Big Data Framework for Finding Patterns in Multi-market Trading Data

Daya Ram Budhathoki, Dipankar Dasgupta(✉), and Pankaj Jain

University of Memphis, Memphis, TN, USA
{dbdhthki,ddasgupt,pjain}@memphis.edu

Abstract. In the United States, multimarket trading is becoming very popular for investors, professionals and high-frequency traders. This research focuses on 13 exchanges and applies data mining algorithm, an unsupervised machine learning technique for discovering the relationships between stock exchanges. In this work, we used an association rule (FP-growth) algorithm for finding trading pattern in exchanges. Thirty days NYSE Trade and Quote (TAQ) data were used for these experiments. We implemented a big data framework of Spark clusters on the top of Hadoop to conduct the experiment. The rules and co-relations found in this work seems promising and can be used by the investors and traders to make a decision.

Keywords: Multimarket · Exchanges · Association rules
FP-Growth · Hadoop · Spark · TAQ · Clusters

1 Introduction

In Multimarket, securities listed in one exchange can also be listed in another exchange, and it can be traded on more than one exchanges [10]. Small liquidity traders generally trade in one trade but large traders split their trades across markets. In this work, we apply a data mining technique based on an FP-Growth algorithm to find out an interesting pattern in multimarket trading.

1.1 Security Information Processor (SIP)

US equities market is highly competitive and fragmented consisting of 13 exchanges and about 40–50 Alternative Trading System (ATS)/dark pools. Security Information Processor (SIP) was created to have a National Market System where investors and professionals can have access to the real time information related to quote (bid and offer) and trade (Fig. 1).

The SIP is operated by NASDAQ and New York Stock exchanges which creates a real-time consolidated record of every exchanges. The SIP is a central consolidated and live stream aggregator, where every stock exchange and ATS sends data stream of the best quotes (bid and offer) and updates public price

F. Y. L. Chin et al. (Eds.): BIGDATA 2018, LNCS 10968, pp. 237–250, 2018.
https://doi.org/10.1007/978-3-319-94301-5_18

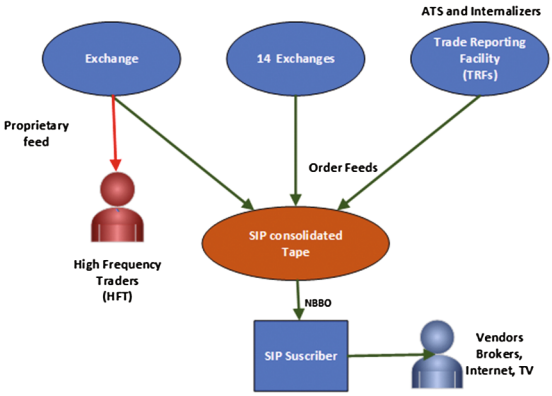


Fig. 1. Architecture of Security Information Processor (SIP)

quotes called the "National Best Bid and Offer" (NBBO) continually. In order to create NBBO, SIP compiles all of the bids and offers for all U.S. stock in one place. All of the exchanges piped the bids and offers into the SIP, and the SIP ultimately calculates the NBBO. SIP is a very easy way to get the current status of the market and acts as a benchmark to determine the NBBO.

Table 1. US stock exchanges

Code	Description	Code	Description
A	NYSE MKT LLC	P	NYSE Arca, Inc.
B	NASDAQ OMX BX, Inc.	S	ConsolidatedTape System
C	National Stock Exchange Inc. (NSX)	T	NASDAQ Stock Exchange, LLC (in Tape A, B securities)
D	Financial Industry Regulatory Authority, Inc. (FINRA ADF)	Q	NASDAQ Stock Exchange, LLC (in Tape C securities)
I	International Securities Exchange, LLC (ISE)	V	The Investors' Exchange, LLC (IEX)
J	Bats EDGA Exchange, INC	W	Chicago Broad Options Exchange, Inc. (CBOE)
K	Bats EDGX Exchange, Inc.	X	NASDAQ OMX PSX, Inc. LLC
M	Chicago Stock Exchange, Inc. (CHX)	Y	Bats BYX Exchange, Inc.
N	New York Stock Exchange LLC	Z	Bats BZX Exchange, Inc.

Table 1 shows the list of exchanges and their corresponding symbols. Exchanges can be classified in terms of Maker-Taker [1].

Table 2. Pricing models

Exchange	Pricing model
NYSE	Maker-taker
ARCA	Maker-taker
Nasdaq PSX	Maker-taker
Direct Edge X	Maker-taker
BATS	Maker-taker
NASDAQ	Maker-taker
Boston	Inverse-Maker-taker
BATS-Y	Inverse-Maker-taker
Direct Edge A	Maker-taker

2 Related Work

Several works have been done in the financial stock market using Association rule mining. Frequent patterns play important roles in association rule mining, finding correlations, and other interesting relationships among data stream [6, 12]. Asadifar [9] presents the application of association rules to predict the stock market. Other works [20,22] used the Deep Learning approach for sentiment analysis and Data Mining of the financial Big Data. In addition, [22] presents Data Mining with Big Data. The focus of our work to build a Big Data framework to process the financial trading data efficiently and apply the association rules to seek the dominance patterns among exchanges (Table 2).

3 Background

3.1 Big Data Analytic

In earlier days, financial data were relatively small as most exchanges only reported Open, High, Low and Close (OHLC) at the end of each day. Now with the advent of high frequency trading in the financial market, the importance of Big Data in finance is increasing day by dayand such data [11] can be characterized by the 5V's of Big data as shown in Fig. 2).

1. **Variety:** It refers to the limitless variety of Big Data. Financial data can be either structured or unstructured. Structured data refers to the information which has fixed structure and length and can be easily represented in a tables in the form of rows and columns. The unstructured data can not be organized into a table (with rows and columns) and does not fall in a pre-determined model. Examples include gathered data from the social media posts, logs and even audio and videos.

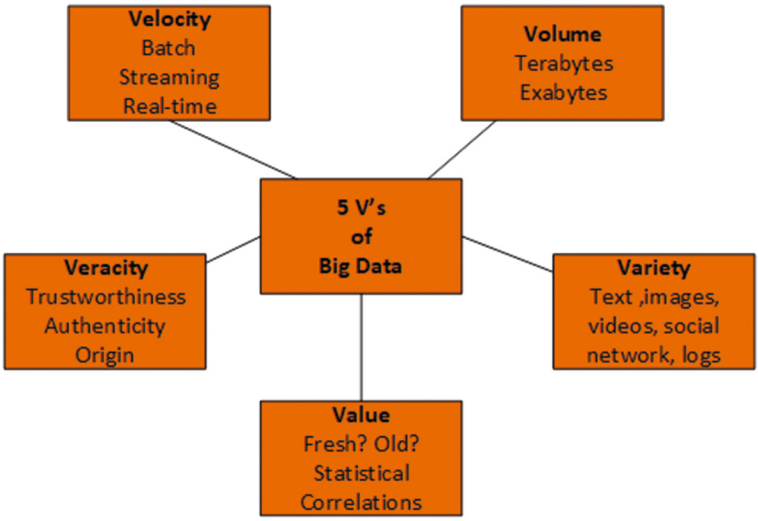


Fig. 2. 5V's of stock market big data [2]

2. **Veracity:** It refers to the truthfulness or accuracy of the data. It can be defined as the bias or abnormality in the data. Generally, 40 to 50% of the time is spent on data preparation cleansing.
3. **Volume:** It refers to the vast amount of data generated every seconds. It could be in Terabytes, Zettabytes or even higher. For example, single day NYSE quote file for August 24, 2017 is 203 GB. The high-frequency data of the financial stock market at each day consists of the information equivalent to 30 years of daily data [8].
4. **Velocity**: Velocity refers to the speed at which data is being generated or produced. As an example, we can think of social media messages that goes viral in few seconds. In the context of financial market, it can be thought of as a high-frequency trading data generated in microseconds. With the Big Data technologies, we can analyze these data as they are being generated very efficiently.
5. **Value:** It refers to the ability to turn the Big Data into business value. For businesses, it's really important to make use of cases before jumping to collect and store the big data. If we were not able to turn the Big Data into usable business value, it's useless.

3.2 Big Data Platform for Stock Market Analysis

In order to process huge amounts of both structured and unstructured financial data, Big Data technologies such as Apache Hadoop and Spark are essential since the conventional relational database and data warehousing system can not handle these efficiently. Big Data platform has been extensively used in the stock market to find the patterns or trends and ultimately predicts the outcome

of the certain behavior in the financial market. Stock market data can be both structured and unstructured and have properties of 5V's of big data as in Fig. 2. The financial institution can extract information and process and analyze them to help investors in trading decisions. For example, by analyzing the changes in Google query volumes for big company names such as Apple, IBM, Microsoft, etc., we can find an interesting pattern that can be interpreted as a warning sign for stock market movements [18].

3.3 Apache Hadoop

Apache Hadoop is an open source implementation of Google File System and Map Reduce, which is a software platform used for distributed data storage and processing. Hadoop has mainly two components, MapReduce and Hadoop Distributed File System (HDFS). It was designed to store very large data sets efficiently and reliably in a cluster and to stream data with higher speed [19].

3.4 Apache Spark

Spark is a fast data processing engine for large-scale data which can run programs up to 100x faster than that of traditional MapReduce in Memory [3].

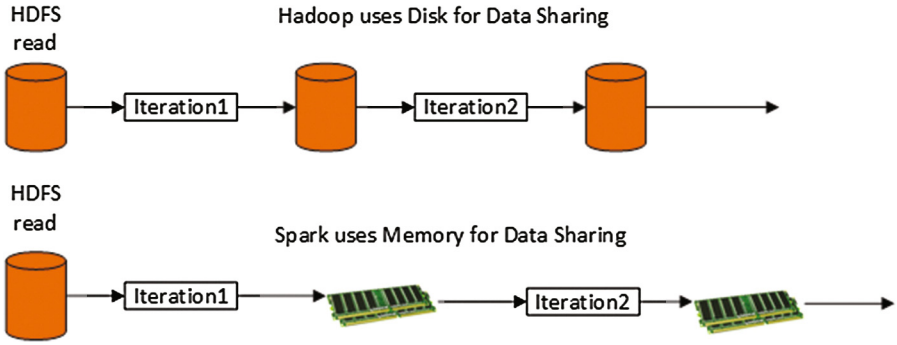


Fig. 3. Comparison of Spark and Hadoop

Since Apache Spark doesn't provide distributed storage like HDFS, we integrated it with HDFS into our system design. It can run on the top of HDFS and process the financial data. Spark also supports real-time data processing and fast queries. One major drawback of the Spark processing is, it needs more RAM as almost all of the data processing job is done in memory (Fig. 3).

3.5 Financial Data Sets

For our work, we used a popular database, the New York Stock Exchange (NYSE)'s Daily Trade and Quote (DTAQ), an academic market micro-structure

research in U.S. equities [4]. DTAQ consists of a set of files containing all trades and quotes listed and traded in US regulated exchanges in a single day. The data sets are generally in a binary format derived from the output of CTA and UTP SIPs, tapes A, B, and C.

- **National Best Bid and Offer (NBBO) Appendix File:** NBBO file contains records when two different exchanges hold the best bid and best offer price in the securities information processors (SIPs).
- **Quote File:** It consists of the quoted price for all U.S equities and flags when both of the exchanges represent the NBBO.
- **Trade File:** It includes the exact trade happens in U.S. equities.

All quotes and trades files are time-stamped to the microseconds; however recent high-frequency trading is time-stamped in nanoseconds as well. According to Holden and Jacoben's work [15] on liquidity measurement problems in fast, competitive markets and expensive and cheap solutions, the NBBO file is incomplete and in order to get the complete NBBO it needs to be merged records flagged as NBBO of the quote file (Fig. 4).

Time	Ticker	BestBidPrice	bestbidsize	bestaskprice	BestAskSize	BestBidExchange	BestAskExchange
32400.168105	SPY	190.31	4	190.32	7	Z	P
32400.566107	SPY	190.31	8	190.32	3	Z	P
32400.566185	SPY	190.31	8	190.32	1	Z	P
32400.566328	SPY	190.32	1	190.32	1	T	P
32400.56634	SPY	190.32	2	190.32	1	T	P
32400.566643	SPY	190.32	2	190.32	3	T	P
32400.56714	SPY	190.32	2	190.32	1	T	P
32400.604909	SPY	190.32	1	190.32	1	T	P
32400.628667	SPY	190.31	4	190.32	1	Z	P
32400.629165	SPY	190.31	2	190.32	1	P	P

Fig. 4. NBBO in DataFrame processed by the Spark Clusters

For multimarket data analysis using association rules, we used the TAQ Trade, Quote and NBBO files of a particular month (August 2015). We first calculated the trading frequency and then used the Association rules based on FP-Growth to find the interesting pattern in the exchanges. The pattern is also analyzed in the perspective of price maker and price taker in the financial market.

3.6 Association Rules Mining

In this section, we describe the unsupervised data mining technique called Association rule mining. Association rules are a very popular rule in machine learning to find out interesting relations among variables in a large data sets. Association rules are used to find the pattern (sub-sequence or substructure of a set of frequent items that occur together). The pattern represents intrinsic and important properties of large data sets and is very useful in business for making a

decision. Formally, an association rule is an implication of the form $X \rightarrow Y$ [5]. Where X = $\{x_1, x_2, ..., x_n\}$ and Y = $\{y_1, y_2, ..., y_n\}$ are the set of distinct items in a transaction T. The X is commonly known as the antecedent and Y as the consequent. The association rules simply says that whenever there is a possibility of happening event X there is also likely to happen event Y as well. One of the earliest applications of the association rules is the market basket analysis done on a large set of costumer transactions. In order to formulate the rules for association the notion of support and confidence are used. Support and confidence are used to measure the strength of the rule. The quality measure of the association rules are represented by three terms: **Support, confidence and lift.**

Support is a fraction of a transaction that that contains both item sets X and Y. It determines how frequent a rule is applicable to a given data set. Formally support for association rule $X \rightarrow Y$ is $support(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{N}$ Where N is the number of instances and $\sigma(X \cup Y)$ is the number of instances covering both X and Y [21]. The idea is to take the number of items that cover both X and Y and divide that by the total number of instances in the database under consideration.

Confidence measures how often items in Y appear in a transaction that contains X. Formally confidence can be defined as $confidence(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)}$ [21].

The third quality metric used in the association rule is lift. The association rule $X \rightarrow Y$ is interesting with high support and confidence but sometimes it tends to be misleading and give false positives. In such cases, the correlation between X and Y is considered and is called lift. Three types of correlations are considered from the lift value.

- Positively correlated when $\text{lift}(X \rightarrow Y) > 1$
- Negatively correlated when $\text{lift}(X \rightarrow Y) < 1$
- X and Y are independent if lift is nearly equal to 1.

Support, confidence, and lift can be expressed in terms of probability.

$$support(X \rightarrow Y) = P(X \cup Y)$$

$$confidence(X \rightarrow Y) = P(Y|X)$$

$$lift(X \rightarrow Y) = \frac{P(X \cup Y)}{P(X)P(Y)}$$

Support, s is the probability that a transaction contains $\sigma(X \cup Y)$ and Confidence, c is the conditional probability that a transaction containing X also contains Y [13]. We count the number of instances which covers both X and Y, and divide that by the number of instances covering just X. Both support and confidence have their own significance. Confidence indicates the strength of the rules however support has a statistical significance. The higher the confidence, the more strength the rule. Another motivation for support is to suppress the rules that are the minimum threshold for business reasons. In order to formulate

the meaningful rules and pattern, the optimum value of minimum support (minsup) and minimum confidence (minconf) are chosen. Both minsup and minconf varies within the range of 0 to 1 [6].

In association rule mining we will consider all the rules, X → *Y* with minimum assumed support and having maximum confidence.

3.7 Scalable Mining Methods

There are three major approaches to mining Level-wise, Join-based approach: Apriori [7], Vertical data format approach: Eclat [24] and Frequent pattern projection and growth: FP-Growth [14]. In this research, we adopted the FP-Growth technique because it is efficient, has less memory usage, takes shorter execution time, and scalable and can be used in the distributed system [17].

The example of Association rule in the context of the financial stock market can be:

- When exchange X is dominant, at 70% of probability exchange, Y is also dominant on the same day.
- If price change occurs in exchange X, then exchange Y simply copies that exchange by 85% probability.
- If the price of X goes up, then, 50% of the time, the price of Y goes down.
- If exchange X is dominant, Y immediately follows it most of the time.

It is clear that investors and traders are more interested in the above rules.

The main purpose of the study is to apply the association rule to find out the conditional dominance in the US stock market. This paper utilizes the concept of inter-transaction rule mining [16]. In order to find the association rules and apply the FP-Growth algorithm we have to convert the data into the transactional format. The example of a transactional format is as shown in Table 3. Here, individual stock exchanges are listed with their corresponding name. The term rise, fall, low and high are used to represent their status in dominance analysis.

4 Implementation of Big Data Framework

In this section, we describe how we used big data framework such as Apache Hadoop and Spark to process and analyze the high-frequency stock trading data. The TAQ data is generally in zipped format. They are unzipped and are kept in the HDFS in the clusters.

Our system consists of the clusters of 4 Nodes; one acts as a master or namenode and other 3 acts as a slave or datanode. The cluster's configuration consists of the following environment:

Hardware: Intel(R) Xeon(R) CPU E5320 @ 1.86 GHz Operating System: Ubuntu 16.04, kernel: 4.4.0-87-generic Hadoop Version: Hadoop-2.7.3, Spark Version: Spark 2.0.0, Memory: 16 GB per node, Hard Drive: 2 TB in Master Node and 1 TB in each data node.

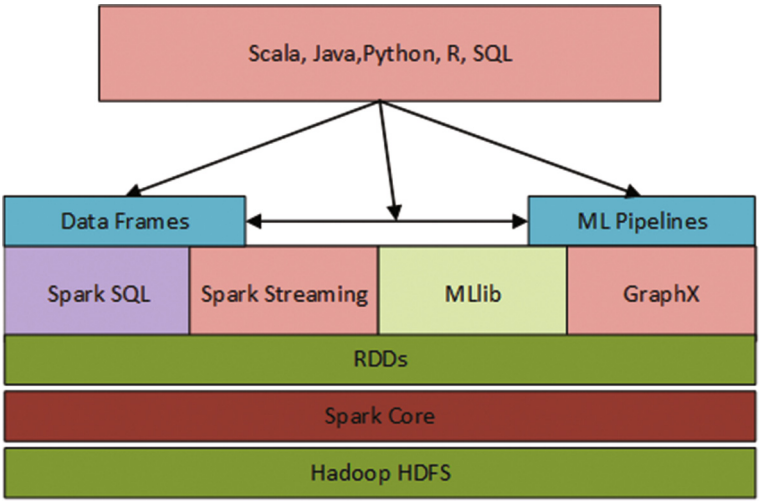


Fig. 5. System architecture showing major big data components

Figure 5 shows the overall system architecture. The lowest block represents the Hadoop File System and the next block represents the Spark Core. The third is Spark resilient distributed data sets (RDD), which is essentially a read-only collection of objects distributed across the set of machines or clusters [23]. The upper block represents the higher level API used to access the spark cores and RDDs. The Spark core takes these data and coverts them to RDD and then ultimately to data frames by applying filtering and certain rule sets. Once a data frame, it is processed by a high-level visualization engine such as R and Python. In addition to visualization, Python also generates transactional rules set for applying the association rules algorithm called FP-Growth.

5 Experimental Results

In this section we describe several experiments that have been done to process the financial data using Big data framework such as Spark and FP-Growth algorithm.

By using association rules on Trade file, Quote and NBBO interesting relations can be formulated. Figure 6 shows different ways to find the inter relations between U.S. exchanges.

5.1 Dominance Pattern Using Trade File

In order to find the dominance pattern, we used the trade file provided by the NYSE. Each trade frequency that happened in the exchange is calculated during time t_i and compared with time t_{i-1}.

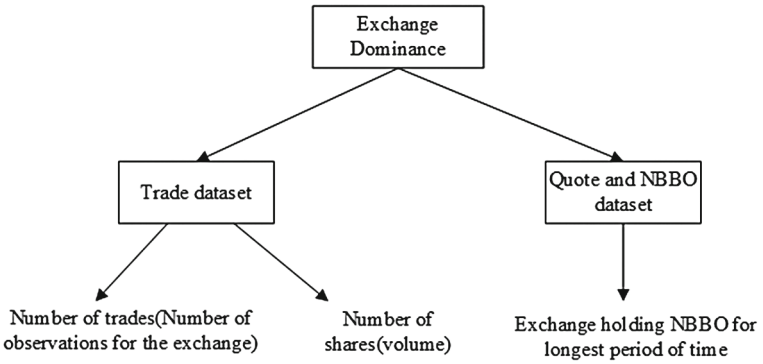


Fig. 6. Finding exchange dominance

Table 3. Transactional data format

fB	rD	rJ	rK	fM	fP	rT	fX	fY
rB	fD	rJ	fK	fM	rP	fT	rX	fZ
fB	rD	fJ	fK	fP	fT	rX	rY	fZ
rB	fD	rJ	fK	rP	fT	fX	rY	rZ

The transactional data set is created using the simple mapping techniques. If the trade frequency in exchange X is rising in time t_i, then it is indicated with rX; similarly, if trade frequency in exchange Y is falling in time t_i, then it is marked with ry. Additionally, if exchange X is the dominant trade with a heavy traded frequency in particular time t_i then it is marked with mX.

Once the transactional data set is ready, we apply it the Association rules based on FP-Growth in Apache Spark clusters.

Table 4 lists the sample association rules produced by applying the FP-Growth algorithm in the trade file of the corresponding day of August 2015. The rules are a conditional probability in the form $X \Rightarrow Y$ meaning, if X is true, Y is also true with a probability of c, the confidence.

Here in this rule, as shown in the Table 3, r implies rising, and f implies falling. The next letter in upper case denotes the name of the Exchange. Here, the rules with higher confidence greater than 0.85 are more significant and carry a stronger meaning in terms of financial interpretations. For example, the rule $[rT, rP] \Rightarrow [rZ]$ with confidence of 0.88 says that whenever NASDAQ Stock Exchange, LLC (**T**) and NYSE Arca, Inc. (**P**) are rising there is a possibility of a rise in Bats BZX Exchange, Inc. (**Z**).

5.2 Dominance Pattern

In this experiment, we took one month of TAQ trade data from August 2015 and conducted the experiment to find out the the dominance pattern. The result is

Table 4. Sample Association Rules with minimum support 0.3

August 21	Confidence	August 24	Confidence	August 25	Confidence
$[fT, fZ] \Rightarrow [fP]$	0.84	$[rT, rP] \Rightarrow [rZ]$	0.88	$[fT, fZ] \Rightarrow [fP]$	0.84
$[fT, fZ] \Rightarrow [fD]$	0.81	$[fZ] \Rightarrow [fT]$	0.81	$[fY, fP] \Rightarrow [fZ]$	0.90
$[fY, fP] \Rightarrow [fZ]$	0.90	$[rP, rZ] => [rT]$	0.85	$[rK, rT] \Rightarrow [rZ]$	0.86
$[rK, rT] \Rightarrow [rZ]$	0.86	$[fZ, fT] \Rightarrow [fP]$	0.81	$[fP, fD] \Rightarrow [fZ]$	0.83
$[fP, fD] \Rightarrow [fZ]$	0.83	$[fZ, fP] \Rightarrow [fT]$	0.86	$[rP, rZ] \Rightarrow [rT]$	0.84
$[rP, rZ] \Rightarrow [rT]$	0.84	$[rT] \Rightarrow [rZ]$	0.81	$[fK, fP] \Rightarrow [fZ]$	0.86
$[fT, fP] \Rightarrow [fZ]$	0.87	$[fP, fT] \Rightarrow [fZ]$	0.86	$[rZ, rK] \Rightarrow [rT]$	0.83
$[fZ, fP] \Rightarrow [fT]$	0.81	$[fB, fT] \Rightarrow [fZ]$	0.86	$[fZ, fD] \Rightarrow [fP]$	0.83
$[rP, rT] \Rightarrow [rZ]$	0.84	$[fZ, fB] \Rightarrow [fT]$	0.88	$[rZ, rT] \Rightarrow [rP]$	0.80
$[fY, fZ] \Rightarrow [fP]$	0.87	$[rT, rZ] \Rightarrow [rP]$	0.81	$[fZ, fP] \Rightarrow [fT]$	0.81

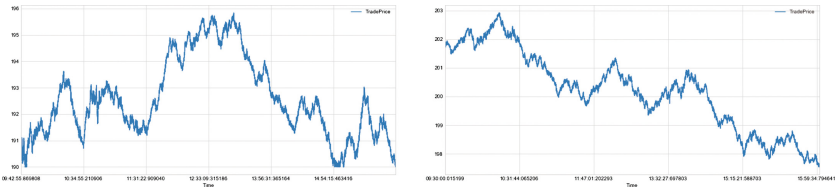
Table 5. Dominance Pattern on Trade, Aug 2015

Day	Dominance Patterns	Day	Dominance Patters
03	D P T Z K Y J B X M	18	D P Z T K Y J B X M
04	P Z D T K Y B J X M	19	D P Z T K Y J B X M
05	D P T Z K J B Y X M	20	D P T Z K J Y B X M
06	D P T Z K Y J B X M	21	D Z T P K J B Y X M
07	D P T Z K Y J B X M	24	D P Z T K J Y B X M
10	D P T Z K Y J B X M	25	D Z T P K J B Y X M
11	D P Z T Y K J B X M	26	P D T Z K J Y B X M
12	D T N P Z Y K J B X A M	27	T Z P D K J B Y X M
13	D P T Z K B J Y X M	28	D P T Z K Y J B X M
14	D P Z T K J Y B X M	31	D P Z T K J B Y X M
17	D P T Z Y K J B X M		

presented in Table 5. The results show an interesting pattern where Dark pools trading Finra (**D**) is immediately followed by the NYSE Arca (**P**).

5.3 Price Analysis

The trade file reflects the exact trade that happens during the day. Here we present the price analysis of the SPY ticker during the flash crash day (August 24, 2015) and normal day (just a day before the flash crash day August 21, 2015). Figure 7 shows a comparison of the price between normal day (Aug 21) and flash crash day (Aug 24) of 2015 for SPY ticker. It was found that there was a sharp fall in SPY ticker price of about 10% in flash crash day than that of a normal day.



(a) August 24,2015, Flash Crash day (b) August 21, 2015, Normal day

Fig. 7. Price Analysis of SPY Ticker

It can be clearly seen that the SPY declined to more than 5% below its closing price on the previous day (Friday, August 21, 2015).

6 Performance Test of the Big Data Platform

To test the performance of the spark clusters, we used the TAQ quote data of Aug 24, 2015 with uncompressed size of 203 GB. In the test, we measured the execution time of the clusters for given data vs. the number of nodes. The Table 6 shows the total time taken by the clusters to process the quote data of 203 GB with a decreasing number of clusters. It is clear that more data nodes have a shorter execution time for processing large volume of data.

Table 6. Execution time in Apache Spark Cluster

Number of data nodes	Execution time
3	17 min 44 s
2	21 min 17 s
1	33 min 56 s

7 Conclusion

In this work, we developed a Big Data framework based on Apache Spark to process the financial Daily Quote and Trade file to find out the hidden relations among the stock exchanges using data technique called FP-Growth algorithm. In addition, we also performed the dominance analysis based on the NYSE's DTAQ NBBO and quote data sets. The results appear promising and can be used by investors and other companies to find in which exchange they can trade. Moreover, the framework can be reused to find the future stock prices of certain companies and predict flash crash.

References

1. Exchange pricing model (2011). http://www.nomura.com/europe/resources/pdf/ExchangePricingModels_20110614.pdf
2. 5vs big-data (2015). http://www.ibmbigdatahub.com/blog/why-only-one-5-vs-big-data-really-matters
3. Apache spark (2018). https://spark.apache.org/
4. Nyse daily taq (trade and quote) (2018). http://www.nyxdata.com/Data-Products/Daily-TAQ
5. Agarwal, R.C., Aggarwal, C.C., Prasad, V.: A tree projection algorithm for generation of frequent item sets. J. Parall. Distrib. Comput. **61**(3), 350–371 (2001)
6. Agrawal, R., Imieliński, T., Swami, A.: Mining association rules between sets of items in large databases. In: ACM SIGMOD Record, vol. 22, pp. 207–216. ACM (1993)
7. Agrawal, R., Srikant, R., et al.: Fast algorithms for mining association rules. In: Proceedings 20th International Conference on Very Large Data Bases, VLDB, vol. 1215, pp. 487–499 (1994)
8. Aldridge, I.: High-Frequency Trading: A Practical Guide to Algorithmic Strategies and Trading Systems, vol. 459. Wiley, Hoboken (2009)
9. Asadifar, S., Kahani, M.: Semantic association rule mining: a new approach for stock market prediction. In: 2017 2nd Conference on Swarm Intelligence and Evolutionary Computation (CSIEC), pp. 106–111. IEEE (2017)
10. Chowdhry, B., Nanda, V.: Multimarket trading and market liquidity. Rev. Financ. Stud. **4**(3), 483–511 (1991)
11. Fang, B., Zhang, P.: Big data in finance. In: Yu, S., Guo, S. (eds.) Big Data Concepts, Theories, and Applications, pp. 391–412. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-27763-9_11
12. Han, J., Cheng, H., Xin, D., Yan, X.: Frequent pattern mining: current status and future directions. Data Mining Knowl. Discov. **15**(1), 55–86 (2007)
13. Han, J., Kamber, M., Pei, J.: Data Mining: Concepts and Techniques. The Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann, Burlington (2000)
14. Han, J., Pei, J., Yin, Y.: Mining frequent patterns without candidate generation. In: ACM SIGMOD Record, vol. 29, pp. 1–12. ACM (2000)
15. Holden, C.W., Jacobsen, S.: Liquidity measurement problems in fast, competitive markets: expensive and cheap solutions. J. Financ. **69**(4), 1747–1785 (2014)
16. Luhr, S., Venkatesh, S., West, G.: Emergent intertransaction association rules for abnormality detection in intelligent environments. In: Proceedings of the 2005 International Conference on Intelligent Sensors, Sensor Networks and Information Processing Conference, pp. 343–347. IEEE (2005)
17. Mythili, M., Shanavas, A.M.: Performance evaluation of apriori and fp-growth algorithms. Int. J. Comput. Appl. **79**(10), 279–293 (2013)
18. Preis, T., Moat, H.S., Stanley, H.E.: Quantifying trading behavior in financial markets using google trends. Sci. Rep. **3**, 1684 (2013). https://doi.org/10.1038/srep01684
19. Shvachko, K., Kuang, H., Radia, S., Chansler, R.: The hadoop distributed file system. In: Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST), MSST 2010, pp. 1–10. IEEE Computer Society, Washington, DC (2010). https://doi.org/10.1109/MSST.2010.5496972

20. Sohangir, S., Wang, D., Pomeranets, A., Khoshgoftaar, T.M.: Big data: deep learning for financial sentiment analysis. J. Big Data **5**(1), 3 (2018)
21. Tan, P.N., et al.: Introduction to Data Mining. Pearson Education, India (2006)
22. Wu, X., Zhu, X., Wu, G.Q., Ding, W.: Data mining with big data. IEEE Trans. Knowl. Data Eng. **26**(1), 97–107 (2014)
23. Zaharia, M., Chowdhury, M., Franklin, M.J., Shenker, S., Stoica, I.: Spark: cluster computing with working sets. HotCloud **10**(10–10), 95 (2010)
24. Zaki, M.J., Parthasarathy, S., Ogihara, M., Li, W., et al.: New algorithms for fast discovery of association rules. In: KDD, vol. 97, pp. 283–286 (1997)