

Automated Exam Question Set Generator Using Utility Based Agent and Learning Agent

Tengku Nurulhuda Tengku Abd Rahim, Ma. Stella Tabora Domingo, Mohamed Farid Noor Batcha, and Zalilah Abd Aziz

Abstract—Exam is an evaluation tool to measure teaching and learning outcomes of educators and their learners respectively. Nowadays, an automated exam question set generator is a must have to reduce educator’s time on preparation of exam question set and increase the quality of exam question set. This paper proposes an Automated Exam Question Set Generator (AEQSG) using Utility Based Agent (UBA) and Learning Agent (LA). Furthermore, AEQSG applies Bloom Taxonomy (BT) scaling to automate Bloom’s Taxonomy (BT) difficulty level distribution and Genetic Algorithm (GA) to optimize the generation of exam question set and generate high quality exam question set that follow educational institution’s guide-lines. The purpose of utility based agent in AEQSG is to give the user an option to choose actions based on a user’s preference (utility) for each generation state. Meanwhile, the purpose of learning agent in AEQSG is to learn from its past exam results (past generation experiences).

Index Terms—Automated exam question generator, bloom’s taxonomy scaling, genetic algorithm, learning agent, utility based agent.

I. INTRODUCTION

Exams are implemented in educational institutions to monitor the progress of educators’ teaching outcome and learners’ learning outcome. The manual preparation process of exam question set requires educators’ great commitments and is time-consuming. In order to reduce educators’ commitments in preparing exam question set, we have proposed an Automated Exam Question Set Generator (AEQSG) using Utility Based Agent (UBA), Learning Agent (LA), Bloom Taxonomy (BT) Scaling and Genetic Algorithm (GA).

UBA chooses actions based on a preference (utility) for each state that maximizes the expected utility. Meanwhile, LA starts to act with basic knowledge (question mark) and then are able to act and adapt automatically (average past question result mark) through learning. In our proposed system, UBA and LA will be applying BT scaling to achieve a desirable result. BT scaling automate the distribution of Bloom’s Taxonomy difficulty level and automate learning experience using Bell Curve Analysis based on the average past exam question result mark to get optimal result for exam question set. GA optimizes the generation of exam question

set based on chosen preference (utility) in AEQSG.

Bloom’s Taxonomy (BT) and GA will not be detailed since this paper is an extended paper from the Automated Exam Question Generator using Genetic Algorithm paper [1].

II. INTELLIGENT AGENT

A. Utility Based Agent (UBA)

Utility based agent is almost similar to the goal based agent that acts based on goals but it also finds the optimum way to achieve the goal. It is beneficial when there are multiple possible alternatives and an agent has to choose before performing the best action. The utility function maps each state to a real number to evaluate the efficiency of each action in achieving the goals [2].

UBA as in Fig. 1 uses a world model together with a utility function that measures its preferences among states and then chooses the action that leads to the best expected utility. By averaging over all possible outcome states, weighted by the probability of the outcome; expected utility is computed [3].

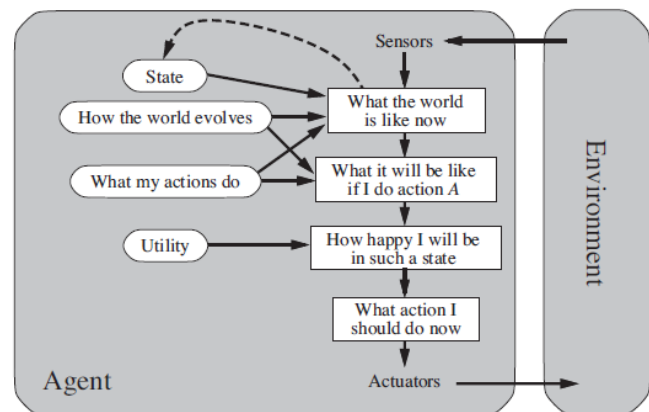


Fig. 1. A model-based, utility-based agent [3].

B. Learning Agent (LA)

A general intelligent agent known as a learning agent is the preferred method for creating state-of-the-art systems in Artificial Intelligence where any type of agents can be generalized into a learning agent to generate better actions [4]. A learning agent has the ability to learn from its past experiences even though it starts with basic knowledge and then adapts automatically through learning [2].

There are four main conceptual components in a learning agent as shown in Fig. 2. The learning element is responsible on improvements based on learning from the environment. The learning element takes feedback from the critic.

Manuscript received September 25, 2019; revised December 22, 2019
Tengku Nurulhuda Tengku Abd Rahim, Ma. Stella Tabora Domingo, Mohamed Farid Noor Batcha are with the MIMOS Berhad, Technology Park Malaysia, Kuala Lumpur 57000 Malaysia (e-mail: huda.rahim@mimos.my, stella.domingo@mimos.my, farid.batcha@mimos.my).

Zalilah Abd Aziz is with the Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Boulder, 40450 Shah Alam, Selangor, Malaysia (e-mail: zalilah@tmsk.uitm.edu.my).

Meanwhile the critic describes how well the agent is doing with respect to a fixed performance standard. The performance element is responsible for external action selection. The problem generator is responsible for actions suggestion that will lead to new and informative experiences. Therefore, a learning agent is able to learn, analyze performance and look for new ways to optimize its performance [2].

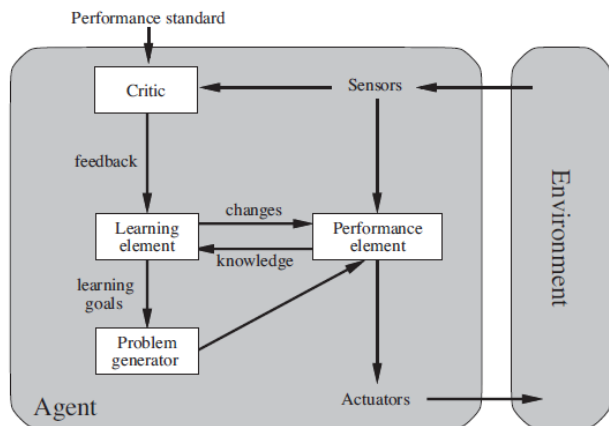


Fig. 2. A general learning agent [3].

III. DEEP REINFORCEMENT LEARNING

Deep Reinforcement Learning (DRL) is the combination of Reinforcement Learning (RL) and Deep Learning (DL); whereas RL with function approximation by deep neural networks [5]. RL and DL are both systems that learn autonomously. The difference between RL and DL is RL learns dynamically by adjusting actions using continuous feedback in order to optimize the reward, while DL learns from a training set and then applies that learning to a new data set [6].

Meanwhile, the difference between RL and DRL is RL is dynamically learning with a trial and error method to maximize the outcome, while DRL is learning from existing knowledge and applying it to a new data set [7]. Unlike supervised and unsupervised learning algorithms, RL is about training an agent to interact with an environment and maximize its reward [8].

RL model as shown in Fig. 3 consist of three basic concepts: state, action, and reward. The state describes the current situation of the agent. The action describes the action taken by agent at each state. The reward describes the feedback from the environment [8].

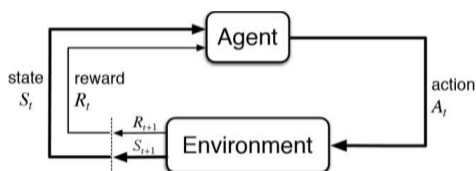


Fig. 3. Reinforcement learning model [8].

IV. RELATED WORK

Automated Test Paper Generation [9] is a web-based application system prototype implementing utility-based agent to cater exam difficulty level, and shuffling algorithm to design question set randomly without repetition or

duplication. On the basis of utility value, two sets of question items are selected. The first question items set consist of the highest utility value, whereas the second question items set consist of the second highest utility value of the test paper. Accumulative difficulty level of the test paper is calculated based on the question items' utility value.

An intelligent agent is introduced to **Automatic Question Paper Setting (AQPS)** [10] to automate the generation of question paper using random number generation technique and backtracking algorithm known as Automatic Question Paper Generator Agent (AQPA). Basic components of AQPS are Hash Table (HT) database, Repeated Question Checker Module (RQCM) and Question Paper Generation Module (QPGM). The HT database is used to store questions with their keys with optimum space; in order to find required question very quickly. RQCM ensures that a new retrieved question does not match with any question in selected question database. QPGM generates the question paper based on selected pre-defined question format and selected question database.

The proposed framework for **Automatic Question Generation (AQG)** [11] is multi-agent which consists of Document Processing Agent (DPA), Information Classification Agent (ICA) and Question Generation Agent (QGA). DPA will first start processing input text using Tree Tagger tool to produce a one word per line form together with its tag and lemma; the processed output of Tree Tagger is ranked based on the frequency of each word occurrence; then prefix and suffix of each word is removed with a stemming process to get the root word. Bloom's category is found in ICA based on selected keywords after the stemming process and word count from DPA by matching keyword with action verb in the repository. The output from ICA will be used in QGA to generate questions based on template that suit the selected keywords according to the Bloom's levels.

The two main components in **Utility Based Test Paper (UBTP) agent** [12] are Knowledgebase Developer (KD) and Test Paper Generator (TPG). KD is a component that stores question items with its utility value with integer value in range between 0 and 5 in file form. TPG is a component that generate a test paper from selected question items (filtered by its utility value) from knowledgebase based on overall difficulty level (from 1 percent to 100 percent) entered by examiner. Three sets of question items will be selected on the basis of utility value in TPG algorithm. In other words, first set has the highest utility value calculated, second set has one less than highest utility value calculated and third set has 0 utility values.

Most of papers above implement Utility Based Agent (UBA) only in their system but none of them are using Learning Agent (LA) or Deep Reinforcement Learning (DRL). In order to improve the quality of Exam Question Set Generator, we propose multi agent with combination of two agents; UBA and LA. We also suggest to apply DRL for LA.

V. PROPOSED SYSTEM

The overall architecture of the proposed system i.e. Automated Exam Question Set Generator (AEQSG) using Utility Based Agent (UBA) and Learning Agent (LA) is shown in Fig. 4.

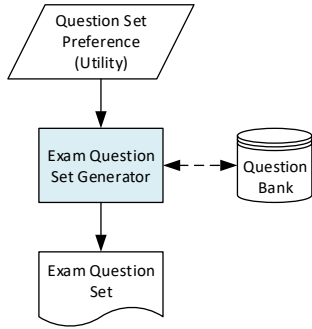


Fig. 4. Automated exam question set generator overall architecture.

Question collection workflow process as shown in Fig. 5 will be done before hand to provide the proposed system with a collection of questions either new or previous question from past exams. User can either key in or uploaded file for new and previous questions into the system. Then the system will automatically validate each uploaded file and validate each questions keyed in or uploaded. After that, all validated questions will be stored in the question bank.

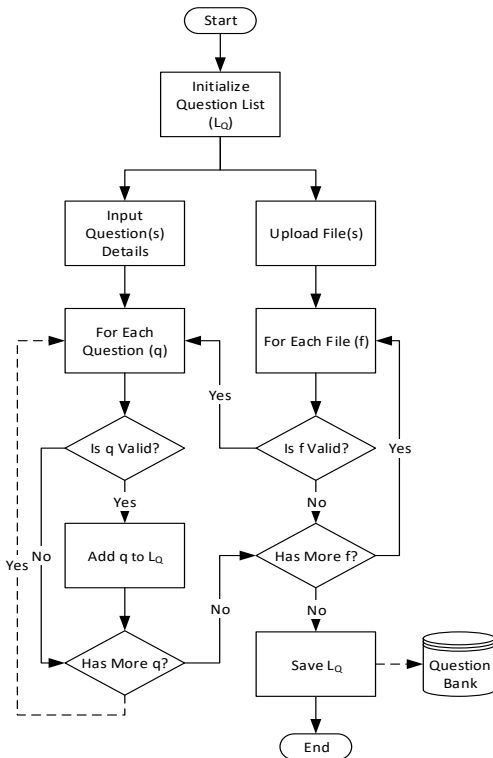


Fig. 5. Question collection workflow process.

All uncategorised questions either new question or previous question from past exam that does not have any BT category or chapter category assigned when the questions are keyed in or uploaded using file; the proposed system will automatically categorise based on BT categories (as shown in Table I) and chapter during question categorisation workflow process (as shown in Fig. 6).

Therefore the person in charge to key in or upload questions to proposed system can be anyone other than Subject Matter Expert (SME). Thus, in the same time will helps to reduce educators' commitments and reduce significant amount of time in preparing exam question set. In addition to above benefits, the question categories would be more appropriated to course syllabus guideline since it not only categorised BT category but also chapter category for

the specific question saved into question bank.

TABLE I: BLOOM'S TAXONOMY CLASSIFICATION [1]

ID	Category	Level
C1	Knowledge	Easy
C2	Comprehension	
C3	Application	Medium
C4	Analysis	
C5	Synthesis	Hard
C6	Evaluation	

Question categorisation workflow process will be started with pre-processing of uncategorised questions using Natural Language Processing pre-processing (NLP pre-processing) included lower case standardization, remove non alphabet character, tokenization, Part of Speech (POS) tagging, noun extraction, verb extraction and stemming verbs before Bloom's Taxonomy classification (BT classification) and chapter classification parallel workflow process begin.

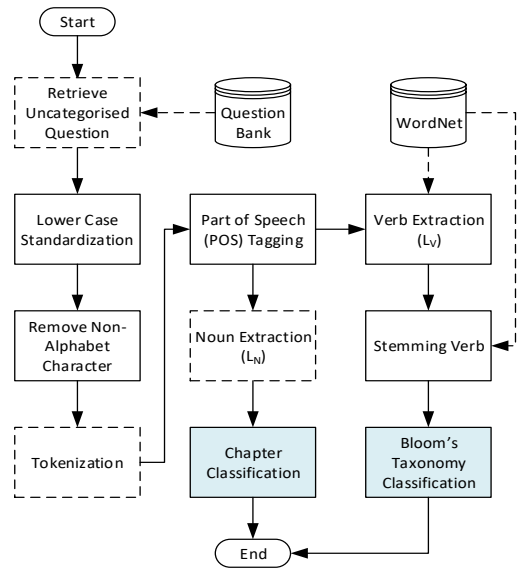


Fig. 6. Question categorisation workflow process.

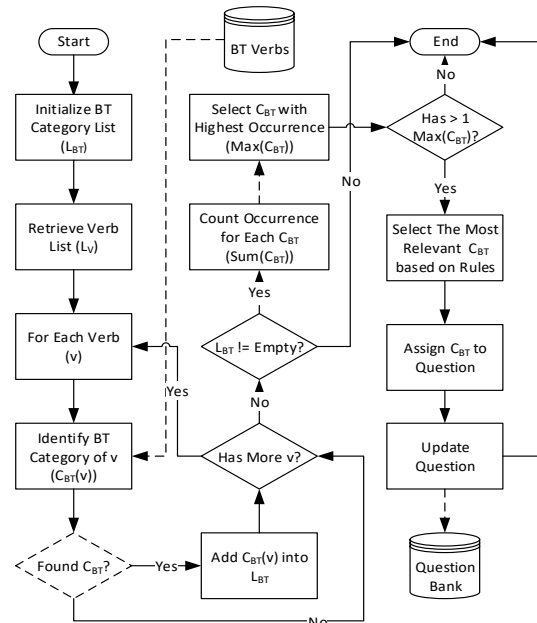


Fig. 7. Bloom's taxonomy classification workflow process.

In BT classification workflow process (as shown in Fig. 7) each verb in verb list extracted after NLP pre-processing

done; will be identified with respective BT categories based on Bloom's Taxonomy (BT) verbs. Then respective BT categories with highest occurrence will be selected after the summation of each BT categories done. If more than one highest occurrence found, the most relevant BT category will be selected based on rules. The selected BT category will be assigned to the question and the question in question bank will be updated with selected BT category.

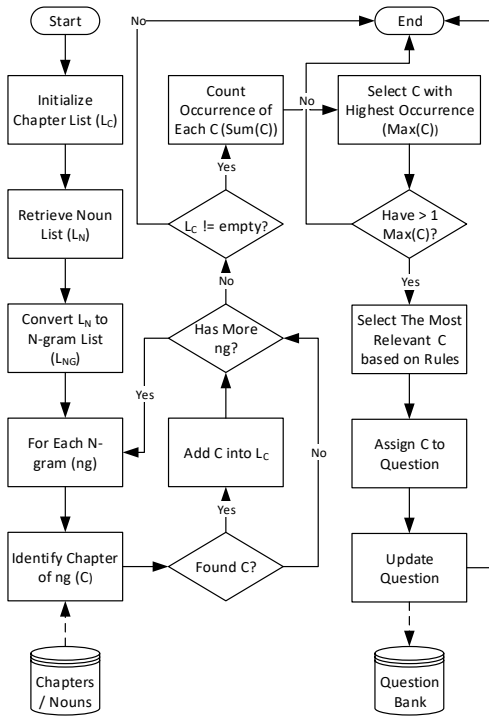


Fig. 8. Chapter classification workflow process.

Meanwhile in chapter classification workflow process (as shown in Fig. 8), the noun list extracted from previous NLP pre-processing will be converted into n-gram list. After that each n-gram in n-gram list will be identified with respective chapter categories based on Chapters / Nouns reference table. Then, each chapter categories summation done, the respective chapter categories with highest occurrence will be selected. The most relevant chapter category will be selected based on rules, if more than one highest occurrence found. Later, the question in question bank will be assigned and updated with selected chapter category.

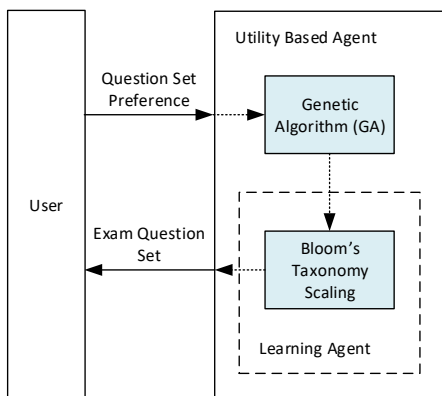


Fig. 9. Proposed model for automated exam question set generator.

Our proposed system, AEQSG will be implemented using both Utility Based Agent (UBA) and Learning Agent (LA). It

also implement Bloom's Taxonomy (BT) Scaling and Genetic Algorithm (GA) to optimize the process and enhance the quality of the new exam question set. The proposed model for AEQSG using UBA and LA is shown in Fig. 9.

The UBA and LA are chosen as a model for the proposed system to achieve our AEQSG goals while optimizing exam question set generation using GA. Moreover, LA is chosen to increase the quality of exam question set using Bell Curve Analysis based on the average past exam question result mark in BT Scaling. Furthermore, BT scaling also provides the mechanism to automate the distribution of Bloom's Taxonomy difficulty.

AEQSG will generated the new exam question set from question bank based on user's question preference. The user is given the option to select excluded year range, academic year, semester, education level, exam duration, exam difficulty, subject, chapter selection, total question and question margin for question set preference (utility function) before the new exam question set is generated. User's question preference workflow process as shown in Fig. 10 will be executed before GA workflow process begin.

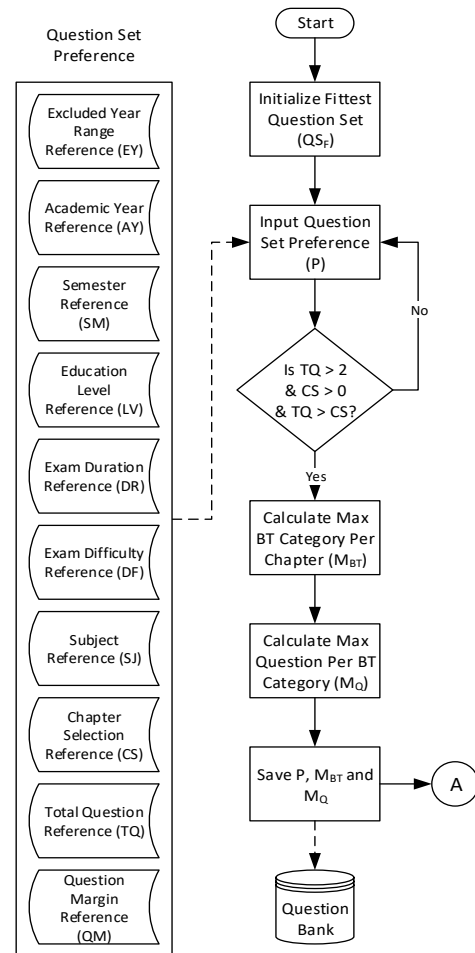


Fig. 10. User's question preference workflow process.

GA workflow process as shown in Fig 11, start with initialize question sets based on user's preferences question set. After question set been initialized, the fitness value for each question sets will be evaluated and question set is selected using Roulette Wheel selection. Then, single crossover and mutation will be performed based on selected question set. After single crossover and mutation done, fitness value for new question sets will be evaluated. Later,

the best fitness will be selected from the current question sets and pass to Bloom Taxonomy (BT) Scaling.

UBA chooses actions based on a preference (utility) for each state that maximizes the expected utility. UBA using GA to optimizes the generation of exam question set based on chosen preference (utility) in AEQSG and generate high quality exam question set that follow educational institution's guide-lines.

The fitness formula as shown in Fig 12, is used to calculate the fitness value in GA (that resides in UBA) to get the best fitness before the fitness value is pass to LA to proceed with BT scaling process. W is represent exam question set quality weightage percentage as shown in Table II. The lowest value of the fitness value is the best fitness.

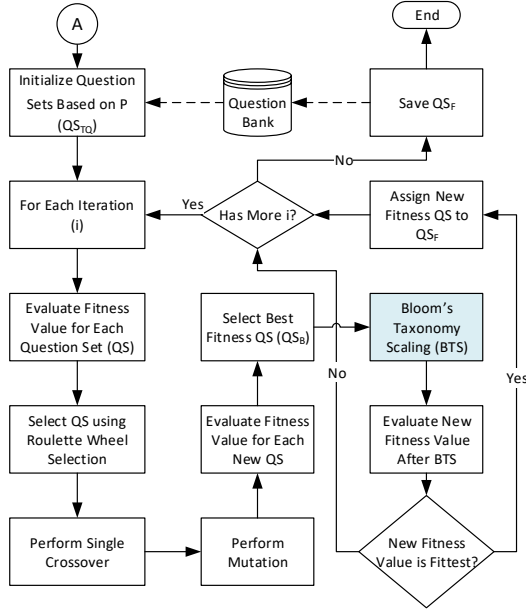


Fig. 11. Genetic algorithm workflow process.

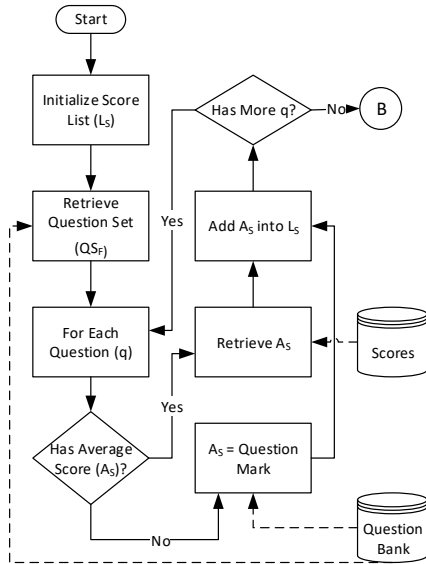


Fig. 12. BT scaling – average score workflow process.

LA starts to act with basic knowledge (question mark) where there is no result mark average (as shown in Fig. 13); and then it is able to act and adapt automatically (average past question result mark) through learning. BT scaling automate the distribution of Bloom's Taxonomy difficulty level (as shown in Fig. 14) and automate learning experience using Bell Curve Analysis based on the average past exam question

result mark (as shown in Fig. 15) to get optimal result for exam question set.

No. of Level	No. of Category	Example	Weightage	Quality
3	6	C1, C2, C3, C4, C5, C6	100	Good
	5	C1, C2, C3, C4, C6	90	
	4	C1, C2, C4, C5	80	
	3	C2, C4, C5	70	
2	4	C3, C4, C5, C6	60	Average
	3	C1, C3, C4	50	
	2	C4, C5	40	
1	2	C3, C4	30	Bad
	1	C5	20	

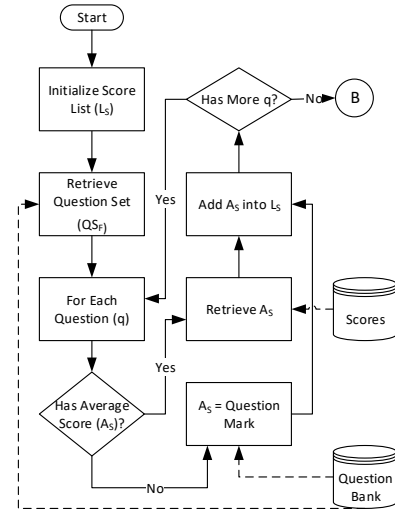


Fig. 13. BT scaling – average score workflow process.

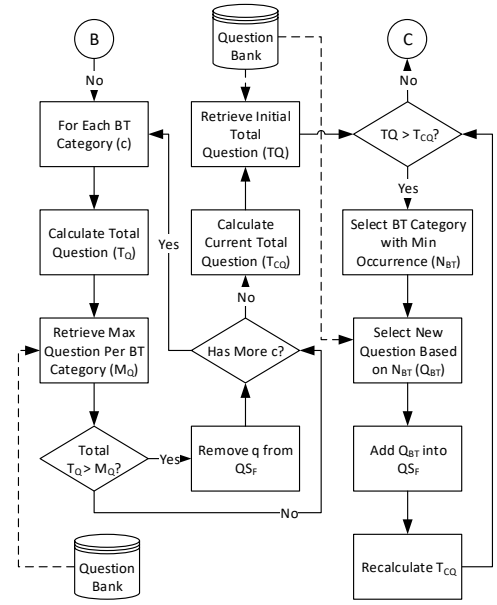


Fig. 14. BT scaling – average score workflow process.

By adopting Deep Reinforcement Learning (DRL) in LA, we will be beneficial from both Reinforcement Learning (RL) and Deep Learning (DL) advantages. RL training an agent to interact with an environment and maximize its reward while DL complement RL by providing RL with function approximation by deep neural networks. RL is dynamically learning with a trial and error method to maximize the outcome but DRL learns dynamically by adjusting actions

using continuous feedback in order to optimize the reward and applying it to a new data set.

Since we are in the midst of exploring DRL for this proposed system, we will not discuss DRL in further details. What kind of algorithm or model or deep neural network that will be used in future for the proposed system will not be described in this paper.

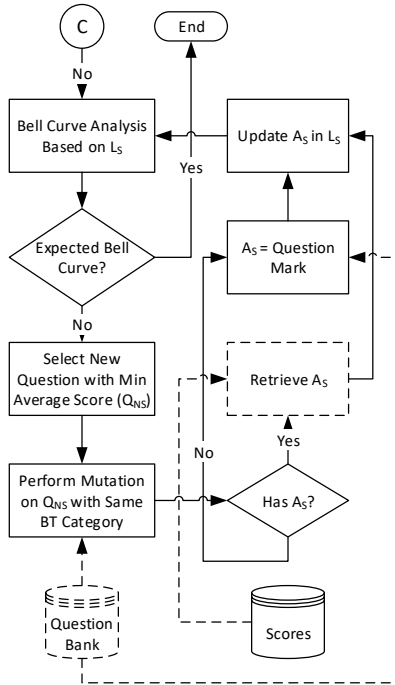


Fig. 15. BT scaling – average score workflow process.

VI. CONCLUSION AND FUTURE WORK

This paper presents a novel and feasible approach for Automated Exam Question Generator using Utility Based Agent (UBA) and Learning Agent (LA) by implementing Genetic Algorithm (GA) and Bloom's Taxonomy (BT) scaling which includes automated Bloom's Taxonomy (BT) difficulty distribution in exam question set and Bell Curve Analysis.

In future, a few researches can be done based on LA to explore more on Deep Reinforcement Learning (DRL) with the combination of other type of algorithms or model or deep neural networks that will be used to improve current proposed system.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

All works in this paper have been done amongst authors.

REFERENCES

- [1] T. N. Tengku Abd Rahim, Z. Abd Aziz, R. H. Ab Rauf and N. Shamsudin, "Automated Exam Question Generator using Genetic Algorithm," in *Proc. 2017 IEEE Conference on e-Learning, e-Management and e-Services (IC3e)*, Shah Alam, 2017.
- [2] Types of AI Agents. Java T Point. [Online]. Available: <https://www.javatpoint.com/types-of-ai-agents>
- [3] S. Russell and P. Norvig, *Artificial Intelligence, A Modern Approach*, Third Edition, New Jersey: Pearson Education, 2010.
- [4] P. Gupta, *Rational Agents for Artificial Intelligence*, Hackernoon, September 2017.

- [5] Funstematics.ai. 2019. [Online]. Available: <https://www.funstematics.ai/ai-case-studies>
- [6] M. Rouse, "deep learning," SearchEnterpriseAI, TechTarget, October 2019.
- [7] T. Williams. (September 2019). Reinforcement learning Vs. deep reinforcement learning: What's the difference? Techopedia Inc. [Online]. Available: <https://www.techopedia.com/reinforcement-learning-vs-deep-reinforcement-learning-whats-the-difference/2/34039>
- [8] Jay, "Machines that learn with reinforcement learning," *Adverai*, July 2018.
- [9] S. A El_Rahman and A. H. Zolait, "Automated test paper generation using utility based agent and shuffling algorithm," *International Journal of Web-Based Learning and Teaching Technologies*, pp. 69-83, 2019.
- [10] K. H. Pinjani, R. Y. Raut, and P. S. Yedey, "Developing an intelligent agent for Automatic Question Paper setting," in *Proc. National Conference on Advanced Technologies in Computing and Networking-ATCON-2015*, Amravati, 2015.
- [11] S. Pandey and K. C. Rajeswari, "Automatic question generation using software agents for technical institutions," *International Journal of Advanced Computer Research*, vol. 3, no. 13, pp. 307-311, 2013.
- [12] M. J. Arshad, M. Naz, Y. Saleem, A. Farooq and K. H. Asif, "Modeling an agent for paper generation system using utility based approach," *Journal of Faculty of Engineering & Technology*, pp. 109-124, 2012.

Copyright © 2020 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited ([CC BY 4.0](https://creativecommons.org/licenses/by/4.0/)).



Tengku Nurulhuda Tengku Abd Rahim received the B.I.T. degree in computer system technology from University of Malaysia Sarawak (UNIMAS), Kota Samarahan, Sarawak, Malaysia in 2000 and the M.C.S. degree in computer science from MARA University of Technology (UiTM), Shah Alam, Selangor, Malaysia in 2016. She is currently a senior engineer at Artificial Intelligence Department, MIMOS Berhad, Kuala Lumpur, Malaysia.



Ma. Stella Tabora Domingo graduated with BSCoE in the field of computer engineering from University of Baguio, Philippines in 2005

She works currently as a staff engineer at the Artificial Intelligence Department, MIMOS Berhad, Kuala Lumpur, Malaysia.



Mohamed Farid Noor Batcha graduated with BSEE in the field of communications from University of Kentucky, United States in 2001 and obtained his masters of engineering (electrical) from University Technology Malaysia in 2010.

He is currently a staff engineer at Artificial Intelligence Department, MIMOS Berhad, Kuala Lumpur, Malaysia.



Zalilah Abd Aziz received the BSc (Hons) degree in computer science from ITM/UKM, Shah Alam, Selangor, Malaysia in 1996, the MSc in computer science (software engineering) from University Putra Malaysia (UPM) in 2003, Serdang, Selangor, Malaysia and PhD in Computer science (artificial intelligence) from University of Nottingham, Nottingham, United Kingdom in 2013.

She is currently a senior lecturer at the Faculty of Computer and Mathematical Sciences, MARA University of Technology (UiTM), Shah Alam, Selangor, Malaysia since 1997. Her primary research include software quality, software engineering, artificial intelligence, combinatorial optimization problems, metaheuristics and programming.