EE / CprE / SE 491 – sdmay24-47
Accurate Cancer Prediction Using AI
Bi-weekly Report 4

Mar 16th – Mar 30th
Client: Ashraf Gaffar
Faculty Advisor: Ashraf Gaffar

Team Members:

Jack Sebahar — *Project Manager / Client Liason*
Nicholas Otto — *Model Design Lead*
Mason Wichman — *UI Design Lead*
Lal Siama - QA Engineer, Tester
Helen Lau - UI Design
Isaiah Mundy - UI Design Test

Weekly Summary

During the past two weeks, we had three main goals: convert our desktop app to a web app, deploy the model on the cloud, and continue refining our model to increase its accuracy. We learned that creating a desktop application was not sufficient enough for our project, so we have to convert it into a web application. We also learned that we cannot embed the model metadata directly into the application and it must be hosted on the cloud and called via endpoints. As such, we have started the process of doing those two things. Our UI is getting converted into a python Flask application that behaves similarly to the one we already had. We also have begun setting up the model endpoint. There are some considerations we have to make regarding costs to deploy the model, but with correspondence with the TA and our advisor, we are arranging the ability to do so. As for our model, we employed a few new strategies to increase its accuracy. One main suggestion provided by our TA was that we should attempt to scale the data differently, as portions of the data may only be significant, and including all of the data points within a set could skew accuracy. Unfortunately, this did not yield much better results. Another recommendation the TA made was to investigate unsupervised learning as opposed to supervised learning. Since we are all new to machine learning, this is something we are going to have to research in the coming weeks to see how we can utilize it for our model.

Past Week Accomplishments

- Continued Model Error Reduction Testing
    - Tried utilizing only certain partitions of each data sample in the training of the model such as the first 250 data points, first 500 data points, etc. Was able to achieve a small reduction in error but ultimately not anything extremely significant. We seem to have reached a plateau in model error reduction, so we also investigated unsupervised learning as a potential alternative method to identifying significant characteristics in our data that correlate to cancer recurrence.
- Model Hosting and Web UI
    - Our initial design was a desktop application that involved embedding the model metadata directly into the front end to return predictions. However, we need to host our model on

the cloud, develop our UI as a web app and obtain model predictions via endpoints on the cloud. This involves replicating our UI on a web app platform and deploying our model as an endpoint. The end result should be similar, but this provides better scalability and maintainability for our application. We have begun developing the app using the python web application framework Flask. We have also started the process of deploying the model on the cloud.

Pending Issues

- Billing requirements
    - Google cloud provides a certain number of hours of model deployment for free when you create a new account, so we must do that in order to utilize the endpoints.

Midterm Feedback

We met with the other team working on the Accurate Cancer Prediction using AI project. The other team generally liked our approach to improving model performance and thought we had a pleasing desktop application set up. Both of our groups seem to have hit a plateau in model accuracy. We both discussed the methods we have tried and are going to try.We both agreed that the next step is to attempt unsupervised learning as a means to potentially extract more information within the data and feed that information into a supervised model. This is a common tactic in machine learning to extract more useful parts within a dataset that may not be obvious. We also discussed our web application and some aspects of it that we liked and needed to improve on.

Insights

- The other team also suggested that we should explore unsupervised learning and see what new information it can provide for our model development
- The other team also realized our desktop web application may not be sufficient for the requirements of the project and we should develop a web application capable of interacting with thee deployed model on the cloud

Steps

1. The first major step we need to take is research unsupervised learning. None of us really have in-depth knowledge on machine learning, so this is a concept we need to familiarize ourselves with to understand how it can benefit our process.
2. Once we have gained an understanding of unsupervised learning, we need to attempt it on our dataset to see what we can extract from it
3. Repeat this process and see if it can yield better accuracy in our model
4. Simultaneously, we need to deploy our model on the cloud. This shouldn't be too difficult, as it mostly entails deploying our current model as an endpoint and have it run on the cloud
5. Web app development. We need to update our front end as a web application that accepts csv data as user input and makes endpoint requests to our model to return predictions
6. Lastly, we need to test our entire application to make sure it runs smoothly and consistently.

Individual Contributions

Team Member	Contribution	Weekly Hours	Total Hours
Jack Sebahar	● Model cloud deployment ○ Began deploying model on the cloud to return predictions to front end as API call ● Web app development ○ Began development of basic UI to accept pathology data	7	35
Nicholas Otto	● Model error reduction ● Investigation of unsupervised learning ● Recording past model error	5	47
Mason Wichman	● Data preprocessing ○ Data peak extraction script ● Model accuracy improvement ○ scaling data and unsupervised learning	5	30
Lal Siama	● Data preprocessing ○ Methods research ● UI design improvements	4	28
Helen Lau	● AWS initialization ○ Set up development environment similar to GCP ● UI Web application design	3	24
Isaiah Mundy	● Data Preprocessing ● Model Design ○ Normalizing and extracting peaks from data	3	24

Plans for Coming Week

- Investigate more effective options for data preprocessing - Mason
    - Determine if there are more suitable options for preprocessing our data
    - Determine if any Keras models are designed to work with data like ours
- Reduce Model Error - Nick, Isaiah
    - Train an autoencoder on our data as a method of unsupervised learning to identify unseen patterns or indicators of cancer recurrence in our data.
- Convert UI to web application - Jack, Siama, Helen
    - Our current desktop application needs to be converted into a web application with similar functionality. This should be fairly trivial, but is an important aspect of our project.
- Deploy mode on cloud - Jack
    - We need to deploy our UI as a web application capable of making endpoint calls to GCP to return model predictions.
    - This needs to be done on GCP and AWS to see if it has any effects on performance