Using Deep Learning and Native Mobile App to Assist Autistic Students' Educational Experience

Zeina Thabet[a], Hurmat Ansari[b], Sara Albashtawi[c], Nur Siyam*[d] and Sherief Abdallah[e]

Faculty of Engineering and IT, British University in Dubai, Academic City, Dubai, U.A.E.

Keywords: Special Education, Autism, Deep Reinforcement Learning, Behaviour Intervention, Mobile Application.

Abstract: Apart from difficulties with social communication, children with autism spectrum disorder (ASD) tend to have limited interest in academic activities. The challenges faced by the educators of these students are abundant, including selecting motivating items or activities that can prompt them to complete a task. In addition to this challenge, the educators also face the issue of the lack of coordination between the teachers, therapists, and parents. This issue is imperative as significant learning opportunities are lost for lack of communication. To address these two issues, we have created a distributed system consisting of a mobile application that tracks the academic objectives and behavioural progress of the students which allows for a centralized place of information for easier coordination between educators, as well as suggesting effective motivators using a Deep Neural Network (DNN), specifically a Deep Q Network, to help autistic students regain their focus in the class. The Deep Q Network is constructed with a custom environment that takes in the state as input and then, based on the current state, calculates the best motivator to suggest. The mobile application was created with an aim of assisting school educators in tracking a student's progress. Moreover, the system includes a staff dashboard to manage users and provide visualizations depicting students' progress. This project is the first of its kind and will help educators select effective motivators in moments that the students need them as well as aid the flow of information between the stakeholders.

1 INTRODUCTION

Autism spectrum disorder (ASD) is a complex neurodevelopmental disorder that affects 1% to 2% of the population (CDC, 2020). ASD is distinguished by difficulties in communication and social interactions, repetitive and stereotyped behaviours, and restricted interests (Schuetze et al., 2017). While there is no known cure for ASD, early intervention has been shown to improve cognitive abilities, language skills, and adaptive behaviour (Dawson et al., 2012).

One difficult aspect of early intervention is students' disinterest in academic activities or assignments. Students with ASD may act in a disruptive manner to avoid academic tasks (Koegel et al., 2010). Such disruptive behaviours are regarded as major impediments to the achievement of educational goals outlined in the student's Individualized Education Program (IEP). If untreated, disruptive behaviours are likely to worsen.

There exists a myriad of interventions that aim to improve core autism symptoms and academic areas. Among those treatment with empirical support is incorporating principles of Applied Behaviour Analysis (ABA), which emphasizes environmental associations and contingencies (Chasson et al., 2007; Koegel et al., 2010; Schuetze et al., 2017). Reinforcement learning is used in ABA-based therapy techniques to increase desirable behaviour (such as eye contact) and reduce atypical behaviour (such as echoing others' phrases) by using motivational variables or rewards (Schuetze et al.,

[a] https://orcid.org/0000-0003-1216-9869
[b] https://orcid.org/0000-0003-2725-8477
[c] https://orcid.org/0000-0001-7168-6917
[d] https://orcid.org/0000-0003-1625-4892
[e] https://orcid.org/0000-0002-1213-2014

Thabet, Z., Ansari, H., Albashtawi, S., Siyam, N. and Abdallah, S.
Using Deep Learning and Native Mobile App to Assist Autistic Students' Educational Experience.
DOI: 10.5220/0011748800003470
In *Proceedings of the 15th International Conference on Computer Supported Education (CSEDU 2023) - Volume 2*, pages 96-103
ISBN: 978-989-758-641-5; ISSN: 2184-5026

2017). Motivators or rewards encourage children with ASD to stay on task, follow directions, and calm down during an outburst. Motivators can be edible items, sensory items, activities, tokens, social interactions, and choice. Edible items would include things like fruits, snacks, candy, or juice. Sensory items or activities would include activities or objects that would simulate the senses of the student, such as listening to music or playing with sand. Playing and drawing are examples of possible activities. Tokens are identified as any tangible items that are valued by the student. Social interactions include any attention given by another person or any interaction with another individual, such as a teacher or parent. Choice refers to giving the student the option to choose from two distinct items or methods.

However, for therapists to recommend an efficient motivator, they are required to use research-based adaptation strategies. This includes collecting and analysing learner's data, update intervention plans when there is an inadequate response, individualized intervention based on their clinical experience, and continuously monitor learners' progress (Siyam, 2019). In addition, different learners are motivated by different motivators. For example, not all learners desire sensor stimuli nor they are all motivated with candy (Riden et al., 2019). Thus, identifying the right motivators for learners with ASD is considered challenging (Mechling et al., 2006). To solve this problem, we develop an ASD motivator suggester powered by Deep Reinforcement Learning. In addition to this motivator selection problem (MSP), we also tackle the Intervention Coordination Problem (ICP), which addresses the challenge of sharing the information regarding the learner between all stakeholders, including teachers, therapists, and parents (Siyam, 2018).

The proposed solution consists of a distributed system containing a native mobile application, a Deep Q Network, an admin dashboard, and staff dashboard. Through the mobile application, teachers and therapists can track the IEP learning objectives for the students. They can also track students' behaviour and use the "Motivator Selection" feature that recommends motivators using the Deep Q Network. Teachers can also visualize students' progress through the staff dashboard. Moreover, administrators can create, edit, and delete users records through the admin dashboard.

This work is a continuation from previous work by Siyam and Abdallah (2021, 2022) and further improves upon it. Previous work aimed to improve the learning experience for students with ASD and tackle the MSP and ICP problems using an HTML5 webapp for its functionality and a Reinforcement Q-Learning framework for suggesting motivators.

Our implementation enhances this previous work by migrating the functionality of motivator selection, objective tracking, and behaviour tracking to a native mobile application. In addition, we also upgrade the reinforcement learning model to a deep reinforcement learning model that has the ability to learn the best motivator to suggest based on the current state that has been input by the educator. Moreover, the teacher dashboards now have visualizations that represent the progress of students. These visualizations include multiple types of charts and can be filtered per student while also showing the overall achieved objectives for all students.

To the best of our knowledge, besides our work, there exists no other integrated system for coordinating the effort of the teachers and therapists, suggesting DQN predicted motivators, and progress tracking, and visualizing for ASD affected students.

Our contributions to the previous work can be succinctly summarized in three points:

1. Native mobile application for academic objectives and behavioural tracking.
2. Deep Reinforcement learning AI model (Deep Q-Network) for motivator suggestion.
3. Dashboards with visualizations that depict the progress of students.

2 LITERATURE REVIEW

The increased prevalence of ASD diagnosis in recent years (CDC, 2020) has fuelled machine learning research with the goal of improving the learning experience of those affected (Alkashri et al., 2020). Research has primarily focused on developing academic or social skills learning applications (Roman et al., 2018), improving diagnosis efficiency (Kosmicki et al., 2015), and modelling social and behavioural aspects of ASD (Stevens et al., 2017). Our work builds on a previous study that used reinforcement learning to solve the problem of selecting a motivator. In this section, we review related work related to Deep Q Networks as well as the use of native apps in education.

Q-Learning algorithms are model-free algorithms that work by placing the agent within an environment trying to find the best possible way to solve a problem or complete a task by learning from the experience of past actions to convert it into a policy. Q-Learning algorithms aim to approximate the action-value function. The Deep Q-learning algorithms do the same thing but employ a deep neural network. The reason a deep neural network is needed is that when data is highly dimensional and exists in a large state space it is not possible to approximate an optimal action-value function by only employing a simple q-learning algorithm (Lazaridis et al., 2020).

Deep Q networks (DQN's) are the primary application of Q networks in deep learning and have achieved superior performance compared to humans in multiple applications. DQN's only take the state as input and use the neural network as a function approximator to calculate possible outputs which are the actions. The actions are chosen based on the highest calculated Q-value by the neural network. Once an action is chosen the agent performs it and then the network is updated using new weights calculated using the Bellman Equation (Fenjiro & Benbrahim, 2018).

Previous studies showed that deep neural networks (DNN's) perform best compared to other machine learning models due to their ability of inferring high-level representations without the need for extensive knowledge or preconstruction of features. Some of the applications of DNN's include pre-emptive interventions, medical diagnosis, and personalization of medication (Durstewitz et al., 2019).

Deep Q-Learning has been used in numerous applications, including recommendation systems in education. For instance, Vijayan et al. (2018) proposed a deep Q-Learning-based intelligent learning assistant for recommending courseware to learners with autism, based on the child's responses to a chatbot equipped with a visual aid. The chatbot uses multiple psychology approaches and performs an autism assessment. The intelligent learning assistant uses reinforcement learning and deep learning to recommend courseware, with scores attached to each courseware based on the child's positive or negative response, and the deep reinforcement learning algorithm maximizing the positive score.

3 METHODOLOGY

In this section, the methods used to implement this work are explained.

3.1 Requirements Gathering

Design and requirements were carried over from the previous work by Siyam and Abdallah (2021, 2022). The authors had done extensive in-field testing and research to attain the existing design which had positive reviews from the ultimate users of the system. Therefore, in this work, the website-based app was recreated as a native app considering the previous design so that its usability and functionality are preserved.

3.2 Mobile Application Design

The first stage in app development was the designing of the app. To create the prototype for the app we used a collaborative web-based app designing tool called Figma. We designed each page of the app using this tool so that we would have a solid roadmap to follow while coding the actual app. During the prototyping process we kept the design as close to the website as possible in terms of usability so that users of the website who are already accustomed to its flows would not feel disoriented when using the new app.

3.3 Creating the Deep Reinforcement Learning Model

To implement an intelligent motivator suggester with low human intervention, Artificial Intelligence was utilized. Reinforcement Learning is one of three main learning approaches in Artificial Intelligence, namely Supervised Learning, Reinforcement Learning, and Unsupervised Learning. A reinforcement learning approach is an approach used when an AI agent takes an action and then gets feedback for taking that action which is either negative or positive feedback. An Artificially Intelligent agent perceives its environment and takes actions in order to reach a specific goal. There are various types of agents in AI, and since we are using reinforcement learning, the type of agent we are using for this implementation is a Learning Agent as it learns from its past experiences in order to enhance its future performance. The learning starts when the agent perceives a state of its environment, and then the agent chooses an action to perform in order to alter that current state. After that, the agent will receive a new state of the environment and a reward for taking the previous action that is either positive or negative. If the reward is positive, the agent will learn to take that same action for the same state in the future, and if the reward is negative the agent will learn to avoid the action for the same state in the future (Russell & Norvig, 2016).

3.3.1 Q-Learning

The reinforcement learning algorithm used for the code implementation is Q-Learning, where a function Q(s,a) produces an estimate of the value of taking the action *a* in the state *s*. We refer to this value as the Q value. In the beginning, the Q value will be equal to 0 as no actions were taken before for any state. When the learning is initialized and the agent receives a state and performs an action for that state and then receives an award for that, two main things happen. First, the value of Q is estimated based on the reward received and the possible rewards that can be acquired in the future. Second, the Q(s,a) gets updated to reflect a new estimate after taking into account the reward obtained using the old estimate, this allows the model to learn from its prior experiences. The reward is an estimation of the scores the state *S* receives under the action *a*, which is 1 (Q-Learning Function), which is based on Bellman's optimality equation (see Equation 1) (Bellman, 1966).

$$Q(s,a) \leftarrow Q(s,a) + \alpha \text{ (new value estimate} - Q(s,a)) \quad (1)$$

The new value of Q(s,a) is the sum of the old Q(s,a) and an updating value. This value is calculated by multiplying the difference between the old and new values by a learning coefficient α. When $\alpha=0$, there will be no updated value, and when $\alpha=1$, the new value is taken for Q(s,a) while the old value gets ignored entirely. By adjusting the value of α, the speed of updating previous knowledge with new knowledge is determined. Q-values are stored in a Q-table as a history record so that it can be used for future states.

3.3.2 Exploration vs Exploitation

Since deep reinforcement learning models do not have datasets to learn from, they must create their own data to learn. This learning happens when the AI agent explores its environment to learn all the possibilities of actions to take in order to maximize reward. Exploitation is when the agent knows to apply its learning from exploration in order to take previously experienced actions to maximize rewards.

3.3.3 Deep Q-Learning

Although Q-learning is a powerful algorithm on its own, it does suffer some limitations due to the fact that this method is slow and is restricted to previous experiments. Despite adding exploration to it, the Q function lacks flexibility. This was the reason we decided on developing a Deep Q-Learning algorithm. A deep Q-Learning algorithm will make up for the shortfalls of a standard Q-Learning algorithm. After training, Deep Learning algorithms would be to take the best action to a state it has never seen before, which is considered better than classic Q-Learning that is only limited to a set list of states.

Deep Q-Learning utilizes neural networks as opposed to a Q-table. The state is fed into the neural network which calculates the q-value corresponding to each action. The best action is the one corresponding to the highest q value. The input layer in the neural network is the same size as the states and the output layer is the same size as the possible actions that can be taken. The inputs are inserted in the neural network which then outputs q-values that correspond to each action taken for that state. Once the state is input an episode begins and it continues until the agent reaches the terminal state which is when the reward related to the action taken is acquired. Episodes do not affect each other. However, the agent does learn from each episode which in turn makes the agent choose better actions with higher rewards in subsequent episodes.

3.3.4 Stable Baselines 3

The library we have used to implement the Deep Q Network is Stable Baselines 3 which is a library used for reinforcement learning implementation.

3.3.5 Custom Gym Environment

In order to build the Deep Q-Learning algorithm, we have to create an environment. The environment is the world which contains the observation space and where the actions happen. In our case the environment consists of observation space, action space, and reward. The observation space is the list of possible states which in our case are Trigger, Time of Day, Subject, Behaviour, and Behaviour Function. Trigger refers to the reason that caused a student to behave in a certain way. Time of Day refers to the time the student displayed a behaviour during the day. Behaviour is the conduct of the student, and Behaviour function refers to the desired target intended to be obtained by exhibiting a specific behaviour. The action space is the list of possible actions that can be taken for any state. Then the reward is calculated after each action is taken. There is a criterion for calculating the reward, which is based on the change in state. The better the change in state, the higher the reward and vice versa (Siyam & Abdallah, 2022).

When the teacher inputs a state and asks for a recommended action, the system sends a recommendation. If the teacher declines the recommendation, nothing happens and the reward is discounted, but if they accept the recommendation, they give the student the recommended motivator. After that, the system waits for feedback which is then used to calculate the reward. The criterion for

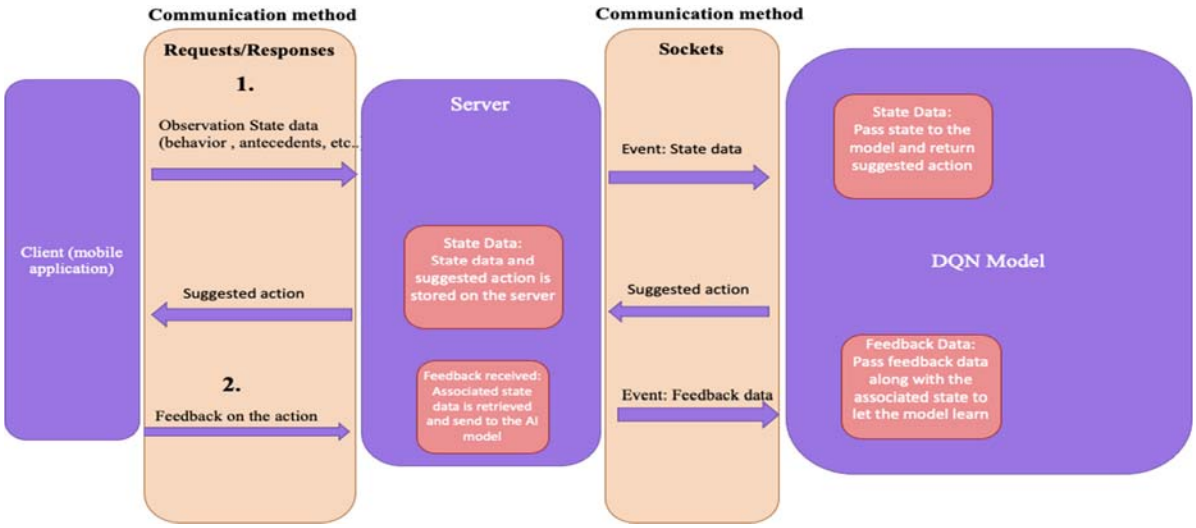

Figure 1: Communication process between the DQN model, the server, and the mobile application.

calculating the reward follows the research by Siyam and Abdallah (2022).

The definition of safe Reinforcement Learning was previously introduced in order to balance the success of the motivator suggester with the long-term avoidance of potentially harmful recommendations like sugary treats and violent movies (Siyam & Abdallah, 2022).

3.4 Creating the Server

Django framework was used to create a server that stores the data and to be able to send it over to the client (mobile application). The server follows the REST architecture where requests are sent to the server and responses are sent back to the client. Several different types of requests can be received over to the server, and they include GET, POST, and PATCH requests.

The server includes many different endpoints to serve data needed to the client and also includes endpoints that send suggested motivators from the Deep Q Networks model to the client.

3.4.1 DQN Structure Server-side

The communication method between the server and the mobile application are HTTP requests and responses, whereas the communication method between the server and the DQN model are socket connections. The whole process is a two-stage process, where the first stage starts by the user requesting a motivator suggestion by providing information about the state. This data gets sent over to the server and is then sent to the DQN model. The DQN model receives the state data as input and outputs a motivator suggestion which then gets sent back and gets saved in the database along with the provided state data. The client receives the suggested action and is displayed to the user (see Figure 1).

The second stage starts when the user sends feedback about the suggested motivator to the server, where the server then retrieves the previously saved state data using the motivator ID and all that information gets sent over to the DQN model to allow it to learn by calculating the reward according to the feedback it received. The results are then stored in the database.

3.5 Building the App

The native mobile app is built using an open-source mobile user interface framework called Flutter. Flutter provides the ability to create cross platform apps using one codebase, which is why it was chosen as the implementation framework. Once the app was created it was manually tested by group members to ensure its usability.

The mobile application consists of different screens to allow the user to view their students, view their student's objectives, and add, edit, delete, or update objective updates. As well as add behaviour updates and the addition of notes under objective updates (see Figure 2).

3.6 Building the Dashboards

Alongside the mobile application, two dashboards were created which consisted of a staff dashboard and an admin dashboard. The staff dashboard displays

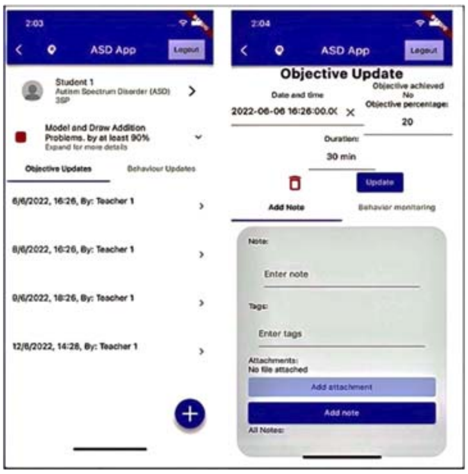

Figure 2: Mobile app screenshots.

visualizations related to the student progress to allow the staff member to view the overall progress of the student in an easy to comprehend manner. Whereas the admin dashboard allows the admin user to add, edit, update, and delete data related to the different users, staff, parents, and students. The Chart.js library was used to construct such visualizations from the student data stored in the database.

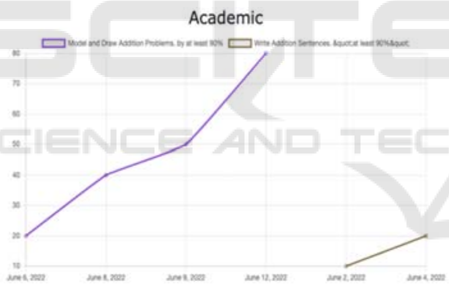

Figure 3: Teacher's dashboards.

3.6.1 Staff Dashboard

The staff dashboard included various chart that allow teachers and therapists to view the progress of different objectives across different types of objectives, percentage of objectives achieved, and the number of achieved objectives per student (see Figure 3).

The dashboard includes several graphs which include line graphs, a bar chart and a tree map.

The line graph that measures the progress of the student's objectives across time. This helps in getting an overview of the students' progress and whether they are progressing well or not.

The bar chart lists the number of completed objectives vs non completed objectives, where this allows the teacher to gauge how close they are to completing their objectives.

And the tree map illustrates the number of completed objectives per student, where this can help the teacher identify high performing students and students who are lacking behind in achieving their objectives.

3.6.2 Admin Dashboard

The admin dashboard allows full control over the data stored in the database as well as profile editing. This allows administrators to have authority to execute administrative tasks in the application.

4 EVALUATION

The evaluation of the proposed system depends on its usage by educators at schools with learners with ASD. In previous work, the mobile app has been developed and tested as an app that facilitates communication and coordination between different parties involved in the therapy and learning of students with ASD (Siyam, 2021; Siyam & Abdallah, 2022). In this paper, we describe the evaluation process for the proposed deep reinforcement algorithm. In comparison to other machine learning algorithms, reinforcement lacks an agreed-upon performance evaluation standard (Liu et al., 2020). While this is not a problem unique to RL, it is more difficult to address when compared to other machine learning algorithms that use accuracy and precision recall as performance indicators. To calculate an algorithm's precision and accuracy, an offline dataset must be divided into training and testing sets. Because this study lacks an offline dataset, we propose evaluating the effectiveness of the algorithm through statistical and qualitative methods (Stratton, 2019). Therefore, the average reward per episode will be used to gauge the improvement in performance of the proposed DQN. With every passing episode, the average reward should be increasing to show that the DQN is learning to take better actions that beget better rewards.

Siyam and Abdallah (2022) evaluated their Q-Learning model by comparing the effectiveness of the motivators when suggested by the model vs random suggestions. And concluded that the Q-Learning model was beneficial due to much higher effectiveness recorded from the users of the model. Building up on that research, we will be recording the DQN's effectiveness and comparing it to the Q-learning effectiveness, where if our DQN model illustrates higher effectiveness than the Q-learning model by Siyam and Abdallah (2022), our work will be proven to be an enhancement over the previous iteration.

5 CONCLUSION

In this work, we have developed a native app which would offer more accessibility compared to the previous web-based application. Moreover, we employed deep reinforcement learning using neural networks to improve the recommendations even when new scenarios arise.

The primary goal of this work is to aid the education of learners with ASD. This goal was achieved by the creation of a distributed system consisting of a native app that tracks the objective and behavioural progress of the students, as well as, suggesting motivators using a Deep Q-network. Moreover, the system includes dashboards for administrators and teachers at the educational institution. The admin dashboard provides the functionality for editing, adding, or deleting users, while the teacher's dashboard allows them to view their students' progress by presenting data visualizations that illustrate it. The server allows to have a way of communication between the database and the mobile app.

The native app for this system was designed using Figma, which is a prototyping graphic designing tool, and actualized using an open-source mobile user interface framework called Flutter. Flutter allows the creation of a cross platform apps using one codebase which is why it was chosen as the implementation framework. This will also allow a higher number of users to be able to use the native app rather than the web-based app.

The Deep reinforcement learning motivator suggester was implemented using the Python library Stable Baselines3 (STB3) in addition to a custom environment that we created using the library gym. A custom environment was needed and constructed to identify observation space, action space, and reward while the STB3 library was used to construct the Deep Q-network.

6 LIMITATIONS

This work has some limitations that should be considered. For instance, the proposed algorithm does not suggest motivators customized to each student. The best action suggestion made by the DQN algorithm only takes into consideration the current state with the criteria Trigger, Time of Day, Subject, Behaviour and Behaviour Function. This treats the student body as a monolithic entity and suggests motivators based on impersonalized factors. Another limitation of this work is that notifications are not sent to the stakeholders when objectives are achieved or when there is progress in behavioural goals.

7 FUTURE WORK

Further improvements can be made in the project on mobile app experience and features level as well as on the deep learning model level.

The mobile application can be improved by adding notifications feature that will allow the stakeholders to be updated in a timely order on student's objectives progress. These notifications will notify users about the most pertinent updates as opposed to every single update. Moreover, it could be improved by adding more visualizations such as a double bar chart where each double bar represents a teacher that depicts the number of completed and non-completed objectives for that teacher's students. In addition, adding a reminder system to remind different stakeholders of students who have not had much progress in their objectives, identified using several data analysis methods, could prove to be beneficial in helping students quickly identify them and direct focus to them.

On the other hand, the Deep Q Network can be improved by recommending motivators to be customized per specific learner instead of a generalized recommendation.

Finally, the system should be deployed in a school environment so that it can be used by teachers, therapists, and staff at a school. Once deployed, in-field testing can be conducted to allow the Deep Q-Network to learn from continual use so that its recommendations can become more effective. The results acquired from the testing will then be evaluated using both DQN performance evaluation, and results and reviews from the stakeholders using the system at the institution. The review, feedback, and recommendations compiled from the stakeholders will be considered to further enhance, strengthen, and streamline the system and its user experience.

REFERENCES

Alkashri, Z., Siyam, N., & Alqaryouti, O. (2020). A detailed survey of Artificial Intelligence and Software

Engineering: Emergent Issues. *2020 Fourth International Conference on Inventive Systems and Control (ICISC)*, 666–672. https://doi.org/10.1109/ICISC47916.2020.9171118

Bellman, R. (1966). Dynamic Programming. *Science*,*153*(3731), 34–37. https://doi.org/10.1126/science.153.3731.34

CDC. (2020). *Data and Statistics on Autism Spectrum Disorder | CDC*. Centers for Disease Control and Prevention. https://www.cdc.gov/ncbddd/autism/data.html

Chasson, G. S., Harris, G. E., & Neely, W. J. (2007). Cost Comparison of Early Intensive Behavioral Intervention and Special Education for Children with Autism. *Journal of Child and Family Studies*, *16*(3), 401–413. https://doi.org/10.1007/s10826-006-9094-1

Dawson, G., Jones, E. J. H., Merkle, K., Venema, K., Lowy, R., Faja, S., Kamara, D., Murias, M., Greenson, J., Winter, J., Smith, M., Rogers, S. J., & Webb, S. J. (2012). Early Behavioral Intervention Is Associated With Normalized Brain Activity in Young Children With Autism. *Journal of the American Academy of Child & Adolescent Psychiatry*, *51*(11), 1150–1159. https://doi.org/10.1016/j.jaac.2012.08.018

Durstewitz, D., Koppe, G., & Meyer-Lindenberg, A. (2019). Deep neural networks in psychiatry. *Molecular Psychiatry*, *24*(11), Article 11. https://doi.org/10.1038/s41380-019-0365-9

Fenjiro, Y., & Benbrahim, H. (2018). Deep reinforcement learning overview of the state of the art. *Journal of Automation Mobile Robotics and Intelligent Systems*, *Vol. 12, No. 3*. https://doi.org/10.14313/JAMRIS_3-2018/15

Koegel, L., Singh, A. K., Koegel, R. L., & Koegel, L. (2010). Improving Motivation for Academics in Children with Autism. *Journal of Autism and Developmental Disorders*, *40*(9), 1057–1066. https://doi.org/10.1007/s10803-010-0962-6

Kosmicki, J. A., Sochat, V., Duda, M., & Wall, D. P. (2015). Searching for a minimal set of behaviors for autism detection through feature selection-based machine learning. *Translational Psychiatry*, *5*(2), e514. https://doi.org/10.1038/tp.2015.7

Lazaridis, A., Fachantidis, A., & Vlahavas, I. (2020). Deep Reinforcement Learning: A State-of-the-Art Walkthrough. *Journal of Artificial Intelligence Research*, *69*, 1421–1471. https://doi.org/10.1613/jair.1.12412

Liu, S., See, K. C., Ngiam, K. Y., Celi, L. A., Sun, X., & Feng, M. (2020). Reinforcement Learning for Clinical Decision Support in Critical Care: Comprehensive Review. *Journal of Medical Internet Research*, *22*(7), e18477. https://doi.org/10.2196/18477

Mechling, L. C., Gast, D. L., & Cronin, B. A. (2006). The Effects of Presenting High-Preference Items, Paired With Choice, Via Computer-Based Video Programming on Task Completion of Students With Autism. *Focus on Autism and Other Developmental Disabilities*, *21*(1), 7–13. https://doi.org/10.1177/10883576060210010201

Riden, B. S., Markelz, A. M., & Randolph, K. M. (2019). Creating Positive Classroom Environments With Electronic Behavior Management Programs. *Journal of Special Education Technology*, *34*(2), 133–141. https://doi.org/10.1177/0162643418801815

Roman, J., Mehta, D. R., & Sajja, P. S. (2018). Multi-agent Simulation Model for Sequence Generation for Specially Abled Learners. In S. C. Satapathy & A. Joshi (Eds.), *Information and Communication Technology for Intelligent Systems (ICTIS 2017)—Volume 1* (pp. 575–580). Springer International Publishing.

Russell, S., & Norvig, P. (2016). Artificial Intelligence: A Modern Approach, Global Edition. *Pearson*.

Schuetze, M., Rohr, C. S., Dewey, D., McCrimmon, A., & Bray, S. (2017). Reinforcement Learning in Autism Spectrum Disorder. *Frontiers in Psychology*, *8*. https://doi.org/10.3389/fpsyg.2017.02035

Siyam, N. (2018). *Special Education Teachers' Perceptions on Using Technology for Communication Practices*. https://bspace.buid.ac.ae/handle/1234/1349

Siyam, N. (2019). Factors impacting special education teachers' acceptance and actual use of technology. *Education and Information Technologies*, *24*(3), 2035–2057. https://doi.org/10.1007/s10639-018-09859-y

Siyam, N. (2021). *Using Mobile Technology for Coordinating Educational Plans and Supporting Decision Making Through Reinforcement Learning in Inclusive Settings* [Thesis, The British University in Dubai (BUiD)]. https://bspace.buid.ac.ae/handle/1234/1879

Siyam, N., & Abdallah, S. (2021). A Pilot Study. Investigating the Use of Mobile Technology for Coordinating Educational Plans in Inclusive Settings. *Journal of Special Education Technology*, 1–14. https://doi.org/10.1177/01626434211033581

Siyam, N., & Abdallah, S. (2022). Toward automatic motivator selection for autism behavior intervention therapy. *Universal Access in the Information Society*. https://doi.org/10.1007/s10209-022-00914-7

Stevens, E., Atchison, A., Stevens, L., Hong, E., Granpeesheh, D., Dixon, D., & Linstead, E. (2017). A Cluster Analysis of Challenging Behaviors in Autism Spectrum Disorder. 2017 *16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 661–666. https://doi.org/10.1109/ICMLA.2017.00-85

Stratton, S. J. (2019). Quasi-Experimental Design (Pre-Test and Post-Test Studies) in Prehospital and Disaster Research. Prehospital and Disaster Medicine. Cambridge University Press, vol. 34(6), pp. 573–574

Vijayan, A., Janmasree, S., Keerthana, C., & Baby Syla (2018). A Framework for Intelligent Learning Assistant Platform Based on Cognitive Computing for Children with Autism Spectrum Disorder. *2018 International CET Conference on Control, Communication, and Computing (IC4)*, 361–365. https://doi.org/10.1109/CETIC4.2018.8530940.