韓國 컴퓨터情報學會 論文誌
第12卷 第6號, 2007. 12.

2007-12-6-3-3

자료편집기법과 사례기반추론을 이용한 한국종합주가지수 예측

김 경 재*

Prediction of KOSPI using Data Editing Techniques and Case-based Reasoning

Kyoung-jae Kim*

요 약

본 연구에서는 한국종합주가지수 (KOSPI)의 예측을 위하여 사례기반추론에서의 유전자 알고리즘을 이용한 새로운 자료편집기법을 제안한다. 사례기반추론은 복잡한 문제 해결에서의 편의성과 강점으로 인하여 여러 분야에서 광범위하게 활용되고 있다. 그럼에도 불구하고 사례기반추론은 다른 기계학습기법에 비하여 낮은 예측정확도를 나타내기에 비판을 받아 왔다. 일반적으로 사례기반추론으로부터 성공적인 성과를 도출하기 위해서는 주어진 문제에 유용한 선행 사례를 효과적으로 추출하는 것이 핵심이다. 그러나 사례기반추론 시스템에서 우수한 대응과 추출방법을 설계하는 것은 여전히 논란이 있는 연구 주제이다. 본 연구에서는 사례기반추론 시스템에서 우수한 대응과 추출을 위하여 유전자 알고리즘이 동시에 속성 가중치와 적합한 사례를 선택하는 것을 최적화한다. 본 연구에서는 제안된 모형을 주식시장분석에 응용한다. 실험결과는 유전자 알고리즘 접근법이 사례기반추론에서 유망한 사례편집기법이라는 것을 보여준다.

Abstract

This paper proposes a novel data editing techniques with genetic algorithm (GA) in case-based reasoning (CBR) for the prediction of Korea Stock Price Index (KOSPI). CBR has been widely used in various areas because of its convenience and strength in complex problem solving. Nonetheless, compared to other machine learning techniques, CBR has been criticized because of its low prediction accuracy. Generally, in order to obtain successful results from CBR, effective retrieval of useful prior cases for the given problem is essential. However, designing a good matching and retrieval mechanism for CBR systems is still a controversial research issue. In this paper, the GA optimizes simultaneously feature weights and a selection task for relevant instances for achieving good matching and retrieval in a CBR system. This study applies the proposed model to stock market analysis. Experimental results show that the GA approach is a promising method for data editing in CBR.

▸ Keyword : Data editing, Instance selection, Genetic algorithms, Case-based reasoning, Stock market prediction

• 제1저자 : 김경재
• 접수일 : 2007.11.23, 심사일 : 2007. 11.27, 심사완료일 : 2007. 12.2.
* Assistant Professor, Dept. of Management Information Systems, Dongguk University
※ This work was supported by the Dongguk University Research Fund of 2005(DRIMS 2005-2007-0)

Ⅰ. 서 론

Case-based reasoning (CBR) is a popular inference technique and has been applied to many business problems. The basic idea of CBR is to find a solution to new problems by adopting solutions that have been used in the past. Although most artificial intelligence techniques pursue generalized relationships between problem descriptors and conclusions, it just refers to specific knowledge of previously experienced, concrete problem situations, so it is effective for complex and unstructured problems and easy to update.[12] CBR is considered to be a five-step process shown in 〈Figure 1-1〉.[1]

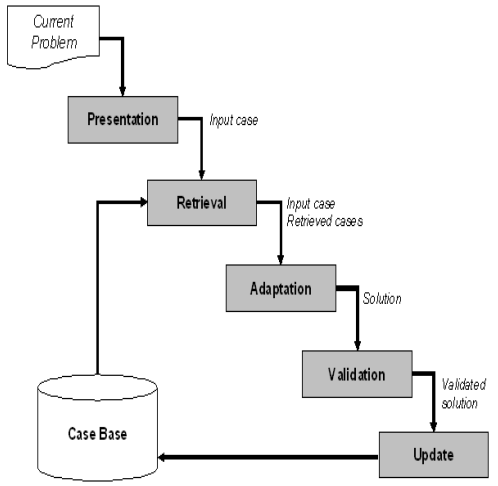

Figure 1-1. The general CBR process
그림 1-1. CBR의 일반적 프로세스

Among the steps of the process, the second process, case retrieval, is the most important step because the performance of CBR systems usually depends on it.[7] In this step, the CBR system retrieves the most similar cases from the case memory, which become the bases for solution of the input problem. Thus, it is crucial to determine appropriate similar cases. In particular, feature weighting or selection and instance selection for measuring similarity have been controversial issues in designing CBR systems. There have been many studies to determine these factors. Among many methods of instance selection and feature weighting, GAs are increasingly being used in CBR systems.

This paper proposes a new hybrid model of CBR and genetic algorithms (GAs) for feature weighting and data editing in the context of stock market prediction. An evolutionary data editing technique reduces the dimensionality of data and may eliminate noisy and irrelevant instances. In addition, this study searches the optimal feature weights for the relevant features in case retrieval process.

The rest of this paper is organized as follows: The next section presents the research background. Section 3proposes the evolutionary instance selection algorithm and describes the benefits of the proposed algorithm. Section 4 describes the application of the proposed algorithm. In the final section, conclusions and the limitations of this study are presented.

II. Research Background

2.1 Prior Research on Data Editing Techniques

Instance-based learning algorithms often faced the problem of deciding which instances to store for use during generalization in order to avoid excessive storage and time complexity, and to improve generalizability by avoiding noise and overfitting.[17] Many researchers have addressed the problem of training data reduction and have presented algorithms for maintaining an instance base or case base in instance-based learning algorithms.

Kuncheva[8] classified data editing techniques (or instance selection techniques) into the following three categories: *Condensed Nearest Neighbor rule*, *Generated or Modified Prototypes*, and *Two-Level Classifiers*. The following presents some basic concepts of each category as described by prior research. A detailed explanation may be found in the references in this paper.

Condensed nearest neighbor rule: Hart[5] made one of the first attempts to develop an instance selection rule. Hart's algorithm, the *Condensed Nearest Neighbor rule* finds a subset S of the training set T such that every member of T is closer to a member of S of the same class than to a member of S of a different class. Subsequent work extended Hart's algorithm, specifically the *Selective Nearest Neighbor rule*[11] and the *Reduced Nearest Neighbor rule*[3]. In addition, Wilson[16] introduced the *Edited Nearest Neighbor algorithm* and Tomek[15] proposed the *All k-NN method* of editing.

Generated or modified prototypes: This category is composed of techniques that establish new prototypes or adjust a limited number of instances. A large group of studies within this category are implemented by artificial neural networks (ANNs) including feature-map classifiers, learning vector quantiziers.[9]

Two-level classifiers: This category employs two or more classifiers and allocates a part of all instances to the classifier which appears most appropriate. Tetko & Villa[14] proposed the *Efficient Partition Algorithm* which is used to obtain an efficient partition of noisy instances, whose distribution is proportional to the complexity of the analyzed function. This is to focus the training of ANN on the most complex and informative domains of the data set and accelerate the learning phase. They concluded that the efficiently partitioned instances enhance the predictability of ANN in comparison with a random selection of instances.

Instance selection in instance-based learning algorithms may be considered as a method of knowledge refinement and it maintains the instance-base. In this sense, some researchers proposed many data editing techniques for maintaining the case-base in case-based reasoning systems. Smyth[13] presented an approach to maintenance which is based on the deletion of harmful and redundant cases from the case-base. In addition, McSherry[10] suggested an instance selection method in the construction of a case library in which evaluation of the coverage contributions of candidate instances are based on an algorithm called *disCover*. This algorithm reverses the direction of CBR to discover all cases that can be solved with a given case-base.

Although many different approaches have been used to address the problem of case authoring and data explosion for instance-based algorithms, there is little research on data editing application in business context. Thus, we propose a novel data editing technique for the financial forecasting in this study and address some new research issues for the business application of data editing techniques.

2.2 Genetic Algorithms

Genetic algorithm is a popular optimization method that attempts to incorporate ideas of natural evolution. Its procedure improves the search results by constantly trying various possible solutions with the some kinds of genetic operations. In general, the process of GA proceeds as follows.

First of all, GA generates a set of solutions randomly which is called an initial population. Each solution is called a chromosome and it is usually in the form of a binary string. After the generation of the initial population, a new population is formed to consist of the fittest chromosomes as well as offspring of these chromosomes based on the notion of survival of the fittest. The value of the fitness for each chromosome is calculated from a user-defined function. Typically, classification accuracy (performance) is used as a fitness function for classification problems.

In general, offspring are generated by applying genetic operators. Among various genetic operators, selection, crossover and mutation are the most fundamental and popular operators. The selection operator determines which chromosome will survive.

In crossover, substrings from pairs of chromosomes are exchanged to form new pairs of chromosomes. In mutation, with a very small mutation rate, arbitrarily selected bits in a chromosome are inverted. These steps of evolution continue until the stopping conditions are satisfied.[2][4]

III. GA Approach to Data Editing for CBR

As mentioned earlier, there are many studies on data editing for the instance-based learning algorithm. However, there are few studies on data editing for CBR. Thus, there are few relevant theories concerning data editing for CBR. This paper proposes the GA approach to data editing for CBR (GDCBR). The overall framework of GDCBR is shown in 〈Figure 3-1〉. In this study, the GA supports the simultaneous optimization of feature weights and selection of relevant instances.

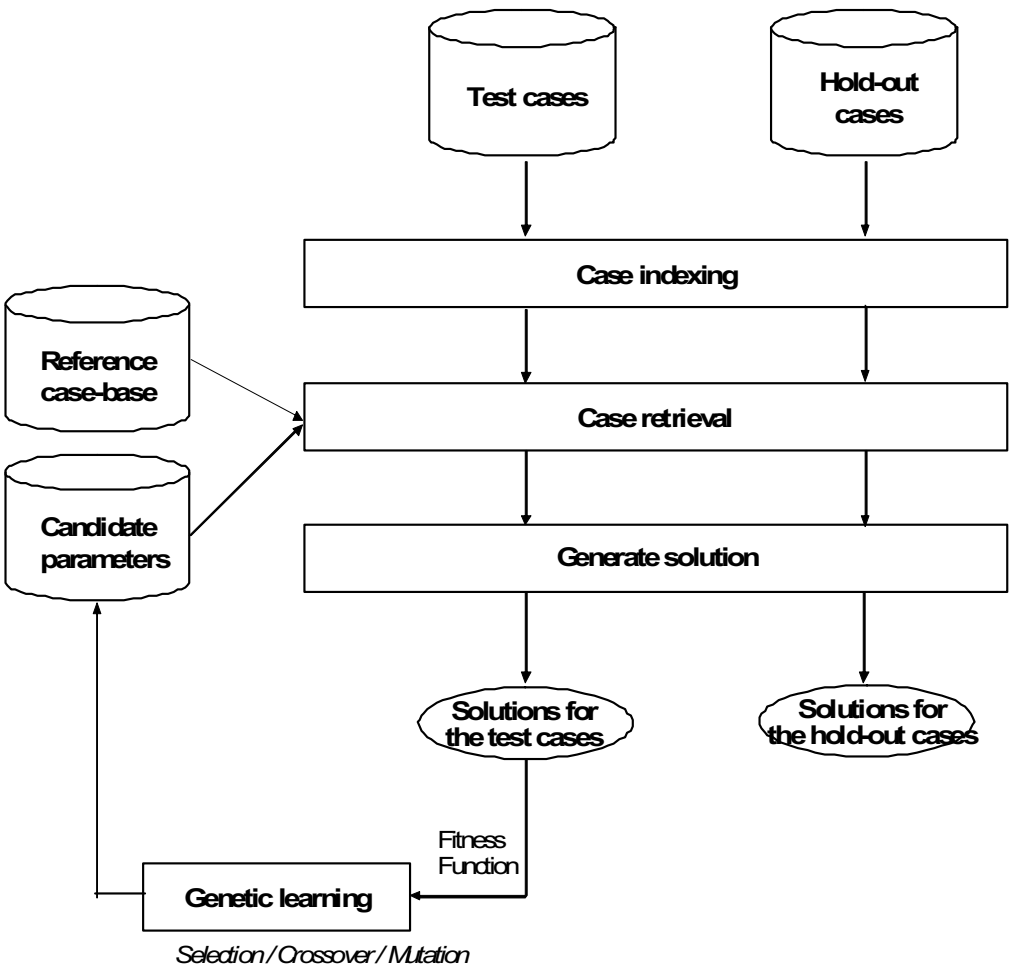

Figure 3-1. Framework of GDCBR
그림 3-1. GDCBR의 프레임워크

The detail explanation for each phase of GDCBR is presented as follows.

Step 1. For the first step, the system searches the space to find optimal or near-optimal parameters (feature weights and selection variables for each instance). To apply GA to search these optimal parameters, they have to be coded on a chromosome. The value of the code for instance selection is set to '0' or '1'. '0' means the corresponding instance is not selected and '1'means selected. Because a sign for each instance selection requires just 1 bit, so n bits are required to implement instance selection by GA where n is the number of total instances. On the other hand, the codes for feature weights are varied in some range specified.

The population (a set of seed chromosomes for finding optimal parameters) is initiated into random values before the search process. And, the encoded chromosome is searched to maximize the specific fitness function.

Step 2. In the second step, the parameters that are set in Step 1are applied to the CBR system and general reasoning process of CBR goes on. We use the weighted average of Euclidean distance for the each feature as a similarity measure. And, we use 1-NN(one-nearest neighbor) matching as a method of case retrieval. After adoption reasoning process for all of test cases, the values of the fitness function (f_T) for the items of test set T are updated.

Step 3. In third step, the process of GA's evolution goes on towards the direction to maximize the value of the fitness function. It includes selection of the fittest, crossover and mutation. Step 2 and 3 are iterated again and again until the stopping conditions are satisfied.

Step 4. In the last stage, the system determines the parameters - the optimal weights of features and selection of instances - whose performance for the test data is the best. And, it applies them to the hold-out data to check the generalizability of the selected parameters. Sometimes, optimized parameters by GA fit to the test data, but they don't fit to the unknown data, i.e. overfitting. Thus, this step is required to check the possibility of overfitting.

IV. Application to the Korean Stock Market Data

This section applies GDCBR to the Korean stock market prediction. The efficiency and effectiveness of GDCBR may be properly tested because the stock market data is very noisy and complex. Many studies on stock market prediction using artificial intelligence techniques were performed in the past decade. Some of them, however, did not produce outstanding prediction accuracy partly because of the tremendous noise and non-stationary characteristics in stock market data. If these factors are not appropriately controlled, the prediction system does not produce significant performance.

4.1 Research Data

The application data used in this study consists of technical indicators and the direction of change in the daily Korea stock price index (KOSPI). The total number of samples is 2928 trading days, from January 1989 to December 1998. This study divides the samples into ten data sets according to the trading year. Experiments are repeated ten times for each data set to reflect specific knowledge as time passes.

The direction of daily change in the stock price index is categorized as "0" or "1". "0" means that the next day's index is lower than the today's index, and "1" means that the next day's index is

higher than today's index. We select twelve technical indicators as feature subsets by the review of domain experts and prior research. 〈Table 4-1〉 gives selected features and their formulas.

Table 4-1. Selected features and their formulas
표 4-1. 선정된 특성들과 그들의 산식

Name of feature	Formula
Stochastic %K	$\frac{C_t - LL_{t-5}}{HH_{t-5} - LL_{t-5}} \times 100$
Stochastic %D	$\frac{\sum_{i=0}^{n-1} \%K_{t-i}}{n}$
Stochastic slow %D	$\frac{\sum_{i=0}^{n-1} \%D_{t-i}}{n}$
Momentum	$C_t - C_{t-4}$
ROC (rate of change)	$\frac{C_t}{C_{t-n}} \times 100$
LW %R (Larry William's %R)	$\frac{H_n - C_t}{H_n - L_n} \times 100$
A/D Oscillator (accumulation/distribution oscillator)	$\frac{H_t - C_{t-1}}{H_t - L_t}$
Disparity 5 days	$\frac{C_t}{MA_5} \times 100$
Disparity 10 days	$\frac{C_t}{MA_{10}} \times 100$
OSCP (price oscillator)	$\frac{MA_5 - MA_{10}}{MA_5}$
CCI (commodity channel index)	$\frac{(M_t - SM_t)}{(0.015 \times D_t)}$
RSI (relative strength index)	$100 - \frac{100}{1 + \frac{\sum_{i=0}^{n-1} Up_{t-i}/n}{\sum_{i=0}^{n-1} Dw_{t-i}/n}}$

Notes: C, Closing price; L, Low price; H, High price; LL_n, Lowest low price in the last n days; HH_n, Highest high price in the last n days; M, Moving average of price; Mt, $\frac{(H_t + L_t + C_t)}{3}$ SMt, $\frac{\sum_{i=1}^{n} M_{t-i+1}}{n}$ Dt, $\frac{\sum_{i=1}^{n} |M_{t-i+1} - SM_t|}{n}$ Up, Upward price change; Dw, Downward price change.

4.2 Experiments

Experiments are carried out for the following three models:

Whole training data. The whole reference cases are used as the training data. This is the conventional method of data analysis.

Selected instances with GDCBR. Experiments on stock market data are implemented using GDCBR. The procedure of the experiment is as follows. The GA searches for optimal or near-optimal feature weights and relevant instances for CBR. As mentioned earlier, this study needs two sets of parameters: The weight codes for the relevant features and the codes for data editing.

This study uses the following encoding for the strings: 12 input features are used. Thus, the first 12 bits represent the feature weights for the relevant features. These bits are searched from 0 to 1. The following bits are instance selection codes for the training data. The chromosome of these bits consists of n genes (where n is the number of initial training instances), each one with two possible states: 0 or 1. "1" means the associated instance is selected into the analysis and "0" means the associated instance is not chosen.

The encoded chromosomes are searched to maximize the fitness function. The fitness function is specific to applications. In this study, the objective of the study is to determine appropriate the feature weights and instance selection of CBR systems, which produce the highest prediction accuracy for the test data. Thus, we set the prediction accuracy of the test data as the fitness function for GA.[6][12] Mathematically, the fitness

function (f_T) for the test set T can be expressed as Equation (1):

$$f_T = \frac{1}{n}\sum_{i=1}^{n} CA_i$$
$$CA_i = 1 \text{ if } PO_i = AO_i \text{ for the item } I_i \quad (1)$$
$$CA_i = 0 \text{ if } PO_i \neq AO_i \text{ for the item } I_i$$

where CA_i is the classification accuracy for the i th test case, I_i, which is denoted by 1 or 0 ('correct'=1, 'incorrect'=0), PO_i is the predicted output from the model for the ith test case, AO_i is the actual output from the model for the ith test case and test set T is $\{I_1, I_2, I_3, \cdots, I_n\}$.

For the controlling parameters of the GA search, the population size is set at 100 organisms and the crossover and mutation rates are varied to prevent CBR from falling into a local minimum. The value of the crossover rate is set at 0.7 while the mutation rate is 0.1. For the crossover method, the uniform crossover method is considered better at preserving the schema, and can generate any schema from the two parents, while single-point and two-point crossover methods may bias the search with the irrelevant position of the variables. Thus, this study performs crossover using the uniform crossover routine. For the mutation method, this study generates a random number between 0 and 1 for each of the variables in the organism. If a variable gets a number that is less than or equal to the mutation rate, then that variable is mutated. As the stopping condition, only 100 generations are permitted.

4.3 Experimental Results and Discussions

This study compares GDCBR to the conventional CBR. GDCBR uses the GA to determine the feature weights and learns the patterns of the stock market data from the selected instances through an evolutionary search process. For the conventional CBR model, about 20% of the data is used for holdout and 80% for reference case-base. The holdout data is used to test the results with the data that is not utilized to develop the model. The number of the reference instances in the conventional CBR and the number of the selected instances within the reference instances in GDCBR for each year are presented in 〈Table 4-2〉.

Table 4-2. Number of cases
표 4-2. 사례수

Data set	Year										Total
	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	
Original reference cases for CBR	232	233	234	236	237	237	235	235	234	234	2347
Selected reference cases for GDCBR	84	93	76	73	83	95	63	83	92	78	820
Holdout cases for two models	57	58	58	58	59	59	58	58	58	58	581

〈Table 4-3〉 describes the average prediction accuracy of each model for the holdout cases.

Table 4-3. Average predictive performance (hit ratio: %)
표 4-3. 평균 예측성과 (적중율: %)

Year	Conventiona ICBR	GDCBR
1989	56.1	56.1
1990	50.0	58.6
1991	51.7	56.9
1992	44.0	51.7
1993	49.2	54.2
1994	52.5	54.2
1995	58.6	55.2
1996	62.1	60.3
1997	51.7	51.7
1998	44.8	53.4
Average	52.1	55.3

In 〈Table 4-3〉, GDCBR outperforms the conventional CBR model by 3.2% for the holdout data. This result may be caused by the benefits of the data editing through evolutionary search techniques.

V. Concluding Remarks

Some of prior research tried to optimize the controlling parameters of CBR using global search algorithms. In general, they only focused on the optimization of the feature weights of CBR. Others placed little emphasis on the optimization of the learning algorithm itself, but most studies focused little on instance selection for CBR.

In this paper, we use the GA for CBR in two ways. We first use the GA to determine the feature weights. In addition, we adopt the evolutionary instance selection technique. This directly removes irrelevant and redundant instances from the training data. We conclude that GA-based learning and the data editing technique significantly outperforms the conventional CBR model in stock market prediction.

The prediction performance may be more enhanced if the GA is employed not only for data editing but also for relevant feature selection, and this remains a very interesting topic for further study. Although data editing is a direct method of noise and dimensionality reduction, feature selection effectively reduces the dimensions of feature space.

References

[1] Bradley, P.(1994), "Case-based reasoning: Business applications," Communication of the ACM, Vol. 37 No. 3, pp. 40-43.

[2] Chiu, C.(2002), "A case-based customer classification approach for direct marketing," Expert Systems with Applications, Vol. 22, pp. 163-168.

[3] Gates, G. W.(1972), "The reduced nearest neighbor rule," IEEE Transactions on Information Theory, Vol. 18, No. 3, pp. 431-433.

[4] Han, J. and M. Kamber(2001), "Datamining: Concepts and Techniques," Morgan Kaufmann Publishers: San Francisco, CA.

[5] Hart, P. E.(1968), "The condensed nearest neighbor rule," IEEE Transactions on Information Theory, Vol. 14, pp. 515-516.

[6] Kim, K.(2004), "Toward global optimization of case-based reasoning systems for financial forecasting," Applied Intelligence, Vol. 21, No. 3, pp. 239-249.

[7] Kolodner, J.(1993), "Case-based Reasoning," Morgan Kaufmann, San Mateo, CA.

[8] Kuncheva, L. I.(1993), "'Change-glasses' approach in pattern recognition," Pattern Recognition Letters, Vol. 14, pp. 619-623.

[9] Kuncheva, L. I.(1995), "Editing for the k-nearest neighbors rule by a genetic algorithm," Pattern

Recognition Letters, Vol. 16, No. 8, pp. 809-814.
〔10〕 McSherry, D.(2000), "Automating case selection in the construction of a case library," Knowledge Based Systems, Vol. 13, No. 2-3, pp. 133-140.
〔11〕 Ritter, G. L., H. B. Woodruff, S. R. Lowry, and T. L. Isenhour(1975), "An algorithm for a selective nearest neighbor decision rule," IEEE Transactions on Information Theory, Vol. 21, No. 6, pp. 665-669.
〔12〕 Shin, K. S., and I. Han(1999), "Case-based reasoning supported by genetic algorithms for corporate bond rating," Expert Systems with Applications, Vol. 16, pp. 85-95.
〔13〕 Smyth, B.(1998), "Case-base maintenance," Proceedings of the 11th International Conference on Industrial & Engineering Applications of Artificial Intelligence & Expert Systems, pp. 507-516.
〔14〕 Tetko, I. V., and A. E. P. Villa(1997), "Efficient partition of learning data sets for neural network training," Neural Networks, Vol. 10, No. 8, pp. 1361-1374.
〔15〕 Tomek, I.(1976), "An experiment with the edited nearest neighbor rule," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 6, No. 6, pp. 448-452.
〔16〕 Wilson, D. L.(1972), "Asymptotic properties of nearest neighbor rules using edited data," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 2, No. 3, pp. 408-421.
〔17〕 Wilson, D. R., and T. R. Martinez(2000), "Reduction techniques for instance-based learning algorithms," Machine Learning, Vol. 38, pp. 257-286.

저 자 소 개

김 경재
중앙대학교 경영대학 경영학사
KAIST 경영공학석사
KAIST 경영공학박사
현재: 동국대학교 경영정보학과 조교수
논문 : Applied Intelligence, Expert Systems, Expert Systems with Applications, Intelligent Data Analysis, Intelligent Systems in Accounting, Finance & Management, Neural Computing & Applications, Neurocomputing 등의 국제학술지에 논문을 게재하였음