Intelligent Search Mechanism based on Neuro-fuzzy System for the Distributed Object Groups*

Romeo Mark A. Mateo and Jaewan Lee
School of Electronic and Information Engineering, Kunsan National University
{rmmateo, jwlee}@kunsan.ac.kr

Abstract

Current research in distributed systems focuses on object group models which are designed for scalable systems and provide accessible objects to the entire group. However, dynamic grouping using knowledge extraction from group of objects is not considered which provides accurate search for appropriate objects. In this paper, an integration of intelligent models in distributed object groups is presented in intelligent distributed framework. This paper proposes a locator service for the framework which implements the intelligent search mechanism based on neuro-fuzzy system. To have an accurate search of objects, the fuzzy system of the locator service is trained using the neuron-fuzzy algorithm. The proposed method is compared to other methods in accuracy of classification and result shows that it outperformed other algorithm based on processing time and accuracy.

1. Introduction

Implementation of services based on Common Object Request Broker Architecture (CORBA) is popular design for robust systems in distributed environment [1,2]. The availability of CORBA services for transactions, concurrency control, security and access control, events and persistent objects make it a desirable choice for use in many applications that are intended for use within an organization or a related of organizations [3]. Moreover, designing the complex interactions of object and services are challenging tasks for system architects. The previous researches of distributed environment focus on efficient search [4,5,6] and load distribution [7,8,9] of objects. Moreover, managing the distributed objects is necessary by means of object group models. Object group models primarily use grouping techniques in managing the system based on distributed objects. In object groups, the communications reflect the inter-dependence and take place from one group to another. To define, an object group is a set of objects related logically. A group acts as a logical addressable entity where an entity that requests a service from a group is a client of the group. The properties of the object groups [10] are shown below.

- Group behavior – A set of action taken by the members of the group.
- Group reference – Acts same as an object reference where it designates a single object while group reference designates a set of objects.
- Group membership – This determines the group where the object belongs.
- Encapsulation of groups – This is a property of an object group to behave like a singleton object where it can act as an identifiable, encapsulated entity that may be invoked by a client.

* This research was supported by the Program for the Training of Graduate Students in Regional Innovation which was conducted by the Ministry of Commerce Industry and Energy of the Korean Government.

The effective management of object group is critical for coordination of objects. However, previous researches on object group models lack of intelligent resource finding using the properties from the group of objects which provides accurate search of appropriate objects.

Fuzzy system is used to manage the problem of uncertainty by assigning a degree of membership in each data where it uses the fuzzy set to represent a suitable mathematical tool for modeling of imprecision and vagueness [11]. Neuro-fuzzy classifiers are a hybrid of fuzzy systems which are used to represent knowledge in an interpretable manner [12]. The algorithm borrows the learning ability of neural networks to determine the membership values. Fuzzy sets are trained by neuron-fuzzy algorithm to classify the data and this strategy is suitable in distributed object grouping to efficiently classify objects.

In this paper, the intelligent distributed framework is presented and proposes an intelligent search mechanism based on neuro-fuzzy system. The components of the framework are consisted of services, objects and agents which implement the efficient management of distributed objects by object group bindings and the integration of multi-agent system for intelligent resource finding in the distributed environment. This paper focuses on the locator service using the intelligent search mechanism based on the proposed classification model. The locator service processes the client request in classification method by collaborating with the mobile agents to select the appropriate object for the request. The grouping service adjusts the classification model by using neuro-fuzzy algorithm and distributes the new values to mobile agents. To evaluate the design performance, other learning algorithms are used to compare the accuracy of the proposed method based on the processing time and accuracy.

2. Related work

2.1. Intelligent systems

Intelligent systems are characterized by its ability of optimizing the system based on constraints in the environment. Intelligent controls (IC) are the first application field of intelligent systems. ICs are used for flexible and scalable automatic control systems [13]. Recently, various researchers and industries use software agents to implement intelligent systems [14,15]. Current researches in agent technology focus on providing industries with a new approach of solving problems in a distributed manner, new software tools and automated functions of the system [16]. In multi-agent systems, communications, resource sharing and cooperation to solve problems are considered. Many organization and researcher are working on standards like in FIPA [17] and other technologies of multi-agents. Also, more contributions in designing and implementing new ideas in multi-agent system are at work. In this paper, intelligence is integrated in the distributed environment by using multi-agent approach. Agents in the proposed framework automate the task and services transparently. Moreover, the mobile agents store the fuzzy rule and membership values to be use in processing the intelligent resource finding.

2.2. Neuro-fuzzy system

Neuro-fuzzy systems are fuzzy classifiers and uses neural networks for learning by performing induction of the structure and adaptation of the connection weights [12]. The NEFCLASS is a fuzzy classification system which borrowed the learning ability of the neural network for learning fuzzy rules and fuzzy sets. Currently, research works use the fuzzy system for data analysis and pattern classification [18]. The fuzzy classifiers provide a means

to extract fuzzy rules for information mining that leads to comprehensible method for knowledge extraction from various information sources. In the proposed intelligent search mechanism, neuro-fuzzy algorithm is used to construct the fuzzy system for classification. The classification model is used to search the appropriate object based on the request contents.

3. Intelligent distributed framework for object groups

In the previous study, a dynamic replication scheme [19] to handle large request from clients and a highly scalable and efficient load distribution using the cooperation model for object group [20] are proposed. In this paper, the intelligent distributed framework for objects groups is presented in addressing the use of intelligent models to optimize the system performance of distributed object groups. The framework's global view is shown in Figure 1.

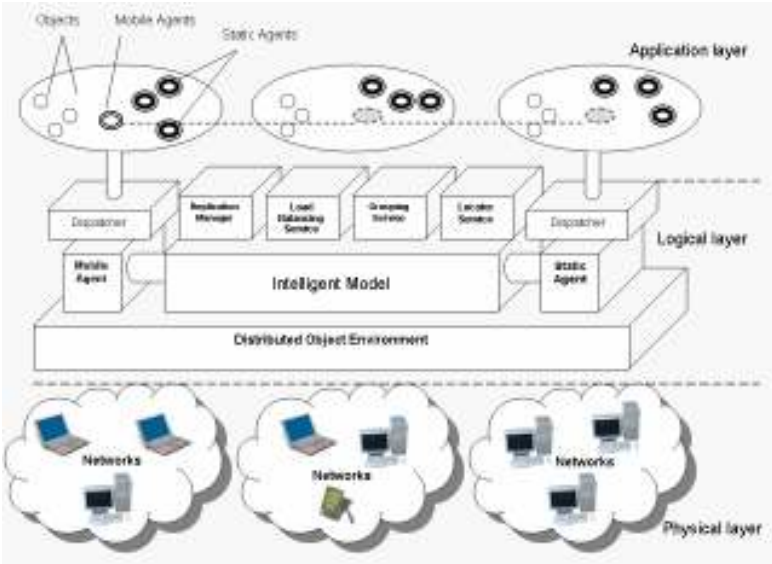

Figure 1. The global view of intelligent distributed framework divided in three layers.

This is consisted of three layered components: physical, logical and application layers. In the framework, the multi-agent systems implement task automation and the distributed components transparently manage clients and servers. The physical layer represents networks of different computers, like personal computers (PC), laptops and personal digital assistants (PDA). The logical layer acts as the middleware where services are transparently operating on serving the clients. Interaction of clients and objects are handled by the logical layer. Agents are working for resource finding and optimizing the system performance. The application layer is consisted of objects and agents utilizing the distributed environment. The intelligent distributed framework meets the following design requirements:

- Dynamic replication with load balancing and fault tolerance: quality of service is important to clients where object replication is implemented to handle the large request of service. In contrast of QoS, the management of the replicated is necessary and each load distribution must be balanced to provide reliable services to clients and provide an optimal system performance. Also, improves fault tolerance of the system by using object replicas in time of object processing failures.

- Scalability and transparency: the system scalability is important where the expansion of services and additional schemes are considered. By using the CORBA framework, these changes are transparent to clients and servers so that these entities do not need to know how to configure and where to find the resources.
- Group management support: object grouping offers efficient management of objects. The group binding scheme manages the objects within the group and enhances the search for objects. Moreover, efficient management of object grouping is considered.
- Multi-agent support: multi-agents are used for the intelligent model. An agent acts as an individual which promotes intelligence on how to utilize the objects autonomously. Also, the inter-communication from agents is considered in able to cooperate on task and share the knowledge to other agent.
- Mobility support: support for mobile agent's migration is used for searching of information or providing service to a local node.
- Data mining support: support for data mining is considered to extract the knowledge represented by rules from data contents of objects. Significant rules are integrated to the intelligent model in implementing intelligent system of the framework.

3.1. Components of the intelligent distributed framework

Grouping service. Grouping service (GS) manages object groupings in the intelligent distributed framework. When an object registers to the system, a dynamic classification is done by adjusting the properties of classification model. The properties of objects are configured by the object providers. Object groups are bounded to other object group to make more efficient objects search. Figure 2 presents the dynamic group bindings of object groups where group information is shared to other object groups.

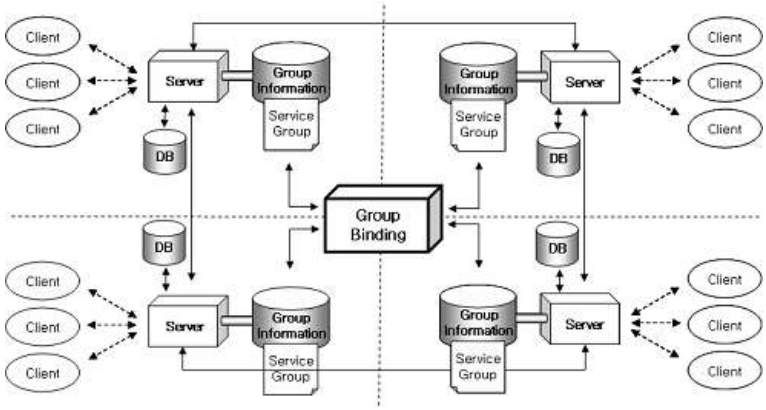

Figure 2. Group bindings of object groups where each group shares its object information.

Locator service. Client requests are queried through the object groups of the system. However, the bounded object groups contain large number of objects information which is difficult to find the most appropriate object for the request. Locator service classifies the request content to choose the appropriate object. The locator service processes the clients request by sending the processed input to mobile agents and calculates the membership

function. Each mobile agent which is initialized by the grouping service represents object groups assigned by the group service. A mobile agent uses its fuzzy rule and process the classification. Section 4 discusses more details of the method.

Replication service. The replica service (RS) creates and manages object replicas in the server already presented in previous work [19]. RSs within the server are coordinating to other RS managing the replicated objects. If an object has changed its values then RS of that object communicates through other RS to inform the changes. Monitoring and analyzing the access of clients to the objects are necessary in adapting the system requirements where the result of the analysis is the creation or termination of an object replica.

Load balancing service. The load balancing service is responsible for balancing the load initialized by clients accessing the objects. A load is defined as a single access of client to an object. In accessing objects, the loads are distributed to the object replicas and follow the adaptive scheme from the previous work [20]. The load balancing service coordinates the object groups to distribute the loads within the object replicas.

4. Intelligent search mechanism based on neuro-fuzzy system

The intelligent distributed framework implements the intelligent search mechanism based on NEFCLASS [12]. A server can host one or several objects and objects are grouped by the grouping service using its properties as an input to the fuzzy system. After the procedures, the grouping service creates mobile agents and assigns each fuzzy rules and membership values to the mobile agents. In Figure 3, mobile agents are represented by A_1, A_2,…, A_n, where n is the number of fuzzy rules. The locator service preprocesses the client requests to convert into input parameters represented by x_1 and x_2 and classify the request to the mobile agents. Each input is mapped through the mobile agents to be processed and replies the weighted value.

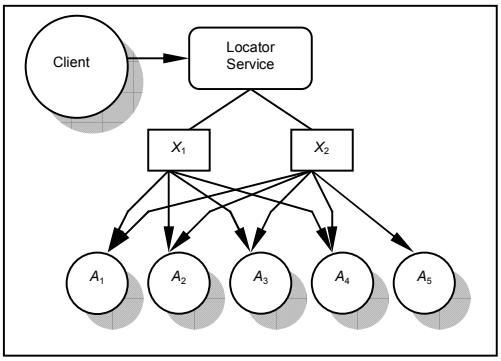

Figure 3. Locator service sends the input parameters to mobile agents in processing its classification.

4.1. Learning the structure of fuzzy system

4.1.1. Fuzzy rule generation. The initial fuzzy rules are combinations of the fuzzy sets from the procedure of finding the highest degree of membership for each value p_i from pattern P. This combination of fuzzy sets is the antecedent of a prospective rule. A suitable consequent for each antecedent is determined by adding up the degrees of fulfillment for all patterns separately for each class. The consequent is set to the class label that obtains the largest sum.

Rules from the training are determined by computing performance values for each rule shown in Equation 1 where c is a class index of p_i. If a rule correctly classifies a pattern, the degree of fulfillment value is added to its performance value, if not, the degree of fulfillment is subtracted. The best rules are stored and used for the proposed classification model.

$$performance_j = C_j(c) - \sum_{k \in \{1,\ldots,m\}} C_j(k) \tag{1}$$

After generating the best rules, fuzzy set learning is processed based on rules extracted. The procedure performs iteration until a certain criterion is achieved. A maximum iteration of building the model is used for the proposed search indicated by *max*. After the procedure, each fuzzy rule and corresponding membership values are stored by mobile agents.

4.1.2. Adjusting the fuzzy sets. After determining the fuzzy rules, fuzzy sets are trained. The calculation of the delta value is given by subtracting the target value to the value from the activation function in Equation 2.

$$\delta_c = t_i - activation(c_i) \tag{2}$$

The value from Equation 2 is used for calculating the error in Equation 3

$$e_R = o_R(1 - o_R)\sum_{c \in U_3} W(R, y)\delta_c \tag{3}$$

where e_R is the error factor for calculating the fuzzy sets. This procedure implements the learning weights between the rule and output unit. Find x' such that

$$W(x', R)(p_i') = \min_{x \in U_1}\{W(x, R)(p_i)\} \tag{4}$$

then the fuzzy sets are adjusted using the parameters from Equation 5.

$$\begin{aligned} \delta_b &= \sigma \cdot e_R \cdot (C\max_i - C\min_i) \cdot \mathrm{sgn}(p_i - Ccen_\mu); \\ \delta_a &= -\sigma \cdot e_R \cdot (C\max_i - C\min_i) + \delta_b; \\ \delta_c &= \sigma \cdot e_R \cdot (C\max_i - C\min_i) + \delta_b; \end{aligned} \tag{5}$$

The adaptation of the fuzzy sets is carried out by changing the parameters of its membership function in a way it increase or decrease respectively. Then apply the changes to W(x',R) if this does not violate against a given constraint.

4.2. Intelligent search using mobile agents

The locator service collaborates with mobile agents in performing the classification of client request where it sends the input parameters to mobile agents. These inputs contain values which describes the request. An example of input is a client searching for the appropriate health specialist so he or she needs to choose the symptoms specified from the system represented by x_i with its corresponding value. Equation 6 shows the function of choosing the candidate agent after the classification. A_n are agents and x are input parameters where an agent that has the maximum value from aggregation is the candidate agent.

$$candidate\_agent = \max\{\sum_{n=1}^{N} A_n(x)\} \tag{6}$$

Moreover, the processing of a single agent is presented in Equation 7. The output value from membership function in the fuzzy system is sent back to the locator service.

$$A_n(x) = \sum_{j=1}^{J} \mu_n(x_j) \tag{7}$$

The chosen candidate agent also means that it has the appropriate object for the client requests. The request is forwarded to candidate agent and finds the appropriate object using fuzzy least load algorithm [20] to implement the load balancing.

5. Experimental evaluation

The Weka 3.5, a popular tool for evaluating data mining algorithms, and the NEFCLASS-J, an implementation of neuro-fuzzy classification in Java were used in evaluating the performance of the proposed classification method. The proposed framework used Borland Visibroker 7.0 for CORBA and Jade Framework for multi-agent system. The simulation used the grouping service to group the objects. In Figure 4, finding the appropriate object by the locator service is shown. It uses the request content to classify the inputs from the properties of the mobile agents. Also, replication and load balancing services are executing at this time.

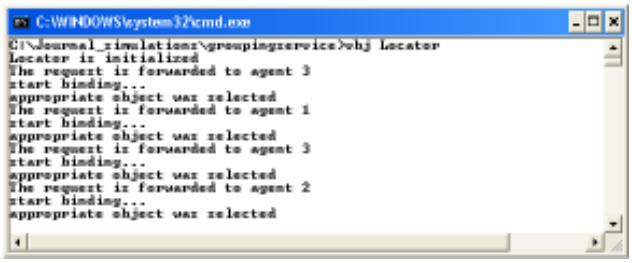

Figure 4. Locator service searching the appropriate object

5.1. Performance measure

In measuring the performance of classification method, cross-validation of the classified instances was used and the formulation is shown in Equation 8. This is done by initializing the classification model, processing the input pattern in the model and introducing again the same input patterns to the system. $f(x_k)$ is the function from model to classify the pattern x_k. If the function classified correctly the data, then it increase to 1 and the iteration depends on the number of data represented by N. The incremented value of x_k is divided to N. The procedures from Equation 8 have C folds of cross validation and get the average.

$$CC = \frac{1}{C}\sum_{n=1}^{N}\frac{1}{N}\sum_{k=1}^{K} f(x_k) \tag{8}$$

5.2. Result

In our simulation, a synthetic data was used to perform the proposed data mining model. A standard distribution of generating random data, which contains 5 attributes

and 100 tuples were done, and then make these data as object properties. These input patterns were used to process to the classification algorithms. To perform the online simulation of the proposed algorithm, the properties of each object contained pattern from the synthetic data. All objects were classified by the grouping procedure using neuro-fuzzy classification (NFC). The trained fuzzy system containing fuzzy sets and fuzzy rule values were used to configure the classification structure of the proposed intelligent search mechanism. Comparing the performance to other algorithms, two accurate learners were selected; multilayered perceptron (MLP) and radial basis function (RBF).

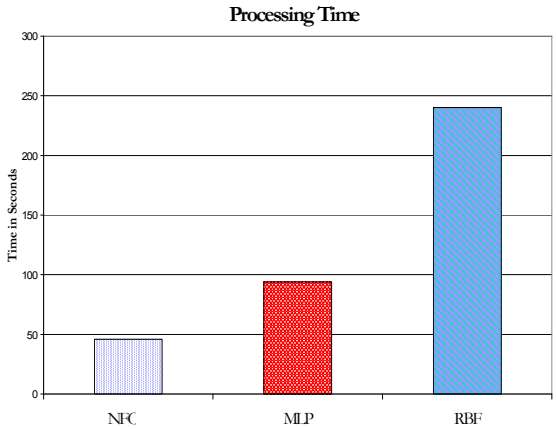

Figure 5. Processing time performance in building the model for the classification

In the simulation time shown in Figure 5, the fastest was the NFC (48 sec.) compared to MLP (97 sec.) and RBF (238 sec.). The proposed algorithm optimized the model more efficient compared to other algorithm in a short time. The result of accuracy based on the number of epoch is shown in Figure 6. NFC has a constant result of accuracy from 10 to 100. The goal of the learning schemes is to process the patterns and use the errors as a factor of learning which is expected to have a less accuracy at the start. However, in less looping of epoch, NFC is already accurate on classifying compared to other methods. Also, it was observed that from the start of the epoch (10 epochs), NFC has already adjusted an accurate system compared to MLP and RBF which is better of 7% and 17%, respectively.

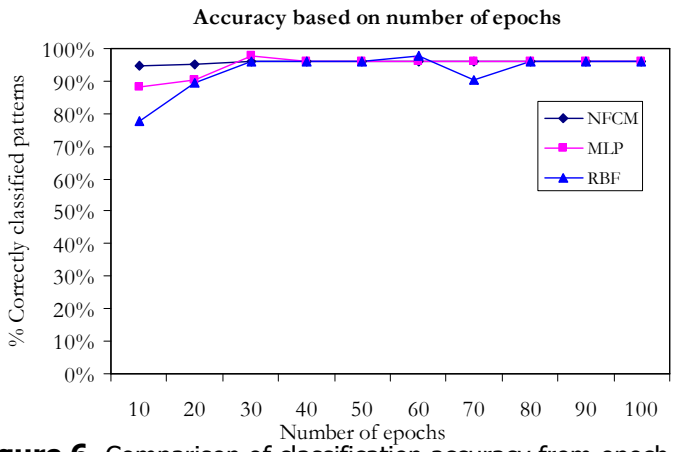

Figure 6. Comparison of classification accuracy from epoch counts

6. Conclusions

The integration of intelligent system contributes to the efficiency of system management. In this paper, intelligent search mechanism based on neuro-fuzzy classification for intelligent distributed framework is proposed. The components of the proposed framework are consisted of services, objects and agents which implement the efficient management of distributed object and the intelligent resource finding in distributed environment. Based on object properties, the grouping service constructs the classifier for intelligent search mechanism while the locator service implements the intelligent search for finding appropriate objects. CORBA and Jade Framework were used for simulating the proposed mechanism. Other methods were compared for classification accuracy. Result shows that NFC outperformed other algorithm in terms of processing time as the fastest to build the classification model and accuracy by having 7% and 17% better compared to MLP and RBF, respectively.

References

[1] Object Management Group, available at www.omg.org

[2] Z. Yang and K. Duddy, CORBA, "A Platform for Distributed Object Computing", Operating Systems Review, ACM, Vol. 30, No. 2, 1996, pp. 4-31.

[3] Coulouris, G., Dollimore, J., Kindberg, T., "Distributed Systems: Concepts and Design, Fourth Edition", Addisson-Wesley, 2005, pp. 799-800.

[4] E. Badidi, R. K. Keller, P. G. Kropf, V. Van Dongen, "The Design of a Trader-based CORBA Load Sharing Service", In proceedings of 12th International Conference on Parallel and Distributed Computing Systems, 1999, pp. 75-80.

[5] Y. K. Kwok, L. S. Cheung, "A New Fuzzy-Decision based Load Balancing system for Distributed Object Computing", Journal of Parallel and Distributed Computing, Vol. 2, No. 64, 2004, pp. 238-253.

[6] M. Van Steen, G. Ballintijn, "Achieving Scalability in Hierarchical Location Services", In proceeding of the 26th International Computer Software and Applications Conference, 2002.

[7] Y. M. Chen, S. C. Wang, Framework of Agent-based Intelligence System with Two-stage Decision-Making Process for Distributed Dynamic Scheduling, Applied Soft Computing, Vol. 7, 2007, pp. 229-245.

[8] Othman, C. O'Ryan, and D. C. Schmidt, "The Design of an Adaptive CORBA Load Balancing Service", IEEE Distributed Systems Online, Vol. 2, No. 4, 2001.

[9] T. Schnekenburger, "Load Balancing in CORBA: A Survey of Concepts, Patterns, and Techniques", The Journal of Supercomputing, Vol. 15, 2000, pp. 141-161.

[10] P. Felber and R.Guerraoui, "Programming with Object Groups in CORBA", Concurrency, IEEE, Vol. 8, No. 1, 2000, pp. 48-58.

[11] Zadeh, L. A., Fuzzy Sets, Information and Control, 1965, pp. 338-353.

[12] A. Klose, A. Nürnberger, D. Nauck, and R. Kruse, "Data Mining with Neuro-fuzzy Models", Data Mining and Computational Intelligence, Springer-Verlag, 2001 pp. 1-36.

[13] Brown M., Haris C., "Neurofuzzy Adaptive Modeling and Control", Prentice Hall, 1994, pp. 4-11

[14] J. Hendler, "Introduction to the Special Issue: AI, Agents, and the Web", Intelligent Systems, IEEE, Vol. 21, No. 1, 2006, p. 11.

[15] L. Yang, and F. Y. Wang, "Driving into Intelligent Spaces with Pervasive Communications", Intelligent Systems, IEEE, Vol. 22, No. 1, 2007, pp. 12–15.

[16] V. Marik, D. McFarlane, "Industrial Adoption of Agent-based Technologies", Intelligent Systems, IEEE, Vol. 20, No. 1, 2005, pp. 27 – 35.

[17] FIPA, "The Foundation of Intelligent Physical Agents", available at http://www.fipa.org

[18] S. Mitra and Y. Hayashi, "Neuro-fuzzy Rule Generation: Survey in Soft Computing Framework", IEEE Trans. in Neural Networks, Vol. 11, 2000, pp. 748-768

[19] R. M. A. Mateo, M. Lee and J. Lee, "Dynamic Replication Scheme for Load Balancing the Distributed Object Group", In proceedings of International Conference in Hybrid Information Technology, 2006.

[20] R. M. A. Mateo, I. Yoon and J. Lee, "Cooperation Model for Object Group using Load Balancing", International Journal of Computer Science and Network Security, Vol. 6, No. 12, 2006, pp. 138-147.

Authors

Romeo Mark A. Mateo

Romeo Mark A. Mateo received his B.S. degree in Information Technology from West Visayas State University, Philippines in 2004 and M.Eng. degree in Information and Telecommunications from Kunsan National University, South Korea in 2007. He is now a Ph.D. candidate in Information and Telecommunications Engineering at the Kunsan National University, South Korea. Also, he is currently working as research assistant at the Distributed Systems Laboratory (DSL). His research interests include distributed systems, mobile computing, data mining and artificial intelligence.

Jaewan Lee

Jaewan Lee received his B.S., M.S., and Ph.D. degrees in Computer Engineering from Chung-Ang University in 1984, 1987, and 1992, respectively. Currently, he is a professor at the School of Electronic and Information Engineering and supervisor of the Distributed Systems Laboratory (DSL) in Kunsan National University, Kunsan City, South Korea. His research interests include distributed systems, database systems, data mining and computer networks.