



A Prototype for the analysis of Conflict resolving Algorithms in Context aware Environment using simulation

*Thyagaraju.GS, ** Poornima.D

*Faculty and research scholar, Department of Computer science, SDM CET,
Dharwad, Karnataka, India.

thyagaraju_gs@yahoo.co.in

**pursuing 4th Semester M.tech, Department of Computer science SDM CET Dharwad, Karnataka, India
poornima.udaykumar@gmail.com

ABSTRACT

In this paper we are proposing a GUI based Prototype for user centered environment like class room, library hall, laboratory, meeting hall, coffee shop, kitchen, living room and bedroom, which recommends useful services based on the user's context. Service recommendation is mainly based on parameters such as user, location, time, day and mood. In addition whenever the conflict arises among different users it will be resolved using some conflict resolving algorithms. The motivation behind the proposed work is to improve the user satisfaction level and to improve the social relationship between user and devices. The prototype contains simulated sensors which are used to capture the raw context information, which is then described with meaningful English sentence and services are recommended based on user's situation. The proposed conflict resolving algorithms are Rule based algorithm, Bayesian probability based algorithm and Rough set theory based algorithm. The amount of conflicts resolved by these algorithms is also analyzed at the end.

Indexing terms/Keywords

CEAR Diagrams context, context aware systems conflict degree, decision table, fitness degree, precision, ubiquitous, rough set, UID (user identity)

Academic Discipline and Sub-Disciplines

Computer science.

SUBJECT CLASSIFICATION

Artificial Intelligence.

TYPE (METHOD/APPROACH)

- Quasi-Experimental
- Simulation based analysis of proposed conflict resolving algorithms.
- Design and Implementation of prototype.

Council for Innovative Research

Peer Review Research Publishing System

Journal: INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY

Vol 10, No 6

editor@cirworld.com

www.cirworld.com, member.cirworld.com



INTRODUCTION

The advancement of pervasive computing, sensor technology and artificial intelligence techniques such as rough set theory, Bayesian probability, Fuzzy logic is motivating many researchers across the world to improve the intellectual and social relationship between man and machine. This trend has the given scope for the development of context aware applications. [1,2,3,4]

Context aware computing refers to the technology where computing devices can sense and react to the physical environment where they are deployed. A system is context-aware if it uses context to provide suitable information and/or services to the user, depending upon user's situation. The context awareness is one of the most important technologies in pervasive computing, which facilitates information acquisition and execution by supporting interoperability between users and devices based on user's situation. User environment contain context information such as location, identity of the user, activity, mood, time and the state of the nearby people. Context aware computing involves sensing this information to implicitly provide appropriate information and services Context aware applications are usually associated with one or more users competing for availing one or more multiple services. Ubiquitous applications need to resolve conflicting context information. Humans resolve conflicts with heuristics such as starting a discussion, compromising the agreement, retiring gracefully, or ignoring the conflict altogether. But conflicts for device access and its services among multiple users can be resolved with some algorithms known as conflict resolving algorithms which are proposed in this paper.

In the past several years there has been an increased effort and interest in developing and deploying context aware applications. Currently, developing such application requires developers to carryout lots of laborious works, experiments and direct interaction with hardware, sensors and devices. For developers, it is difficult to work in real context aware environment since context aware appliances such as sensors and other devices are very expensive. Therefore before developing and deploying such applications in real time, it will be much easier if we test the working of such application via simulation. Hence we are using simulation based approach for developing and deploying context aware application for some selected user centered environment like context oriented living room, kitchen, bedroom, laboratory, class room, meeting room, library hall, and coffee shop. whenever the conflict arises among different users it will be resolved using the selected algorithm such as Rule based Algorithm, Bayesian Probability based algorithm and Rough set theory based algorithm. The algorithms are compared in terms of amount conflict resolution, by making use of simulated context.

The motto of this proposed work is to enable context aware applications to offer socialized and personalized services to the users and also improve the user-device interoperability according to user needs to increase the comfort level of users.

RELATED WORKS

Context aware applications and services use context information to provide relevant services to the user and task at hand [1,2,3]. The author Matthew Lee in his paper "Intro to context aware computing"[4] explains some basic concepts of context aware computing. Some challenges for the development of context aware applications have also been identified in this paper, namely, how to represent the context internally? What sensors are necessary? how to sense location information?. "context aware computing applications"[5] by Bill Schilit has identified and categorized context aware applications as manual, automatic information and commands. The concept of context aware application is explained in an organized and effective way using these four categories.

The idea of Cyberguide [6] is introduced by Gregory D Abowd in his paper "context awareness in wearable ubiquitous computing. It is a prototype that provides location specific services to the user, developed specially for tourists. It gives the list of places to visit and also provide some practical information such as nearby hotels, restaurants using interactive map and user modifiable database.

The common architectural principles of context aware systems, layered conceptual design framework is explained in "survey on context aware systems"[7]. In this paper, the author Matthias Baldauf has explained different architectures such as direct sensor access, middleware infrastructure and context server, based on different methods of acquiring contextual information. In our paper, we have used direct sensor access method for acquiring context information. This survey paper has also given satisfactory explanation regarding context models, different types of sensors and frameworks.

A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications[8] explains Context Toolkit and demonstrates how the Context Toolkit can be used to design and build context-aware applications and how it supports reusability of components, evolution of applications and the acquisition and use of complex context.

Various applications have also been developed by exploiting the concept of context awareness, such as context aware TV [1], context aware mobile [9] context awareness for internet things [10]. These works are achieving success and motivating many researchers to contribute their dedication and innovative ideas in this field. Such works are making human life easy and comfortable where in devices will be human friendly and serve according to their needs and situation.

Over last few years, several researchers have been working on some tools to support context aware applications. Following are few such works.



- Visual RDK (Rapid development kit) [11]: It is a high level programming language for prototyping ubiquitous applications. Since the context information is highly integrated to programming language itself, it becomes infeasible when more than one class or sensors or devices are used.
- PCOM (component system for pervasive computing) [12]: It offers application programmers a high level programming abstraction. It has a configuration engine which determines the components available in the scenario and dynamically composes applications. PCOM address adhoc scenario but visual RDK relies on infrastructure such as context server.
- Tiny OS [13]: It is open source operating system designed for wireless embedded sensors networks. Due to the technical advancements, nowadays researchers are trying to compresses the sensor devices onto to a single chip by taking the advantage of CMOS and semiconductor process. To support this task, a system software is required to manage and operate these sensors, which is accomplished by Tiny OS.
- Jigsaw [14]: with the emergence of ubiquitous computing technology, it is important to know how the devices are placed within a given context, how the combination of devices is practically managed and how the information flow takes place between the devices. Jigsaw is an editor where devices are represented by jigsaw pieces. Each jigsaw piece(which represent device) is dragged and dropped into the editor's work space and have to be configured. The editor provides audio feedback on the interaction among the devices.
- Ubiwise [15]: it is a simulator for ubiquitous computing which concentrates on computation and communication devices situated in physical environment.

In the literature the problem of conflict resolution or service recommendation is done based on some fixed database using different approaches: Content Based Approach and Content Boosted Collaborative Approach. In both the cases the factors like item quality, user rating and user priority are used for resolving the conflict and hence recommending the services.

2.1 Content Based Approach

Content – based filtering items are matched either to a user's interest profile or query on the basis of content rather than opinion. One of the strengths of this approach over collaborative filtering is that as long as the system has some information about each item, recommendations can be made even if the system has received a small number of ratings or none at all. The downside is that each item must be characterized with respect to the features that appear in a user's profile and further the profile of each user must be collected and modelled. The work presented in Content-based Recommender Systems: State of the Art and Trends et al Pasquale Lops, Marco de Gemmis and Giovanni Semeraro, et al [16] explains the pros and cons of content based recommender.

2.2 Content Boosted Collaborative Approach

The content boosted collaborative approach is a hybrid of content and collaborative based conflict resolution and service recommendation system. In content based conflict resolution one tries to resolve conflict by considering the past most frequently recommended items or services. In collaborative based conflict resolution one tries to resolve conflict by considering the more similar users. In a hybrid technique both user rating and similarity among user is considered for resolving conflict. The research work presented by Prem Melville and Raymond J. Mooney and Ramadass Nagarajan, et al [17] , proposes the recommender systems which make use of content (item rating) boosted collaborative (collective user or group rating) to predict new items of user interest. In the user-based content boosted collaborative filtering recommendation system, the user ratings data are used to determine the similarity between users.

2.3 Comparison between the related and proposed work

Some of the comparisons between the proposed work and previous works are as follows:

1. In our proposed work, Conflict resolution will be based on user personal and social context unlike in collaborative approach which considers only user ratings but not the user's personal parameters.
2. To enhance the performance of existing content boosted conflict resolving algorithms we are making use of AI tools like fuzzy logic, Bayesian Network, Rule Base and Rough Set Theory based Decision Table.
3. Problems Context aware application such as how to handle the data obtained from sensors and how to resolve conflicts in multiuser environment are addressed and resolved in this paper.

3. CONTEXT MODELLING

Context modeling is a process of representing user context effectively from the end user prospective. User Context is any information that can be used to characterize the situation of user. It is derived from the categories of primitive contexts described in table 1.

**Table 1. Primitive contexts and Examples**

Slno	Primitive contexts	Example
1.	Environment	Temperature, light intensity, sensors and devices
2.	Social/role	Father, mother, son, daughter, students etc
3.	Personal	Age factor(AF), role factor(RF), family dependency factor(FDF) and social dominance factor(SDF)
4.	Location	Home and college campus (in our project)
5.	Sub location	Kitchen, bed room, living room, class room, lab, library, meeting hall and coffee shop(considered in our project)
6.	Activity	Sitting, standing, discussing, lying etc
7.	Mood	Happy, sad, active, relaxed, depressed etc
8.	Temporal	Day and time.

Depending on the number of users participating in particular situation user context is divided into two categories as described in table 2. The service recommendation in single user context is quite easy because the issue of user conflicts does not arise in single user context unlike multiuser context, where user conflicts arise. Such conflicts can be resolved using appropriate algorithms proposed in this paper.

Table 2. Context category with description and Examples

Slno	Context category	Description	Example
1.	Single User Context	The context which describes the situation of single user.	Father is watching TV in the Living Room in Relax Mood during late night”.
2.	Multiuser Context	The context which describes the situation of multiuser	Family members Watching TV during late night

3.1 CEAR Diagrams

The context modeling is presented graphically using Context Entity Attributes Relationship Diagram (CEAR). The diagram is a graphical representation of association between the context entities and attributes, representing context of the given entity in context aware environment.

Figure 1 gives the CEAR diagram for the context “Father is reading newspaper in living room”. In order to identify the given context we have used seven simulated sensors in this diagram. The user is identified using UID (user identity) which is extracted from user profile database. The sensors s1 and s2 identifies the main location and sub location of the user respectively in the form of raw vales using some techniques such as Bluetooth or IP MAC address. By using these raw values, the exact location of the user is extracted from location database. The sensors s4 and s5 identifies the user activity in the form of raw values using which, the exact user activity is extracted from activity database. Similarly, the environmental factors such as temperature and light intensity in the current user context are extracted from sensors s6 and s7 and temporal factor such as time is identified using sensor s3. In this way we can easily understand the current user context by making use of simulated sensors and symbolic notations of CEAR diagram. One such example is shown in the figure 1

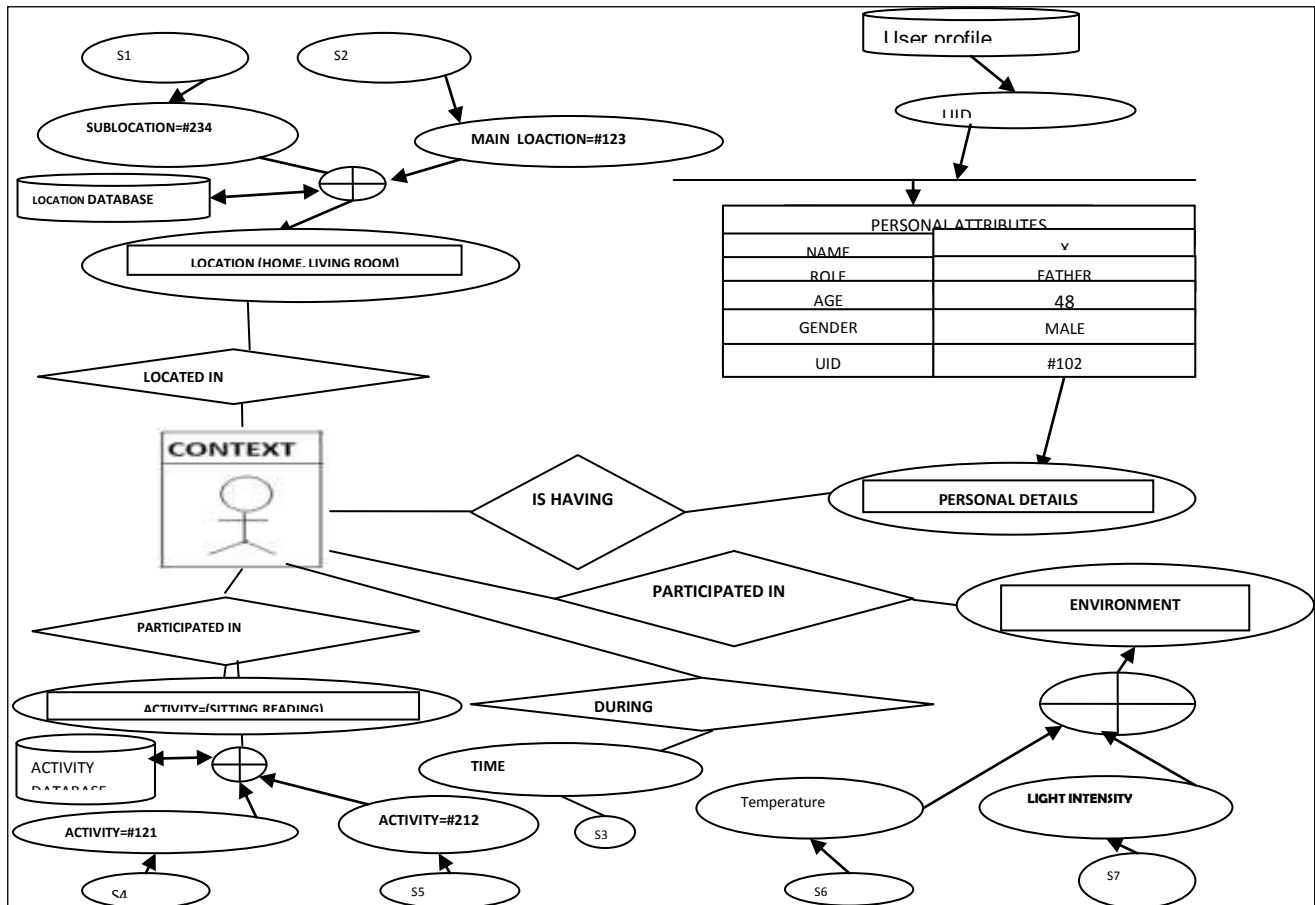


Fig1 CEAR diagram for the context “Father is reading newspaper in living room”

3.2 Benefits of context modeling

Context Modeling is a fundamental step for the development of context aware systems. It has following benefits

- The model helps in understanding context aware problems such as how to handle the data obtained from sensors, how to resolve conflicts in multiuser environment and so on.
- The context model helps in documentation process
- It also guides the developer in the process of Database design.
- Currently, the programming of context-aware applications is complex and laborious. This situation could be remedied by design and developing the appropriate modeling and programming paradigm for context aware applications.

4. ARCHITECTURE OF THE PROPOSED SYSTEM

The architecture of the prototype simulator is shown in the figure 2. It has following three phases

- Conflict detection:** conflict is the degree of disagreement or dissatisfaction among users with respect to the services recommended by prototype simulator. Conflict detection is achieved by providing input to the simulated sensors. Usually conflicts arises when there will be more than one user in a given context.

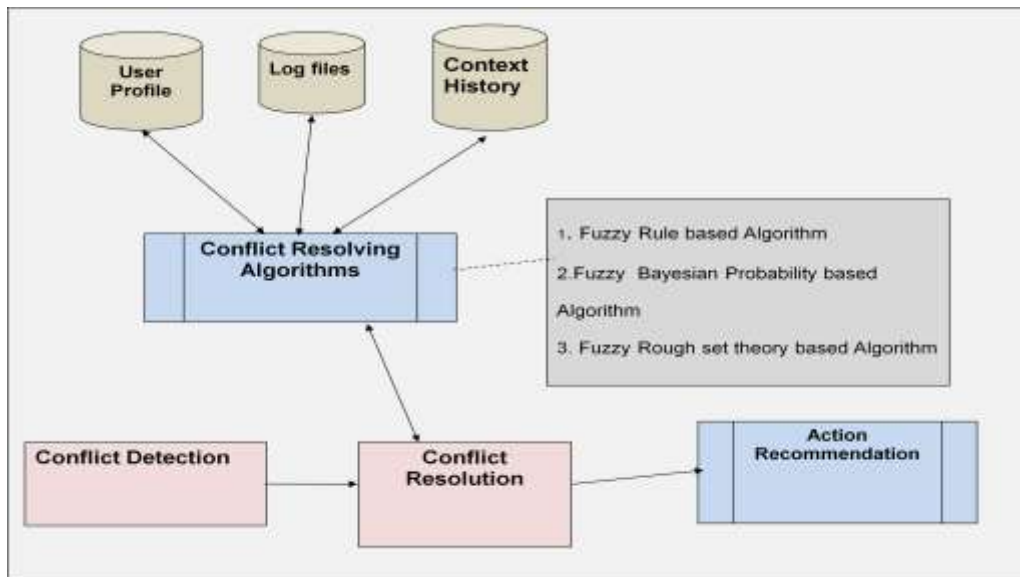


Fig2 Architecture of the proposed system

- ii. **Conflict resolution:** user conflicts arise in multi user environment, it can be resolved using following algorithms. These algorithms make use of information from context database and log files to resolve conflicts.
- Rule based Algorithm
 - Bayesian Probability based Algorithm
 - Rough set theory based Algorithm
- iii. **Action recommendation:** series of predefined actions will be recommended after the conflicts get resolved. It recommends the set of devices and their status based on user context. Figure 2 does not depict sensors and devices. This is because, they get added or deleted depending upon the current user context.

5. CONFLICT RESOLVING ALGORITHMS

Conflict resolution in context-aware computing is getting more significant attention from researchers as pervasive/ubiquitous computing environments take into account multiple users and multiple applications. In multiuser ubiquitous computing environments, conflicts among user's contexts need to be detected and resolved. Conflicts arise when multiple users try to access or try to have a control on an application. In this paper, we propose a series of algorithms to resolve conflict in multi user environment. The algorithms make use of different tools like Probability, Fuzzy Logic, Bayesian Network and Rough set theory and utilize various factors like social, personal and environmental.

5.1 Rule based Algorithm

The algorithm makes use of predefined rules to resolve the conflict as soon as some conflict is detected. These rules are stored in Rule base which are the tables in Database with each row identified by unique Rule number.

Algorithm :

Input : 1) Current Context Attributes
2) Rule Base

Output : 1) appropriate actions based on user situation
2) Fitness degree of the Rule.

Procedure :

```
for(Rule#=1; Rule#<=m; Rule#++) in a RB
  {If ( $\exists$ Rule# $\forall$  $\forall_{i=1}^n$ (Rule#Ci==CCi))
    Then return
  { fitness degree ;
  }
}
for(Rule#=1; Rule#<=m; Rule#++) in a RB
  {If ( $\exists$ Rule# $\forall$  $\forall_{i=1}^n$ (Rule#Context_id=CCContext_id))
    Then return
  { fitness degree;
  }
}
for(Rule#=1; Rule#<=m; Rule#++) in a RB
  {If ( $\exists$ Rule# $\forall$  $\forall_{i=1}^n$ (Rule#Conflict_id==CCConflict_id))
    Then return
  { fitness degree;
  }
}
```

Fig 3. Rule based Algorithm

Where,

Rule Base- Tables in Database with each row identified by unique Rule number.

Service recommendation- Device status according to user situation.

Fitness degree(FD) = No. of matching context attributes of Rule base / Total No. of context attributes Rule in base

In rule based algorithm, the context information will be obtained from simulated sensors, in the form of raw values provided by user. When this information is submitted, the algorithm searches for matching context in the rule base, if the context is found, the algorithm recommends suitable services and also returns fitness degree. If context is unavailable, then the algorithm finds the nearest context and recommends services along with fitness degree.

This algorithm is suitable in the situations like wherein the context history does not changes regularly or context database is very less in size. For example one can use rule based algorithm for ubiquitous computing devices like AC, FAN and Tube light where in the number of rules will be less irrespective of the type of users and user group.

5.2 Bayesian probability based Algorithm

This algorithm is used to resolve conflicts among family members while watching TV. In our work we have considered nuclear family consisting of four members, namely, father, mother, son and daughter. The algorithm resolves conflicts by considering factors like age factor, role factor, family dependency factor, social dominance factor and recommends suitable actions based on user situation.

We are using Bayesian probability to calculate user ratings. We have used Pearson correlation to measure the similarity between the user's interests in a family which is calculated using $\text{sim}(u,v)$. $p(u,l)$ is predicted rating if user u on item(TV program) l . Finally conflict degree is calculated as the difference between user ratings and predicted ratings. The service with least conflict degree is recommended.



Algorithm

Step1 : Retrieve the Current Context .

Step2 : For Each user
 For Each service the preference score is calculated as follows :

$$UP_{ui}(\text{Service}/\text{Context}) = \sum_{v \in C_a} P(\text{Service}|C_a)$$
 Whereas

$$P(\text{Service}/\text{Context}) = \frac{P(\frac{\text{Context}}{\text{Service}}) \times P(\text{service})}{P(\text{Context})}$$

Step3: Socially influenced ratings for each user is computed as follows

$$RF = \begin{cases} 1 & \text{for parents} \\ 0.5 & \text{for children} \end{cases}$$

$$AF = \begin{cases} 1 & \text{if } 50 < x < 100 \\ \frac{\text{age}(x)-10}{40} & \text{if } 10 < x < 50 \\ 0 & \text{otherwise} \end{cases}$$

$$FDF = \begin{cases} 1 & \text{if user is source of income} \\ 0 & \text{otherwise} \end{cases}$$

$$SDF(U_i) = (\text{Role Factor} + \text{Family Dependency Factor} + \text{AgeFactor})/3$$

Step4: For each user
 For each item

$$P(u,i) = UR_u + \frac{\sum_{v \in N_u} (R_{v,i} - R_v) * Sim(u,v)}{\sum_{v \in N_u} |Sim(u,v)|}$$

Where $Sim(u,v) = \frac{n \sum UR_1 * UR_2 - (\sum UR_1)(\sum UR_2)}{\sqrt{[(N \sum UR_1^2 - (\sum UR_1)^2)] [N \sum UR_2^2 - (\sum UR_2)^2]}}$

Step6: Conflict Degree $CD_{xi} = (UR - P_{ui})$

Step7: The Group conflict degree for a given item is calculated as
 For item = 1 to m determine the following

$$\mu_{u,item} = \frac{1}{n} \times \sum_{u=1}^{u=n} CD_{u,i}$$

Step8: Choose service with least conflicting scores and recommend services

Fig4. Bayesian probability based Algorithm

5.3 Rough Set Theory based Algorithm

Rough set theory is regarded as a mathematical tool for the analysis of imperfect data. The theory has found applications in many domains, such as decision support, engineering, environment, banking, medicine and others. In our paper this theory has been exploited as a part of conflict resolving algorithm.

This algorithm resolves conflicts with two phases.

- **Construction of Decision table:** Decision tables are a precise yet compact way to model complicated logic. It includes set of conditions and actions. Each decision corresponds to a variable, relation or predicate whose possible values are listed among the condition alternatives. Each action is a procedure or operation to perform, and the entries specify whether (or in what order) the action is to be performed for the set of condition alternatives the entry corresponds to. In our work, we are using Rosetta Tool which takes log files as input, and outputs decision table.
- **Conflict resolution and action recommendation:** In this phase, the algorithm considers the context information obtained from simulated sensors and search for matching context in decision table. If the context is found, the algorithm recommends suitable services. If context is not found, then the algorithm finds the nearest context and recommends services.

Algorithm:Phase1(Construction of Decision Table)

Input: A set of users present in Context Aware Ambient space, Users item – ratings matrix, users priority in a given group, number of neighbor k.

Output: List of services with Ascending order of Conflicting Degree.

Step1. Retrieve a Service usage Details using a *Service Log file*

Step2 : Obtain the required training sets of users service usage pattern from the Service _Log file.

Step3: Split the Training Databases into Different tables based on the high priority Context attribute. i.e Training Database = {t1,t2,t3,-----tn}

Step4: for table =1 to table =n

for i =1 to n

$IND(A\{a1,a2,a3---ai\}) = \{ (x,x') \sim S2/ _a \sim A a(x) = a(x,x') \}$.

Equivalent_Classes(a1,a2,a3,-----ai) , [X] A = IND (a1,a2,a3---ai);

end for

end for

Step5: for each Ci= [X] A

$\gamma_i(Ci,D) = |POS_c(D)|/|U|$

Step6: Determine the Red(A) // Reducts are the classes with = 1.

Step7:The Minimal reducts = CORED(A) = $\cap Red D(A)$

Step8: The Decision Table is the table with Context Attributes and Actions,

Whereas Context _Attributes = The Minimal reducts = $\cap Red D(A)$

Algorithm Phase2: (Conflict Resolution and Service Recommendation)

Step0: Get_Context()

Step1: Search suitable context in the Decision table

Step2: If (Context is found)

Return the suitable services

Else

Find the nearest context and return the settings and services

Step3: Recommend the services to the user

Fig5. Rough Set Theory based Algorithm

6. ANALYSIS AND EXPERIMENT

In order to analyze the amount of conflicts resolved by each of the three algorithms, we have given 50 random inputs for every context, selecting suitable algorithm each time to resolve conflicts in each selected context. For 50 random inputs, number of useful recommendations is recorded and following graphs are plotted based on results.

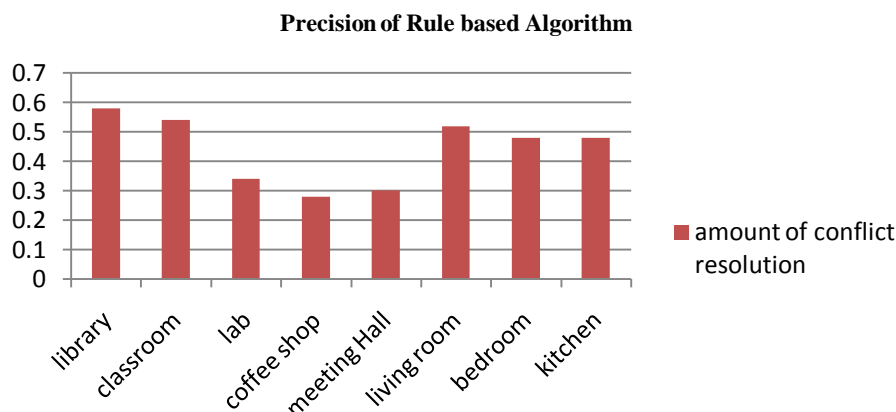


Fig6 Graphical representation of amount of conflicts resolved with rule based algorithm

The precision values for the contexts library, class room and living room lies in between 0.5 to 0.6, where as the precision values for kitchen and bedroom lies in between 0.4 to 0.5. For the contexts like coffee shop and meeting hall, the precision values are less which is in between 0.2 to 0.3 as shown in the Figure 6.

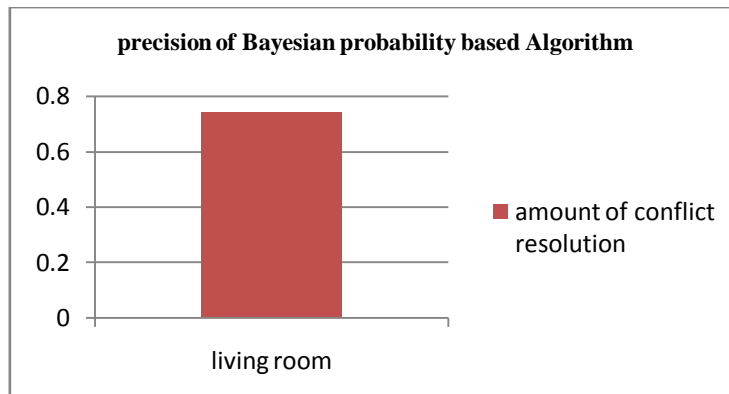


Fig7. Graphical representation of amount of conflicts resolved with Bayesian probability based Algorithm

The Bayesian probability algorithm applied to living room context reveals the graph as shown in the figure 7. Its precision value lies in between 0.6 to 0.8 which is greater compared to the values of Rule based algorithm and Rough set theory based algorithm.

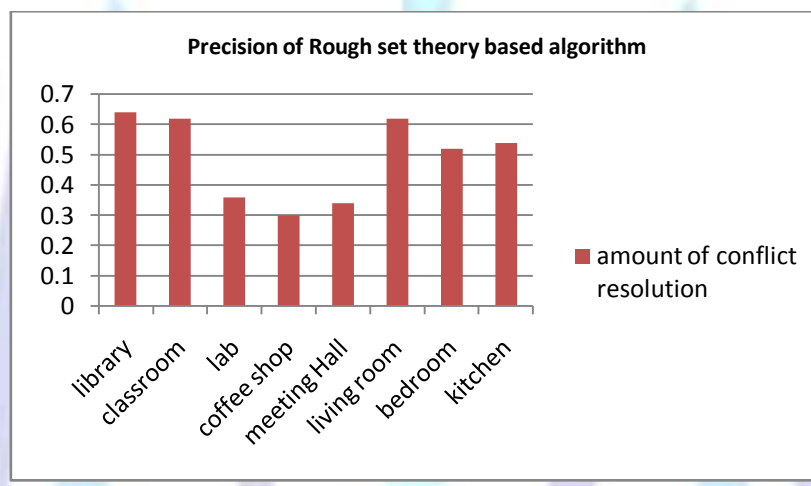


Fig 8. Graphical representation of amount of conflicts resolved with rough set theory based Algorithm

The precision values for the contexts library, class room and living room lies in between 0.6 to 0.7, where as the precision values for kitchen and bedroom lies in between 0.5 to 0.6. For the contexts like coffee shop and meeting hall and lab, the precision values are less which is in between 0.3 to 0.4 as shown in the Figure 8.

In above three graphs

X-axis= locations or context

Y-axis= precision

Precision= Ratio of number of useful recommendations/Total number of recommendations

Sample size= 50

From the above analysis, we can conclude that Bayesian probability algorithm is best because of its good conflict resolving ability.

7. RESULTS

The proposed GUI is as shown in Figure 9. It consists of simulated sensors, devices and users. User can click “File” button on the top left corner to get buttons such as user, sensors, devices, location, sub location, activity and mood. When respective button is clicked, one can get raw values that to be given to the sensors.

Example: on sub location button click, list of sub location and respective code is displayed. The sensors extract raw values, which are assigned with Fuzzy values in order to Identify the current context. Example: For light sensors, values less than 10 will be assigned with fuzzy value “dim”. For temperature sensor, values greater than 35 will be assigned with

fuzzy value “hot”. The context identified is then described with meaningful English sentences which explains who is the person in the current context? What is his activity? What is the temperature and light intensity in the current context? Then set of suitable actions are recommended based on user situation.

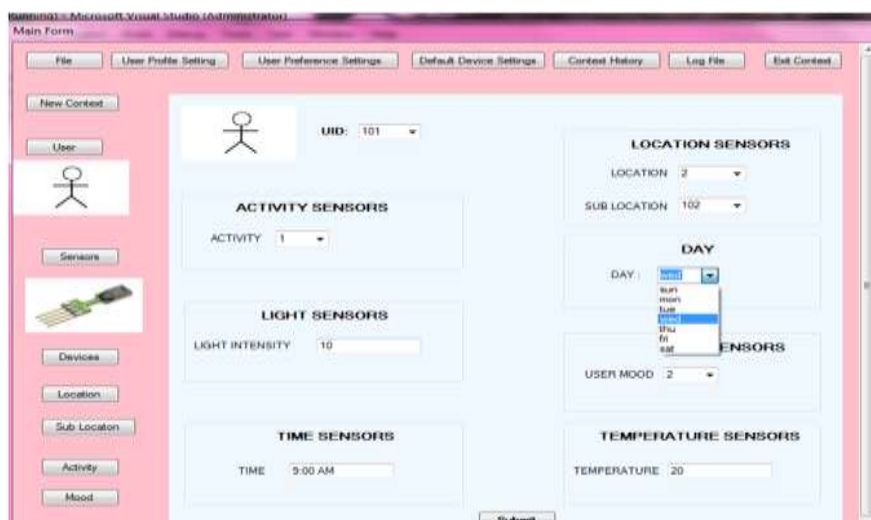


Fig 9. GUI consisting of simulated sensors, devices and users.

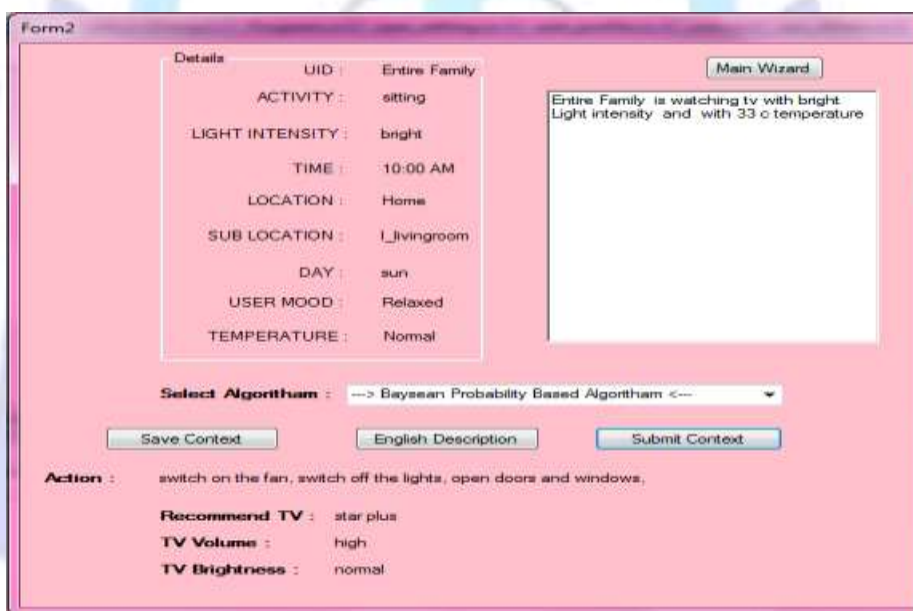


Fig10. GUI that describes current context and recommends services

Figure 10 is the result of set of device recommendations for living room context, when entire family is watching TV. TV channel, TV volume and TV brightness are recommended after resolving conflicts among the family members using Bayesian probability algorithm.

In our proposed work, we can add more devices, more number of users, or remove existing users and devices. Our system is flexible to accept new context information and recommend services.

8. CONCLUSION

The paper is proposing a simulated version of generic prototype based on GUI widgets, database and conflict resolving algorithms, which can be used for developing and prototyping context aware applications in an effective manner. The proposed system is used to design and explain the concept of user context and context aware applications. Context description in English for a given sensor values is automatically generated. The analysis of algorithm shows that Bayesian probability based algorithm with respect to recommendation accuracy is best. The System can also be used as a test bed for newly designed conflict resolving algorithms.

9. REFERENCES

- [1] Rough Set Theory Based User Aware TV Program and Settings Recommender By Prof. Thyagaraju.G.S, U.P.Kulkarni. S.D.M College of Engineering and Technology, Dharwad, Karnataka, India.



- [2] Design and Implementation of Prototyping Simulator of Context Aware Applications By Thyagaraju.G.S, U.P.Kulkarni, S.D.M.C.E.T. Research Centre, VTU, Dharwad, Karnataka, India
- [3] Algorithms to Resolve Conflict in Multiuser Context Aware Ubiquitous Environment By Thyagaraju.G.S, U.P.Kulkarni, S.D.M.C.E.T. Research Centre, VTU, Dharwad, Karnataka, India
- [4] Intro to Context-Aware Computing by Matthew Lee
- [5] Context-aware computing applications by Bill schilit, Norman Adams and Roy Want.
- [6] Context-awareness in wearable and ubiquitous computing by Gregory D Abowd, Anind dey, Robert orr and Jason Brotherton.
- [7] survey on context aware system by Matthias Baldauf
- [8] A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications Gregory D Abowd and Anind dey
- [9] SenSay: A Context-Aware Mobile Phone by Daniel Siewiorek, Asim Smailagic, Junichi Furukawa, Neema Moraveji, Kathryn Reiger, and Jeremy Shaffer.
- [10] CA4IOT: Context Awareness for Internet of Things by Charith Perera y, Arkady Zaslavskyy, Peter Christen and Dimitrios Georgakopoulos
- [11] Torben Weis eta al., "Rapid Prototyping for Pervasive Applications", IEEE Pervasive Computing, vol. 6,no. 2, pp. 76-84, April-June 2007, doi:10.1109/MPRV.2007.41
- [12] C. Becker et al., "PCOM—A Component System for Pervasive Computing," Proc. 2nd Int'l Conf. Pervasive Computing and Comm. (PerCom 04),IEEE CS Press, 2004, pp. 67–77.
- [13] J. Hill et al., "System Architecture Directions for Networked Sensors," SIGPLAN Notes, vol. 35, no. 11, 2000, pp. 93–104.
- [14] "Playing with the Bits" User-configuration of Ubiquitous Domestic Environments Jan Humble*, Andy Crabtree, Terry Hemmings , Karl-Petter Åkesson*, Boriana Koleva, Tom Rodden, Pär Hansson*
- [15] UBIWISE, A Simulator for Ubiquitous Computing Systems Design John J. Barton, Vikram Vijayaraghavan Mobile and Media Systems Laboratory HP Laboratories Palo Alto HPL-2003-93 April 29th , 2003*
- [16] Content-based Recommender Systems: State of the Art and Trends by Pasquale Lops, Marco de Gemmis and Giovanni Semeraro.
- [17] Content-Boosted Collaborative Filtering for Improved Recommendations by Prem Melville and Raymond J. Mooney and Ramadass Nagarajan.
- [18] Rough set theory and its applications by Zdzisław Pawlak.
- [19] An Introduction to Rough Set Theory and Its Applications A tutorial by Zbigniew Suraj
- [20] Rapid Prototyping for Pervasive Applications by Torben Weis and Mirko Knoll Andreas Ulbrich and Gero Mühl Alexander Brändle
- [21] A Framework for Programming Robust Context-Aware Applications by Devdatta Kulkarni and Anand Tripathi, Fellow, IEEE
- [22] A tutorial-based guide to the ROSETTA system: A Rough Set Toolkit for Analysis of Data by Torgeir R. Hvidsten.
- [23] Comparing the performances of Neural network and Rough set Theory to reflect the Improvement of prognostic in medical data By Nur Aniza Bt Alang Ismail
- [24] Rough Set- A Tutorial by Jan Komorowski, Lech Polkowski , Andrzej Skowron
- [25] ROSETTA Technical Reference Manual by Aleksander Øhrn.

Authors biography with Photo

Author name: Thyagaraju.GS



Received the M.Tech Degree in Computer Science And Technology From University Of Mysore ,India in 2002.He has got ten years of experience in academics, four years of Research Experience . He is a member of IETE. He has guided many students at UG and PG level. He is pursuing Ph.D in Computer Science Engineering. His Research Interests are Context Aware Computing in Ubiquitous and Intelligent Systems He is now working as a Senior Lecturer in Dept Of CSE, SDM College Of Engineering, Karnataka, Dharwad. He can be reached at thyagaraju_gs@yahoo.co.in.

Co-Author name: Poornima.D



Secured B.E degree from SLN college of Engineering, Raichur. Now Studying M.tech 4th semester in SDM Engineering college, Dharwad under visvesvaraya technological university, Belgaum Karnataka, India. She is interested in Artificial Intelligence concepts. She can be reached at poornima.udaykumar@gmail.com