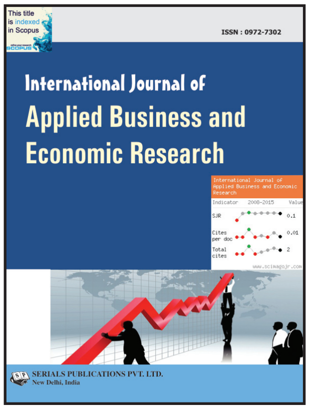

International Journal of Applied Business and Economic Research

ISSN : 0972-7302

available at http: www.serialsjournal.com

© Serials Publications Pvt. Ltd.

Volume 15 • Number 14 • 2017

A Real-Time Data Mining with Genetic Algorithm

Eun-Jin Yang[1], Hyong-Jung Kim[2] and Jin-Hwa Kim[3]

[1] *Dong Bu Information Technology, Korea*

[2] *Ph.D, Graduate School of Business, Sogang University*

[3] *Corresponding Aothor, Professor, School of Business, Sogang University*

ABSTRACT

The purpose of this paper is to present a new mining algorithm for mining a real-time data. This study uses both the genetic algorithm and the rule induction using decision tree to predict stock market index. Input variables used in this study are slow% D, ROC, William% R, CCI. Output variable was also used as KOSPI index. Research procedures are conducted in the following orders: ① data entry after data division, ② agents (rule extraction), ③ performance tests, ④ agent mutation, ⑤ prediction rate tests. As a result of the analysis, prediction rate in each generation improves the decision making ability of the agents by acquiring traits to adapt to the environment. This shows that the ability for prediction performance increases by obtaining a dominant genes. Evolution leads to the better prediction accuracy. This presents that decision-making capabilities of the agent is improved by acquiring a genetic trait to adapt to the environment. In other words, agents continually make a genetic mutation in order to adapt to environmental changes, and the new child agents having better predictive genes survive. In this paper, we compared the proposed algorithm with other methodologies to verify the effectiveness of the model. We performed experiments that compared the proposed algorithm with rule induction methods, neural network, discriminant analysis and logistic regression analysis. All experiments were performed in the same situation. The proposed method showed the best predictive performance. As the proposed algorithm is more effective than other conventional mining techniques, this study will have a good chance that can effectively apply a real-time data in real business environment.

Keywords: Genetic algorithm, Rule induction, Stream data mining, Real-time data, Stock market.

1. INTRODUCTION

Recently it is easy to find large size stream data because of the development of information technologies. In addition, companies want to effectively analyze large stream data in real time. Therefore, there is a necessity to analyze massive stream data. Researches in data mining (Chan & Stolfo, 1995) show studies on methodologies about the size in mass data mining algorithms. As the data size increases, the run time is increased. More researches (Fayyad & Uthurusamy, 2002) also suggest the necessity of mining algorithm to extract the useful information from massive data. It is more useful if multiple mining models are tested and their performances are compared (Achelis, 1995). This study analyzes a stream data with a suggested method and compares its performance with traditional mining models.

In order to classify or predict with the stream data, it can be effectively predicted by preprocessing the stream data in data sets (Ganti, Gehrke & Ramakrishnan, 2002; Cheng & Titterington, 1994). The purpose of this paper is to present a new mining algorithm for mining a mass stream data.

2. LITERATURE REVIEW

Rule Induction

Rule induction methods are analytical methods that can classify and predict with the decision. Because a rule induction algorithm is to visualize the data processing into tree structure, it can easily be understood and can easily be implemented (Apte & Weiss, 1997). In addition, a rule induction method is a method which is used in many research fields. A decision tree is an analytical method composed of nodes and branches. Full data is composed of tree structure, and root node is the starting node.

C4.5, which is used in this study, is one of the popular algorithm to induce a decision tree (Quinlan, 1993). C4.5 is the enlargement of Quinlan's ID3 algorithm. The ID3 algorithm that is basic algorithm of C4.5 creates the classification rules of tree form.

Stream Data Mining

Stream data continuously grows over time, and it has the unlimited data flow. In order to classify or predict with the stream data, it is possible to effectively manage rules from data than raw data itself (Ganti, Gehrke & Ramakrishnan, 2002; Cheng & Titterington, 1994).

The data in modern times is used to model complex phenomena, but it is pointed out that the overly simplistic models are easily produced (Domingos & Hulten, 2000). It is because all the information from the data can't be accessed (Domingos & Hulten, 2000). Therefore, a very large storage space is required in order to mine the stream data, and considerable time is spent in order to analyze the stream data (Guha, Mishra, Motwani & O'Callagan, 2000). It is also difficult to store stream data reliably. Methods to solve this problem are discussed (Greenwald & Khanna, 2001; Hulten, Spencer & Domingos, 2001; Murugananthan & ShivaKumar, 2016) in this study. There is an advanced research (Giannella, Han, Pei, Yan, & Yu, 2003) that proposed a FP-Stream by improving it with the FP-Tree (Han, Pei & Yin, 2000) to mine the frequent pattern of streams data.

Genetic Algorithm

Genetic algorithms are search algorithms based on the mechanism of natural selection (Holland, 1975). The researches in genetic algorithms implement an artificial involving system, which is an important mechanism of natural evolution systems (Goldberg, 1989).

A genetic algorithm is started in a population of candidates that are randomly distributed. The concept of genetic algorithm is simple. The more generations of a population are advanced, the better the performance of it.

The next generation is formed by reproducing and recombining superior candidates. Reproduction means that the better genes are used in reproduction of new generations in accordance with fitness function values. Recombination (crossover) is to randomly select two parent genes among the potential candidate genes. Recombination also combines the parent genes random way.

There are a genetic programming (GP) and an evolution programming (EP) developed to enhance the genetic algorithm (Koza, 1994). The genetic programming (Koza, 1994) is to extend genetic algorithms. The evolution programming was developed as a way to evolve the finite state machine (FSM). The evolutionary programming algorithm has basically processes of selection and mutation (Koza, 1994; Purusothaman & Krishnakumari, 2015).

3. RESEARCH DESIGN AND EXPERIMENTS

Data Collection and Variables Description

This study uses 1541 stock market data from January 2, 1998 to November 26, 2003. This study uses the most useful variables when analyzing stock data. Input variables used here are Slow % D, ROC, William % R, and CCI. Output variable is KOSPI index value.

Table 39.1
Variables description in the stock market data

Variables	*Description*
Slow (%K)	In order to analyze that the current stock price is include in which stock price during the period of the year (Chan & Stolfo, 1995).
Slow (%D)	In order to analyze that the current stock price is include in which stock price during the period of the year.
ROC	ROC shows the change rate of the price as a percentage at some point.
William (%R)	Momentum indicator to measure the oversold or the overbought.
CCI	Indicator for measuring the movement of the stock price from the moving average.
KOSPI	Change (1) or not (0) in the KOSPI index.

Method

This paper suggests a new algorithm using both the genetic algorithm and the rule induction method. Research procedures in this study are shown in Figure 39.1.

Research procedures are in the following order: ① Data entry after data division, ② Prepare agents (rule extraction), ③ Performance tests, ④ Agents mutation, ⑤ Prediction rate tests. We divide the data set into seven generations, and a generation is divided into four agents.

After cutting the stream data into a fixed size, we extract rules from each of them. Rules extracted from each data set are saved into a single agent. Because the rules with the better predictive performance have good genetic traits, we make the dominant rule set by copying a dominant rule among agents.

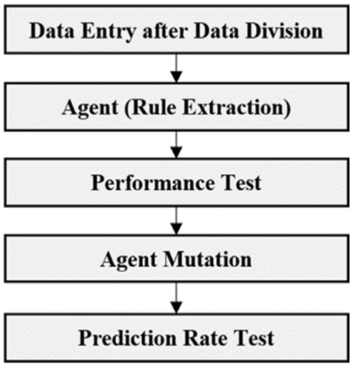

Figure 39.1: Procedure for research method

We extract the rule by using See5 in each generation. Then, each agent will have the different rule set. Detailed study is designed as follows. Firstly, we divide the entire data set into seven generations, and we perform that a generation has four agents. We also assign 50 records to one agent. Secondly, we extract the rules by using See5 in each generation. The rules extracted in each agent become the element of gene traits. Thirdly, we perform performance tests. Performance tests also are performed by comparing the performance of agents with the data in next generation. Fourthly, after performance tests, dominant agents in each generation are selected. Only dominant agents can produce next generations. These processes are repeated until the last generation where there is no improvement in the genetic evolutions.

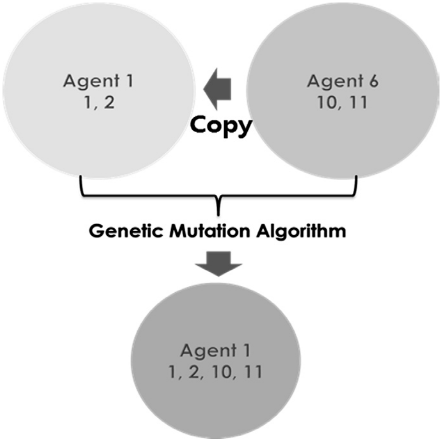

Figure 39.2: The agent after second-generation mutations

All agents of the first generation basically move into the next generation. Figure 39.2 is a schematic view showing the copy process for the trait in the next generation through mutations. In the second mutation, both a genetic trait of first generation and the dominance genetic trait of second generation agent are combined. The combined method is to combine a dominant gene as union (AUB) method, and common genes are combined only once.

Table 39.2 presents the copy process for a genetic trait of first and second generation. In Table 39.2, all of the agents from agent 1 to agent 4 present a genetic trait of parent agent in first generation. In Table 39.2, both agent 6 and agent 7 show the dominance genetic trait selected in second generation. The right column presents the agents combined from mutations after second-generation.

Table 39.2
The process of second-generation mutations

First generation	*Second generation*	*Agents after second-generation mutation*
Agent 1 = {1, 2}	Agent 6 ={10, 11}	Agent 1 – Agent 6 = {1, 2, 10, 11}
Agent 2 = {3, 4}	Agent 7 ={12, 13}	Agent 1 – Agent 7 = {1, 2, 12, 13}
Agent 3 = {5, 6, 7}	…	Agent 2 – Agent 6 = {3, 4, 10, 11}
Agent 4 = {8, 9}	…	Agent 2 – Agent 7 = {3, 4, 12, 13}
		Agent 3 – Agent 6 = {5, 6, 7, 10, 11}
		Agent 3 – Agent 7 = {5, 6, 7, 12, 13}
		Agent 4 – Agent 6 = {8, 9, 10, 11}
		Agent 4 – Agent 7 = {8, 9, 12, 13}

4. EXPERIMENTAL RESULTS

Test result

Table 39.3 presents the average prediction results after mutation at each generation. Seven generations were conducted through the mutations of seven times.

Table 39.3
The average prediction results after mutation

	1G	*2G*	*3G*	*4G*	*5G*	*6G*	*7G*
Agent 1	0.66	0.66	0.65	0.66	0.69	0.69	0.701
Agent 2	0.63	0.69	0.66	0.66	0.69	0.69	0.696
Agent 3	0.55	0.685	0.69	0.69	0.685	0.685	0.691
Agent 4	0.69	0.635	0.685	0.685	0.685	0.685	0.69
Average	0.632	0.667	0.6712	0.6737	0.687	0.687	0.694

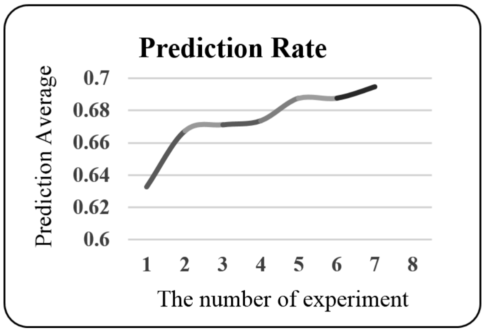

Figure 39.3: Prediction rate graph after mutation per generation

As presented in Figure 39.3, prediction rate is seen the largest growth between the first and second generation. Each generation improves the decision making performance of the agents by acquiring traits to adapt to the environment. In Figure 39.3, increased rate is getting slower because the agents already have most useful rules by adapting to environment. This shows that the ability for genetic variation increases by obtaining a dominant gene. As the generation evolves, the performance of the agents gets better. It, however, eventually saturated to a limit

Comparison with Other Methodologies

In this paper, we compared the proposed algorithm with other methodologies to verify the effectiveness of the suggested algorithm. We compared the performance of the proposed algorithm with that of rule induction methods, neural network, discriminant analysis and logistic regression analysis. All experiments were performed in the same condition.

Table 39.4
The comparison of prediction rate

Comparison of prediction rate					CMR = proposed algorithm in this study, RI = rule induction, NN = neural network, DA = discriminant analysis, LRA = logistic regression analysis
CMR	*RI*	*NN*	*DA*	*LRA*	
69.5	59.3	57.8	67.9	65.5	

Table 39.4 presents the prediction performance of the suggested method compared to other methodologies. In Table 39.4, CMR shows a suggested algorithm proposed in this study. The algorithm proposed in this study shows higher performance than that of other methodologies.

5. CONCLUSION

This study suggests a new algorithm by using both the genetic algorithm and the rule induction algorithm. As agents with better prediction performance are dominant agent with better gene traits, these agents have right to mutate each other, producing next child agents. As a result, the agent remaining at the end has a highest predictive performance and the best rules in them.

Experiments are tested to forecast the KOSPI index by using stock market data. Experimental results show that the suggested method can predict stock market best compared to the traditional methods.

References

Achelis, S.B. (1995), *Technical Analysis from A to Z: Covers Every Trading Tool.* Probus Publishing Co.

Apte, C., & Weiss, S. (1997), Data mining with decision trees and decision rules. *Future generation computer systems, 13(2-3)*, 197-210.

Chan, P.K., & Stolfo, S.J. (1995, August), Learning Arbiter and Combiner Trees from Partitioned Data for Scaling Machine Learning. *In KDD*, 95, 39-44.

Cheng, B., & Titterington, D.M. (1994), Neural networks: A review from a statistical perspective. *Statistical science*, 2-30.

Domingos, P., & Hulten, G. (2000, August), Mining high-speed data streams. *In Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, 71-80.

Fayyad, U., & Uthurusamy, R. (2002), Evolving data into mining solutions for insights. *Communications of the ACM, 45(8)*, 28-31.

Ganti, V., Gehrke, J., & Ramakrishnan, R. (2002), Mining data streams under block evolution. *ACM SIGKDD Explorations Newsletter, 3(2)*, 1-10.

Giannella, C., Han, J., Pei, J., Yan, X., & Yu, P.S. (2003), Mining frequent patterns in data streams at multiple time granularities. *Next generation data mining, 212*, 191-212.

Goldberg D. (1989), *Genetic Algorithms in Search.* Optimization and Machine Learning, Reading, Massachusetts.

Greenwald, M., & Khanna, S. (2001, May), Space-efficient online computation of quantile summaries. *In ACM SIGMOD Record, 30(2)*, 58-66.

Guha S., Mishra N., Motwani R., O'Callagan L. (2000), Clustering Data Streams. *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, 359-366.

Han, J., Pei, J., & Yin, Y. (2000, May), Mining frequent patterns without candidate generation. *In ACM Sigmod Record, 29(2)*, pp. 1-12.

Holland, J.H. (1975), *Adaptation in natural and artificial systems.* University of Michigan Press, Ann Arbor.

Hulten, G., Spencer, L., & Domingos, P. (2001, August), Mining time-changing data streams. *In Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, 97-106.

Koza, J.R. (1994), *Genetic programming II: Automatic discovery of reusable subprograms.* Cambridge, MA, USA.

Murugananthan, V., & ShivaKumar, B.L. (2016), An adaptive educational data mining technique for mining educational data models in elearning systems. *Indian Journal of Science and Technology, 9(3).*

Purusothaman, G., & Krishnakumari, P. (2015), A survey of data mining techniques on risk prediction: Heart disease. *Indian Journal of Science and Technology, 8(12).*

Quinlan, J.R. (1993), *C4.5: Programs for Machine Learning.* Morgan Kaufmann, San Mateo, CA.