining) algorithm with real life data. In this paper transaction data is collected from Dhaka Stock Exchange. Sequential continuous patterns are invented. The patterns serve as rules that enable us to determine the occurrence of an event on a particular stock-transaction day.

Shona Ulagapriya et al.[10] focused on analysis of stock grouped under different sector indices of NSE India. For analysis, daily closing prices are taken and Apriori associative algorithm for mining association rules has been applied. They set minimum confidence 0.7 and used top 1000 rules for further analysis.

Priti Saxena et al. [11] proposed data mining approach to search interesting patterns and associations between stock attributes. A comparative study of Apriori and modified Reverse-Apriori has been done. In this paper, proposed work converts the numeric stock data to symbolic notations and has done a comparative study. To predict the intra-day movements and trends the proposed work is used. The researcher proved modified Reverse-Apriori takes less execution time than Apriori algorithm.

Kanti et al.[20] has investigated the application of Inter-transaction association rules mining in stock price predication and the possibility of generalizing this method to futures market. They have compared EH-Apriori and FITI (First Intra then Inter) algorithms according to their algorithmic structures and itemsets used; they found that FITI is much better than EH-Apriori. FITI generates many extra and meaningless rules and makes the process complex. Thus the researches have stated another technique called granule -based transactions to have efficient mining process. It uses sliding window setting on decision attribute or constraints and uses SUM Measure. Due to sliding window set only for decision attribute less memory is required. It also scans the data twice same as with FITI. It becomes more applicable in case of large database.

Rajesh V. Argiddi et al.[4] has worked for analysing the behaviour of the stock market data and based on this data to predict the future trading of the stock market. The researchers used dataset of BSE, different companies such as Infosys, TCS and Oracle etc. from Yahoo Finance to find the association among the large scale IT companies and small scale IT companies. Their aim in this research was to find dependencies among different IT companies in the stock market and generate their rules. In this paper they used high values of the shares and have applied fragment based mining algorithm to generate some useful rules which influences the behaviour of the stock market.

By some experimental analysis they have found that fragment based approach generates more generalized rules as compared to FITI approach. Fragment based mining groups all the attributes once and performs the operation group wise instead of single attribute, which results into more generalized rules. Also time needed to process the data is less as we reduce the size of the input table. The rules generated from fragment based approach can be recommended to the customers who invest their money in the stock market [5].

Sanjeev Rao et al. [6] presented the use of an ARM (Association rule mining) driven application is to manage retail businesses. The intent was to provide retailers reports regarding prediction of product sales trends and customer behaviour. The researchers studied two algorithms specially Apriori and FP-growth on the dataset of super market and concluded that FP-growth is faster than other association mining algorithms and is also faster than tree- Researching. The algorithm reduces the total number of candidate item sets by producing a compressed version of the database in terms of an FP-tree.

Hitesh et. al. [18] recommended a framework called InterTARM on real datasets to find out inter-transaction association rules. InterTARM can find inter-transaction association rules for the dataset and to analyse the movement of stock price with the usage of defined minimum support and confidence value. They used historical data and closing price and traded quantity. They used FP-tree concept in ARM of mega-transactions. They also applied effective pre-processing, pruning, techniques and sliding window concept to mine inter-transaction association rules. They experimented and evaluated that as size of the sliding window increases number

of items in mega-transactions are also increased and also takes more items for execution.

III. LITERATURE REVIEW FINDINGS

Several research papers have been presented using the data of various stock exchanges like National Stock Exchange India, Thailand Stock Exchange, Dhaka Stock Exchange, China Stock Exchange, Tokyo Stock Exchange etc. As the stock market is very dynamic, sensitive and volatile in nature, it is essential to make continuous research for achieving some concrete and important rules as well as some easy way to understand and predict the stock movement. We can get some meaningful information using association rule mining to help the investor make more profit. ARM is one of the most popular techniques to get hidden patterns between different attributes.

We have come to know from the reviewed literature that some researches have used sectors, indices, and equity data for their research. Intra-day and inter-day stock data are collected and analysed. Some researchers have worked on limited companies related to IT sector, bank sector etc. Most researchers have used data of 1 year or 2 years which could be increased so as to provide better and accurate results. Different parameters and methodologies are used by the researchers for finding rules. Most researchers have used apriori, modified apriori, FITI, Fragment based, FP-growth and etc. Also the accuracy of the result has not been measured properly. One more thing we found during the survey is that when we are talking about association rule mining, everyone discusses about finding the frequent patterns and algorithms and methods related with that but no one has discussed about finding the association rules in proper format except the Agrawal in 1994 while he developed apriori [25]. No one has used a hybrid method of two different algorithms FP-growth and algorithm suggested by Agrawal in 1994 for association rule mining.

IV. RESEARCH METHODOLOGY

The source of data is NSE historical data. The end-of-day data between the 1st Jan 2010 to 30th Sep 2016 is used. Database consists of the Symbol of company, Date, Prev close, Open Price, High Price, Low Price, Last Price, Close Price, Average Price, Total Traded quantity. This research is based on close price, so we have taken symbol of company, date and close price for analysis purpose. We have collected information for 1250 stock companies which are in .csv format but before processing it we have to do some data pre-processing tasks and then after collected data is converted into proper format for further research.

A. Data Pre-processing

Data pre-processing is applied on the collected data which is in .csv format. Some of the major tasks in data pre-processing are mentioned below.

1) Cleaning

The source of data consists of 1250 symbols (for stock companies). But we have taken only those scripts which have less missing values. Missing values are greater in numbers i.e. missing values for consecutive six months or more gives wrong result so they have been avoided. We have filled other missing values by using binning method. We have calculated average of 10 records located in the neighbour.

2) Data Integration and Data Transformation

Data integration combines data from multiples sources to form a coherent data store. So we have 15 to 17 lacks rows. But we need data in different format so it is needed to transform data. We have data in the form of date in row and close price in column and in the cell value for particular day close price. But then we have transformed the data to convert in the form of date in row and script symbol in column with the cell value end of date close price. Data transformation was done to create database. After transformation we have worked on total traded days 1499 and around 700 symbols (stock companies).

3) Attribute Construction

We have derived the per_chg attribute which shows difference with the previous and current day close price in percentage. We have also derived an attribute numeric code for different criteria of per_chg which correlates the performance of one script to another.

4) Data Reduction

Discretization is also used to obtain a reduced representation of the data while minimizing the loss of information content. We have generated numeric code for representing multi-dimensional data which is very necessary while doing association rule mining. We created two datasets as shown in below table 1(with 2% change) and table 2 (with 0.5% change) in bin value.

Table I CODES GIVEN TO EACH CRITERION FOR SET WITH DIFFERENCE 2 %

Bin Number	*Criteria for per_chg*	*Bin Value*
Bin 1	>=8	9
Bin 2	>=6 AND <8	8
Bin 3	>=4 AND <6	7
Bin 4	>=2 AND <4	6
Bin 5	>=0 AND <2	5
Bin 6	>=-2 AND <0	4
Bin 7	>=-4 AND <-2	3
Bin 8	>=-6 AND <-4	2
Bin 9	>=-8 AND <-6	1

Table II CODES GIVEN TO EACH CRITERION FOR SET WITH DIFFERENCE 0.5 %

Bin Number	*Criteria for per_chg*	*Bin Value*
Bin 1	>=4.5	9
Bin 2	>=4 AND <4.5	8
Bin 3	>=3.5 AND <4	7
Bin 4	>=3 AND <3.5	6
Bin 5	>=2.5 AND <3	5
Bin 6	>=2 AND <2.5	4
Bin 7	>=1.5 AND <2	3
Bin 8	>=1 AND <1.5	2
Bin 9	>=0.5 AND <1	1
Bin 10	>=0 and <0.5	0
Bin 11	>=-0.5 AND <0	-1
Bin 12	>=-1 AND <-0.5	-2
Bin 13	>=-1.5 and <-1	-3
Bin 14	>=-2.0 and <-1.5	-4
Bin 15	>=-2.5 and <-2.0	-5
Bin 16	>=-3.0 and <-2.5	-6
Bin 17	>=-3.5 and <-3.0	-7
Bin 18	>=-4.0 and <-3.5	-8
Bin 19	<-4.0	-9

This criterion is created by difference with 0.5 bin values in per_chg as shown in table 2. We generally believe that as the difference is small, more variety of rules can be generated. Because of this small difference we can also have the idea before trading how much profit is associated with the particular relation between scripts. Before this we have tried to find out all the association rules with 2% difference in per_chg as shown in table 1. We thought that as the range is high more number of rules will be generated but we were wrong and we just found the rules with same kind of industries like gold and bees and we avoided those scripts for our next experiments.

Table III SHOWS THE SCRIPT CODE REQUIRED BY INPUT FILE WITH DATASET2

Sc1_Code	*Sc2_Code*	*Sc3_Code*	*Sc4_Code*	*Sc5_Code*
-14	23	-31	-63	94
-13	23	-33	62	-93
10	-25	31	-64	99
19	21	-31	65	94
16	29	-31	60	95
-14	-28	-31	61	-97
-15	21	30	-63	95
10	-21	-31	64	-93
12	23	-31	-66	92
-15	-24	30	-61	-94

In above table III, -14 indicates that the code is for script 1 and -4 is the bin value as shown in table 1 for first date.

B. Selection of Algorithm

According to the Agrawal 1993, Association rule mining has two steps (1) Find all the item sets with the support greater than the minimum support (frequent items) (2) Based on the above obtained frequent set, all the association rules will be generated.

Mining frequent patterns is introduced first by Jiawei Han et al. [3] in transaction databases. They proposed a new frequent pattern tree (FP-tree) structure, and developed FP-tree-based mining method. It proves it is better than Apriori based algorithms on three fronts: (1) a large database is compressed into a highly condensed, much smaller data structure, which avoids costly, repeated database scans, (2) our FP-tree-based mining adopts a pattern fragment growth method to avoid the costly generation of a large number of candidate sets, and (3) a partitioning-based, divide-and-conquer method is used to decompose the mining task into a set of smaller tasks for mining confined patterns in conditional databases, which dramatically reduces the search space.

The apriori-based algorithms are most popular algorithms that can be shown by the reviewed paper[10]. Number of researchers has tried to apply modified version of Apriori [11][19][20][21] but the disadvantage of it is that it requires costly repeated scan and generation of large candidate sets. FP-growth method is an efficient and scalable algorithm for mining both long and short frequent patterns and main advantage of it is that it avoids costly candidate-set generation process by generating FP-tree [3][6][13][18].

In this proposed research we have tried to combine the core concept of both the most famous algorithms. ARM is traditionally performed in two steps : (1) Mining frequent itemsets and (2) Generating association rules by using frequent itemsets. In this implementation, we have used the FP-growth algorithm for Step 1 because it is very efficient. For Step 2, we have used the Discovery Rule Algorithm proposed by Agrawal and Srikant in 1994 [2].

1) Agrawal's Apriori and 94 Discovery Rule Algorithms

i) APRIORI algorithm

Apriori is the algorithm invented by Agrawal in 1993. It uses generate and test approach. It first generates the candidate itemsets and test if they are frequent. Generation of candidate itemsets requires repeated scan so it is very expensive in two perspective both space and time. Support counting is also very expensive. Subset checking is also

expensive. The Apriori discovers the frequent itemsets which shows market trend.

Apriori uses bottom-up approach, where frequent subsets are extended known as candidate itemsets. Then tests the data until there are not successful extensions are found.

Apriori uses breadth-first search and hash tree data structure to count candidate item sets successfully. It generates candidate itemsets of length k from item sets of length K + 1. Then it prunes the candidates which have an infrequent sub pattern.

Procedure Apriori(t, minSup)

L1= Frequent items of length 1

For (k=1; L_k != ϕ ; k++) do

Ck+1 = Candidates generated from Lk.

For each transaction t in database do.

Increment the count of all candidates in Ck=1

that are contained in t.

Lk+1 = Candidates in Ck+1 with minimum

support.

End for each.

End for

Return the set Lk as the set of all possible frequent item-sets.

ii) Agrawal's discovery algorithm

Agrawal's discovery algorithm's functionality is to use the large itemsets extracted from Apriori and generate the desired rules.

Here is a straightforward algorithm for this task.

1) For every large itemset l, find all non-empty subsets of l.
2) For every such subset a, output a rule of the form a => (l - a) if the ratio of support(l) to support(a) is at least minconf.
3) We need to consider all subsets of l to generate rules with multiple consequents.

iii) FP-growth Algorithm

FP-growth uses compact data-structure called FP-tree and extracts frequent itemsets directly from this structure. An FP-tree is a compressed representation of the input data. It is constructed by reading the data-set one transaction at-a-time and mapping each transaction onto a path in the FP-tree. As different transactions can have several items in common, their paths may overlap. The more the paths overlap with one another, the more compression we can achieve using the FP-tree structure using depth-first logic. If the size of the FP-tree structure is small enough to fit into main memory, this will allow us to extract frequent itemsets directly from the structure in memory instead of making repeated passes over the data stored on disk.

Advantages of FP-growth are that it only requires 2 passes over data-set. It compresses data-set. It does not require candidate generation but it is much faster than Apriori in overall effect. FP-tree requires time to build but once it is built, it is also easy to find frequent itemsets from FP-tree but the

Algorithms for discovering large itemsets make multiple passes over the data. In the first pass, we count the support of individual items and determine which of them are large, i.e. have minimum support. In each subsequent pass, we start with a seed set of itemsets found to be large in the previous pass. We use this seed set for generating new potentially large itemsets, called candidate itemsets, and count the actual support for these candidate itemsets during the pass over the data. At the end of the pass, we determine which of the candidate itemsets are actually large, and they become the seed for the next pass. This process continues until new large itemsets are found.

Notation :

We assume that items in each transaction are kept sorted in their lexicographic order.

k-itemset :An itemset having k items.

L_k :Set of large k-itemsets (those with minimum support).

Each member of this set has two fields: i) itemsets and ii) support count.

//Simple Algorithm

For all large itemsets l_k, k>=2 do

Call genrules(l_k, l_k);

//The genrules (l_k: large k-itemset, a_m : large m-itemset)

A = { (m-1) - itemsets a_{m-1} |$a_{m-1} \subset a_m$ };

For all $a_{m-1} \in$ A do begin

a. Conf = support (l_k) / support (a_{m-1});
b. If (conf >= minconf) then begin

Output the rule $a_{m-1} \Rightarrow (l_k - a_{m-1})$, with confidence = conf and support = support(l_k);

If (m-1 > 1) then

Call genrules(l_k, a_{m-1});//to generate rules with subsets of a_{m-1} as the antecedents

c. end

end

disadvantage is that FP-Tree may not fit in memory and it is very expensive to build.

1) The first phase of frequent pattern tree approach is the construction of a frequent pattern tree. The tree is constructed as follows:
    a) Scan the original database once to identify set L1 and then rearrange L1 in the descending order of support count (count).
    b) Grow a tree with an empty root node (null). During the second scan, starting from the null root node, add paths to the tree corresponding to the reordered transactions (transaction whose items have been reordered according to the descending order property of L1), updating the node's count. For transactions that share common ancestor nodes, use the existing nodes in the tree and update the node's count suitably. In cases where a common ancestor node is not present, new nodes are added from the null root node and the procedure repeated as mentioned above.
2) The second phase of FP tree algorithm is to mine or generate frequent patterns from the constructed FP tree (earlier step output).
    a) For every element (reverse order of descending order L1), now locate positions in the tree where the nodes appear in the tree and list the paths as the conditional pattern bases, with the considered element treated as a common suffix.
    b) From each of the conditional pattern bases, determine the cumulative frequent pattern tree counts.
    c) Finally, mine the frequent patterns by suffixing L1's element with the frequent pattern tree values obtained in the earlier step.

iv) HybridFPgrowth Algorithm

According to the Agrawal 1993, Association rule mining has two steps (1) Find all the item sets with the support greater than the minimum support, which is called frequent itemsets (2) Based on the above obtained frequent itemsets, all the association rules will be generated.

As per the reviewed papers, we found that there are three basic algorithms, Apriori, Eclat and FP-growth for frequent pattern mining. Out of these three, FP-growth is the best among the three algorithms in terms of performance because it is most scalable. Eclat performs poorer than FP Growth and the Apriori performs the worst [23].

Finding the frequent patterns is a very important step in ARM algorithms. Here, the problem is that from the frequent patterns how can we find out association rules to get important hidden rules because without association rules we cannot exactly specify the trend and mining the qualitative rules which is also necessary task in business application. To get the qualitative rule we should have some technique to find out association rules from frequent patterns from FP-growth. As FP-growth algorithm comes in two phases. The first phase is for generation of a frequent pattern tree and the second phase is for finding frequent patterns from the constructed FP tree. The major requirements of the data mining applications are for finding the association rule mining. So, in our research, we have added one more phase in FP-growth for extracting association rules and proposed a novel hybrid method to combine FP-growth with Agrawal's Discovery rule algorithm and named this algorithm to HybridFPgrowth algorithm.

1) The first phase is for generation of a frequent pattern tree.
2) The second phase is for finding frequent patterns from the constructed FP tree.
3) The third phase is for extracting association rules from frequent pattern.

HybridFPgrowth Algorithm

1) The first phase is for generation of a frequent pattern tree.

    Step-1: First of all Scan the database to define frequency of each item and prepare set L1 as same as Apriori algorithm. This step is for filtering out the infrequent items
    Step-2: Sort the L1 in descending order as their support counting.
    Step-3: During second scan of the database, a tree is grown starting from an empty root node that is null node.
    Step-4: Add paths to the tree as per the sorted transactions (sorting done in step-2) and update the nodes count.
    Step-5: For the first transaction path from root node to its count set 1 new node is added.
    Step-6: For the transactions, where ancestor nodes are present, use the existing nodes in the tree and update the node's count.
    Step-7: Repeat steps 4 to 6 until there is not item remained in list L1.
2) The second phase is mining FP-tree for finding frequent patterns. (FP tree algorithm also maintains an item header table to aid the process of tree traversal and pattern extraction.)

    Step-1: Start from each frequent 1 item-set or pattern (from the last element of reordered L1) and construct its conditional pattern base. (all those paths in the tree that finally lead to the considered element of L1.)
    Step-2: From each of the conditional pattern bases, determine the cumulative pattern tree counts.
    Step-3: From conditional pattern base, patterns are constructed recursively satisfying the support threshold retaining those that satisfy the minimum support threshold.
3) The third phase is for extracting association rules from frequent patterns which is added with the purpose to

achieve qualitative rules. This idea is suggested by Agrawal in 1994 to get the frequent patterns from apriori algorithm [2]. But we want to merge this algorithm into FP-growth to acquire qualitative rule. (The process of finding association rules is a multi-pass.)

First, we sort all itemsets having the same size by lexical order for optimization purpose.

Step-1: In the first pass, we count the support of individual items and determine which of them are large, i.e. have minimum support. (For every large itemset l, find all non-empty subsets of l.)
Step-2: In each subsequent pass, we start with a seed set of itemsets found to be large in the previous pass. We use this seed set for generating new potentially large itemsets, called candidate itemsets, and count the actual support for these candidate itemsets during the pass over the data. (For every such subset a, output a rule of the form a => (l - a) if the ratio of support(l) to support(a) is at least minconf.)
Step-3: In the final pass, we determine which of the candidate itemsets are actually large, and they become the seed for the next pass. This process continues until new large itemsets are found. (We need to consider all subsets of l to generate rules with multiple consequents.)

C. *Accuracy*

The accuracy of rules is very important. We have used hold-out method for accuracy determination in which two third of the data were allocated to training set (data from 01-Jan-2010 to 01-Dec-2013) and remaining one third was allocated to test set (data from 01-Jan-2014 to 30-Sep-2016). The training set was utilized for deriving the association rules and the accuracy was checked with the test set.

V. RESULTS AND ANALYSIS

We have experimented data with Intel Core i3-4005U CPU @1.70GHz processor and 4 GB RAM. We have implemented the algorithms in java and taken help from the LUCS-KDD software Library and www.github.com. We have implemented our code in Netbeans 8.0.2 with latest Java 8 technologies--Java SE 8, Java SE Embedded 8, and Java ME Embedded 8.

We have generated two different data sets dataset1 with 2% change as shown in table I and dataset2 with 0.5% change as shown in table II.

We have experimented hybrid algorithm with dataset1 but we did not get interesting rules. So we have generated dataset2 and got the interesting rules with higher accuracy. We have used result achieved from dataset1 for comparison of experiments and results shown in Fig 2, and 3 and 4 because our dataset2 contains negative numeric code for negative bin values because we have large number of categories for 0.5 differences in percentage.

We have seen that these multi-categorical data cannot return proper rules with Apriori association rule algorithm. Both algorithms found same set of rules when input file is with only positive bin values but for input file with negative bin value we got accurate result with the usage of HybridFPgrowth only. We have observed that HybridFPgrowth performs well with negative numeric code as input data and we can say that HybridFPgrowth gives accurate result.

For the experiments with dataset1, we have fixed the minconf with 0.2 and set minsup with different values 0.2 to 0.9 and measured the runtime for finding the frequent itemsets, for finding association rules and for finding association rules in overall. We have shown the runtime data in tabular form, done comparison by chart and analysed with both the algorithm taken by Apriori Association Rule and HybridFPgrowth Association Rule.

Table IV RUNTIME PERFORMANCE BASED ON CPU-TIME FOR FINDING FREQUENT PATTERNS

(Above values are in ns (nanoseconds) in millions)

Minsup	*HybridFPGrowth*	*Apriori*
0.2	1016	7047
0.3	1063	7063
0.4	1141	7438
0.5	1063	6859
0.6	1172	7078
0.7	1016	7109
0.8	922	7250
0.9	1047	7219
1	1094	7063

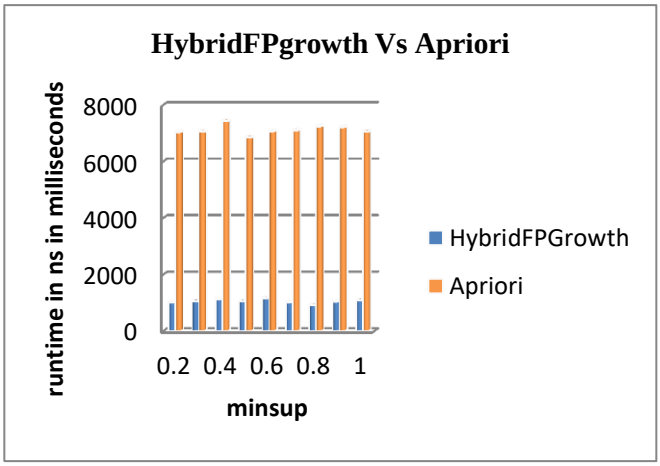

Figure 2 Comparison of Runtime Performance based on CPU time for Finding Frequent Patterns

Comparison of the runtime performance on the bases of CPU time using both the algorithms FP-growth and Apriori is shown in Figure 2. Table 4 shows the runtime for finding Frequent Patterns from the database in nanoseconds in millions. We have shown the runtime based on CPU time for pass 1 using both the algorithms HybridFPgrowth and Apriori. We can see that HybridFPgrowth is faster and efficient than Apriori.

Table V RUNTIME OVERALL PERFORMANCE BASED ON CPU-TIME FOR FINDING ASSOCIATION RULES

(Above values are in nanoseconds in millions)

Minsup	*HybridFPgrowth*	*Apriori*
0.2	1063	7078
0.3	1109	7094
0.4	1188	7469
0.5	1094	6891
0.6	1203	7109
0.7	1063	7141
0.8	953	7281
0.9	1078	7234
1	1109	7078

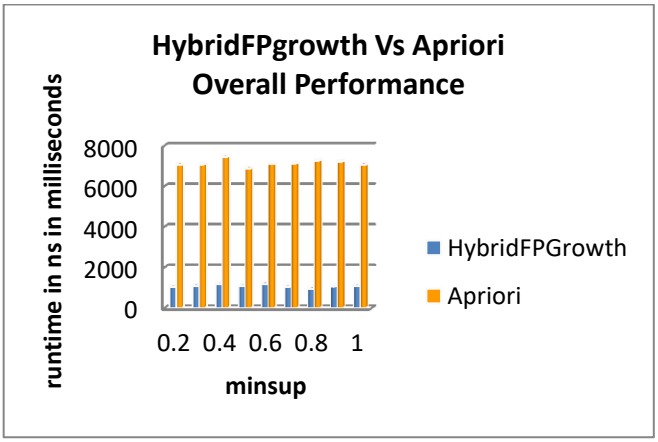

Figure 3 Comparison of Runtime Overall Performance based on CPU time for Finding Association Rules

Comparison of the runtime overall performance on the bases of CPU time using both the algorithms HybridFPgrowth and Apriori is shown in Figure 3. Table 5 shows the runtime for finding frequent patterns and association rules from frequent patterns from the database in nanoseconds in millions. We have displayed the runtime based on CPU time for step 1 and step 2 of association rule mining using both the algorithms HybridFPgrowth and Apriori. We can see that HybridFPgrowth is so faster and efficient.

Table 6 Runtime Performance based on Number of Records for Finding Association Rules .

(Above values are in nanoseconds in millions)

No of Records	HybridFPGrowth	Apriori
500	5250	289516
400	4422	243781
300	4203	181922
200	2938	134047
100	2406	90750

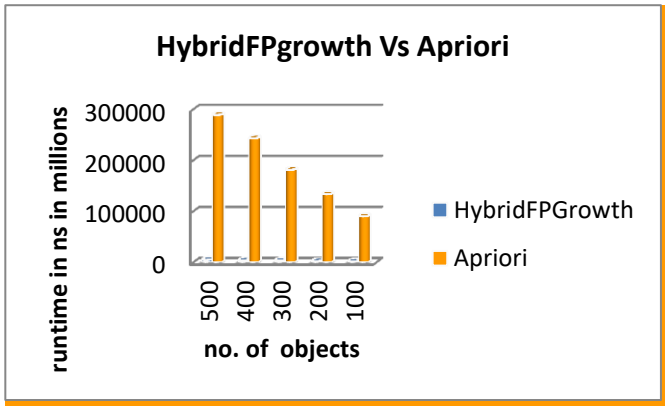

Figure 4 Comparison of Runtime Performance based on Number of Objects for Finding Association Rules

Comparison of the runtime performance on the based on number of records measuring by both the algorithms HybridFPgrowth and Apriori is shown in Figure 4. Table 6 shows the runtime for finding frequent patterns and association rules from frequent patterns from the database in nanoseconds in millions. We have shown the runtime based on record numbers using both the algorithms HybridFPgrowth and Apriori. We can see that HybridFPgrowth is faster and efficient than Apriori because when the change occurs in number of records (in hundreds) very little change occurs in runtime.

In our research, We have used java.lang.management package introduced by Java 1.5 to monitor the JVM. For measuring efficiency of our hybrid method, we have compared the result and found that CPU time of HybridFPgrowth is very less than Apriori for finding the frequent patterns. The overall CPU time required by HybridFPgrowth is also very less than Apriori to find association rules.

We have found that the rules for on dataset2 with 0.5% change. we have fixed the minconf with 0.8 and set minsup with different values 0.03 and measured the runtime for finding association rules. We got so many concrete rules and presented the rules whose accuracy is greater than or equal to

80% and total 79 rules are very prominent . We could get the knowledge from the rules which show one script dependencies with other scripts on the bases of criteria. We have shown only 10 rules in below table 7.

Table 7 Association Rules achieved by HybridFPgrowth

Sr. No.	*Association Rules*	*Freq. of Rule*	*Accu-racy (%)*
1	-1224 ==> 3049 When BALAJITELE goes down between >=2.0 and <1.5 % then ECEIND also goes up by 4.5 %	25	100
2	-9129 -7159 ==> -4599 When RELCAPITAL and MAXWELL go down by 4% then HDIL also goes down by per < 4 %	24	100
3	-8509 -5109 ==> -4549 When PHILIPCARB and IDBI go down by 4.0 % then HCL-INSYS also goes down by 4.0 %	21	100
4	-9823 -6963 -2313 ==> -4273 When RELINFRA ,MAHABANK, CLNINDIA go down between >=1.5 and <1 then GRUH also goes down between >=1.5 and <1	21	100
5	-10609 -8719 ==> -7159 When TALBROAUTO and RELINFRA go down by 4.0 % then MAXWELL also goes down by 4.0 %	20	100
6	-8749 -3583 ==> -4033 When PREMIER go down by 4.0% and FORTIS goes down between 1.5 and <1.0 % then GODREJCP also goes down by 4.0 %.	20	100
7	-7529 -2669 ==> -4929 When MUKANDENGG and DCW go down by 4.0 % then HOCL also goes down by 4.0 %	20	100
8	-10609 -8719 ==> -7159 When TALBROAUTO and PRATIBHA go down by 4.0 % then MAXWELL also goes down by 4.0 %	20	100
9	-5109 -3819 ==> -3699 When IDBI and GAMMONIND go down between 1.5 and 1.0 % the GENUSPOWER also goes down between 1.5 and 1.0%	20	100
10	-4599 -4549 -4529 ==> -10869 When MAXWELL and HCL-INSYS and HCC go down by 4.0 % then TFCILTED also goes down by 4.0 %.	20	100

We found number of rules associated with different scripts (symbols) and give prominent information about the variations in script's prices. This research has tried to find out the associations between scripts to analyse the movement in the STOCK Market. We found the knowledgeable rules which show one script dependencies with other scripts on the bases of different criteria.

VI. CONCLUSION

Our work consists of generating accurate association rules on the stock market data provided by National Stock Exchange. In this research we have downloaded end-of-day data for six years from 01st January, 2010 to 30th September, 2016 from NSE. We have applied necessary pre-processing steps like cleaning, integrating, transforming etc. Our data is multi-dimensional and that is why discretization became very important and is done by binning method. The research has used a Novel approach which combines two most popular algorithms FP-growth to find frequent item-sets and Algorithm proposed by Agrawal and Srikant in 1994 to find association rules. The rules generated by this hybrid method have given satisfactory results. We found so many rules with 100% accuracy using this method but we have selected those rules for analysis and interpretation which have greater than or equal to 80% accuracy. Out of 716 scripts of companies we got relationships between 150 scripts of different categories with each other to produce meaningful co-relations. We observed a striking fact that when minsup is low and frequent patterns are too high, Apriori association rule algorithm takes too much time to execute and its performance becomes poorer than HybridFPgrowth. Both algorithms found same set of rules when input file is with only positive bin values but when input file with negative bin value is given, we got accurate result with the usage of HybridFPgrowth only. We found number of rules associated with different scripts (symbols) and give prominent information about the variations in script's prices. We are very confident about our rules that these rules will definitely guide the investors to invest their money and get higher profit.

REFERENCES

[1]. RakeshAgrawal, Tomasz Imielinski, and Arun N. Swami, "Mining Association Rules Between Sets of Items in Large Databases", Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, pp. 207-216, Washington, D.C., May 1993.

[2]. Rakesh Agrawal and Ramakrishna Srikant, "Fast Algorithms for Mining Association Rules", Proceedings of the Twentieth International Conference on Very Large Databases, pp. 487-499, Santiago, Chile, 1994.

[3]. Jiawei Han, Jian Pei, Yiwen Yin, " Mining frequent patterns without candidate generation ", Proc. of ACM SIGMOD International Conference on Management of Data, pp. 1- 12, Volume 29 Issue 2, June 2000.

[4]. Rajesh V. Argiddi, Sulabha S. Apte, "Fragment Based Approach to Forecast Association Rules from Indian IT Stock Transaction Data" IJCSIT, Vol 3(2), pp. 3493-3497, 2012

[5]. Rajesh V. Argiddi, Sulabha S. Apte," AN EVOLUTIONARY FRAGMENT MINING APPROACH TO EXTRACT STOCK MARKET BEHAVIOR FOR INVESTMENT PORTFOLIO", (IJCET) ISSN 0976 – 6367(Print),ISSN 0976 – 6375(Online) Volume 4, Issue 5, September – October (2013), pp. 138-146, 2013
[6]. SanjeevRao, Priyanka Gupta, ―Implementing Improved Algorithm Over APRIORI Data Mining Association Rule Algorithml, ISSN: 0976-8491 (Online) | ISSN: 2229-4333 (Print) IJCST Vol. 3, Issue 1, Jan. - March 2012.
[7]. K Utthammajai and P. Leesutthipornchai, "Quality-based Association Rules for Stock Index Data by using Rough Set Theory" ,Proc.12th IEEE International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2015 , pp. 1 - 6, 2015.
[8]. A. Srisawat, "An Application of Association Rule Mining Based on Stock Market," Inter. Proc. 3rd IEEE International Conference on Data Mining and Intelligent Information Technology Application (ICMiA), pp. 259 - 262, 2011.
[9]. A. Galib, M. Alam, N. Hossain, R. Rahman," Stock Trading Rule Discovery Based on Temporal Data Mining",Proc. IEEE International Conference on Electrical and Computer Engineering (ICECE), pp. 566 - 569, 2010
[10]. Shona Ulagapriya, Dr. P Balasubramanian, "Study on Inter sector Association rules in National Stock Exchange, India", Proc. IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 859 - 865, 2015
[11]. Priti Saxena, Bhaskar Pant, R.H. Goudar, " Inter – Transactional Pattern Discovery Applying Comparative Apriori and Modified Reverse Apriori Approach", Proc. IEEE 8th Proceedings International Conference on Intelligent Systems and Control (ISCO), pp. 300-305 2014
[12]. A.Asbern, P.Asha," Performance Evaluation Of Association Mining In Hadoop Single Node Cluster With Big Data", Proc. IEEE International Conference on Circuit, Power and Computing Technologies [ICCPCT], pp. 1-5 ,2015
[13]. Aurangzeb Khan, Khairullah khan, Baharum B. Baharudin, "Frequent Patterns Mining Of Stock Data Using Hybrid Clustering Association Algorithm", IEEE International Conference on Information Management and Engineering (ICIME), pp. 667 – 671, 2009
[14]. Jiayi Yao, Shuhui Kong,"The Application of Stream Data Time-Series Pattern Reliance Mining in Stock Market Analysis", IEEE International Conference on Service Operations and Logistics, and Informatics, pp. 159 – 163, 2008
[15]. Harya Widiputra**,** BagusPahlevi**,** "Inter-transaction Association Rule Mining in the Indonesia Stock Exchange Market", IEEE International Conference on Uncertainty Reasoning and Knowledge Engineering, pp. 149 – 152, 2012
[16]. Sheikh Shaugat Abdullah and Mohammad SaiedurRahaman, "Stock Market Prediction model using TPWS and Association Rules Mining", IEEE 15th International Conference on Computer and Information Technology (ICCIT), pp. 390-395, 2012
[17]. Chirag A. Mewada, Rustom D. Morena, "Model using Improved Apriori Algorithm to generate Association Rules for Future Contracts of Multi Commodity Exchange (MCX)", International Journal of Advanced Research in Computer Science, Volume 8, No. 3, ISSN No. 0976-5697, March – April 2017
[18]. Hitesh Chhinkaniwala, P.Santhi Thilagam, "InterTARM: FP-tree based Framework for Mining Inter-transaction Association Rules from Stock Market Data", 978-0-7695-3308-7/08 $25.00 © 2008 IEEE, DOI 10.1109/ICCSIT.2008.173
[19]. Goswami D.N., Chaturvedi Anshu. ,Raghuvanshi C.S., "An algorithm for Frequent Pattern Mining Based on Apriori.", (IJCSE) International Journal on Computer Science and Engineering, Vol. 02, No. 04, pp. 942-947, ISSN : 0975-3397, 2010
[20]. Kranti M. JaybhayÀ, R.V.ArggiddiÀ, " A Comprehensive Overview of ARM Algorithms in Real Time Inter Transactions", International Journal of Current Engineering and Technology, Vol.4, No.4 (Aug 2014), E-ISSN 2277 – 4106, P-ISSN ,pp. 2347 – 5161, 2014
[21]. Praveen Pappula, Ramesh Javvaji, " Experimental Survey on Data Mining Techniques for Association rule mining", International Journal of Advanced Research in Computer Science and Software Engineering 4(2), pp. 566-571, 2014
[22]. Lijuan Zhou, Xiang Wang, "Research of the FP-Growth Algorithm Based on Cloud Environments", JOURNAL OF SOFTWARE, VOL. 9, NO. 3, MARCH 2014
[23]. Dr. Kanwal Garg, Deepak Kumar, "Comparing the Performance of Frequent Pattern Mining Algorithms"**,** International Journal of Computer Applications (0975 – 8887) Volume 69– No.25, May 2013
[24]. Rahul Thakkar, "Data Mining Techniques and Stock Market ", International Journal of World Research, Vol: I Issue XIII, December 2008, Print ISSN: 2347-937X
[25]. Pandya Jalpa P., Morena Rustom D.," A Survey on Association Rule Mining Algorithms Used in Different Application Areas", International Journal of Advanced Research in Computer Science, Volume 8, No. 5, May-June 2017, ISSN No. 0976-5697
[26]. N. P. Gopalan, B. Sivaselvan, "Data Mining Techniques and Trends", PHP Learning Private Limited, New Delhi-110001, ISBN-978-81-203-3812-B, 2009.
[27]. Han Jiawei; Kamber Micheline, "Data Mining Concepts and Techniques", Second Edition, Morgan Kaufman, pp. 227 – 378, 2006.
[28]. G.K. Gupta, "Introduction To Data Mining With Case Studies", Second Edition, PHP Learning Private Limited, New Delhi-110001, ISBN-978-81-203-4326-9, 2011.
[29]. http://www.nseindia.com/
[30]. http://codereview.stackexchange.com/questions/125372/mining-association-rules-in-java
[31]. https://cgi.csc.liv.ac.uk/~frans/KDD/Software/
[32]. www.github.com
[33]. https://en.wikipedia.org/wiki/National_Stock_Exchange_of_India
[34]. http://www.ijctjournal.org/Volume2/Issue3/IJCT-V2I3P15.pdf
[35]. http://www.world-stock-exchanges.net