Some the notes here are found in [Autism Skill Tutor GRV](https://docs.google.com/document/d/1Ole2RrNISLHZmBheafPXf2QG-YJWr4QDVLrZv8z3cHI/edit#)

# Social Development

‘Talk to AI’ could use Rules programming or pattern matching. It will have a database of words and punctuations that are classified as different emotion types, and from the emotion types and word order the reaction is developed. The program DeepMoji could be used as an implementation reference. <https://deepmoji.mit.edu/> seeing as it also has to do with emojis and reading words. Of course DeepMoji uses deep learning, so it is a bit different than what I was imagining by using rules.

Each model has a wide range of emotions that will probably use Rules to dictate when the model emotions change, or it can just be implemented with if-else statements or case switching

Punctuation

Exclamation = surprised

Question = confused

Three Periods = sad

Exclamation + Question = angry

Three Periods + Exclamation = surprised

Three Periods + Question = confused

Multiple Question = confused

Literals

Yells

Sigh

Sighs

Hugs

Laughs

Glomps

Swears

Giggles

Finger Guns

Slams

Interjection

Oh

Ew

Gross

Bleh

Meh

Ugh

Hmmm

Grrr

Tch

Eeeeee

Ooph

Ow

Yuck

Wow

Descriptives

Gross

Amazing

Awesome

Incredible

Pretty

Beautiful

Horrible

Embarrassing

Bad

Worse

Worst

Good

Nice

Confusing

Confused

Hate

Love

Stupid

Strong Word Choice

Fuck

Shit

Damnit

Motherfucker

Dick

WTF

Enhancers

Very

Too

Kind of

Kinda

Sorta

All that

That

Possible Emojis:

Happy - 😀 😁 🌞❤️😊🤠

Surprised -🤯😮 😲 ⁉️ ‼️👀

Sad - 😢 🙁☹️😭 😟 😿

Angry - ❌😒 😡 😤😠 👿

Confused/Thinking - 🤨❓😵‍💫😶 🤔🧐

Silly/Laughing - 🙃 😝👅🤭 😸 😜

Fear - 😱 😨 😬 😖 😣 😮

Disgust - 🤢🤮 😵 🥵☢️

Embarrassed - 😓 😳

Neutral - 👾🤖✨👁️‍🗨️🎶🎵✔️

| Expression | Reference Label | Emojis |
| --- | --- | --- |
| Happy | HF | 😀 😁 🌞❤️😊🤠 |
| Surprised | SRF | 🤯😮 😲 ⁉️ ‼️👀 |
| Sad | SDF | 😢 🙁☹️😭 😟 😿 |
| Confused | CF | 🤨❓😵‍💫😶 🤔🧐 |
| Thinking | TF |
| Neutral | NF | 👾🤖✨👁️‍🗨️🎶🎵✔️ |
| Laughing | LF | 🙃 😝👅🤭 😸 😜 |
| Fear | FF | 😱 😨 😬 😖 😣 😮 |
| Disgust | DF | 🤢🤮 😵 🥵☢️ |
| Embarrassed | EF | 😓 😳 |
| Angry | AF | ❌😒 😡 😤😠 👿 |
| Relief | RF | 😮‍💨😅😌 |
| Emotional Reference Table | | |

#### Examples:

1. The user sends the message: Wow, I had such a bad day!

The AI would scan the text and pull out the words “Wow” and “bad” and the exclamation point as well as the places in the text that the words occur.

Next, the AI will look at the distance between the recorded words. All recorded words with a position that is within three of each other will be considered related and linked.

“Bad” and “!” are linked. Linked words hold more weight than non-linked words.

The AI will check to see if any words recorded are the first word. In this case that is true. Because “Wow” is not linked and is the first word, it takes on its base emotion type of happy.

Because the first word is happy, the AI will check for recorded punctuation.

In this case there is recorded punctuation, “!”. Happy first word + ! = surprised.

The AI will check to see if any recorded words come after the punctuation. Because none exist, the AI will ignore the rest of the recorded words.

Happy First Word + “!” = surprised response.

That pattern of speech would lead to the AI choosing a surprised response.

1. The user sends the message: I ate something very gross today, it was not bad…

The AI records the words “gross”, “bad”, and the “...” at the end.

Next, the AI will look at the distance between the recorded words. All recorded words with a position that is within three of each other will be considered related and linked.

“bad” and “...” are linked.

Since “gross” is not the first word, the AI will move on to check if any word that has an unrecorded word before it by looking at the positions of the words. “Gross” and “bad” both have unknown words before them.

The AI will look at the word before gross, “very”. “Very” is considered an enhancer word, so the AI checks to see if there is an unknown word before that one. In this case, it does, and the AI checks the word before “very” which is “something”. The AI does not know that word and moves onto the next case.

The word before bad is “not”, which the AI will recognize and note that.

Finally, the emotions of the recorded values are processed.

“Gross” is unlinked and unmarked, indicating that it has its primary emotional value of disgust.

“Bad” unmarked would produce a sad response.

“....” unlinked would produce a sad response.

“Bad” linked with “...” would produce a sad response.

“Bad” marked would produce a happy response.

“Bad” linked with “....” and marked would produce a confused response.

Happy + Sad = Confused response.

Because the linked words hold more weight, the AI would return a confused response.

1. The user sends the message: Bleh I ate something very gross today, but it was not bad…

The AI records the words “Bleh, “gross”, “bad”, and the “...” at the end.

“bad” and “...” are linked.

Because “Bleh” is the first word, and unlinked, the AI will take its emotion type of disgust and check it against the punctuation.

“....” does not go with disgust words, and therefore the AI will continue on.

From here it goes the same as it did in example 2 until the end.

Because both “bleh” and “gross” are disgust words, they hold the same weight as “bad” linked with “....”.

Disgust + confused = surprised response.

The AI would return a surprised response.

1. The user sends the message: I am who I am.

The AI records no words. Because no words are recorded, it returns a neutral response.

‘Talk It Out’ can be modeled after ELIZA: a very basic Rogerian psychotherapist chatbot (<https://web.njit.edu/~ronkowit/eliza.html>), only more basic and niser. Functionally the programs are meant to do similar things after all, though mine may be slightly more basic as it is only trying to help the user work through a problem rather than function as a therapist. I think mine would use pattern matching in a way similar to ELIZA.

# Games

The games section is similar to the ideas presented by Brain Scasselati, Laura Boccanfuso, Chien-Ming Huang, Marilena Mademtzi, Meiying Qin, Nicole Salomons, Pamela Ventola, and Frederick Shic in their in-home social robot. The in-home social robot teaches autistic children socialization skills through games.

### ‘Match The Feeling’

Each level will have a randomized emotion for the user to match with the character. The AI will first select four random characters from a list specifically made for the level. If any of the emotions are the same, the AI will re-generate characters until no emotions are the same.

From those emotions, the AI will randomize which one the user will have to match.

After getting five correct in a row, the next difficulty level will unlock.

### ‘What To Say’

The starting mood of the character will be randomized from the list of emotions (excluding the objective emotional state).

The starting mood of the character will determine the possible conversation routes available at the start.

Game characters will have special dialogue options. Nene will be able to have a conversation thread about music. Pat has a conversation about sensory processing and getting new noise-canceling headphones. Bob can ask for advice on talking. Devan will sign about playing basketball with Devin. Devin will sign about trying out for a basketball team. Winn will need help finding his blanket.

The player can end the conversation at any time, or can play out the conversation until the character being spoken to ends it. The user wins if when the conversation ends, the character is at the emotional goal.

For most conversation routes, the exchange of dialogue will only last five to ten rounds of back and forth. Only the special dialogue options can last longer as those tend to tell a story.

For each starting emotional state of a user created character, there are 12 potential starting dialogue options. For game characters, there are 13. Three of which will be chosen randomly by the program as options.

These 11 starting points per emotional starting state are not unique across all starting moods, many are shared to decrease the size of the dialogue tree.

Below is a visual of how the dialogue tree could work with just the first two sets of dialogue options.



On the second dialogue exchange phase, the player can backtrack on both saying “I love you” and saying “Your Shoes Look Stupid” by selecting “Just joking”.

Just joking will always cause the character to feel relief regardless of what came before it and allow for the generation of three new conversation starters.

To calculate the emotional state of the character, the AI will look at the pattern of emotion the user caused the character model to feel.

All starter dialogue has a base emotion that is independent of all past emotions.

When the conversation ends, one last emotion will be generated.

Example Game Play:



The user chooses the dialgoue “Your shoes look stupid”.



The character will have a sad face, and the new response options will be given. The user chooses to “Just joking”.



New starter dialogue is generated at “just joking”. The user decides to end the conversation.



Because the user ended the conversation after making the model character sad and then relieved, the character model ended up confused.

### ‘Who Am I?’ and ‘Take Out The Trains’

‘Who Am I?’ is basically Guess Who?, which has single-player versions online. <https://www.memory-improvement-tips.com/guess-who-game.html>

‘Take Out The Trains’ is basically just battleship and tons of single player versions of it exist. <https://www.mathsisfun.com/games/battleship.html>, <https://www.battleshiponline.org/> .

The most complex feature is the ability to lie. ‘Who Am I?’ and ‘Take Out The Trains’ both have reactions if the player is caught lying. This feature is pretty basic, there is just a variable containing the base odds of getting caught and another for getting caught during a game. The AI will add the base to the game odds every time the user lies, and will basically choose a random number between 1-100 and if the AI rolls within the percentile the user gets caught. It then will reset the game odds base to base odds. If after the game ends there are no more characters that will play with the user, base odds are doubled because the program is learning to expect the user to cheat. This is a bit like how Sans in Undertale knows your previous game data.

‘Just Breathe’

The game ‘Just Breathe’ works similarly to Flowly’s tranquility VR (<https://www.flowly.world/>), which uses heart rate to detect breathing via a controller. It also could be like the game One Hand Clapping, which is a game controlled by using the voice. In this case the voice would just be breathing, but it’s the same audio processing idea of using the voice in a certain way to get a certain result.

‘Finding Items’

‘Finding Items’ is basically just a bunch of randomized variables and functions like “Where’s Waldo?” or a spot-the–difference game. Both of which have versions for the computer.

‘Same Time’

‘Same Time’ is based on an improv game and needs to implement pattern matching or something similar. This is one of the most complex games because I did not have a computable game as the basis, or one that I knew of that was similar that could utilize AI effectively. I suppose it is similar to predictive text, only in game format. A similar structure used for text-prediction may be implementable for this game.