



BiLSTM and dynamic fuzzy AHP-GA method for procedural game level generation

Murat İnce¹

Received: 17 December 2020 / Accepted: 27 May 2021, Published online: 19 June 2021
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2021

Abstract

The use of games in daily life, especially in education, has been in an incline during the COVID-2019 pandemic. Thus, game-based learning environments have caused an increase in the need of game contents, but generation of the game contents and levels is a time-consuming and costly process. Generated game contents and levels should be balanced, dense, aesthetic and reachable. Also, the time as well as the costs spent should be decreased. In order to overcome this problem, automatic and intelligent game content and level generation methods have emerged, and procedural content generation (PCG) is the most popular one of these methods. Artificial intelligence techniques are used for procedural game level generation instead of traditional methods. In this study, bidirectional long short-term memory (BiLSTM) and fuzzy analytic hierarchy process-genetic algorithm (FAHP-GA) methods were used to generate procedural game levels. This proposed hybrid system was used in a developed educational game as a case study to create game levels. The performance of the proposed study was compared to the other multi-criteria decision-making (MCDM) methods, and also further statistical analyses were investigated. The results showed that the BiLSTM-based FAHP-GA method can be used for procedural game level generation effectively.

Keywords Fuzzy analytic hierarchy process · Genetic algorithm · Bilstm · Game level generation

1 Introduction

Today, the use of computers as well as other information technologies (IT) in education is inevitable. Moreover, during times of crises like COVID-19 pandemic, the importance of IT is even higher than usual times [1]. Traditional education methods have evolved into concepts such as web-based education, computer-based education, and distance education thanks to IT. Computers, tablets, smartphones, and many other IT devices are used by the majority of people, which has led to the development of individualized education systems. Individualized educational environments offer users learning activities independent from time, place and other people. Educational games are one of the best examples of such individualized

educational environments. Games are as effective as other educational tools in terms of achievement [2]. Thanks to the increase and the availability of IT technologies, there has been a shift in online gaming [3]. COVID-19 led millions to spend more time at homes, and this has led online games to be more popular than before [4]. Games are popular activities for people at all ages, and it is an inevitable piece of social and cultural life because they provide enjoyment, involvement, adrenaline and social interaction to the players. The most recent growing trend is game-based learning (GBL) by using the online games [5]. Game-based learning is in widespread use in education methodology [6] with many areas [7] such as Mathematics education [8], English language education [9], and Science education [10]. Game-based learning has a positive effect on learning [11] and traditional e-learning environments benefit a lot from games because games include educational goals, rules, restrictions [12]. GBL helps learners to develop effective learning, critical thinking and problem-solving skills, achievement, cognitive enhancement and intensive motivation [13]. Moreover, GBL is an effective

✉ Murat İnce
muratince@isparta.edu.tr

¹ Vocational School of Technical Sciences, Isparta University of Applied Sciences, Cunur West Campus, Isparta 32200, Turkey

environment in both formal and informal educational settings and improves analysis, synthesis and evaluation skills [14]. It also helps learners to solve daily life problems [15]. The traditional games have features such as fun, rules, goals, interactions, results, feedback, conflict, challenge, and creativity [16]. In the educational GBL environment, these features also exist, but it also aims to teach the player certain subjects. It combines learning content with game elements. Educational games aim to teach something and can also be entertaining. It increases interest, concentration and motivation [17]. Thus, it increases the player's interest in learning and satisfaction. Therefore, they provide flow experience, which is a phenomenon where people enjoy and concentrate on activity without any disruption from outside [18]. Moreover, GBL environments increase student achievement [19] and engagement, so helps learning [20] and it becomes an effective learning environment [21].

The contribution of game-based learning to education cannot be underestimated, but there are many differences among students. Students' preferences and experiences during the game affect the game environment [22]. One of the most important features of web-based learning environments is that it engages learners. Game-based learning methods can enhance standard learning environments by providing individual learning and motivation. If the challenge is higher than the skill of the player, anxiety will occur, and if the opposite happens, boredom might arise. As the skill increases, the challenge must get harder [23]. In order to provide motivation, independence, responsibility and flexibility, student centered game-based learning has emerged. Thus, different and adaptive game levels can be generated automatically, but this can be a time-consuming and a costly process if it is performed on the basis of conventional game generation [24]. Also, game structure, presentation, rules, event, scenario, object and player are all important [25] for achieving the best game level. Moreover, gaining attention, informing learning objectives, recalling prior learning, presenting learning content, providing learning guidance, eliciting performance, provide feedback and evaluate performance are among other important aspects of game level generation [26]. The process of generating game levels requires speed, accuracy, convenience, and diversity. Random generator, constructive generator and searched-based generator are three types of PCG methods. Density, aesthetics, balance, suitability, symmetry and reachability are important in searched-based level generators [27]. Moreover, content generation reduces human workload and provides specific type of content. Level design combines challenge, competition and interaction, which all make games enjoyable [28]. In the literature, there is no single best type of PCG method or algorithm.

In this proposed study, regarding the essentials of the PCG, the game environment can change adaptively to the preferences of the player. For this purpose, BiLSTM which is a popular deep learning method and FAHP-GA are used together. The game is started according to many performance criteria at the beginning of the game as follows: question difficulty (P_a) which is difficulty level of question in the span for player to answer correctly, obstacle count (P_b) which is generated obstacle count in the time span for player to pass without crashing or being trapped and coin count (P_c) which is shown coin count in the time span for player to collect. During further game levels, these criteria are recalculated with previously obtained trained level data dynamically according to the player's preferences. Thus, the game environment is constantly changed.

The rest of the paper is as follows. Section 2 gives related work about procedural content generation. Section 3 describes the methods used in the proposed system. Section 4 explains the proposed approach on game level generation. Section 5 details the application and evaluation of the proposed system on an educational game. The final section presents the conclusion of the paper.

2 Related work

Procedural content generation is producing game content by algorithms with limited human intervention or without any intervention [29]. There are many methods for developing game content. These are constructive, searched based, and machine learning-based techniques [30]. Constructive techniques are fast and effective but are not suitable for complex contents. Searched-based methods are based on scoring content according to a fitness function. The common method used in game content development is the searched-based procedural content generation method. In the study of Zafar's et al. [27], general levels for 2D games were produced using the FI2POP genetic algorithm. The aesthetics and difficulty of the game are emphasized. Symmetrical, intense, and accessible game levels were successfully produced in 5 different games. Game level generation provides variety and reduces development time and cost, augmenting human creativity and enabling adaptivity [31]. Generally, rule generation [32], levels [33], maps [34] are produced. Searched-based PCG functions are direct evaluation function (preferences of the player), simulation-based evaluation function (agent based) and interactive evaluation functions (eye, speech quality) [35]. Moreover, machine learning [36] and deep learning methods are also available for level generation [37–39]. A multi-faceted surrogate model for searched-based PCG used deep learning architecture that is trained on a large corpus of randomly generated levels, classes and

simulations. This system adjusts human design to desired gameplay environment [40]. Answer set programming [41] and Quality Diversity algorithm [42] are among other PCG methods. Data-driven PCG used GA and support vector machines to automatically generate adaptive educational game content. Players realized greater performance from playing contents tailored to their capabilities rather than uncustomized game content [43]. Experience-driven PCG provides personalization of player experience with the affective and cognitive modeling in a real-time adjustment of content [44]. Satisfaction of the player requirements and preferences is important for the effective and meaningful game level generation [45].

3 Methods

Game level and content generation is a time-consuming and costly process. There are many game level generation methods, but the most popular one is procedural content generation method. In this part, artificial intelligence and decision-making methods such as BiLSTM, FAHP and GA methods that were used in the study were explained.

3.1 Fuzzy AHP and genetic algorithm

Multi-criteria decision making methods are used for many different purposes to solve complex problems considering the user defined preferences and criteria [46–48]. Analytic Hierarchy Process (AHP), the most popular multi criteria decision making method, is easy to use, flexible and effective because of having basic mathematical expressions [49] and executable both quantitative and qualitative criteria by changing comparison matrix values of priorities to get more sensitive results. However, imprecise and uncertain users' preferences cannot be satisfied sufficiently by AHP, and thus it is criticized [50] because of using Crisp numbers in calculations [51]. Sometimes, using exact numbers (Crisp) for comparison gives imponderable results [52]. Therefore, Fuzzy Analytic Hierarchy Process (FAHP) is needed to satisfy unforeseeable and imprecise conditions of the user's preferences.

Fuzzy AHP method uses fuzzy numbers to compare alternatives and criteria for imprecise judgments and vagueness of the human choices [53]. Uncertain judgments are used in daily life, usually instead of certain judgments. FAHP uses fuzzy triangular numbers (TFNs) to represent users' choices rather than crisp values that AHP uses [54]. AHP is based on perception, so the FAHP is more expressive for user decisions as compared to AHP [55]. Thus, FAHP can effectively be used for many different areas such as supplier selection [56] and test sheet preparation problems [57]. Traditional FAHP was modified with

Extent Analysis Method by Chang [58] to be used easily in among other MDCM methods. In many studies, FAHP was supported with genetic algorithm (GA) as a hybrid method to satisfy unforeseeable and imprecise preferences of the users.

One of the popular evolutionary algorithms is genetic algorithm, which is used for solving difficult and complex problems for many purposes by using encoded problem solution parameters [59]. GA selects best solution candidates and ensures them to live in population of solution space [60]. GA has some basic steps such as defining the initial population, evaluating convenience of chromosomes in population with using fitness function, selection of best suitable chromosomes and mutation of chromosomes. These steps are repeated until the best solution is found or stop condition is provided [61]. The GA is used to search and optimize solutions for problems in solution space with defined mathematical models and functions, and it tries without being trapped to local maximum or minimums. Therefore, MCDM methods are combined and enhanced with GA to find the best solutions for problems [62]. In many studies, the AHP method includes GA as a tool [63–65]. Conversely, in some studies, the weights of each criterion for the fitness evaluation of chromosomes in the GA are calculated with AHP [57, 66–68]. In this proposed study, different from the aforementioned studies, hybrid FAHP-GA model was used for procedural game level generation, but criteria and sub-criteria values are changed dynamically in an automatic and intelligent way by using BiLSTM method. Adaptively calculated FAHP weights are used for fitness evaluation of chromosomes in GA to generate the best game level regarding the player's preference parameters.

In this proposed study, hybrid FAHP-GA module is used for defining new level's game dynamics such as question difficulty (P_a) which is difficulty level of question in the span for player to answer correctly (MC1), obstacle count (P_b) which is generated obstacle count in the time span for player to pass without crashing or being trapped (MC2) and coin count (P_c) which is shown coin count in the time span for player to collect (MC3). These properties are considered as main criteria of the game level generation problem. Each main criterion also has sub-criteria that are shown in the judgment hierarchy of the proposed FAHP-GA method in Table 1.

The proposed method uses Chang's FAHP method (first six steps) which is expressed as follows [58]:

Step 1: The qualitative and quantitative criteria and sub-criteria are determined, based on game level properties. Then, pairwise comparisons of criteria and sub-criteria were calculated by using TFNs instead of crisp numbers as in AHP (Table 2) [68]. TFNs were adapted from Kahraman et al. [53] and Chan et al. [69].

Table 1 Sub-criteria of proposed FAHP-GA method

| Sub-criteria of MC1 | Sub-criteria of MC2 | Sub-criteria of MC3 |
|---------------------|---------------------|---------------------|
| SC1: Very easy | SC6: Very few | SC11: Very few |
| SC2: Easy | SC7: Few | SC12: Few |
| SC3: Medium | SC8: Medium | SC13: Medium |
| SC4: Difficult | SC9: Several | SC14: Several |
| SC5: Very difficult | SC10: Many | SC15: Many |

Table 2 Crisp and TFN value scale for the proposed study [68]

| Linguistic variables | Crisp | TFNs | Reciprocal of TFNs |
|-------------------------|-------|--------------|--------------------|
| Equally preferred | 1 | 1, 1, 1 | 1, 1, 1 |
| Moderately preferred | 3 | 0.66, 1, 1.5 | 0.66, 1, 1.5 |
| Strongly preferred | 5 | 1.5, 2, 2.5 | 0.4, 0.5, 0.66 |
| Very strongly preferred | 7 | 2.5, 3, 3.5 | 0.285, 0.333, 0.4 |
| Extremely preferred | 9 | 3.5, 4, 4.5 | 0.222, 0.25, 0.285 |

Step 2: Let $X = \{x_1, x_2, \dots, x_n\}$ be an object set, and $U = \{u_1, u_2, \dots, u_n\}$ be a goal set. The extent analysis for each goal object is performed, respectively. Thus, m extent analysis parameters for each object are obtained with the following expressions:

$\tilde{M}_{gi}^1, \tilde{M}_{gi}^2, \dots, \tilde{M}_{gi}^j$, where all the $\tilde{M}_{gi}^j = (i = 1, 2, \dots, n \text{ and } j = 1, 2, \dots, m)$ are TFNs. The steps of extent analysis defined as follows:

Step 3: The value of fuzzy synthetic extent with respect to the i th object is described as:

$$S_i = \sum_{j=1}^m M_{gi}^j \otimes \left[\sum_{i=1}^n \sum_{j=1}^m M_{gi}^j \right]^{-1} \tag{1}$$

To obtain $\sum_{j=1}^m M_{gi}^j$, and succeed the fuzzy addition operation of m extent analysis values for a particular matrix such that:

$$\sum_{j=1}^m M_{gi}^j = \left(\sum_{j=1}^m l_j, \sum_{j=1}^m m_j, \sum_{j=1}^m u_j \right) \tag{2}$$

and to obtain $\left[\sum_{i=1}^n \sum_{j=1}^m M_{gi}^j \right]^{-1}$, and succeed the fuzzy addition operation of $M_{gi}^j (j = 1, 2, \dots, m)$ values such that:

$$\sum_{i=1}^n \sum_{j=1}^m M_{gi}^j = \sum_{i=1}^n l_i, \sum_{i=1}^n m_i, \sum_{i=1}^n u_i \tag{3}$$

and then assess the inverse of the vector above, such that:

$$\left[\sum_{i=1}^n \sum_{j=1}^m M_{gi}^j \right]^{-1} = \left(\frac{1}{\sum_{i=1}^n u_i}, \frac{1}{\sum_{i=1}^n m_i}, \frac{1}{\sum_{i=1}^n l_i} \right) \tag{4}$$

Step 4: As $\tilde{M}_1 = (l_1 m_1 u_1)$ and $\tilde{M}_2 = (l_2 m_2 u_2)$ are two TFNs, the degree of possibility of $M_2 = (l_2 m_2 u_2) \geq M_1 = (l_1 m_1 u_1)$ described as:

$$V(\tilde{M}_2 \geq \tilde{M}_1) = \sup_{y \geq x} \min(\mu_{\tilde{M}_1}(x), \mu_{\tilde{M}_2}(y)) \tag{5}$$

and can be equally stated as follows:

$$V(\tilde{M}_2 \geq \tilde{M}_1) = \text{hgt}(\tilde{M}_1 \cap \tilde{M}_2) = \mu_{M_2}(d) = \begin{cases} 1, & \text{if } m_2 \geq m_1 \\ 0, & \text{if } l_1 \geq u_2 \\ \text{otherwise, } & \frac{l_1 - u_2}{(m_2 - u_2) - (m_1 - l_1)} \end{cases} \tag{6}$$

where d is the ordinate of the highest intersection point D between $\mu_{\tilde{M}_1}$ and $\mu_{\tilde{M}_2}$, as illustrated in Fig. 1.

Step 5: The presumption degree of a convex fuzzy number to be greater than k convex fuzzy $M_i(1, 2, k)$ numbers can be expressed with:

$$V(M \geq M_1, M_2, \dots, M_k) = V[(M \geq M_1) \text{ and } (M \geq M_2) \text{ and } \dots \text{ and } (M \geq M_k)] = \min V(M \geq M_i), i = 1, 2, 3, \dots, k \tag{7}$$

Assume that $d(A_i) = \min V(S_i \geq S_k)$ for $k = 1, 2, \dots, n; k \neq i$.

Then the weight vector is given by: $W' = (d'(A_1), d'(A_2), \dots, d'(A_n))^T$ where $A_i = (i = 1, 2, \dots, n)$ are n elements.

Step 6: Normalized weight vectors are defined as:

$$W = (d(A_1), d(A_2), \dots, d(A_n))^T \tag{8}$$

where W is a non-fuzzy number and is used as weights for the corresponding genes of chromosomes in the GA fitness function.

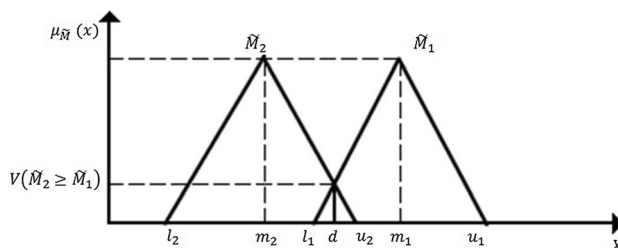
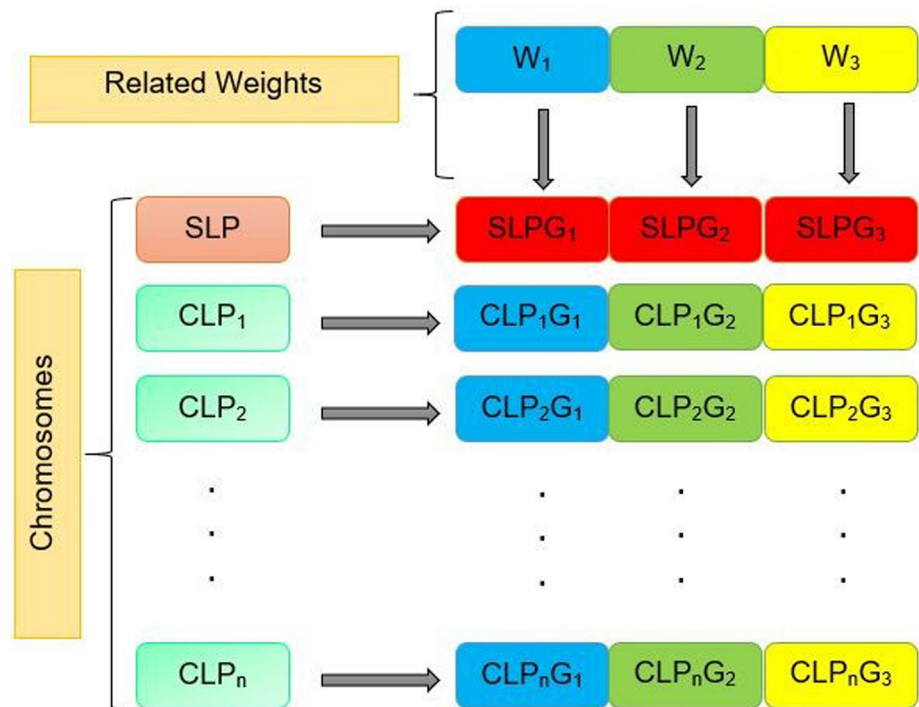


Fig. 1 Intersection point “d” between two fuzzy numbers M_1 and M_2 [58]

Fig. 2 Encoded game level properties as chromosome



Step 7: Then, game level properties (P_a, P_b, P_c) are encoded as gene in the chromosomes for the fitness function evaluation by using related W for each gene (Fig. 2).

SLP is the expected properties for the following game level which will be generated according to the previous level’s difficulty. CLP₁, CLP₂... CLP_n are candidate game level properties in the solution area of the problem. Each gene corresponds to a game level property. SLPG_j to SLPG_m are genes (criterion) of the searched game level property chromosome where m is the gene count (criterion) in the chromosome. CLP_iG_j to CLP_nG_m are genes (criterion) of the candidate game level property chromosome where n is the criteria count in the solution search area, and m is the gene count in chromosome, ($i = 1, 2, \dots n$), ($j = 1, 2, \dots m$). Each gene is multiplied with related weight in W matrix as quotient of the precedence. SCM is related sub-criteria synthetic value in comparison matrix for game level properties (genes). Thus, the fitness function is expressed as:

$$F(x) = \sum_{i=1}^n W_i SCM_{cs} \tag{9}$$

where c is index of CLP_iG_j in corresponding candidate sub-criteria comparison matrix (SCM_j), and s is index of SLPG_j in substituted searched sub-criteria comparison matrix (SCM_j). Tables 3, 4 and 5 show sub-criteria comparison matrixes, which are SC₁ to SC_m where m is gene (criterion) count. These comparison matrixes are steady in the proposed system.

Step 8: Each chromosome in the solution population is subject to $F(x)$ fitness function in order to evaluate the suitability. Then, the best chromosomes that maximize the $F(x)$ are picked to the new population by election rate. Simultaneously, new chromosomes are regenerated and mutated for the new population. In this study, population size was 100, mutation rate was 0.25, election rate was 0.15, and maximum generation count was 200. When stop condition is provided or maximum generation count is reached, the GA ends. The last population chromosomes, in this way, game level properties are listed as best solutions

Table 3 Question difficulty criterion (MC1) comparison matrix SCM₁

| | SC1 | SC2 | SC3 | SC4 | SC5 |
|-----|--------------|--------------|----------------|-------------------|--------------------|
| SC1 | 1, 1, 1 | 0.66, 1, 1.5 | 0.4, 0.5, 0.66 | 0.285, 0.333, 0.4 | 0.222, 0.25, 0.285 |
| SC2 | 0.66, 1, 1.5 | 1, 1, 1 | 0.66, 1, 1.5 | 0.4, 0.5, 0.66 | 0.285, 0.333, 0.4 |
| SC3 | 1.5, 2, 2.5 | 0.66, 1, 1.5 | 1, 1, 1 | 0.4, 0.5, 0.66 | 0.285, 0.333, 0.4 |
| SC4 | 2.5, 3, 3.5 | 1.5, 2, 2.5 | 1.5, 2, 2.5 | 1, 1, 1 | 0.66, 1, 1.5 |
| SC5 | 3.5, 4, 4.5 | 2.5, 3, 3.5 | 2.5, 3, 3.5 | 0.66, 1, 1.5 | 1, 1, 1 |

Table 4 Obstacle count criterion (MC2) comparison matrix SCM_2

| | SC6 | SC7 | SC8 | SC9 | SC10 |
|------|--------------|--------------|----------------|-------------------|--------------------|
| SC6 | 1, 1, 1 | 0.66, 1, 1.5 | 0.4, 0.5, 0.66 | 0.285, 0.333, 0.4 | 0.222, 0.25, 0.285 |
| SC7 | 0.66, 1, 1.5 | 1, 1, 1 | 0.66, 1, 1.5 | 0.4, 0.5, 0.66 | 0.285, 0.333, 0.4 |
| SC8 | 1.5, 2, 2.5 | 0.66, 1, 1.5 | 1, 1, 1 | 0.4, 0.5, 0.66 | 0.285, 0.333, 0.4 |
| SC9 | 2.5, 3, 3.5 | 1.5, 2, 2.5 | 1.5, 2, 2.5 | 1, 1, 1 | 0.66, 1, 1.5 |
| SC10 | 3.5, 4, 4.5 | 2.5, 3, 3.5 | 2.5, 3, 3.5 | 0.66, 1, 1.5 | 1, 1, 1 |

Table 5 Coin count criterion (MC3) comparison matrix SCM_3

| | SC11 | SC12 | SC13 | SC14 | SC15 |
|------|--------------|--------------|----------------|-------------------|--------------------|
| SC11 | 1, 1, 1 | 0.66, 1, 1.5 | 0.4, 0.5, 0.66 | 0.285, 0.333, 0.4 | 0.222, 0.25, 0.285 |
| SC12 | 0.66, 1, 1.5 | 1, 1, 1 | 0.66, 1, 1.5 | 0.4, 0.5, 0.66 | 0.285, 0.333, 0.4 |
| SC13 | 1.5, 2, 2.5 | 0.66, 1, 1.5 | 1, 1, 1 | 0.4, 0.5, 0.66 | 0.285, 0.333, 0.4 |
| SC14 | 2.5, 3, 3.5 | 1.5, 2, 2.5 | 1.5, 2, 2.5 | 1, 1, 1 | 0.66, 1, 1.5 |
| SC15 | 3.5, 4, 4.5 | 2.5, 3, 3.5 | 2.5, 3, 3.5 | 0.66, 1, 1.5 | 1, 1, 1 |

and sorted by their fitness function values from largest to smallest.

3.2 Long short-term memory

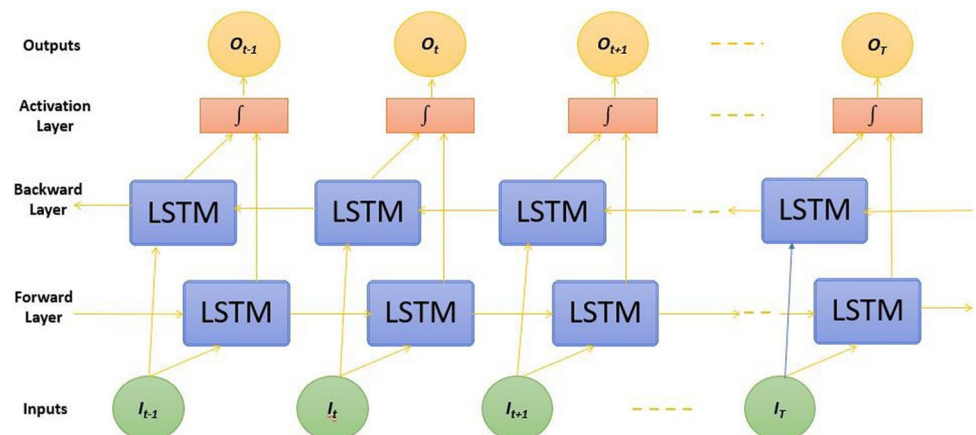
Deep learning algorithms are used in many areas such as image processing, classification and natural language processing [70]. These methods are different from classical Artificial Neural Networks (ANN) in various ways such as layer numbers. Recurrent Neural Networks (RNN) are the well-known deep learning algorithms [71] which can process input sequences such as time series problems, but sometimes gradient descend or ascend problem may occur [72]. Thus, gradient loss problems can cause learning problems, so correct relations between sequences cannot be found appropriately by the RNNs. For this reason, LSTM, a special RNN version, was developed [73]. However, LSTM can fail in sequential operations such as time series because of having process on data in one direction [74]. For this reason, BiLSTMs, having two LSTM (backward and forward direction), were developed for process input

sequence is all time steps with two direction (Fig. 3). The first LSTM processes the input sequence from backward to forward, while the second one does vice-versa [75]. Both LSTM are on the copy of the same input sequence [76]. Therefore, the BiLSTM system can learn the problem faster and more effectively.

3.3 Evaluation metrics

In this proposed study, hybrid FAHP-GA model was used for procedural game level generation, but criteria and sub-criteria values are changed dynamically in an automatic and intelligent way. Adaptively calculated FAHP weights are used for fitness evaluation of chromosomes in GA to generate the best game level regarding the player’s preference parameters. The quality of the generated game levels must be evaluated. Procedural content generation algorithms mostly rely on the quality of the content which they generate [77]. There are a lot of computational metrics generally for 2D games but, in this proposed study, the developed educational game is a 3D game, and 20 levels

Fig. 3 BiLSTM schema [75]



were generated. Thus, suitable computational metrics are chosen for evaluation as follows [78]:

- Leniency is the player's challenge in a level which is calculated as the sum of the lenience value of all the objects by divided level length. Then, it is normalized into the range [0–1] and high leniency value means more challenging game level.
- Density is the distribution of the objects on the same axes in the game level which is calculated as the sum of the objects in the defined axes and normalized into the range [0–1]. Higher value means more denser level.
- Negative space is the percentage of the empty space which can be used by the player to escape or to use for other purposes. Higher values mean more enjoyable and aesthetically pleasing game level.
- Balance measures how the objects are well distributed.
- Reachability measures the proportion of elements which are reachable by the player in the level.

Furthermore, the success of the proposed FAHP-GA method is compared to other AHP and FAHP methods as regards the performance. Statistical methods and Pearson correlation were used, and results were analyzed. Also, the success of the BiLSTM is compared to other prediction methods such as LSTM, ANN and some other regression methods.

4 Proposed procedural game level generation system

Procedural content game generation methods are mostly used on the basis of direct evaluation of the player preferences. Challenge goal of the game must be uncertain by producing variable difficulty levels, multiple level goals and randomness [45]. Thus, in this proposed study, BiLSTM-based hybrid FAHP-GA algorithm generates variable levels with multiple goals from player's preferences which are gained with previous game level. In the first level, player starts to the game predefined with main criteria preferences which are medium for P_a , P_b and P_c parameters. Each generated game level has the same time length, which is 30 min. At the end of each 20 s time period (span) in the level; P_a proportion (correct answer count/total question count in the time period), P_b proportion (avoided P_b without crash or trapped/total P_b in the time period), P_c proportion (collected P_c /total P_c in the time period), and total score are calculated to produce the dataset. At the end of the level, each proportion of the criteria (input values) and the total score (output value) are given to the BiLSTM for training. Then, in the following game level, at the end of each 20 s time period (span); P_a proportion, avoided P_b and total P_c proportion are gained

again. Current proportions are used to predict the sequencing total score. Then, if the predicted total score is greater than the real total score of the previous level, so the game level is easy. Thus, questions should be made more difficult, obstacles should be increased, and coin count should be reduced for the following game level. If the predicted total score is smaller than the real total score of the previous level, the game level is hard. Thus, questions should be made less difficult, obstacles should be decreased, and coin count should be increased for the following game level. While defining the values of these P_a , P_b and P_c game properties (sub-criteria), the difference between current proportion and related proportion in the previous level, which interval it falls, is used (Table 6). The corresponding interval of the game property is calculated with (current proportion – previous level proportion)/(current proportion + previous level proportion). To sum up answered question proportion, avoided obstacle proportion and total coin proportion are used as input for BiLSTM. Total level score is used as output for BiLSTM. According to the input proportion values of the current level, total score is predicted as output and this score is compared with the previous level's total score. Thus, whether the current level is difficult or easy is determined.

When the sub-criteria values (input values for FAHP-GA) are determined by the intervals which are defined according total score difference between the current level and the previous level, the FAHP-GA method is used for generating the next level according to the defined criteria values. When stop condition is provided or maximum generation count is reached, the GA ends. The last population chromosomes (output values of FAHP-GA), in this way, game level property parameters (P_a , P_b , and P_c) are listed as best solutions and sorted by their fitness function values from largest to smallest.

5 Application and evaluation

5.1 Game genre and scenario

The proposed hybrid method was applied in an educational game which was developed for this study. In the game, it is aimed to train users either consciously or unconsciously by asking questions according to the progress and the level of the user, as well as performing certain tasks and learning new information. Type of this game is task based, adventure, puzzle and role-playing. The player's progress depends on the performance. When the player performs in the game level, collected coins (awards), correct answered question count, avoided obstacle count without crashing or being trapped and total level score are calculated at the end of the level. For example, if the player spends a lot of time,

Table 6 Values of the sub-criteria according to the intervals of the difference

| MC1 values for next level | | MC2 values for next level | | MC3 values for next level | |
|------------------------------|-----------------|------------------------------|-----------------|------------------------------|-----------------|
| Sub-criteria value (P_a) | Interval | Sub-criteria value (P_b) | Interval | Sub-criteria value (P_c) | Interval |
| Very easy | (-1)–(-0.67) | Very few (4) | (-1)–(-0.67) | Very few (4) | (-1)–(-0.67) |
| Easy | (-0.66)–(-0.34) | Few (8) | (-0.66)–(-0.34) | Few (8) | (-0.66)–(-0.34) |
| Medium | (-0.33)–(0.33) | Medium (12) | (-0.33)–(0.33) | Medium (12) | (-0.33)–(0.33) |
| Difficult | (0.34)–(0.66) | Several (16) | (0.34)–(0.66) | Several (16) | (0.34)–(0.66) |
| Very difficult | (0.67)–(1) | Many (20) | (0.67)–(1) | Many (20) | (0.67)–(1) |

while the obstacles in the game are reduced, the number and difficulty of the obstacles may increase for the fast-moving players. Similarly, when the player answers the questions too correctly, the difficulty level of new questions may increase. Depending on the condition of such criteria, the total score to be obtained from the game can vary and the game environment can be shaped according to the player’s preferences.

5.2 Application and evaluation on game

At the beginning of the game, player can adjust P_a criteria (MC1), P_b criteria (MC2) and P_c criteria (MC3) by game preferences menu. For example, when player wants to play a difficult level game, MC1, MC2 and MC3 criteria values can be adjusted to the very difficult, many and many, respectively. Then, main criteria comparison matrix is found as Table 7 for this example.

Each candidate game levels based on the properties is evaluated by using the FAHP-GA hybrid model, and then, the highest valued game levels according to the fitness value of the GA are listed as a result. The FAHP produces weights for each criterion in order to evaluate chromosome fitness in the GA (Fig. 4). Also, calculated criteria weights for each generated game level are shown in Fig. 5.

The performance of the proposed FAHP-GA algorithm is compared with AHP and FAHP methods according to the same criteria based on distribution of the game level suitability points. These algorithms are investigated by statistical methods to get descriptive information. The FAHP-GA has 2.084, FAHP has 1.972, and AHP has 1.926

Table 7 Main criteria comparison matrix as TFNs

| | MC1 | MC2 | MC3 |
|-----|----------------|--------------|--------------|
| MC1 | 1, 1, 1 | 0.66, 1, 1.5 | 1.5, 2, 2.5 |
| MC2 | 0.66, 1, 1.5 | 1, 1, 1 | 0.66, 1, 1.5 |
| MC3 | 0.4, 0.5, 0.66 | 0.66, 1, 1.5 | 1, 1, 1 |

standard deviations. High standard deviation indicates that generated game levels more different. Thus, the FAHP-GA algorithm showed better performance because of having higher standard deviation. In addition, the correlation between AHP, FAHP and FAHP-GA were analyzed by the Pearson Correlation Coefficient. Calculated correlations for the AHP and the FAHP-GA is 0.638, for the FAHP and the FAHP-GA is 0.724, and for AHP and the FAHP is 0.847. The FAHP-GA is different from other algorithms as it is seen regarding to these calculated coefficients. Also, in this study, the process of searching for the best game level has to be fast and accurate, so GA accelerates the process rather than solely FAHP or AHP which is why FAHP-GA hybridized in this study.

In this study, 20 levels were generated, and these levels were tested with computational metrics such as leniency, density, negative space, balance, and reachability (Fig. 6). Results showed that generated levels are generally balanced and reachable. Leniency (in other words challenge) changes between levels because the proposed game level generator is an adaptive and dynamic level generator, so for providing challenge in the game, the negative space and density of the levels are also changing during the developed educational game (Fig. 7).

Computational metrics may not be enough, so, to examine the effectiveness and quality of the generated game levels, the game is evaluated by the players regarding the difficulty (easy, medium, difficult) for leniency (challenge), visual aesthetics (low, medium, high) for balance and density, and enjoyment (low, medium, high) for negative space and reachability [77]. The participants are secondary school students, and a total of 18 students (10 of whom is male, 8 of whom is female) played the game (Table 8). Results show that 61.38% of the generated game levels are difficult, 71.66% of the levels have high visual aesthetics, and 63.05% of the levels are high enjoyable.

The performance of the BiLSTM is compared with other prediction methods such as LSTM, ANN, Support Vector Regression (SVR), Decision Tree Regression (DTR), and Random Forest Regression (RFR). Implementation of these

Fig. 4 The fitness values of the generated game level properties according to the FAHP-GA hybrid model

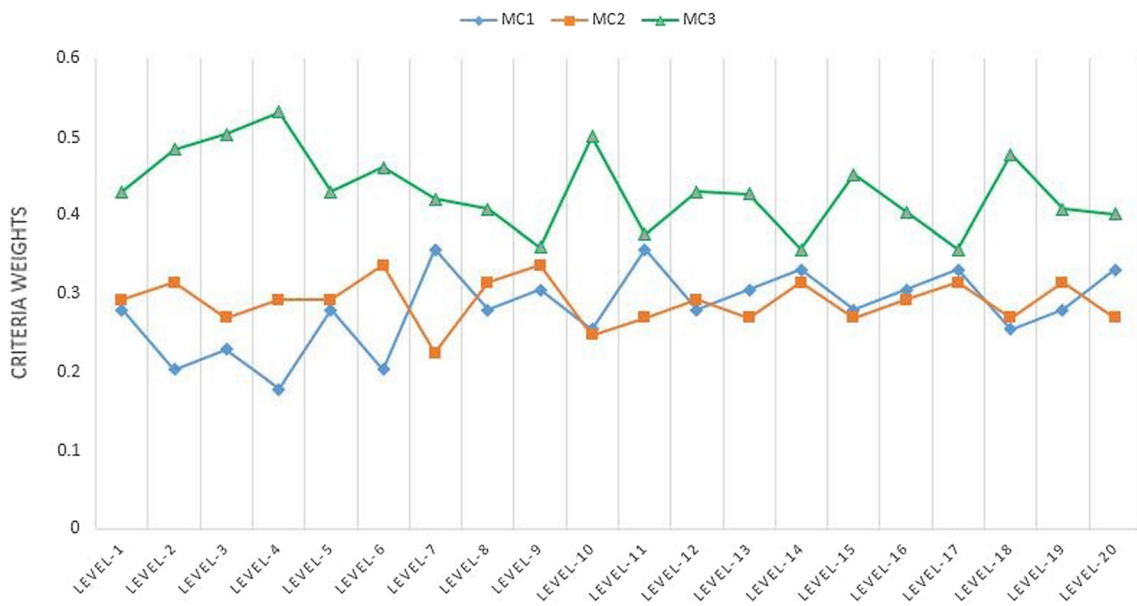
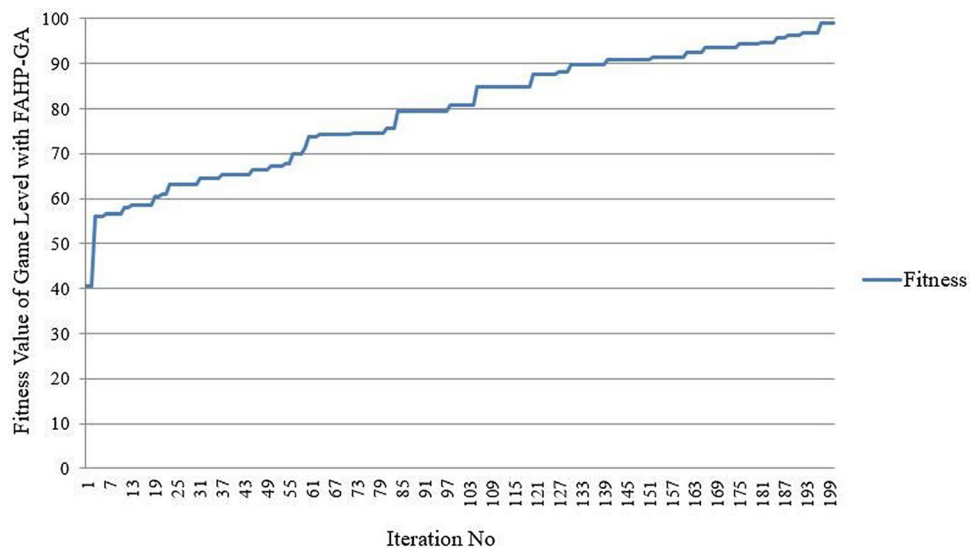


Fig. 5 The FAHP-GA criteria weights for each game level

methods is applied on the same data with optimized parameters in order to predict whether the game level is difficult or not in the corresponding time span for the game level-1 which has an actual score of 1260 (Table 9). BiLSTM gives the best result with the minimum error rate among these prediction methods.

Our proposed study makes remarkable contributions regarding the evaluation results:

- Procedural game level generation is a highly cost and time-wasting problem. Using the proposed system, adaptive game levels can be generated according to

the player preferences in a dynamic and automatic way by the help of artificial intelligence methods.

- BiLSTM makes Fuzzy AHP-GA dynamic and adaptive for game level generation which is the first study for this purpose. Also, fuzzification of the AHP reflects the more suitable player preferences.
- Deep learning, fuzzy decision-making and genetic algorithm provide an effective and interesting perspective for procedural game level generation problem. The use of hybrid systems is likely to be more popular in the future.

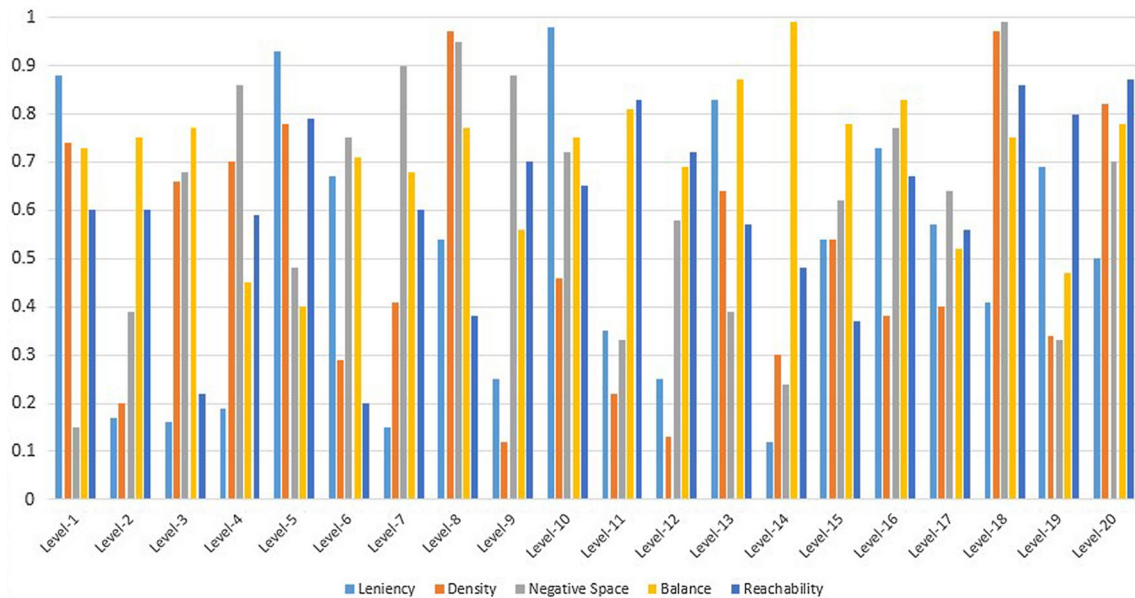
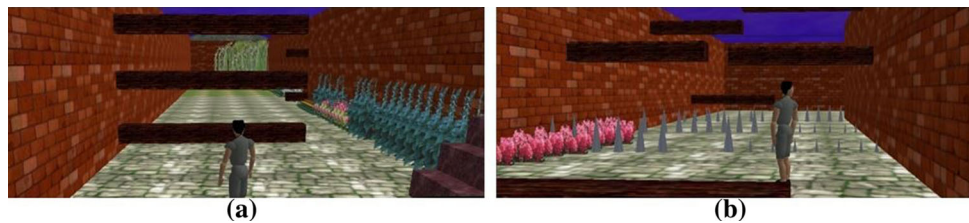


Fig. 6 Computational metrics results for generated levels

Fig. 7 Diversity of the level (obstacles) in the educational case study game **a** Low density, high negative distance **b** High density, low negative distance



There are also some limitations that should be expressed as contributive suggestions for further studies by interested researchers as follows:

- Application parameters of the proposed methods in the system may be optimized in detail for more improvements.
- This study was evaluated on developed an educational game, but further comparison and evaluation may be done with other genre of games.

By means of the proposed LSTM-based dynamic FAHP-GA hybrid model on developed game, game levels can be generated in a fast, stable, balanced and adaptive way. Furthermore, the search space of a problem is explored in a fast way to find the peak values without trapping in local minimums or maximums by using GA. Therefore, GA is combined with many of methods to produce the best, objective, reliable and robust solutions to complex problems in the literature.

6 Conclusion

Increasing demand in games leads to a rise in the need for content. Evaluating and generating procedural content game levels is a difficult and time-consuming process. Also, game levels must be dynamic and reshaped according to the players' preferences. This proposed method is the first study that combines player preferences with FAHP-GA. Moreover, criteria and sub-criteria values are changed dynamically, adaptively and automatically according to the player's preferences and the performance on the game by the help of the BiLSTM method. This is the case, even if AHP is used for calculating the preferences on game level generation with numerical values, but it does not overcome unforeseeable and uncertain human choices exactly. Hence, to handle these issues, the FAHP method is used and also is enhanced with GA method to get more reliable and best performance by using fuzzy set theory with TFNs. The weights are calculated as the synthetic values of the pairwise comparisons by the FAHP in order to be used for fitness function evaluations in the GA. Thus, game levels are generated in an automatic and intelligent way according

Table 8 Comparison of the generated game levels according to the 18 students

| Levels | Difficulty | | | Visual aesthetics | | | Enjoyment | | |
|--------|------------|--------|-----------|-------------------|--------|-------|-----------|--------|-------|
| | Easy | Medium | Difficult | Low | Medium | High | Low | Medium | High |
| 1 | 4 | 3 | 11 | 1 | 4 | 13 | 5 | 4 | 9 |
| 2 | 9 | 1 | 8 | 1 | 3 | 14 | 2 | 3 | 13 |
| 3 | 7 | 2 | 9 | 2 | 4 | 12 | 1 | 3 | 14 |
| 4 | 10 | 1 | 7 | 3 | 2 | 13 | 3 | 1 | 14 |
| 5 | 3 | 4 | 11 | 1 | 4 | 13 | 4 | 2 | 12 |
| 6 | 8 | 2 | 8 | 2 | 1 | 15 | 5 | 1 | 12 |
| 7 | 3 | 1 | 14 | 2 | 6 | 10 | 4 | 4 | 10 |
| 8 | 5 | 2 | 11 | 1 | 3 | 14 | 3 | 1 | 14 |
| 9 | 5 | 1 | 12 | 1 | 2 | 15 | 5 | 2 | 11 |
| 10 | 5 | 3 | 10 | 5 | 2 | 11 | 5 | 3 | 10 |
| 11 | 3 | 1 | 14 | 2 | 4 | 12 | 4 | 2 | 12 |
| 12 | 3 | 4 | 11 | 1 | 4 | 13 | 6 | 1 | 11 |
| 13 | 4 | 2 | 12 | 3 | 3 | 12 | 5 | 3 | 10 |
| 14 | 2 | 3 | 13 | 1 | 3 | 14 | 8 | 3 | 7 |
| 15 | 5 | 2 | 11 | 2 | 4 | 12 | 3 | 2 | 13 |
| 16 | 4 | 2 | 12 | 4 | 1 | 13 | 4 | 3 | 11 |
| 17 | 3 | 2 | 13 | 2 | 2 | 14 | 5 | 2 | 11 |
| 18 | 6 | 2 | 10 | 1 | 5 | 12 | 3 | 2 | 13 |
| 19 | 4 | 3 | 11 | 3 | 1 | 14 | 2 | 3 | 13 |
| 20 | 3 | 2 | 13 | 3 | 3 | 12 | 6 | 5 | 7 |
| Mean | 5.33 | 2.38 | 12.27 | 2.27 | 3.38 | 14.33 | 4.61 | 2.77 | 12.61 |
| (%) | 26.66 | 11.94 | 61.38 | 11.38 | 16.94 | 71.66 | 23.05 | 13.88 | 63.05 |

Table 9 Comparison of the BiLSTM results with other methods

| Prediction method | Prediction (actual value is 1260) | Error (%) |
|-------------------|-----------------------------------|-----------|
| BiLSTM | 1264.83 | 0.383 |
| LSTM | 1275.32 | 1.215 |
| ANN | 1276.04 | 1.273 |
| SVR | 1230.75 | 2.321 |
| DTR | 1233.18 | 2.128 |
| RFR | 1241.60 | 1.460 |

to the player performance and preferences in the game. In the future studies, not only game levels and also whole game can be generated completely by adding other artificial intelligent methods.

Declarations

Conflict of interest The author declares that there is no conflict of interest.

References

- King DL, Delfabbro PH, Billieux J, Potenza MN (2020) Problematic online gaming and the COVID-19 pandemic. *J Behav Addict* 9(2):184–186
- Huizenga JC, Ten Dam GTM, Voogt JM, Admiraal WF (2017) Teacher perceptions of the value of game-based learning in secondary education. *Comput Educ* 110:105–115
- Kriz WC (2020) Gaming in the time of COVID-19. *Simul Gaming* 51(4):403–410
- Laato S, Islam AN, Laine TH (2020) Did location-based games motivate players to socialize during COVID-19? *Telemat Inform* 54:101458
- Hwa SP (2018) Pedagogical change in mathematics learning: harnessing the power of digital game-based learning. *J Educ Technol Soc* 21(4):259–276
- Chen SY, Chang YM (2020) The impacts of real competition and virtual competition in digital game-based learning. *Comput Hum Behav* 104:106171
- Chen CH, Shih CC, Law V (2020) The effects of competition in digital game-based learning (DGBL): a meta-analysis. *Educ Technol Res Dev* 68(4):1855–1873
- Heng LC, Said MNHM (2020) Effects of digital game-based learning apps based on Mayer’s cognitive theory of multimedia learning in mathematics for primary school students. *Innov Teach Learn J* 4(1):65–78
- Yeh YT, Hung HT, Hsu YJ (2017) Digital game-based learning for improving students’ academic achievement, learning motivation, and willingness to communicate in an English course. In:

- IEEE 6th international congress on advanced applied informatics, pp 560–563
10. Hussein MH, Ow SH, Cheong LS, Thong MK, Ebrahim NA (2019) Effects of digital game-based learning on elementary science learning: a systematic review. *IEEE Access* 7:62465–62478
 11. Huizenga J, Admiraal W, Ten Dam G, Voogt J (2019) Mobile game-based learning in secondary education: students' immersion, game activities, team performance and learning outcomes. *Comput Hum Behav* 99:137–143
 12. Erhel S, Jamet E (2013) Digital game-based learning: impact of instructions and feedback on motivation and learning effectiveness. *Comput Educ* 67:156–167
 13. Chen CH, Law V (2016) Scaffolding individual and collaborative game-based learning in learning performance and intrinsic motivation. *Comput Hum Behav* 55:1201–1212
 14. Liao CW, Chen CH, Shih SJ (2019) The interactivity of video and collaboration for learning achievement, intrinsic motivation, cognitive load, and behavior patterns in a digital game-based learning environment. *Comput Educ* 133:43–55
 15. Behnamnia N, Kamsin A, Ismail MAB, Hayati A (2020) The effective components of creativity in digital game-based learning among young children: a case study. *Child Youth Serv Rev* 116:105227
 16. Yeh YC, Chang HL, Chen SY (2019) Mindful learning: a mediator of mastery experience during digital creativity game based learning among elementary school students. *Comput Educ* 132:63–75
 17. Hamari J, Shernoff DJ, Rowe E, Coller B, Asbell-Clarke J, Edwards T (2016) Challenging games help students learn: an empirical study on engagement, flow and immersion in game-based learning. *Comput Hum Behav* 54:170–179
 18. Chang CC, Liang C, Chou PN, Lin GY (2017) Is game-based learning better in flow experience and various types of cognitive load than non-game-based learning? Perspective from multimedia and media richness. *Comput Hum Behav* 71:218–227
 19. Denham AR (2019) Using the PCaRD digital game-based learning model of instruction in the middle school mathematics classroom: a case study. *Br J Educ Technol* 50(1):415–427
 20. Breien FS, Wasson B (2020) Narrative categorization in digital game-based learning: engagement, motivation and learning. *Br J Educ Technol*. <https://doi.org/10.1111/bjet.13004>
 21. All A, Plovie B, Castellar EPN, Van Looy J (2017) Pre-test influences on the effectiveness of digital-game based learning: a case study of a fire safety game. *Comput Educ* 114:24–37
 22. Yang JC, Chen SY (2020) An investigation of game behavior in the context of digital game-based learning: an individual difference perspective. *Comput Hum Behav* 112:106432
 23. Kiili K (2005) Digital game-based learning: towards an experiential gaming model. *Internet High Educ* 8(1):13–24
 24. Hafis M, Tolle H, Supianto AA (2019) A literature review of empirical evidence on procedural content generation in game-related implementation. *J Inf Technol Comput Sci* 4(3):308–328
 25. Tang S, Hanneghan M (2011) Game content model: an ontology for documenting serious game design. In: *IEEE developments in e-systems engineering*, pp 431–436
 26. Gagné RM, Gagné RM (1985) Conditions of learning and theory of instruction. Holt, Rinehartand Winston
 27. Zafar A, Mujtaba H, Beg MO (2020) Search-based procedural content generation for GVG-LG. *Appl Soft Comput* 86:105909
 28. Adrian DFH, Luisa SGCA (2013) An approach to level design using procedural content generation and difficulty curves. In: *IEEE conference on computational intelligence in games*, pp 1–8
 29. Summerville A, Snodgrass S, Guzdial M, Holmgård C, Hoover AK, Isaksen A, Togelius J (2018) Procedural content generation via machine learning (PCGML). *IEEE Trans Games* 10(3):257–270
 30. Barriga NA (2019) A short introduction to procedural content generation algorithms for video games. *Int J Artif Intell Tools* 28(02):1930001
 31. Togelius J, Champanand AJ, Lanzi PL, Mateas M, Paiva A, Preuss M, Stanley KO (2013) Procedural content generation: goals, challenges and actionable steps. In: Togelius J et al (ed) *Artificial and computational intelligence in games*. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Germany, pp 61–75
 32. Togelius J, Schmidhuber J (2008) An experiment in automatic game design. In: *IEEE symposium on computational intelligence and games*, pp 111–118
 33. Sorenson N, Pasquier P (2010) Towards a generic framework for automated video game level creation. In: *European conference on the applications of evolutionary computation*, pp 131–140
 34. Togelius J, Preuss M, Yannakakis GN (2010) Towards multi objective procedural map generation. In: *Proceedings of the 2010 workshop on procedural content generation in games*, pp 1–8
 35. Togelius J, Yannakakis GN, Stanley KO, Browne C (2011) Search-based procedural content generation: a taxonomy and survey. *IEEE Trans Comput Intell AI Games* 3(3):172–186
 36. Risi S, Togelius J (2020) Increasing generality in machine learning through procedural content generation. *Nat Mach Intell* 2:428–436
 37. Jain R, Isaksen A, Holmgård C, Togelius J (2016) Autoencoders for level generation, repair, and recognition. In: *Proceedings of the ICCG workshop on computational creativity and games*, pp 1–9
 38. Summerville A, Mateas M (2016) Supermario as a string: platformer level generation via lstms.
 39. Volz V, Schrum J, Liu J, Lucas SM, Smith A, Risi S (2018) Evolving mario levels in the latent space of a deep convolutional generative adversarial network. In: *Proceedings of the genetic and evolutionary computation conference*, pp 221–228
 40. Karavolos D, Liapis A, Yannakakis GN (2019) A multi-faceted surrogate model for search-based procedural content generation. *IEEE Trans Games*. <https://doi.org/10.1109/TG.2019.2931044>
 41. Smith AM, Mateas M (2011) Answer set programming for procedural content generation: a design space approach. *IEEE Trans Comput Intell AI Games* 3(3):187–200
 42. Gravina D, Khalifa A, Liapis A, Togelius J, Yannakakis GN (2019) Procedural content generation through quality diversity. In: *IEEE conference on games*, pp 1–8
 43. Hooshyar D, Yousefi M, Wang M, Lim H (2018) A data-driven procedural content generation approach for educational games. *J Comput Assist Learn* 34(6):731–739
 44. Pedersen C, Togelius J, Yannakakis GN (2010) Modeling player experience for content creation. *IEEE Trans Comput Intell AI Games* 2(1):54–67
 45. Alessi SM, Trollip SR (2001) *Multimedia for learning: methods and development*. Allyn & Bacon, Boston
 46. Li G, Kou G, Peng Y (2016) A group decision making model for integrating heterogeneous information. *IEEE Trans Syst Man Cy-S* 48:982–992
 47. Zhang H, Kou G, Peng Y (2019) Soft consensus cost models for group decision making and economic interpretations. *Eur J Oper Res* 277:964–980
 48. Kou G, Yang P, Peng Y, Xiao F, Chen Y, Alsaadi FE (2020) Evaluation of feature selection methods for text classification with small datasets using multiple criteria decision-making methods. *Appl Soft Comput* 86:105836
 49. Yu CS (2002) A GP-AHP method for solving group decision-making fuzzy AHP problems. *Comput Oper Res* 29:1969–2001
 50. Deng H (1999) Multicriteria analysis with fuzzy pairwise comparison. *Int J Approx Reas* 21:215–231

51. Zhou X, Hu Y, Deng Y, Chan FT, Ishizaka A (2018) A DEMATEL-based completion method for incomplete pairwise comparison matrix in AHP. *Ann Oper Res* 271:1045–1066
52. Beskese A, Sen T (2013) A fuzzy multi attribute approach to help measure quality of online classifieds systems. *J Mult Valued Log Soft Comput* 20:121–141
53. Kahraman C, Cebeci U, Ruan D (2004) Multi-attribute comparison of catering service companies using fuzzy AHP: the case of Turkey. *Int Prod Econ* 87:171–184
54. Buckley JJ (1985) Fuzzy hierarchical analysis. *Fuzzy Sets Syst* 17:233–247
55. Cheng AC, Chen CJ, Chen CY (2008) A fuzzy multiple criteria comparison of technology forecasting methods for predicting the new materials development. *Technol Forecast Soc Chang* 75:131–141
56. Mangla SK, Govindan K, Luthra S (2017) Prioritizing the barriers to achieve sustainable consumption and production trends in supply chains using fuzzy Analytical Hierarchy Process. *J Clean Prod* 151:509–525
57. İnce M, Yiğit T, Işık AH (2020) A novel hybrid fuzzy AHP-GA method for test sheet question selection. *Int J Inf Technol Decis Mak* 19(02):629–647
58. Chang DY (1996) Applications of the extent analysis method on fuzzy AHP. *Eur J Oper Res* 95:649–655
59. Goldberg DE, Holland JH (1988) Genetic algorithms and machine learning. *Mach Learn* 3:95–99
60. Wang YZ (2003) Using genetic algorithm methods to solve course scheduling problems. *Exp Syst Appl* 25:39–50
61. Goldberg DE (1989) Genetic algorithms in search optimization and machine learning. Addison-Wesley, Reading Menlo Park
62. Ince M, Yigit T, Isik AH (2019) A hybrid AHP-GA method for metadata based learning object evaluation. *Neural Comput Appl* 31(1):671–681
63. Yeh J, Kreng B, Lin C (2001) A consensus approach for synthesizing the elements of comparison matrix in the analytic hierarchy process. *Int J Syst Sci* 32:1353–1363
64. Pendharkar P (2003) Characterization of aggregate fuzzy membership functions using Saaty's eigen value approach. *Comput Oper Res* 30:199–212
65. Lin C, Wang W, Yu W (2008) Improving AHP for construction with an adaptive AHP approach (A3). *Autom Constr* 17:180–187
66. Terano T, Ishino Y (1996) Knowledge acquisition from questionnaire data using simulated breeding and inductive learning methods. *Exp Syst Appl* 11:507–518
67. Ding L, Yue Y, Ahmet K, Jackson M, Parkin R (2005) Global optimization of a feature-based process sequence using GA and ANN techniques. *Int J Prod Res* 43:3247–3272
68. Ince M, Isik AH, Yigit T (2016) Multi-Criteria approach to learning object selection through fuzzy AHP. *J Mult Valued Log Soft Comput* 27:47–62
69. Chan FT, Kumar N, Tiwari MK, Lau HC, Choy KL (2008) Global supplier selection: a fuzzy-AHP approach. *Int J Prod Res* 46:3825–3857
70. Deng L, Yu D (2014) Deep learning: methods and applications. *Found Trends Signal Process* 7(3–4):197–387
71. Salamon J, Bello JP (2017) Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal Process Lett* 24(3):279–283
72. Bengio Y, Simard P, Frasconi P (1994) Learning long-term dependencies with gradient descent is difficult. *IEEE Trans Neural Netw* 5(2):157–166
73. Yildirim Ö (2018) A novel wavelet sequence based on deep bidirectional LSTM network model for ECG signal classification. *Comput Biol Med* 96:189–202
74. Graves A, Schmidhuber J (2005) Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Netw* 18(5–6):602–610
75. Cui Z, Ke R, Pu Z, Wang Y (2018) Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction.
76. Kiperwasser E, Goldberg Y (2016) Simple and accurate dependency parsing using bidirectional LSTM feature representations. *Trans Assoc Comput Linguist* 4:313–327
77. Mariño JR, Reis WM, Lelis LH (2015) An empirical evaluation of evaluation metrics of procedurally generated Mario levels. In: Eleventh artificial intelligence and interactive digital entertainment conference, pp 44–50
78. Summerville A, Mariño JR, Snodgrass S, Ontañón S, Lelis LH (2017) Understanding mario: an evaluation of design metrics for platformers. In: Proceedings of the 12th international conference on the foundations of digital games, pp 1–10

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.