Rodrigo, M. M. T. et al. (Eds.) (2021). Proceedings of the 29th International Conference on Computers in Education. Asia-Pacific Society for Computers in Education

TinkerBot: A Semi-Automated Agent for a Learning Environment Based on Tinkering

Shruti JAIN*, Ashutosh RAINA & Sridhar IYER
IDP in Educational Technology, Indian Institute of Technology Bombay, India
*jainshruti429@gmail.com

Abstract: Tinkering is an effective approach to develop Engineering Design Skills. Tinkering based robotics kits can aid in nurturing tinkering and associated design skills. Due to a limited understanding and experience, novice designers face several challenges like access to basic information via rudimentary sources like manuals leading to complications in accessing basic information. Design fixation is another challenge that limits the learners in utilizing such robotics kits to their full potential. Though a mentor can address these challenges by scaffolding and aiding reflection, a mentor supported by a semi-automated agent can have a tremendous impact on a seamless familiarization and with the robotics kits. In this paper, we present an idea of a chatbot based Semi-Automated agent as a companion for tinkering kits. After analyzing interactions between a mentor and a participant in the robotics workshops we have classified the interactions and used them to develop a scaffolding logic to automate certain routines of interactions using the chatbot. Such a chatbot can act as a scaffolding agent as well as a companion for journaling and also open the possibilities to remote or virtual mentoring.

Keywords: Tinkering, Chatbots, Engineering Problem Solving, Reflection Journal

1. Introduction

It is increasingly important that the next generation of students must acquire problem-solving and critical thinking skills to succeed in the 21st century (Afari & Khine, 2017). Tinkering is a productive approach to learn problem-solving skills. It can be defined as the process of trying an idea, evaluating and refining the solution. While solving problems with the LEGO robotics kit, students are involved in the tinkering process (Resnick & Robinson, 2017) and such kits are known to help learners in developing innovative ideas, disruptive thinking and higher-order learning skills (Afari & Khine, 2017).

Literature on nurturing tinkering with robotic companions (Raina et al., 2019); and on reflection (Patel & Dasgupta, 2019) suggest that scaffolding in the form of reflection from a mentor plays an important role in learning to solve engineering design problems and nurturing tinkering behavior. Nowadays, Chatbots are gaining popularity in a wide range of applications especially in systems that provide intelligent support to the user (Colace et al., 2018). They are considered as cheap and easy to use educational tools, which are closer to nowadays students and adequate to modern styles of learning (Georgescu, 2018). There is very little research on using Chatbots in the context of problem-solving or engineering design.

While working with Robotics Kits, novice designers face several challenges. Firstly, they lack experiential knowledge of working with the available materials or the means of gathering such knowledge as and when required. This lack inhibits their ability to tinker. This could be addressed, by introducing a mentor companion who gives *Just In Time Information* and *Just In Time Tinkering Triggers (JITI and JIT3)* (Raina et al., 2019). The second problem they face is fixation. Design fixation, which is defined as adherence to one's own design, is a common problem in engineering. Engineers tend to stick to the same idea throughout the design cycle (Kershaw et al., 2011). Through various studies, it was found that reflection can help overcome fixation. Various elements including externally prompted questions can act as *Triggers of Reflection* (Patel & Dasgupta, 2019). Reflection may also be actively facilitated through interactive journal writing (Daive, 1997).

Previous solutions like *TinkMate* (Raina et al., 2019), to build an automated tinkering companion, solved only the first problem. Through the robot, COZMO, the authors could give conversational triggers through audio and behavioral triggers through animated facial expressions, but it was limited to the triggers which were coded into it, there was no room for creating design journals or logging participant activity, it could not be used beyond a workshop setting and all the interactions with the mentor remained outside the app. These limitations can be addressed through chatbots. Chatbots can rely on programmed logic or use Artificial Intelligence (AI) and Natural Language Processing (NLP) to interpret user interactions providing quick and fast responses, seamlessly. (Colace et al., 2018). In this paper, we attempt to develop a semi-automated scaffolding agent in form of a chatbot, that can be used with engineering design kits like LEGO Mindstorms in workshop settings and help novice tinkerers learn problem-solving skills in the context of tinkering.

2. TinkerBot

To aid reflection and overcome challenges of getting stuck while working with Robotics Kits we designed a chatbot, "TinkerBot". TinkerBot is based on Slack APIs and is primarily designed to assist mentor participant interactions in a workshop setting. Mentors can send prompt, trigger and check on participant progress in a partially-automated manner in a conversational mode. TinkerBot provides the participants with an interactive logging journal and allows them to see the problem statement, required resources; and interact with mentor on the same platform. It maintains participant states by monitoring their activity, this can be leveraged to offload mentor checks, allowing the mentor to be engaged with multiple participants at the same time. So, in theory, using TinkerBot, mentor can keep a bird-eye view of the participants and get involved with them in a seamless manner. TinkerBot has three major components the Slack API infrastructure, Data Store and Scaffolding Logic.

Slack API infrastructure: Slack offers a wide range of APIs for designing powerful Chatbots. In addition to automated conversations, they can send scheduled messages and share files and documents. They are also equipped with interactive messages that can be leveraged for structured conversations. They involve forms that can be used for collecting user feedback or creating a log journal. They can store and retrieve data from databases and help in monitoring user activity.

Data Store: The data store has an organized list of challenges divided into phases based on the workshop sequence or complexity. It also has resources in the form of files, images and weblinks. These resources are mapped with tasks so that only the resources relevant for a given task are provided to the participant. Individual participant data: their current phase, challenge state, collectively referred to as participant state and logging journal are also stored in the data store.

Scaffolding Logic: The scaffolding logic governs the automated and semiautomated prompts and triggers, based on progression of the participant in a given challenge, time-based events, participant's activity on the app or commands explicitly sent by the mentor. Certain routines of

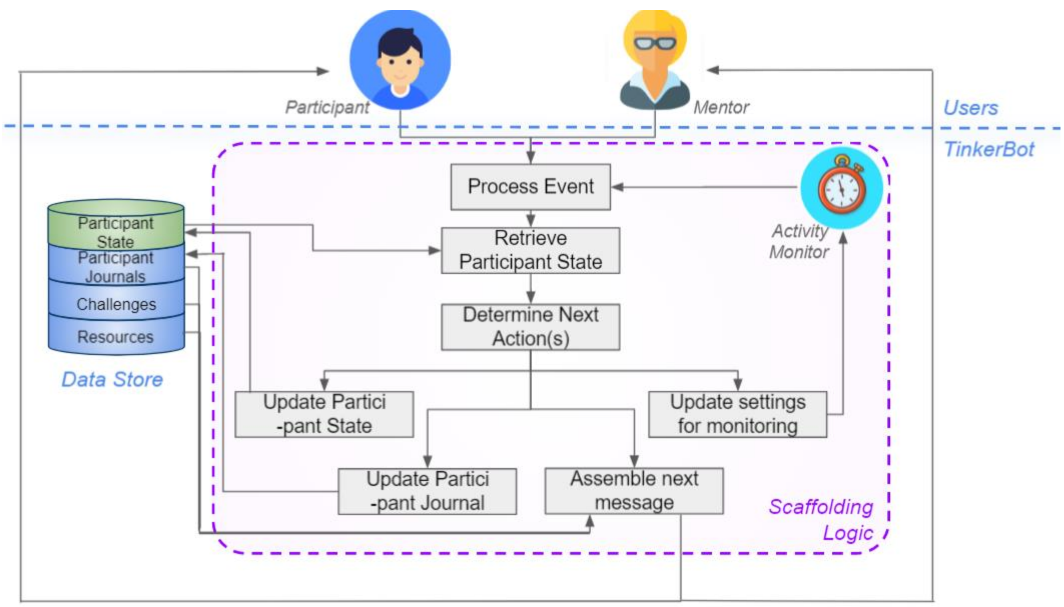

Figure 1. Scaffolding Logic and its Interaction with Users, Data Store and Monitoring App.

mentor-participant interactions identified based on the classification, presented in the next sections, were automated in form of conversations. The Scaffolding Logic is designed as a decision tree. When an event is raised, in form of a message from the users, or internally by the activity monitor, the logic takes this event and participant state and as input, computes the response as seen in figure 1 and makes updates if any. The required resources or information is then pulled from the data store to structure the output which could be the next message sent to either the mentor or the participant.

In the current state of development, we have coded several routines into TinkerBot, a few of them are shown in figure 2. TinkerBot can manage the complete flow of a challenge from introducing the problem statement to completion. When a new participant is registered, TinkerBot sends an interactive message as shown in figure 2(a), it introduces itself as a friendly companion and then talks about different components of the LEGO Mindstorm Kit. Figure 2(b) shows the routine after the generic introduction: TinkerBot waits for the participant to hit "Ready" after they have gone through the shared resources and then the bot would send the problem statement along with few detailed resources required to solve that problem. A few intuitive commands are added for both participant and mentor. Figure 2(b) shows the commands that can be used by a participant, for e. g., "help" displays the list of commands and "ask-mentor" sends a notification to the mentor so that they can join and help the participant. Figure 2(c) shows *a* form to add an entry to the participant's journal, participant can also add pictures to it by attaching files in the chat. Figure 2(d) shows another form, *using which* the mentor can select the next task(challenge) for the participant, they can select from the list or create a custom challenge.

3. Exploration of Participant Interactions

To understand the participant's interaction with mentors and other resources, video data from a tinkering workshop was analyzed. The workshop was conducted as a trial study for designing a learning environment for nurturing tinkering using the Lego Mindstorms EV3 kit. A first-year engineering undergraduate having little experience in robotics volunteered for the study. The pedagogy required the study to be divided into 3 phases- "Explore", "Solve" and "Evolve" (Raina et. al., 2018). Each phase was 3 hours long and comprised of multiple challenges to be solved by the participant, the complexity of challenges increased in every phase. The initial challenges were to measure the volume of the room, identify colors with sensors to building a four-wheel bot to be controlled by remote and eventually making the bot autonomous. The participant was free to search for resources on the internet and a mentor (researcher and experienced tinkerer) was present to guide the participant.

The mentor was to interact only on the initial two days, hence video/audio data of 3 hours for two days of participant activities and interaction with the mentor was observed. The study was set in the research lab where the participants were in the study room and observers sat in the observation room to observe both participant and mentor behavior from a one-way mirror. Participant interactions with the system were triangulated with the screen recording. An informal and open-ended interview of the participant was conducted at the end of each phase. The observations were followed by a round of discussions between the observer and mentor to confirm the intent and potential classification of data. Different stages of the participant's problem-solving process were observed. The instances of mentor-participant interaction and events initiating them were identified and further classified based on intent into different types of prompts/triggers given by the mentor. Finally, routines were derived from frequently occurring sets of conversations which could be automated using a Chatbot.

4. Findings

Through five challenges, spread out across three phases of the study, we observed patterns followed by the participant while solving the challenges. After the challenge was introduced, the participant's trajectory started with finding information about components of the kit, followed by drafting an initial design. With the design ready, the participant started constructing/programming as required. This was followed by testing and closure. These stages may not occur in sequence, for e. g., the participant may revisit resources while constructing the required structure. Most of these stages were more explicit in the case of a novice designer, as seen in “Explore” and early part of the “Solve” phase, but as they gain experience, the process of reflection and building become intertwined.

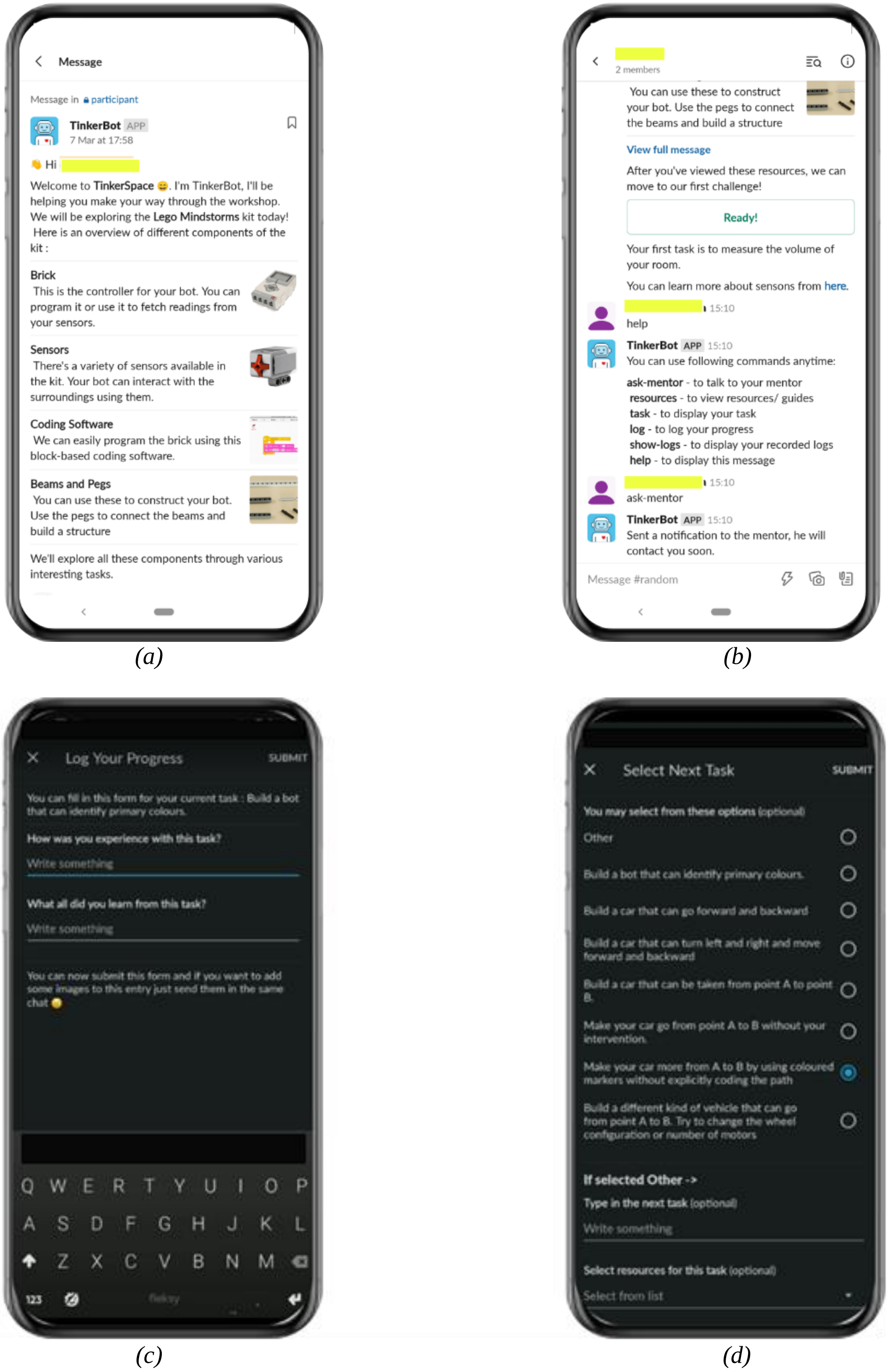

Figure 2. (a) Interactive Introductory Message received by the Participant. (b) Participant can discuss with mentor on the same platform. (c) Form to create a logging journal. (d) Mentor can select next challenge from the list or can create a custom challenge.

4.1 Events Initiating Mentor-Participant Interaction

Mentor-participant interaction was defined as a continuous set of dialogue that occurred between them. Multiple instances of such interactions were observed and classified based on the initiating events.

Participant asks a question: After the problem is introduced, if the participant is completely unaware of components/functionality they directly seek the mentor's help. Example: *P: I have to use this(brick)... I need to see how to get started… I don't know…*

Participant is Fixated: It is observed at several stages, while browsing resources, constructing/programming or testing, that the participant is unable to proceed further or is fixated(stuck), the mentor helps by posing reflective questions or giving hints to the participant.

- *Stuck while browsing resources:* Novice learners often get lost among the wide range of videos and manuals available on the internet. In literature (Raina et. al., 2019) this is referred to as a switch in context, and it takes up a lot of valuable time and causes frustration. *Example: M: Are you looking for something in particular? P: I am not sure how to …*
- *Stuck while constructing something:* As seen in literature (Patel & Dasgupta, 2019) and our data, a pre-built or demo structure serves as a good reference for the participant, but given the range of building blocks and unfamiliarity with the kit, the participant may not be able to find correct parts for their design. Mentor helps the participant to build the structure from available components. Example: *M: Do you really need it (building component) for your design?*
- *Stuck on programming software:* A participant with limited exposure to computational thinking (use of conditional statements or loops) often struggles while trying to program the Lego EV3 brick. The mentor provides alternates and debugging tips. Example: *M: Why do you think the loop is stuck? P: I am not sure. M: Try tracking the value of this variable.*
- *Stuck on a Design:* Even after realizing that their initial design is not adequate, the participant continues to adhere to it, also referred to as Design Fixation. At this point, the participant needs to reflect on their actions that lead to this current state, rectify the step where the solution went wrong or think of a new approach altogether. To trigger such reflection, the mentor poses a reflection question. Example: *M: Why do you think the bot is not turning? What is making it behave this way?*

Time-based events: Time also acts as a trigger for reflection (Patel & Dasgupta, 2019). When given a time prompt, participant was observed realizing being stuck working at the same step/design/idea for a long time. This realization was followed by reflection on their decisions to figure out a faster approach.

4.2 Prompt Types

On further analysis of mentor-participant interactions, a classification was made among different types of prompts given by the mentor.

Scaffold for Initiating a plan: Novice participants may need some scaffolding to arrive at an initial plan. These can be detailed discussions in which the mentor helps the participant to relate to their previous experiences or real-life examples which can be used to start with an initial plan for the given challenge. Example: *M: In your previous project you had 2/4 motors?*

Triggers to tinker: Such prompts were required during the ideation phase if participant got obsessed with the details of the plan or the working of sensors or bricks. In such cases, the mentor prompted the participant to start building something or to start tinkering. Example: *M: Why don't you try building the bot like you said initially?*

Direct Information: When asked by the participant or while introducing a new challenge, the mentor gives out direct information. Example: *M: These are some sensors you can use…*

Active Mentor Intervention: These were instances when the mentor was actively helping the participant. They are usually ill-structured and involve large sequences of dialogues, for instance, mentor demonstrating some brick functionality or construction of a small structure.

Triggers for reflection: Both time based and contextual prompts act as triggers for reflection (Patel & Dasgupta, 2019). Mentor was seen giving these prompts at various stages of a challenge, these helped the participant to make better design decisions. and traceback their steps and identify alternative approaches. Example: *M: Based on the height, do you think the length is correct?*

5. Study Plan

To understand the impact of a bot-based mentor for a workshop setup we initially aim at understanding the interaction of the automated system such as TinkerBot as an aid in the tinkering process by scaffolding reflection. We plan to do this by answering questions like, (i) How does interactive logging and articulation help in triggering reflection? (ii) How do prompts given by TinkerBot, help to solve engineering design problems? (iii) How does interaction vary as the participant gains expertise? Similar to the exploratory study, this study will be conducted with a novice tinkerer but the mentor will not be physically available and shall only interact via TinkerBot. Video data of the participant's activities and will be captured along with the app interaction log data. The study will be followed by interviews with the mentor and the participant to understand their experience and the challenges they faced while using TinkerBot. This study will help us understand the useability of such a solution in a workshop setting. Due to the closure of lab facilities because of COVID-19, the initial plan of lab-based studies was halted and now we plan to conduct the studies in a remote workshop mode where the participants will work remotely and the mentor will communicate via TinkerBot.

6. Conclusion

In this paper, we explored the possibility of developing an automated scaffolding agent to nurture tinkering behavior in novice designers. We attempted to understand the process of solving a challenge and did an analysis of mentor-participant interaction, we identified events that initiate a conversation between mentor and participant and classified different types of prompts given by the mentor. These states, events and prompts together helped us to develop routines of conversations that were coded into the chatbot in form of decision trees. We plan to conduct studies in future, using the proposed chatbot as the scaffolding agent and explore further possibilities. While a mentor is irreplaceable but developing a hybrid model can prove to be very efficient as a mentor's presence is limited. Chatbot's conversational nature can allow it to act as a companion which is limited with a mentor. Through TinkerBot, a single mentor can manage multiple participants, especially helping the mentor offload various tasks.

References

Afari, E., & Khine, M. S. (2017). Robotics as an educational tool: Impact of lego mindstorms. International Journal of Information and Education Technology, 7(6), 437-442.

Resnick, M., & Robinson, K. (2017). Lifelong kindergarten: Cultivating creativity through projects, passion, peers, and play. MIT press.

Raina, A., Murthy, S., & Iyer, S. (2019, December). Designing TinkMate: A Seamless Tinkering Companion for Engineering Design Kits. In *2019 IEEE Tenth International Conference on Technology for Education (T4E)* (pp. 9-14). IEEE.

Patel, A., & Dasgupta, C. (2019, July). Scaffolding structured reflective practices in engineering design problem solving. In 2019 IEEE 19th International Conference on Advanced Learning Technologies (ICALT) (Vol. 2161, pp. 287-289). IEEE.

Colace, F., De Santo, M., Lombardi, M., Pascale, F., Pietrosanto, A., & Lemma, S. (2018). Chatbot for e-learning: A case of study. *International Journal of Mechanical Engineering and Robotics Research*, *7*(5), 528-533.

Georgescu, A. A. (2018). Chatbots for education–trends, benefits and challenges. In *Conference proceedings of» eLearning and Software for Education «(eLSE)* (Vol. 2, No. 14, pp. 195-200). " Carol I" National Defence University Publishing House.

Kershaw, T., Holtta-Otto, K., & Lee, Y. S. (2011). The effect of prototyping and critical feedback on fixation in engineering design. In *Proceedings of the Annual Meeting of the Cognitive Science Society* (Vol. 33, No. 33).

Daive, L. (1997). Facilitating reflection through interactive journal writing in an online graduate course: A qualitative study. *Journal of distance education*, *1*(12), 103-126.

Raina, A., Murthy, S., & Iyer, S. (2018, July). " Help Me Build": Making as an Enabler for Problem Solving in Engineering Design. In *2018 IEEE 18th International Conference on Advanced Learning Technologies (ICALT)* (pp. 455-457). IEEE.