AN ANXIETY AND STRESS REDUCING PLATFORM BASED ON MINIGAMES AND EMOTIONAL RELEASE USING MACHINE LEARNING AND BIG DATA ANALYSIS

Selina Gong[1], John Morris[2] and Yu Sun[2]

[1]University High School
4771 Campus Dr, Irvine, CA 92612
[2]California State Polytechnic University, Pomona, CA, 91768

ABSTRACT

Today's students are faced with stress and anxiety as a result of school or work life and have added pressure from social media and technology. Stress is heavily related to many symptoms of depression such as irritability or difficulty with concentration as well as symptoms of anxiety like restlessness or feeling tired. Some of these students are able to find a healthy outlet for stress, however other students may not be able to. We have created a program where students will be able to destress and explore their emotions with the help of suggestions from our system based on previously explored thoughts. Our program uses machine learning to help students get the most effective stress relief by suggesting different mental health exercises to try based on input given by the user and provides emotional comfort based on the user's preferences.

KEYWORDS

relaxing, destress, game, journal.

1. INTRODUCTION

The two biggest factors of teen depression are stress and technology [1]. Over 4 million children aged 3-17 in the US have either anxiety or depression, and this number has only gone up in recent times. Many teenagers spend over 7 hours on screen (from phones to computers and tvs) not including the amount of time they spend on schoolwork. In these hours, students frequent entertainment sites such as social media and video watching sites, which may have negative impacts on the student's self-esteem [2]. The Covid-19 pandemic also placed pressure on teenagers due to the sudden spike in fear and isolation. Not only that, but when school resumed, many students found themselves facing a higher level of academic stress from having to adjust to online learning as well as the inefficiency of teaching through online meetings. Having to self quarantine for a long period of time also ruins many teenagers' routines, specifically their sleep schedules, which could lead to an imbalance in hormones. All of these factors put teenagers at risk for depression, which is characterized by a loss of energy, poor school performance, self-harm, and even suicidal thoughts [3]. The issue of mental health has stigma surrounding it, making it difficult for teenagers to discuss the topic with people in their immediate lives. Instead, it is easier for students to go online, where no one who truly knows who they are and vent about whatever happened in their lives that day.

There are self-care applications on the app store, as well as relaxing games, but our program aims to combine them both and help the user relax and de-stress while also helping the user explore emotions. Some applications that aim to help the user self-reflect as a means of relieving stress rely entirely on the user being willing to continue putting in the effort of thinking about their own emotions. However, this assumes that the user has enough energy to open themselves up and discuss the most vulnerable parts of their emotions. And this is also assuming that the user will remember to use the application in the first place. Not only that, but some of these applications are simply online journals that ask the user questions and store their answers [4]. This kind of application is marketed toward helping depressed or stressed users, but does not provide any incentive for the user to continue since they are so reliant on user input and the users it is marketed toward do not have the energy to continue a long journey of self-reflection without instant gratitude. A second problem would be that the user input is not used to help the user. Users are allowed to customize their own journal, but that's about it. This results in users having to put in their own effort to analyze their own responses and figure out the best way to ease their stress. Once again, considering that the target audience is students with high levels of stress and depression, not all users will have the time to do so.

In this paper, we research if our game is useful in helping students destress. We believe that this method may not be as successful with helping the user understand themselves better in comparison to other applications that are more focused on guiding the user through exploring their own thoughts, however our method aims to help the user either relax or get happier through a few fun minigames. Our goal is to optimize relaxation for users through the collection of data that they input into the game, specifically the journal minigame. Some features included in our method are a journal that records user input and multiple minigames that do not have a time limit pressure or a large pressure for failure. We also utilize the tendency for younger adults to relate with inanimate objects in our method with our number of Non-Playable Characters (NPCs) [5]. Since the recent Covid-19 pandemic required most citizens to self-quarantine, there were increased feelings of loneliness throughout the world. Having a digital "friend" gives the users a character to relate to and through this, they may be able to grow or come to a new understanding of the self. The NPCs in our game each have their own personalities and elaborate backstories that users can read about. Finishing certain tasks may also trigger a message from the NPCS. Other methods may not have an incentive for success and may not give the user a sense of attachment. We also wanted to help users relax through lowering breathing and heart rates [6]. Our game uses low BPM music in every scene to achieve this.

We have developed a program that analyzes whether a user will have had an improvement in mood or not depending on the user's frequency in playing a game, the amount of time spent playing that game, and the user's score. We show the usefulness of our approach with an experiment that logs many aspects of user actions.

The rest of the paper will follow this outline: Section 2 describes challenges we encountered during the process of creating and carrying out this experiment, Section 3 presents related works, Section 4 explains the methodology we used and the solution we came up with, Section 5 evaluates the experiment and provides extra details, and Section 6 gives concluding remarks.

2. CHALLENGES

In order to build the tracking system, a few challenges have been identified as follows.

2.1. Making the Game Interesting

Since we were making a game, we had to consider how we would gain the user's attention and interest for an extended period of time. We didn't want to make this a method that users would use a few times and forget about. We wanted to give the users a sense of accomplishment that is relatively easy to get, but still requires a bit of effort to obtain. With this, we came up with the scoring system and friendship points. We created NPCs that each have their own unique personality and backstory so users will want to become better acquainted with them [7]. However, we did not want to make it so that the user would just click through the stories without much gratitude. We decided that these backstories would be locked behind a friendship point cost. Friendship points would be easily obtainable through daily use of our game. In the end, we laid out a plan for how to retain user interest in our game.

2.2. Scoping an Accurate Data-Driven Model

We didn't want to have to continually ask our users for their emotions before and after their experience with our game, because we would not always be there to ask and users were not guaranteed to always remember to complete the journal that logs emotion at the correct times. In order to be able to determine whether the user was able to relieve stress by playing our game without having to ask them to log it, we wanted to create a data-driven model that was relatively accurate. We had to test multiple machine learning algorithms each with their own parameters and specialized abilities. We wanted to determine which machine learning algorithms would prove to be more successful in determining whether the user experienced stress relief [8]. Since there are many machine learning algorithms available for us to use, we had to research which ones would fit best for our problem. Not only that, but most algorithms we chose had multiple parameters. In order to get the best accuracy, we needed to change each parameter and test them individually to determine which factors together would return the best accuracy while still being reasonable.

2.3. Figuring Out how to Make Matched Objects

One minigame, the matching game, contained a function that required the game to open a different scene, the journal scene, and then return to the minigame scene. The problem was that whenever users returned to the matching scene, everything got reset since everything that users could interact with was a clone, so all of the NPCs left the screen and came back in and any previously matched items reappeared. However, we needed to make it so that they only did not reset if the user was returning from the journal scene. For this, we needed to ensure that the game kept track of which scene the user was returning from and we needed the scene to hold references to which NPCs were in the scene previously [9]. In the end, we created a system that kept track of which NPCs were previously in the scene, the position they were in, and which items were already matched.

3. RELATED WORK

Identification and categorization of digital game experiences: a qualitative study integrating theoretical insights and player perspectives [10] aims to analyze the effects of different kinds of games and the experience its players get from them. This work summarized that some of the main reasons why someone would play a game are: having free time, wanting to relax, or wanting to socialize. Some of the experiences the participants had related to sound, exploration, and relaxation. These findings were very broad in terms of finding the experiences of different gamers when playing, and were not as specifically focused on games and their effects on relaxation.

Stress Relieving Video Games: Creating a Game for the Purpose of Stress Relief and Analyzing Its Effectiveness [11] successfully analyzes the relaxing effect of two different kinds of games and de stressing exercises on a variety of people. One of the games was a researcher made game and the other was Tetris. The researcher found that all three methods of relaxation mostly resulted in a decrease in heart rate, save for an outlier, however the participants themselves were split on whether they felt distressed after the researcher made a game and the breathing exercises. The participants who played Tetris did feel distressed, though. This researcher did something very similar to what we did, however they included different kinds of games with a very few participants [12]. Having only six participants may have led to skewed results.

Implementation of serious game techniques in raising the social awareness of the depression disease [9] analyzes how effective a serious depression awareness game was in informing users of the risks and effects of depression. The study had a large participant group, 89 participants, that showed that the majority of the participants were uninformed about depression before the game and most participants felt much more informed about depression after the game.

4. SOLUTION

Flory is a video game meant to provide relaxation and happiness through the use of three different minigames and a small low-poly world to explore, this is also an adaptive and interactive gaming platform for depression and stress relief using machine learning.

There are many Non Play Characters (NPCs) in this game, each with unique backstories and personalities [11]. The first minigame is a journal minigame meant to help the user explore or release any pent up emotions.

The second minigame is a click and drag matching game. The user will help NPCs find items they want to buy from a store's shelves.

The third minigame is a popping game. Flower seeds will pop up on screen and when tapped, the flower will bloom and fall to the ground. The goal is to pop enough flowers to fill the field.

The game uses machine learning to recognize which NPCs the user likes to spend friendship points on and will prioritize use of those NPCs. Through these three minigames, the game will learn to process player emotions using machine learning basics with data collection and data training. Eventually, the game will be able to provide recommendations for which minigame to play based on the emotion input in the journal minigame. The ultimate goal is to use machine learning as the backend for emotional & mental support through a gaming experience.
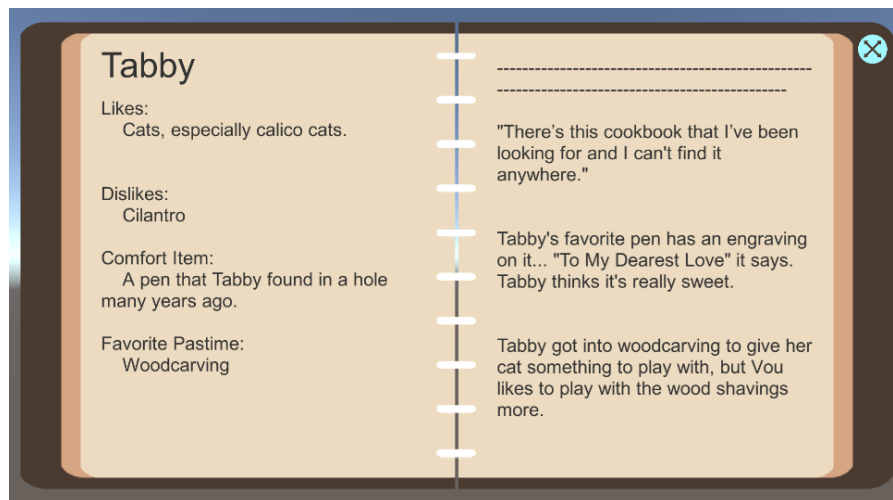
Figure 1. Mini game 1

In minigame 1, the journal minigame, users are asked a set of questions related to their day and emotions that they feel. This will help the users vent any unhealthy emotions in a way that doesn't harm anyone around them and will also prompt users to reflect upon themselves and grow as a person. The user's answers are recorded and the user will be allowed to go back to previous days to recall what they were feeling before. The user is allowed to complete one entry per day and each complete entry awards the user with 10 friendship points. (See Appendix A for code segment.)
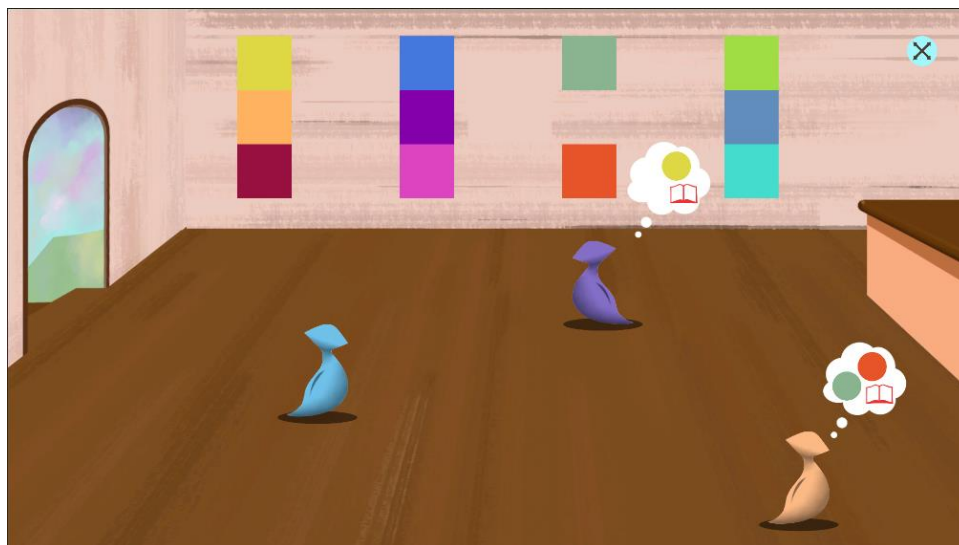

Figure 2. Mini game 2 (1)

The game also includes a matching minigame featuring the NPCs and different items that they want to buy. Each NPC has their own personal journal where the user can learn the NPCs likes and dislikes, comfort items, and favorite pastime(s) [12]. The second page of the NPCs journal is full of entries that can be unlocked for 15 friendship points that contain more insight into the NPCs personality, but more importantly, these entries will tell the user what items that specific NPC wants to buy. (See Appendix B for code segment.)
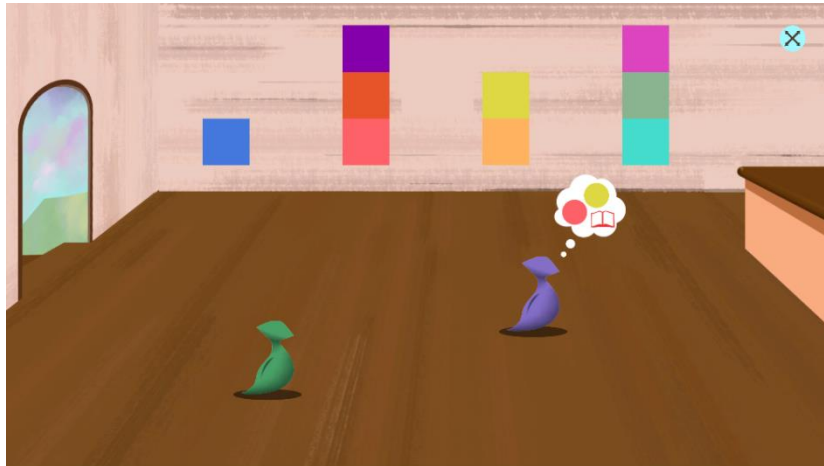
Figure 3. Mini game 2(2)

Since these entries need to be unlocked, it is entirely possible that an NPC featured in the matching game will want to buy an item that the user does not know. In the minigame, items will appear on shelves. Three random NPCs will come into the store and ask for two items in a thought bubble. One item will be a random general item, and one will be an NPC unlockable item. If the item is locked, then the user will not see it. When an item is dragged off of the shelf and near the correct NPC, the item and its icon in the NPCs thought bubble will disappear. Once both items have been matched, the NPC will leave the room.
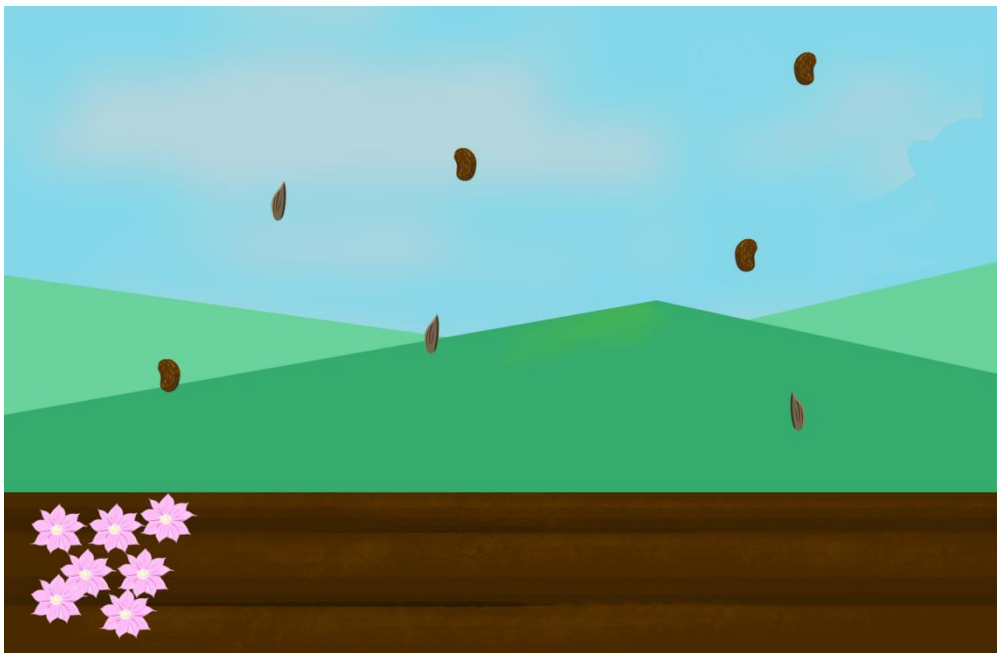

Figure 4. A screenshot of mini game 3

In the flower popping minigame, seeds will randomly spawn on screen. There will be a maximum of 10 seeds on screen at any time and a minimum of 5. Clicking on a seed will cause that seed to bloom into a flower and fall into the flower field at the bottom of the screen. When the user pops enough flowers to fill the flower field, the user wins and is given the option to continue if they would like. Each 20 flowers popped will give the user 10 friendship points.

5. EXPERIMENTS

Five high school students participated in this study. Participants ranged from 14 years old to 18 years old. All participants said they felt stressed from school work and extracurriculars. Three participants' main source of stress came from college applications.

5.1. Experiment 1: Does our game reduce student stress?

To evaluate the effectiveness of the game in improving the user's emotion, we asked 5 students to play the game, logging their emotions before and after using the journal minigame. In other words, they would be playing the game as intended and all data would be collected as they were playing. Students were encouraged to ignore time limits and the score was not displayed in order for students to not be pressured by the game itself. They were also not watched while playing the game. In this way, users were able to fully immerse themselves in the game and relax.

There are four pieces of data that are logged and the fifth piece will be determined from the other pieces of data. In the journal game, users are able to log their current emotion twice per day. Participants were instructed to fill out the journal once at the beginning of their experience and once at the end. This is to determine whether the participant had an improvement in mood and/or stress relief.

In the matching minigame, data is automatically logged at the end of the game. The timestamp at which the user completed the game, the duration of the game and the score as well as whether the user played a full game are logged. The score of the game as well as whether the game was completed is also logged. A perfect score is 12 points.

Table 1 below is the data for an example participant. The emotion before they started playing the game was sad. They played the matching game twice with a combined time of 62 seconds and a combined score of 16. They did not complete the second game, though.

Table 1. Example of Data Collected In-Game

Scene Name	Time	Log Type	Extra
Journal	1629320087.88526,	Emotion Before: sad	
Matching Game	1629320110.57972,	Duration: 39	Score: 12, True
Matching Game	1629320127.3878,	Duration: 23	Score: 4, False
Journal	1629320155.85095,	Emotion After: neutral	

Table 2 below shows the data of all eight students that participated in this study. Of the eight students, five of them had an improvement in mood after playing the game. The average time of the five participants who had an improved mood was 28.9 seconds per round with an average score of 10.9 points per round.

Table 2. User-study data from participants

Participant #	Frequency	Total Duration (s)	Total Score	Mood Improved?
1	1	15	12	No
2	3	63	34	Yes
3	1	42	5	No
4	5	134	56	Yes
5	2	62	16	Yes
6	2	84	24	Yes
7	2	31	24	No
8	4	120	45	Yes

The first participant did not have a mood improvement had a perfect score of 12, which is higher than the average of the participants that had mood improvement, but only had a time of 15 seconds for a single round which is 51.9% of the average time it took for participants with an improvement in mood. After asking for feedback, the first participant was rushed and tried to finish the game as fast as possible, leading to a slight increase in stress. The third participant took 42 seconds to play one round, 145% of the average time it took for participants with an improvement in mood, and got a score of 5. The third participant wrote that they did not understand the rules very well and spent most of their time figuring out how to play. The seventh participant was very similar to the first participant. The seventh participant played two rounds and spent an average of 15.5 seconds per round and a perfect score on both rounds. As feedback, they said the game was too easy and it wasn't fun to play.

Of the participants that did have an improvement in mood, some of the common responses to the question of why they had an improvement in mood included the music and the calming repetition of clicking and dragging in addition to the fact that the matching pairs were very obvious to spot.

5.2. Experiment 2: How can we determine if the game is successful in stress relief without having to ask the user?

We used different machine learning models to determine the accuracy with which we can predict whether a user's stress will be relieved. We used a Linear Kernel Support Vector Machine, a Random Forest Classifier, a Logistic Regression Model, and a Gradient Boosting classifier. These models were trained using 500 pieces of dummy data that were randomly generated. Each model was tested five times and the average of the five scores was used to determine which model is most accurate in its predictions. The Random Forest Classifier had a max depth of 10 and a random state of 0, the Logistic Regression model had a random state of 0, and the Gradient Boosting Classifier had 100 n estimators, a learning rate of 1.0, a max depth of 5, and a random state of 0.
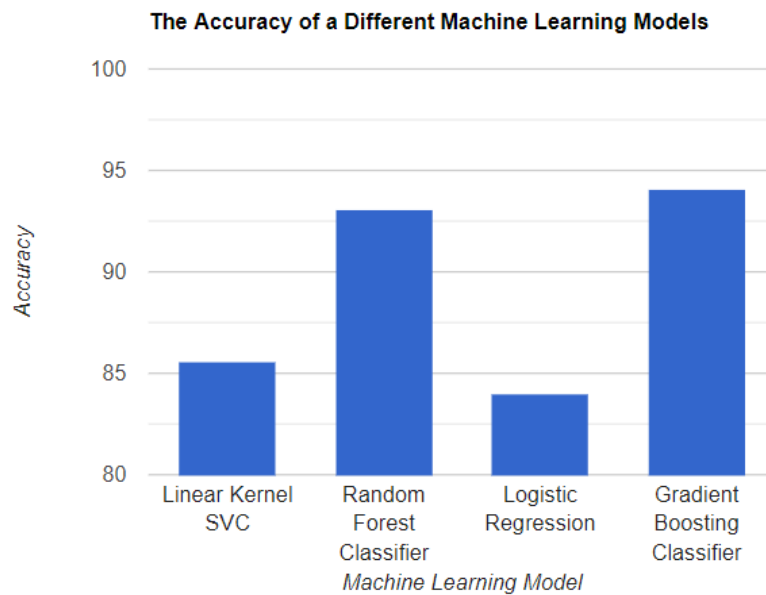
Figure 7. The accuracy of a different machine learning models

Of the four models we tested, the Gradient Boosting Classifier had the highest average accuracy at 94.8% accuracy followed by the Random Forest Classifier with an accuracy of 93.6%. The Linear Kernal SVC had an accuracy of 85.6% while the Logistic Regression model had an accuracy of 84%. This shows that our problem is not a logistic relationship. We wanted to get the highest accuracy possible, so we decided to test a few parameters in the Gradient Boosting Classifier.

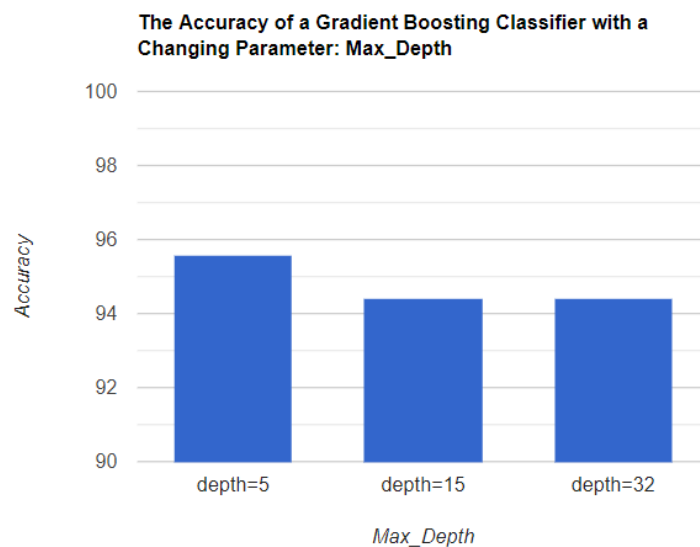## 5.3. Experiment 3: How do the parameters of the Gradient Boosting Classifier affect our accuracy?

Figure 8. The accuracy of a gradient boosting classifier (1)

The Gradient Boosting Classifier had a parameter called max_depth that adjusted the amount of leaves on the learning tree. We tested the Gradient Boosting Classifier using the depths of 5, 15, 32. The Gradient Boosting Classifier had an accuracy of 95.6% accuracy at a max depth of 5 and 94.4% at max depth of 15 and 32. This is contradictory to what is expected, because having more leaves on the learning tree usually leads to having a higher accuracy. This unexpected result could be because our pattern is relatively simple and having a higher number of leaves on the learning tree was unnecessary.
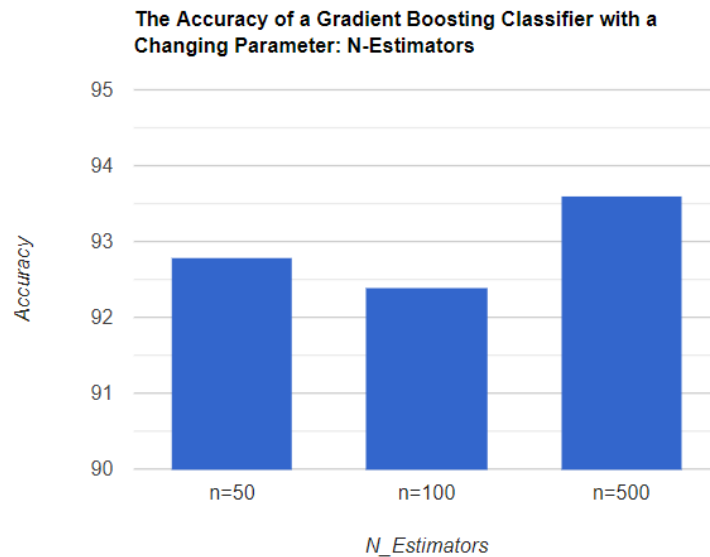

Figure 9. The accuracy of a gradient boosting classifier (2)

When adjusting the n estimators, the Gradient Boosting Classifier had an accuracy of 92.8% at n=50, 92.4% at n=100, and 93.6% at n=500. While adjusting both of these parameters had an impact on accuracy, the differences in accuracy were very small. We did, however, change our algorithm to have 500 n estimators instead of the original 100.

5.4. Experiment 4: How does the size of a data set affect the accuracy of the Gradient Boosting Classifier?

Changing the data set size also affected the accuracy of our experiment. When only using 50 pieces of data, we had an accuracy of 84%. We had an accuracy of 82% with 100 pieces of data, 92.8% with 250 pieces of data, and 96.4% with 500 pieces of data. The average of 100 pieces of data was lower than expected. With the exception of the experiment with 100 pieces of data, the general trend is that the larger size of the data set, the more accurate the Gradient Boosting Classifier will be.
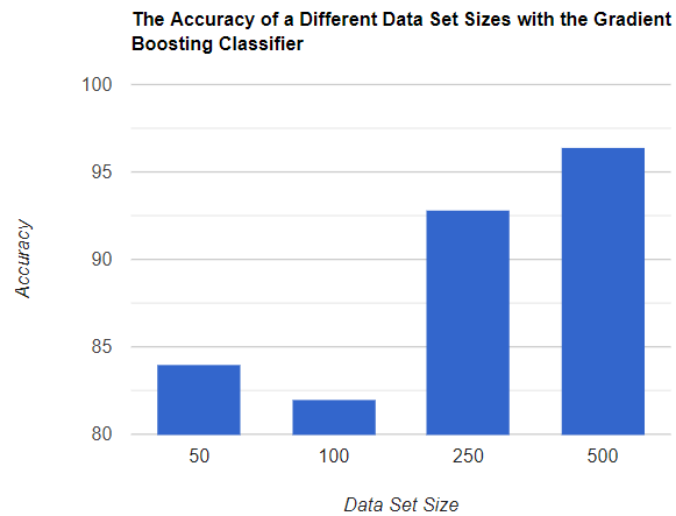
The Accuracy of a Different Data Set Sizes with the Gradient Boosting Classifier

Figure 10. The accuracy of a different data set sizes

5.5. Experiment 5: How does the cross evaluation test ratio affect our accuracy?

We had an accuracy of 95.2% with a cross evaluation test size of 0.1, which is the test size we used for our previous experiments. Using a cross evaluation test size of 0.5, we got an accuracy of 90.16%. We got an accuracy of 100% with the test size of 1. Of the tested values, having a test size of 1 is not ideal, because a machine learning algorithm having a 100% accuracy is highly unlikely.
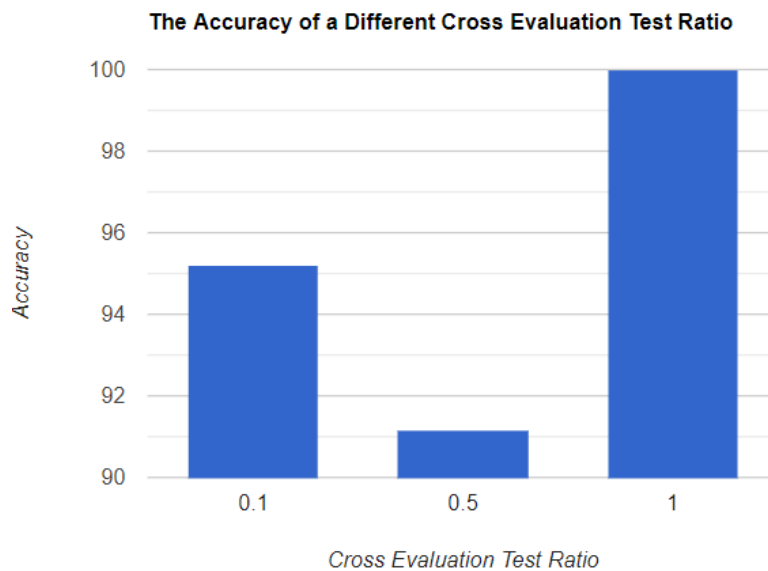
The Accuracy of a Different Cross Evaluation Test Ratio

Figure 11. The accuracy of a different cross evaluation test ratio

5.6. Analysis of Results

Our game seems to relieve the user's stress if the user takes the time to play the game through without rushing and if the user understands the rules of the game. The second condition is a gaming standard since many users would not be able to enjoy any game if they did not understand the rules even if they were not penalized for it. In the end, we were able to develop an algorithm that was able to relatively accurately predict whether a user would have benefitted from the game without having to ask them directly. We decided to use the Gradient Booster Classifier as opposed to any other algorithm because it gave us the highest accuracy at default settings. In order to increase the accuracy higher, we adjusted the max depth, the number of n estimators, data set size, and the cross evaluation test ratio. We found that we had the highest reasonable accuracy with a max depth of 5, 500 n estimators, 500 pieces of data, and a cross evaluation ratio of 0.1.

6. CONCLUSIONS

We created a game for the purpose of relaxation and de-stressing using different minigames including: a journal minigame where users are encouraged to reflect upon their emotions, a matching minigame where users will be able to connect with NPCs, and a flower popping minigame where users will repeat a clicking motion for relaxation. All three of these minigames provide an incentive for success so the user will continue to play the game. In order to find out which minigame is most effective in destressing its users, we asked many different people to rate the games [13].

Our application may not be as aesthetically pleasing as some other games, which may have a large factor in determining the ability for users to fully enjoy the game. The idea itself seems to be very entertaining, which means that as the game is developed further, it is possible for it to become even more successful in relieving stress. However, the most successful machine learning algorithm, the Linear Kernal SVC, also has an accuracy only a bit better than random chance. There is also still nowhere that this prediction is put into play.

In the future, we plan on placing more weight on certain factors in the machine learning algorithm so we will be able to determine whether a person's mood improved without having to ask them for the second time before they close the application. Once this has been improved, we may be able to implement a function that encourages users to continue playing if their mood has not been improved.

REFERENCES

[1]   Jaycox, Lisa H., et al. "Impact of teen depression on academic, social, and physical functioning." Pediatrics 124.4 (2009): e596-e605.

[2]   Cast, Alicia D., and Peter J. Burke. "A theory of self-esteem." Social forces 80.3 (2002): 1041-1068.

[3]   Prince, Martin, et al. "No health without mental health." The lancet 370.9590 (2007): 859-877.

[4]   Hodkinson, Paul. "Interactive online journals and individualization." New Media & Society 9.4 (2007): 625- 650.

[5]   Mondesire, Sean, and R. Paul Wiegand. "Evolving a non-playable character team with layered learning." 2011 IEEE Symposium on Computational Intelligence in Multicriteria Decision-Making (MDCM). IEEE, 2011.

[6]   Brunyé, Tad T., et al. "Learning to relax: Evaluating four brief interventions for overcoming the negative emotions accompanying math anxiety." Learning and Individual Differences 27 (2013): 1-7.

[7]   Perou R, Bitsko RH, Blumberg SJ, Pastor P, Ghandour RM, Gfroerer JC, Hedden SL, Crosby AE, Visser SN, Schieve LA, Parks SE, Hall JE, Brody D, Simile CM, Thompson WW, Baio J, Avenevoli S, Kogan MD, Huang LN. Mental health surveillance among children — United States, 2005—2011.

MMWR 2013;62(Suppl; May 16, 2013)

[8]   Son, Changwon et al. "Effects of COVID-19 on College Students' Mental Health in the United States: Interview Survey Study." Journal of medical Internet research vol. 22,9 e21279. 3 Sep. 2020, doi:10.2196/21279

[9]   Ma ł gorzata, Plechawska-W ó jcik, and Jakub Grzesiak. "IMPLEMENTATION OF SERIOUS GAME TECHNIQUES IN RAISING THE SOCIAL AWARENESS OF THE DEPRESSION DISEASE."

[10] Poels, Karolien, Yvonne De Kort, and Wijnand IJsselsteijn. "Identification and categorization of digital game experiences: a qualitative study integrating theoretical insights and player perspectives." Westminster Papers in Communication and Culture 9.1 (2012): 107-129.

[11] Mercer, Nicole. "Stress Relieving Video Games: Creating a Game for the Purpose of Stress Relief and Analyzing Its Effectiveness." (2015).

[12] Nacke, Lennart Erik, et al. "Biofeedback game design: using direct and indirect physiological control to enhance game interaction." Proceedings of the SIGCHI conference on human factors in computing systems. 2011.

[13] Konicek, Petr. "Destressing." Rockburst. Butterworth-Heinemann, 2018. 453-471.

Appendix A

Code segment of Minigame 1

```csharp
6   public class Journal : ScriptableObject
7   {
8       public List<PageEntry> pages = new List<PageEntry>();
9
10      public List<PageEntry> pagePool = new List<PageEntry>();
11
12      public List<NPCJournal> NPCPages = new List<NPCJournal>();
13      public NPCJournal NPCPage;
14      public bool PlayerJournal;
15      public string ReturnScene;
16      public List<NPCJournal> MatchGameNPCs;
17      public bool MatchGameReturn;
18
19
20
        1 reference
21      public void AddPageEntry(PageEntry page)
22      {
23          if(pagePool.Count > 0)
24          {
25              PageEntry newPage = pagePool[0];
26              pagePool.RemoveAt(0);
27              newPage.Answer1 = page.Answer1;
28              newPage.Answer2 = page.Answer2;
29              newPage.Answer2 = page.Answer3;
30              newPage.Date = page.Date;
31              pages.Add(newPage);
32          }
33          else
34          {
35              Debug.LogError("pagePool is out of pages!");
36          }
37      }
38
```

Appendix B

Code segment of Minigame 2

```csharp
protected override void MouseUP()
{
    bool matched = false;
    List<Collider2D> results = new List<Collider2D>();
    ContactFilter2D filter = new ContactFilter2D();
    int count = GetComponent<Collider2D>().OverlapCollider(filter, results);
    foreach(Collider2D result in results)
    {
        if (result.GetComponent<NPC>())
        {
            NPC npc = result.GetComponent<NPC>();
            if(npc.MatchingItem1.GetComponentInChildren<MatchingIcon>().ItemType == ItemType){
                print($"{ItemType} Found MatchingItem 1");
                CheckItemMatch(npc);
                npc.MatchingItem1.GetComponentInChildren<MatchingIcon>().Match();
                matched = true;
            } else if(npc.MatchingItem2.GetComponentInChildren<MatchingIcon>().ItemType == ItemType) {
                print($"{ItemType} Found MatchingItem 2");
                CheckItemMatch(npc);
                npc.MatchingItem2.GetComponentInChildren<MatchingIcon>().Match();
                matched = true;
            }
        }
    }
    if (matched)
    {
        gameObject.SetActive(false);
    }
    else
    {
        transform.position = BoardPos;
    }
}
```