Weekly Schedule Schedule: Monday, 7:00 PM - 8:00 PM
In-Person, Wednesday 3:00 - 5:00 PM, Saturday 1:00 PM

Our github repo: https://github.com/acmucsd-projects/fa23-ai-team-2
Attendees: Vincent, Hargen, Phillip, Sia, Ryan

Summary of Meeting

Action Items

- [x] ~~**Brainstorm project ideas**~~
- [x] ~~Copy over the folder structure from https://github.com/acmucsd-projects/fa21-lion/tree/ML-setup/ML~~ ✅
- [ ] Update the README -> make it look nicer
  - [x] ~~Get a team banner (1280w x 680h)~~
    - [x] ~~A loading GIF~~
  - [x] ~~Table of contents~~
  - [x] ~~requirements/getting started/installation section <- tonight~~
  - [ ] Author info (maybe add your contacts)
  - [ ] Add emojis to the headers
- [x] ~~Change repo name and discord server name to Team TBD~~
- [x] ~~Within the README Requirements -> detail step by step how an outsider can contribute~~
  - [x] ~~Git clone <repo>~~
  - [x] ~~Pip install -r requirements.txt~~
- [x] ~~Update meeting note folder + gitignore~~
  - [x] ~~.gitignore should ignore the contents of the input/ (and maybe the models/ folder if models are large) *like including them right now? We do not have any content in input/model folders yet.*~~
- [x] ~~Look into your top project idea picks~~
  - [x] ~~Elaborate more on the idea~~
  - [x] ~~See where you can go with it~~
  - [x] ~~Identify possible issues with the project (i.e. hardware constraints, time constraint, team constraints, success factors)~~
  - [x] ~~Identify the data (where the data comes from and how we process it)~~
  - [x] ~~Write all of this up in some shared document between you guys~~
- [ ] Learn the Basics — PyTorch Tutorials 2.1.0+cu121 documentation
- [ ] (1) PyTorch Prerequisites - Syllabus for Neural Network Programming Course - YouTube
- [x] ~~Any useful resources for the team can be organized into this resources/ folder~~
- [ ]

Ideas

- Personality test; Using nlp to create a chatbot with a personality shaped by how you talk to it/replicates the user's personality
    - What if the model retains its personality over time
- Processes video/movie and summarizes it with context (NLP + CV?)
- Cuisine identification\
    - Also could generate recipe/nutrition facts
- AI generated image detection (generating writing pieces)
    - https://www.kaggle.com/datasets/birdy654/cifake-real-and-ai-generated-synthetic-images
- Skin condition/disease detection
- Suggests optimal webreg/course schedule, search problem
- AI to play game (tetris? Or more complicated)

Timeline

Votes

- Aryaman
    - 1) Cuisine identification
    - 2) AI to play game
    - 3) Personality Test
- Hargen
    - 1) Personality test
        - Elaboration: leverage LLM to train a model where we can match word semantics with personalities (by a chunk of text inputs/audio clips). Use the model to form a chatbot to 1) adaptively learn a person's personality and replicate the exact same personality to talk back. We'd like to cache the history of conversations (or save the model somewhere) so that the chatbot could replicate the same personality next time. Not sure about exact LLM techniques available, but clusterings analysis might be an idea to distinguish among different personalities.
        - Issues: if we want to cache the conversation history, we would probably need much memory space in the training process or later if we want to turn the model into a usable application. The NLP/LLM type of stuff seems new to most of us in the team (but this would be the exciting part where we can explore a new niche of machine learning and AI). We might want to utilize GPU to improve computational time, as we might train our model on a pretty large dataset for the model to obtain above-average accuracy (compared with random guessing or some sort).
        - Dataset: write a computer program to generate some text from chatgpt and then feed in these text. Or find some online database with personalities as labels so we can leverage the existing dataset to train the model (obviously we need and want to make sure the dataset is reliable enough for it to apply to our target users….).
    - 2) Process video (gone bc of low vote, so I'm skipping this… to save time)
    - 3) AI to play game
        - Elaboration: find an adversarial game so we could possibly train an AI agent to play the game against human player(s). We could possibly

gather some game data from real time or some sort of database to train a RL model (Q-learning or something like that) so the AI agent knows the reward/punishment system of the game. Then we fine-tune the parameters to suit the needs of our AI agent. Really cool idea as we could possibly play an interactive game in the showcase! Or we can design an agent for a single player game where AI plays to gain as much utility as possible.
        - Issue: Also memory issue bc we are dealing with games (video/text or some similar input which would take much memory space and time for the computer to process. Otherwise, we would need a summary of the game reward/punishment system that we could probably feed into the computer to train the RL model.
        - Data source: we play the game! Or find some potential open source dataset from the website. Or we could probably find some existing AI agents and then see how we can improve on the existing work.
- Phillip
    - 1) AI to play game
        - Use reinforcement learning in order to teach the AI what to do and let it improve.
        - Interested in making an AI to play watermelon game (which is similar to tetris and 2048 and puyo puyo)
            - A lot of people have already made Tetris AIs
        - Reinforcement learning may take a long time and a lot of iterations since the AI will be taking its own actions and the system in place to reward it might not be very effective initially, wasting a lot of time in the process.
    - 2) Personality test
        - Have an initial dataset of words, phrases, etc and personality associated with those things.
        - Take text input from the users in order to copy their personality based on the initial dataset.
        - Use a chatbot to write back using that personality of the user writing to it.
    - 3) Cuisine Identification
- Sia
    - 1) Personality Test
        - Will need to find online data sets of speech and text samples and clean them (tokenization etc) along with a data set of extracted/ key phrases with corresponding personalities
        - Tokenization of the text to find key words/phrases that can help determine personality(speech to text and then tokenization)
        - Will need to research more in depth about speech recognition
        - Train the model to be able to communicate effectively with the user based on the user's personality/motivations. Therefore, each interaction must be personalized.
        - Issues: there can be biases in the personality based on age, gender etc, ensure all biases are removed/accounted for, accuracy will be an issue as we will require a great amount of memory to take into consideration all data ie our initial dataset needs to be large enough to account for all possibilities however we don't have enough memory for the same, for speech recognition, accents/languages/ constraints, background noise

    - 2) AI generated image detection
    - 3) AI to play game
- Catherine
    - 1) AI to play game
        - Possible games we could do: Fighting game, poker, turn-based city building strategy games, World's Hardest Game, mario
        - Create horror-maze game where AI tries to find player, and player must hide from them
            - May come into difficulties learning how to do image generation/design
                - Time constraints: would have to learn new tools
            - Not sure what dataset to use to train model, or how to select what behavior to reinforce
            - Could also do reverse -> train AI to beat an existing popular stealth-type horror game so we can avoid problem of needing to create game
                - Training set could be a compilation of yt tutorials/playthroughs of the game and reinforcement learning
    - 2) AI generated image detection
    - 3) Cuisine Identification
- Ryan
    - 1) WebReg recommendation, course plans
        - Would be really interesting to create as well as very useful for students
        - Students would input their transcript and then based off it we could recommend specific courses and plans
        - They could ask questions regarding each course and the AI would provide feedback and data
        - Difficulties would be the raw computational power needed to run this
        - Lots of storage would be needed
        - Would need a proprietary model to train based off our data
        - Difficult to create and maintain an accurate dataset of classes
        - Would be a lot of issues testing as there would be many many combinations of course plans
        - Data would come from UCSD catalog, unsure how accurate it is between quarters
    - 2) Personality Test
    - 3) Video processor

Result (3 points for 1, 2 points for 2, 1 point of 3):

AI to play game: 10; Personality Test: 11;

Cuisine Identification: 5; Process video: 3; AI generated image detection: 4; WebReg: 3

Total: 36 points.

Project: Personality Test

Overview

Until Wednesday:

- Research frameworks and datasets
    - Plenty of existing ones
    - https://arxiv.org/abs/2105.11798
    - https://ygsl-crew.medium.com/personality-detection-and-prediction-using-natural-language-processing-c2cd5cb4a2c7
    - https://medium.com/@faaezriaz/briggs-myer-personality-prediction-with-nlp-ec7b30a08942
- Send author info
    - github, linkedin
- Sentiment analysis
    - Categorize sentiment positive/negative
    - Use pytorch
    - no libraries!!
    - https://github.com/bentrevett/pytorch-sentiment-analysis
- Take the existing code from previous MBTI classification project and learn the pipeline of the model

Resource Section

https://www.kaggle.com/code/arunmohan003/sentiment-analysis-using-lstm-pytorch

https://www.kaggle.com/datasets/tunguz/big-five-personality-test

https://huggingface.co/datasets/Shunian/kaggle-mbti-cleaned/blob/main/README.md?code=true#L19

- Learn PyTorch and setup a basic pipeline