

MULTIAGENT SYSTEM FOR SCRUTINY OF ADMISSION FORMS USING AUTOMATIC KNOWLEDGE CAPTURE

Puneet Modgil¹ and M. Syamala Devi²

¹Department of Computer Science and Applications,
Panjab University, Chandigarh, India
Punmod@rediffmail.com

²Department of Computer Science and Applications,
Panjab University, Chandigarh, India
Syamala@pu.ac.in

ABSTRACT

Multiagent Systems are autonomous intelligent systems. In many academic institutions student admissions are performed after generating merit lists. Generation of merit lists is preceded by manual scrutiny of admission forms. This manual scrutiny is a knowledge-intensive, tedious and error-prone task. In this paper the design, implementation and testing of Multiagent System for Scrutiny of Admission Forms (MASAF) using Automatic Knowledge Capture is presented. MASAF consists of three agents namely: Form agent, Record agent, and Scrutiny agent. These three agents, using ontology, cooperatively fulfill the goal of highlighting the discrepancies in filled forms. MASAF has been tested by scrutinizing about 1000 forms and all of discrepancies found were correct as verified by human scrutinizer. Thus it can be concluded that using Multiagent system for scrutiny of forms can reduce human intervention, improve performance in terms of speed and accuracy. The system can be enhanced to automatically correct the discrepancies in forms.

KEYWORDS

Multiagent system, Knowledge Capturing, Scrutiny of admission forms, Ontology

1. INTRODUCTION

An Agent is a software entity that can automatically perceive environment and can behave reactively as well as proactively with social ability. Multiagent System is distributed artificial intelligence system in which a number of agents interact with each other to achieve the desired objectives. A global solution arises from cooperation amongst these agents but no one agent is capable of solving the problem. Thus agents are well suited for solving the problems which are knowledge intensive and distributive in nature.

In academic institutions before actual admissions take place a merit list of admission seekers is generated. Merit list is generated on the basis of certain parameters such as marks in qualifying

examination. These parameters are course specific. The information about these parameters is obtained from the forms. The next step is to verify the information provided in the forms. This scrutiny is done by responsible persons of institution on the basis of documents attached along with forms. Thus scrutiny is a tedious, error-prone and knowledge intensive task. There is a need to automate the above task and this is feasible by employing agents to capture knowledge automatically from the web and to compare with the information available in the forms. Such an automated system based on agents with knowledge capturing capabilities is presented in this paper and referred to as MASAF in the rest of the paper. Section 2 highlights the review of literature. Design of MASAF is presented in section 3. Section 4 includes implementation details, testing and results. Section 5 contains conclusions and scope for future work.

2. RELATED WORK

In this section work performed by many researchers related to Multiagent system and applications in academics and knowledge capture is reviewed and presented.

The authors in [1] present knowledge and information network approach for managing research assets in a knowledge-oriented organization using a Multiagent system. The integration of existing knowledge and information technologies to develop new ways to manage scientific and technical knowledge in modern organizations is the challenge addressed in this article.

Multi-Agent System for University Course Timetable Scheduling: MAS_UP-UCT is described in [2]. The problem is divided into two sub problems: 1. Faculty course timetable scheduling (which involves only allocation of course day and time), and 2. University course rooms allocation (which involves allocation of rooms for courses). Four types of agents are used by the system: Main Scheduler Agent (MSA), Faculty Scheduler Agent (FSA), Expert Assistant Agent (EAA), and Personal Agent (PA). Each faculty has a FSA which has to schedule the courses of that faculty.

The details of Campus Automation Web Information System (CAWIS) project is discussed in [3]. The Multi agent System is used for course planning, resource allocation, class timetabling, and office allocation, exam scheduling and other administrative and organizational activities of the university. The six basic agents defined are: (i) Students Agent (ii) Training and Learning Agent (iii) Lecturer Agent (iv) Infrastructure Agent (v) Task Agent (vi) Strategical / Goal Agent .

In [4] authors explain how wrappers are able to accept a query against the source and return a set of structured results, thus enabling applications to access web data in a similar manner to that of information from databases. A significant problem in this approach arises as Web sources may undergo changes that invalidate the current wrappers. Thus authors have presented novel heuristics and algorithms to address this problem.

In [5] Authors present their experience in building Information Retrieval (IR) systems using Multiagent technology. In particular they presented their results in using X.MAS, a generic Multiagent architecture aimed at retrieving, filtering and reorganizing information according to user interests.

3. DESIGN OF MULTIAGENT SYSTEM FOR SCRUTINY OF ADMISSION FORMS (MASAF)

In this section roles and responsibilities of agents, ontology design, agents' design and agent interaction are discussed.

3.1 Agents and their Responsibilities

Based on requirement to achieve goals as specified in goal diagram (created using agent tool [6]) shown in Figure 1, three agents are defined in MASAF model: Form agent, Record agent and Scrutiny agent. These three agents together perform the job of persons who are responsible for checking a list of application forms for validity of various fields by verifying certificates, marks lists and other documents and pointing out the discrepancies observed.

The Multiagent model of MASAF is shown in Figure 2. MASAF mainly consists of three agents but fourth agent gateway agent is kept as mediator between user and system thus it is not considered as part of the system itself. In the model the capabilities possessed by all the three agents are shown with 'possess' tag. Three agents are chosen because there are three cohesive tasks which need to be performed namely: retrieving applications, capturing knowledge about key fields and comparison of information. Individual responsibilities of these three agents are as follows:

Form Agent:

This agent retrieves the applications one by one from the website (in case third party is employed), or database of the organisation which is going to admit the students and save it in the database specifically created for MASAF. The key fields like roll number of qualifying examination, year of passing are communicated to the next agent i.e. record agent using Agent Communication Language (ACL) messages.

Record Agent:

Based on the key field values received from form agent the record agent logs on to relevant websites such as the board/university in which student studied. From these websites record agent captures knowledge about various attributes of the candidate like total marks, subjects studied etc. These attributes contribute to assignment of different weightages in the creation of merit list. The captured knowledge is kept in database as well as communicated to scrutiny agent using ACL messages.

Scrutiny Agent:

The scrutiny agent performs the comparison of information in the application forms, stored in local database, with that communicated by record agent. On the basis of comparison Scrutiny agent points out discrepancies (if any) present in the application forms.

Thus the scrutiny of applications by accessing relevant data, verifying and highlighting discrepancies is performed with the cooperative and joint efforts of all the three agents in MASAF.



Figure 1. Goal diagram for MASAF

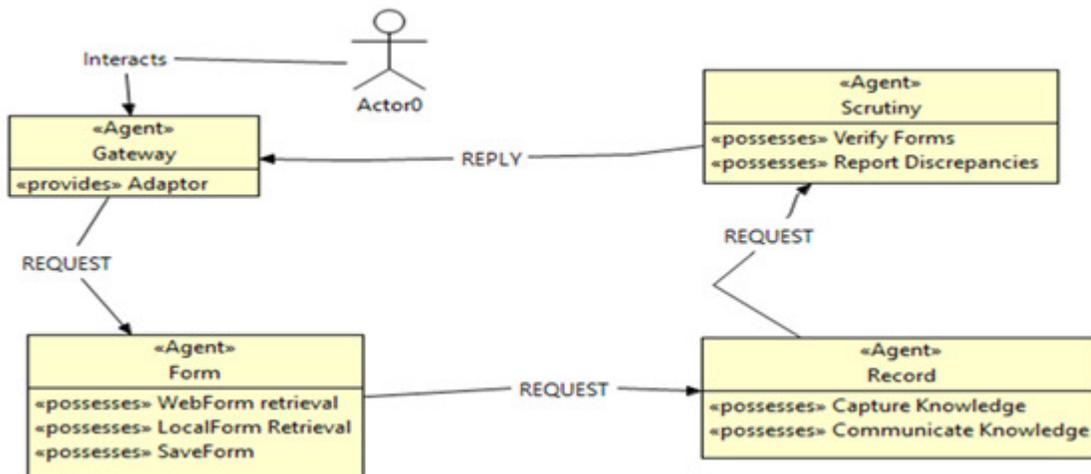


Figure 2. Agent Model for MASAF

3.2 Ontology Design

In the context of knowledge sharing and agent communication, the term ontology is used as a specification of a conceptualization. That is, an ontology is a description (like a formal specification of a program) of the concepts and relationships that can exist for an agent or a community of agents. In MASAF Ontology is stored in a frame-based knowledge model [7]. This model consists of classes i.e. concepts in the domain of discourse, slots i.e. properties or attributes of these classes and Facets which describe properties of slots. The design of the ontology for MASAF is called AcademicOntology. Conforming the model the following Agent Actions are defined:

- StartRetrievingForms, i.e. start retrieval of forms from website.
- StartRetrievingRecords i.e. start capturing knowledge about admission seekers.
- StartScrutiny i.e. start finding discrepancies in forms (if any)

The agent actions use both concepts and relations. The following primitive concepts are defined:

- Form: An entity which contains entire information about individual admission seeker.
- PrimInfo: Primitive Information about admission seeker required to capture knowledge about him/her from web.

- CourseWeightage: Contains information about which course has weightage related to which particular subject of previous class.
- Discrepancy: Contains information about kind of discrepancy (if any) present in filled form.
- CaptKnowledge: Contains knowledge captured about particular candidate from web.

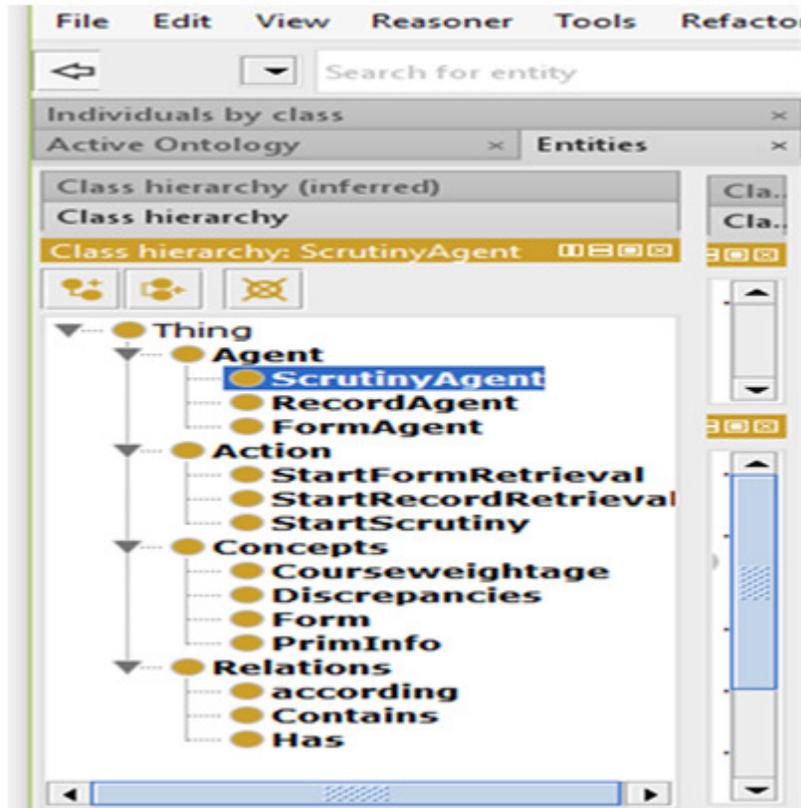


Figure 3. Partial Ontology for MASAF

The following relations are defined: ‘According’ between Form and Course i.e. form is given weightage according to particular course. ‘Has’ between Form and Discrepancy i.e. form has a particular discrepancy. ‘Contains’ between Form and PrimInfo. The ontology classes developed using protégé tool are shown in Figure 3.

3.3 Agent Design

The design of three agents internal to the system along with design of gateway agent which is external to the system is separately presented as follows:

Form Agent: Form agent is comprised of one shot behavior which is triggered whenever user asks to start the scrutiny. Form agent’s one shot behavior when triggered goes to website or database of organisation to find out whether any new forms have arrived and if they have, it retrieves those forms and store them in system specific database. Form agent decides whether a

form is new or not by checking whether form with same form number is present in system's database already or not. However while retrieving forms form agent keeps sending values of key fields to record agent so that it can keep working in parallel.

Record Agent: This agent implements the cyclic behavior i.e. whenever form agent sends key fields this agent starts its work on the basis of these key fields. Firstly this agent on the basis of previous board/university wraps up the website of that board/university. Wrapping simply means controlling the website by means of code rather than by human [4]. After that it retrieves detailed record of the student from website on the basis of roll number. Detailed record retrieved contains knowledge on the basis of which scrutiny takes place. The attributes in detailed record include total marks, subjects studied, subjects passed and so on. In order to retrieve information about various attributes of student this agent has to parse the detailed record page of different board websites. Parsing is required because data presented in websites is semi-structured in nature. So agent makes use of defined algorithm to capture required knowledge from semi-structured information in web page. This knowledge is passed to scrutiny agent as well as saved in system's database for future reference.

Scrutiny Agent: This agent implements cyclic behavior, i.e. whenever request for scrutiny comes this agent executes behavior of verifying information in application form with that provided by record agent. If there are any discrepancies they are reported to the user and also saved in database for future reference.

Gateway Agent: This agent acts as a mediator between user interface and MASAF. This agent passes requests made on user interface to the system by converting the request into proper ACL messages and returns results to user interface from MASAF which are in different format than to be shown on user interface.

3.4 Agent Interaction

The three agents in MASAF communicate with each other using Agent communication Language (ACL) messages on the basis of predefined ontology. ACL is a specification by Foundation for Intelligence Agents (FIPA) for intercommunication of agents.

In MASAF model firstly gateway agent sends ACL message to form agent with intention REQUEST, action specifying 'StartFormRetrieval' and content containing ontology concept 'CourseWeightage' class which specify course for which scrutiny to take place.

Further Form agent sends ACL message to Record agent with intention REQUEST, action specifying 'StartRecordsRetrieval' and content containing ontology concept 'PrimInfo' which contains key fields.

In next interaction record agent sends ACL message to scrutiny agent with intention REQUEST, action specifying 'StartScrutiny' and content containing ontology class 'CaptKnowledge' which contains values of key attributes. On the basis of provided information scrutiny agent verifies application form and saves discrepancies and sends them to gateway agent as ACL message with content having 'Discrepancy' ontology class. The gateway agent finally retrieves information from this class and shows it on user interface. Interaction amongst agents expressed in Agent Unified Modelling Language is shown in Figure 4.

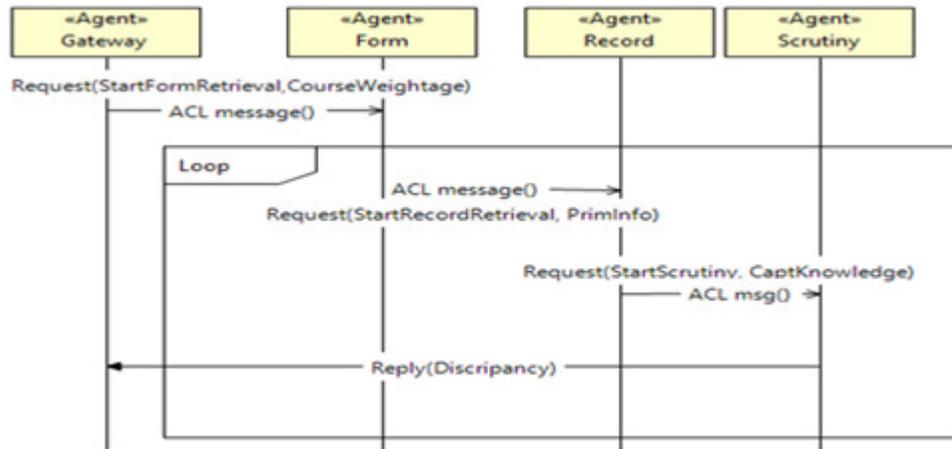


Figure 4. Interaction diagram of agents

4. IMPLEMENTATION

In section 4.1 tools and techniques used for creation of MASAF are discussed and in section 4.2 testing and results are presented.

4.1 Tools and Techniques

Agents are created in Java Agent Development Environment (JADE). User interface of the system is provided by means of Java Enterprise Edition (Java EE) Servlets. JADE Gateway agent is working as a mediator between Java EE user interface and multiagent system created on JADE platform. Agents Communicate with each other using messages created using Agent Communication Language (ACL) framework which is available with JADE.

Form Number	Student Name(in form)	Percentage(in form)	Percentage(in dmc)	Weightage(in form)	Weightage(based on dmc)	GrandPercentage(in form)	GrandPercentage(in dmc)	Difference
A50711	NIKHIL SHARMA	45.11	45.11111	5	0	50.1	45.11111	Only Weightage is wrong
A19404	SAVERA KUMAR SINGH	45.4	45.4	5	0	50.4	45.4	Only Weightage is wrong
A18389	PRIYANKA THAKUR	45.6	45.6	10	5	55.6	50.6	Only Weightage is wrong
A44088	PARSHANT MEHRA	42.0	46.0	5	0	47.0	46.0	Percentage and weightage are wrong
A21361	KRISHAN	47.0	47.0	5	0	54.0	47.0	Only Weightage is wrong

Figure 5. Results of Scrutiny

In order to retrieve data from websites both form and record agent use similar technique. This technique involves wrapping the website [4] and parsing the semi-structured data in HTML and

retrieving the required information from it. Tool used for wrapping the websites is java package Selenium. For efficient parsing of semi-structured data JavaSoup package for java is used. The user interface provided by the system is web-interface.

Thus requests can be made to system and results can be seen from system through a web browser. Agents run in the backend on the server in order to deliver the results on the user Interface. A screenshot of User Interface is shown in Figure 5. Analysis and design of MASAF is depicted using agenttool [6]. Ontology is developed using open source software tool protégé [7].

4.2 Testing and Results

MASAF is tested for the admissions of Bachelor of computer Applications (BCA) an undergraduate course run in Government colleges, Chandigarh, India. 997 student applications are processed by the system and discrepancies are highlighted correctly. MASAF with the help of Form Agent read all the filled in forms and saved them in local database according to pre-specified ontology. After retrieving necessary information from the form, record agent retrieved information of all admission seekers from three different online sites, namely: Central board of School Education (CBSE), Haryana Board of School Education (HSEB), and Punjab School Education board (PSEB).

The sample results of scrutiny are shown in Figure 5 highlighting the discrepancies. For example student with form number 'A50711' and name Nikhil sharma has filled different weightage in form than captured from the web. Weightage (in form) column specifies weightage mentioned in the form, Weightage (based on dmc) column specifies weightage on the basis of knowledge captured from the website and discrepancy is specifically mentioned in difference column as for this instance 'only weightage is different'.

5. CONCLUSIONS AND SCOPE OF FUTURE WORK

Multiagent system due to the properties of proactive, reactive and social ability can perform a number of knowledge intensive tasks without much human intervention. The MASAF performed scrutiny of 997 applications in about one hour which otherwise would have completed by seven persons in one week. Thus the system is dependable and efficient. The system can be extended to rectify any discrepancies if found in forms thus generating the merit list automatically. For this correction agent could be introduced with required functionality which would remove all the discrepancies in the forms.

REFERENCES

- [1] Cantú j, Ceballoss G. "A multiagent knowledge and information network approach for managing research assets". *Expert Systems with Applications*. Elsevier 2010; 37:5272–5284.
- [2] Oprea M. "MAS_UP-UCT: A Multi-Agent System for University Course Timetable Scheduling". *International Journal of Computers, Communications & Control* 2007; 2:904-102.
- [3] Taskin H, Gozetepe K, Taskin M, Canay O. "Agent Based University Planning of Sakarya University: "CAWIS" Project". *Journal of Theoretical and Applied Information Technology* 2008; 4:753-759.
- [4] Raposo J, Pan A, Ivarez A, Hidalgo J. "Automatically maintaining wrappers for semi-structured web sources", *Data & Knowledge Engineering* 61 (2007) 331–358

- [5] Addis A, Armano G, Vargiu E. “Multiagent systems and information retrieval our experience with X.MAS”. Expert Systems with Applications. Elsevier 2012; 39:2509-2523.
- [6] Juan C. Garca-Ojeda, Scott A. DeLoach, and Robby, agentTool III: From Process Definition to Code Generation, Proc. of 8th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009), Decker, Sichman, Sierra, and Castelfranchi (eds.), May, 10–15., 2009, Budapest, Hungary, pp. 1393-1394.
- [7] Gennari J, Musen M, et al. “The evolution of Protégé: an environment for knowledge-based systems development”. International journal of Human-Computer studies 2003; 58(1):189-213.
- [8] Tu SW, Eriksson H, Gennari JH, Shahar Y, Musen MA. “Ontology-based configuration of problem-solving methods and generation of knowledge-acquisition tools: application of PROTEGE-II to protocol-based decision support.” Artif. Intell. Med. 1995 Jun; 7(3):257-89.

AUTHORS

Puneet Modgil is a research scholar in the Department of Computer Science and Applications, Panjab University (PU), Chandigarh. He has served as Assistant Professor in Government College, sector - 11, Chandigarh, and Panjab University, Chandigarh. His Academic qualifications include M.Sc (Information Technology) M.Phil (Computer Science and Applications). He has teaching experience of 10 years.



Dr. M. Syamala Devi is a professor in the department of Computer Science & Applications, Panjab University, Chandigarh. She received her Ph.D degree in Computer Science & Systems Engineering from Andhra University, Visakhapatnam and M.E. in Computer Science & Engineering, from NIT, Allahabad. She has completed M.Sc in Applied Mathematics from Andhra University, Visakhapatnam. Before joining Panjab University, she served Indian Space Research Organization, Sriharikota, and National Institute of Technical Teachers' Training & Research, Chandigarh. Her areas of expertise include algorithms, Image Processing, Distributed Artificial Intelligence and Educational Computing.

