A Computation Model for Learning Programming and Emotional Intelligence

MEMOONA RAFIQUE1, MUHAMMAD AWAIS HASSAN1, ABDUL JALEEL2, HINA KHALID1, AND GULSHAN BANO3

1Department of Computer Science, University of Engineering and Technology Lahore, Lahore 54890, Pakistan
2Department of Computer Science, Rachna College, University of Engineering and Technology Lahore, Lahore 54890, Pakistan
3Department of Information Technology, University of Sialkot, Sialkot 51040, Pakistan
Corresponding author: Memoona Rafique (m.rafiq455@gmail.com)

ABSTRACT Introducing coding in early education improves the logical and computational thinking in kids. However, cognitive skills are not sufficient for a successful life. Understanding and managing the emotions of oneself is another crucial factor in success. The current state of the art teaching methods educates the kids about programming and emotional intelligence independently. In our opinion, it is advantageous to teach kids emotional intelligence, along with the programming concepts. However, the literature lacks the studies that make students emotionally aware while teaching them programming. This research aims to prepare students to be cognitively healthy as well as emotionally intelligent with the hypothesis that a kid’s emotional intelligence can be enhanced while teaching them cognitive skills. We proposed a computational model that teaches programming and emotional intelligence side by side to students. The model provides a curriculum and related tools. For evaluations, five hundred students of a public school were involved in different activities to find the effectiveness of the proposed model. These students were divided into five groups (A, B, C, D, and E), each having a mean age of 4, 5, 6, 7, and 8 years, respectively. Students performed multiple adaptive scenarios of path-finding that were based on self-awareness, social-awareness, sharing, and empathy emotions. Students provide the programming instructions such as sequencing, conditional statements, and looping to a robot. The children have successfully improved in both fundamental programming constructs and emotional intelligence skills. The research also successfully reduced screen time problem by providing a screen-free student interface.

INDEX TERMS Emotional intelligence, robots based learning, basic programming, and screen-free interface.

I. INTRODUCTION Emotions are an integral part of cognitive reasoning as the brain cannot organize priorities and make decisions without them [1]. Emotional intelligence makes a person capable of understanding and managing self-emotions by examining one’s own emotions and the emotions of others. It uses learned facts to guide one’s analytical thinking and actions [2]. In recent years, Emotional intelligence has become a popular topic of research [3], [4], and it is mostly accepted that individuals with high emotional intelligence have higher life satisfaction [5], [6], and they experience more positive effects [7].

However, the current schooling system puts less emphasis on improving the emotional intelligence constructs in children. Coaching of the basic constructs of emotional intelligence needs to be included in the early schooling curriculum [8], [9], but introducing it as a new subject for the kids may get opposition from academia with an argument of overburdening the kids and lack of expertise of teachers. Alternatively, this work suggests combining the learning of emotional intelligence with other educational activities, e.g., along with the teaching of cognitive skills. With the advancement of technology, academia is focusing on teaching the programming concepts to kids at early stages through the use of block technology [10]–[12]. The motivation behind this work is to take one more step forward to develop students’ emotional intelligence along with programming skills at an early schooling stage by embedding the learning of...
dimensions of emotional intelligence with the teaching of programming skills.

The importance of teaching the programming concepts to kids lies in the fact that it enhances their creativity, logical thinking, and problem-solving skills [13], [14]. Moreover, programming also makes them mathematically smart and computationally resilient [15], [16]. Such skills are very useful for any kid throughout his career regardless of the field. Our idea is to teach kids emotional intelligence along with programming as it is a common observation that kids are becoming smarter than ever, but somehow their moral values and ethical behaviors are declining.

Emotional intelligence is as important as cognitive skills [6]. It encourages the utilization of human resources and brings long-term instinctive advantages [17]–[19]. The researchers have modeled the emotional intelligence of a person by emphasizing four dimensions which are self-management, self-awareness, social awareness, and social-relationship management [20], [21]. Learning emotional intelligence through the use of technology and robotics is getting popular [22]–[24]. Children generated stronger emotional expressions while playing and learning with a robot than the children who were using screen tools like the phone, tablet, or PC [25]. Young learners consider robots as interactive partners [26], [27], and learn new things from them [28], [29]. Moreover, without a partner, kids face many problems in developing their skills [30], [31], and a robot can fill the vacuum [32].

Young learners can generate programs by sequencing the instructions for their robots [33]. The use of robotics tends to demonstrate and improve the programming concepts in youngsters, for example, sequences, logical commands, conditional statements, and iterations [13], [15], [16]. Experimental research [34] found that preschool children were able to complete necessary programming activities after completion of the robotics curriculum. Similarly, in another study [35], 5-years-old preschool children were able to design, build, and program a robot.

The research community has focused mainly on teaching emotional intelligence and programming concepts separately (Table 1), but both have not been tried together. Integrating the learning of emotional qualities (such as goal-achieving, self-awareness, social awareness, sharing, and empathy) with academic activities of young kids using robots can increase the interest of students in the studies, and their moral values and ethical behaviors can be improved.

A computational model is proposed in this research to teaching the basic concepts of programming to kids who control the movement of a physical robot by interacting using a tool for visual programming or drawing on paper and having it scanned. And the phases of the activity can be customized by the teachers. The curriculum activities are built on the dimensions of goal-achieving, self-awareness, social-awareness, empathy, and sharing and teach very basic programming constructs of sequencing, conditionals, and loops. Performing these activities by kids not only teaches them basic constructs of programming but also instinctively improves their emotional intelligence in the dimensions of goal-achieving, self-awareness, social-awareness, empathy, and sharing.

The proposed computational model consists of four modules named as 1) scenario defining, 2) scenario building, 3) scenario implementing, and 4) Feedback. A system is developed using HTML, JavaScript, Python Flask [36], and Blockly [10] to operate the proposed model. The system works in the following four steps.

1. Initially, a number of scenarios are defined as path-finding problems and stored in the scenarios database.
2. Then, a teacher builds a customized scenario from the specimens stored in the database and assigned to the students to solve the puzzle. The system provides a teacher interface (GUI) for this purpose.
3. After login, the student assesses the problem and prepares a movement pattern. The system takes a movement plan as input in two formats that guides the robot to traverse in the grid. The first format does not involve any screen interaction; it consists of blank paper and hand-drawn symbols. The second method consists of a keyboard-based system. The use of paper-based input increases the kids’ (learners) involvement and reduces screen usage time.
4. The robot scans the paper prepared by the student to get the movement pattern (or it reads the instruction via system-generated calls, in case of input from a computer). We programmed the robot with a 2D-convolution neural network to detect the movement order from the scanned symbol page prepared by the little learner. When the robot moves on the scenario’s grid, an appreciation-based-reward is generated.

The scenarios help kids to learn programming concepts of sequencing, conditional statements (to move forward, left or right to avoid a hurdle), and looping (number of steps in the same direction), while preparing a movement guide for the robot. The kids also learn and understand the emotional intelligence concepts of empathy (understand the feelings of others), sharing (motivates the kid to share his belongings with others), and help (motivates the kid to help others in needy situations).

The proposed model successfully taught basic programming concepts to kids and improved their intuitive, emotional intelligence abilities, along with moral values. The study has made the following contributions:

1) A computational model is developed to teach programming and emotional intelligence to kids (age between 4 to 8 years) side-by-side through short-term interactions.
2) The model identifies the type of mistakes those kids make while working on the multiple adaptive scenarios.
3) The effect of age on the learning process of kids is quantified. The learning process in kids appeared to be age-dependent.
The rest of the paper is organized as follows. The literature review is presented in section II. The proposed solution is introduced in section III. An experimental case study is described in section IV, and section V presents evaluation results. The discussions are given in section VI. The paper is concluded in section VII.

II. LITERATURE REVIEW

We divide the related works into two groups. The first group relates to the use of robots in developing emotional intelligence, whereas the second group contains the researches relevant to the use of robots in the teaching of programming.

A. IMPROVING EMOTIONAL INTELLIGENCE USING CURRICULUM AND ROBOTS

Emotional learning is a process of evolving social and emotional abilities of an individual to identify, understand, and accomplish the emotions. Emotional intelligence capabilities, such as expressing, understanding, and managing emotions, can be improved by specialized training to deal with everyday social matters [37].

There are several studies where emotional intelligence has been taught to kids in a variety of ways. Nathanson et al. investigated that there is a strong need for social and emotional learning to students as well as teachers. Authors also discovered that the way tutors and learners process and respond to emotions can either improve or obstruct the social and emotional development of a child [38].

Ulutas and Omeroglu hypothesized that the emotional intelligence of kids could be increased. They designed an educational program, and after passing through this program, the emotional intelligence of kids was examined. After 12 weeks of training, the emotional intelligence of the children was checked using the Sullivan Emotional Intelligence Scale, and the experimental results were compared using covariance analysis. A T-test based final result of the experimental study showed that the education program contributed significantly to increase children’s emotional intelligence [39].

These days, educational robots are a great learning tool at all levels. The use of robotics in educational institutes tends to improve the learning of students from kindergarten to the university. Interaction with robots offers many opportunities for collaborative learning, logic building, and skills achievement. To improve the learning of students via robotics, bravo et al. proposed a method for teaching and learning of non-technical subjects through drama-based activities with multiple robots. They used human-robot interaction and educational drama in combination to support learning objectives in non-technical areas. It was found that through the creation of meaningful dramatic plays with robots, students acquired new knowledge-based skills, and were able to communicate their thoughts, feelings, and desires with ease [40].

Leite et al. attempted to build emotional intelligence skills in children using multiple robotic characters. They conducted a story-telling interaction study where a single child or a group of three children interacted with the robots. The goal was to use socially assistive robots to develop children’s emotional intelligence abilities through interactive activities. The result showed that regardless of the type of interaction, multiple robots improved kids’ emotional intelligence and social skills [41].

Spaulding et al. determined that children are more emotionally expressive when engaged in an interactive educational task with a social robot as compared to a situation when children are involved in the same task with a tablet alone. They presented a novel extension to the Bayesian Knowledge Tracing model that uses affective data derived from video records of children playing an interactive story-telling game with a robot autonomously, to understand learner knowledge of reading abilities. Integrating this affective data into model training increases the quality of the learned knowledge inference models [25].

B. LEARNING PROGRAMMING USING CURRICULUM AND ROBOTS

Robotics provides a powerful platform for knowledge building [42], [43] through interaction with robot tutors. Such interactions are performed through different activities of stimulating technical skills, such as technical creativity, innovation, and problem-solving skills [43], [44]. With the advancements of teaching methods, robots are being used to enhance the knowledge of computer sciences and basic programming to kids. For this purpose, Forquesato and Borin introduced basic programming to kids by using block coding games. Their model provided help in teaching basic computational thinking and logic building to children via block programming [45].

Rogozhkina and Kushnirenko trained kids to instruct and program a virtual robot by using specific pictograms. A story was created where a robot named Fidget living in piktomir recovers coatings damaged during space shuttle launches, but it was unable to make decisions to turn left, right or to move forward. The commands of forwarding, turn right, turn left, etc. are instructed by kids who drag and drop the appropriate pictograms into the grid of the main method. As a result of this study, it was found that in addition to simple linear programs, piktomir helps to create programs using loops, conditionals, and subroutine. In this way, piktomir provides the simplest textless environment for teaching programming to kids in a lively environment [46].

Heljakka and Ihmaki introduced a playful method for kids to explore, operate, control, and execute Dash robot. The kids learned the use of the path app to code the different tracks for the Dash robot. In this experimental study, twenty children, 5 to 6 years old, and their preschool instructors participated in a three-month study of playful learning about coding for the STEM (science, technology, engineering, and mathematics). It was found that the use of robotics in curricular learning encourages kids to learn the logical thinking and skills related to coding [47].
Bers et al. presented a project where they integrated the Positive Technological Development framework with the KIBO robotic kit, the robot specially designed for young learners. KIBO was programmed by connecting wooden blocks that give different commands to the robot without using screens. In this research, 172 kids aged 3 to 5 years old participated who learned coding and computational thinking incorporated into their curriculum activities. They used “coding as a playground” strategy. It was found that teaching appropriate basic programming in childhood classrooms is possible by integrating coding into multiple curricular areas like math, science, and engineering through a project-based approach [14].

Elkin et al. used the KIBO robotic kit to design a curriculum for introducing initial engineering and programming concepts for kids. It was made possible to teach fundamental programming concepts like sequencing and repeat loops to preschool children as young as three years of age. The results of the study were outstanding as children as young as three built and program a KIBO robot successfully due to the proper framework and time to explore. Although, older children of age five performed programming tasks even better comparatively [48].

A thoroughly conducted literature survey is presented in Table 1. Some of the research works [14], [45]–[48] efficiently tutored programming concepts to kids, but they were somehow lagging in improving their emotional intelligence capabilities. The researches [38], [39], [41] work on the learning of emotional intelligence. Whereas, the works [14], [25], [40], [41], [47], [48] involve robots and the works [38], [39], [45], [46] used other methods.

It is advantageous for kids to learn emotional intelligence along with the learning of programming concepts involving robots. The purpose of current research is to improve the creativity, logical thinking, and problem-solving skills of kids and make them emotionally better in parallel.
III. PROPOSED SOLUTION

The proposed computational model consists of four significant modules to teach young learners about programming constructs and emotional intelligence concepts, side by side, through multiple adaptive scenarios. A general overview of the proposed system and its modules is given in Figure 1. The first module is ‘scenario defining,’ where we design multiple paths finding activities by involving the basic programming concepts and building blocks of emotional intelligence.

The second module is ‘scenario building’ where the teacher builds customized path-finding scenarios and assigns to the students, considering their learning perspectives.

The third module is ‘scenario implementing’ where a student implements an assigned task by using a screening tool (e.g., mobile, tablet, laptop) or without using a screening tool, according to the teacher instructions. If the student is using a screening tool, the instruction set is prepared in Blockly, and system calls are generated for the robot sprite. In the second case, a student prepares the movement scheme by writing directional arrows on the paper. The robot scans the paper and uses 2D-CNN to identify the instructions. A physical robot, COZMO, is programmed for this purpose.

The fourth module is ‘feedback’ where the robot or its sprite traverses the path by following the movement scheme prepared by the student. This module generates an appreciation based reward according to the robot’s progress towards the goal and the activities that come along the path. A detailed description of each module of the proposed computational model is given below.

A. SCENARIO DEFINING

This module provides an interface to design multiple scenario-based paths finding activities and teach different concepts of programming and emotional intelligence side-by-side. The set of defined activities is stored in a database and needs to be executed in sequence to enhance emotional intelligence and necessary programming skills. Kids solve the challenge by drawing a sequence of symbols (left, right, up, and down) on the paper or through keyboard inputs. A description of the example activities is given below.

1) INTRODUCTORY ACTIVITY (GOAL ACHIEVING)

The introductory activity is about reaching a goal stage. In this activity, a program is designed by the kid using different notations, e.g., left, right, up, and down sequences according to the scenario.

The robot follows the instructions given by the kid and moves on the interactive board (for example, Figure 2). When the robot reaches the goal successfully, it makes dancing moves and gives feedback through a “smiley face”. If it finds a hurdle in the path while following the given instructions set, it gives feedback through a “sad face”.

By performing this activity, a kid learns about the sequencing construct of programming, and goal achieving conception of emotional intelligence.

2) SELF-AWARENESS BASED ACTIVITY

This activity is designed to increase the self-awareness of kids to recognize feelings. The goal is to provide choices to kids so they can select the path and understand what makes the...
robot either happy or sad. (Figure 3). To achieve the desired goal, the kid prepares a path that avoids the sad things and includes a maximum of the glad things. The robot follows the instructions on a specially designed interactive board. The kid learns about the sequence and selection (if-else) construct of programming, and self-awareness conception of emotional intelligence.

3) EMPATHY BASED ACTIVITY
Empathy based activity enhances compassionating behavior in the kids. It helps kids to understand the problems of others. Kids prepare a sequence of movements for the robot to avoid any ambulance in the path (Figure 4) while reaching towards the destination.

4) SOCIAL-AWARENESS BASED ACTIVITY
This activity is based on social awareness. The kid becomes able to sense and manage terrible situations in the surroundings. In the storyline, a fire brigade car has to reach the fireplace to extinguish the fire and save the people and their belongings in that place (Figure 5). The kid writes the code using left, right, up, and down sequences by avoiding the hurdles in the path. Traversing multiple blocks in the same direction (a loop concept, as described in Figure 6) is used to move the car in specific directions. Thinking on a possible way to extinguish the fire to save others’ lives and belongings enhances the kid’s social-awareness. Also, the kid learns about the sequences, selection, and repetition constructs of programming.

5) SHARING BASED ACTIVITY
This sharing based activity motivates a kid to share his belongings with others. If a kid chooses a path where another poor kid is standing and shares his belongings with the needy (Figure 6), he gets a higher reward.

In this activity, we applied an additional constraint on the sequence of commands. For example, if the robot has to move
four blocks toward the left, the sequence cannot use four left symbols; instead, it has to use one left sign and loop symbols. These restrictions are meant to teach the kids about the repetition (loop) concept of programming. The if-else conditionals are learned while kids avoid hurdles. Thus, the kids learn about the sequences, selection and repetition of programming, and the sharing concept of emotional intelligence.

B. SCENARIO BUILDING

We believe that each student should be given an adaptive environment in the learning process according to his learning dimensions. These adaptive learning experiences lead to extrinsic as well as an intrinsic motivation among the learners. For a random problem formulation of activity, we developed an adaptive teacher interface module (Figure 7) where a teacher can generate a random scenario for the student according to the learning perspectives.

The teacher interface is designed using HTML and JavaScript (ECMA Script 6). A teacher can add or remove columns or rows and build the grid according to a problem. The interface provides a drag and drop mechanism to place the desired hurdles (images/sprites) into the grid according to the pedagogical viewpoint. The teacher can also set the background by selecting the image from the table of pictures. Finally, after setting the desired grid, the teacher saves this customized-built scenario. The designed scenario is saved in a text file with a specific format, as depicted in Figure 8; then, it is assigned to the registered students.

![Figure 7. Teacher interface.](image)

The format uses the symbols S, B, O, R, and G to denote the starting point, blocked area or hurdle, open path, reward, and goal, respectively. When the teacher clicks the save button, inner HTML property is used to detect the exact location of the starting point, hurdle, reward, and goal. After converting the environment file in the S-B-O-R-G format, the system reads the file row-wise. Due to the specific file format (S-B-O-R-G), the program identifies the starting point, hurdle, reward, and goal on board. The program keeps track of its motion with the help of its internal data structure. As the robot move, a virtual-navigation also performed within the data structure. When the robot reached the destination, the system checks its internal state as well. If the robot found the G’s location that matched the teacher interface’s desired goal, the final goal was achieved.

C. SCENARIO IMPLEMENTING

The teacher creates a situation through the scenario creation module and assigns the tasks to the student. When a student login into the application, the student interface shows all the scenarios assigned by the teacher (Figure 9) at the homepage. The student implements the tasks by giving commands to the robot either by using a screening tool, e.g., mobile, tablet, laptop, or without using a screening tool, as directed by the teacher. Next, we elaborate on the use of both of these techniques.

1) IMPLEMENTING USING SCREEN TOOL

This interface uses Blockly (2.20190722.1) [10], an open-source web-based visual programming editor, to design the direction blocks for instructing the robot to move in a given scenario (Figure 10).

The python-based framework Flask (1.1.1) [36] is used as a lightweight application programming interface in screen tool implementation. As a web application framework, it sends calls to the robot or its sprite to run and execute the instruction. An architectural flow of the Screen Tool interface is shown in Figure 11.

2) IMPLEMENTING WITHOUT USING SCREEN TOOL

Increasing screen time is becoming a significant health problem in the new generation. The problem is successfully reduced by providing a screen-free method. In this case,
the student performs a task by giving instructions to the robot with graphical written instructions on paper, as shown in Figure 12 (a). The robot scans these instructions as depicted in Figure 12 (b) and Figure 12 (c), and understands these symbols with the trained model of a neural network technique.

The architecture of the Convolutional Neural Network is presented in Figure 13. We used 2D Convolutional Neural Network (CNN) that is a common approach to solve patterns or image recognition and has a vital role in image processing and image classification [49]. It detects different objects, and find the depth estimation from images [50]. The 2D-CNN extract features directly from the raw input imagery data [51]. We trained the CNN model on data of more than 4000 directional images. Among them, 80% of data is taken as a training data set, and the rest of 20% data is taken as a testing data set.

The 2D-CNN was applied to the image dataset with the size of $80 \times 80$ that was converted into eight convolutional layers. These layers were separated by a max-pooling layer, followed by a fully-connected (FC) layer and an activation layer of soft-max. The output could be one of the eight classes, including right, left, up, down, loops, and counters (Figure 13). A gradient-descent optimization algorithm is used for accurate and faster results. In this research, features of images were extracted on the bases of angle. The training was completed under 500 iterations with a learning rate of 0.001. The obtained accuracy level of the whole training was 95%.

The flow chart of the execution (without using a screening tool) is presented in Figure 14. In this research, TensorFlow (a deep learning library) is used for training and inference on deep neural networks [52]. We used TensorFlow to recognize graphical notations of different signs, drawn by the kid.

**D. FEEDBACK**

The robot reads the instructions code and acts accordingly to reach the goal (Figure 15). The feedback module is designed...
to give appreciative feedback (according to performance) to the participant on completing the assigned activity. As feedback, on reaching the destination, the robot expresses success by making joyful expressions and exciting sounds. If the robot does not reach its destination and hits an obstacle on its way, it makes a sad expression and produces worrying sounds.

In failure cases, a kid debugs the faults and edits the program to give correct instructions. The system also offers a suggestion mode to the participant to understand his mistake(s). A child’s efficacy in completing a given task is used to assess the kid’s programming and emotional knowledge.

IV. EXPERIMENTAL STUDY

An experimental study was conducted with school-going young kids to evaluate our proposed computational framework. The kids who participated in experiments were aged between 4 and 8 years. The research was performed on 5 sample groups (N = 500); each consisted of 100 kids. The groups were divided according to the students’ age. The average age of groups A, B, C, D, and E was 4, 5, 6, 7, and 8 years, respectively. For evaluation, we created an experimental setup and trained the kids to participate. The procedure of the proposed experimental setup with outcome stages is shown in Figure 16. Description of the intervention components is provided in the next subsections (A, B). The implementation stage and proximal outcomes are given in section V (i.e., Results). The distal outcomes are discussed in section VI (i.e., Discussions).

A. CURRICULUM, ENVIRONMENT, AND COZMO

In the experimentation, COZMO, an artificial intelligence-powered robot (Figure 17), is used that has a screen panel and effectively displays stylized graphical facial expressions and conveys emotional states and feelings through different eye shapes. It produces different sounds to express the emotions of joy, frustration, anger, and disappointment. It can recognize people and things. The COZMO accompanied application allows the user to give instructions to play several activities or games. COZMO is complemented by a software development kit (SDK), which allows users to program the robot by providing a range of pre-programmed commands. Alterations can be made in these commands by Python programmers [53]. We reprogrammed COZMO’s SDK to make it work according to our research specifications and trained it on symbols of left, right, up, down, and loop sequencing.

We have used a layered approach that allows robotic movements and related actions to be independent of our methodology. An API could enable the linking of our system with any robot as far as it provides all basic functionalities and movements. Any robot with all the functionalities of movements, e.g., left, right, up, down, and having the camera for scanning purposes, can replace the COZMO. However, the robot needs to be connected to our system using API calls to follow the instructions. The significant achievement in experimentations is owed to the proposed methodology.
A summary of emotional activities, curriculum-focused, and robotic actions, used for kids training is given in Table 2.

**TABLE 2.** Curriculum activities, their focus, and related robotic actions used for kids training.

| Activities                  | Focus                                         | Robotic Actions                                                                 |
|-----------------------------|-----------------------------------------------|---------------------------------------------------------------------------------|
| Introductory activity       | Introduction to robotics and Goal Achieving   | Use of left, right, up, down sequencing to achieve the goal.                    |
| Self-Awareness Activity     | Recognition of feelings to increase self-awareness skills and programming knowledge of sequencing | Use of left, right, up, down sequencing to achieve the goal.                    |
| Empathy Based Activity      | Enhancing the empathy in young learners and programming knowledge of sequencing, conditional statement and debugging | Use of left, right, up, down sequencing to achieve the goal.                    |
| Social-Awareness            | Motivating towards social responsibilities and programming knowledge of sequencing, conditional statement, and debugging | Use of left, right, up, down sequencing, and loops (digits to represent same directional movements) |
| Sharing Based Activity      | Motivating towards sharing using previous programming knowledge with additional restrictions to learn loop concept | Use of left, right, up, down sequencing, and loops (digits to represent same directional movements) |

**B. TRAINING OF PRESCHOOLERS**

In this research experiment, we had twenty COZMO robots and five hundred students from a local public school. We designed this whole experiment to be completed in one week. We had to perform this whole experiment in two batches of two hundred and fifty students each. Over one week of our course, six sessions were conducted for tutorial and implementation. Each session was comprised of one hour. On the first day of the week, an introductory lecture of thirty minutes was organized for preparing the kids to participate in experimentation (Figure 18).

![FIGURE 18. Introductory lecture.](image)

Then a training session of thirty minutes was conducted to explain the functionality of the robot, the working environment, and the summary of activities (Figure 19). After completion of the first session, students were able to perform the curriculum activities by giving instructions to the robot. On the remaining five days of the week, students performed one activity per session. A ten-minute training lecture was also organized before performing the activities so that children could better understand each activity’s purpose for emotional intelligence and programming.

**C. EVALUATION**

The students performed multiple real-world scenario-based activities using robotic toy COZMO. The evaluation schemes are given below.

1) **EVALUATION OF EMOTIONAL INTELLIGENCE IMPROVEMENT**

Various tests and questionnaires are available to measure emotional intelligence. We used emotional intelligence online assessment questionnaire created by ‘The Greater Good Science Center’ [54]. Emotional intelligence measured by this questionnaire is based on the recognition of facial expressions. The test consists of 20 multiple-choice questions that evaluated individuals’ ability to deduce emotions from graphical images. Each multiple-choice consists of one image and four multiple-choice answers per question. Once participants attempted the questionnaires, their scores were recorded, but the system did not show them the right answers. Then, computational learning-based activities were performed with the COZMO robot and our interactive environment. After playing the activities, the same emotional intelligence test was re-conducted.

2) **EVALUATION OF PROGRAMMING LEARNING**

The researchers who worked on educating programming concepts to kids has evaluated the learning of programming according to the appropriate use of sequencing, loops, and conditionals [35], [48], [55]. In our experimental study, we also evaluated the younger learners’ programming skills based on these basic constructs and determined how successfully the kids guided the COZMO robot to achieve the specified goals. In the training sessions, students were educated about how to write instruction for the robot using the left, right, up, and down sequences of arrows to reach the goal. The arrow based movements order prepared by a student against an assigned activity is used for evaluating his/her learning of programming concepts. We assigned five numbers
of specifically designed activities for each construct of the
programming to a student to evaluate his/her performance in
the learning of programming. The kid prepared a sequence
of movements with notations (left, right, up, and down) to
direct the robot to reach the goal. We determined how much
a student is successful in taking the robot towards the goal
stage, and how many checkpoints are cleared by the robot. If a
kid was successful in taking the robot to goal stage in three
numbers of activates (out of five) for a specific programming
construct evaluation, he/she was considered as passed.

V. RESULTS
A. RESULTS OF EMOTIONAL INTELLIGENCE
IMPROVEMENT
The comparative results of emotional intelligence improve-
ment (before and after computational learning-based
activities) are presented in Table 3.

TABLE 3. Comparison of emotional intelligence results before and after performing the experiment.

| Groups (N=100) | Mean ages | Mean score of emotional intelligence before the experiment (out of 20) | Mean score of emotional intelligence after the experiment (out of 20) |
|----------------|-----------|---------------------------------------------------------------------|---------------------------------------------------------------------|
| A              | 4.2 years | 7.4                                                                  | 8.2                                                                  |
| B              | 5.1 years | 7.7                                                                  | 8.8                                                                  |
| C              | 6.4 years | 8.3                                                                  | 10.2                                                                 |
| D              | 7.3 years | 8.9                                                                  | 10.9                                                                 |
| E              | 8.2 years | 9.1                                                                  | 12.1                                                                 |

The results show that the emotional intelligence level
has improved in all age groups (Figure 20). However, the
emotional intelligence level of older kids improved more as
compared to the lower aged groups.

FIGURE 20. Evaluation of emotional intelligence improvement.

We also observed the performance of girls and boys of
the same ages separately, and the comparative results are
presented in Table 4. It was found that at a very young age
(4 to 8), there is no considerable difference between the
emotional intelligence of girls and boys of the same age. The
results of this comparative analysis are plotted in Figure 21.

TABLE 4. Comparison of emotional intelligence results of boys and girls.

| Groups (N=100) | Mean ages | Mean score of emotional intelligence of boys (out of 20) | Mean score of emotional intelligence of girls (out of 20) |
|----------------|-----------|----------------------------------------------------------|----------------------------------------------------------|
| A              | 4.2 years | 8.2                                                      | 8.1                                                      |
| B              | 5.1 years | 8.7                                                      | 8.7                                                      |
| C              | 6.4 years | 10.2                                                     | 10.2                                                     |
| D              | 7.3 years | 10.8                                                     | 10.9                                                     |
| E              | 8.2 years | 12.1                                                     | 12.2                                                     |

FIGURE 21. Evaluation of the emotional intelligence of boys and girls.

B. RESULTS OF PROGRAMMING LEARNING
The comparative results of programming learning of all
groups are presented in Table 5.

TABLE 5. Comparison of learning programming concepts.

| Groups (N=100) | Mean ages of students | No. of students learned sequencing | No. of students learned conditional statements | No. of students learned loop concepts |
|----------------|-----------------------|-----------------------------------|-----------------------------------------------|-------------------------------------|
| A              | 4.2 years             | 70                                | 50                                            | 10                                  |
| B              | 5.1 years             | 90                                | 70                                            | 30                                  |
| C              | 6.4 years             | 100                               | 90                                            | 40                                  |
| D              | 7.3 years             | 100                               | 90                                            | 50                                  |
| E              | 8.2 years             | 100                               | 90                                            | 50                                  |

From the results, it is observed that older kids (six and more
years old) demonstrated a higher mean level of progress on
all programming concepts such as sequencing, conditional,
and loop than younger kids (under six years).

We also observed the performance of girls and boys of
the same ages separately by giving the programming tasks
and found that at a very young age (4 to 8), there is no
considerable difference between the learning of programming
in girls and boys of the same (Figure 23). The comparative
results are presented in Table 6.

VI. DISCUSSION
The research results show that appropriate training, properly
designed curriculum and interactive learning environment are
useful to teach basic programming concepts to younger kids
(Figure 22). The proposed curriculum has also improved their
The older students were able to create conditional statements successfully, but younger kids (aged 4 and 5 years) were slow in performing these activities. The results are also consistent with the literature showing that younger children do not have enough working memory to hold many instructions simultaneously in their minds [55]. Some kids that were not successful in creating correct conditional statements to reach the goal ended up hitting the robot to a hurdle in a given situation.

Students also learned loop concepts, but their score was low on loops. These results suggest that programming concepts of loops may be a challenging concept for young learners. The types of mistakes that students made indicate that most of the learners misunderstood the syntactical rules of creating loops.

In all the results, there is a common observation that older kids (six and more years old) demonstrated a higher mean level of progress on all programming concepts than younger kids (under six years). The primary reasons are more significant working memory, higher concentration level, and extended critical ability to plan, as given in [55].

This research covers different programming concepts to help kids improve their analytical, logical, and problem-solving skills at an early age. In this study, the target is achieved by using screen tools and without any screening tool. It is proved to be an excellent opportunity to teach programming concepts without the use of screen tools. We provided an adaptive teacher interface module where the teacher can create and assign customized scenarios to students according to the pedagogical learning perspective.

Besides learning of programming concepts, our study is also useful in improving the emotional intelligence of kids. The research found that the robot, COZMO, is appropriate for children as young as age four because the kids successfully operated COZMO in the experimental study. The results show that when kids play with robots, it makes them productive and creative. Therefore, robotics can be applied to different educational programs in schools.

It is also observed that appreciation tends to increase a child’s motivation for learning. Similarly, a valid criticism or suggestion to correct mistakes also plays a role in improving the learning capabilities of kids. We found it necessary to give proper appreciation and criticism for better results.

VII. CONCLUSION
As there is a lack of successful educational study for teaching programming concepts along with improving the emotional intelligence side by side (Figure 20). The curriculum used in this study focused on essential programming skills, including sequencing, repetitions (loops), and selection (conditionals). It also emphasized on emotional intelligence abilities and moral values.

Preschool children in the study were successfully able to write a logically correct program using the arrow based instructions. It became possible because of active training sessions that developed kids’ interest. Although, when students were given more extended programs, it was harder for them to create a correct program. Kids who were failed to grasp the sequencing concepts were unable to give accurate direction symbol according to the situation and hence unable to reach the goal.
intelligence of kids, this research has proposed multiple adaptive scenarios based on a computational model for achieving the objective. The paper presented a system to teach the basic concepts of programming by controlling the movement of a physical robot. The phases of the activity using the system can be customized by teachers, and students can interact using a tool for visual programming or drawing on paper and having it scanned for the robot to follow. The study has demonstrated that it is possible to teach younger kids the fundamentals of programming and emotional intelligence without active screen time. Also, it is concluded that the use of robots improves the involvement and performance of students. The primary limitation faced is the cost of robotic equipment that can be minimized by developing robots locally instead of importing.

REFERENCES

[1] A. Damasio, Descartes’ Error: Emotion, Reason and the Human Brain, New York, NY, USA: NY Avon Books, 1994.

[2] P. Salovey and J. Mayer, “Emotional intelligence,” Imag., Cogn. Pers., vol. 9, no. 3, pp. 185–211, 1990.

[3] E. R. Kazakoff, A. Sullivan, and M. U. Bers, “Introduction to the special issue on emotional intelligence,” Personality Individual Differences, vol. 65, pp. 1–2, Jul. 2014.

[4] A. Di Fabio and M. E. Kenny, “Promoting well-being: The contribution of emotional intelligence,” Frontiers Psychol., vol. 7, p. 1182, Aug. 2016.

[5] F. Kong, J. Zhao, and X. You, “Emotional intelligence and life satisfaction in Chinese university students: The mediating role of self-esteem and social support,” Personality Individual Differences, Vol. 53, No. 8, pp. 1039–1043, Dec. 2012.

[6] D. Goleman, Emotional Intelligence: Why It Can Matter More Than IQ, Bantam, New York, NY, USA: GolemanEmotional Intel, 1995.

[7] M. Zeidner, G. Matthews, and R. D. Roberts, “The emotional intelligence, health, and well-being nexus: What have we learned and what have we missed?,” Appl. Psychol., Health Well-Being, vol. 4, no. 1, pp. 1–30, Mar. 2012.

[8] J. Mayer and C. Cobb, “Educational Policy on Emotional Intelligence: Does It Make Sense?,” Educ. Psychol. Rev., vol. 12, no. 2, pp. 163–183, 2000.

[9] M. Zeidner, R. D. Roberts, and G. Matthews, “Can emotional intelligence be schooled? A critical review,” Educ. Psychol., vol. 37, no. 4, pp. 215–231, Dec. 2002.

[10] Blockly. Accessed: Aug. 20, 2019. [Online]. Available: https://developers.google.com/blockly.

[11] J. H. Maloney, K. Pepperell, Y. Kafai, M. Restnick, and N. Rusk, “Programming by choice: Urban youth learning programming with scratch,” in Proc. 39th SIGCSE Tech. Symp. Comput. Sci. Educ., Mar. 2008, pp. 367–371.

[12] D. Wang, Y. Zhang, and S. Chen, “E-Block: A tangible programming tool with graphical blocks.,” Math. Problems Eng., vol. 2013, no. 10, 2013.

[13] G. Fessakis, E. Gouli, and E. Mavroudi, “Problem solving by 5–6 years old kindergarten children in a computer programming environment: A case study,” Comput. Educ., vol. 63, pp. 87–97, Apr. 2013.

[14] M. U. Bers, C. González-González, and M. B. Armas-Torres, “Coding as a playground: Promoting positive learning experiences in childhood classrooms,” Comput. Educ., vol. 138, pp. 130–145, Sep. 2019.

[15] E. R. Kazakoff and M. U. Bers, “The impact of computer programming on sequencing ability in early childhood,” in Proc. Amer. Educ. Res. Assoc. Conf. (AERA), New Orleans, LA, USA, 2011, pp. 1–7.

[16] E. R. Kazakoff, A. Sullivan, and M. U. Bers, “The effect of a classroom-based intensive robotics and programming workshop on sequencing ability in early childhood,” Early Child. Educ. J., vol. 41, no. 4, pp. 245–255, 2013.

[17] P. Salovey, B. T. Bedell, J. B. Detweiler, and J. D. Mayer, “Current directions in emotional intelligence research,” in Handbook of Emotions, M. Lewis and J. M. Haviland-Jones, Eds., 2nd ed. New York, NY, USA: Guilford Press, 2000, pp. 504–520. [Online]. Available: https://scholars.unh.edu/psych_facpub/408/
[43] A. Eguchi, “Educational robotics theories and practice: Tips for how to do it right,” in Robots in K-12 Education: A New Technology for Learning, Hershey, PA, USA: IGI Global, 2012, pp. 1–30.

[44] O. Mubin, C. J. Stevens, S. Shahid, A. A. Mahmud, and J.-J. Dong, “A REVIEW OF THE APPLICABILITY OF ROBOTS IN EDUCATION,” Technol. Edu. Learn., vol. 1, no. 1, p. 13, 2013.

[45] L. E. T. Foquesato and J. F. Borin, “Kids block coding Game: A game to introduce programming to kids,” in Proc. Ana. Workshop Sobre, 2018, pp. 1–8.

[46] I. Rogozhina and A. Kushnirenko, “PiktoMir: Teaching programming concepts to preschoolers with a new tutorial environment,” Procedia-Soc. Behav. Sci., vol. 28, pp. 601–605, 2011.

[47] K. Heljakka and P. Ihamäki, “Ready, steady, move! coding toys, preschoolers, and mobile playful learning,” in Int. Conf. Hum.-Comput. Interact., vol. 2019, pp. 68–79.

[48] M. Elkin, A. Sullivan, and M. U. Bers, “Programming with the KIBO robotics kit in preschool classrooms,” Comput. Schools, vol. 33, no. 3, pp. 169–186, Jul. 2016.

[49] K. He, X. Zhang, S. Ren, and J. Sun, “Spatial pyramid pooling in deep convolutional networks for visual recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 37, no. 9, pp. 1904–1916, Sep. 2015.

[50] F. Liu, C. Shen, and G. Lin, “Deep convolutional neural fields for depth estimation from a single image,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2015, pp. 5162–5170.

[51] Y. Li, H. Zhang, and Q. Shen, “Spectral–Spatial classification of hyperspectral imagery with 3D convolutional neural network,” Remote Sens., vol. 9, no. 1, p. 67, Jan. 2017.

[52] M. Abadi, “Tensorflow: A system for large-scale machine learning,” in Proc. 12th Symp. Oper. Syst. Des. Implement., vol. 2016, pp. 265–283.

[53] J. Skágeby, “Well-Behaved Robots Rarely Make History,” Coactive Technol. Partner Relations, Des. Cult., vol. 10, no. 2, pp. 187–207, 2018.

[54] (2001). Greater Good. Accessed: Jul. 15, 2019. [Online]. Available: https://greatergood.berkeley.edu/quizzes/et_quiz

[55] T. C. S. Gomes, T. P. Falcáo, and P. C. de Azevedo Restelli Tedesco, “Exploring an approach based on digital games for teaching programming concepts to young children,” Int. J. Child-Comput. Interact., vol. 16, pp. 77–84, Jun. 2018.

MEMOONA RAFIQUE received the B.S. degree in computer science from the University of Engineering and Technology at Lahore, Lahore, Pakistan. She is currently a Post-Graduate Student with the University of Engineering and Technology at Lahore. Her research interests include machine learning, emotional intelligence, and deep learning.

MUHAMMAD AWAIS HASSAN received the B.S. degree in computer science from Punjab University and the M.S. and Ph.D. degrees in computer science from the University of Engineering and Technology at Lahore, Lahore, Pakistan. He is currently working as an Assistant Professor with the Computer Science and Engineering Department, University of Engineering and Technology at Lahore. His research interests include artificial intelligence, reinforcement learning, adaptive elearning systems, and affective computing. He received the Gold Medal for his B.S. degree.

ABDUL JALEEL received the B.S. degree in computer science and engineering and the M.S. and Ph.D. degrees in computer science from UET Lahore, in 2006, 2010, and 2019, respectively. He is currently working as an Assistant Professor with the Rachna College, UET Lahore. His research interests include the development of self-managing software applications, machine learning, and autonomic computing.

HINA KHALID is currently pursuing the Ph.D. degree in computer science with the Computer Engineering Department, University of Engineering and Technology (UET) at Lahore, Lahore, Pakistan. She is also an Assistant Professor with UET Lahore. Her research interests include computer-aided education, emotional intelligence, and computational linguistic.

GULSHAN BANO received the M.S. degree in computer science from the University of Engineering and Technology at Lahore, in 2018. Since October 2018, she has been working as a Lecturer with the Information Technology Department, University of Sialkot Pakistan. Her specialization is in software engineering and machine learning. Her current research interests include machine learning, data sciences, and artificial intelligence for social media usage and gratification on youth behaviors.

**MEMOONA RAFIQUE** received the B.S. degree in computer science from the University of Engineering and Technology Lahore, Lahore, Pakistan. She is currently a Post-Graduate Student with the University of Engineering and Technology Lahore. Her research interests include machine learning, emotional intelligence, and deep learning.