Sharing Biomedical Learning Knowledge for Social Ambient Intelligence

Sabah Mohammed
Lakehead University/Computer Science Department, Thunder Bay, Canada
sabah.mohammed@lakeheadu.ca

Jinan Fiaidhi
Lakehead University/Computer Science Department, Thunder Bay, Canada
jfiaidhi@lakeheadu.ca

Osama Mohammed
Lakehead University/Software Engineering Department, Thunder Bay, Canada
omohamme@lakeheadu.ca

Abstract— In this paper, we describe a Bio-SAmI system, which is a Biomedical learning system that is context-aware and responsive to mobile learners sharing information on a social network. Bio-SAmI is a Web 2.0 enabled system which employees Social Ambient Intelligence techniques. The Bio-SAmI infrastructure is based on the Actor model that treats mobile users or "actors" as the universal primitives of computation. The actor model is implemented using a combination of Java enabled APIs including SALSA, JADE, LEAP and tuPrologME. The implemented prototype enable learners to share biomedical information represented by the DICOM SR standard in relation to the notion of inflammation as well as to respond to variety of learning queries including classifying learning case studies, finding learning case studies, locating a FOAF learner and syndication and aggregation of learning case studies .

Index Terms— Ambient Intelligence, Web 2.0, Biomedical Learning, JADE, Actor-Oriented Programming, SALSA, tuPrologME, DICOM.

I. INTRODUCTION

Ambient intelligence (AmI) is not only limited to rooms and buildings. In future whole cities will become intelligent environments – with people networking with each other, dating, finding interesting places (e.g. restaurants, museums, and meeting places), getting along with public transportation or dealing with traffic and parking problems. In such a city, millions of inhabitants interact with each other and benefit from information other people or sensors provide. It just feels like a village where somebody always helps in finding a restaurant, bar or theatre, were citizen's choices, moves and opinions influence urban planning and public intervention. Such a city and its applications can be realized by combining two major trends in mobile computing: Ambient Intelligence and Web 2.0 [1]. AmI involves the convergence of several computing areas. The first is ubiquitous or pervasive computing where its major contribution is on the development of various ad hoc networking capabilities that exploit highly portable and very-low cost computing devices. The second key area is intelligent systems research, which provides learning algorithms and pattern matchers as well as other classification, interpretation and situation assessment capabilities. A third element is on context awareness (e.g. track and position objects). However, AmI research may be addressed via variety of technological pathways that have emerged in recent years: ubiquitous computing, pervasive computing, disappearing computing, pro-active computing, sentient computing, affective computing and wearable computing. The different pathways used can imply different focus as well different perspectives (e.g. technological, social or a political one). In particular, the technological focus/perspective of AmI has been investigated with some level of detail by so many researchers [2,3]. Figure 1 illustrates the technological elements involved in the AmI research.

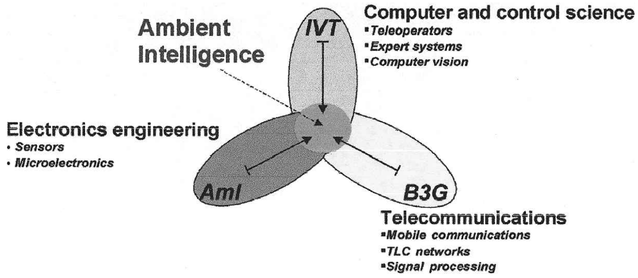

Fig. 1: The Technological view of AmI [2].

However, the social focus/perspective of the AmI research is largely neglected [4]. Instead of enhancing interactions with technological objects, there is a need for possible AmI applications that can enhance interactions with other people [5]. Enhancing the interactions among people contributes to what is currently termed as the *Social Ambient Intelligence* [6]. Central to the social ambient intelligence research is the use of Web 2.0 techniques within AmI applications. In fact, the Web 2.0 approach has revolutionized the way we use the web and certainly, it can have major positive impact on the AmI research. On one hand, Web 2.0 enables the active participation of users with new contents such as wiki pages, blogs or online multimedia tagged. On the other hand, Web 2.0 transforms the Web into an application-enabling platform. The call for using Web 2.0 technologies at AMI research and applications have been emphasized by several societies (e.g. the ISTAG--European Information Society Technologies Advisory Group http://cordis.europa.eu/ist/istag.htm) and the European Science and Technology Observatory [7]. Based on such hybrid vision, AmI research should facilitate the following additional social capabilities:

- AmI should be orientated towards community enforcement and collaboration enhancement.
- AmI should help to build knowledge and skills for work, better quality of work, citizenship and consumer choice.
- AmI should inspire trust and confidence.
- AmI should be consistent with long-term sustainability - personal, societal and environmental - and with life-long learning.
- AmI should be made easy to live with and controllable by ordinary people.

In this article, a Web 2.0-inspired AmI framework is introduced for enabling medical learners collaboration based on a simplified context-aware principles.

I. Realization of Ambient Social Intelligence Systems

Ambient computing demand that applications be subsumed into the everyday context in an unobtrusive manner with interaction modalities such that they are natural, simple and appropriate to both the individual user and their associated context. Thus ambient systems need to address some key issues [8]:

1. Provision of support for collaboration and cooperation between distributed ambient system components;
2. Delivery of systems that exhibit autonomic characteristics yielding self-management and self-healing capabilities
3. An understanding of the dynamics of context;
4. Recognition and accommodation of the diversity of devices that contribute to the organic nature of the ambient and ubiquitous computing nervous system;
5. The need for personalization and system adaptivity;

In addressing these core issues from the perspective of social intelligence, there are three important factors need to be materialized:

1. The pro-active/autonomous nature of the ambient system components
2. The generic representation of the prior knowledge that people use in describing their tasks and goals, and
3. The awareness of the new or changing context.

The task of maintaining control over pro-active/autonomous components that decides its own actions can be a difficult one as it requires mitigating the human awareness of constraining the components behavior. The solution requires the ambient system to operate according to a predefined plan library that is deterministic and restricts possible actions [9]. It demands detailed knowledge and analysis of the plans; procedures and actions needed to complete known and envisioned tasks. This is compounded by the fact that often the future situations of operation are dynamic and unpredictable. Consequently it can be difficult to make ambient systems activity adaptive and flexible and devolve much decision-making and problem solving. However, as social beings we learn about the world around us through other people, both actively and passively. People have evolved information gathering, decision making, problem solving and anticipatory judgment processes and mechanisms to understand other people's likely actions and intentions because it is advantageous to their own survival and well-being to anticipate the future actions of others [10]. These social processes and mechanisms can certainly assist the pro-active/autonomous system to be more focused to achieve its required goals. However, we need to note that a social network is a dynamic net. This means that such network must be able to continue working despite changes in the network. To provide autonomy, developers require a programming model where the communication primitives are non-blocking, where there is support for information lookup in a program's current network and where network outages are considered natural rather than as errors [11].

Luckily, programming autonomous systems is becoming more feasible with the use of Actors programming [12]. Actors are self-contained, interactive, autonomous components of a computing system that communicate by asynchronous message passing [13]. Actually, the actor model provides a unit of encapsulation for a thread of control along with internal state. An actor is either unblocked or blocked. It is unblocked if it is processing a message or has messages in its message box, and it is blocked otherwise. Communication between actors is purely asynchronous: non-blocking and non-First-In-First-Out (non-FIFO). However, communication is guaranteed: all messages are eventually and fairly delivered. In response to an incoming message, an actor can use its thread of control to modify its encapsulated internal state, send messages to other actors, create actors, or migrate to another host. Many programming languages have partial or total support for actor semantics, such as SALSA, ABCL, THAL, Erlang, E, and Nomadic Pict [14]. However, only SALSA is proved to be effective in the area of mobile or ambient programming [15]. If actor semantics are intermixed with some Web 2.0 capabilities (e.g. syndication and aggregation of knowledge objects via the RSS protocol) this will certainly enable the collective intelligence required for the ambient systems [16].

Related to the issue of arriving at a generic representation for user knowledge, the key solution of this problem is the Use of an RDF Model which can be used to represent different user goals and requests as sets of generic features as well as to represent their content resources as sets of actor components [17]. The other key element then is to develop an RDF server utilizing this Model to support transformations across different users and presentation channels. Finally, the design of ambient systems and applications in these environments needs to take account of heterogeneous devices, mobile users and rapidly changing contexts. Most importantly, actors in ubiquitous and mobile environments need to be context-aware so that they can adapt themselves to different situations. In this direction, we argue that ambient systems must provide middleware support for context-awareness. We also propose a middleware that facilitates the development of context-aware actors. The middleware allows actors to acquire contextual information easily, reason about it and then adapt themselves to changing contexts. Another key issue in these environments is allowing autonomous, heterogeneous actors to have a common semantic understanding of contextual information. The proposed middleware can tackles this problem by using common ontologies to define different types of contextual information even with existence of new tagging and folksonomies.

II. LEARNING FOR SOCIAL AMBIENT INTELLIGENCE

The ultimate goal of our working project is the design of an open AMI application for providing useful biomedical information to a social network of learners. This type of application belongs to so-called personal agent's assistant applications. Such applications consist of learning assistants which are characterized by several elements: they are not necessary designed simultaneously or coordinately, they do not refer to a central and unique system. In addition, learning assistants can appear or disappear without any global influence on the functioning of the whole system. Biomedical knowledge encompasses any piece of information available into the environment. Actors defining for themselves goals in terms of knowledge to gain provide the dynamic of the whole system. This new learning vision is based on an information society of the future, where user-friendliness, effective and distributed support of learning agents and services, mobility, interoperability, ubiquity and support for interactive work are strongly emphasized. To realize such learning perspective based on AmI and Web 2.0, it is paramount that personal assistant agents perform a large number of lightweight queries in search for some information. These agents may even have the reasoning and decision making authority to make binding contracts on behalf the user. Such AmI systems may be composed of distributed heterogeneous learning assistant agents, which can both conflict and co-operate with each other but have no necessary central locus of control. Multi-agent system paradigm is a convenient abstraction for designing such systems [18]. Agents are usually used for modeling and engineering complex articulated systems including AmI systems that assists learning [19]. Knowledge is no longer external to an agent's processes; it is part of the 'agent's world', and knowledge exchanged can be seen as 'result sharing' between agents. From the perspective of learning, agents cooperate to reason about a unique high-level query. According to this vision, learning agents can be agents may be represented based on Multiple Agents Systems (MAS). Actually, MAS have been proposed as a suitable conceptual framework for building such information systems [20,21]. While several approaches to agent technology are showing significant promise, many critical issues remain unsolved. For one thing, agents created within one agent framework can seldom communicate with agents created within another. Although several attempts tried to solve this problem by developing open agent systems including EMAF, JADE, InfoSeuth and OAA [22], still there are a number of broad technological challenges for research and development over the next decade [23]. These challenges mainly include the followings [24,25,26]:

- Semantics
- Mutability and Mobility
- Social Networking Infrastructure
- Scalability
- Security

Within the presented work, we are trying to partially cover some of them. Our attempt provided solutions to the following issues:

- Supporting learning assistant agents communicating in a social networking environment.
- Supporting mutable and mobile agents through adopting the Actor model.
- Providing better semantic infrastructure through adopting common ontology

III. INFRASTRUCTURE FOR SOCIAL AMBIANT INTELLIGENCE

Nowadays, distributed ambient information systems need to operate in open, evolving, heterogeneous environments. Trust in these systems by their owners and users entails ever-increasing expectations for robustness, fault tolerance, security, flexibility and adaptability. The unit of distribution in our model is an "ambient-aware object". We model such objects to be close relatives to actors. The actor model is a natural concurrent extension of object-orientation and has been recurrently employed in the past as the foundation for object-oriented concurrent and distributed languages. An actor can be concisely described as an object with its own computing capabilities, able to send asynchronous messages to other actors. To support such asynchrony, an actor encapsulates an incoming and outgoing message queue. For the purposes of this paper, we will regard an ambient-aware object as a basic actor plus some additional properties, which will be introduced when appropriate. The actors society is created using a middleware that provides completely distributed information, resources, controls, etc, for learning agent services. The middleware is based on 4-layers framework (see Figure 2).

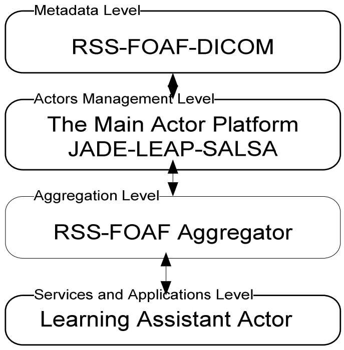

Fig. 2: The 4-Layer Framework.

The first level of the framework integrates RSS and FOAF by several extension modules designed for Weblog biomedical learning. These metadata are maintained by a content management actor (CMA), the RSS-FOAF tracking actor (RFTA) and the DICOM global ontology constructor (DGOC). The JADE-LEAP-SALSA level consists of containers, platforms, AMS, DF, etc. It provides a runtime environment for enabling FIPA learning assistant actors to execute on a wide range of devices varying from servers to Java enabled cell phones. The aggregation level is maintained by RSS and FOAF aggregator (RFA) which supports information distribution process such as collection, creation and publication. Finally, the services and application level perform search and recommendation for the various learning assistant actors. Figure 3 illustrates our biomedical social ambient intelligence architecture (Bio-SAmI) based on the 4-level framework. It describes the main classes involved in designing our Bio-SAmI system using a form of UML.

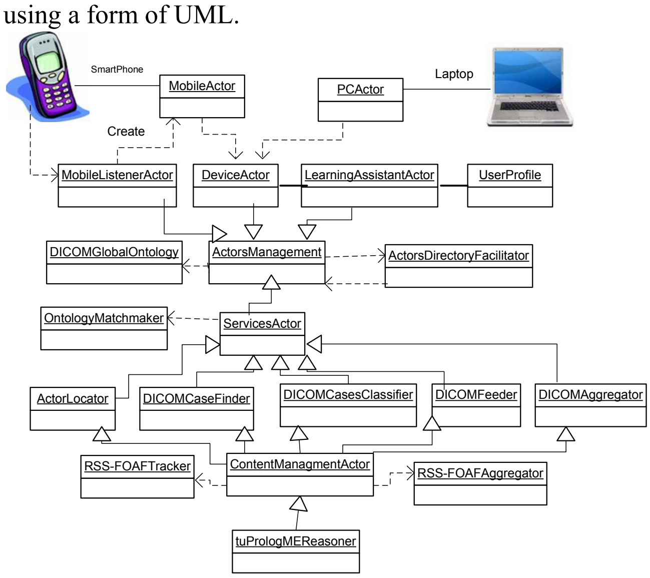

Fig. 3: The major classes involved in Bio-SAmI.

The Bio-SAmI provide access to variety of services (e.g. locating a peer/friend, finding a DICOM[1] biomedical case study, classifying FOAF[2] group DICOM case studies, syndication of a new DICOM case study and aggregating a DICOM case study). Based on the DICOM global ontology, the ServicesActor map the syntactical level of the services to the semantically level of the service actor. Figure 4 illustrates how the learning assistant actor searches for a friend on the FOAF social network. The DeviceActor is the first object to be consulted as it sends the FOAF query to the ActorsManagement object where it send it ServicesActor and from there to the ActorLocator and finally to the RSS-FOAFAggregator object to respond back whether the required person is on the FOAF network or not.

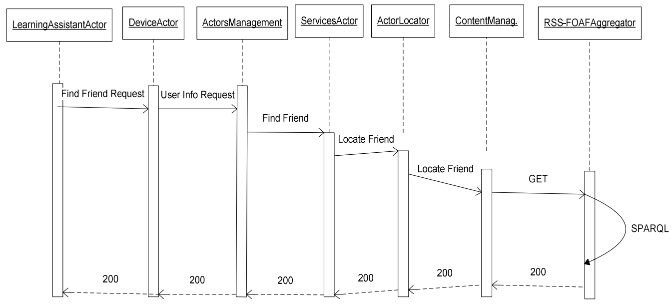

Fig. 4: Using Bio-SAmI for Peer Finding.

IV. THE BIO-SAMI APPLICATION PROTOTYPE

The Bio-SAmI architecture has been exploited in implementing a biomedical social AmI application for sharing inflammation case studies described using the DICOM ontology. Although, the Digital Imaging and Communication in Medicine (DICOM) contains around 2,000 terms/tags [27]. The DICOM standard does not ensures that everyone uses the same set of attributes for describing similar case study. The vocabulary used for the values of attributes can be different making searches across multiple repositories within a social networking environment difficult if not impossible. For example, consider the attribute Body Part in a DICOM image of the lower half of the human face. One image might contain the word ‘Jaws’ as a value of the attribute Body Part. Another similar image might contain the word ‘Mandible’. Yet another image might contain the SNOWMED id for that body part, such as T-D1217. The situation is further complicated by the fact that ‘Mandible’ and ‘Jaws’ are not equivalent – ‘Mandible’ (and ‘Maxilla’) are a sub-part of ‘Jaws’. Similarly when we talk about inflammation, there are greater variety of interpretations and use. According to Gangemi, et al [28], inflammation is an intrinsic concept for biomedical research as it contributes to almost every disease type as it originates from a tissue response to injury or imitation (see Figure 5).

[1] Digital Imaging and Communication in Medicine (DICOM) Standard
[2] Friend-Of-A-Friend (FOAF) Social Networking Ontology

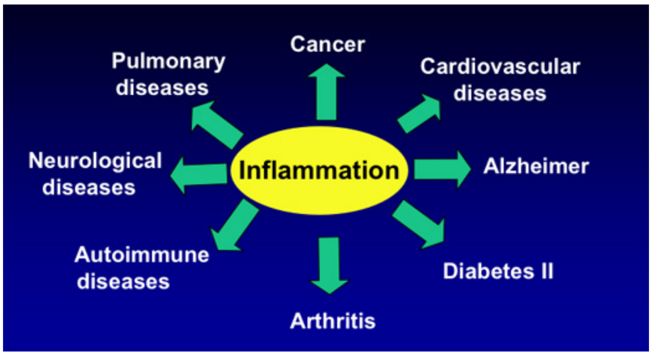

Fig. 5: The Wide use of Inflammation [28].

Using the inflammation concept in describing biomedical case studies could result in different taxonomic interpretations:

- a physiological function performing segregation of external effects;
- a portion of a body part which embodies that physiological function;
- a specific abnormal morphology (texture, color, shape, various abnormalities) of a portion of a body part.

Taxonomic issues and contexts are important for biomedical research as it can point the finger to the right learning intersubjective area. For this purpose, researchers classify the use of inflammation into five general categories (Figure 6):

Type 1. "Inflammation segregates external agents/effect" (i.e. physiological function)
Type 2. "The inflammation has a diameter of 5 cm." (i.e. portion of a body part)
Type 3. "The inflammation has changed its shape" (i.e. abnormal morphology)
Type 4. "The inflammation evolved during three weeks" (i.e. clinical condition)
Type 5. "The inflammation is severe" (i.e. diagnosis)

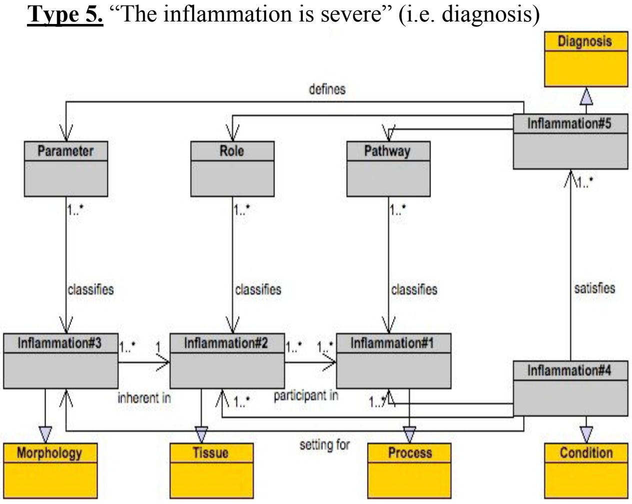

Fig. 6: The Inflammation Categories.

The various categories of inflammation contribute to the inflammation ontology. Figure 7 illustrates the inflammation ontology used in our Bio-SAmI prototype:

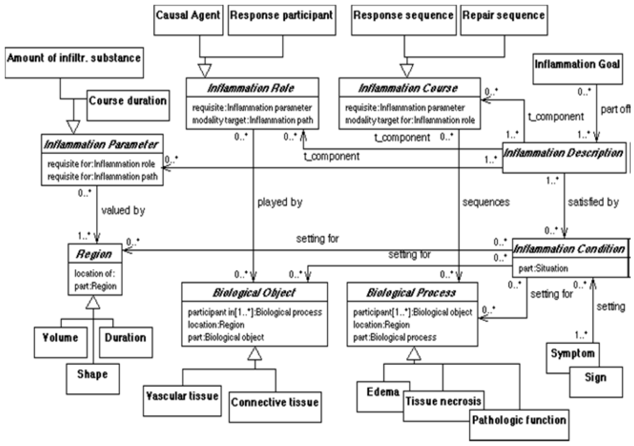

Fig. 7: Inflammation Ontology [28].

Although there are many software tools that can be used to extract the metadata from the DICOM SR file (e.g. ExifTool (see www.sno.phy.queensu.ca/~phil/exiftool/)), there is no effective application that can utilize such metadata for searching and identifying relevant DICOM objects. Actually, using only metadata or ontology is not sufficient to solve the problems of object accessibility [29]. Consequently, we see a variety of silo implementations today making it almost impossible to achieve effective classifying, indexing, retrieval and process information about DICOM objects. Indeed, these problems will be solved only if the biomedical system and the learners use a common meaning for metadata values. However, it is natural that users use variety of concepts synonyms and alternatives as well as using the same concept in a different context. These criterions are called "folksonomy" and "Polysemy" and they play important role in the Web 2.0 [30]. For biomedical learning, dealing with ontology, polysemy or folksomomy is of great importance. Making bonds and relations such as content, sequencing, and grouping between objects and concepts need to be included to make it possible, not only to carry out automatic tasks on these objects, but also to produce new knowledge from what already exists. For example, "Inflammation" is a typical polysemous word with many additional synonyms (e.g. symptom, burning, combustion, rousing, arousal). The designed Bio-SAmI is part of an ongoing project for developing a P2P learning framework for sharing and searching of biomedical learning objects [31,32,33,35]. The Bio-SAmI includes a metadata management actor (MMA) that takes the responsibility of identifying relating and grouping similar concepts. It is different from the traditional research of mapping ontologies where the main goal is on finding ontology alignment [34]. In the MMA heuristics are used for identifying corresponding concepts based on DICOM ontology, polysomy or folksonomy and checking the closeness of two concepts in the concept hierarchy and creating a common RDFS (www.w3.org/TR/rdf-schema/) ontology structure. The MMA uses a new approach that utilizes an argumentation process to prove any two concepts posses certain degree of contextual similarity [36,37]. Our approach is based on such argumentation process employing several context-matchers for identifying complex and sensitive DICOM biomedical relationships between any two FOAF persons concepts. The context-matchers represent tiny reasoners employing inferential rules written in tuPrologME (www.alice.unibo.it/xwiki/bin/view/Tuprolog/TuprologMe) that is integrated with the JADE-LEAP-SALSA environment.

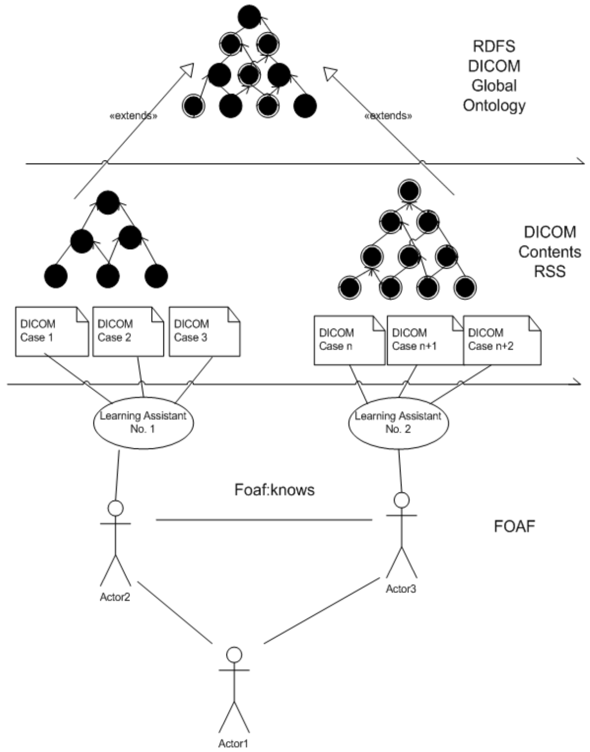

Fig. 8: Bottom-up Approach for Global Ontology.

If the learning assistant actor requested a DICOMCasesClassifier service then the findings of these context-matchers are feed to the JUNG API (**jung**.sourceforge.net/doc/index.html) for visualizing the DICOM biomedical cases for a FOAF social network. The DICOMCasesClassifier actor employs an argumentation engine that utilizes Prolog inference rules expressed by the tuPrologME to activate series of context-matchers. This engine accepts arguments from the requesting learning assistant actor and tries to build a global ontology from relevant case studies available at his/her FOAF social network. The rules of the argumentation engine are written to yield certain required classifications according to the parameter supplied by the learning assistant actor. For example, if the learning assistant actor requested to classify the available DICOM case studies at his/her FOAF social network according to the "inflammation" parameter then the argumentation engine based on the generated global ontology will classify every FOAF case studies according to the five inflammation categories described earlier. Figure 9 shows the algorithm used by the argumentation engine.

```
import alice.tuprolog.lib
public class ArgumentationEngine {

public boolean isType(Ontology globalOntology, Ontology
caseOntology) {
if(caseOntology.getDescriptor() == "physiologicalFunction")
                caseOntology.setType =
globalOntology.getType(1);
        else if (caseOntology.getDescriptor() ==
"inflammationDiameter")
        caseOntology.setType = globalOntology.getType(2);
        else if (caseOntology.getDescriptor() ==
"inflammationShape")
        caseOntology.setType = globalOntology.getType(3);
        else if (caseOntology.getDescriptor() == "evloutionSpeed")
                caseOntology.setType =
globalOntology.getType(4);
        else if (caseOntology.getDescriptor() ==
"inflammationSevere")
        caseOntology.setType = globalOntology.getType(5);

public boolean compareOntologies(Ontology O1, Ontology O2) {
    // Compare all schemas in O1 and O2 to see if they match
        boolean conceptSchemaMatch = true;
        boolean predicateSchemaMatch = true;
        boolean agentActionSchemaMatch = true;
    /* Try to prove the first argument on Concept Match */
        for (int i = 0; i < number of concept schemas; i++) {
                if(O1.getConceptSchema(i) !=
    O2.getConceptSchema(i))
                        conceptSchemaMatch = false;
        }
    /* Otherwise try to prove the Second argument on Predicate
    Match */
        for (int i = 0; i < number of predicate schemas; i++) {
                if(O1.getPredicateSchema(i) !=
    O2.getPredicateSchema(i))
                        predicateSchemaMatch = false;
        }
    /* Finally try to prove the Third argument on Action Match if the
    previous two fails*/
            for (int i = 0; i < number of concept schemas; i++) {
                if(O1.getSchema(i) != O2.getSchema(i))
                        agentActionSchemaMatch = false;
        }
        if (conceptSchemaMatch = true && predicateSchemaMatch
    == true && agentActionSchemaMatch == true) {
        return true
    }
    else {
    /* The two agents does not share common ontology */
        return false;
    }
    public boolean Part_Off(Ontology O1, Ontology O2) { ...}
    public boolean Contained_In(Ontology O1, Ontology O2) { ...}
    ....
        }
}
```

Fig. 9: The Argumentation Engine Methods.

Figure 10 shows the result of the running our Bio-SAmI prototype on a pool of FOAF related eight learning assistant actors, with one learning actor requesting classification to the available DICOM case studies based on the inflammation categories.

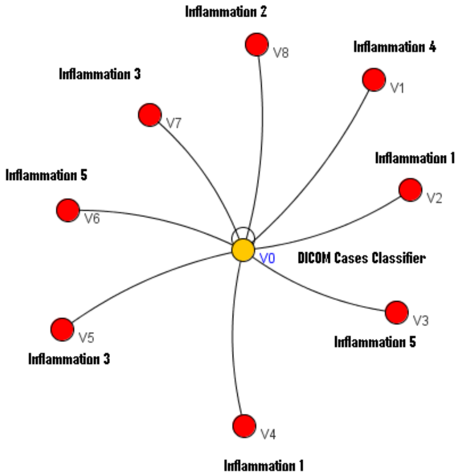

Fig.10: Using Bio-SAmI for Categorizing 8 Inflammation Based DICOM Case Studies.

Figure 11 illustrates a block diagram of the major components involved at the Bio-SAmI system.

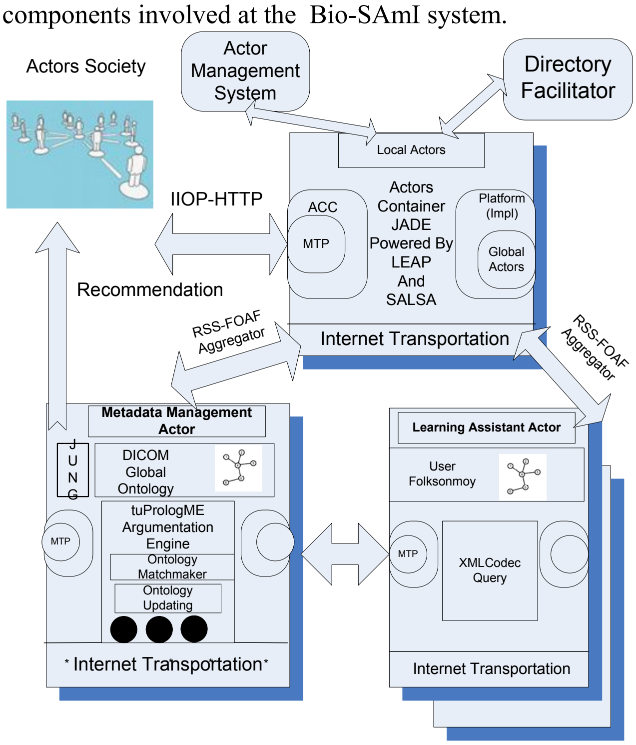

Fig. 11: The Overall Bio-SAmI Framework.

V. CONCLUSIONS

This article introduces an overview for the design of a social ambient intelligence system (BioSAmI) capable of serving variety of user devices and providing transparent services to social network of biomedical learners. The BioSAmI architecture provides a context-aware open environment where mobile users can enter and leave. BioSAmI is based on the Actor model that treats mobile users or "actors" as the universal primitives of computation: in response to a message that it receives, an actor can make local decisions, create more actors, send more messages, and determine how to respond to the next message received. The actor model is implemented in SALSA by extending the FIPA agent programming environment JADE and incorporating the LEAP API for supporting mobile actors. Moreover, Bio-SAmI is social network centric and context-aware. It enables variety of learning assistant actors to acquire and share biomedical case studies expressed in DICOM SR for a FOAF-based social network. Furthermore, Bio-SAmI provides variety of ambient services including syndicating a new DICOM case study using the RSS protocol, aggregating a DICOM case study using the RSS protocol, tracing a FOAF friend, finding a DICOM case study as well as classifying the DICOM case studies that are available on the FOAF social network.

The Bio-SAmI is currently able to classify DICOM case studies that are related to the concept of "inflammation" by incorporating an ontology for inflammation and a tuPrologME rule-based argumentation engine for resolving polysonmy and folksonomy issues related to inflammation.

VI. ACKNOWLEDGMENTS

This research is supported by the NSERC Discovery Grants of both first and second authors.

REFERENCES

[1] E.H.L. Aarts, J.L Crowley, B. de Ruyter, H.Gerhäuser, A. Pflaum, J. Schmidt, R. (Eds.) Wichert, "Ambient Intelligence", Proceedings of Conference AmI 2008 Conference, Nuremberg, Germany, November 19-22, 2008. Lecture Notes in Computer Science, Vol. 5355 2008, Springer Verlag, ISBN: 978-3-540-89616-6,

[2] F. Davide, P. Loreti, M. Lunghi, et al, "Communications through virtual technologies". In: E. Gregori, G. Anastasi, S. Basagni, (eds.), Advanced lectures on networking. Berlin: Springer-Verlag, pp. 124–154, 2002.

[3] K. Sato, R. Bruder, "Multi-Aspect User Interaction Models for Distributed Robotic Systems: Collaborative Activity Generation", Robot and Human interactive Communication, 2007. RO-MAN 2007. The 16th IEEE International Symposium on, Aug 26-29, 2007 Page(s):87 – 92

[4] J. Bohn , V. Coroamă , M. Langheinrich , F. Mattern and M. Rohs, "Social, Economic, and Ethical Implications of Ambient Intelligence and Ubiquitous Computing", In Ambient Intelligence, Editors Werner Weber, Jan M. Rabaey and Emile Aarts,Springer Berlin Heidelberg, 2005

[5] Jörg Cassens, "Explanation Awareness and Ambient Intelligence as Social Technologies", PhD Thesis, 2008, Norwegian University of Science and Technology, Faculty of Information Technology, Mathematics and Electrical Engineering, Department of Computer and Information Science.

[6] C. Rizopoulos, "An activity-based perspective of interacting with ambient intelligence systems", Intelligent Environments, 2007. IE 07. 3rd IET International Conference on, Volume , Issue , 24-25 Sept. 2007 Page(s):81 – 88

[7] M. Friedewald and O. Da Costa, "Science and Technology Roadmapping: Ambient Intelligence in Everyday life", European Science and Technology Observatory, JRC/IPTS-ESTO Study, June 2003,

http://forera.jrc.ec.europa.eu/documents/SandT_roadmapping.pdf

[8] G. M. P. O'Hare, M. J. O'Grady, S. Keegan, D. O'Kane, R. Tynan, and D. Marsh, “Intelligent Agile Agents: Active Enablers for Ambient Intelligence”. ACM's Special Interest Group on Computer-Human Interaction (SIGCHI), Ambient Intelligence for Scientific Discovery (AISD) Workshop,Vienna, April 25, 2004

[9] C. Miller, H. Funk, P. Wu, R. Goldman, J. Meisner, and M. Chapman, “The playbook approach to adaptive automation”, In Proc. Human Factors and Ergonomics Society’s 49th Annual Meeting, (2005)

[10] P. Goodison, P. Johnson, J. Thoms, “Improving Human Interaction with Autonomous Systems: Supporting Intentionality through Increased Awareness”, Human Interaction with Intelligent & Networked Systems, IUI 2009 Workshop, February 8th, 2009, Florida

[11] C. Bryce , C. Razafimahefa5 and M. Pawlak Lana: “An Approach to Programming Autonomous Systems”, Book Series Lecture Notes in Computer Science Publisher Springer Berlin / Heidelberg ISSN 0302-9743 (Print) 1611-3349 (Online) Volume 2374/2002 Book ECOOP 2002 — Object-Oriented Programming DOI 10.1007/3-540-47993-7 Copyright 2002

[12] J. Dedecker ,T. Van Cutsem, S. Mostinckx, T. D'Hondt ,W. De Meuter, “Ambient Oriented Programming”, Conference on Object Oriented Programming Systems Languages and Applications archive Companion to the 20th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications table of contents San Diego, CA, USA SESSION: Onward! table of contents Pages: 31 - 40 Year of Publication: 2005 ISBN:1-59593-193-7

[13] Programming dynamically reconfigurable open systems with SALSA (2001) C Varela, G Agha, ACM SIGPLAN Notices. OOPSLA’2001 Intriguing Technology Track Proceedings

[14] W. Wang, and C. A. Varela, “Distributed Garbage Collection for Mobile Actor Systems: The Pseudo Root Approach”, Book Series Lecture Notes in Computer Science, Publisher Springer Berlin / Heidelberg , ISSN 0302-9743 (Print) 1611-3349 (Online) ,Volume Volume 3947/2006, Book Advances in Grid and Pervasive Computing, Copyright 2006

[15] K. El Maghraoui, T. J. Desell, B. K. Szymanski, C. A. Varela, “The Internet Operating System: Middleware for Adaptive Distributed Computing” , International Journal of High Performance Computing Applications, Vol. 20, No. 4, 467-480 (2006)

[16] T. O'Reilly, “What Is Web 2.0 Design Patterns and Business Models for the Next Generation of Software”, 09/30/2005, http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html

[17] L. Shklar, “RDF Models for Dynamic Syndication and Wireless Applications”, Proceedings of the Second International Workshop on the Semantic Web - SemWeb'2001 Hongkong, China, May 1, 2001

[18] C. H. Felicíssimo, “Dynamic Contextual Regulations in Open Multi-Agent Systems”, ISWC2006 Conference, Athens, GA,USA

[19] P. Stone and M. Veloso, “Multiagent Systems: A Survey from a Machine Learning Perspective”, Journal Autonomous Robots, Publisher Springer Netherlands, ISSN 0929-5593 (Print) 1573-7527 (Online), Issue Volume 8, Number 3 / June, 2000

[20] P. Bresciani, P. Giorgini, F. Giunchiglia, J. Mylopoulos, and A. Perini. Tropos: “An Agent-Oriented Software Development Methodology”, Autonomous Agents and Multi-Agent Systems,8(3):203–236, July 2004.

[21] L. Padgham and P. Lambrix, “Formalizations of Capabilities for Bdi-Agents. Autonomous Agents and Multi-Agent Systems”, 10:249–271, 2005.

[22] M.W. Jang , A. Ahmed and G. Agha, “Efficient Agent Communication in Multi-agent Systems”, Book Series Lecture Notes in Computer Science, Publisher Springer Berlin / Heidelberg, ISSN 0302-9743 (Print) 1611-3349 (Online), Volume Volume 3390/2005, Book Software Engineering for Multi-Agent Systems III , Copyright 2005, ISBN 978-3-540-24843-9

[23] M. Luck, P. McBurney, C. Preist, “Agent Technology: Enabling Next Generation Computing”, A Roadmap for Agent Based Computing, 2003

[24] M. Jeffrey Bradshaw, “KAoS: An Open Agent Architecture Supporting Reuse, Interoperability, and Extensibility”, 1st Workshop on Knowledge Acquisition for Knowledge-Based Systems was held in Banff, Alberta, Canada in October 1986

[25] L. Bölöni , M. Ali Khan, X. Bai, G. Wang, Y. Ji and D. C. Marinescu, “Software Engineering Challenges for Mutable Agent Systems”, Book Series Lecture Notes in Computer Science, Publisher Springer Berlin / Heidelberg , ISSN 0302-9743 (Print) 1611-3349 (Online) Volume Volume 2940/2004 , Book Software Engineering for Multi-Agent Systems II Copyright 2004

[26] J. Sabater, C. Sierra, “Reputation and social network analysis in multi-agent systems”, Source International Conference on Autonomous Agents archive, Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1 table of contents, Bologna, Italy SESSION: Session 2D: group and organizational dynamics table of contents, Pages: 475 - 482 Year of Publication: 2002 , ISBN:1-58113-480-0 ,

[27] Y.H. Wu, S.H. Lin, J.J. Lun, Y.J. Lin, F.C. Chong, “An E-learning Content Authoring tool for Transforming DICOM into SCORM”, Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference, Volume, Issue, 2005, Page(s):2890 – 2893

[28] A. Gangemi, C. Catenacci, Battaglia, M. Inflammation Ontology Design Pattern: an Exercise in Building a Core Biomedical Ontology with Descriptions and Situations”, in Pisanelli D. (ed.), Biomedical Ontologies, IOS Press, Amsterdam, 2004.