AI-VVO sdmay22-36 Weekly Update # 1

9/27/2021 - 10/3/2021

Front-End (Analysis of sdmay21-24)

- Previous group mostly used functions in react and not many components
- There is a not a lot of documentation on the code
- The README is a basic instruction of how to run the application
- The current code base for the frontend includes a login screen, a reset password page, a main homepage, and a power grid view
- Completed ReactJS Tutorial suggested by previous team

Front-End (Plan for Next Week)

- Complete a more in-depth tutorial using ReactJS to learn fundamentals and implementation
- Get a better understanding of the front-end code from the previous team
- Running current App on local machines to understand flow of information and general layout
- Refine front-end requirements

Back-End (Analysis of sdmay21-24)

- Looked into the previous teams implementation of the backend using Django.
- The backend is based off of the guide shown below which shows a basic implementation of a React app using Django as a backend https://datagraphi.com/blog/post/2020/8/30/docker-guide-build-a-fully-production-ready-machine-learning-app-with-react-django-and-postgresql-on-docker
- The backend also contains the machine learning api which the Front End uses to display our predictions.

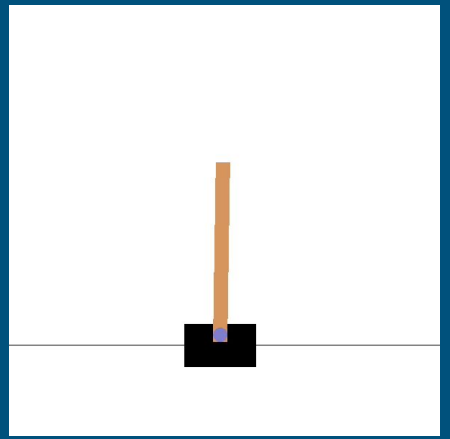
Back-End (Plan for next week)

- Continue to research both Django and it’s implementation as well as continuing to practice with Python.
- Try to look into the possible reasons the previous team may have had issues integrating PostgreSQL into the backend.
- Basic understanding of the Machine Learning algorithms we will be using to ensure that the ML api has all the necessary features.
- Look into ways to make the code more readable especially in the ML api

Machine Learning (Analysis of sdmay21-24)

- Analyzed the previous group's Main Algorithm Python Notebook MLModel in prediction folder
- Previous group used Q Learning algorithm with Pandas and Numpy. Q learning basics are explained in the video linked below. Q learning common example is polecart shown below.
- Enumerated the set of actions that will be used to control the grid:
  - 3 Voltage Regulators with 11 positions
  - 1 Capacitor Bank with 2 positions (on or off)
  - There will be 35 different actions in the action space

https://www.youtube.com/watch?v=xMZE-9WECQE&ab_channel=PabloSamuelCastro



Machine Learning (Further research)

- Explored different libraries: PyTorch, Pandas, Numpy
- Got started with Google Colab
- Created a Machine Learning folder in the shared Google Drive, where all experimentation and development for the new deep learning network will be done
- Will develop mostly on Google Colab and will push to Gitlab when milestones are met
- Learned about TPUs, GPUs, and CPUs for neural network training. GPU in Google Colab is free. It's a single core of an Nvidia K80. Should be sufficient for training our model. If we run into resource limitations, we will investigate other options.

Machine Learning Folder with Colab Files can be found in the shared Google Drive or at the link below.

https://drive.google.com/drive/folders/1FT91I2k50RsjTc_6momeV3-G9QdumxKq?usp=sharing

Machine Learning (New Approach)

- Most exciting reinforcement learning work comes from deep convolutional neural networks.
- Deep convolutional neural networks for reinforcement learning with PyTorch will be the primary strategy for the new machine learning algorithm.
- Researchers at DeepMind created a deep convolutional network that could play old Atari games at superhuman capabilities (Training a model for an Atari game on a high-end GPU takes about 5 days)
- Atari games have large input (a 210x160x3 image matrix at 60Hz). 210x160x3x60 = 6,048,000 param/s
- Preprocessing is used in the Atari project to reduce this input to a grayscale 84 x 84 image. 84x84x1x60 = 423,360 param/s
- However, there are less input parameters for the AI-VVO project, so processing shouldn't take nearly that long

The paper on using deep convolutional neural networks to play Atari games is linked below.

https://deepmind.com/research/publications/2019/playing-atari-deep-reinforcement-learning