Socially Interactive CDSS for u-Life Care

Iram Fatima
Ubiquitous Computing Lab, Kyung Hee University, Korea
iram.fatima@oslab.khu.ac.kr

Muhammad Fahim
Ubiquitous Computing Lab, Kyung Hee University, Korea
fahim@oslab.khu.ac.kr

Donghai Guan
Ubiquitous Computing Lab, Kyung Hee University, Korea
donghai@oslab.khu.ac.kr

Young-Koo Lee
Ubiquius Computing Lab, Kyung Hee University, Korea
yklee@khu.ac.kr

Sungyoung Lee
Ubiquitous Computing Lab, Kyung Hee University, Korea
sylee@oslab.khu.ac.kr

ABSTRACT
Clinical decision support system (CDSS) is an interactive decision support system computer software, which is designed to assist physicians and other health professionals with decision making tasks, such as determining diagnosis of patient data, disease prevention, and alerting adverse drug events. It links health observations with health knowledge to influence health choices by clinicians for improved health care. Different from conventional CDSSs which focus on diagnosis assistance, the focus of our CDSS is to provide recommendations and healthcare services for chronic disease patients by long term monitoring. To make our CDSS more intelligent, it induces the patients to interact with the system. By continuously learning and digesting patients' experience and knowledge, the knowledge base of our CDSS is self-evolutionary and dynamically enhanced. We mainly develop two modules to achieve the function of social interaction. Firstly, Knowledge Authority Module (KAM) is developed which is capable of manipulating and preprocessing social data. Secondly, to support self-evolutionary and dynamical learning, we designed the rough set based inference engine. Through social interaction, the patients can get continuous relevant medical recommendations from the system, so they can get a chance to improve their health conditions which in turns keeping on their quality of life.

General Terms
Algorithms, Management, Performance, Design, Reliability, Human Factors

Keywords
CDSS, Social Interaction, Self-evolutionary Knowledgebase, Knowledge Authority, Inference Engine, Rough Set.

1. INTRODUCTION
Since 50 years ago, researchers have recognized that computers may soon play an important role in medical decision making. In 1961, Warner et al. proposed a mathematical equation to aid in the diagnosis of congenital heart disease. Following naturally from this early work, many CDSS applications have been developed. The most common applications of a CDSS include alerts and reminders, diagnostic assistance, prescription decision support, information retrieval, image recognition and interpretation, and therapy critiquing and planning [1-2].

ICUIMC 2010, February 21-23, 2010, Suwon, South Korea.

Although the applications of clinical decision support system are diverse and have different goals, most of them focus on providing diagnostic assistance to medical doctors and healthcare professionals. This can be reflected by existing works [3-5]. Over the years, our research group consistently works on providing healthcare services for the elderly patients who suffer from chronic disease. In this paper, we want to achieve this goal by developing a new CDSS system.

Firstly, our CDSS targeted users are the patients who suffer from chronic diseases. Specially, we focus on the manageable chronic diseases like diabetes and stroke. With enough care and supervision, usually these patients' health condition could be improved. This focus is important because these diseases are among the most common, costly, and preventable of all health problems in the world.

The second focus is to provide recommendations or suggestions to the patients rather than providing diagnostic assistance for the medical doctors. Here the recommendations include many different types such as medicine, exercise, and entertainment. We aim to improve the patients' health by suggesting them appropriate life patterns and styles. For example, after observing a patient for one month, our CDSS finds some problems with this patient's life pattern. He/she usually sleeps late; does not exercise regularly; does not take medicine on time; eats too much. Obviously, these life styles are not good for rehabilitation. Our CDSS can recommend the patients to change these harmful life patterns.

With above goals, our CDSS has been designed with several novel ideas. Firstly, it is socially interactive. Here socially interactive means that our CDSS can interact with users and automatically learn new knowledge from users' experience. Thus, the knowledge base of our CDSS is dynamic and self evolutionary. This is different with conventional CDSSs which usually utilize static knowledge base and neglect the feedbacks from the patients.

Secondly, it supports user high-level context recognition ability. Most traditional CDSSs need patients or doctors to manually input patient-related information, which is then used for decision making. Generally CDSS tends to generate more intelligent and patient-oriented decision when more patient related information obtained. To monitor a patient's condition and give suggestions/alerts, it is not efficient and realistic to frequently ask the patient to manually input his/her information. Therefore, currently researches begin to combine CDSS with smart home where various sensors are deployed, such as biosensors and ACM 978-1-4503-0571-6.. $10.00 techniques, CDSS can automatically get these information, thus saves user's effort. The problem of exiting CDSSs is lack of high-level user context recognition ability. Here the raw sensor information is regarded as low-level user context information. High-level user context information is derived from low-level information. User activity, location, emotion, speech are typical high-level user context information. For many cases, high-level user context information is more useful for CDSS. For example, for a diabetes patient, his/her activity information is far more important and semantic than the raw low-level sensor information.

Thirdly, integrating with cloud computing. The use of cloud computing architecture helps in eliminating the time and effort needed to roll a life care services [6, 7]. By pooling the various life care IT resources into clouds, hospitals can reduce the cost and increase utilization as the resources are delivered only, when they are required [8]. The cloud based CDSS, provide a flexible platform for patients and doctor's experience to build a continuously evolving knowledge base. The high level architecture of the proposed socially interactive cloud based CDSS is shown in Figure 1.

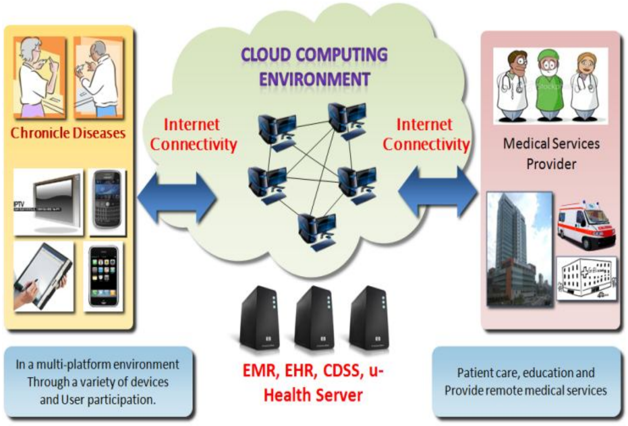

Figure 1. High-Level System Architecture

In our previous work [9], technically we have got the solutions for high-level user context recognition and cloud support. Thus, in this paper, we will only focus on the function of socially interactive of CDSS.

In this research, we mainly developed two distinguished components in our socially interactive CDSS. Firstly, we developed the knowledge authority module (KAM) which can capture and manipulate data from the society. It can process both well-formatted data such as EMR, paper charts [10] and unformatted input data like email and short messages.

In addition, since input data might be incorrect, this component supports data cleaning function and provides reliable data to the inference engine. Secondly, to efficiently handle dynamic data from user feedback, we designed the rough set based reasoning engine with the goal of providing stable and incremental learning ability. With the support of the rough set engine, dynamic rules can be automatically generated with less effort from experts.

We structured our paper as followings: Section II presents existing approaches for CDSS. Section III presents our socially interactive CDSS design. Section IV, covers the case study and finally conclusion and future work are drawn in Section V.

2. RELATED WORK

The authors of [7] developed the health care server using the clinical decision support system and ontology to manage user health data. The major functions are to generate, deploy and manage the user profile. They analyzed the stored health data using the knowledge base like CDSS, ontology and then notified data result. Three u-health care service scenarios are discussed like diagnosis service, remote monitoring service and emergency management service. Mobile systems are used to analyze and transfer user raw data to health care server.

The authors of [11] implemented independent knowledge service and inference mechanism with EHR service or existing hospital information system. They integrated the rule engine and applied knowledge base for executing clinical guideline. The architecture has four components, knowledge authority environment, knowledge base, knowledge engine, and inference server. To validate the architecture, they applied it to the hypertension scenarios. The drawback of their technique is to model the information according the input of knowledge base.

The authors of [12] proposed independent inference engine which separated the knowledge reasoning from knowledge representation. Inference engine separation made it superior then other approaches. To process the information of medical models, XML format is required according to the specification of their approach. The limitation of this approach is that, it cannot accommodate every medical model due to its diversity.

The authors of [13] proposed a practical guideline based clinical decision support system for metabolism synthesis. First they established the medical logic model and then converted it in to the rule set for inference. The basic structure is composed of four parts named data part, model part, inference part, human computer interaction part. The data part may be database system, the inference part is composed of knowledge base or rule set and reasoning machine. Human computer interaction is the user interface which received and verified the user request for clinical decision making. This approach is inefficient to handle unstructured information and needed to enter the information manually by the doctor.

The authors of [14] proposed a network base CDSS to show the holistic picture of patient profile for effective decision making. It comprised on clinical database and web-based application. Their approach combined different system analysis like eye fundus image analysis, cardiology and ophthalmology. They focused on the integration of clinical data processing over the network and make it available for the physicians and researchers. This CDSS is only fruitful for doctors as end users.

While many research groups are developing clinical decision support systems (CDSS), they are lacking of social interaction support and cloud based services. For existing CDSSs, only the domain experts contribute to build and supply the knowledge base. In addition to domain experts, we believe that social interaction is the other important information sources of knowledge base. By utilizing social input information, the knowledge base of CDSS is self evolutionary and augmented, consequently, CDSS is more realistic and intelligent which is able to provide user customized services on cloud.

3. SOCIALLY INTERACTIVE CDSS DESIGN

The proposed design of CDSS is mainly composed by four parts, named database tier, user centric privacy and trust framework, KAM and rough set based inference engine. Figure 2 shows the detailed design of the proposed system. Database tier includes three databases; socially evolving database, dynamic CDSS database and static CDSS database. Socially evolving database is used to store the user provided structure and unstructured information without any processing. Dynamic CDSS database is evolutionary database that stores the processed information through KAM module. It actually facilitates the rough set based inference engine to automatically generate the rules. In existing CDSS, Electronic Health Record (EHR) stores information like patient profile, history and laboratory test. This EHR works as a static CDSS database in our system.

Privacy is the biggest hurdle, users are much conscious about their medical data and don't want to share their personal information, until firm assurance is provided to them about data sharing policies. To achieve this firmness and to provide peace of mind to the patients about their medical records, we have used the notion of user centric privacy & trust framework.

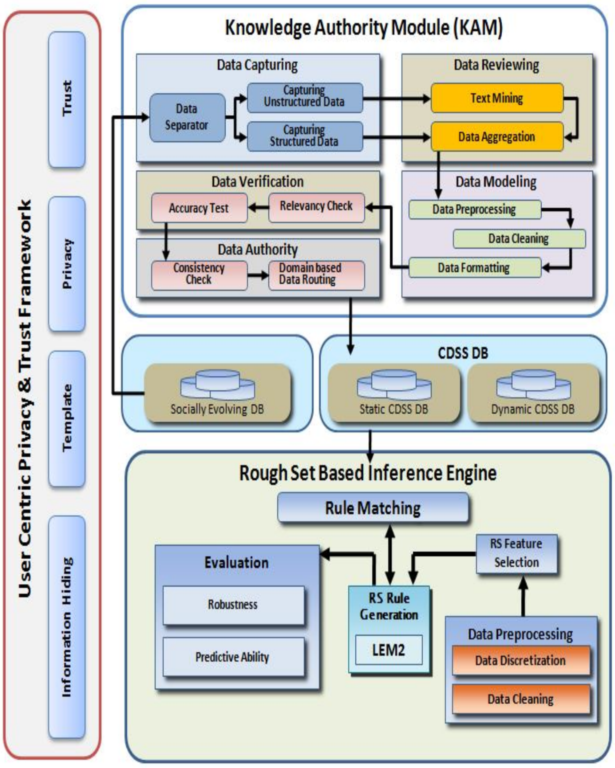

Figure 2. Detailed architecture of the system

To ensure that information provided by the patient is correct, degree of belief / trust is calculated on cloud [15], to assist doctors in deciding which information should be used in decision making process. The primary contribution of this paper is to involve the social interaction with CDSS to make it more realistic and intelligent. To accomplish this, our main focus is to introduce KAM module and rough set based inference engine.

3.1 Knowledge Authority Module (KAM)

Social data acquisition is a very important in construction of dynamic data repository for our CDSS. The first step of social data acquisition is to select the targeted clinical area (i.e. diabetes) and select expert clinicians to gain domain specific knowledge. The next step is to transfer the knowledge into computer interpretable form on the designed data representation schemes. The main focus is on the review of social data acquisition to acquire clinical domain knowledge for intelligent decision making. The KAM captures user feedback, converts it in the system understandable form, verifies its integrity and relevancy, and finally stores it in a dynamic CDSS database. This module provides an open platform to acquire / share clinical knowledge from society which can reflect user experiences in clinical knowledge management.

3.1.1 Data Capturing

To offer a high quality, fast, and general data capturing service, the data provided by the user is based on two types: structured and unstructured. The part of the data captured through pre-formatted input forms are structured data, while user provide unstructured information without any data entry form or guidance. These two types of data are required to be separated for efficient processing of information and extracting knowledge.

Data separator module separates the structured and unstructured data. Once data is separated they are distributed among capturing unstructured data and capturing structured data modules. Former module stores the unstructured data temporarily in a standard fast-accessed data storing device. In other words, it acts as an input data buffer for the text mining module. In order to avoid data loss, the captured structured data are also necessary to be stored in a temporary storage.

Later module, acts as the underlying data repository for the data to be aggregated with the result from text mining module as well as structured data. The algorithm used in this module is shown in Algorithm 1.

Input: ***S_{DB}:*** *user data from socially evolving DB,*
D_K: *domain knowledge,* ***D_F:*** *data formatting guide.*
data_caputuring *(S_{DB}, D_K, D_F)*
begin
  resultSet *← "null";* ***totalSet*** *← "null";*
  while *(not end of SDB)*
    if *(the record is structured data)*
      S_{sdb} ←SDB;
      store ***S_{sdb}*** *in structured data storage;*
    else
      U_{sdb} ← SDB;
      store ***U_{sdb}*** *in unstructured data storage;*
    end if
  end while
end

Algorithm 1. Data capturing.

3.1.2 Data Reviewing Module

The objective of this module is to provide a summarized view of the user feedback. We apply efficient text mining technique, to extract clinical knowledge and information from the unstructured data.

aggregate_data*(rS, S_{sdb})*
begin
  tS *← "null";*
  align *rS;*
  tS *← $rS + S_{sdb}$;*
  return *tS;*
end

Algorithm 2(a). Data aggregation.

The technique is highly supervised and influenced by the knowledge of the domain expert. In proposed approach semantic keyphrases are filtered out according to the available relations that exist in feedback. Level based semantic keyphrase extraction results are more accurate as compare to the existing algorithms. The algorithm used for data aggregation and text mining module is shown in Algorithm 2(a), and 2(b), receptively.

```
TrainLevel ← set TrainLevel
resultSet [] ← returned keyphrases according to TrainLevel
for resultSet[] <> empty
  | if (resultSet(training level)) then
  |  | if (keyphrase level = lower level keyphrases)
  |  |  |     processSet[] = preserving lower level keyphrases
  |  | else
  |  |  |     processSet[] ← identifying and preserving training level
  |  |  |                    equivalent
  |  |  |    processSet[] ← remove redundant keyphrases
  |  |  |    refineSet[] ← processSet[]
  |  | end if
  | elseif (keyphrase level = training level)
  |  |  |    refineSet[] ← processSet[]
  |  | else if (keyphrase level = upper level)
  |  |  |  |        processSet[] ← identifying and preserving training
  |  |  |  |        level equivalent  keyphrases
  |  |  |  else
  |  |  |  | processSet[] ← stemming lower level general keyphrases
  |  |  |  | processSet[] ← remove redundant keyphrases
  |  |  |  | refineSet[] ← processSet[]
  |  |  | end  if
  |  | end if
  | end  if
end for
return refineSet[]
```

Algorithm 2(b). Text mining.

3.1.3 Data Modeling Module

This module is responsible to model the data as it might be possible that user provides same information through the structured and the unstructured method. Identifying and eliminating such redundancy from data increases the data processing efficiency and accuracy. There may be a set of limits for specific data defined by the system. Accommodating the data within the preferred limits is another major part of the tasks of this module.

There may be different type of errors in the data provided by the user; data cleaning is an essential step in populating and maintaining data. Such types of errors are required to be defined. The preprocessed data then is to be checked of determining whether there exist any of such errors. Once errors are identified exact error correction measures are taken to correct them. At the same time errors instances, types, and the correction measures are also logged for future references. Since the user might not be expert in data inputting, the data may contain significant amount of noise. Removing such noise from data is another major task of the module. The algorithm used for this module is shown in Algorithm 3.

3.1.4 Data Verification Module

Data verification is the process where data is checked for accuracy and relevancy after data formatting is done [16]. This module is responsible to apply clinical domain knowledge for ensuring the completeness, relevancy, accuracy, and validity of the data. Even though the data is relevant to the addressed issue, it might not be as complete as required for further processing. Hence accuracy test is performed based on level of confidence appropriately set by the clinical domain expert to maximize the accuracy. Algorithm 4 shows the underline process of this module.

```
model_data(tS, D_F)
  begin
   | remove redundant data from tS;
   | for (each data element di in tS)
   |  |  if (di crosses the limit specified in D_F)
   |  |   |  accommodate  di within the limit;
   |  |  end if
   | end for
   | check error in tS based on D_F;
   | remove noise from tS based on D_F;
   | if (tS is unformatted based on D_F;)
   |  | format tS;
   | end if
  end
```

Algorithm 3. Data Modeling

```
verify_data(tS, D_k)
 begin
   | confLevel ← a fixed confidence recognition set based on D_k;
   | identify irrelevant information in tS based on confLevel;
   | report irrelevant information;
   | if (user is not satisfied)
   |  |     lower confLevel;
   | end if
   | remove irrelevant information from tS;
   | if (tS contains any incompleteness data based on D_k)
   |  | report incomplete data;
   |  | complete tS;
   | end if
   | if (tS contains any invalid data based on D_k)
   |  | report invalid data;
   |  | validate tS;
   | end if
 end
```

Algorithm 4. Data Verification

3.1.5 Data Autohority Module

The accurate data obtained from previous module still may suffer from a number of limitations. For example, the data may consist of information which is not at all reasonable keeping in mind the addressed issues and/or the person providing it. This module can identify the reasonableness of the captured information. It is also necessary to test whether the data produced by the knowledge authority module is compatible against the next interface part of the system. A predefined compliance check is, therefore, performed on data before routing it to the next module. The processed data is stored in dynamically evolving database for the CDSS. Algorithm 5 depicts the functionality of this module.

The core of our CDSS is a knowledge authority module which could capture and manipulate data from the society. One of the important features of this module is that it can extract useful knowledge from not only the structured user input data but also the unstructured form of information that user shares with the system. This important flexibility of the KAM is facilitated by a data reviewing module that uses the proposed text mining technique to extract useful information from unstructured data. Hence, hierarchical level-based multi-stage information processing mechanism is the key uniqueness of the text mining approach.

```
authorize_data(tS, D_K)
  begin
    for (each each data element di in tS)
        if (di exceeds reasonable level of data value based on D_K)
            report di as unreasonable data in tS;
            remove di from tS;
        end if
        if (di is not compliant to CDSS DB ||  di is not compliant to
            Inference Engine)
            report di as incompatible data in tS;
            modify di based on D_K to make it compliant;
        end if
    end for
  end
```

Algorithm 5. Data Authority

Since the proposed algorithm focuses more on accuracy rather than efficiency in extracting effective knowledge from user feedback, it will be significantly useful to researchers and academicians. Because social data (e.g., patients experience on a particular diseases or diagnostic) are more sensitive to privacy, sharable to the society, and unstructured in nature, medical data are expected to be as specific as possible. The use of the accuracy concerned-based proposed approach may contribute in developing more realistic system for such domains.

3.2 Rough Set Based Inference Enginge

Traditionally inference engine mainly deals with relative static data. While, our proposed inference engine utilizes the social feedback (i.e. patient). Therefore, we need an inference engine which can efficiently handle dynamic data. Machine learning reasoning is required since we cannot frequently ask the experts to analyze the data which is dynamically added. Rules are automatically generated on the behalf of social feedback; this new rule set becomes the part of inference engine. Domain experts can validate the knowledge and remove some unreasonable rules as user's provided data is not very reliable due to their knowledge limitation in medical area.

3.2.1 Data Preporcessing

The output of the KAM module is stored in socially dynamic database. To process this information in our inference engine needs a proper format for processing. In this step we descritized the information on the behalf of our proposed algorithm input. At this stage we have a tremendous amount of data, so if some information is missing then replace it with *"don't care terms"* and its impact on the rule generation is negligible.

3.2.2 Rough Set Feature Selection and Rule Generation

In our social dynamic database, a lot of attributes or redundant key phrases are generated during text mining phase. These are identified and accurate by KAM module but some of them are unnecessary for rule discovery. So these attributes are discarded by selecting the feature set according to our proposed feature selection and rule generation algorithm. The algorithm for this module is shown in Algorithm 6.

3.2.3 Rule Evaluation and Matching

Rough set does not require any prior information which is normally required by other machine learning algorithms [16]. The evaluation of rules is based on the confidence value how much they obtained during rule extraction phase. In next step these newly extracted rules are matched with the existing rules. If some new rule which is more generalized then the existing one or redundant one can be reveal by the system. These rules are in human understandable format. Domain expert approved this new rule set to become the part of inference engine by analyzing it on the criteria of confidence and his own expertise.

3.2.4 Robust Feature Selection and Incremental Rule Generation

Although there are many algorithms available for rough set feature selection and rule generation, they are still suffered from some limitations. Firstly most existing rough set feature selection methods only focus on classification performance. Robust is a newly recognized problem in feature selection area. When the training data used for feature selection slightly changed, robust feature selection algorithm will output the similar results. However, a feature selection algorithm which is not robust will generate quite different feature subsets.

Input: $DT = < U, C \cup D >$ *core, n(# of ensemble classifiers)*
Output: *reduced feature set*
begin
    $E \leftarrow$ *EnsebleLearning (U,n)*
    $Core \leftarrow CoreSearch(E)$
    for *(each attribute in feature set)*
        $B \leftarrow C - core$
        $B_j^* \leftarrow sorted\ B_j\ in\ the\ order\ of\ assending\ g(b),$
        where $|POS_A(U|D)|/|U| + |U/\{a\}|/|U|$
        $C_j^* \leftarrow B_j^* + core$
    for $i=1$ *to* $|B_j^*|$
        $SIG(b_i, C, D) = |POS_{Cj}*(D)|/|U| - |POS_{cj-ba}(D)|/|U|$
        if $SIG(b_i C,D)$
            $C_j^* \leftarrow C_j^* - \{b_i\}$
        end if
    end for
    end for
    $reduct = WeightFunction(C^*)$
end

Algorithm 6. Attribute Reduction

In our CDSS design, the output of feature selection algorithm is used for rule generation, which in turn checked by the domain experts. If the feature selection algorithm is not robust, then the rules generated are also inconsistent. Consequently, domain experts will not trust in our reasoning engine. For this reason, both accuracy and robustness have to be considered. One straightforward idea is to combine ensemble learning with feature selection. By fusing the feature subsets which are obtained from different ensemble members, the features selected are expected to be more robust.

The main limitation of existing rule generation methods in rough set is lack of incremental learning support. The incremental technique is a way to solve the issue of added-in data without re-implementing the original algorithm in a dynamic database. It often occurs in using the RS theory that there are millions of data records, and the number of records increases dynamically in the database. To obtain new decision rules from the changed data set obviously consumes a huge amount computation time and memory space, and therefore the efficiency of these algorithms is very low. An efficient incremental rough set based approach is required. In this proposal, we adopt the following traditional method: deal with the new added data set by using the same reduction algorithm, and merge these new rules obtained from the incremental data set with those existing rules extracted from the original data set [17].

3.3 Workflow of inter-intra components

To achieve the objective of system integration and smooth functioning of overall system, input to each component, its internal processing, and its output needs to be formally defined. Integration process of inter/intra components for each component needs to be formalized. To specify the interfaces among all the components that which modules are available as public for the other modules and which are remain private. Specification of hooks for available interfaces and specification of configuration parameters to make these modules communicate with one another. Specification of information exchange format is a kind of requirement to fulfill before integrating the overall system. The reason is that different components have different functional flow but contributes to the same end objective. These components are working on data in different format, to cope this issue a prior specification of information exchange format is necessary.

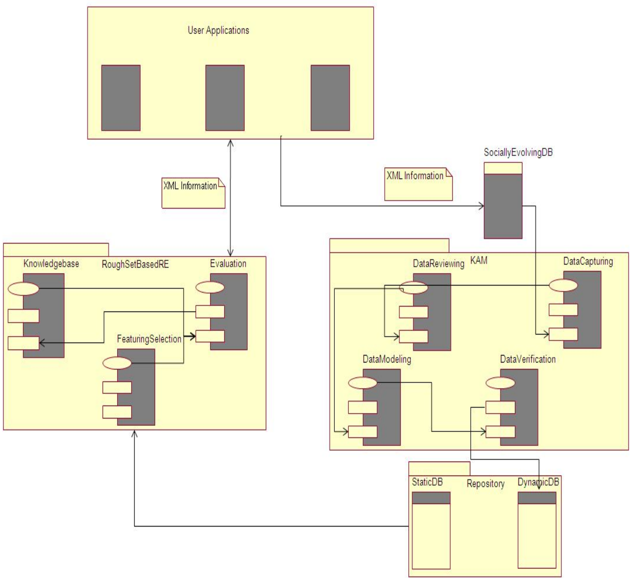

Figure 3. Work flow on inter-intra components

All the components like, KAM, and Rough Set based inference engine, as shown in Figure. 2, are tightly integrated together to achieve the end objective of proposed CDSS. For robustness and accuracy in integration with other components data repository is required by almost all the components. The basic restriction on all of them is not to change the structure of the repository but only have particular credentials for manipulation of user information. Figure 3 shows the activation of different components and input-output flow.

4. CASE STUDY OF IMPLEMENTED SYSTEM DESIGN

The view of main windows of the prototype system is presented in Figure 4 to 8. This system is developed in the lab environment using diabetes case studies. The home page of the system provides the interface to interact with our socially interactive cloud based CDSS.

It is based on the user privileges, the user group is categorized in to three groups; patients, doctors and domain experts as shown in Figure 4. Patient can enter his/her medical information as well as personal experience. Patients have emancipated to provide structured information and their personal experience in unstructured format as shown into Figure 5.

Internally this provided information is processed by our KAM and rough set module, respectively. On the basis of provided information our CDSS recommends appropriate life style and medications to the user. Patients and ordinary users can search others profile along their personal experiences.

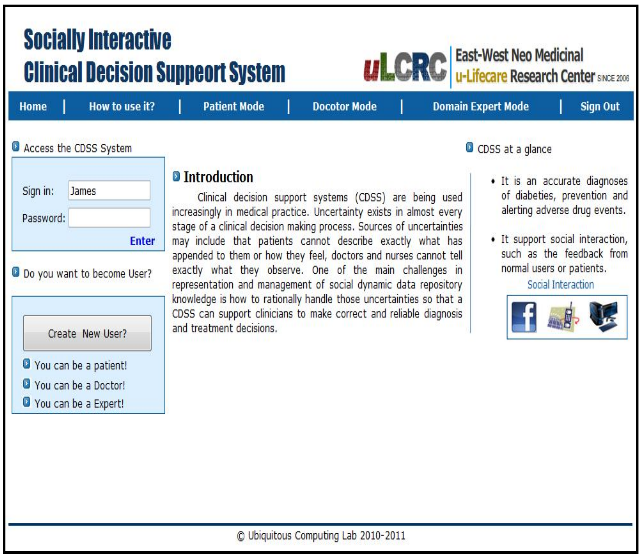

Figure 4. Home page of Socially Interactive CDSS

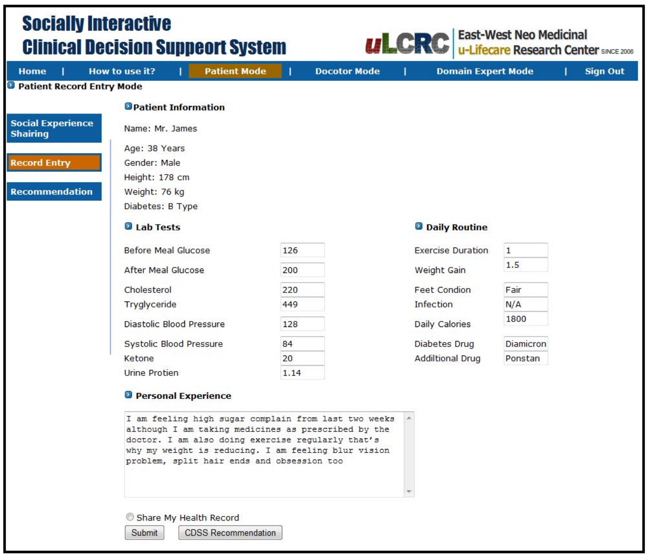

Figure 5. Entry form for structured – unstructured data

Doctors use the system to view the CDSS recommendation as well as to prescribe new medication according to the current situation on the basis of their knowledge as shown in Figure 6.

If the patient needs more care then doctors can generate the alerts for the specific patient to communicate on smart phones. This new recommendations are also store in the static CDSS database to update the user profile. The third group who can interact with our CDSS is domain experts.

Domain experts can be doctors, they can monitor and analyze the newly automatic generated rule set. These rule set is generated on the basis of patient's feedback. They are automatically cross checked with the existing one and automatically resolved the conflict issues like duplication, generalization. Domain experts can add, update and delete the generated rules according to their domain knowledge as shown in Figure 7. Our socially interactive cloud based CDSS is self evolutionary. With the passage of time, social interaction makes system more intelligent and realistic.

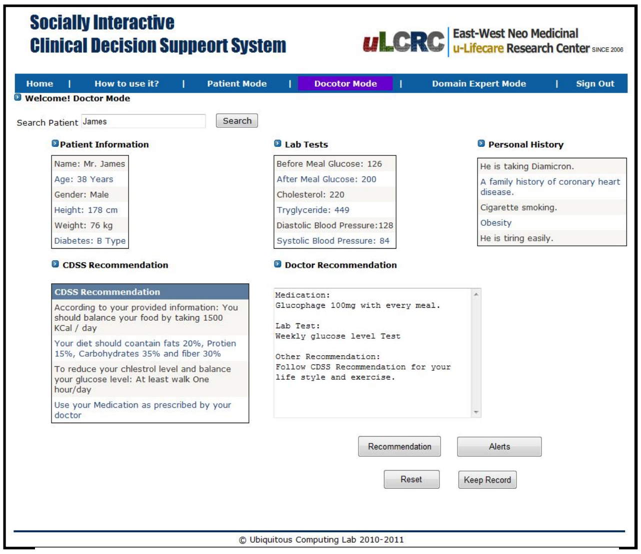

Figure 6. Doctor Recommendations

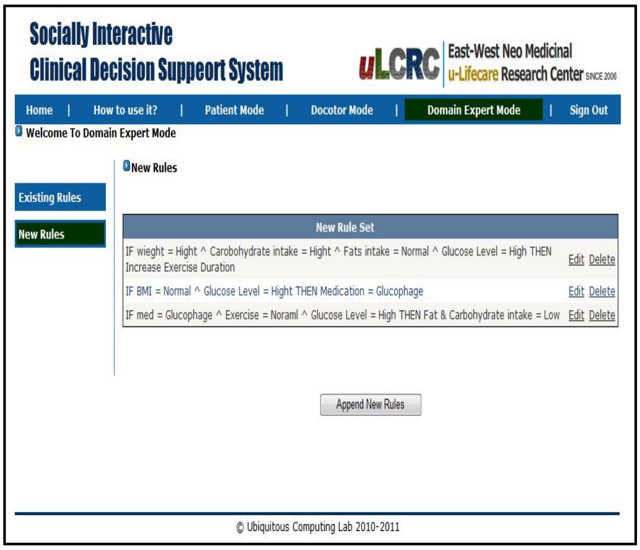

Figure 7. Autogenerated rules

Rules generated by the rough set based inference engine gradually become more generalized and compact. The whole process is automatic and very less intervention of the domain experts is needed during information processing, rules generation and final recommendations.

5. CONCLUSION AND FUTURE WORK

Socially interactive CDSS has been presented in this paper. Its goal is to provide more realistic and intelligent recommendations and health care services for the users.

We mainly proposed two modules to achieve the function of socially interactive, including knowledge authority module and rough set based inference engine. Knowledge authority module is designed for intelligent processing of structured and unstructured information. It converts the social feedback into system understandable form, for dynamically evolving CDSS database, after verifying its integrity and relevancy. Rough set based inference engine generates rules automatically to efficiently handle dynamic data from user feedback without frequent intervention of domain experts. Our CDSS integrate knowledge from clinical domain expert and from experience of users to enhance the decision making capability by manipulating dynamic social information.

In future, we have planned to integrate our socially interactive CDSS system with our Secured WSN-integrated Cloud Computing for u-life care (SC^3) [9] environment. The cloud environment and user high level context recognition (such as activity recognition and location awareness) solutions in SC^3 are expected to make our CDSS more intelligent, more cost-effective, and more easy to access.

6. ACKNOWLEDGMENTS

This research was supported by the MKE (The Ministry of Knowledge Economy), Korea under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National Industry Promotion Agency) (NIPA-2010-(C1090-1021-0003)).

7. REFERENCES

[1] Osheroff JA. 2009. Improving medication use and outcomes with clinical decision support: a step-by-step guide. The Healthcare Information and Management Systems Society. IL: Chicago

[2] Subramanian S., Hoover S., Gilman B., Field T. S., Mutter R., Gurwitz J. H. , 2007. Computerized Physician Order Entry with Clinical Decision Support in Long-Term Care Facilities: Costs and Benefits to Stakeholders. Jornal of American Geriatrics Society, pp. 1451-1457

[3] Eslami S., Bu-Hanna A., Keizer De. 2007. Evaluation of outpatient computerized physician medication order entry systems: a systematic review. Journal of the American Medical Information Association.

[4] 2009. Draft roadmap for expansion of health IT certification. Certification Commission for Healthcare Information Technology.

[5] 2009. Healthcare Information Technology Standards Panel (HITSP). Healthcare Information Technology Standards Panel

[6] Khattak A. M., Pervez Z., Ho K. K., Lee S., and Lee Y. K. 2010. Intelligent Manipulation of Human Activities using Cloud Computing for u-Life Care. The 10th Annual International Symposium on Applications and the Internet(SAINT 2010), Seoul, Korea.

[7] Ko E. J., Lee H. J., and Lee J. W. 2007. Ontology and CDSS based Intelligent Health Data Management in HealthCare Server, world Academy of Science, Engineering and Technology.

[8] Buyya R., Yeo C. S., Venugopal S., Broberg J., and Brandic I. 2009. Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation Computing Systems, Netherlands

[9] Hung, L., X., Lee S., Truc P. T. H., Vinh, L. T., Khattak A. M., Han M., Hung D. V., Hassan M. M., Kim M., Koo K.H., Lee Y. K., and Huh E. N. 2010. Secured WSN-integrated Cloud Computing for u-Life Care. 7th Annual IEEE CCNC 2010 Conference , Las Vegas, USA.

[10] Ammenwerth E., Schnell-Inderst P., Machan C. 2008. The effect of electronic prescribing on medication errors and adverse drug events: a systematic review. Journal of the American Medical Information Association . pp. 585-600

[11] Kim J. A., Cho I., Kim Y. 2008. CDSS(Clinical Decision Support System) Architecture in Korea, International Conference on Convergence and Hybrid Information Technology, Washington, DC, USA

[12] Zhou Q. 2009. A Clinical Decision Support System for Metabolism Synthesis, International Conference on Computational Intelligence and Natural Computing, China.

[13] Jegelevicius D., Krisciukaitis A., Lukosevicius A., Marozas V., Paunksnis A., Barzdziukas V., Patasius M., Buteikiene D., Vainoras A., Gargasas L. 2009. Network Based Clinical Decision Support System. 9th International Conference on Information Technology and Applications in Biomedicine, ITAB 2009, Larnaca, Cyprus.

[14] Pervez Z., Lee S., Hur S. J., and Lee Y.K. 2010. Multi-Tenant, Secure, Load Disseminated SaaS Architecture. The 12th International Conference on Advanced Communication Technology (ICACT2010), Phoenix Park, Korea, pp.214-219

[15] Stead, WW., Lin, HS. 2009. Computational Technology for effective health care: immediate steps and strategic directions. Washington, DC: The National Academies Press

[16] Fan, Y., Tseng, T., Chern, C., and Huang, C. 2009. Rule induction based on an incremental rough set. Expert Syst. Appl. 36, 9 (Nov. 2009), 11439-11450