A Service Retrieval Model with Enhanced Dominance-Based Rough Sets

Bin Tang, Leqiu Qian, Ruzhi Xu,Yunjiao Xue
Department of Computer Science and Engineering, Fudan University, Shanghai, China
{tangbin, lqqian, yjxue, rzxu}@fudan.edu.cn

Hui Tang
Technical Support Section, Business Support Center, Guangdong Mobile Communication Co., Ltd., Guangzhou, China

Abstract

As an important part of service-oriented software engineering, service retrieval receives much attention from industry and academia, but which still leaves much to be expected: how the truth of the advertisement of service can be assured; how incomplete retrieval requirement can be dealt with; how a service fit a user's need, existing methods based on ontology service model, which can be roughly classified as based on logic reasoning or similarity computing, still involve much effort of users to judge the fitness of the service in the result set. The paper introduces a new approach with Dominance-Based Rough Sets to obtain knowledge from past applications of services to deal with the three problems. Experiments verify the efficiency of the approach introduced in the paper.

1. Introduction

With an increasingly abundant web service market, it is very important that the services needed can be retrieved quickly, precisely and easily. The quality of service retrieval has direct influence on the quality of service reuse, the matching of service composition and substitution, and the plug and play of services. So as an important part of service-oriented software engineering, service retrieval receives much attention from industry and academia.

The performance of a service retrieval system is measured from the recall, precision, efficiency and easiness [9]. In the paper efficiency is broadly measured by the time from the user beginning to give a retrieval requirement to after his selection of the services needed finally, which can be divided into three parts: the time that users use to give the retrieval requirement, the time that the system use to retrieve services based on retrieval requirement, the time that the user use to select services from the result retrieval set. In literature, the retrieval time usually only include the middle part time. Easiness is the measured by the efforts needed to retrieve the needed services.

According to the richness of the information contained in the service description model, web service description and retrieval models can be divided into two categories [2][3]:

- Syntactical level

In description, this category of models emphasize the Syntax of the interface of web service and lack of constraints of behavior, in retrieval, them mostly base on key words. The representative systems based on this category of models are the UDDIs of IBM, Microsoft, SUN, which are simple to realize but the retrieval effect are not desirable [6][7].

- Semantic level

This kind of service models introduces ontology to describe services thoroughly and provide description of the function and behavior, which can be expected to give good retrieval result. The matching of retrieval requirement and service description is based on logic reasoning and the result can be divided into five classes according to the matching degree: exact match, generalized match, specialized match, partial match and no match. Recently there are researches to calculate matching degree based on similarity counting, which calculate the matching degree as the weighted sum of the similarity degree of the attributes in the retrieval requirement and service advertisement [2][3][5]. The latter kind of methods, which is adopted in the paper, has higher efficiency and recall and precision and can let the user better know how the retrieved services fit his need. The reprehensive studies fallen into this category include the augment UDDI Registry of Carnegie Mellon University [6], the Meteor-s project University of Georgia [5], the OWL-S ontology service standard of W3C [4], WSMO working group's WSMO ontology service standard [8].

There are different kinds of ontology service models, in the paper we adopt the following model [2]:

$$WS=<CP, SP, Is/Os, QoS>$$

Proceedings of The Sixth IEEE International Conference on Computer and Information Technology (CIT'06)
0-7695-2687-X/06 $20.00 © 2006 IEEE

Where CP are the common service attributes, in other words, the attributes all services should describe, such as service name, application domain, the key words of the function of service, the category of service. SP are special service attributes, which refer to concrete service's peculiar attributes and often domain dependant. Is/Os attributes are the inputs and outputs respectively. QoS is the quality attributes of services, such as availability, security, performance, price, and reliability.

Three problems that existing service retrieval approaches still do not deal well with are analyzed as following, which are very important if the expected superiority of advanced but complex ontology service models is to be encased.

The first problem is how the truth of the service advertisements can be assured [1][17]. The imprecision of service advertisements may come from three sources: the complexity of ontology service models make it difficult for the provider of services to describe the services accurately; the provider may even do not describe services correctly in purpose; it also can be that an advertisement of a service is correct in initial, but with time and the change of situation, some parts of the service advertisement may become wrong.

The second problem is how incomplete retrieval requirement can be dealt with. The complexity of ontology service models makes it difficult for the service user to understand and use. A service retrieval method may in theory have high recall, precision and efficiency, but if it cost user too much to give an exact and precise retrieval requirement, user may not take the efforts finally to give high quality retrieval requirement the service retrieval method expected, which can heavily influence the recall and precision of the retrieval result. This means that give high quality retrieval requirement is a difficult task for users and incomplete retrieval requirements are normal things.

The third problem is how to determine the matching degree of retrieval requirement and service advertisements, which can be classified as based on logic reasoning or similarity computing for methods based on ontology model in literature but both still involve much efforts of users to judge the fitness of the service in the result set. The former kind of approaches only roughly divides retrieval result into five kinds. Through determining the matching of service advertisement and retrieval requirement based on similarity calculating gives more precise matching degree than based on logic reasoning that only give five kinds of matching result, there are still some limitations: because different methods usually obtain obviously different similar degrees of the service advertisements and retrieval requirement and the difference between the corresponding QoS attribute values of them has different function relation with the total similarity of them, which need preprocessing and it is a difficult task. The weights are also difficult to give, [10] introduces an approach to learn weights for quality attributes. There still lack good methods to give weights for all attributes. And in consideration of different needs of different users, the weights of these attributes should be different, giving weights of attributes become more difficult.

The paper introduces a service retrieval model with Dominance-Based Rough Sets [18], which is enhanced, to solve the above three problems. The paper is organized as following: The second part of the paper introduces and analyzes the service retrieval model system with enhanced Dominance-Based Rough Sets, the third part verifies the introduced model through experiments, and the last part gives the conclusion and feature work of the paper.130

2. A service retrieval model with enhanced Dominance-Based Rough Sets

In the part, we will first give the fundamental knowledge of Dominance-Based Rough Sets[18], then enhance it and introduce a service retrieval model with enhanced Dominance-Based Rough Sets, and 2.3 analyzes the introduced model and show how it can deal with the problems of service retrieval pointing out above and improves the quality of service retrievals.

2.1. Introduction to Dominance-Based Rough Sets

Rough Sets was developed by Zdzislaw Pawlak [13] in the early 1980's and has been applied successfully to many domains. To deal with multicriteria classification [18] proposes to use a Dominance-Based rough set approach (DRSA). This approach is different from the classic rough set approach (CRSA) because it takes into account preference orders in the domains of attributes and in the set of decision classes. Given a set of objects partitioned into pre-defined and preference-ordered classes, the new rough set approach is able to approximate this partition by means of dominance relations instead of indiscernibility relations used in the CRSA. The syntax of these rules is adapted to represent preference orders. The DRSA keeps the best properties of the CRSA and does not need any prior discretization of continuous-valued attributes.

Formally, a data table is the four-tuple $S = <U, Q, V, f>$, where U is a finite set of objects (universe), $Q = \{q_1, q_2, . . . , q_m\}$ is a finite set of attributes, V_q is the domain of the attribute q, $V = \cup_{q\in Q} V_q$, and $f : U \times Q \rightarrow V$ is a total function such that $f (x, q) \in V_q$ for each $q \in Q$, $x \in U$, called the information function.

If the objects in the data table are classification examples, then the set of attributes is divided into condition *attributes* and a *decision attribute*. In multicriteria classification, condition attributes are *criteria*. Furthermore, decision attribute *d* makes a partition of *U* into a finite number of classes ***Cl*** = $\{Cl_t, t \in T\}$ and *T* = {1, . . . , *n*}. The classes from ***Cl*** are preference-ordered according to increasing order of class indices.

In multicriteria classification, due to the preference order in the set of classes ***Cl***, the sets to be approximated are not the particular classes but *upward unions* and *downward unions* of the classes, respectively:

$$\mathrm{Cl}_t^{\geq} = \bigcup_{s \geq t} Cl_s, \qquad \mathrm{Cl}_t^{\leq} = \bigcup_{s \leq t} Cl_s, \quad t = 1, \ldots, n.$$

Let $\geq_q$ be a *weak preference relation* on *U* representing a preference on the set of objects with respect to criterion *q*; $x \geq_q y$ means "x is at least as good as *y* with respect to criterion *q*." We say that *x dominates y* with respect to $P \subseteq C$ (or, *x P-dominates y*), denoted by xD_Py, if $x \geq_q y$ for all $q \in P$.

Given $P \subseteq C$ and $x \in U$, the "granules of knowledge" used in DRSA for approximation of the unions $\mathrm{Cl}_t^{\geq}$ and $\mathrm{Cl}_t^{\leq}$ are:

- A set of objects dominating *x*, called *P-dominating set,* $D_P^+(x) = \{y \in U: yD_Px\}$,
- A set of objects dominated by *x*, called *P-dominated set,* $D_P^-(x) = \{y \in U: xD_Py\}$.

For $P \subseteq C$, the set of all objects belonging to $\mathrm{Cl}_t^{\geq}$ without any ambiguity constitutes the *P-lower approximation* of $\mathrm{Cl}_t^{\geq}$, denoted by $\underline{P}(\mathrm{Cl}_t^{\geq})$, and the set of all objects that could belong to $\mathrm{Cl}_t^{\geq}$ constitutes the *P-upper approximation* of $\mathrm{Cl}_t^{\geq}$, denoted by $\overline{P}(\mathrm{Cl}_t^{\geq})$:

$\underline{P}(\mathrm{Cl}_t^{\geq}) = \{x \in U: D_P^+ \subseteq \mathrm{Cl}_t^{\geq}\}$,

$\overline{P}(\mathrm{Cl}_t^{\geq}) = \{x \in U: D_P^- \cap \mathrm{Cl}_t^{\geq} \neq \phi\}$, for $t = 1, \ldots, n$.

All the objects belonging to $\mathrm{Cl}_t^{\geq}$ and $\mathrm{Cl}_t^{\leq}$ with some ambiguity constitute the *P-boundary* of $\mathrm{Cl}_t^{\geq}$ and $\mathrm{Cl}_t^{\leq}$, denoted by $Bn_P(\mathrm{Cl}_t^{\geq})$ and $Bn_P(\mathrm{Cl}_t^{\leq})$, respectively. They can be represented in terms of upper and lower approximations as follows:

$Bn_P(\mathrm{Cl}_t^{\geq}) = \overline{P}(\mathrm{Cl}_t^{\geq}) - \underline{P}(\mathrm{Cl}_t^{\geq})$,

$Bn_P(\mathrm{Cl}_t^{\leq}) = \overline{P}(\mathrm{Cl}_t^{\leq}) - \underline{P}(\mathrm{Cl}_t^{\leq})$, for $t = 1, \ldots, n$.

When defining non-ambiguous objects, it is reasonable to accept a limited proportion of negative examples, particularly for large data tables. Such an extended version of DRSA is called a variable-consistency DRSA model (VC-DRSA)[19].

For a given upward or downward union of classes, $\mathrm{Cl}_t^{\geq}$ or $\mathrm{Cl}_t^{\leq}$, the decision rules induced under a hypothesis that objects belonging to $P(\mathrm{Cl}_t^{\geq})$ or $P(\mathrm{Cl}_t^{\leq})$ are *positive,* and all the others *negative*. On the other hand, the decision rules induced under a hypothesis that objects belonging to the intersection $P(Cl \leq s) \cap P(Cl \geq t)$ are *positive,* and all the others *negative*.

When applying $D_{\geq}$ -decision rules to object *x*, if x matches the LHS of rules $\rho_1, \rho_2, \ldots, \rho_m$, having the RHS $x \in \mathrm{Cl}_{t1}^{\geq}$ 1, $x \in \mathrm{Cl}_{t2}^{\geq}$, . . . , $x \in \mathrm{Cl}_{tm}^{\geq}$, then x is assigned to class Cl_t, where t=max{t1, t2, . . . , tm}. When applying $D_{\geq\leq}$ -decision rules to object x, it is concluded that x belongs to the lowest class of the union of all classes suggested in the RHS of rules covering x.

Duo to limited space, the rule induction algorithm is not given here, interesting reader can reference to [18]. How to deal with incomplete data table can refer to [16].

2.2 A service retrieval model based on Rough Sets

To utilize DRSA, a retrieval requirement can be seen as an object, each CP, SP and QoS can be seen as an attribute, Is and Os are seen an attribute respectively, in service retrieval, users will can give a minimum requirement for each QoS attribute [17]. In [18], all the domains of attributes have preference orders. But in dealing with service retrieval, domains of CP, SP, Is and Os do not have orders, and domains of QoS have "requirement orders", In other words, the criterions in DRSA measure goodness or badness form come aspect of an object, while the criterions in service retrieval problems measure the requirements of users. So we modify the fundamental concept $\geq_q$ to represent a comparison of the set of requirements with respect to criterion *q*; $x \geq_q y$ means "x is less than *y* with respect to criterion *q*." We say that *x dominates y* with respect to $P \subseteq C$ (or, *x P-dominates y*), denoted by xD_py, if $x \geq_q y$ for all criterion $q \in$ P and $x \neq_q y$ for all other $q \in$ P. Other concepts and the rule acquisition algorithm is the same as [18], which is easy to prove and not discussed here.

As in figure 1, the retrieval requirement of a user is sent to both the retrieval system and the learning system and the ID of the corresponding services selected by the user and the feedback of the performance of the selected services are also sent to the learning system. The feedback is the evaluation of the performance of the services by users, which can be graded by a range of integer [11][12][17]. The learning system analyzes an amount of retrieval requirements and feedback for a service and obtains decision rules of the form $C \Rightarrow D$, where C is the attribute-values appeared in the user's retrieval requirements and D is the grade of the performance of the service selected correspondingly by the user. The learning system delivers the obtained decision rules to the retrieval system. In later retrieval, the fitness of a service can be judged based on corresponding decision rules. The learning system will periodically replace some old data and learning decision rules again.

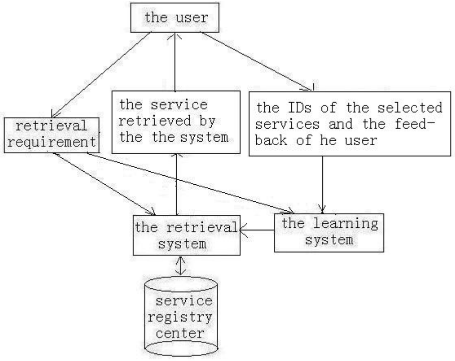
the user
retrieval requirement
the service retrieved by the the system
the IDs of the selected services and the feed-back of he user
the retrieval system
the learning system
service registry center

Figure 1. **a service retrieval model based on Rough Sets**

For efficiency, we will retrieve service based on similarity calculating first with a low similarity threshold, then for services in the intermediate result set, their fitness is judged by the correspondingly decision rules learned. If there are no decision rules are applicable, weather or not a service is kept in the result set is judged by a high similarity threshold. In the result return to users, services whose fitness can be determined by decision rules precede the service that cannot. If two services are classified as the same class by the decision rules, which one has higher similarity with the retrieval requirement based on similarity counting is the better one. Following is the core algorithm of the model.

```
Algorithm: the retrieval algorithm of the introduced service retrieval model Based On Rough Sets
Input: decision rule sets R for services, retrieval requirement Q, similarity threshold $s_1$, $s_2$ ($s_1<s_2$), performance grade threshold g
Output: the sorted result service set RQ
(1) Find intermediate retrieval set P based on similarity calculating with similarity threshold $s_1$ (If a attribute in the retrieval requirement is empty, then the corresponding attribute in the service advertisement and the attribute in the retrieval requirement are seen as the same);
(2) while (P•$\phi$){
(3)     judge the fitness of the services in P based on their corresponding decision rules and put the services with performance grade equal or greater than g into RQ and delete them from P or delete them form P directly;
(4) }
(5) for services remained in P, put them into RQ if their similarity with the retrieval requirement is larger than $s_2$
(6) sort services in RQ, services whose fitness can be determined by decision rules precede the service that cannot. If two services are classified as the same class by the decision rules, which one has higher similarity with the retrieval requirement based on similarity counting is the better one.
(7) return RQ to the user.
```

Figure 2. **The introduced service retrieval model based on enhanced DRSA**

2.3 Analysis of the algorithm

In the part, first why the introduced service retrieval model can deal with the problems of service retrieval mentioned in the first part is analyzed. Then its goodness to the quality of service retrieval discussed.

Reuse of the knowledge of the application of service is a good solution to the problem of the truth of the service advertisement. The introduced algorithm judges the applicability of the services based on the decision rules synthesized form history retrieval requirements and the corresponding feedback of the performance of the selected services, which can reflect the actual capability and performance of the services and avoid intentionally or unintentionally incorrect description of the of service advertisements or the incorrectness of service advertisement caused by the evolution of situation.

Judgment of the fitness of service based on decision rules instead of similarity calculating avoids the need of weights and the preprocessing of attributes and calculating the similarity of attribute values. And because decision rules are synthesized by reduction of attributes and attribute values, the model introduced in the paper can deal better with incomplete retrieval requirement comparing to determine the matching of retrieval requirement and service advertisement based on similarity calculating or logic reasoning.

The goodness to the quality of retrieval result of the model introduced in the paper are analyzed as following:

- Because the introduced algorithm can deal with the problems of possible untruth of the advertisement of service, incomplete retrieval requirement and give a better approach to measure the fitness of services to users' need, so the precision of service retrieval is improved.
- The matching algorithm based on similarity calculating will balance between precision and recall. For precision, the recall of retrieval may be scarified to some degree. With the introduced learning algorithm, the similarity threshold can take a lower value first and the recall is improved.

Service retrieval can be divided as three phrases in a broad sense: the phrase the user gives a retrieval requirement, the phase the system retrieves service based on the retrieval requirement, the phrase the user selects the service needed. Classify the time after the system retrieves service based on similarity calculating first with a low similarity into the third phrase. Because the precision of the retrieval result is improved, it is easy for the user to select the services needed and the time of the third phase is reduced and the times of first two phases remain unchanged and the efficiency of service retrieval is improved, which also means the service retrieval system is easier for user to use.

3. Experiment

We have implemented a prototype service retrieval system X-COM and verify the algorithm introduced in the paper.

We select twenty-two services such as mobile phone message services, data mining services and investigate the recall, precision and efficiency of JAXR Registry based on UDDI [14], the augment UDDI Registry that integrates DAML-S [9], calculating the match of retrieval requirement and service advertisements based on the similarity of ontology and words-Method I in [2] and the introduced service retrieval mechanism in the paper (in calculating matching degree, Method I in [2] is adopt) respectively. All systems run on windows 2003 and Eclipse SDK 3.1. (1) Register services described by WSDL according to NAICS standard to JAXR Registry, the retrieval is based on key words and according to NAICS standard; (2) Register services described by DAML-S to augment UDDI Registry, the retrieval is based on logic reasoning; (3) register services described by the ontology model given in the paper to X-COM, the retrieval is carried out based on the method I in [2] and result contains service the advertisement of which have a similar degree greater than a threshold, (4) register services described by the ontology model given in the paper to X-COM, the retrieval is carried out based on the introduced service retrieval mechanism in the paper.

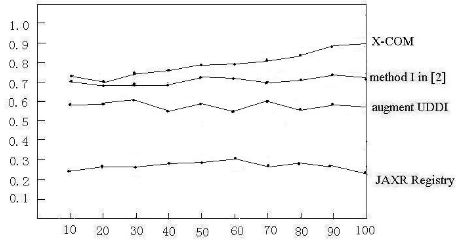

Figure 3. The comparison of the recall

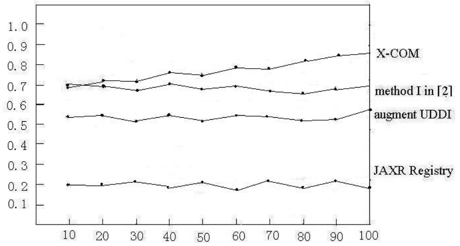

Figure 4. The comparison of the precision

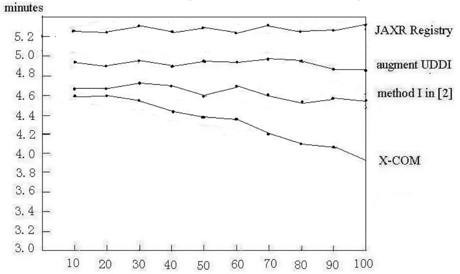

Figure 5. The comparison of the efficiency

For every system, two hundred service retrievals are carried out. The recall, precision and efficiency are calculated for every twenty times. The average recall, precision and efficiency, which is defined as the time sum of the three phrase, are around 0.27, 0.21 and five minutes and eighteen seconds respectively for JAXR Registry and around 0.59, 0.53 and four minutes and fifty-seven seconds respectively for augment UDDI Registry and around 0.72, 0.67 and four minutes and thirty-five seconds respectively for method I in [2]. For the introduced service retrieval mechanism in the paper, the recall, precision and efficiency of service retrieval increase as the figure 3, 4, 5, respectively.

4. Conclusion and feature work

Service retrieval receives much attention from industry and academia as an important part of service-oriented software engineering. The paper analyzes existing methods based on advanced but complex ontology model and points out three problems that existing approaches still do not deal well with, which are very important if the expected superiority to be encased: how the truth of the advertisement of service can be ensured; how incomplete retrieval requirement can be dealt with, which is very common because of the complexity of the service description models; how a service suit a user' need, existing methods based on ontology service model, which can be roughly classified as based on logic reasoning or similarity computing, still involve much effort of users to judge the fitness of the service in the result set.

The paper introduces a model of learning service retrieval system based on Rough Sets to deal with the three problems. Experiments demonstrates the effectivity of the introduced model based on Rough Sets The feature work is to optimize the efficiency of the algorithm and verify and improve the algorithm.

Acknowledgements

This paper is supported by the National Natural Science Foundation of China under Grant No. 60473062, the National High Technology Development 863 Program of China under Grant No. 2004AA113030, Science Technology Committee of Shanghai under Grant No. 04DZ15022.

5. References

[1]Y. Liu, A. Ngu, and L. Zheng. QoS computation and policing in dynamic web service selection. In *WWW Alt. 04: Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, New York, NY, USA, ACM Press, pp. 66–73, 2004.

[2]Wu J, Wu CH, Li Y, Deng SG. Web Services discovery based on ontology and similarity of words. Chinese Journal of Computer, Vol.28, No.4, pp.595-602, 2005, 4.

[3]Hu JQ, Zhou P, Wang HM, Zhou B. Research on Web Service Description Language QWSDL and Service Matching Model. Chinese Journal of Computer, Vol.28, No.4, pp.505-513, 2005, 4.

[4]W3C. OWL-S: Semantic Markup for Web Services. http://www.w3.org/Submission/OWL-S/, 2004.11

[5]Large Scale Distributed Information Systems Lab of the Georgia University, METEOR-S: Semantic Web Services and Processes, http://lsdis.cs.uga.edu/Projects/METEOR-S/ .

[6]Massimo Paolucci, Takahiro Kawamura, Terry R. Payne, Katia Sycara. Importing the Semantic Web in UDDI. In Proceedings of Web Services, E-business and Semantic Web Workshop, Toronto, Canada, pp.225~236, 2002.

[7]Dogac A, Kabak Y, Laleci GB. Enriching ebXML registries with OWL ontologies for efficient service discovery. Proceedings of the 14th International Workshop on Research Issues on Data Engineering (RIDE'04), pp.69–76, 2004.

[8]WSMO working group. D2v1.2. Web Service Modeling Ontology (WSMO). www.wsmo.org/TR/d2/v1.2/D2v1-_20050414.pdf . 2005,4.

[9]Ali Mili, Rym Mili, and R. T. Mittermeir. A survey of software storage and retrieval. Annals of Software Engenieering,Baltzer Science Publishers. 1998, 5.

[10]Yang WJ, Li JZ, Wang KH. A domain adaptive web service evalution model. Chinese Journal of Computer, Vol.28, No.4, pp.595-602, 2005, 4.

[11]Yang S, Shi ML. A web service retrieval model supporting the constraint of QoS. Chinese Journal of Computer, Vol.28, No.4, pp.595-602, 2005, 4.

[12]Zhang Z, Zhang C. An improvement to matchmaking algorithms for middle agents. In: Gini M, Ishida T and Lewis J W (editors), Proceedings of the First International Joint Conference on Autonomous Agents and Multi agent Systems, New York, USA, ACM Press, pp.1340-1347, 2002.

[13]Pawlak Z. Rough Sets: Theoretical Aspects of Reasoning about Data. Boston: Kluwer Academic Publishers, 1991.

[14]ZIARKO W. The Discovery , Analysis and Representation of Data Dependencies in Databases[A] . Piatetsky - Shapiro G, Frawley WJ eds. Knowledge Discovery in Databases[C] , AAA/MIT Press , 1990. 213 ~ 228.

[15]Liu H., Farhad, Chew Lim Tan, Manoranjan Dash. Discretization: An Enabling Technique. Data Mining and Knowledge Discovery, Kluwer Academic Publishers. Manufactured in The Netherlands. pp.393–423, 2002, 6.

[16]Shao Mingwen, Zhang Wenxiu. Dominance relation and rules in an incomplete ordered information system. International Journal of Intelligent Systems, pp.13~27, 20(1), 2005.

[17]Le-Hung Vu, Manfred Hauswirth, Karl Aberer: Towards P2P-Based Semantic Web Service Discovery with QoS Support. Business Process Management Workshops pp.18-31, 2005.

[18]S. Greco, B. Matarazzo and R. Slowinski, Rough approximation by dominance relations, International Journal of Intelligent Systems, pp.153—171, 17, 2002.

[19]Greco S, Matarazzo B, Slowinski R, Stefanowski J. Variable consistency model of dominance-based rough set approach. In: Ziarko W, Yao Y, editors. Rough Sets and Current Trends in Computing. Second International Conference, RSCTC 2000 Banff, Canada, October 16 – 19, 2000. LNAI Vol. 2005, Berlin, Springer-Verlag, pp 170–181, 2001.