AGENT-BASED CONTEXT-AWARE HEALTHCARE INFORMATION RETRIEVAL USING DROPT APPROACH

Agbele Kehinde[1], Ademola Adesina[2], Ekong Daniel[3] and Seluwa Dele[4]

[1,2]*Machine Learning and Intelligent System Research Group, Dept. of Computer Science, University of the Western Cape, Cape-Town, South Africa*

[3]*Dept. of Mathematical Sciences (Computer Science Option), Ekiti State University, P.M.B. 5363, Ado-Ekiti, Nigeria.*

[4]*Leprosy and Tuberculosis Unit, Ekiti State University Teaching Hospital, Ado-Ekiti, Nigeria.*

Abstract*: As the volume of information available on the Web information systems is growing continuously, browsing this content becomes a tedious task given the presentation of data that does meet user's aims and needs. In this paper, to satisfy user needs, an agent-based paradigm is an appropriate solution which gives outputs suitable to the user in the form of highly ranked documents. Conversely, patient care and a health condition commonly require collaboration between healthcare providers. The emergence of agent's technology motivates radical changes of how information is obtained. This paper addresses this problem by proposing a novel DROPT (Document Ranking OPTimization) measure for information retrieval results to validate the effectiveness of the information management tasks. We propose information retrieval system architecture, which main components are context-aware agent's technology to meet users' information needs.*

Keywords*: Mobile Healthcare, Context-awareness, Agents System, Medical Informatics, Healthcare Information Management, New Ranking Algorithm.*

1. INTRODUCTION

Healthcare is a vast open environment characterized by shared and distributed decision making and management of care, requiring the communication of complex and diverse forms of information between a variety of clinical and other settings, as well as the coordination between groups of healthcare professionals with very different skills and roles. However, healthcare is one of the most difficult domains to tackle due to its inherent complexity.

Agent technology has emerged in the last years as a new and promising paradigm focused on the modeling, design and development of complex systems. It has become a leading area of research in artificial intelligence (AI) and the focus of a number of major initiatives [1, 2, 3]. Agents and, more generally, multi-agent systems allow to model in a realistic way complex, heterogeneous and distributed systems and environments, by assigning an agent to each real-world entity involved. The use of agents in healthcare has a wide range of applications such as

supporting the expert's decision-making, accessing distributed data sources or the coordination of the execution of healthcare activities [1, 4]. Therefore, it is the aim of healthcare software systems to operate effectively in this environment, in order to meet the information needs of patients and healthcare providers.

The emergence of elegant types of mobile computing devices and developments in wireless networking are driving a spread in the domain of computing from the workplace, home, office and to other facets of everyday life. This trend will lead to the scenario, often termed pervasive computing, in which cheap, interconnected computing devices are ubiquitous and capable of supporting users in a range of tasks like healthcare environments. In particular, pervasive computing demands applications that are capable of operating in highly dynamic environments and of placing fewer demands on user attention. In order to meet these requirements, pervasive computing applications will need

to be sensitive to context. By context, we refer to the circumstances in which a computing task takes place. For instance, healthcare environments are characterized by the need for coordination and coordination among healthcare providers with diverse areas of expertise, the integration of data from many devices, and mobility of healthcare providers, patients, documents, and equipment. Ubiquitous computing enables us to meet these characteristics of healthcare environment.

Currently, the programming of context-aware applications is complex and laborious. This situation could be remedied by the creation of an appropriate infrastructure that facilitates a variety of common tasks related to context awareness, such as modeling and management of context information. In this paper we study the adoption of the agent paradigm in the healthcare domain. We propose a ubiquitous computing Healthcare Information System (HIS), which main components are context-aware agent's technology that autonomously accomplishes information management tasks such as retrieving and presenting information which are accessible via the internet. In this context, we use DROPT measure to validate the effectiveness of the information management tasks as a strategy to evaluate the relevance of the ranked document, by giving a score to each document.

The rest of the Chapter is structured as follows. *Section* 2 reviews state of the art domain of application agent technology in healthcare. *Section* 3 presents the agent paradigm and how its attributes are adequate to address healthcare problems. *Section* 4, examines the contextual retrieval of medical information in healthcare and describes an application scenario for information retrieval and management serving as a practical example in this paper. *Section* 5 presents our ranking approach while *Section* 6 describes our proposed agent system architecture that allows personalized information retrieval. *Section* 7 presents some concluding remarks.

2. REVIEW OF PREVIOUS RELATED WORK

In this section, we show how agent techniques have been applied to information retrieval systems in healthcare informatics domain. The complexity of the system is reduced by using agents in healthcare informatics domain, while the objective of the system remains the same. Among the most important related agent applications in the medical informatics domains we can describe the following:

Context-aware Hospital Information System (CAHIS) is a multiagent system that provides capabilities of intelligence and pro-activity to healthcare environments resulting in an ambient intelligence (AmI) system [5]. Autonomous agents enable ubiquitous technology to respond to users' particular conditions and demands (*e.g.*, access to real-time patient data to make critical-care decisions) providing security to all transmissions [6]. Based on the system's architecture, the *context-aware clients* are agents that allow nurses and physicians to fill requests and communicate with other staff members. The *hospital information system agent* accesses the legacy electronic information service. This agent knows the interests of practitioners and communicates the appropriate data to the appropriate agent pro-actively. The *location estimation agent* embeds a method to estimate the location of all entities of the hospital using RFID signals. This information is used by the *public display agent* to show the location of users and other devices in a covered area. The *map agent* filters the information to show (in personal computers and public displays) according to the user's permissions. The *user's proxy agent* acts as proxy between the user (patient) and his/her authorized data. The *broker agent* handles communication between all agents using XML-coded messages.

AmI refers to a new paradigm in information technology, in which people are empowered through a digital environment that is aware of their presence and context and is sensitive, adaptive and responsive to their needs, habits, gestures and emotions. CAHIS designs the basic pieces of this puzzle into different agents that cooperate in order to provide the right information to the right people in the right location. On the one hand, agents monitor and locate users in the system and, on the other hand, agents process and pro-actively supply these data to the users. Agent-related features add some interesting characteristics to these systems in comparison to classical approaches, such as coordination capabilities (*e.g.*, CAHIS) in order to discover the location of the requested data in a distributed environment.

Moreover, one of the most important features of this agent-based system is their ability to access heterogeneous and distributed sources. In many real environments, partial medical data is too critical and sensitive to be moved to other places for processing and, in consequence, distributed data accessing is mandatory. However, CAHIAS did not exploit agent mobility capabilities which may open the possibility of distributing agents in runtime according to the information requirements. Even though being interesting from the data management point-of-view, the fact that an agent may move to destinations not necessarily located in a

controlled environment raises complex security issues which are still unresolved [7].

According to [8] present an *agent-based healthcare intelligent assistant*. This assistant was designed to be used by medical practitioners to retrieve and use existing organizational knowledge in order to help solving current medical episodes. They use case-based reasoning, which captures the experiential knowledge of healthcare practitioners of past cases and uses it to classify new ones [9]. Using a grid-like approach, all the resources available in hospitals can be accessed and exploited by geographically distributed medical practitioners to classify medical episodes.

The system's architecture is composed by several nodes (representing distributed hospitals) that are managed by two central agents: The *Manager Agent* and the *Master Presentation Agent*. The user is localized at the top of the architecture and interacts with the system through a *Web Server Agent*. The star-based topology distributes a query containing a new case among a set of available nodes (through the manager agent). Each node includes a reasoner agent, called *Case-Broker Agent*, which matches the received case with its database, which represents the set of past cases in the context of a node hospital. The matched results are formatted by the *Presentation Agent* that forwards them to the central MPA. Finally, the results are compiled and presented to the user. The simple coordination method (distribution) allows collecting information from local heterogeneous sources in a transparent manner. In addition, this system adds security to all transmitted data (queries and results). Only authenticated users can access the system and retrieve data.

Agent.Hospital [10, 11] is an open agent-based framework for highly distributed applications in healthcare that provides different interfaces to integrate existing information systems. The framework contains numerous healthcare actors and consists of detailed partial models of the healthcare domain. It enables the examination of modeling methods, configuration problems, and agent-based negotiation strategies and coordination algorithms. *Agent.Hospital* architecture includes a set of facilities shared across all delivered services. These facilities include a directory of agents (with the services they offer), management of events, an ontology repository, a knowledge base of medical terms based on the OntoHos ontology [11], a generic *Actor Agent* that can be customized depending on the role of an actor, a repository of source files of services, and a simulation environment named SeSAm.

Inside this framework, several subprojects have been implemented. ADAPT focuses on coordination of appointments for treatment and examination actions. Its goal is to develop and provide flexible and adaptable coordination strategies that can be applied to real hospitals. This hospital scheduling scenario is typically implemented by a computer based simulation model, where different strategies and their consequences can be evaluated. The *Policy-Agent* project aims to develop a scheduling system for centralized operating theatres based on intelligent software agents. In addition, the Medpage-Agent project undertakes the task of planning and scheduling different functional units as departments. The *EMIKA* project deals with the messaging service in remote mobile devices. *Agil* simulates the behavior of an emergency unit. Finally, in the middle of these projects, the *directory facilitator* maintains an up-to-date list of the offered services in the *Agent.Hospital* platform and permits a transparent use of these features by all agents.

The *Aingeru system* supports intelligent, continuous and pervasive monitoring of elderly people [12, 13]. Each monitored person carries a PDA connected to sensors that sample some physiological parameters (*e.g.*, heart rate, oxygen percentage in blood). The system evaluates the data (locally at the PDA) and establishes if it is necessary to contact a specialist. A central element located in a hospital receives all the data from external patients (through GPRS transmissions) and stores them in a global database, handling particular cases, like alarms, accordingly. In addition, physicians can access all patients' related data through a Web-based application.

The system's architecture includes diverse components: The user's PDA (monitors the user), the control centre (in charge of the tele assistance service), the health centre (that offers medical services), and the technical centre (that maintains the entire infrastructure). Inside the PDA two kinds of agents are implemented: sensors and reasoners. Each *Sensor Agent* reads samples of a particular sensor. An ontology-based reasoner, called *Conditions Checker Agent*, pro-actively detects potentially anomalous situations. The *Emergency Agent* is able to manage emergencies detected by the *Conditions Checker Agent* and the explicit alarms sent by users. An auxiliary agent (*Majordomo Agent*) can exchange information with the user. Alarms sent from the PDA are received by the control centre, and a set of *Emergency Distributor Agents* forward the events to the correct health care centre which, after receiving the alarm, creates a contingency plan and advices all involved practitioners. Finally, these actors access the up-to-date patient's data through a Web browser. One of the contributions of this

project is that it optimizes the data transmitted from patients to the central controller, avoiding unnecessary costs and improving the autonomy of these patient devices. The reasoner agent can be customized according to the diseases that should be monitored by means of tailoring the domain ontology.

Geriatric Ambient Intelligence (GerAmI) is an intelligent supervision system that provides up-to-date patient data [14]. Its main aim is to support elderly patients in all aspects of daily life, predicting potential hazardous situations and delivering physical and cognitive support. The system combines simple tasks like the verification of the location of patients with complex tasks like the creation and supervision of a daily planning for a nurse. The authors of this system propose four different agents: *Patient, Doctor, Nurse, and Manager*. They are connected using WiFi to transmit data to/from doctor's PDAs and, RFID to locate the patients that bring a transponder according to the system architecture. Nurses and doctors interact with the system through PDAs.

The *Patient Agent* maintains all the current information related to the patient, which is analyzed in order to know whether the goals established for the patient are being achieved. The patient agent's goals and beliefs depend on the treatment and medication prescribed by the doctor. The *Nurse Agent* schedules the list of daily tasks of the nurse according to his/her profile, preferences, requested actions over patients, and available resources. A *Manager Agent* controls all patients' locations, manages the medical records and monitors all assignments of patients with nurses and doctors. The *Doctor Agent* transmits all actions to be performed by patients (through *Patient Agent* and *Nurse Agent*) and updates the medical records (through the *Manager Agent*).

3. AGENTS AND MULTI-AGENT SYSTEMS IN HEALTHCARE ENVIRONMENT

An agent is a software entity that applies AI techniques to choose the best set of actions to perform in order to reach a goal specified by the user. In this context, an agent is an entity that must be able to *perceive* the physical or virtual world around it using *sensors* [4]. A fundamental part of perception is the ability to recognize and filter out the expected events and attend to the unexpected ones. Intelligent agents use *effectors* to *take actions* either by sending messages to other agents or by calling application programming interfaces or system services directly. Agents exhibit some characteristics [15] like *autonomy* to operate without the direct intervention of humans, reactivity to respond to the perceived changes in the environment, *proactiveness* to take the initiative when

appropriate, and social ability to interact with other agents. Multi-agent systems are applications in which many autonomous software agents are combined to solve large problems.

Analyzing the healthcare domain [14, 16, 17], we find some common problems. It is very usual that the knowledge and data required to solve a problem are spatially distributed in different locations, which adds several constraints in the planning of coordinated actions. The provision of healthcare typically involves the coordination of the effort of several individuals (*e.g.*, physicians, nurses, carers, social workers, managers, receptionists) with different skills and needs and located in different places, usually without the supervision of a single centralized coordinator. There is a great amount of medical knowledge available electronically that can be accessed and reused, but it is expressed in many different formats and, very commonly, in textual form. Due to the reluctance of individuals involved in healthcare, the penetration of computerized systems is a hard and slow process.

Under these circumstances, agent technology is a good option to be used in healthcare applications. As argued by [3], agents offer a natural way of tackling inherently distributed problems with heterogeneous sources, by cooperating and coordinating their activities, and also acting pro-actively performing tasks that may be beneficial for the user. Moreover, Fox et al. [18] identified other benefits of agents applied to healthcare: (*a*) agent technology offers advanced platforms for building expert systems to assist individual clinicians in their work and (*b*) distributed agent systems have the potential to improve the operation of healthcare organizations, where failures of communication and coordination are important sources of error.

4. CONTEXTUAL RETRIEVAL OF HEALTHCARE INFORMATION

Traditional Web information retrieval systems are isolated from the context in which a request occurs. Information requests occur for a reason (related to the current user's activity), and that reason grounds the query in contextual information necessary to interpret and process it. Without access to this context, requests become highly ambiguous, resulting in incoherent results, and unsatisfied users [19]. This is evident in the healthcare domain, in which vast quantities of medical information are now available through the web and can easily lead to information overload [20]. One way to overcome such a problem is to provide an environment that proactively retrieves and presents information based on the healthcare

professionals' context, thus providing Context-Aware information Retrieval (CAR) [21]. Context-aware computing technology is a key element to construct this new generation of Web retrieval systems by sensing the changes on the users' activities, to predict users' interests and then, retrieve information and present using the DROPT measure.

Ubiquitous and context-aware computing technology may provide support to healthcare professionals for opportunistically acquiring, managing, and sharing knowledge, which are issues faced everyday by healthcare providers. It is for these reasons; we have generated scenarios to bridge the gap between the medical knowledge management practices and an idealized pervasive healthcare environment. These scenarios will address and simplify time consuming tasks, such as searching in digital libraries from the Web through the use of ubiquitous computing technology.

5. OUR RANKING TECHNIQUE

Based on equation (4), a DROPT measure: an acronym for Document Ranking OPTimization approach for documents retrieved from a corpus is developed with respect to document index keywords and the query vectors. This based on calculating the weight (w_{ij}) of keywords in the document index vector, calculated as a function of the frequency of a keyword k_j across a document d_i.

Step 1:

Let a query vector, Q, be defined as:

$$Q = [q_1 \ q_2 \ q_3 \ ... \ q_1] \tag{1}$$

Where, $q_1 = (x_i, 1)$, x_i being a term string with a weight of 1.

Step 2:

Let the indexed document corpus be represented by the matrix:

$$D = \begin{bmatrix} d_{11} & d_{12} & d_{13} ..... & d_{1l} \\ d_{21} & d_{22} & d_{23} & .....d_{2l} \\ & & \vdots & \\ & & \vdots & \\ d_{N1} & d_{N2} & d_{N3} & \cdots & d_{Nl} \end{bmatrix} \tag{2}$$

Where $d_{jk} = (y_{jk}, w_{jk})$ y_{jk} being an index string, with weight w_{jk}.

Step 3:

We compute the convolution matrix, representing:

$$W = D \times Q = \begin{bmatrix} w_{11} & w_{12} & w_{13} & \cdots & w_{1l} \\ w_{21} & w_{22} & w_{23} & \cdots & w_{2l} \\ & & \vdots & & \\ & & \vdots & & \\ w_{n1} & w_{n2} & w_{n3} & \cdots & w_{n1} \end{bmatrix} \tag{3}$$

where $w_{ij} = w_{kl}$ iff *IsEqualStringIgnoreCase* (q_p, d_{rs}); 0, *otherwise*; $|l| \leq |n|$, l being the number of terms in the query vector and n the number of retrieved documents that are indexed by at least one keyword in the query vector.

Step 4:

Reference [22] studied weighted relevance of terms in a document by considering term frequency (*tf*) and term document frequency (*idf*). Term frequency is the number of times a given term occurs in a given document, while document frequency is the number of times the term occurs in all documents. The author argued that the more a term occurs in one document, but less in other documents, the more relevant it is to that document. Consequently the relevance weight is proportional to the term frequency and inverse document frequency. The work reported in [23] shows the relevance weight is given by:

$$w_{ij} = \frac{(\log tf_{ij} + 1.0) \quad \times idf_j}{\sum_{j=1}^{m}[(\log tf_{ij} + 1.0) \quad \times idf_j]^2} \tag{4}$$

Where $tf = \dfrac{freq_{ij}}{total\ keyword\ cnt}$, $idf = \log\left(\dfrac{N}{n_k}\right)$,

where $freq_{ij}$ is the frequency of the K_{ith} user in D_{ith} query; totalkeywordcnt is the total keyword count in the document databases; n_k is the number of documents indexed by the keyword k_j and finally, N is the total number of documents containing keyword k_j.

Step 5:

Using the weighted root mean squares (RMS) averages the overall fitness of the document; ? with respect to a given query is calculated as:

$$\varpi = \prod_{i=1}^{n} \frac{1}{l} \sqrt{\sum_{j=1}^{i} w_{ij}^2} \tag{5}$$

Where W_{ij} is the weight assigned to the keyword K_j in document $D_{i,l}$ is the total length of keywords, while ? is the mean weight.

Step 6:

In order to prevent the relevance of any model representation to increase without bounds, the overall relevance judgment is given by,

$$G = [g_{ij}]_{n \times l}, \text{ where } g_{ij} = \min(w_{ij}, \overline{q}_{ij}) \qquad (6)$$

$$1 \le i \le n, 1 \le j \le l$$

Where G is a query vector with a small-operator defined as a matrix.

Step 7:

Therefore, any weight component of matrix G greater than the mean weight values will be retained to add to a matrix T is given by,

$$T = [t_{ij}]_{n \times l},$$

$$\text{where } \begin{cases} t_{ij} = g_{ij}, & \text{if } g_{ij} \ge \varpi \\ t_{ij} = 0, & \text{if } g_{ij} < \varpi \end{cases} 1 \le i \le n, 1 \le j \le l \qquad (7)$$

Step 8:

It is also necessary to compute the weights, Sco_i of the term that makes up the user information needs, which are the largest weighting value of each corresponding vector given by:

$$Sco_i = \max \{t_{ij}\}, 1 \le i \le n \qquad (8)$$

$$1 \le j \le l$$

Where n is the total number of retrieved documents.

Step 9:

Term-weight relevance judgment is also carried out using the output of the context-aware system and is given by,

$$D = \{d_i \mid \text{if } Sco_i > 0, 1 \le i \le n\} \qquad (9)$$

Document d_i is retrieved if scoi is greater than zero ($Sco_i > 0$) and then added into the retrieved document set D. Term-weights are bound within [0.0…1.0] for each document. Hence documents are sorted in ascending order of Sco_i, ranked and given to the user to meet their information needs.

Step 10:

The keyword set K provided by the documents and the weight values will be updated by the feedback of the users.

(*i*) Any new query term not belonging to K will be added and a new column of weight value will be computed and expanded for documents routinely.

(*ii*) If any retrieved document d_i is retrieved by the users, the corresponding weight values with respect to the query keywords will be increased by equation (9). The default of β is set to increase the corresponding weight values.

$$w_{ij} = (w_{ij})^{\beta}, \text{ where } 0 < \beta < 1, i \in \{i \mid d_i \in D\} \text{ and } j \in \{j \mid q_j = 1\}$$

$$(10)$$

We coined the acronym DROPT measure to name our ranking-driven approach that will provide a limited number of ranked context documents in response to a given query. It will also improve the ranking mechanism for the search results in an attempt to adapt the retrieval environment of the users and amount of relevant context information according to each user's request. Finally, the DROPT measure is self-learning that can automatically adjust its search structure to a user's query behavior.

6. THE PROPOSED SYSTEM

"A system is a context-aware system if it uses context to provide relevant information and/or services to the user, where relevancy depends on the user's tasks". The intension behind proposed Context-Aware Healthcare Information System (CAHIS) designs is to allow different agents within the healthcare environments to cooperate in order to reduce tasks complexity by providing the right relevant information to the right user in the right location. The basic components of the proposed CAHIS architecture comprises of seven agents. The users carry a handheld computer that estimates their location, provides them with information relevant to their location, and allows them to fill requests and communicate with other healthcare providers. The architecture of the proposed system, Overall Context Aware IR System as illustrated in figure 1 groups seven independent task categories, which are allocated to agents. There are seven types of agents recognized in the system to represent the IR solution: user interface agent, reformulate agent, search agent, document agent, match agent, user model agent, and display agent.

Each of these agents is discussed in more detail as follows:

1. **User Interface Agent:** This Agent is designed for interaction with humans and responsible for mediating between the external user and the rest of the system (other agents). The interface allows the user to enter keyword based query terms. This agent allows the user to evaluate the relevance of the ranked documents, by giving a score to each document as a function of the frequency of keyword

across the document. The interface notifies users the availability of search results and in turn provides feedback.

2. **Reformulate Agent:** This agent processes the input raw query from the user agent by pre-processing techniques (i.e. stemming operations). The refined queries are then sent to the match agent.

3. **Search Agent:** This agent carries out the task of submitting queries in the correct form and gathering information from the Web. This agent uses the keywords to retrieve documents, hence the results of this task are then sent to the document agent.

4. **Document Agent:** The goal of this agent is to index the documents using normalized keywords. The representation typically used is a set of features derived from the document collection, which is a vector weighted keywords. Consequently, the highest indexed documents are sent to the match agent.

5. **Match Agent:** This agent performs the task (matching process) of comparing the refined queries

against the indexed documents. The matching result is a list of potentially relevant documents that are then sent to the display agent, according to the user information needs.

6. **Display Agent:** This agent displays the results of the matching process (relevant documents) and performs ranking processes. The ranking function provides more accurate matching according to the representation of the current user model of user information need. Consequently, the ranked documents, according to their relevance weights are then shown to the user through the user interface agent.

7. **User Model Agent:** This agent is a user-feedback that modifies the representation of user-needs. The task of this agent is to guide the user in the query formulation process and to store and manage the user's interest in the form of a user profile. This agent is a major element of the system architecture and is composed of Document Ranking OPTimization (DROPT), relevance feedback (RF) and a context awareness component.
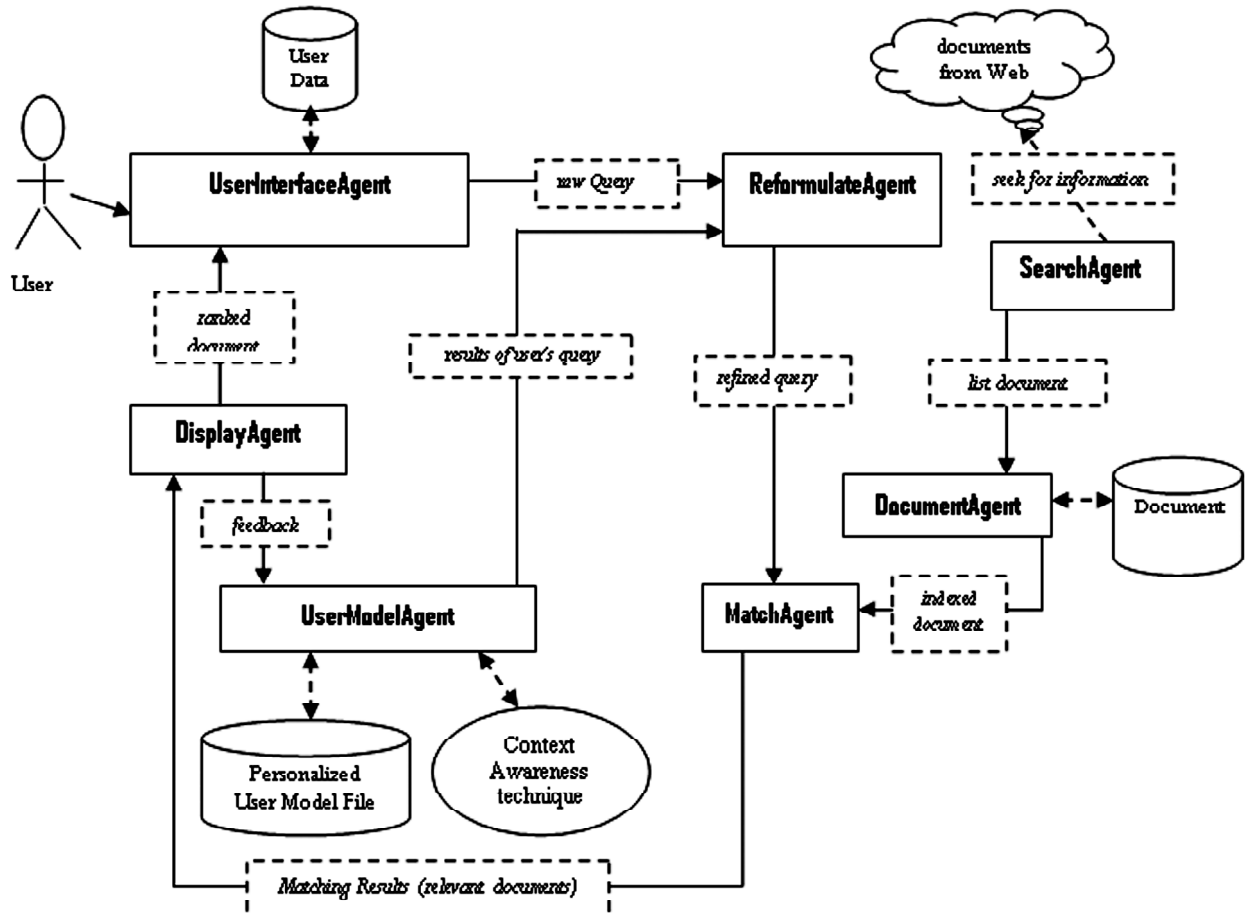

Figure 1: Overall Context-Aware IR System Architecture

Relevance feedback is one of the processes in information retrieval system that seeks to improve the system's performance based on user's feedback. This user-feedback modifies the queries using judgments of the relevance of a few, highly-ranked documents to make a better clinical decision. Historically, relevance feedback has been an important method for increasing the performance of IRS. In this context, the personalized user model aims at improving the retrieval process by taking into account the particular interests of individual users. However, not all user preferences are relevant in all circumstances.

6.1 System Plan Model

The context aware IR system plan is illustrated in the sequence diagram of Figure 2. The system plan illustrates two duties of the user interface agent: reformulate query and provide feedback. The reformulate agent, the search agent, the document agent, the match agent, user model agent, and the display agent provide their services autonomously. The modeling notations presented in the figure follows the approach [5]. In an attempt to achieve the system goal, the user communicates with the system,

through an interface with two goals: (*i*) provide interest keywords and (*ii*) provide feedback on the retrieved documents for ranking according to the relevance weight of the documents. The system automatically creates new user profiles most relevant for an existing user. The retrieved document is then indexed to become an actual document, based on the user profile. The ranking result is displayed to the user, so that the user can provide feedback on the documents.

A sequence diagram for context aware IR system plan, shown in Figure 2 below, is a diagram that shows actual events and interactions between events in the horizontal direction and sequence in the vertical direction. The vertical dotted lines represent the lifetime of the events and the horizontal arrows the interactions of messages between events. These messages can represent any kind of message (specifically a call to an operation on the target event). Narrow elongated boxes on the event lifelines represent the activation of the event when interactions are sequential and represent calls to operations. The operation remains active until all the sequential operations, which it calls, have completed and returned, thus, allowing it to return control to its caller.

Figure 2: Sequence Diagram for Context Aware IR System Plan

7. CONCLUSION

In this paper we have survey for adoption of application of agents-based information system in healthcare environments to show how in particular an agent is suitable paradigm used to design and implement system. This paper has provided a review application of agents-oriented approaches to system design and implementation in healthcare informatics domains. The agent-based systems proposed in this paper exhibits a wide range of interesting characteristics, which may be difficult to find in applications developed with more classical software engineering methodologies or programming paradigms. The basics of agent technology are described as well. The development process of design and architecture stages was described. Conversely, we have presented in this paper how our ranking technique validates healthcare providers to timely access healthcare information and evaluate ranked documents based on relevance values to satisfy user's information need.

References

[1] R. Annicchiarico, U. Cortes, C. Urdiales (Eds.), "Agent Technology and e-Health", *Whitestein Series in Software Agent Technologies and Autonomic Computing*, Birkhauser Verlag, Basel, Switzerland, 2008.

[2] M. Luck, P. McBurney, O. Shehory, S. Willmott, "Agent Technology: Computing as Interaction", *(A Roadmap for Agent Based Computing)*, AgentLink, 2005.

[3] J.L. Nealon, A. Moreno, "Applications of Software Agent Technology in the Health Care Domain", Birkhauser Verlag, Basel, Switzerland, 2003 (Ch. Agent-Based Applications in Health Care, pp. 3-18), (2003).

[4] M. Wooldridge, "An Introduction to Multiagent Systems", 2nd ed., John Wiley and Sons, Ltd., West Sussex, England, 2009.

[5] M.D. Rodriguez, J. Favela, A. Preciado, A. Vizcaino "Agent-based Ambient Intelligence for Healthcare", *AI Communications*, **18(3)**, pp. 201-216, (2005).

[6] M. Tentori, J. Favela, M. Rodriguez, "Privacy-aware Autonomous Agents for Pervasive Healthcare", *IEEE Intelligent Systems*, **21(6)**, pp. 55-62, (2006).

[7] P. Vieira-Marques, S. Robles, J. Cucurull, R. Cruz-Correia, G. Navarro, R. Marti, "Secure Integration of Distributed Medical Data Using Mobile Agents", *IEEE Intelligent Systems*, **21(6)**, pp. 47-54, (2006).

[8] S. Singh, B. Ikhwan, F. Haron, C.H. Yong, "Architecture of Agent-based Healthcare Intelligent Assistant on Grid Environment", in: K.-M. Liew, H. Shen, S. See, W. Cai, P. Fan, S. Horiguchi (Eds.), *Proceedings of the 5th International Conference on Parallel and Distributed Computing: Applications and Technologies*, PDCAT 2004, **3320**, of LNCS, Springer, Berlin/Heidelberg, Singapore, pp. 58-61, (2005).

[9] S. Schmidt, R. Montani, L. Bellazzi, L. Portinale, L. Gierl, "Cased-based Reasoning for Medical Knowledge-based Systems", *International Journal of Medical Informatics*, **64(2-3)**, pp. 355-367, (2001).

[10] S. Kirn, C. Anhalt, H. Krcmar, A. Schweiger, "Multiagent Engineering: Theory and Applications in Enterprises", *International Handbook on Information Systems, Springer, Berlin-Heidelberg, 2006 (Ch. Agent. Hospital-Health Care Applications of Intelligent Agents*, pp. 199-220), (2006).

[11] M. Becker, C. Heine, R. Herrler, K.-H. Krempels "Applications of Software Agent Technology in the Health Care Domain", Birkhauser Verlag, Basel, Switzerland, (Ch. OntHoS-an Ontology for Hospital Scenarios, pp. 87-104), (2003).

[12] M.I. Bagues, J. Bermudez, A. Illarramendi, A. Tablado, A. Goni, "An Innovative System that Runs on a PDA for a Continuous Monitoring of People, in: P. Kokol, T. Welzer-Druzovec, D. Micetic-Turk, M. Zorman (Eds.), *Proceedings of the 19th IEEE Symposium on Computer-Based Medical Systems*, IEEE CBMS 2006, IEEE Press, Maribor, Slovenia, pp. 151-156, (2006).

[13] A. Tablado, A. Illarramendi, M.I. Bagues, J. Bermudez, A Goni, "An Intelligent System for Assisting Elderly People", in: M.-S. Hacid, N.V. Murray, Z.W. Ras, S. Tsumoto (Eds.), *Proceedings of the 15th International Symposium Foundations of Intelligent Systems, ISMIS 2005*, **3488** of LNCS, Springer Berlin/Heidelberg, Saratoga Springs, NY, USA, pp. 466-474, (2005).

[14] J.M. Corchado, J. Bajo, A. Abraham, "GerAmI: Improving Healthcare Delivery in Geriatric Residences", *IEEE Intelligent Systems*, **23**, pp. 19-25, (2008).

[15] M. Wooldridge, N.R. Jennings "Intelligent Agents: Theory and Practice", *Knowledge Engineering Review* **10(2)**, pp. 115-152, 1995.

[16] D.W. Glasspool, A. Oettinger, J.H. Smith-Spark, F.D. Castillo, V.E.L. Monaghan, J. Fox, "Supporting Medical Planning by Mitigating Cognitive Load", *Methods of Information in Medicine*, **46(6)**, pp. 636-640, (2007).

[17] H. van der Linden, D. Kalra, A. Hasman, J. Talmon, "Inter-organizational Future Proof EHR Systems: A Review of the Security and Privacy Related Issues", *International Journal of Medical Informatics*, **78(3)**, pp. 141-160, (2009).

[18] J. Fox, M. Beveridge, D. Glasspool, "Understanding Intelligent Agents: Analysis and Synthesis", *AI Communications*, **16(3)**, pp. 139-152, (2003).

[19] J. Budzik and K. Hammond, "User Interactions with Every Applications as Context for Just-in-time Information Access", *Proceedings of Intelligent User Interfaces*, *ACM Press*, pp. 44-51, (2000).

[20] W. Pratt and L. Fagan, "The Usefulness of Dynamically Categorizing Search Results", *Journal of the American Medical Informatics Association*, **7(6)**, pp. 605-617, (2000).

[21] P.J. Brown and G.J.F Jones "Context-aware Retrieval: Exploring a New Environment for Information Retrieval and Filtering. Personal Ubiquitous Computing", **5**, pp. 253-263, (2001).

[22] G. Salton and C. Buckley, "Term-Weighting Approaches in Automatic Text Retrieval", *Information Journal of Information Process Management*, **24**, pp. 513-523, (1988).

[23] G. Salton. "Automatic Text Analysis", *Science*, **168(3929)**, 1970, pp: 335-343 (1970).