



University of HUDDERSFIELD

University of Huddersfield Repository

Meng, Zhaozong and Lu, Joan

A Rule-based Service Customization Strategy for Smart Home Context-aware Automation

Original Citation

Meng, Zhaozong and Lu, Joan (2016) A Rule-based Service Customization Strategy for Smart Home Context-aware Automation. *IEEE Transactions on Mobile Computing*, 15 (3). pp. 558-571. ISSN 1536-1233

This version is available at <http://eprints.hud.ac.uk/25913/>

The University Repository is a digital collection of the research output of the University, available on Open Access. Copyright and Moral Rights for the items on this site are retained by the individual author and/or other copyright owners. Users may access full items free of charge; copies of full text items generally can be reproduced, displayed or performed and given to third parties in any format or medium for personal research or study, educational or not-for-profit purposes without prior permission or charge, provided:

- The authors, title and full bibliographic details is credited in any copy;
- A hyperlink and/or URL is included for the original metadata page; and
- The content is not changed in any way.

For more information, including our policy and submission procedure, please contact the Repository Team at: E.mailbox@hud.ac.uk.

<http://eprints.hud.ac.uk/>

A Rule-based Service Customization Strategy for Smart Home Context-aware Automation

Z. Meng, and J. Lu

Abstract—The continuous technical progress of the smartphone built-in modules and embedded sensing techniques has created chances for context-aware automation and decision support in home environments. Studies in this area mainly focus on feasibility demonstrations of the emerging techniques and system architecture design that are applicable to the different use cases. It lacks service customization strategies tailoring the computing service to proactively satisfy users' expectations. This investigation aims to chart the challenges to take advantage of the dynamic varying context information, and provide solutions to customize the computing service to the contextual situations. This work presents a rule-based service customization strategy which employs a semantic distance-based rule matching method for context-aware service decision making and a Rough Set Theory-based rule generation method to supervise the service customization. The simulation study reveals the trend of the algorithms in time complexity with the number of rules and context items. A prototype smart home system is implemented based on smartphones and commercially available low-cost sensors and embedded electronics. Results demonstrate the feasibility of the proposed strategy in handling the heterogeneous context for decision making and dealing with history context to discover the underlying rules. It shows great potential in employing the proposed strategy for context-aware automation and decision support in smart home applications.

Index Terms—Smart home, context-aware automation, decision support, service customization, rule generation



1 INTRODUCTION

THE flexible built-in sensors of mobile devices and embedded sensing techniques have fostered the context-aware computing paradigm, which involves the sensor data as implicit input to customize the computing service to specific contextual situations. The smart environment systems such as homes, workplaces, hospitals, and vehicles are typical use cases which integrate sensors and actuators to assist occupants to interact with the physical environment more efficiently. It is widely recognized that the context-aware computing concept based on various mobile device built-in modules and embedded sensing techniques can be considered a promising solution to better satisfy users' expectations and facilitate people's daily lives [1]. The Wireless Sensor Actuator Network (WSAN) [2,3] becomes an active research area in which sensors and actuators can be used to enhance the interaction between human and the physical world. It is estimated that there will be over 50 billion devices connected to the Internet by 2020, and paradigm shift is now being promoted, in which every object becomes interactive [4,5]. In addition to the embedded sensing techniques, the smartphone becomes more and more powerful in computation and communication. The shipment of various built-in sensors, the wide acceptance amongst users, and its popularity in daily use make the smartphone an ideal platform appropriate for characterizing users' preferences and the ambient environment [6].

The wide penetration of the novel sensing techniques

and users' greedy expectations may require efficient user interaction without imposing undue technological complexity, effort, or inconvenience [7]. In addition, the cost reduction of the embedded electronics provides solutions to utilize smartphones and low-cost commercially available electronics for context-aware automation and decision support, to assist people's daily lives by reducing their supervision of home facility control and management. The prosperity of supporting techniques and lack of context-aware service customization strategy have motivated the work of this investigation to explore approaches that take advantage of the computational context to effectively customize the computing service to the contextual situation. In this investigation, a context-aware service customization strategy for smart home applications is proposed with a proof of concept implementation. This strategy not only makes use of the context for context-aware service composition, but also takes advantage of history context for high-level supervision to better characterize the users by enabling the system to learn through its observation.

The proposed context-aware service customization strategy regards context-aware system as Decision Information System (DIS), where context and service are the condition and decision attributes respectively. It consists of a semantic distance-based rule matching method and a Rough Set Theory (RST)-based rule generation method. The rule matching method is for context-aware computing service composition with the context according to the rules in rule repository, and the rule generation method is used to derive new rules to better supervise the service customization. A prototype system with smartphone and low-cost commercially available electronics is designed and implemented to demonstrate the feasibility of the

- Z. Meng is with the Department of Informatics, University of Huddersfield, Huddersfield, UK, HD1 3DH E-mail: zhaozong.meng2@hud.ac.uk, zmeng83@gmail.com
- J. Lu is with the Department of Informatics, University of Huddersfield, Huddersfield, UK, HD1 3DH E-mail: j.lu@hud.ac.uk

proposed methods. The results reveal the low time complexity in rule matching and rule generation. It shows great potential to employ the proposed strategy to deal with the context information for service customization.

The rest of this paper is organized as follows: section 2 reports the peer studies on context-aware smart home systems and context-aware service customization methods; section 3 presents the proposed approaches for context-aware service customization; section 4 illustrates the design and implementation of a prototype system; and section 5 provides the evaluation of the methods with the prototype system. Then, conclusion is drawn and future work is suggested in section 6.

2 BACKGROUND

2.1 Related Work

Context-aware service in smart home can be explained with a simple example: "Suppose Dave goes home after work and sits in the living room, the TV is turned on automatically playing his favorite TV program at that time. The ambient light and temperature are adjusted by controlling the home facilities according to his preference. His smartphone reminds him his traveling plans, recommends a route, and notify fuel quantify. When time reaches 11:00 pm, Dave is suggested to prepare to go to bed, and the home facilities are turned off when Dave leaves." The unobtrusively executed functions such as TV program playing, home facility control, and traveling reminding are considered as computing service. The decision making is based on timely observation of the varying environment, and a reduced description set of context and actions including human behavior, user preference, and physical environment which are regarded as rules.

Dey and Abowd [8] defined context-aware as: "a system is context-aware if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task". The context-aware computing paradigm is not a new emerging research topic, while its integration with mobile devices and novel sensing techniques to enhance user interaction and facilitate people's activities has recently been recognized as a separate area of research. Investigations in this area mainly focus on five topics: unobtrusive observation of context, efficient and secure infrastructure, representation and semantic understanding, decision making and knowledge discovery, and domain specific applications.

The early stage investigations on smart home mainly focus on context acquisition and distribution with the novel sensing techniques and wireless network solutions for proof of concept development, such as indoor environment monitoring [9], wireless smart home sensor network [10], WSN topology control for network coverage and connectivity [11], WSN for production automation [12], interoperable device usage [13, 14], graphic user interface for remote control [15], network protocols and standards [16], trust and security [17], and service oriented smart home architecture [18]. With the advance of relevant techniques, more and more research work transfers to high-level smart home applications and the models and methods considering how to facilitate occupants'

daily lives, such as RADL (Recognizing Activities of Daily Living) [19], elderly care [20], medical care [21,22], real-time energy consumption load forecasting [23], and sustainable homes [7]. These investigations may address more technical issues around how to implement the techniques into the home environments to assist people's daily lives. More and more studies have moved to high level knowledge processing, mathematical models, and human involvement. Semantic space, a pervasive computing infrastructure employing the semantic Web technologies for knowledge management and information processing is proposed in [24]. Semantic smart home, a conceptual architecture for semantic smart homes focusing on the methodology of semantic modeling, content generation and management is discussed in [25]. In addition, a semantic space model employing semantic Web technology to facilitate the integration of hardware and middleware elements in the scope of ubiquitous computing is introduced in [26]. For these semantic based systems, context modeling, semantic representation, query, and reasoning through Resource Description Framework (RDF) scheme and Web Ontology Language (OWL) have become popular research topics.

Although there have already been heuristic studies on context handling and intelligence service provisioning in related areas, there are few systematic service customization strategies presented for smart environment use cases. The focus of this investigation is the service customization strategy which predicts users' expectation and provides suitable service, namely the method for decision making and method enabling the system to learn through its observation. Since the real-world evaluation is a major concern and big challenge facing the research in this area due to the varying user behaviors, a complex, real, and natural environment is significant to this investigation. To deal with this problem, the "Live-in Laboratory" comes into being for the study of people and technologies in home environment [27].

The traditional decision making solutions for service adaptation are based on IF-THEN logic [28]. Since the IF-THEN based rules are independent of each other, the context change must be specified for each action of the application. With the advent of rich sensing devices, the increase in quantity of context and the complexity of its relationship require more powerful approaches to deal with the sophisticated situations. For the context-aware computing systems, some mathematical models for context-aware service decision making are introduced, such as Fuzzy Multi-Attribute Decision Making (FMADM), Multi-facet Item based method, Multi-Attribute Utility Theory (MAUT), probability-based model, Adaptive Neuro-Fuzzy Inference System (ANFIS), Fuzzy Logic, and Rough-Fuzzy method. This section will give an in-depth discussion on these theories and methods in dealing with context-aware services and applications.

(1) *FMADM* - In order to allow the computing system to make appropriate decisions on behalf of users according to dynamic user context, TalebiFard and Leung introduced a FMADM method and a context similarity measurement method [29]. This method can be used as a gen-

eralized approach to collect the context of mobile devices, and compare the context with the advertised service based on feature similarity. The fuzzy set is appropriate for presenting the uncertainty of context data, and the similarity measurement provides a way to evaluate the context and the service conditions. The capability for handling the context data with complicated structure needs further verification.

(2) *MAUT* - The MAUT is widely used for product evaluation in consumer organizations. It is also an applicable method for context-aware service customization. Schäfer employed the MAUT to estimate the user's interests for information regarding product recommendation [30]. For context-aware computing systems, the context-aware service can be regarded as the evaluation result, the relevant context information can be considered to be the attributes, and the weight is the impact of context to the decision of service composition. This method is simple and the logic in computation is explicit. However, it is difficult to determine the weight of the attributes of the context items. Moreover, the system is not strong in presenting the complicated relationship of the service and context items for some use scenarios.

(3) *Probability-based model* - Wang et al. present a probability-based model for music recommendation based on context information and music content analysis by exploring the rich sensing capability of mobile devices [31]. By monitoring the user's operations, user preference is updated in real-time to adapt to a particular user. This probability-based method is lightweight in computation and easy to implement, and it can adapt to new uses immedi-

ately without training procedures. It is appropriate to fulfill context-aware systems without many context items taking effect.

(4) *ANFIS* - The ANFIS is introduced to adapted learning content distribution in mobile-learning systems by Al-Hmouz et al. [32]. It aims to adapt the learning content to learners' needs within different learning context scenarios. ANFIS uses Fuzzy Logic to transform given inputs into a desired output through highly interconnected Neural Network processing elements and weighted information connections. ANFIS integrates the Fuzzy Logic and Neural Network, which can tune the parameter of Fuzzy Inference System (FIS) with Neural Network learning. It refines the IF-THEN rule for complex context-aware applications. It eases the implementation and enables fast and accurate learning as well. This method is also strong in dealing with the incompleteness of human experts made rules by fuzzy rule training.

(5) *Rough-Fuzzy method* - Duan et al. propose a rough-fuzzy hybridization for preference-based Web information retrieval [33]. In this rough-fuzzy method, fuzzy sets are used to handle real-valued weight in the document, and the rough-fuzzy method named Variable Precision Rough Set Model (VPRSM) is used to discover user preference. This rough-fuzzy method integrates the rough set and fuzzy set, which are appropriate to deal with important tasks of personalized Web information retrieval: weight evaluation, and ambiguities of language and preference discovery. However, the particular training process for each user is required, where automation may be expected in practical use.

TABLE 1
MATHEMATICAL MODELS FOR CONTEXT-AWARE SERVICE CUSTOMIZATION

Mathematical model	Purpose or use case	Context used	Evaluation method	Strength and weakness
FMADM	General method for context-aware service delivery	Network attributes: bandwidth, bandwidth variation, availability, stability, and error rate	Through case study on improving Quality of Experience (QoE) for DSL services, such as VOIP and IPTV	General for context-aware systems Quality of Service (QoS) is taken into account Not strong in dealing with complicated context model
MAUT	Estimation of user's interest for product recommendation	Attributes of the product	Brief case study without evaluation	Simple and easy to implement Difficult to determine weight of attributes Weak in dealing with complicated context model
Probability-based model	Mobile phone music recommendation for daily activities	User activity and music content	Model accuracy and system usability are evaluated by experiments on Android devices	Solves the cold-start problem by combining music content analysis and activity inference The adaptation can be done directly on mobile phone without use of backend server
ANFIS	Adaptive learning content distribution for mobile learning	Learning materials, personal information, location, time, mobile device, environment, network, bandwidth	Evaluated with standard error measurement, which reveals the optimal setting necessary for better predictability	Integrates fuzzy inference with neural network Strong in dealing with the incompleteness of rules with fuzzy rule training Complex in structure
Rough-Fuzzy method	Personalised Web information retrieval	User preference and keywords of documents	Evaluated by comparing the results of proposed approach with results of Google search	Integrates rough sets and fuzzy sets Fuzzified weight and rough similarity are suitable for context based decision making A training process is required.
Fuzzy computing	System, method, and learning strategy for Aml service	Temperature, humidity, airflow, clothing, luminosity level, user preference, weather, seasonal change, etc.	Test experiment to: (1) comparing learned rules with users' archive entries, (2) evaluating the number of rules learned over time.	Fuzzy theory is proper for handling vagueness and imprecision feature of context Learning strategy is employed to customize the fuzzy service

(6) *Fuzzy computing* – An Ambient Intelligence (AmI) fuzzy computing system based on multi-agent and fuzzy theory is proposed to pursue autonomous intelligence satisfying the needs of inhabitants without human intervention [34]. The proposed AmI fuzzy computing focuses on the learning strategy capable of capturing the dynamic context feature and user preference to generate intelligent service that controls the environment to satisfy users' requirements. In order to handle the vagueness and imprecision feature of context, the fuzzy theory is employed to present the context with linguistic variables derived from numerical variables. The fuzzy inference method is also used in [35] for context-aware control of appliances in home environment, and a lighting control system is implemented as a case study.

These aforementioned mathematical models are summarized in TABLE 1. They each have their strengths in handling the context data and provide the appropriate context-aware computing service. Methods (1), (2), and (3) are simple in logic and easy to use with lightweight computation. However, they may not be powerful enough to deal with context models that are complex in structure. Methods (4), (5), and (6) employ fuzzy logic to deal with the vagueness and uncertainty of context data, which is better in handling the complexity of context. Study in (6) emphasizes learning capability of the system to anticipate users' requirements and provide intelligent services. For the above mentioned methods, the inference are mostly IF-THEN based, and the history context which may contain underlying information valuable for the context-aware service customization is not effectively used. Therefore, major concerns in the context-aware service customization are identified: (1) the decision making method to deal with the heterogeneous and uncertain context to pursue the intelligent context-aware service; (2) method enabling the system to learn by taking advantage of history context to derive underlying knowledge to supervise the service customization. This investigation provides a context-aware service customization strategy to make further improvements in the above points.

2.2 Aim of this Investigation

The strategy customizing the computing service with the context data is an effective way to pursue user satisfaction and assist inhabitants' daily lives. The purpose of service customization is to integrate the dynamic context into the computation and adapt the computing service to the contextual situations [36].

To deal with the challenges stated in the above section, this investigation aims to provide a context-aware service customization strategy to take advantage of the context information to provide the appropriate computing service. The proposed method emphasizes the capability in dealing with the heterogeneity and uncertainty of context data. It also enables the system to learn by taking advantage of the recorded history context to derive underlying knowledge to supervise the service customization. To evaluate the proposed methods, a proof of concept prototype system is built as an experimental platform. Experimental studies are carried out to demonstrate the feasibility

and evaluate the performance of the methods.

3 METHOD EMPLOYED

3.1 Context-aware Service Customization Strategy

The primary strength of the context-aware system compared with the traditional ones is its capability to automatically gather the context information as implicit input to customize the computing service. Thus, the essential task of the context-aware system is to predict users' expectations and proactively make a decision to provide the appropriate service accordingly.

According to Weiser's viewpoint on context-aware computing [37], the context-aware system provides the context-aware computing service S_C according to the user's explicit request R_E and computing context information C related to the computation. This investigation presents the functional model of context-aware computing system with formula (1).

$$S_C = F(R_C) = F(R_E, C) \quad (1)$$

Moreover, the system can predict the expected service with the context only to provide the proactive computing service automatically as formula (2) presents.

$$S_C = F(R_C) = F(C) \quad (2)$$

where $C = \{Cu, Cd, Cn, Ce, Ch, \dots\}$, and $Cu, Cd, Cn, Ce, Ch, \dots$ are the context information of computing tasks such as user context, device context, network context, environment context, and history context. Thus, with the implicit input C , the computing service S_C is customized to the computational context.

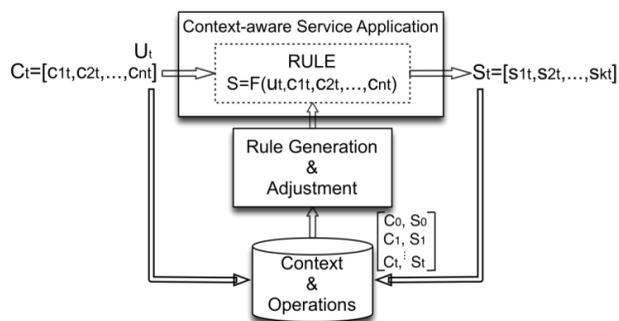


Fig. 1. Functional Model of Context-aware Service Customization

According to this theory, a rule-based context-aware service customization strategy is therefore proposed:

(1) The context-aware computing system is regarded as a decision information system, where the context items are the condition attributes and the computing service parameters are the decision attributes. The principle can be described with the functional model diagram in Fig. 1.

(2) Use the rule matching method for decision making of service customization, and use the rule generation method to derive new rules using the history context. In this investigation, the semantic distance-based rule matching method and the rough set theory-based rule generation method are selected.

As shown in Fig. 1, the computing service at time t - St is determined by the Rule $St=F(Ut,Ct)$, where Ut and Ct are the user operation and context at t . The context data, user operations, and service decisions are saved for rule generation to supervise the service composition. Then, the service customization method and rule generation method are illustrated as follows.

3.1.1 Semantic Distance-based Rule Matching Method

In the context-aware service model, the service customization can be accomplished with semantic distance based rule matching, which compares the current context and the context-aware customization rules. Let C be the attribute vector of context items, U be a sample of context value, and V be the rule vector. The distance between vectors can be calculated with the Manhattan Distance, which is also named Taxicab Geometry, presented in [38]. In this research, Manhattan Distance is normalized to compute the distance of context vectors $dist(U, V)$, which is defined as follows:

$$dist(U, V) = \sum_{i=1}^n w_i \frac{|u_i - v_i|}{Range(c_i)} \quad (3)$$

Formula (3) defines the approximation of context value vectors U and V , where w_i ($0 < w_i < 1$, $\sum w_i = 1$) and $Range(c_i)$ are the weight and value range of the i^{th} attribute c_i . The Manhattan Distance-based rule matching method may be suitable for numeric context values. For non-numeric, the difference between the values can be described with semantic distance, and researchers have already proposed many different methods of measuring the semantic distance [39]. In this investigation, the Generalized Cosine-Similarity Measure (GCSM) by Ganesan et al. in [40] is employed to compute the semantic distance between context and rule sets, which is defined as follows:

Definition 1 Lowest Common Ancestor (LCA) *LCA denotes the common ancestor of the maximum depth of two concepts in a tree-hierarchy.*

Definition 2 Generalized Cosine-Similarity Measure (GCSM) *If c_1 and c_2 are two concepts in a tree-hierarchy of indexing terms and $depth(c_1)$ and $depth(c_2)$ are their depth in the hierarchy, the GCSM similarity between them is:*

$$GCSM(c_1, c_2) = \frac{2 \times depth(LCA(c_1, c_2))}{depth(c_1) + depth(c_2)} \quad (4)$$

For non-numeric data, formula (4) can be used to calculate $GCSM(c_1, c_2)$ ($0 < GCSM(c_1, c_2) < 1$) to present the similarity of concepts c_1 and c_2 . Then, the semantic distance between concepts c_1 and c_2 can be calculated with formula (5), and the distance between context vector and rule vector can be calculated with formula (6).

$$dist(c_1, c_2) = 1 - GCSM(c_1, c_2) \quad (5)$$

$$dist(V, R) = \sum_{i=1}^n w_i \cdot dist(v_i, r_i) \quad (6)$$

where w_i ($0 < w_i < 1$, $\sum w_i = 1$) is the weight of context item c_i , v_i and r_i are the values of context vector V and rule vector R on attribute c_i , and $dist(V, R)$ denotes the distance between vector V and vector R .

In order to be applicable for the rule matching method, the context data describing the computation context needs to be presented in a formal structure. Since real world entities and inter-relationship can be represented with ontological models in tree-hierarchy, their semantic distance can be calculated with the above methods.

3.1.2 RST-based Rule Generation to Supervise the Service Customization

Essentially, the rule generation in the context-aware automation is a Knowledge Discovery in Database (KDD) method, and the rule is the knowledge to supervise control of home facilities. Some mathematical models can be employed, such as Decision Tree, Bayesian Networks, Fuzzy Logic, Support Vector Machine, and K-Nearest Neighbors. In this investigation, the Rough Set Theory proposed by Pawlak [41,42] is employed for context-aware service rule generation due to its strength in handling uncertainty and imperfection of context data [43].

Definition 3 Information System $I=(U, A, V, f)$ is an information system provided U is non-empty finite objects set, A is non-empty finite attributes set, V_a is the value range of attribute a and $V = \bigcup_{a \in A} V_a$ is the union of attribute domains, and $f: U \times A \rightarrow V$ is an information function so that for any $x \in U$ and $a \in A$, $f(x, a) \in V_a$.

Definition 4 Decision Information System (DIS) *Information system $I=(U, A, V, f)$ is a decision information system if I satisfies conditions: $A = C \cup D$ and $C \cap D = \emptyset$, where, C is the condition attribute and D is the decision attribute.*

Definition 5 Indiscernibility Relation *Given $\forall P \subseteq A$, the equivalence relation $IND(P)$ is defined indiscernibility relation, where $IND(P) = \{(x, y) \in U^2 \mid \forall a \in P, a(x) = a(y)\}$. As context may contain instance data rather than numeric only, the $IND(P)$ can be defined as:*

$$IND(P) = \{(cv_i, cv_j) \in CV^2 \mid \forall a \in P, a(cv_i) = a(cv_j) \vee class(a(cv_i)) = class(a(cv_j))\} \quad (7)$$

where $P \subseteq A$, $a(x)$ is the value of attribute a in vector x , and $class(v)$ denotes the class that the v belongs to.

Definition 6 Reduction and Core *Given $P \subseteq A$ in an information system I , the $IND(P)$ divides object set U into k equivalence classes, denoted $U/P = \{X_1, X_2, \dots, X_k\}$. Suppose $Q \subseteq P$ and Q is independent and $IND(Q) = IND(P)$, then Q is a Reduction of P . If $RED(P)$ is all the reduction of P , $CORE(P) = \bigcap RED(P)$ is the Core of P .*

Definition 7 Discernibility Matrix *Given a DIS, $C = \{a_i \mid i = 1, 2, \dots, m\}$ and $D = \{d\}$ are the condition attributes and decision attributes, $U = \{x_1, x_2, \dots, x_n\}$ is the discourse domain, $a_i(x_j)$ is the value of sample x_j on attribute a_i . The discernibility matrix $DM_{i \times j}$ can be defined as:*

$$DM(i, j) = \begin{cases} \{a_k \mid a_k(x_i) \neq a_k(x_j) \vee class(a_k(x_i)) \neq class(a_k(x_j))\}, & d(x_i) \neq d(x_j) \\ 0, & d(x_i) = d(x_j) \end{cases} \quad (8)$$

where $i, j = 1, 2, \dots, n$.

With the RST-based rule generation method, the criti-

cal attributes can be determined to create new rules or revise the existing rules to supervise the customization of the computing service. The potential information about users' preferences about the relevant computing service may be identified and used to supervise the adaptation.

3.2 Algorithm Design

According to the fundamentals in 3.1, the algorithm exploring the context information towards context-aware automation is developed. Since the context data is heterogeneous in data type and structure, the algorithm should be designed with consideration of this concern which is common to the context-aware computing environment. The semantic distance-based rule matching algorithm is designed as follows in Algorithm 1. For the rule generation, the objective is to determine the influential context attributes for decision making of service customization, the discernibility matrix based attribute reduction method is employed to identify the key context attributes. The algorithm for attribute reduction and rule generation is designed as follows in Algorithm 2. With the algorithms designed, the system can then provide the appropriate computing service adapted to the contextual situations.

Algorithm 1. Semantic Distance-based Rule Matching Algorithm

Inputs of the algorithm: *Context* - current context; *Rule* - rule set for context-aware adaptation; *Reduction* - result of attribute reduction; *Weight* - weight of context attributes.

Outputs: *Matching_rule* - the matching rule.

Step1. Initiate minimum distance $min_dist=1$, $i=0$, $j=0$, $dist=0$, and rule ID= min_ruleid ;

Step2. if($i==Rule.length()$) Go to Step5;
else Calculate Rule[i] in Step 3;

Step3. //Calculate each attributes in reduction $i-Reductioni[j]$;
if($Reductioni[j]$ is numeric)
 $dist+=Weight[j]*abs(Context[j],Rule[i].getValue(Reductioni[j].id))/Rule[i].getRange(Reductioni[j].id)$;
else if($Reductioni[j]$ is non-numeric)
 $dist+=Weight[j]*(1-GCSM(Context[j],Rule[i].getValue(Reductioni[j].id)))$;

if($j==Reductioni.length()$) Go to Step4;
else $j++$;

Step4. if($dist<min_dist$) {
 $min_dist=dist$; $min_ruleid=i$; $dist=0$;
Go to Step2;}
else {
 $i++$; Go to Step3; }

Step5. The minimum distance is min_dist and the matching rule $Matching_rule=Rule[min_ruleid]$.

Algorithm 2. Attributes Reduction and Rule Generation Algorithm

Inputs of the algorithm: $CV = (CV_1, CV_2, \dots, CV_n)$ - History context; $A = (a_1, a_2, \dots, a_n)$ - Context attributes set; O - User approved service operations.

Outputs: *Rules* - context-aware rule set; *Core* - the core; *Reduction* - result of reduction.

Step1. Calculate discernibility matrix DM of decision table consists of CV and O, initiate reduction set $Reduction=\Phi$;

Step2. Add the Core (The attributes in the DM element whose cardinal number is 1) to Reduction, and eliminate the attribute items that contain the Core attribute;

Step3. Calculate the attribute frequency of the rest attribute items using the function $g(a_i)=\sum_{j=2}^n(num_{a_i}/j)$ (where num_{a_i} is the number of a_i in DM, j is the number of attributes in the element of DM that contains a_i , and n is the maximum value of j);

Step4. Add the attribute of highest frequency a_i to Reduction, eliminate the attribute sets that contain a_i ;

Step5. If ($DM \neq \Phi$) Go to Step3; else go to Step6;

Step6. Create the Rules according to the Reduction.

3.3 Simulation Study of the Proposed Methods

The complexity of the algorithms may largely influence the performance of the whole system. In order to determine the time complexity of the two proposed methods, the simulation study is conducted.

The simulation platform is MATLAB 2012 on a Lenovo Laptop with Intel Dual-Core T6600 2.2GHz, FSB Speed 800M, L2 Cache 2M, and 4GB DDR3 RAM. The number of rules and number of context items in the rules are set from 1 to 1000 for rule matching, and there are an equal numbers of numeric and non-numeric context items in all the rules. From the practical application point of view, the number of context samples and context items for rule generation are set from 1 to 15 and 1 to 50 respectively. The simulation results are as shown in Fig. 2 to Fig. 5.

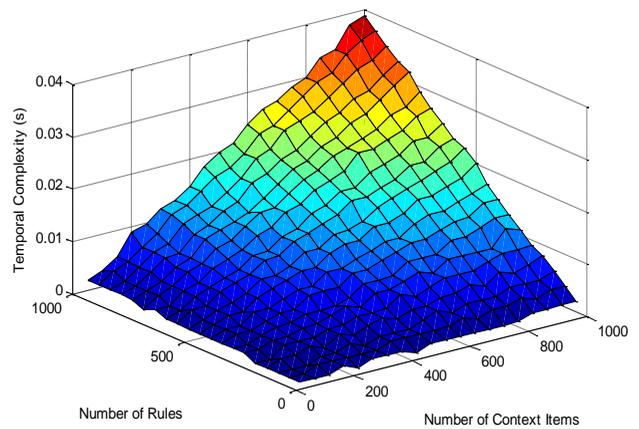


Fig. 2. Time Complexity of the Rule Matching Algorithm

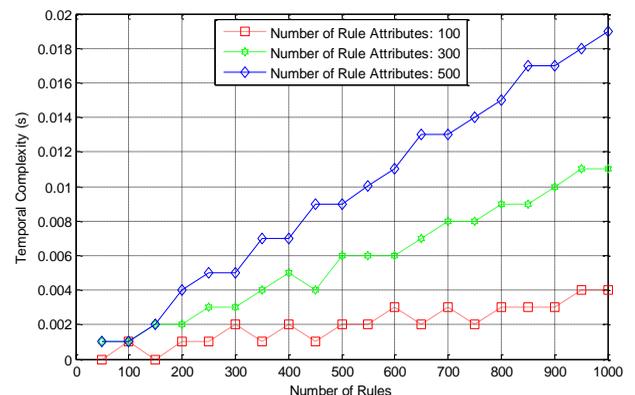


Fig. 3. Speed of the Rule Matching Algorithm with 100, 300, and 500 Attributes

Fig. 2 and Fig. 3 indicate the performance of the rule matching algorithm. From the figures, the running time of the algorithm increases with the number of rules and the context items in the rules roughly in linear. It is evident that the algorithm is not time consuming. Therefore, the real-time performance of the system is not challenging.

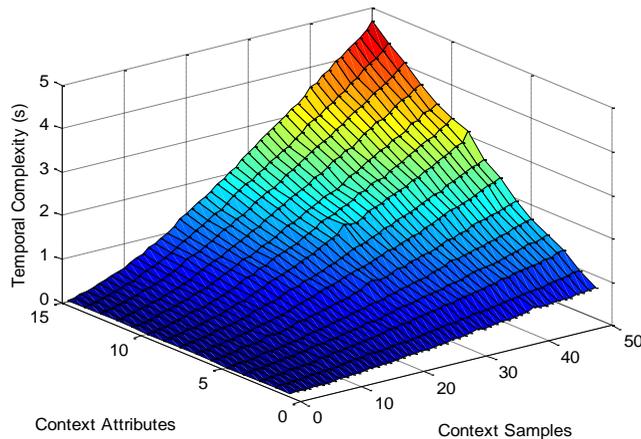


Fig. 4. Time Complexity of the Rule Generation Algorithm

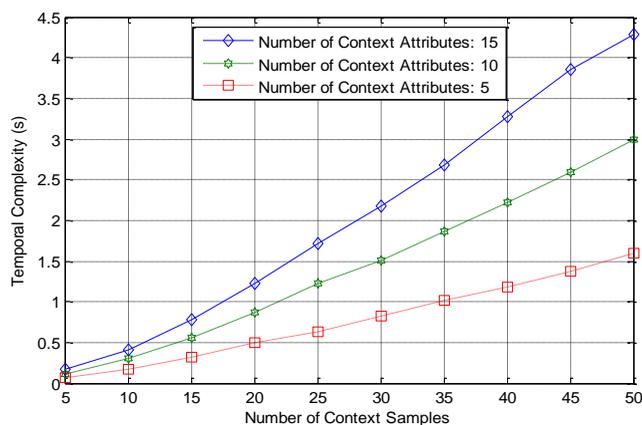


Fig. 5. Speed of the Rule Generation Algorithm with 5, 10, and 15 Attributes

Fig. 4 and Fig. 5 give the trend of how time complexity of rule generation algorithm varies with the number of context samples and condition attributes. From the figures, it is easy to find that the time consumption of the algorithm increases with the quantity of context samples and the quantity of context condition attributes quickly. According to the curve, the speed changes with the number of context samples nearly in linear and it becomes too time consuming for practical use. Therefore, to select the significant context attributes and the appropriate number of context samples is very important for rule generation.

From the discussion in section 2, most of the investigations on context-aware computing systems in the early years are IF-THEN based systems. Some new proposed methods mainly focus on method design and theoretical study. There is a lack of comprehensive experimental study which puts the proposed methods into practical use with evaluations. In this investigation, the above simula-

tion results just give the trend, and the real performance at context-aware service end depends on the computation power of the service provider. Therefore, practical experimental studies are needed to demonstrate the feasibility of the proposed method in both effectiveness and efficiency. In order to prove the effectiveness and evaluate the efficiency, implementation of a prototype system and the experimental studies are presented in Section 4.

4 DESIGN AND IMPLEMENTATION

In order to prove the proposed context-aware service customization methods and algorithms, a prototype smart home system is implemented and evaluations are performed through experimental studies. In this prototype system, a table light and table fan testing case is adopted, because the operations can be immediately observed. For this implementation, only several sensors are need to observe and characterize the environmental context.

4.1 Prototype System Architecture Design

The prototype system can be described using the diagram in Fig. 6. Commercially available low-cost sensors and electronics are employed for the implementation of the system. The Wi-Fi network is employed for the connection between the electronic devices as it is widely available in most home areas. The sensors and actuators are connected with the wireless infrastructure to collect the sensor data and perform the control actions. The user interacts with the system through his/her smartphone. The control service corresponding to the context data is produced at the central controller with the decision making algorithms, and execution of decision commands is performed with relays as actuators.

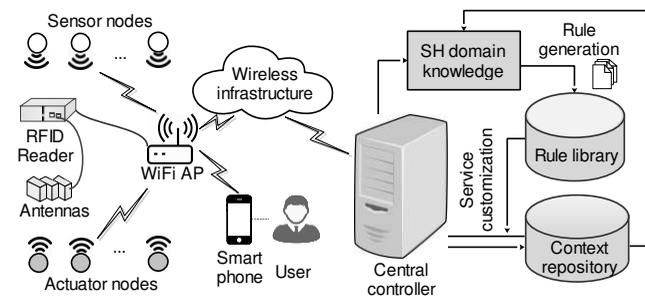


Fig. 6. Architecture of the Prototype System

In this prototype system, the interface and communication are through the following ways:

- (1) Sensors are interfaced with the embedded controllers which are integrated with WiFi adaptors
- (2) WiFi adaptors access to the wireless infrastructure with TCP socket
- (3) The service end of central controller employs Apache/MySQL/PHP server
- (4) User's ID is recognized by RFID reader controlled by a laptop, which talks to central controller with TCP socket

The context information is observed by sensor modules and updated in the database of remote central con-

troller through wireless infrastructure and TCP/IP protocol. Decision commands are produced with the current context and rules, and then responded to actuators for execution. Rule generation is executed in the service end with supervision of smart home domain knowledge.

The devices employed in the prototype system are listed in TABLE 2.

TABLE 2
DEVICES EMPLOYED IN THE PROTOTYPE SYSTEM

Electronic Devices	Devices/Models
Embedded sensors	Temperature, brightness, sound, humidity, human body, smoke, barometer sensors
Actuator	Relays
Embedded controller	MSP430F5739
WiFi adaptor	TI CC3000 WiFi Adaptor
WiFi access point	WD N900 Router
Smartphone	iPhone 4S
RFID reader	Alien 9900+ UHF Reader

The smartphone in this system is used to collect some context data from online providers, such as weather information. It also works as an application interface for users to interact with the system, namely to manually control the home facilities.

4.2 Context Modeling

4.2.1 Context modeling for Smart Home Use Case

In context-aware systems, requirements for context model of being able to express the context situations are put forward to support the context-aware application adaptation [44]. Peer investigations prefer ontology-based context modeling in several modeling approaches [45]. Compared with other alternatives, ontology-based context models are strong in representing complex context information. It provides formal semantics for context knowledge about the objects, relationships and domain constraints, which supports the sharing and integration of structured context information [46]. Hence, ontology based modeling is a competitive candidate for expressing context knowledge in pervasive home environments.

For most cases, a user request or computation task of some applications may not be related to so many context items. However, a holistic context model that incorporates the relevant context items covering most of the use scenarios and for different application domains is really in need for general use. The typical attributes indicating the performance of the context model may be its generality, flexibility, and interoperability. This investigation regards context as the information used to characterize the relevant objects and entities that affect the computation tasks. To this end, this work categorizes the context into the following dimensions: User Context (UC), Computation Context (CC), Network Context (NC), Environment Context (EC), Location Context (LC), Time Context (TC), and History Context (HC).

The above context classes constitute the high-level ontology of the model, and context source modules such as physical sensors, soft sensors, and the user interaction provide raw context data to constitute the low-level on-

tology. For the use case of smart home, the main tasks are about user interaction with the physical environment and control of home facilities. The most relevant context may be the EC, UC, TC, and LC. With the above taxonomy of context, the context model can be tailored to be as shown in Fig. 7 to fit the smart home use case. The sensors and mobile devices are used to characterize the contextual situation to provide proactive decisions executed with actuators.

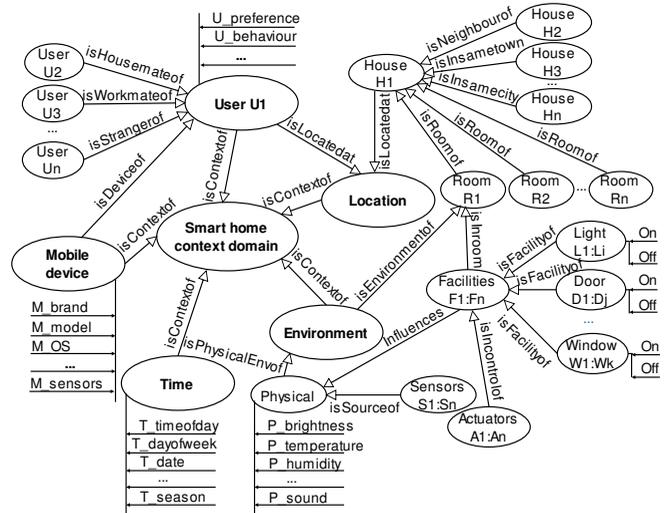


Fig. 7. Smart Home Context Model

4.2.2 Context Items and Their Presentation

In order to justify the feasibility utilizing the semantic distance-based rule matching and the rule generation methods, the system with the proposed system architecture as Section 4.1 presents is established. The context items are presented with linguistic variables, some of which are generated by fuzzifying the dynamic numeric values using the same method presented in [34] and [35]. The linguistic variables characterizing the home environment are given in TABLE 3.

TABLE 3
CONTEXT ITEMS

C1: Brightness (B)	UL, L-ultra low, low M-Medium UH, H-ultra high, high	C5: Noise(N)	L-low H-high
C2: Human(HU)	T-true F-false	C6: Humidity(HD)	L-low H-high
C3: Weather(W)	S-sunny R-rainy C-cloudy	C7: Smoke(S)	L-low H-high
C4: Temperature (TP)	UL, L-ultra low, low M-medium UH, H-ultra high, high	C8: Time(TM)	M-morning A-afternoon E-evening

4.3 User Manual Operations Handling

Although the system anticipates the users' expectation and provides control action as a service automatically, it is required to be able to respond to users' manual operations appropriately as well. In this work, the method to respond to users' manual operations is learned from [35].

The difference is that users in this system carry out the operation by smart phone applications. Therefore, users' manual operations can be easily observed, recorded, and linked to users' profiles. Users' manual operations in this system hold a higher priority than automatic decision of the system.

When user manually controls the home facilities, the state of the object facility is kept for 20 minutes. After that, the system will generate a service decision with rule matching method using current observation of context.

5 TESTING AND EVALUATION

The prototype system integrates the smartphone and embedded sensors into a home environment for home facilities automatic control with a context-aware computing framework. This section presents the application of context-aware service adaptation and rule generation methods in the prototype system and the performance evaluation.

5.1 Testing



Fig. 8. Embedded Sensing Module with WiFi Adaptor and Mobile Application Interface

The testing is carried out in a live-in laboratory environment for three months. The lighting and table fan are selected as the control objects because it is easy to observe their conditions by both human and computer system. User control of the objects is triggered by operations on smartphone, so that the operations and states can be recorded in the database of the system. The embedded sensing module with WiFi adaptor and mobile application interface are given in Fig. 8.

TABLE 4
PRE-DEFINED RULES

ID	Rules (Condition and Decision Attributes)
Rule1	$(C_{Human} \text{ is True}) \wedge (C_{Brightness} \text{ is Ultra Low}) \Rightarrow \text{Light is On}$
Rule2	$(C_{Human} \text{ is True}) \wedge (C_{Brightness} \text{ is Low}) \Rightarrow \text{Light is On}$
Rule3	$(C_{Human} \text{ is True}) \wedge (C_{Temperature} \text{ is High}) \Rightarrow \text{Fan is On}$
Rule4	$(C_{Human} \text{ is True}) \wedge (C_{Temperature} \text{ is Ultra High}) \Rightarrow \text{Fan is On}$
Rule5	$(C_{Human} \text{ is True}) \wedge (C_{Temperature} \text{ is Ultra Low}) \Rightarrow \text{Heater is On}$
Rule6	$(C_{Human} \text{ is True}) \wedge (C_{Temperature} \text{ is High}) \wedge (C_{Weather} \text{ is ny}) \Rightarrow \text{Window is Open}$

In the beginning of the testing, some pre-defined rules as shown in TABLE 4 are set in the system. New rule will be generated and existing ones will be updated by the system when it interacts with users and the context environment changes.

5.2 Results and Evaluation

5.2.1 Testing Results

When the testing starts, some rules are generated and updated according to users' behaviors and contextual situation changes. The number of active rules and the number of rule update by date are recorded by the system, which are as shown in Fig. 9.

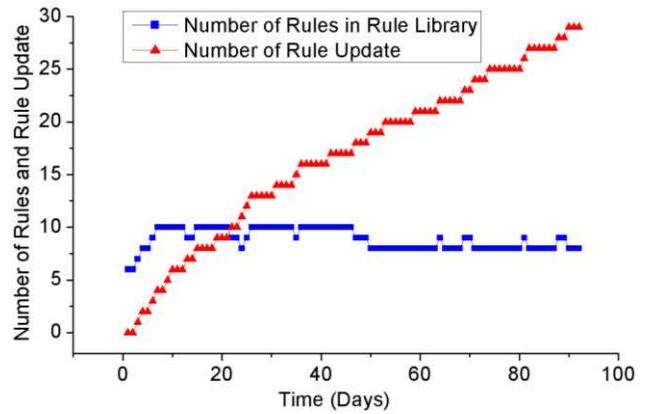


Fig. 9. Number of Rules and Rule Update

The number of rules increases when user starts to use the system. It decreases because of the weather change and the table fan is no longer used. The rule update happens frequently due to seasonal change of day and night alternating time and user's change in habit. Then, in the following sub-sections, feasibility of semantic distance-based rule matching and rough set theory-based rule generation is demonstrated, and performance evaluation of the methods in real application is conducted.

5.2.2 Service Customization with Semantic Distance-based Rule Matching

In TABLE 4, take *Rule1* for example, if human body is detected by a sensor and the ambient brightness is ultra low, the light is turned on automatically. Provided the condition attributes in the smart home are $C = \{C_{Human}, C_{Brightness}, C_{Temperature}, C_{Noise}, C_{Weather}, C_{Humidity}, C_{Smoke}, C_{Time}\}$, and the decision attributes are $D = \{d_{light}, d_{fan}, d_{heater}, d_{window}\}$. The control of the facilities is according to the semantic distance between the sample context value CV_i and the pre-defined rules. Suppose $CV_i = \{\text{True}, \text{Low}, \text{Low}, \text{Low}, \text{Rainy}, \text{Low}, \text{Low}, \text{Morning}\}$, and $dist(CV_i, Rule_i)$ denotes the semantic distance between the context sample CV_i and rule *Rule_i*. Then, according to the context model in section 4.2.1 and rule matching method in section 3.2, the normalized semantic distance between the context sample CV_i and the rules in Table 4 can be calculated.

According to Fig. 7, the temperature in the tree hierarchy indexing structure in the context model can be presented with the diagram in Fig. 10.

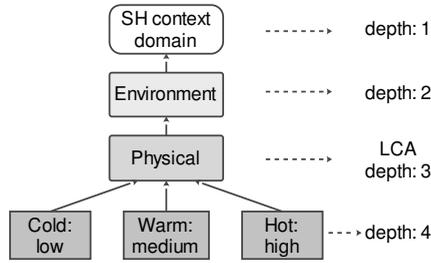


Fig. 10. Tree Hierarchy Structure of Temperature in Smart Home Context Domain

In the context sample CV_i , $CV_i[C_{Temperature}] = Low$. According to the tree hierarchy structure, $depth(LCA(Cold:Low, Hot:High)) = 3$, $depth(Cold:Low) = 4$, $depth(Hot:High) = 4$. According to formula (4) and (5),

$$dist(CV_i[C_{Temperature}], Rule3[C_{Temperature}]) = 1 - 2 * 3 / (4 + 4) = 0.25 \quad (9)$$

With the same method:

$$dist(CV_i[C_{Human}], Rule3[C_{Human}]) = 1 - 2 * 4 / (4 + 4) = 0 \quad (10)$$

Then, according to formula (6),

$$dist(CV_i, Rule3) = 0.5 * dist(CV_i[C_{Temperature}], Rule3[C_{Temperature}]) + 0.5 * dist(CV_i[C_{Human}], Rule3[C_{Human}]) = 0.125 \quad (11)$$

where 0.5 is the weight of the two relevant context items. So, the results of the semantic distance between CV_i and pre-defined rules are given as follows:

$$\begin{aligned}
 dist(CV_i, Rule1) &= (1 - 2 * 4 / (4 + 4) + 1 - 2 * 3 / (4 + 4)) / 2 = 0.125 \\
 dist(CV_i, Rule2) &= (1 - 2 * 4 / (4 + 4) + 1 - 2 * 4 / (4 + 4)) / 2 = 0 \\
 dist(CV_i, Rule3) &= (1 - 2 * 4 / (4 + 4) + 1 - 2 * 3 / (4 + 4)) / 2 = 0.125 \\
 dist(CV_i, Rule4) &= (1 - 2 * 4 / (4 + 4) + 1 - 2 * 3 / (4 + 4)) / 2 = 0.125 \\
 dist(CV_i, Rule5) &= (1 - 2 * 4 / (4 + 4) + 1 - 2 * 3 / (4 + 4)) / 2 = 0.125 \\
 dist(CV_i, Rule6) &= (1 - 2 * 4 / (4 + 4) + 1 - 2 * 3 / (4 + 4)) / 3 = 0.167
 \end{aligned}$$

In this system, weight of attributes in the rules is equal. The weight is $1/n$ when there are n attributes in a rule. The semantic distance values are presented in Fig. 11.

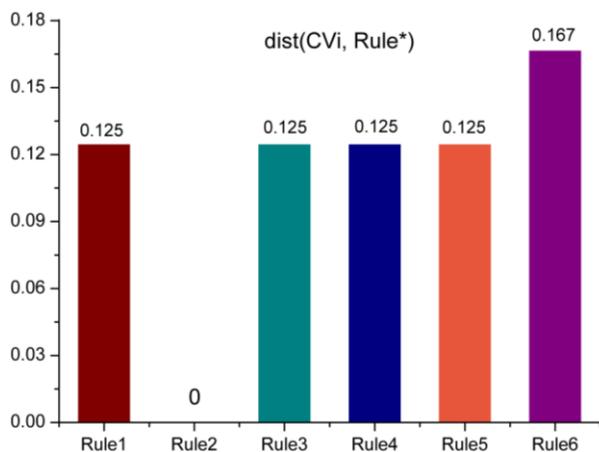


Fig. 11. Semantic Distance between CV_i and Rules

It is evident that the minimum semantic distance between CV_i and the pre-defined rules is $dist(CV_i, Rule2)$. Thus, $Rule2$ is considered to be the best matched rule for execution, and the relevant service 'Light is ON' is executed automatically.

5.2.3 Rough Set Theory-based Rule Generation

There are a lot of devices and relevant context items in the smart home environment, and it is quite difficult to chart the rules from the complex relations between the facility control and context value. In this prototype system, the rule generation method is demonstrated by the use case of table light control with particular context values and user operations. The user operations with mobile devices and the corresponding context saved in the context repository database are listed in TABLE 5, where LO denotes 'Light ON' and LF denotes 'Light OFF'.

TABLE 5

HISTORY CONTEXT AND USER APPROVED OPERATIONS

ID	Context Items and Values							User Operation	
	C1 (B)	C2 (HU)	C3 (W)	C4 (TP)	C5 (N)	C6 (HD)	C7 (S)		C8 (TM)
1	L	T	S	H	L	L	L	E	LO
2	H	T	S	H	H	L	L	M	LF
3	L	T	C	H	L	L	L	A	LO
4	H	T	S	H	L	L	L	A	LF
5	L	T	C	H	H	L	L	M	LO
6	L	F	C	H	H	L	L	M	LF
7	L	T	C	H	L	L	L	A	LO
8	H	T	S	H	L	L	L	A	LF
9	L	T	C	H	L	L	L	M	LO
10	H	T	S	H	L	L	L	M	LF
11	L	T	R	M	L	H	L	M	LO
12	H	F	S	H	H	L	L	A	LF
13	L	T	R	H	L	H	L	A	LO
14	H	T	S	M	L	L	L	M	LF
15	L	T	S	M	L	L	L	E	LO
16	H	T	S	H	L	L	L	A	LF
17	L	T	R	H	L	H	L	M	LO
18	H	T	S	M	L	L	L	M	LF
19	L	T	S	H	H	L	L	M	LO
20	H	T	S	H	L	L	L	M	LF
...

According to formula (8), for the first row in TABLE 5, $DM(1,1)=0$, since decision attribute $d(1)=d(1)$. For the second row, $DM(1,2)=[C_1, C_5, C_8]$ represented as $C_1C_5C_8$, which means the values on the 1st, 5th, and 8th attributes are different when decisions $d(1) \neq d(2)$. With the same method, the discernibility matrix $DM_{20 \times 20}$ for the 20 context value samples in TABLE 5 can be obtained as shown in (12).

According to algorithm 2, when the discernibility matrix is simplified to Φ , the reduction $Red=\{c1, c2\}$ is obtained. The result means that brightness and human play the decisive role in the task of light control. Therefore, the following rules can be generated accordingly:

- (1) $(c_1 = Low) \wedge (c_2 = True) \Rightarrow Light\ is\ On$
- (2) $(c_1 = High) \wedge (c_2 = True) \Rightarrow Light\ is\ Off$
- (3) $(c_1 = Low) \wedge (c_2 = False) \Rightarrow Light\ is\ Off$
- (4) $(c_1 = High) \wedge (c_2 = False) \Rightarrow Light\ is\ Off$

Take Rule (1) as an example: it represents that if the ambient brightness is low and the user is in the particular area, the light will be turned on. The rules created in the

	1	2	3	4	5	6	...	16	17	18	19	20	
1	0												
2	$C_1C_5C_8$	0											
3	0	$C_1C_3C_5C_8$	0										
4	C_1C_8	0	C_1C_3	0									
5	0	C_1C_3	0	$C_1C_3C_5C_8$	0								
6	$C_2C_3C_5C_8$	0	C_2C_5	0	C_2	0							
			:				...						
16	C_1C_8	0	C_1C_3	0	$C_1C_3C_5C_8$	0		0					
17	0	$C_1C_3C_5C_6$	0	$C_1C_3C_6C_8$	0	$C_2C_3C_5C_6$		$C_1C_3C_6C_8$	0				
18	$C_1C_4C_8$	0	$C_1C_3C_4C_8$	0	$C_1C_3C_4C_5$	0	...	0	$C_1C_3C_4C_6$	0			
19	0	C_1	0	$C_1C_5C_8$	0	C_2C_3		$C_1C_5C_8$	0	$C_1C_4C_5$	0		
20	C_1C_8	0	$C_1C_3C_8$	0	$C_1C_3C_5$	0		0	$C_1C_3C_6$	0	C_1C_5	0	

rule generation can then be added to the rule library to supervise the context-aware automation service. Therefore, the system is empowered with the capability to generate new context-aware automation rules and the capability to adapt the system to dynamic context environments.

5.2.4 Efficiency of the Algorithms in Practical Use

In order to evaluate the efficiency of the rule generation algorithm, it is tested as a service application with the above use case. The algorithm is tested on two platforms: (1) local host Apache/MySQL/PHP (XAMPP 1.82) server on an iMAC machine with 2.66GHz Intel Core 2 Duo CPU and 4G 1067MHz DDR3 RAM; and (2) remote Apache/MySQL/PHP (XAMPP 1.82) server as well with Intel Core i7-3770 CPU@3.40GHz×8 and 16GB 1066 MHz DDR3 RAM shipping Ubuntu 12.04. The testing results on the two platforms are as shown in Fig. 12 and Fig. 13.

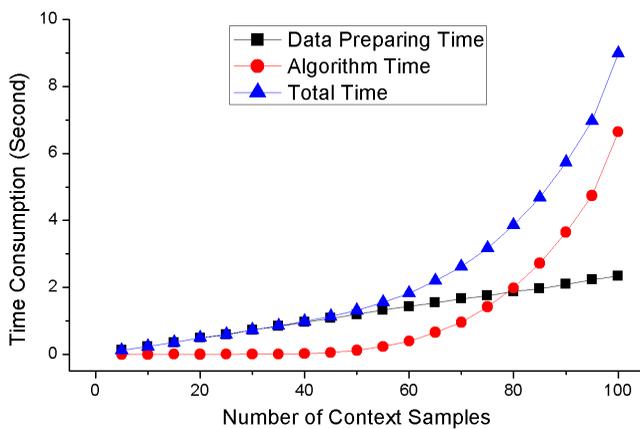


Fig. 12. Time Complexity of Rule Generation on iMac Local Host Server

the performance of the algorithm is much better on a remote server than on the local host server. That is to say, performance of the algorithm is highly related to the computation power of testing platforms. Therefore, the rule generation process is divided into two procedures for further analysis: (1) data initiation, and (2) rule generation algorithm. The data initiation procedure is the process of Database data query, and the rule generation algorithm is the process of rule generation computation with the data in memory that is accessed from Database.

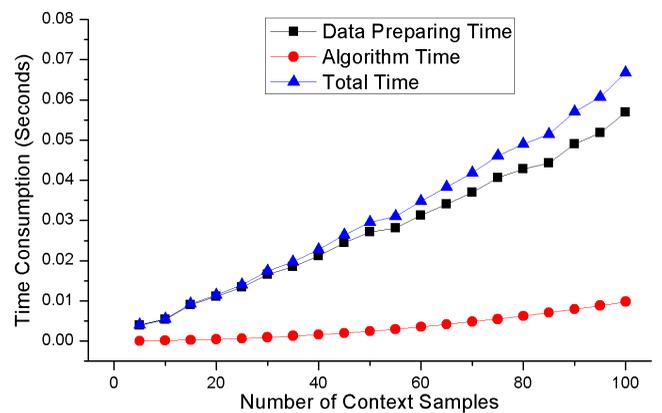


Fig. 13. Time Complexity of Rule Generation on Remote Testing Server

Fig. 12 and Fig. 13 illustrate the relationship between time consumption and number of context samples of rule generation algorithm on the two platforms with 8 context attributes for the light control use case. It is apparent that

From Fig. 12 the rule generation is fast when the number of context samples is not large, especially when the number of context samples is less than 40, while the data initiation time is much longer. But with the number of context samples increases over 50, the time consumption

of the rule generation algorithm increases very fast. When the number of context samples is over 80, the rule generation algorithm time is longer than the data initiation time and the difference increases steeply with the number of context samples. In the whole process, the data initiation time increases with the number of context samples linearly, while the rule generation algorithm time has a nearly exponential relationship with the number of context samples. From Fig. 13, both the data initiation time and rule generation algorithm time are very short and they are nearly in linear with increase of context samples. With 100 context samples, the time consumption is only about 0.067 second. Therefore, the real-time performance can be guaranteed for practical use in context-aware computing systems.

Evidently, the varying trends of the curves in Fig. 12 and Fig. 13 are different. That is probably because of the difference in the computational power of the two experiment platforms. When the computation power is not strong enough, the time consumption of the rule generation is in exponential growth.

5.3 Discussion

Normally, the computation tasks are related to users' preferences and the ambient environment in a sophisticated relationship, and the practical home environment is usually complicated. The prototype system presented in Section 4 conducts a proof of concept verification of the proposed context-aware computing service customization strategy including both the computing service decision making and the rule generation methods.

The proposed service customization strategy separates the two steps of computing service decision making and rule generation. The rule matching method can be used for decision making of control service using the current context, and the rule generation method is then used to determine new rules. Since the computation in this process is only to calculate the semantic distance between the context set and a limited number of rules in the rule set, the computation load of the decision making is lightweight. Compared with some existing methods, this method can get rid of the limitation of traditional IF-THEN logic and make the system capable of determining the rules with history context. In addition, the semantic distance based method explicitly defines decision making with Manhattan distance and GCSM, which is suitable for the diverse data types and tree-hierarchy relations of context compared with FMADM and MAUT. It also needs fewer context samples and less computation compared with the probability-based method, which requires numerous context samples to guarantee accuracy. Then, compared with the ANFIS, Rough-Fuzzy, and Fuzzy computing methods, the separation of service decision making step and rule generation step of the proposed strategy makes it more lightweight in computation and easier to implement.

In smart home environment, the interrelationship between the entities is presented with the context model for context-aware service provisioning, and the structure may be complex. Meanwhile, the data type, accuracy,

source, and validation time period of context may be significantly different. The heterogeneous data requires the service customization strategy to be able to effectively handle the heterogeneous context data and provide the appropriate computing service in real-time. The semantic distance-based rule matching method is strong in handling context data of different data structures, both numeric and tree-hierarchy of indexing terms in ontology describing sophisticated relationships. Therefore, it is appropriate for the context-rule matching for decision making, and it is easy to implement in the service end of computing systems.

For the rule generation method, the rough set theory is a mathematical method strong in handling information with some degree of uncertainty. In Section 5, it successfully determines the key context attributes affecting the results of the context-aware service decision making. From the experimental study, results show that the rule generation algorithm is fast enough for practical use, provided the service platform is competent in computation power. The computation power of the platform, efficiency of the Database data access, and number of context samples are key factors influencing the speed of the rule generation process. Theoretically, the more context value samples involved in the rule generation, the more complete and accurate the rules for the use case will be. But more computation load and risk of failures in real-time processing may be brought to the system. Therefore, how the proper quantity of context samples is selected to create effective rules efficiently and how the generated rules are integrated in the rule library is important for context-aware systems.

6 CONCLUSION AND FUTURE WORK

This investigation introduces a context-aware computing service customization strategy and implements a smart home prototype system for home facility automatic control, with smartphones and commercially available low-cost sensing techniques. Results and evaluation demonstrate that the proposed service customization strategy is feasible to determine the appropriate computing service and is capable of timely generation of the potential rules. By generating and updating new rules, the system can autonomously learn through its observation to anticipate users' expectations. Lessons learned in the proof of concept implementation can be summarized as follows:

- (1) The proposed context-aware adaptation model and the service customization strategy taking advantage of various sensing techniques to provide proactive control service in home environment is feasible and practicable.
- (2) The context-aware adaptation strategy simplifies the complexity of structure and reduces the computation load by separating the steps of rule matching based decision making and rule generation.
- (3) The semantic distance-based rule matching method is competent in handling the heterogeneous context data and its interrelationship for context-aware service provisioning in terms of smart home use scenario.
- (4) The RST-based rule generation method can effec-

tively determine the key context items contribute to the decision making of context-aware service and generate rules to supervise the customization with accepted time complexity.

This investigation focuses on the proof of concept implementation and performance evaluation of the service customization strategy. Although the proposed methods are successfully implemented and good results are generated, there are still open issues which deserve further study. The future work may fall on the aspects as outlined below:

(1) To investigate the method of selecting the proper quantity of context samples for rule generation, considering accuracy of generated rules, computation power of the platform, and real-time performance of the context-aware system.

(2) To integrate the Quality of Context (QoC) into computation to effectively characterize the user and physical environment towards a more effective decision as a computing service.

(3) To determine the weight of the context items and distinguish them for different computation tasks in service customization, since context items may make different contributions in the final decision making.

(4) To design a complete performance evaluation strategy, through comparison of the context-aware service with users' expectations in a satisfaction model, and put the system into comprehensive evaluation.

The future work will focus on the above key issues towards efficient and accurate decision support. The methods to accurately and comprehensively characterize the contextual situation with sensing techniques, efficiently generate the effective rules, and evaluation strategy for service customization and rule generation are central concerns in further investigation.

REFERENCES

- [1] M.R. Alam, M.B.I. Reaz, M.A.M. Ali, "A review of smart homes – past, present, and future," *IEEE Transactions on Systems, Man, and Cybernetics–Part C: Applications and Reviews*, vol. 42, no. 6, pp. 1190-1203, 2012.
- [2] F. Xia, Y.-C. Tian, Y. Li, Y. Sung, "Wireless sensor/actuator network design for mobile control applications," *Sensors*, vol. 7, no. 10, pp. 2157-2173, 2007.
- [3] Y. Zeng, D. Li, A.V. Vasilakos, "Real-time data report and task execution in wireless sensor and actuator networks using self-aware mobile actuators," *Computer Communications*, vol. 36, no. 9, pp. 988-997, 2013.
- [4] M.A. Feki, et al., "The internet of things: the next technological revolution," *Computer*, vol. 46, no. 2, pp. 24-25, 2013.
- [5] W. Lumpkin, "The internet of things meets cloud computing," *IEEE Consumer Electronics Magazine*, vol. 2, no. 2, pp. 47-51, 2013.
- [6] I. Jantunen, et al., "Smart sensor architecture for mobile-terminal-centric ambient intelligence," *Sensors and Actuators A: Physical*, vol. 142, pp. 352-360, 2008.
- [7] S. Makonin, L. Bartram, F. Popowich, "A smarter smart home: case studies of ambient intelligence," *IEEE Pervasive Computing*, vol. 12, no. 1, pp. 58-66, 2013.
- [8] G. D. Abowd, et al., "Towards a better understanding of context and context-awareness," *Proc. of the 1st International Symposium on Handheld and Ubiquitous Computing*, pp. 304-307, 1999.
- [9] W.-Y., Chung, S.-J. Oh, "Remote monitoring system with wireless sensors module for room environment," *Sensors and Actuators B*, vol. 113, pp. 64-70, 2006.
- [10] L. Liang, L. Huang, X. Jiang, Y. Yao, "Design and implementation of wireless smart-home sensor network based on ZigBee protocol," *International Conference on Communications, Circuits and Systems*, pp. 434-438, May 2008.
- [11] M. Li, Z. Li, A. V. Vasilakos, "A survey on topology control in wireless sensor networks: taxonomy, comparative study, and open issues," *Proceedings of the IEEE*, vol. 101, no. 12, pp. 2538-2557, 2013.
- [12] H.-J. Kober, H. Wattar, G. Scholl, "Modular wireless real-time sensor/actuator network for factory automation applications," *IEEE Transaction on Industrial Informatics*, vol. 3, no. 2, pp. 111-119, 2007.
- [13] J. Parra, et al., "Flexible smart home architecture using device profile for Web service: a Peer-to-Peer approach," *International Journal of Smart Home*, vol. 3, no. 2, pp. 39-56, 2009.
- [14] P. Chen X. Zhi, "Smart home architecture based on event-driven DPWS," *J. Shanghai Univ. (Engl. Ed.)*, vol. 15, no. 5, pp. 386-390, 2011.
- [15] C. Lai, et al., "A RF4CE-based remote controller with interactive graphical user interface applied to home automation systems," *ACM Transactions on Embedded Computing Systems*, vol. 12, no. 2, pp. 1-20, 2013.
- [16] Z. Sheng, et al., "A survey on the IETF protocol suite for the Internet of Things: standards, challenges, and opportunities," *IEEE Wireless Communications*, vol. 20, pp. 91-98, 2013.
- [17] Z. Yan, P. Zhang, A. V. Vasilakos, "A survey on trust management for Internet of Things", *Journal of Network and Computer applications*, vol. 42, pp. 120-134, 2014.
- [18] S.-T. Cheng, C.-H. Wang, G.-J. Horng, "OSGi-based smart home architecture for heterogeneous network," *Expert Systems with Applications*, vol. 39, pp. 12418-12429, 2012.
- [19] I.-H. Bae, "An ontology-based approach to ADL recognition in smart homes," *Future Generation Computer Systems*. Submitted for publication.
- [20] Y. Charlon, et al., "Activity monitoring system for elderly in a context of smart home," *IRBM*, vol. 34, pp. 60-63, 2013.
- [21] H. Medjahed, et al., "A pervasive multi-sensor data fusion for smart home healthcare monitoring," *2011 IEEE International Conference on Fuzzy Systems*, pp. 1466-1473, 2011.
- [22] G. Acampora, et al., "A survey on ambient intelligence in healthcare," *Proceedings of the IEEE*, vol. 101, no. 12, pp. 2470-2494, 2013
- [23] C.-M. Huang, Y.-C. Huang, C.-J. Huang, "An intelligence-based fuzzy inference system for smart home real-time load forecasting," *Procedia Engineering*, vol. 50, pp. 297-304, 2012.
- [24] X. Wang, et al., "Semantic space: an infrastructure for smart spaces," *IEEE Pervasive Computing*, Vol. 3(3), pp.32-39, 2004.
- [25] L. Chen, et al., "Semantic smart homes: towards knowledge rich assisted living environments," *Intelligent Patient Management – Studies in Computational Intelligence*, Vol. 189, pp. 279-296, 2009.
- [26] J. Soldatos, et al., "Semantic web technologies for ubiquitous computing resource management in smart space," *International Journal of Web Engineering and Technology*, Vol. 3(4), pp. 353-373, 2007.
- [27] S. S. Intille, et al. "Using a Live-in Laboratory for ubiquitous

computing research," *Pervasive Computing Lecture Notes in Computer Science*, Vol. 3968, pp. 349-365, 2006.

- [28] A. Raráu, K. Pusztai, I. Salomine, "MultiFacet item based context-aware applications," *International Journal of Computing & Information Science*, vol. 3, No. 2, pp. 10-18, 2005.
- [29] P. TalebiFard, V. C.M. Leung, "A data fusion approach to context-aware service delivery in heterogeneous network environments," *Procedia Computer Science*, vol. 5, pp. 312-319, 2011.
- [30] R. Schäfer, "Rules for using Multi-Attribute Utility Theory estimating a user's interests," *Workshop Adaptivity and User Modeling*, 2001.
- [31] X. Wang, D. Rosenblum, Y. Wang, "Context-aware mobile music recommendation for daily activities," *Proceedings of the 20th ACM International Conference on Multimedia*, 29 October - 2 November, Nara Japan. pp. 99-108, 2012.
- [32] A. Al-Hmouz, J. Shen, R. Al-Homouz, J. Yan, "Modeling and simulation of an Adaptive Neuro-Fuzzy Interface System (ANFIS) for mobile learning," *IEEE Transaction on Learning Technologies*, vol. 5, no. 3, pp. 226-237, 2012.
- [33] Q. Duan, D. Miao, H. Zhang, J. Zheng, "Personalized Web retrieval based on Rough-Fuzzy method," *Journal of Computational Information System*, vol. 3, No. 2, pp. 203-208, 2007.
- [34] G. Acampora, et al., "Interoperable and adaptive fuzzy service for ambient intelligence applications," *ACM Transactions on Autonomous and Adaptive Systems*, vol. 5(2), pp. 1-26, 2010.
- [35] A. M. Vainio, et al., "Proactive fuzzy control and adaptation methods for smart homes," *IEEE Intelligent System*, vol. 23, pp. 42-49, 2008.
- [36] V. Loia, A. V. Vasilakos, "Editorial: special issue on ambient intelligence," *ACM Transactions on Autonomous and Adaptive systems*, vol. 5, no. 2, pp. 5-6, 2010.
- [37] M. Weiser, "The computer for 21st century," *Scientific American*, vol. 265, no. 3, pp. 94-104, 1991.
- [38] E. F. Krause, "Taxicab Geometry: An Adventure in Non-Euclidean Geometry," New York: Dover Publications, 1987.
- [39] S. Benabderrahmane, M. Smail-Tabbone, O. Poch, A. Napoli, M.-D. Devignes, "IntelliGO: a new vector-based semantic similarity measure including annotation origin," *BMC Bioinformatics*, vol. 11, pp. 1-16, 2010.
- [40] P. Ganesan, H. Garcia-Molina, J. Widom, "Exploiting hierarchical domain structure to compute similarity," *ACM Transactions on Information System*, vol. 21, pp. 64-93, 2003.
- [41] Z. Pawlak, "Rough Sets," *International Journal of Computer & Information Sciences*, vol. 11, no. 5, pp. 341-356, 1982.
- [42] Z. Pawlak, A. Skowron, "Rough sets: some extensions," *Information Science*, vol. 177, pp. 28-40, 2007.
- [43] Z. Huang, X. Lu, H. Duan, "Context-aware recommendation using rough set model and collaborative filtering," *Artificial Intelligence Review*, vol. 35, pp. 85-99, 2011.
- [44] D. Preuveneers, et al., "Towards an extensible context ontology for ambient intelligence," *Proc. of the Second European Symposium on Ambient Intelligence*, pp. 148-159, 2004.
- [45] C. Hoareau, I. Satoh, "Modeling and processing information for context-aware computing: a survey," *New Generation Computing*, vol. 27, pp. 177-196, 2009.
- [46] M. Poveda-Villalon, et al., "A context ontology for mobile environments," *Workshop on Context, Information and Ontologies Co-located with EKAW 2010*. [No pagination], October 2010.



Z. Meng received his BSc and MSc degrees on instrumentation and measurement from Sichuan University (China) in 2006, and Beihang University (China) in 2009 respectively. He received his PhD degree on computer science from University of Huddersfield (UK) in 2014. He then joined The University of Manchester (UK) as a research associate in June 2014. His research interests include embedded electronics and real-time software, wireless sensor network, pervasive and mobile computing, and smart environments and ambient intelligence.



J. Lu is Professor in Informatics at the University of Huddersfield (UK). Her extensive research covers information access, retrieval and visualization, XML technology, object oriented technologies, agent technology, data management systems, security issues, wireless response technology and Internet computing. She has been an invited speaker for industrial-oriented events and published 5 academic books. Professor Lu has acted as the founder and a program chair for the International XML Technology workshop and XMLTech (USA) for 11 years (2003-2011). She also serves as Chair of 5 separate international conferences, is a regular reviewer for several international journals, and a committee member for 16 international conferences. Professor Lu serves as a member of the British Computer Society (BCS), BCS examiner of Advanced Database Management Systems, and fellow of the Higher Education Academy (UK). She is a founder and Editor-in-Chief for the International Journal of Information Retrieval Research (IJRR).