Volume 8, No. 9, November-December 2017



International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

MULTIAGENT SYSTEM FOR MANAGEMENT AND EVALUATION OF EXAMINATIONS

Himani Mittal GGDSD College, Chandigarh,India

Abstract: Several efforts have been made to computerize Examination System. Agents are autonomous software components capable of making decision on behalf of other entities to realize desired goals. A Multiagent System is a collection of agents that cooperate to solve a problem. In this paper, a Multiagent System is designed, implemented and tested for Management and Evaluation of Examination (MASMEE). Examination can be subjective, objective or practical based on programming languages. The functions of paper setting, evaluation of answers' and result compilation are performed by intelligent agents. MASMEE mainly consists of two agents: Examiner Agent (EA) and Student Agent (SA). EA sets the paper, evaluates the submitted answers and compiles the results. The SA enables the students' to take the examination by providing question paper and collecting the answers. The paper setting is performed using a database of questions. A web based interface is developed for conduct of examination. The evaluation of subjective and objective examinations is performedbased on techniquesof information retrieval and Ontology. Practical evaluation is done using 4-stage framework, evaluating on the basis of compilation, functional testing, metrics, efficiency and similarity of programs to standard program. The MASMEE system is implemented using Java andJava Agent Development Environment (JADE). Testing is performed by conducting tests with post-graduate students. The scores generated by MASMEE are compared with scores assigned by human examiner using correlation and are found to be satisfactory. It is concluded that MASMEE system can help to reduce the effort of human examiner.

Keywords: Multiagent System (MAS), Subjective Evaluation, Information Retrieval techniques, Practical Evaluation,

1. INTRODUCTION

The Evaluation of performance of students is performed by conducting Subjective, Objective and Practical Examinations. The nature of Subjective and Objective examination is same for all subjects. For Computer Science, Practical Examinations includes assessment of programs written by students.

Examination System consists of several activities- paper setting, conduct of examination, evaluation and result declaration. The conduct of examination itself involves a lot of planning, scheduling and resources (both human and tangible).

In subjective examination, the paper setting is started much in advance before the actual conduct of examination. There are format constraints, time constraints and difficulty constraints that must be kept in mind while designing the questions for test paper. After question paper is finalized, it is kept in top secrecy. The students write answers according to Test paper questions. The evaluator evaluates the answers and results are declared. In this complete process, there is a lot of physical movement of answer-books. There is a limitation on the quality of evaluation due to time constraint of expert examiners. The evaluation itself is cumbersome as the marks allotted by two examiners for the same answer may be different.

Practical examination starts by students assembling at examination center. The examiner asks students to make computer programs on specific problems. At the end examiner personally checks each student program. The compilation errors, correct output and efficiency are used to evaluate programs. There are many programs to evaluate and time is limited. This leads to compromise in quality of evaluation.

The computerization of management, conduct and evaluation of examinations can help to reduce the time, effort

© 2015-19, IJARCS All Rights Reserved

and increase accuracy of results [1]. Paper setting is faster and error free. It is convenient in terms of time and effort to edit an auto-generated paper than to make fresh test paper. Evaluation through computers using intelligent techniques uses same inference mechanism for all the students, so there is uniformity in marking scheme. The computer can work diligently as it is electronic device and therefore, errors due to fatigue and lack of concentration are avoided. The time and effort involved in evaluation is saved and the human evaluators can just supervise evaluation process. This allows them to devote their precious time for research activities. There are savings in shipping, handling, and printing costs.

Agent is a software entity [2];[3] acting for another entity (e.g., person or agent) for achieving desired goals. Agent is autonomous, goal-oriented, reactive and social. Multiagent system (MAS)[4]involves multiple agents that cooperate for realization of predefined goals. Some of the applicationsof MAS are resource allocation [5], supply chain management [6], urban traffic management[7], e-commerce[8] and disaster management[9].

This paper includes design of Multiagent System for management and evaluation of examinations. The paper is organized as follows. Review of related work is included in Section 2. Section 3 contains the MASMEE design. Section 4 includes implementation details of MASMEE. Section 5 contains details of testing performed and discussion of results. Section 6 concludes that use of MASMEE system can greatly help in simplifying the examination system.

2. REVIEW OF RELATED WORK

The review of related work includes – review based on Multiagent examination systems, review based on test paper generation techniques, review based on Subjective Evaluation techniques and review based on Practical evaluation techniques and tools.

Review of Agent based Examination System: There are several web based architectures proposed for online conduct examination[10]–[12]. of objective Jamwal and Iyer[13]proposed Multiagent based examination system which creates workflow for activities like paper setting, evaluation, etc. However, these activities are performed manually by human experts. The role of agents is to carry the information from one place to another and create workflow.In[14], Moe Moe Aye and Mie MieThetThwin presented a mobile agent based solution for conduct of examination. It provides interface for performingactivities like setting up of paper, conduct and evaluation on different servers. Mobile agents synchronize the servers. In Gawali and Meshram,[15]multiagentbasedsystem was developed for conducting objective examination. There is a main agent responsible for conducting the examination. Static agents exist on client, server and database machine. Data is moved between servers using mobile agents.

In the papers reviewed, the focus is on conduct of examination and facilitation of paper setting and evaluation, which are performed manually. The use of mobile agents has been identified to have limitations like if single agent has to visit all the sights then static agents are better[16]. It was identified that in Examination System, the agents can be spared from carrying data to client side as no intelligent task needs to be performed at client side So, use of static agents is suggested in our work for examination system.

In [17], a review of several tools and techniques is presented for computerized evaluation of subjective, objective and practical examinations. Subjective Evaluation techniques found in literature are information retrieval techniques. Black box testing is performed for Practical evaluation of Computer Science examinations. Multiple choice questions are focused in Objective evaluation. Little work is done for white box testing of practical and one-line answers in Objective examinations.

Review Based on Test Paper Generation techniques : In [18], a web based test paper generation system is developed. The author categorizes the test paper generation as random question selection, backtracking method and AI based method. The AI based technique is proposed in Ant colony method for test paper generation[19] that ensures nonrepetition of questions but is time consuming. In [20], a multiagent system is developed for test paper generation. There are several agents like Teacher Agent, Exam Agent, and TP Agent. The input to the selection algorithm is number of questions, weight-age and difficulty level. The system then finds all the questions meeting the specified criteria. Then they use selection of first question in each category as question choice. It can lead to repetition of questions. The randomized selection algorithm [21]modifies the technique used in [20] and uses random function to generate the unique question number to be selected from all the questions selected by SQL query.

<u>Review of Subjective Evaluation Techniques and Tools:</u> Foltz et al. [22] applied mathematical technique called latent semantic analysis (LSA) to computerize evaluation in a tool called intelligent essay assessor (IEA). In this method a matrix is made with keywords to be searched as rows and documents as columns. The frequency of each word in each document is recorded. Then singular value decomposition is done on this matrix. The reported correlation between LSA and human-assigned grades varied from 0.59 to 0.89. The correlation between two human graders is from 0.64 to 0.84. So the performance of LSA and human graders is comparable. Perez et al. [23], [24] developed a system using latent semantic analysis (LSA) and BLEU (bilingual evaluation understudy) algorithm to essay evaluation. LSA performs semantic analysis and modified BLEU as used by the authors performs syntactic analysis. Linear equation is used to combine results of LSA and BLEU. However, the amount of weightage that should be given to BLEUgenerated score and LSA-generated score is not fixed. Author has shown multiple combinations and average success rate is 50 %. In [25], work done in [19] is extended by combining the scores generated by LSA and BLEU using fuzzy logic. This hybrid technique is able to identify discrepant answer from a regular answer. Some tools are used for performing evaluation in [25]. The WordNet tool [26] is a lexical ontology that is used for synonym search. Java-WordNet Interface (JWI) library [27] is provided by MIT and is used widely for connecting WordNet with Java programs. Porter's stemming [28] is used to reduce the word to its basic stem and helps in counting different forms of same word as one. Guava library [29] provided by Google is used to calculate frequency of words in text. MatLab Control library [30] provided by Google is used for connecting MatLab with Java code. C-rater [31]is another evaluation tool that uses Maximum Entropy technique (MaxEnt). It has 80 percent agreement with the score assigned by a humangrader for short answers. It finds lexicon similarity between phrases, provides for morphological analysis of the answers, matches the subject and predicates, matches the negative role with positive role. Protégé tool [32] and Jena Library [33] are used for implementation and use of Ontology.

Review of Practical Evaluation Techniques and Tools: In 2010, Skupas[34] wrote a paper on feedback improvement in automatic program evaluation systems. The technique used in all the tools for evaluation is black box testing. A more holistic approach is required for evaluating the student programs. Apart from black box testing, the program quality and semantic similarity of program with the algorithm needs to be established. Similarity checking of student program to model program can be done using System Dependence Graphs (SDG)[35]–[37]and Rapid Subgraph Calculation (RASCAL) algorithm[38]. Aframework for practical evaluation of C, C++ and Java programs is proposed in [39]. It evaluates on basis of syntactical correctness, logical correctness through unit testing (white box testing), style (metrics), efficiency (time and space complexity) and similarity to standard program (SDG and RASCAL).Unit Testing of programs is performed with the help of CUnit[40], CppUnit (MingW)[41] and JUnit[42]. System dependence graph[35] is generated using Frama-c [43]and SDG Library[44].

3. MASMEE DESIGN

The high-level designis shown in Figure 1. The MASMEE model is divided into three subsystems: Examiner Agent subsystem, Student Agent Subsystem and User Interface. There is one agent in each subsystem. Mainly, two agents are defined: Examiner Agent (EA) and Student Agent (SA). The EA is the heart of the MASMEE model. It has the intelligence to perform important tasks like paper setting, evaluation and assist the human examiner. The SA interacts

with the students, aids in conduct of examination and keeping track of student requests. User Interface is provided for students to interact with the Student Agent. Human Examiners interact with the Examiner Agent directly.In addition to EA and SA, Gateway Agent (GA) is used to make services of SA available to students through web interface.

The database of questions with model answers, student information and subject information is created using interfaceprovided by EA. The Human Examiner asks the Examiner Agent (EA) to generate test paper from database using specified paper format. Examination can be subjective, objective or practical based on programming languages. EA generates the paperandsavesit in he database. Student Agent (SA) proactively checks the database to find if an examination is scheduled for the day. The SA facilitates the login by authenticating the students. After successful login, the SA starts the examination and presents the paper to all the students through user interface (GA). After the students submit the answers, they are stored in database and EA is notified by SA. The EA performs the evaluation using intelligent techniques discussed in agent design section. The Gateway Agent is used as a link between Gateway Servlet and Student Agent. Web Pages are used by students for interacting with Gateway Servlet.

3.1. Roles and Responsibilities of Agents

The roles and responsibilities of MASMEE agentsare given below:

Examiner Agent (EA)

- Facilitates adding and deleting student login details, subjects and subtopics.
- Facilitates in creation of database of questions of all types: practical, subjective, and objective (MCQ, Fill-in-the-Blanks, and One-Line answers questions).
- Facilitates the examiner in setting the test paper for practical and theoretical examination using database.
- Facilitate and assist in evaluation of subjective tests.
- Evaluates the theoretical objective tests.
- Evaluates the practical programming language based tests.
- Generates explanation justifying the score assigned to students for each answer.



Figure 1. MASMEE Design

Student Agent (SA)

- Authenticate the user.
- Responsible for distribution of question papers and collection of answers.
- Ensures that student does not take the same test twice.
- Fetches the marks assigned to a student and displays them to students.
- Ensures that all students submit the test within stipulated time.

Gateway Agent (GA)

- This agent just acts as a link between student and Student Agent.
- Provides web-based interface for conduct of practical and theoretical examination.

3.2. Agent Interactions

The inter-agent communication is outlined in Figure-2.All the agents will make use of Agent Communication Language (ACL) for communication. The ACL messages used are: Request and Inform.

The user interacts with the Gateway Agent (GA). The GA forwards all requests to Student Agent (SA). The GA interacts with SA with messages for login/logout, score request, fetch and submit test. Student Agent will start test, collect the answers and notify the Examiner Agent (EA) to perform evaluation. The SA and EA share the database.



Figure 2. Model of Agent Communication

3.3. Agents Design

The design of agents with respect to methodologies and techniques used for performing different functions is discussed below.

Examiner Agent

EA performs test paper generation using Randomized Selection Algorithm[21]. In this algorithm, the database of questions for subject and the format for the paper is given as input. The EA constructs the paper using SQL queries. SQL queries fetch all the questions meeting the specified criteria. EA generates as many unique random numbers as number of questions required and chooses the questions from list selected by SQL queries.

EA performs Subjective Evaluation in 2-stages. In the first stage, a hybrid techniqueis used. LSA technique projects the students' answers in n-dimensional plane as vectors and cosine similarity is calculated between these vectors. BLEU is a word average technique. The complementary nature of two algorithms was understood and they were combined using Fuzzy rules resulting into a hybrid technique [25]. This hybrid technique is able to identify the irrelevant answers. The use of hybrid technique helps in reduction of number of keywords to be given, as it finds synonyms of given keywords. This ensures student can make use of words of his choice. In the second stage of evaluation, Domain Ontology is used with classification technique. Ontology is a concept map of a domain that can make the evaluation process holistic. Maximum Entropy (MaxEnt) [31] is the classification technique used to classify sentences in students' answers into Ontology Concepts. This second stage of evaluation is more holistic and near to human evaluation.[45]

EA performs Practical Evaluationusing 4- stage evaluation process[39]. The four stages are: compilation, Functional testing, metric calculation and Similarity checking. Small program was written to parse compiler output and extract the actual number of errors removing redundancy. Testing of programs and metrics are calculated using tools and libraries. Metrics used are Lines of Code (LOC), Lines of comment (LOCOM), number of modules (NOM) and McCabe's Cyclomatic Complexity (MCC). The efficiency is calculated using time and space complexity (TPC and SPC). Similarity checking is done using System Dependence Graphs [37]and RapidSubgraph Calculation algorithm [38].

Objective evaluation is performed by EA for MCQ, Fillin-the-blanks and One-line answer Questions. MCQ evaluation is done by simple matching. Fill-in-the-blanks are evaluated using lexicon Ontology and matching. One-line answer evaluation is done using BLEU technique.

Student Agent:

The SA receives the request from students through GA. The student agent makes use of database queries for fetching the information requested by the students.

4. IMPLEMENTATION

MASMEE is implemented as a prototype application in six incremental phases. Java Programming language is used to implement MASMEE. Several tools, libraries and techniques/technology have been used as shown in Table 1. The six phases in which MASMEE was developed are: Test Paper Generation, MASMEE Agent model, Practical Evaluation Module, Subjective Evaluation Module, Objective Evaluation Module and Web Interface.

Test Paper generation is implemented in Java Programming Language. MASMEE agent model is implemented using JADE. Swings are used for Examiner Agent interface design.In Practical Evaluation module, the compilation is done using Turbo C++, DosBox and JDK. Testing is performed with the help of CUnit,CppUnit (MingW)and JUnit. The Program Dependence Graphs are generated using Frama-C and SDG Library. Metrics are calculated using CCCC tool. All these tools have been invoked from a Java Program that holds the framework together and calculates marks. In Subjective evaluation, preprocessing steps of tokenization and stop word removal are implemented using Java Programming language. The synonym search is performed using WordNet tooland JWI library. Porters stemming is done using source-code written by Originator of the algorithm. Latent Semantic Analysis technique is implemented using Java Programming Language. Word frequency is calculated using Guava library and SVD is performed using MATLAB. Protégé is used to develop Ontology. Jena Libraryis used to reasontheOntology from Java Code.MaxEnt library is downloaded from sourceforge website. Objective module uses techniques of Subjective Evaluation itself. Web Interface is implemented in Java Server Pages, HTML 5, CSS 3 and Apache Tomcat Server.

Table 1. Implementation Tools, Libraries and Techniques used in MASMEE

Phase	Tool	Library	Technology/	Purpose		
			Technique			
MASMEE	Java Development Kit			It is used for development of		
Programming	1.7			MASMEE		
Test Paper	Oracle		Randomized	Implemented through coding using		
Generation	JDK		Selection	Java and Oracle		

MASMEE Agent Model	Oracle	Java Agent DevelopmentEnvironment (JADE)	For agent development and interactions Database is implemented in Oracle.			
		Swings		Interface design		
Subjective Evaluation		Stemmer		Porter's Algorithm implemented by originator of the algorithm		
Objective Evaluation	MatLab 2013	Matlab Control Library	Fuzzy Logic Matrix Operations	Used for LSA matrix calculations and Fuzzy Logic implementation for hybrid technique.		
	WordNet 2.1	JWI 2.2.3	Semantic Networks	Used for finding word Synonyms		
		Guava library	Multi Hash Lists	Used for counting frequency of words		
	Protégé 5.0 beta	Jena library	RDF format of ontology	Subject Specific Ontology Development		
		OpenMaxEnt Library Maximum Entropy		Performs classification of Input on basis of Training data.		
Practical Evaluation	Turbo C++		Compilation of			
	DosBox			To Run Turbo C++		
	MingW			To run CppUnit for testcases of C++		
	JDK1.7		Compilation of Java programs			
	CCCC Compiler			To find metrics of Java, C and C++ programs		
	Frama-C	SDG library	PDG	To generate program dependence graphs		
		CUnit CppUnit JUnit	XUnit Principle	To execute test cases on student programs		
Web Interface		Java Agent DevelopmentEnvironment	Apache Tomcat Server	Web Interface hosting		
		(JADE)	Java Server Pages	Server Side Scripting		
		HTML5	Structuring			
			CSS	Designing		

The database is implemented in Oracle 12c. The Examiner Agent, Student Agent can access the database using SQL queries. The ODBC Drivers are used for connecting to the database. The development platform is Windows 8.

The Domain Ontology designed and developed for subjective module for Computer Graphics subject of Computer Science is shown in Figure-3. The test paper generation input screen is shown in Figure-4. The user interface of MASMEE is shown in Figure-5.The agentsinteraction is shown in Figure-7.For example - The Gateway Agent (named other in the figure) sends login request to Student Agent (stugateway). The student Agent responds with inform message. Similarly other request and inform messages are sent. After all students submit, Student Agent sends inform message to Examiner Agent (e1) to perform evaluation.



Figure 3. Domain Ontology for Subjective Evaluation

📓 Auto Test Generation – 🗖 🗙						
Subject Al v type of test subjective v						
Specific type mcq Difficulty Level of Paper 1						
Header for Paper(Except Subject name and code)						
Generate						



Figure 5. Student Interface





For Verification of the system, simulated and real data was used. Simulated data was obtained by digitizing manually conducted tests of student. Real data was collected by actually conducting the tests of students using MASMEE system. This helped in testing of Multiagent System for management and conduct of examinations; and testing and verification of evaluation methods. The simulated data for subjective evaluation includes 50 questions with 60 answers each, for practical includes 18 questions with 30 answers each and for objective includes 32 questions with 30 answers each. The real data for subjective includes 10 questions with 60 answers each and objective includes 15 questions with 15 answers each.Practical Evaluation module evaluates C, C++ and Java.

The success of the application can be judged by comparing the human rating with computer generated ratings using correlations. Table 2 includes the accuracy of results achieved in MASMEE. The subjective evaluation has a correlation of 72 to 99 percent with human evaluator. Practical evaluation has accuracy of 97percent. Objective evaluation has accuracy of 100 percent for MCQ, 82 percent for Fill-in-the-blanks and 79 percent for one-line answers. Figure 7 includes the results of subjective evaluation using BLEU, LSA and Hybrid technique shown in bar graph. It can be seen that performance of Hybrid technique is better than LSA and BLEU.Table 3 shows the results of practical module using different parameters in stages of evaluationlike errors, functional testing and metrics.

Table 2. Results of Testing of MASMEE

Module	Type of Questions	Accuracy			
Objective	MCQ	100%			
Objective	One-line questions	79%			
Objective	Fill-in-blanks	82%			
Subjective	Long Answers	72-99%			
Practical	C Programs	97%			
	C++ Programs	95%			
	Java Programs	97%			



Figure 7. Results of Subjective Evaluation

Table 3. Results of Practical Evaluation

Criteria		Testing	Locom	Loc	Nom	мсс	ТРС	SPC	Marks
Correlation of Computer generated values	0.91	0.90	1.00	1.00	1.00	1.00	1.00	1.00	0.97
with human Examiner calculated values									

6. CONCLUSION AND SCOPE FOR FURTHER WORK

MASMEE system saves time and effort of examiners and ensures speedy, uniform and efficient evaluation. The randomized selection algorithm in test paper generation module allows for non-repeating questions in test paper. The 2-stage process for subjective evaluation identifies the irrelevant answers in the first stage. In the second stage, evaluation is performed using Ontology. Use of Ontology makes the evaluation more near to human evaluation. The 4stage evaluation framework for practical evaluation enables both black-box and white box testing. Practical module compares the logic of student program with model program which helps in identifying discrepant as well as unique programs.

The Multiagent model can include behaviors for interacting with agents of other universities and colleges and fetching question banks and model answers which helps in paper setting and evaluation. The subjective module can be improved by making use of model answers to extend Ontology automatically. The system can be extended for evaluation of programming languages like LISP, SQL, etc. The practical module can also be extended to handle complex projects.

7. REFERENCES

- C. P. Newhouse, "Computer-based Exams in Schools: Freedom From the Limitations of Paper?," Research and Practice in Technology Enhanced Learning, vol. 8, no. 3, pp. 431–448, 2013.
- [2] M. Wooldridge and N. R. Jennings, "Agent Theories, Architectures and Languages: A Survey," Intelligent Agents, pp. 1–39, 1994.
- [3] M. J. Wooldridge and N. R. Jennings, "Towards a Theory of Cooperative Problem Solving," Distributed Software Agents and Applications, vol. 1069, pp. 15–26, 1994.
- [4] K. Sycara and D. Zeng, "Coordination of Multiple Intelligent Software Agents," International Journal of Cooperative Information Systems, vol. 5, no. 2–3, pp. 1– 31, 1996.
- [5] M. et. al. Jacyno, "Understanding decentralised control of resource allocation in a minimal multiagent system," in 6th international joint conference on Autonomous agents and multiagent systems, 2007.
- [6] A. Nikolopoulou and M. G. Ierapetritou, "Hybrid simulation based optimization approach for supply chain management," Computers & Chemical Engineering, vol. 47, pp. 183–193, 2012.
- [7] and S. G. Lamotte, Olivier, Nicolas Gaud, "the Dynamic Evaluation of a Public Bus Network for
- Small Size Urban Environments," Procedia Computer Science, vol. 56, pp. 168–176, 2015.
- [8] P. Maes, R. H. Guttman, and A. G. Moukas, "Agents That Buy and Sell," Communications of the ACM, vol. 42, no. 3, pp. 81–91, 1999.

[9] et al. Ramchurn, Sarvapali D., "Hac-er: a disaster response system based on human-agent collectives," Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems, 2015.
[10] L Li "Decign of online systemistics system based on web

- [10] J. Li, "Design of online examination system based on web service and COM," 2009 1st International Conference on Information Science and Engineering, ICISE 2009, pp. 3276–3279, 2009.
- [11] L. Xu and W. Huang, "The architecture design of examination system in digital campus based on J2EE technology," 2007 IEEE International Conference on Control and Automation, ICCA, vol. 0, pp. 738–741, 2008.
- [12] S. Vasupongayya, T. Kamolphiwong, S. Kamolphiwong, and S. Sae-Wong, "Interactive examination management system," ICETC 2010 - 2010 2nd International Conference on Education Technology and Computer, vol. 2, pp. 55–59, 2010.
- [13] V. Jamwal and S. Iyer, "Mobile agent based realization of a distance evaluation system," 2003 Symposium on Applications and the Internet, 2003. Proceedings., 2003.
- [14] M. M. Aye, M. Mie, and T. Thwin, "Mobile agent based online examination system," 2008 5th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, pp. 193–196, 2008.
- [15] R. D. Gawali and B. B. Meshram, "Agent-based autonomous examination systems," 2009 International Conference on Intelligent Agent and Multi-Agent Systems, IAMA 2009, 2009.
- [16] S. a. O'Malley, a. L. Self, and S. a. Deloach, "Comparing performance of static versus mobile multiagent systems," Proceedings of the IEEE 2000 National Aerospace and Electronics Conference. NAECON 2000. Engineering Tomorrow (Cat. No.00CH37093), pp. 282–289, 2000.
- [17] M. S. Devi and H. Mittal, "Review of Computerized Evaluation Tools in Education," International Journal of Artificial Intelligence and Computational Research, vol. 4, no. 2, pp. 111–117, 2012.
- [18] G. Cen, Y. Dong, W. Gao, L. Yu, S. See, Q. Wang, and Y. Yang, "A implementation of an automatic examination paper generation system," Mathematical and Computer Modelling, vol. 51, no. 11–12, pp. 1339–1342, 2010.
- [19] D. Liu, J. Wang, and L. Zheng, "Automatic Test Paper Generation Based on Ant Colony Algorithm," Journal of Software, vol. 8, no. 10, pp. 2600–2606, 2013.
- [20] W. Hairui and W. Hua, "Research and Implementation of Multi-agent Based Test Paper Generation Algorithm," 2008 International Conference on Computer Science and Software Engineering, pp. 493–496, 2008.
- [21] H. . . Mittal and M. S. Devi, "Test Paper generation using Selection Algorithm," in International Conference on Communication Systems and Network Technologies, 2016.
- [22] T. K. Landauer, P. W. Foltz, and D. Laham, "An introduction to latent semantic analysis," Discourse Processes, vol. 25, no. 2–3, pp. 259–284, 1998.
- [23] D. Pérez, D. Pérez, A. Gliozzo, A. Gliozzo, C. Strapparava, C. Strapparava, E. Alfonseca, E. Alfonseca, P. Rodriguez, P. Rodriguez, B. Magnini, and B. Magnini, "Automatic Assessment of Students' Free-text Answers Underpinned by the Combination of a BLEU-inspired Algorithm and Latent Semantic Analysis," Proceedings of the Eighteenth

International Florida Artificial Intelligence Research Society Conference, FLAIRS 2005 - Recent Advances in Artifical Intelligence, pp. 358–363, 2005.

- [24] D. Pérez-Marín, I. Pascual-Nieto, and P. Rodríguez, "Computer-assisted assessment of free-text answers," The Knowledge Engineering Review, vol. 24, no. 4, p. 353, 2009.
- [25] H. Mittal and M. S. Devi, "Computerized Evaluation of Subjective Answers Using Hybrid Technique," Proceedings of ICICSE 2015: Innovations in Computer Science and Engineering, vol. 413, pp. 295–303, 2016.
- [26] G. A. Miller, "WordNet: a lexical database for English," Communications of the ACM, vol. 38, no. 11, pp. 39–41, 1995.
- [27] M. A. Finlayson, "Java Libraries for Accessing the Princeton Wordnet: Comparison and Evaluation.," .), Proceedings of the 7th International Global WordNet Conference, pp. 78–85, 2014.
- [28] M. F. Porter, "Snowball: A language for stemming algorithms," 2001.
- [29] B. Bejeck, "Getting Started with Google Guava.," Packt Publishing Ltd., 2013.
- [30] "Matlab Control." [Online]. Available: https://code.google.com/p/matlabcontrol/. [Accessed: 31-Dec-2015].
- [31] J. Z. Sukkarieh, "Using a MaxEnt classifier for the automatic content scoring of free-text responses," AIP Conference Proceedings, vol. 1305, pp. 41–48, 2010.
- [32] "Protege." [Online]. Available: http://protege.stanford.edu/about.php. [Accessed: 28-Feb-2016].
- [33] A. Jena, "Apache jena. jena. apache. org [Online].," 2013. [Online]. Available: http://jena.apache.org. [Accessed: 28-Feb-2016].
- [34] B. Skupas, "Feedback Improvement in Automatic Program Evaluation Systems," Informatics in Education, vol. 9, no. 2, p. 229, 2010.

- [35] N. Walkinshaw, M. Roper, and M. Wood, "The Java system dependence graph," Proceedings Third IEEE International Workshop on Source Code Analysis and Manipulation, 2003.
- [36] P. E. Livadas and S. Croll, "System dependence graphs based on parse trees and their use in software maintenance," Information Sciences, vol. 76, no. 3–4, pp. 197–232, 1994.
- [37] T. Wang, X. Su, Y. Wang, and P. Ma, "Semantic similarity-based grading of student programs," Information and Software Technology, vol. 49, no. 2, pp. 99–107, 2007.
- [38] J. W. Raymond, E. J. Gardiner, and P. Willett, "RASCAL: Calculation of graph similarity using maximum common edge subgraphs," Computer Journal, vol. 45, no. 6, pp. 631–644, 2002.
- [39] H. Mittal and S. D. Mandalika, "Framework for Evaluation of Programming Language Examinations," SpringerBriefs in Applied Sciences and Technology, pp. 121–128, 2015.
- [40] J. Kumar, A., & St Clair, "Cunit-a unit testing framewor for c," 2005. [Online]. Available: http://cunit.sourceforge.net/doc/index.html.
- [41] B. Madden, Using cppunit to implement unit testing. Game Programming Gems, 6, 2006.
- [42] G. Tahchiev, P., Leme, F., Massol, V., & Gregory, JUnit in action. Manning Publications Co., 2010.
- [43] B. Cuoq, P., Kirchner, F., Kosmatov, N., Prevosto, V., Signoles, J., & Yakobowski, "Frama-c. In Software Engineering and Formal Methods," Springer, pp. 233–247, 2012.
- [44] "Library." [Online]. Available: http://www4.comp.polyu.edu.hk/~cscllo/teaching/SDGAPI /. [Accessed: 28-Feb-2016].
- [45] S. D. M and H. Mittal, "Machine Learning Techniques with Ontology for Subjective Answer Evaluation," International Journal on Natural Language Computing, vol. 5, no. 2, pp. 01–11, 2016.