International Journal of Innovative Computing, Information and Control
Volume **15**, Number **1**, February **2019**
ICIC International ©2019 ISSN 1349-4198
pp. 143–161

A RULE-BASED EXPERT SYSTEM FOR AUTOMATIC QUESTION CLASSIFICATION IN MATHEMATICS ADAPTIVE ASSESSMENT ON INDONESIAN ELEMENTARY SCHOOL ENVIRONMENT

Umi Laili Yuhana[1,2], Siti Rochimah[2], Eko Mulyanto Yuniarno[1,3]
Aliia Rysbekova[4], Alex Tormasi[5], Laszlo T. Koczy[4,5]
and Mauridhi Hery Purnomo[1,3]

[1]Department of Electrical Engineering
[2]Department of Informatics
[3]Department of Computer Engineering
Institut Teknologi Sepuluh Nopember
Kampus ITS, Sukolilo, Surabaya, Jawa Timur 60111, Indonesia
{ yuhana; siti }@if.its.ac.id; { ekomulyanto; hery }@ee.its.ac.id

[4]Department of Telecommunications and Media Informatics
Budapest University of Technology and Economics
Magyar tudósok krt. 2, 1117, Budapest, Hungary
rysbekova@edu.bme.hu; koczy@tmit.bme.hu

[5]Department of Informatics
Széchenyi István University
Egyetem tér 1, 9026 Gyõr, Hungary
{ tormasi; koczy }@sze.hu

Received April 2018; revised September 2018

Abstract. *This paper is part of research in developing a competency-based assessment system for mathematics in Indonesian elementary school environment. An essential task is to accurately classify questions based on competency and difficulty level. Thus, an expert system is needed to classify those questions since competency information is often manually defined by experts. The objectives of this work are replacing a human expert's role in the related knowledge engineering process and providing a rule-based expert system to supersede an expert to classify the questions. Five types of the rule-based algorithm: OneR, RIPPER, PART, FURIA, and J48, were applied to the dataset, which is comprised of 9454 real mathematics examination questions collected from several Indonesian elementary schools. Following the knowledge engineering principles, these algorithms generated the classification rules based on a pattern of the data. The rules of the best performing algorithm were utilized by a knowledge base for inference. Finally, to be able to fully measure the system performance, ten expert teachers were involved in the question classification step. The results confirm that the system meets the stated objectives in classifying the competency and the difficulty level of a question automatically.*
Keywords: Adaptive assessment, Automatic question classification, Rule-based expert system

1. **Introduction.** From recent times most education systems started placing competence achieving at the core of the curriculum because acquiring a defined competency, one of the education system's goals, per se is limited to one's capabilities only. As an example, in Indonesia, there are 147,503 elementary schools with 4,172,791 students as for 2016/2017 academic year. All of these schools practice competency-based curricula established by the National Education Standards Agency and regulated by the Ministry of Education

DOI: 10.24507/ijicic.15.01.143

and Culture of Indonesia. The list of all competencies can be found in the appendix of the regulation document issued by the Ministry of Education and Culture. Consequently, the school and its stakeholders must provide a competency-based assessment system [1]. For assessing the competency, each school must provide questions and classify the questions based on the competency.

The goal of the competency-based assessment system is to evaluate how well a student's competencies were achieved. Competency expresses the formulation of knowledge, attitudes and skill capabilities that must be possessed by a student [2]. Slightly different from other countries, in Indonesia one's skills are assessed on the following four levels: learning outcome, core competency, basic competency, and specific competency. One learning outcome consists of several core competencies, which are sets of basic competencies. And each basic competency is, in turn, comprised of specific competencies. Another characteristic of Indonesian assessment system is evaluating one's competencies using competency-based questions. One specific competency, the lowest level of competency, can be evaluated using several questions. Therefore, it is essential to correctly classify the questions according to their relationship with some specific competency. And then the student's answers can be analyzed to decide whether a specific competency was achieved by one. Subsequently, the student's higher levels of competency, i.e., basic competency, core competency and learning outcome, can be all inferred from the results related to the specific competency. So apparently, assessing competency-based tests requires a different approach compared with conventional grading of knowledge-based paper-and-pencil tests [3]. The adaptive assessment system is a designed solution for this problem. The core idea is that during a test a student is asked only the questions matching the assessed competency and one's knowledge level.

An adaptive assessment system is an assessment tool, which in addition to the standard set of features, is capable of automatic self-adjustment according to the knowledge level of a student. The adaptive assessment system is described as an interactive approach to assessing a learner in the learning system that can distinguish the ability of the learner and choose the item test according to it [4]. Particularly, the mentioned system develops an adaptive question displaying scheme, in which the exam questions corresponding to the assessed competency are matched with the knowledge level of a student. To determine one's competency several questions of various difficulty levels need to be asked. For each competency, there are thousands of questions that can be selected to assess the student's competency. The size of such a big database is too large to be tackled by human experts, not to speak about the subjectivity of such expert opinions, where different experts may evaluate competencies in different ways. This is obviously a problem of data analysis that requires an automated tool that copes with the amount of data and produces uniform results. Therefore, it is essential to correctly classify the questions according to the competency and the difficulty level automatically.

An expert system is needed to classify the questions based on competency and difficulty level since information about it is often manually defined by experts. Thus, our paper proposes a particular solution to classify the competency-based questions according to the difficulty level of a question in an automatic way. A rule-based expert system is designed to solve the previously mentioned classification task. Commonly, rules are manually assembled by knowledge engineer based on experts' explanation in the knowledge engineering process [5]. Thus, the motivation of this research is seeking the possibility to extract the rules automatically for replacing the human expert's role in knowledge engineering process and develop an expert system to classify the competency and the difficulty level of the questions using the extracted rule.

In knowledge engineering process five types of the rule-based algorithm: OneR, RIPPER, PART, Fuzzy Unordered Rule Induction Algorithm (FURIA) and J48, were applied to a dataset and evaluated to generate rules for a knowledge base. The proposed solution classifies arithmetical questions with a certain pattern and supersedes the expert judgment. The addition and subtraction questions from mathematics were used as the base examples to show how the proposed method works.

2. **Problem Statement and Preliminaries.** The research on competency-based assessment and question classification focuses on many aspects. One of the examples is that the assessment system can integrate with a learning system [6-11] or can be a separate system [12,13]. Therefore, the following section further discusses the previous works on competency-based assessment and rule-based classification method.

2.1. **Competency-based assessment system.** Recently, some researchers have proposed new approaches for competency-based assessment. Gulikers et al. have developed innovative competency-based performance assessment to be used in pre-VET, prevocational secondary education, in the Netherlands. The novelty of the method consists in checking one competency by several assessments. For competency-based assessment to work, they suggest to carefully interpret the competencies and methods for their assessments. Information and Communication Technologies (ICT) have been used to support this type of assessment and competency-based learning. Klinkenberg et al. proposed a model for children to practice adaptive test on arithmetic and monitor their progress in a web application [14]. The individual approach of the app, explained by its test adaptively, helped to boost problem-solving competency and math performance of the children [15]. Another scientists, Ilahi-Amri et al. have developed a framework for the assessment of formal and informal learning competencies [16]. The authors focus on modeling the architecture of a web-based competence assessment. It is concluded by revealing a relationship between the difficulty of questions being assessed and a competence but lacks the information on how to classify the questions based on competence.

Luckily with the increase of computing power and memory in the computers nowadays, question classification has become a trending topic in the field of natural language processing and machine learning [17]. Several works have been discussing about replacing expert judgment in the classification process. Their proposed replacement methods were differing by question types used in the test.

Aysel et al. categorized mathematical questions using levels of cognitive demand framework developed by the QUASAR Project, when comparing Turkish and Irish mathematics examinations in their work [18]. Since the classification in this work is a mere preliminary process that must be done to run further investigation on the effects of tests on the teaching and learning of mathematics at post-primary level in Ireland and Turkey, this approach cannot be applied to competency-based assessment in Indonesia. Other researchers, Omar and Haris, used a rule-based method to classify and analyze the questions based on bloom taxonomy classes [19]. They developed the rules for six categories: knowledge, comprehension, application, analysis, synthesis, evaluation, using a syntactic structure of each question. The study analyzed the questions of the computer programming written examination through natural language processing. This approach helped the lecturers to assess a student's cognitive levels from written examination questions even though the authors constructed rules manually. Similar to Omar and Haris, Kusuma et al. proposed an automatic question classification system by utilizing bloom taxonomy classification [20]. The authors used natural language processing to classify questions in

Indonesian language and define knowledge level, i.e., competency of the student. The system extracted lexical and syntactic features from mathematics, science, social, and civic subjects' questions. The Support Vector Machines (SVM) method successfully classified the questions, but it still did not infer any information about specific competency related to the question.

Some researchers proposed methods to classify question using machine learning [21-23]. Several others used rule-based method for question classification because of highly accurate predictions on guessing the category of a certain question [19,24]. While Verdú et al. proposed a genetic fuzzy expert system for classifying the questions to introduce a competitive learning environment [25]. Verdú et al. argued that some approaches proposing functioning classification were hard to apply to other cases.

2.2. **Rule-based classification method.** Rule-based classification method represents a classifier utilizing a set of IF-THEN rules [26]. The "IF" part is recognized as the rule antecedent or precondition. It consists of one or more attribute tests or conditions with AND relationships between them. The "THEN" part is the rule consequent that consists of class prediction. Rules can be mined from a decision tree or from a training data using Sequential Covering Algorithm (SCA). SCA works in three steps: (1) learn a rule, (2) remove the data it describes, (3) repeat steps (1) and (2) until no data has remained.

Five rule-based methods – OneR, PART, Fuzzy Unordered Rule Induction Algorithm (FURIA), RIPPER, and J48, were analyzed to find the best performing among them to be used later in the rule-based classifier. These five algorithms are chosen because these algorithms are commonly used for prediction and classification in educational data mining [17,27,28].

OneR is a simple one-level decision tree algorithm, thus performing fast. It chooses one by one parameter from a dataset and produces a new set of rules from a training set. After selecting a parameter that suggests rules with the smallest error rate, OneR builds the final decision tree [29]. RIPPER, expanded as Repeated Incremental Pruning to Produce Error Reduction, is mainly more efficient on large noisy datasets and produces small error rate [30]. Several researchers used RIPPER for question categorization. RIPPER handles multiple classes and considers each organized as a distinct two-class problem [31]. Radev et al. compared RIPPER and heuristic approach to identify semantic types of question to support natural language question answering on the web [32]. Even though the heuristic approach is better in this case based on the experiment, RIPPER produced a small error for a large dataset. PART is a rule-based classifier that repeatedly utilizes partial decision tree algorithm to infer the rules. It combines two principal concepts for rule creation: constructing rules from decision trees and applying the divide-and-conquer rule-learning technique. Frank and Witten found that PART was simpler than C.4.5 and more accurate than RIPPER [33]. Fuzzy Unordered Rule Induction Algorithm (FURIA) is made up of simple and comprehensible rule sets learning fuzzy rules instead of conventional rules and unordered rulesets instead of rule lists. It makes use of an efficient rule stretching method to deal with uncovered examples [34]. Lastly, J48 solves classification problem by constructing a tree to model the classification process. For predicting and classification student data, OneR, PART, and J48 algorithm are commonly used by the researchers [35,36]. In predicting drop out from social behavior of the student, PART and J48 have better performance than OneR.

3. **Proposed System.** This section explains our proposed system.

3.1. **Adaptive competency-based assessment system.** Taking account of the previously mentioned works and applying their approaches, whether possible, to our dataset,

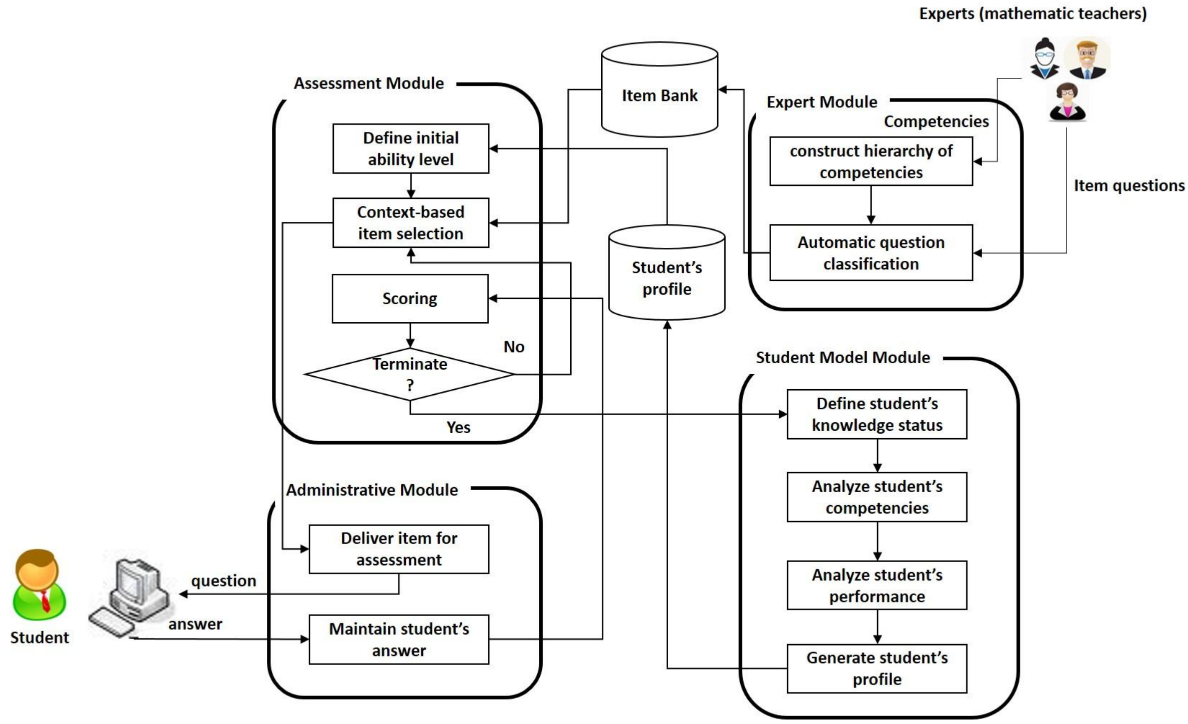

FIGURE 1. Components of an adaptive competency-based assessment system

the adaptive competency-based assessment system has been designed to assess student competencies adaptively. The system is comprised of four modules as can be seen from Figure 1: expert, assessment, administrative and student model modules. The expert module provides the interface for experts to input competencies and questions to the system. So this module can automatically determine the competency and the difficulty level of a question. All the competencies, questions, relations between these two, difficulty levels of the questions are then stored in the custom database named Item Bank. The assessment module is made of three sequential processes: defining the initial ability of a student [36], based on this – selecting the question to be asked next, calculating the score. It also defines a condition for the assessment to terminate.

The administrative module regulates all activities of the assessment system and provides the student exam interface. And lastly, the student model module keeps track of the student's knowledge level, student's competencies and performance, analyzes this information and generates the student's profile.

The assessed 4 levels of competency are shown in Figure 2. Figure 2 shows the relationship between competencies, topics, and questions. In the competency-based curriculum, there are four competencies that must be assessed, i.e., graduate competencies standard, core competencies, basic competencies, and specific competencies. The output of such assessment is a set of standard competencies of a student, i.e., learning outcomes showing the student's capabilities. Graduate competency standard denotes the ability that students should have when they graduate. Each graduate competency standard has the relationships with some Core Competency (CC). CC represents the level of ability to achieve the graduate competency standards that a learner should have at each grade level. Indonesian government groups the CC into four categories, those being: spiritual attitude, social attitude, knowledge, and skill core. A student possessing competency of spiritual attitude admits and obeys the religious guidance, while achievement of social attitude competency implies the student is honest, disciplined, responsible, careful

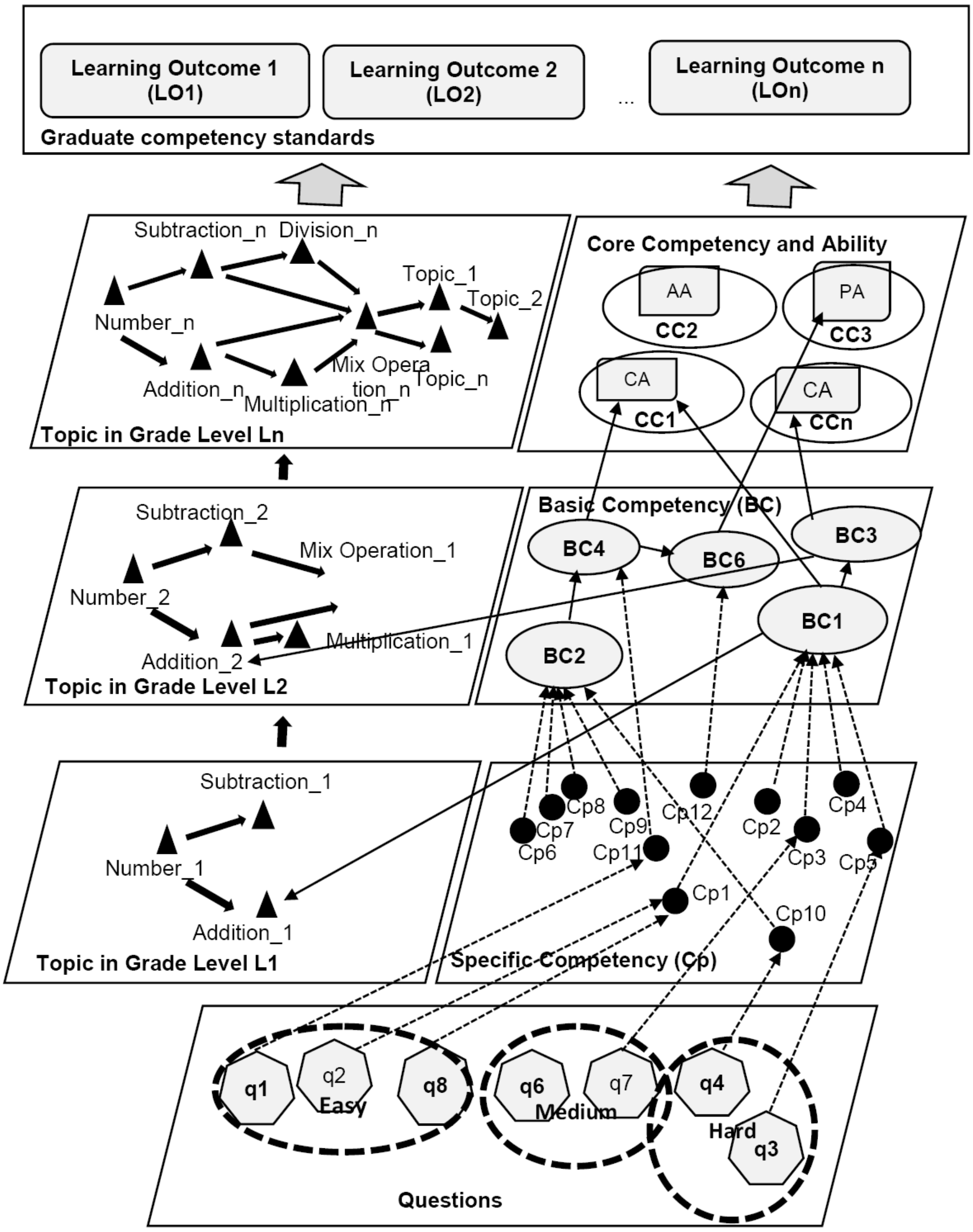

Figure 2. Concept map of hierarchy of competencies

and confident when interacting with family, friends or teachers. Each competency in CC can be categorized as affective ability, cognitive ability, or psychomotor ability in bloom taxonomy concept. The spiritual and social attitude groups together make up Affective Ability (AA), while the knowledge and skill core groups represent respectively Cognitive Ability (CA) and Psychomotor Abilities (PA) defined in the bloom taxonomy [37]. Each CC can be categorized into AA, CA, or PA, as shown in Figure 2. The meaning of basic competence is the ability and the minimal learning materials that must be achieved by learners for a subject in each educational unit that refers to the core competencies. Each core competency has the relationship with several basic competencies. Those are the goals to be achieved in the related grade levels. Some topics and basic competencies in mathematics are interconnected between levels. As an example, for the core competency cognitive ability CC1, basic competency BC1 "adding numbers up to 99" has to be learned on the 1st-grade level, while basic competency BC3 "adding numbers up to

999" belongs to the curriculum of the second-grade level. Also, the topic of "multiplication" and "division of numbers" in grade 2 is related to the topic of "addition" and "subtraction" that are given in grade 1. Basic competency BC1 "adding number up to 99" is to be the prerequisite of basic competency BC3 "adding numbers up to 999". The basic competency, in turn, has the relationship with several specific competencies. The achievement of specific competencies is considered to be measurable, and it is assessed by several questions of different levels of difficulty: easy, medium, and hard. As an example, specific competency Cp1 can be assessed using questions q2 and q8. Both questions are classified as easy questions. Therefore, it is essential to classify the building blocks of our system – questions – correctly, according to the competency and the difficulty level.

Table 1 contains the specific competencies of our dataset with descriptions, related basic competencies and sample questions with the difficulty levels provided next to them. For example, specific competencies Cp1, Cp2, Cp3, Cp4, and Cp5 make up a basic competency BC1. Observe that both specific competencies Cp1, composed of the easy questions, and Cp5, composed of the hard questions, contribute to the same basic competency. The next table, Table 2, has a list of basic competencies from number one to six with descriptions, grade levels they belong to, and the related core competency groups [38].

3.2. **Rule-based expert system.** This topic is part of research in developing a competency-based assessment system for assessing student's competence at elementary schools in Indonesia, as shown in Figure 1. This paper focused on automatic question classification in the expert module. The contributions of this work are replacing human expert's role in knowledge engineering process and supersede expert judgment for classifying the competency and the difficulty level of questions. Figure 3 shows an architecture of the rule-based expert system for automatic question classification. The classification of competency-based question and difficulty level of question occurs in two stages, knowledge engineering stage and question classification stage. The rule sets of the knowledge base are delivered by classification rules generator. The rule sets are used to identify the related competency and difficulty level of a question in question classification stage.

3.2.1. *Knowledge engineering.* Typically, most of the expert systems involve expert as a knowledge engineer to produce the rules in knowledge engineering process [39]. In our system, a set of rules for knowledge base are extracted using classification rules generator. The knowledge base consists of the set of rules for the inference engine. During the first stage, the generator of classification rules learns from a dataset. The dataset contains addition and subtraction of numbers up to 99. It has a structure as shown in (1). There are five elements in this structure. N1 is number; it expresses the first operand. O is operator plus (+) or minus (−). N2 is number; it is stated as the second operand. S is '=' sign. And N3 is number, declared as result. A question mark ("?") replaces one of numbers as shown in Table 1.

$$\text{N1 O N2 S N3} \tag{1}$$

Features extraction is the series of operations executed to convert the dataset into classification features. There are four operations performed in features extraction, i.e., tokenization, tagging, define a missing number, and extract features. Tokenization is accomplished to break a stream of the question into symbols as tokens. There are five tokens as an output of each question statement, for example, the tokens from input "6 + ? = 14" are 6, +, ?, =, and 14. Tagging is the process of marking up a token as corresponding to a particular label. The token is labeled based on (1) structure. For each question, the first token is tagged with N1, the second token is tagged with O, the third token is tagged with N2, the fourth token is tagged with S, and the fifth token is tagged with N3. Outputs of tagger from the question "6 + ? = 14" are "6/N1", "+/O", "?/N2", "=/S", "14/N3". Table 3 shows the description of each token.

TABLE 1. List of example for specific competencies and questions

Specific competency	Description	Basic competency (Topic)	Difficulty level of related task	Example
Cp1	The student can calculate addition of two numbers, single digit number with single digit number, and the result is single digit number.	BC1 (Addition)	Easy	$1+2=?$ $3+?=7$ $?+5=8$
Cp2	The student is able to calculate addition of two numbers, single digit number with single digit number, and the result is two digits number.	BC1 (Addition)	Medium	$8+7=?$ $6+?=14$ $?+9=17$
Cp3	The student can calculate addition of two numbers, single digit number with two digits number, and the result is two digits number.	BC1 (Addition)	Medium	$3+15=?$ $7+?=18$ $5+?=17$
Cp4	The student can calculate addition of two numbers, two digits number with single digit number, and the result is two digits number.	BC1 (Addition)	Medium	$15+4=?$ $11+?=17$ $?+3=19$
Cp5	The student can calculate addition of two numbers, two digits number with two digits number, and the result is two digits number.	BC1 (Addition)	Hard	$25+21=?$ $20+?=45$ $?+14=70$
Cp6	The student can calculate subtraction of two numbers, single digit number with single digit number, and the result is single digit number.	BC2 (Subtraction)	Easy	$8-3=?$ $9-?=4$ $?-3=2$
Cp7	The student can calculate subtraction of two numbers, two digits number with single digit number, and the result is single digit number.	BC2 (Subtraction)	Medium	$15-7=?$ $12-?=4$ $?-6=8$
Cp8	Student can calculate subtraction of two numbers, two digits number with single digit number, and the result is two digits number.	BC2 (Subtraction)	Medium	$15-2=?$ $17-?=12$ $?-5=42$
Cp9	Student can calculate subtraction of two numbers, two digits number with two digits number, and the result is single digit number.	BC2 (Subtraction)	Medium	$16-13=?$ $19-?=6$ $?-21=8$
Cp10	Student can calculate subtraction of two numbers, two digits number with two digits number, and the result is two digits number.	BC2 (Subtraction)	Hard	$53-10=?$ $43-?=17$ $?-23=28$

TABLE 2. List of example for basic competencies

Basic competency	Description	Ability	Grade level
BC1	The student can describe and solve addition of numbers up to 99.	Cognitive ability	L1
BC2	The student can describe and solve subtraction of numbers up to 99.	Cognitive ability	L1
BC3	The student can describe and solve addition of number up to 999.	Cognitive ability	L2
BC4	The student can describe and solve subtraction of numbers up to 999.	Cognitive ability	L2
BC5	The student can solve daily problem related with addition of numbers up to 99.	Psychomotor ability	L1
BC6	The student can solve daily problem related with subtraction of numbers up to 99.	Psychomotor ability	L1

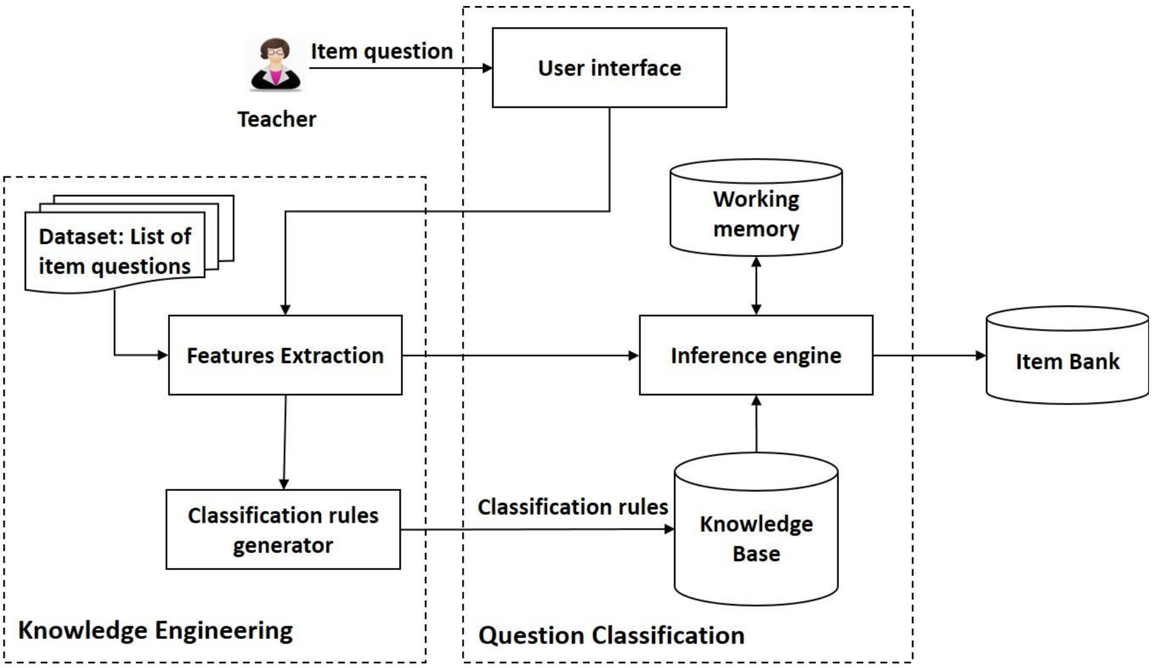

FIGURE 3. The architecture of the rule-based expert system for automatic question classification

Define a missing number is a process of calculating the value of the question mark. After calculation process, the value will replace the question mark. The following rule set, which consists of six rules, is used to define a missing number.

- Rule 1: if N3 = '?' AND O = '+' then N3 = N1 + N2;
- Rule 2: if N3 = '?' AND O = '−' then N3 = N1 − N2;

- Rule 3: if N2 = '?' AND O = '+' then N2 = N3 − N1;
- Rule 4: if N2 = '?' AND O = '−' then N2 = N1 − N3;
- Rule 5: if N1 = '?' AND O = '+' then N1 = N3 − N2;
- Rule 6: if N1 = '?' AND O = '−' then N1 = N3 + N2.

TABLE 3. Description of token

No.	Token	Description
1)	N1	First operand
2)	O	Operator, it can be addition operator or subtraction operator, plus (+) or minus (−) respectively
3)	N2	Second operand
4)	S	Equation sign
5)	N3	Result

If the missing number is the result (N3 = '?') and the operator is plus (O = '+'), then a value of the missing number is addition result of the first operand and the second operand (N3 ← N1 + N2). If the missing number is the result (N3 = '?') and the operator is minus (O = '−'), then the value of the missing number is subtraction result of the first operand and the second operand (N3 ← N1 − N2). If the missing number is the second operand (N2 = ?) and the operator is plus (O = '+'), then the value of the missing number is subtraction result of the result and the first operand (N2 ← N3 − N1). If the missing number is the second operand (N2 = '?') and the operator is minus (O = '−'), then the value of the missing number is subtraction result of the first operand and the result (N2 ← N1 − N3). If the missing number is the first operand (N1 = '?') and the operator is plus (O = '+'), then the value of the missing number is a subtraction of the result and the second operand (N1 ← N3 − N2). And if the missing number is the first operand (N1 = '?') and the operator is minus (O = '−'), then the value of the missing number is an addition of the result and the second operand (N1 ← N3 + N2). As an example, tag for the question "6 + ? = 14" is "6/N1 +/O ?/N2 =/S 14/N3", and then the missing number is N2. After running the rule, the value of N2 becomes 8, a subtraction result of 14 and 6. The last process in the features extraction is how to extract the features. It returned five features from each question, i.e., N1, O, N2, S, and N3. These features were applied to generating rules set for classification.

Classification rule generator is a generator to produce rules for classification. Classification is a common process associated with categorization, the process in which questions are identified, distinguished, and predicted. Dataset is divided into training and testing data. Class labels are assigned to the training data to generate the rules. Rule generator utilized five rule-based methods, i.e., OneR, RIPPER, PART, Fuzzy Unordered Rule Induction Algorithm (FURIA), and J48 to obtain the rules sets.

OneR method generated one rule, as follows.

- Rule 1: If O = '+' then *class* = Cp5 *else* if O = '−' then *class* = Cp10.

This rule only detects questions related to two competencies, Cp5 and Cp10. The rule classifies all questions with plus operator as competency Cp5 and all questions with minus operator as competency Cp10. Therefore, the questions related to Cp1, Cp2, Cp3, and Cp4 will be classified as Cp5. Also, the questions related to Cp6, Cp7, Cp8, and Cp9 will be classified as Cp10.

RIPPER method produced ten rules, as follows.

- Rule 1: If N3 <= 8 AND N1 <= 9 AND O = '−' Then class = Cp6;

- Rule 2: If N1 <= 7 AND N3 <= 9 Then class = Cp1;
- Rule 3: If N3 <= 9 AND N2 <= 9 Then class = Cp7;
- Rule 4: If N1 <= 9 AND N2 <= 9 Then class = Cp2;
- Rule 5: If N2 <= 8 AND O = '+' Then class = Cp4;
- Rule 6: If N1 <= 9 Then class = Cp3;
- Rule 7: If N2 <= 9 Then class = Cp8;
- Rule 8: If N3 <= 9 Then class = Cp9;
- Rule 9: If O = '−' Then class = Cp10;
- Rule 10: class = Cp5.

There are ten rules to represent ten specific competencies. The rules can recognize all the specific competencies in the dataset according to the competency's description.

PART method also generated ten rules which identify ten specific competencies, as follows.

- Rule 1: If O = '+' AND N1 <= 9 AND N2 > 9 Then class = Cp3;
- Rule 2: If O = '+' AND N2 > 9 Then class = Cp5;
- Rule 3: If O = '+' AND N1 > 9 Then class = Cp4;
- Rule 4: If N2 > 9 AND N3 > 9 Then class = Cp10;
- Rule 5: If N3 > 9 AND O = '−' Then class = Cp8;
- Rule 6: If N2 > 9 Then class = Cp9;
- Rule 7: If O = '−' AND N1 > 9 Then class = Cp7;
- Rule 8: If O = '+' AND N3 > 9 Then class = Cp2;
- Rule 9: If O = '+' Then class = Cp1;
- Rule 10: class = Cp6.

FURIA method also produced ten rules for ten specific competencies, as follows.

- Rule 1: If (N1 in [−inf, −inf, 8, 9]) AND (N3 in [−inf, −inf, 9, 10]) AND (O = '+') Then class = Cp1 (CF = 0.95);
- Rule 2: If (N1 in [−inf, −inf, 9, 10]) AND (N2 in [−inf, −inf, 9, 10]) AND (N3 in [−inf, −inf, 9, 10]) Then class = Cp2 (CF = 0.96);
- Rule 3: If (N1 in [−inf, −inf, 8, 10]) AND (N2 in [9, 10, inf, inf]) Then class = Cp3 (CF = 1.0);
- Rule 4: If (N2 in [−inf, −inf, 8, 10]) AND (O = '+') AND (N1 in [9, 10, inf, inf]) Then class = Cp4 (CF = 1.0);
- Rule 5: If (O = '+') AND (N1 in [9, 10, inf, inf]) AND (N2 in [9, 10, inf, inf]) Then class = Cp5 (CF = 1.0);
- Rule 6: If (N3 in [−inf, −inf, 8, 9]) AND (N1 in [−inf, −inf, 9, 10]) AND (O = '−') Then class = Cp6 (CF = 0.95);
- Rule 7: If (N3 in [−inf, −inf, 9, 10]) AND (N2 in [−inf, −inf, 9, 10]) AND (N1 in [9, 10, inf, inf]) Then class = Cp7 (CF = 0.96);
- Rule 8: If (N2 in [−inf, −inf, 9, 10]) AND (O = '−') AND (N3 in [9, 10, inf, inf]) Then class = Cp8 (CF = 1.0);
- Rule 9: If (N3 in [−inf, −inf, 9, 10]) AND (N2 in [9, 10, inf, inf]) Then class = Cp9 (CF = 1.0);
- Rule 10: If (O = '−') AND (N2 in [9, 10, inf, inf]) AND (N3 in [9, 10, inf, inf]) Then class = Cp10 (CF = 1.0).

The J48 method yielded ten rules for ten specific competencies, as follows.

- Rule 1: If O = '+' AND N1 <= 9 AND N2 <= 9 AND N3 < = 9 Then class = Cp1;
- Rule 2: If O = '+' AND N1 <= 9 AND N2 <= 9 AND N3 > 9 Then class = Cp2;
- Rule 3: If O = '+' AND N1 <= 9 AND N2 > 9 Then class = Cp3;

- Rule 4: If O = '+' AND N1 > 9 AND N2 <= 9 Then class = Cp4;
- Rule 5: If O = '+' AND N1 > 9 AND N2 > 9 Then class = Cp5;
- Rule 6: If O = '−' AND N2 <= 9 AND N3 <= 9 AND N1 <= 9 Then class = Cp6;
- Rule 7: If O = '−' AND N2 <= 9 AND N3 > 9 Then class = Cp7;
- Rule 8: IF O = '−' AND N2 <= 9 AND N3 > 9 THEN class = Cp8;
- Rule 9: IF O = '−' AND N2 > 9 AND N3 <= 9 THEN class = Cp9;
- Rule 10: IF O = '−' AND N2 > 9 AND N3 > 9 THEN class = Cp10.

The knowledge base in question classification stage uses rule sets from the best method. The best method is chosen based on the experiment result in Section 4.

3.2.2. *Question classification.* During a second stage, the rule-based expert system infers the competence of question and the difficulty level of question. The teacher communicates with the system using user interface. Features extraction extracts the features from the inputted question. Inference engine infers the competence of question and difficulty level of the question using rules in the knowledge base. Then the question, the competency of the question, and the difficulty level of question will be stored in the item bank. After inferring the competence of question, the inference engine infers the difficulty level of question. Following rules are rules to infer the difficulty level of question.

- Rule 1: If class = Cp1 or class = Cp6 then qdifflevel = 'easy';
- Rule 2: If class = Cp2 or class = Cp3 or class = Cp4 or class = Cp7 or class = Cp8 or class = Cp9 then qdifflevel = 'medium';
- Rule 3: If class = Cp5 or class = Cp10 then qdifflevel = 'hard'.

The questions in specific competency Cp1 or Cp6 are identified as the easy level of questions. The questions in specific competency Cp2, Cp3, Cp4, Cp7, Cp8, or Cp9 are identified as the medium level of questions. And the questions in specific competency Cp5 or Cp10 are identified as the hard level of questions.

4. **Results and Discussion.** The premise to be proved in this paper is that the constructed expert system behaves like a human expert to classify questions to the specific competency and the difficulty level of question. To analyze and validate the system, the system has been tested using addition and subtraction question for first-grade elementary schools. The study was done using rules set generated from five rule-based methods. This experiment also was designed to find the best method of five rule-based methods to classify the specific competency of the question. The results are analyzed using performance analysis.

4.1. **The experiment.** There were two setup of experiments. The first experiment is a performance analysis of five rule-based methods and the second experiment is question classification. Table 4 shows the number of questions used for the first and second experiments. There are 9454 questions used as a dataset in the first experiment and 1843 questions for the second experiment. Ten expert teachers from three elementary schools are involved to construct and label the dataset. Experts labeled the questions with specific competency label in Table 2.

Performance analysis of rule-based methods is a process to evaluate the result and the performance of five rule-based methods for classification in the first experiment. Classification of competency-based questions applied two test options, namely: cross-validation and percentage split. Evaluation is done in a variety of numbers of folds, i.e., 10, 15, 20, 25, and 30 and percentage of splits 70%, 75%, 80%, and 90%. The purpose of performance analysis is to find the best rule-based approach for classifying competence based questions. The analysis process utilizes several performance measurements for analysis; i.e.,

Table 4. Number of questions per class in dataset

No.	Specific competency	Difficulty level	Number of questions First experiment	Second experiment
1)	Cp1	Easy	37	37
2)	Cp2	Medium	45	45
3)	Cp3	Medium	685	280
4)	Cp4	Medium	685	280
5)	Cp5	Hard	3242	280
6)	Cp6	Easy	36	36
7)	Cp7	Medium	45	45
8)	Cp8	Medium	765	280
9)	Cp9	Medium	765	280
10)	Cp10	Hard	3240	280

the accuracy, Kappa, Mean Absolute Error (MAE), Root Mean Square Error (RMSE), precision, recall, F-measure, ROC, and time to build the model. The accuracy represents the amount of Correctly Classified Instances (CCI) divided by the total of Classified Instances (CI), multiplied by 100 to convert it into a percentage. The higher the accuracy is, the better performance of the method is. Kappa value is a metric used as a measure of reliability between the actual and predicted values of each classified instance. It compares a predicted accuracy and expected accuracy. The higher Kappa value is, the better the performance of the classifier is. MAE and RMSE are both used to appraise models by summarizing the distinctions between the actual value and predicted value. MAE yields the same weight to all errors, while RMSE yields extra weight to large errors. The lower MAE and RMSE errors are, the better performance classifier provides. Precision defines the fraction of records that in fact fits out to be positive in the group the classifier has stated as a positive class, while recall describes the fraction of pertinent instances that have been retrieved over the total amount of relevant instances. F-measure combines precision and recall to measure the performance. The higher the values of precision, recall, and F-measure are, the better the classifier is.

The second experiment is done by inputting the questions into the rule-based expert system. The purpose of the second experiment is to classify the competence and the difficulty level of question and evaluate the result. Rules set from the best method in the first experiment are used in the knowledge base. Correctly classified instances from the system are compared with the expert judgment for validation of the system.

4.2. **Performance analysis of rule-based methods and validation of the system.** Table 5 and Table 6 provide a comprehensive summary of the classification performance. As both tables shown, each approach is measured with an average of Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), precision, recall, F-measure, and ROC. Measurement used fold cross validation (10, 15, 20, 25, and 30 Fold) and percentage split (70%, 75%, 80%, 85%, and 90% PS). PART and J48 method can achieve the best performance value with the average of MAE to 0, the average of RMSE to 0, the average of precision to 0, average of recall till 0, the average of F-measure to 0, and the average of ROC to 1. Conversely, OneR only achieves 0.0642 for the average of MAE using fold cross-validation and 0.0673 using percentage split. The average value of RMSE using OneR is 0.2533 using fold and 0.2594 using percentage split. OneR only can reach the average of precision at 0.461 using fold and 0.4408 using percentage split. The average value of recall with OneR is 0.679 using fold and 0.663 using percentage split. The average value

Table 5. Summary of the average of all methods classification performance on the first experiment using fold cross validation

Method	MAE	RMSE	Precision	Recall	F-measure	ROC
OneR	0.0642	0.2533	0.461	0.679	0.549	0.757
RIPPER	0.0001	0.00936	0.9998	0.9998	1	1
PART	0	0	1	1	1	1
FURIA	0.0001	0.0058	1	1	1	1
J48	0	0	1	1	1	1

Table 6. Summary of the average of all methods classification performance on the first experiment using percentage split

Method	MAE	RMSE	Precision	Recall	F-measure	ROC
OneR	0.0673	0.2594	0.4408	0.663	0.5296	0.692
RIPPER	0	0.0121	1	1	1	0.692
PART	0	0	1	1	1	1
FURIA	0	0	1	1	1	1
J48	0	0	1	1	1	1

Table 7. Correctly and incorrectly classified instances (CCI and ICI) for the first experiment

	OneR		RIPPER		PART		FURIA		J48	
Test Mode	CCI	ICI	CCI	ICI	CCI	ICI	CCI	ICI	CCI	ICI
10 Fold CV	6482	3063	9541	4	9545	0	9542	3	9545	0
15 Fold CV	6482	3063	9541	4	9545	0	9541	4	9545	0
20 Fold CV	6482	3063	9540	5	9545	0	9541	4	9545	0
25 Fold CV	6482	3063	9541	4	9545	0	9541	4	9545	0
30 Fold CV	6482	3063	9541	4	9545	0	9541	4	9545	0
70% PS	1928	935	2863	0	2863	0	2863	0	2863	0
75% PS	1595	791	2386	0	2386	0	2386	0	2386	0
80% PS	1262	647	1909	0	1909	0	1909	0	1909	0
85% PS	941	491	1432	0	1432	0	1432	0	1432	0
90% PS	628	326	954	0	954	0	954	0	954	0

of F-measure by OneR is 0.549 using fold and 0.5296 using percentage split. The average value of ROC with OneR reached 0.757 using fold and 0.692 using percentage split. From this result, PART and J48 perform the best performance compared with OneR, RIPPER, and FURIA.

Table 7 shows correctly classified instances and incorrectly classified instances of all methods. Figure 4 depicts the level of accuracy for all methods using fold cross validation (10, 15, 20, 25, and 30) and percentage split (70%, 75%, 80%, 85%, and 90%). On average, OneR can classify 67% of questions correctly and 33% incorrectly. RIPPER and FURIA can classify 99% of questions correctly and 1% incorrectly. As well, PART and J48 can classify 100% of questions correctly.

Figure 5 depicts the Kappa performance of classification methods. PART and J48 can reach the best performance with Kappa = 1. Contrariwise, on average OneR can only achieve 0.50 Kappa value. Figure 6 shows time spent to generate the rules. OneR can generate rule fastest with average 0.01 seconds. On the other hand, on average RIPPER, PART, FURIA, and J48 need 0.23 seconds, 0.04 seconds, 0.40 seconds, and 0.08 seconds to generate the rules, respectively. From the first experiment it can be concluded, the best rules set is PART rules, because the PART method can generate rules set fastest and classify question 100% accurately.

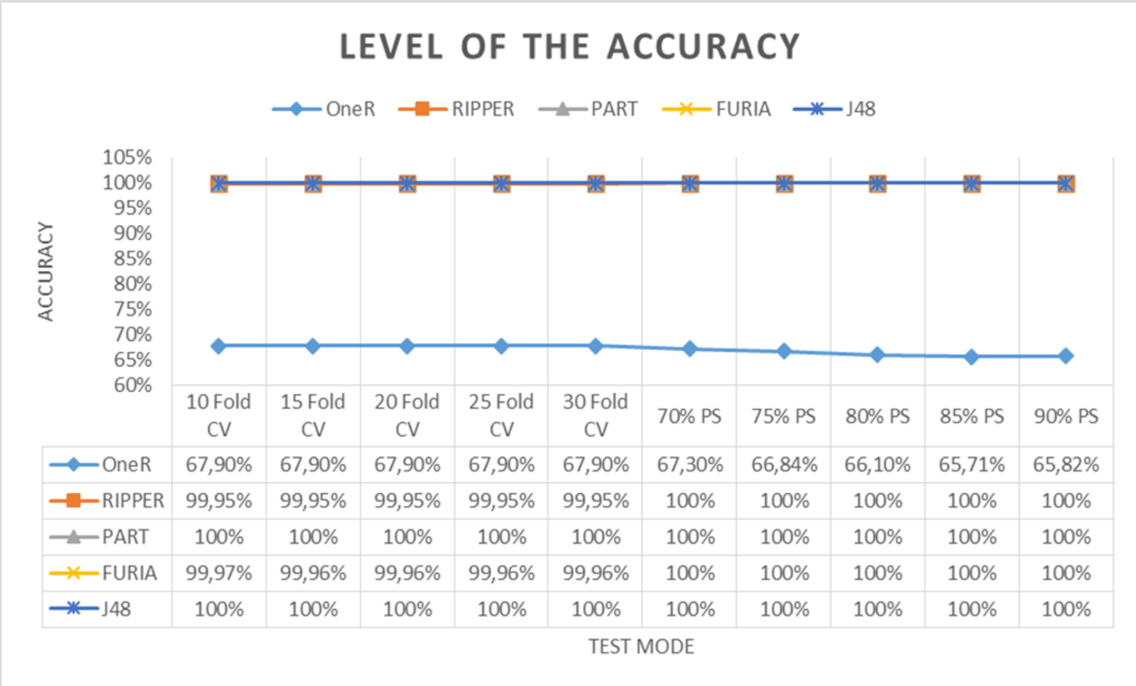

Figure 4. Level of the accuracy of classification methods in the first experiment

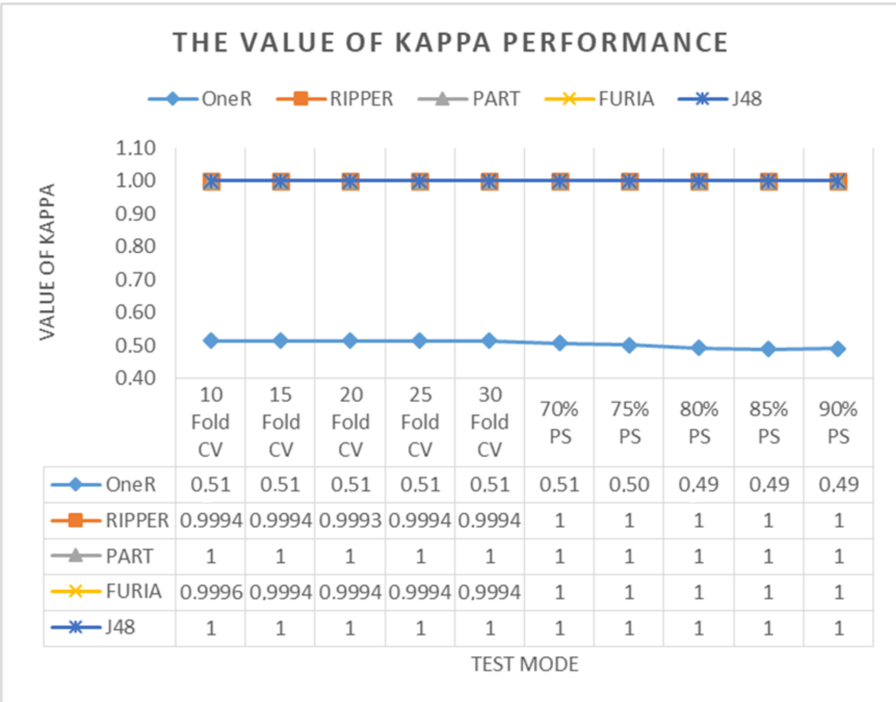

Figure 5. Kappa performance of classification methods in the first experiment

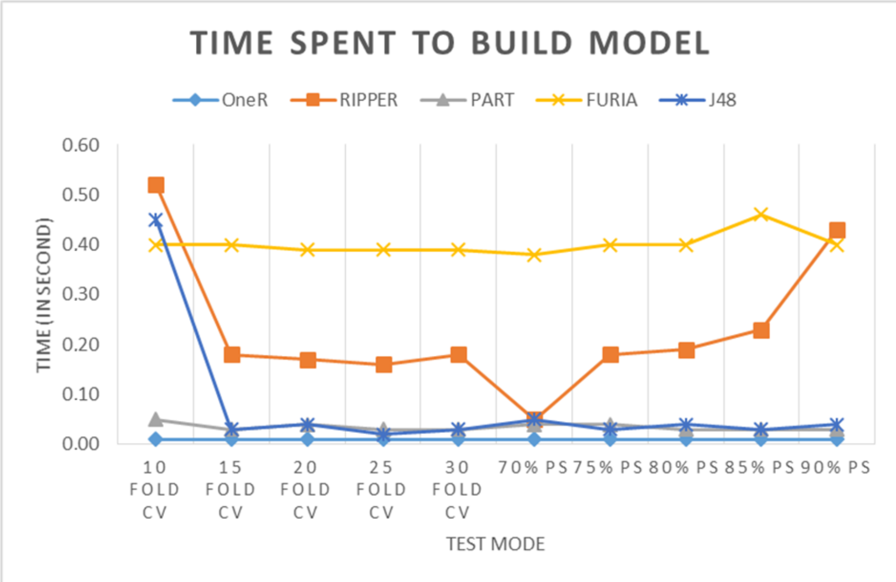

FIGURE 6. Time spent for rule generation using classification methods in the first experiment

TABLE 8. Correctly and incorrectly classified instances (CCI and ICI) for the second experiment

Specific competency	Number of questions	Classification by expert	Classification by system: CCI	ICI	Difficulty level of questions: Easy	Medium	Hard
Cp1	37	37	37 (100%)	0 (0%)	37	–	–
Cp2	45	45	45 (100%)	0 (0%)	–	45	–
Cp3	280	280	280 (100%)	0 (0%)	–	280	–
Cp4	280	280	280 (100%)	0 (0%)	–	280	–
Cp5	280	280	280 (100%)	0 (0%)	–	–	280
Cp6	36	36	36 (100%)	0 (0%)	36	–	–
Cp7	45	45	45 (100%)	0 (0%)	–	45	–
Cp8	280	280	280 (100%)	0 (0%)	–	280	–
Cp9	280	280	280 (100%)	0 (0%)	–	280	–
Cp10	280	280	280 (100%)	0 (0%)	–	–	280

Rules from PART method were implemented to the knowledge base for the second experiment. Table 8 shows the result for the second experiment. All the questions can be correctly classified using the rule-based expert system. Therefore, it can be concluded that the system can do the classification successfully.

5. **Conclusions.** This paper firstly introduces the competency-based assessment system and the significance to classify the question based on the competence. The rule-based methods for question classification are discussed and analyzed. An adaptive competency-based assessment system has been designed to assess the competencies of student adaptively. A rule-based expert system for automatically classifying the competency-based and the difficulty level of questions is proposed. In order to replace the human expert

role in defining the rules in the knowledge engineering process, a classification rules generator is built using the best rule-based method. Therefore, five rule-based methods are implemented in the dataset to find the best rule-based method that is proper with question pattern. Rule set from the best method is used in the knowledge base to infer the classification of question. Two contributions have been proposed in this work. The first contribution is automatically defining the rules in the knowledge engineering process by replacing the human expert using a rule generator. The second contribution is providing a rule-based expert system to classify the competency and the difficulty level of question. Using this system, teachers input the questions, and the system can classify the competency and the difficulty level of questions automatically and then store it in the item bank. The system has been tested using a real dataset, and the teachers have validated the result as human experts.

The first experiment has been done to find the best performance and the best rules set of the generator from five rule-based methods. The authors have compared five different rule-based models, i.e., OneR, RIPPER, PART, FURIA, and J48, in applying them to the mathematics questions database. The result shows that PART and J48 perform better than OneR, RIPPER, and FURIA with the lowest MAE, lowest RMSE, highest precision, highest recall, highest F-measure, and highest ROC. Conversely, OneR performed worse than RIPPER, PART, FURIA, and J48 in MAE, RMSE, precision, recall, F-measure, and ROC. Nevertheless, OneR executed fastest in building the model. Based on the experiment result, PART and J48 also showed the best performance than three other methods with 100% accuracy and Kappa = 1. Time spent PART is shorter than J48 in producing the rules set, meaning PART better than J48. It can be concluded that the best rule method to classify the competency-based question is PART.

The second experiment has been done to prove that the constructed expert system behaves like a human expert in classifying questions to the specific competency and the difficulty level of question. From the experimental result, it can be concluded that using PART rules in the knowledge base, the rule-based expert system can classify the competency and the difficulty level of questions perfectly. This method is promising to classify the same pattern of the question, such as multiplication and division of two numbers using multiplication and division sign ((x), (:)). The rule generator can produce the rules set for the knowledge base adaptively using a new dataset and PART rule-based method. However, other patterns of question are necessary to be classified and investigated in the next research.

Acknowledgment. The research for this paper was financially supported by The Ministry of Research and Higher Education Indonesia through Enhancing International Publication Program. The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation. The authors also thankfully acknowledge to the EFOP-3.6.1-16-2016-00017 Internationalization, initiatives to establish a new source of researchers and graduates, and development of knowledge and technological transfer as instruments of intelligent specializations at Szechenyi University, Hungary.

REFERENCES

[1] X. Roegiers, *A Conceptual Framework for Competencies Assessment*, UNESCO International Bureau of Education, 2000.

[2] J. Scheerens, C. Glas and S. M. Thomas, *Educational Evaluation, Assessment, and Monitoring: A Systemic Approach*, Swets & Zeitlinger, 2003.

[3] J. T. M. Gulikers, P. Runhaar and M. Mulder, An assessment innovation as flywheel for changing teaching and learning, *J. Vocat. Educ. Train.*, vol.6820, pp.1-20, 2017.

[4] J. M. Linacre, S. Chae, U. Kang and E. Jeon, Computer-adaptive testing: A methodology whose time has come, *MESA Memorandum No. 69*, 2000.
[5] C. Grosan and A. Abraham, *Rule-Based Expert Systems BT – Intelligent Systems*, 2011.
[6] G.-J. Hwang, A conceptual map model for developing intelligent tutoring systems, *Comput. Educ.*, vol.40, no.3, pp.217-235, 2003.
[7] R. Kavitha, A. Vijaya and D. Saraswathi, Intelligent item assigning for classified learners in ITS using item response theory and point biserial correlation, *Int. Conf. Comput. Commun. Informatics*, 2012.
[8] Ö. Özyurt, H. Özyurt and A. Baki, Design and development of an innovative individualized adaptive and intelligent e-learning system for teaching-learning of probability unit: Details of UZWEBMAT, *Expert Syst. Appl.*, vol.40, no.8, pp.2914-2940, 2013.
[9] T. H. Wang, Developing an assessment-centered e-learning system for improving student learning effectiveness, *Comput. Educ.*, vol.73, pp.189-203, 2014.
[10] I. Widiastuti and N. Z. Fanani, Adjustment levels for intelligent tutoring system using modified items response theory, *The 1st International Conference on Information Technology, Computer, and Electrical Engineering*, pp.217-221, 2014.
[11] K. Dolenc and B. Aberšek, TECH8 intelligent and adaptive e-learning system: Integration into technology and science classrooms in lower secondary schools, *Comput. Educ.*, vol.82, pp.354-365, 2015.
[12] G.-F. N. Ju and A. Bork, The implementation of an adaptive test on the computer, *Proc. of the 5th IEEE International Conference on Advanced Learning Technologies*, pp.822-823, 2005.
[13] N. Chaimongkol, S. Pasiphol and S. Kanjanawasee, Computerized adaptive testing with reflective feedback: A conceptual framework, *Procedia – Soc. Behav. Sci.*, vol.217, pp.806-812, 2016.
[14] S. Klinkenberg, M. Straatemeier and H. L. J. Van Der Maas, Computer adaptive practice of maths ability using a new item response model for on the fly ability and difficulty estimation, *Comput. Educ.*, vol.57, no.2, pp.1813-1824, 2011.
[15] B. R. J. Jansen, J. Louwerse, M. Straatemeier, S. H. G. Van Der Ven, S. Klinkenberg and H. L. J. Van Der Maas, The influence of experiencing success in math on math anxiety, perceived math competence, and math performance, *Learn. Individ. Differ.*, vol.24, pp.190-197, 2013.
[16] M. Ilahi-Amri, L. Cheniti-Belcadhi and R. Braham, A framework for competence based e-assessment, *Interact. Des. Archit. J. – IxD&A*, vol.N.32, pp.189-204, 2017.
[17] A. Mishra and S. K. Jain, A survey on question answering systems with classification, *J. King Saud Univ. – Comput. Inf. Sci.*, vol.28, no.3, pp.345-361, 2016.
[18] T. Aysel, A. O'Shea and S. Breen, A classification of questions from Irish and Turkish high-stakes examinations, *Proc. of the British Society for Research into Learning Mathematics*, pp.13-18, 2011.
[19] N. Omar and S. S. Haris, A rule-based approach in Bloom's taxonomy question classification through natural language processing, *The 7th Int. Conf. Comput. Converg. Technol. (ICCCT)*, pp.410-414, 2012.
[20] S. F. Kusuma, D. Siahaan and U. L. Yuhana, Automatic Indonesia's questions classification based on Bloom's taxonomy using natural language processing a preliminary study, *Proc. of International Conference on Information Technology Systems and Innovation (ICITSI 2015)*, 2015.
[21] D. Roth and X. Li, Learning question classifiers, *Proc. of the 19th International Conference on Computational Linguistics*, pp.556-562, 2002.
[22] Z. Huang, M. Thint and Z. Qin, Question classification using head words and their hypernyms, *Proc. of Conf. Empir. Methods Nat. Lang. Process.*, pp.927-936, 2008.
[23] F. Li, X. Zhang, J. Yuan and X. Zhu, Classifying what-type questions by head noun tagging, *The 22nd International Conference on Computational Linguistics*, pp.481-488, 2008.
[24] P. Biswas, A. Sharan, R. Kumar and S. Sciences, Question classification using syntactic and rule based approach, *International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp.1033-1038, 2014.
[25] E. Verdú, M. J. Verdú, L. M. Regueras, J. P. De Castro and R. García, A genetic fuzzy expert system for automatic question classification in a competitive learning environment, *Expert Syst. Appl.*, vol.39, no.8, pp.7471-7478, 2012.
[26] J. Han, M. Kamber and J. Pei, *Data Mining: Concepts and Techniques*, 2012.
[27] A. Peña-Ayala, Educational data mining: A survey and a data mining-based analysis of recent works, *Expert Syst. Appl.*, vol.41, no.4, pp.1432-1462, 2014.
[28] R. S. J. D. Baker and K. Yacef, The state of educational data mining in 2009: A review and future visions, *J. Educ. Data Min.*, vol.1, no.1, pp.3-16, 2009.

[29] R. C. Holte, Very simple classification rules perform well on most commonly used datasets, *Mach. Learn.*, vol.11, no.1, pp.63-91, 1993.

[30] W. W. Cohen, Fast effective rule induction, *Proc. of the 12th International Conference on Machine Learning*, pp.115-123, 1995.

[31] S. B. Kotsiantis, Supervised machine learning: A review of classification techniques, *Informatica*, vol.31, pp.249-268, 2007.

[32] D. Radev, W. Fan, H. Qi, H. Wu and A. Grewal, Probabilistic question answering on the web, *J. Am. Soc. Inf. Sci. Technol.*, vol.56, no.6, pp.571-583, 2005.

[33] E. Frank and I. H. Witten, Generating accurate rule sets without global optimization, *Proc. of the 15th International Conference on Machine Learning*, pp.144-151, 1998.

[34] J. Hühn and E. Hüllermeier, FURIA: An algorithm for unordered fuzzy rule induction, *Data Min. Knowl. Discov.*, vol.19, no.3, pp.293-319, 2009.

[35] J. Bayer, H. Bydzovská, J. Géryk, T. Obsivac and L. Popelínský, Predicting drop-out from social behaviour of students, *Proc. of the 5th Int. Conf. Educ. Data Min.*, pp.103-109, 2012.

[36] U. L. Yuhana, R. G. Mangowal, S. Rochimah, E. M. Yuniarno and M. H. Purnomo, Predicting math performance of children with special needs based on serious game, *The 5th IEEE Conference on Serious Game and Application for Health*, 2017.

[37] D. R. Krathwohl, A revision of Bloom's taxonomy: An overview, *Theory Pract.*, vol.41, no.4, 2002.

[38] Ministry of Education and Culture, Republic of Indonesia, *Appendices of the Regulation of Ministry of Education and Culture No. 24 Year 2016*, 2016.

[39] R. Studer, V. R. Benjamins and D. Fensel, Knowledge engineering: Principles and methods, *Data Knowl. Eng.*, vol.25, nos.1-2, pp.161-197, 1998.