ELE 408 Final Project

RI Vaccine Appointment Scraper

Giles Lanowy and John Hunter

Introduction:

Overview:

The main idea for this project is to create a user interface that would make it easier for people of all ages to book vaccine appointments. The project starts with code for scraping information from the government's website pertaining to vaccine appointments. This information included the location, the number of appointments available, the vaccine type, and more. The second component was a client-server interaction with the raspberry pi microcontroller. This component displays the number of available appointments on the Sense Hat depending on the location the user has chosen. The third component was implementing a machine learning algorithm. Our model was used to predict the number of appointments available for a location on any given day of the week. The last component was creating a graphical user interface or GUI that tied all other components together. The idea was to create a user friendly app that was much easier to navigate than the government's website. Additional details for all sections including methods and difficulties are provided in the pages below.

Related Work:

A major inspiration from our project came from a project very similar to ours in New York. The project was created by Huge Ma, a software engineer at Airbnb. He created a website called TurboVax, 2021 © TurboVax. This site, much like our project, finds appointment data from government vaccine sites for New York City. Ma however went a few steps further than our group. His site automatically shows all appointments available in New York. Ours requires the user to input their zip code first. The user is then allowed to filter out certain locations based on their area of NYC. Our only filter is the date. Huge Ma’s software is designed to benefit the people of New York, while our software is catered to Rhode Island citizens. Huge Ma’s software

is on a website that allows users to access the data from anywhere and his web server automatically refreshes the appointments before users access the webpage. Future development on the RI Vaccine Appt Scraper would involve implementing the scraper into a website, establishing a web server to host the site, and programming automatic scraping to provide consistent and accurate data to users.

System Design:

Website Scraper:

The project currently only has the capability to scrape information from the government's vaccination appointment site. The code is written in python and uses two well known packages when working with web pages. These packages are requests and Beautiful Soup. We used the get method from requests to acquire the content of the webpage. Once we had that a Beautiful Soup object was created from the source content. Next we isolated all , <p>, and <a> headers useful to our project. We then parsed the information found by using soup's find method to store individual pieces of information as variables to be used later. These pieces of information are as follows; Name of location, Date, Address, Vaccine type, Appointments available, the link to signup, and the time accessed. These fields were taken for each location listed on the government's website and stored into a pandas dataframe. The data frame format allowed efficient access to these fields for the other components of the project. All of the data collected from the website is stored locally in the server's files to provide us with a history of past appointment data. This data is filtered and sent to the Graphical User Interface.

Graphical User Interface:

Our project's GUI was created using the tkinter python software library. Its purpose is to incorporate all the functional parts of the project into one. When the program is launched there are two text entries and a button. Without entering a zip code the program will fail however this is the only required entry. The date entry is only for people looking to make an appointment on a given day or narrow down their results. This makes sure only the appointments for that day are scraped from the website. Once all the data is acquired the GUI updates itself to show location, the number of appointments, the date for the appointments, and the vaccine type. To the right of

these columns is a new button for each site that will open two things. The first is the webpage where the user can start to enter their information for the appointment. The second is a new window that shows our machine learning model's prediction for how many appointments will be available throughout the week. It also shows the model's accuracy. If the user decides they want to book a different location or day they can close the webpage and prediction window and click on a different "Book Here" button. Additionally if they want to do another search or narrow it down by picking a date all they have to do is press "Find Appointments" again. This will update the list with the new data from the website.

Raspberry Pi:

One of our tasks for this project was to include the Raspberry Pi microcontroller. For this inclusion, the team decided we would set up a server on the raspberry pi. The client could then connect to the server using socket programming. When the connection is established the server loops waiting for the client to send it messages. As of right now, there are only two messages it handles. The first is a message including the number of appointments available for a particular location. When the pi server receives the message it turns on the 8x8 array on the Sense Hat and displays the color green when there are 10 or more appointments available at the location. When the number is between 9 and 5 the Sense Hat shows the number in yellow. A location with less than five available appointments displays the number in red. When the GUI is closed the server receives a message with a negative number letting it know it is time to terminate the connection.

Machine Learning:

The machine learning component was largely created using the scikit-learn library for python. Throughout the last few weeks of the semester we have been running our web scraping program and storing appointment data onto the host server. All of the collected data is combined into a

pandas dataframe. For our machine learning algorithm, we constructed the algorithm using Location and Date as our input “X” dataset. Since the machine learning algorithm requires integers or floats as inputs we had to encode the location and date into a numerical format. For location, this was done by assigning the index number from our list of locations for each location. For example, if Sockanosset POD was the first location listed on the government's website we assigned this location to 0. We use datetime’s method weekday to assign a number 0-6 to the date. Our output “Y” dataset is the predicted number of appointments on a given weekday. The model we used was a Decision Tree Classifier and we built the algorithm from the scikit-learn library. We used 80% of the data we had collected to train the model, and 20% of our existing data to test the model for accuracy. Our testing results showed that our model had on average a 62% accuracy rate. Our finished machine learning model can predict the number of appointments that will be available on any day of the week at any location. This algorithm tends to list Fridays through Sundays as having generally more available appointments. The more times the RI Vaccine Appt Scraper is run, the more accurate the machine learning model will become.

Discussion:

Difficulties:

Most private institutions like CVS or Walmart required login credentials to access site information. We were not sure how trusting people would be with their usernames and passwords so we decided to omit this from our project. Instead we made a tool that made navigating the government's website much easier and more forgiving.

One of our stretch goals for this project was to implement a maps section that would show the location of the site on a map. For this we would be using the Google Maps API. This has not yet been achieved but will be included in the future of this project if it continues.

Our data collection for generating the machine learning model has not had an appropriate amount of time to collect enough data to produce a reasonably reliable machine learning algorithm. Every time the software is run, the current data is stored on the server's harddrive, and over time the amount of data collected will increase the machine learning model's accuracy. In the beginning of the data collection process, we were seeing between 1 - 10 appointments for each location. However later in the development these postings started showing numbers in the hundreds and thousands. The amount of available appointments has dramatic changes throughout the few weeks we have been collecting data. We predict that as the number of appointments stabilizes in the coming weeks, our algorithm will refine itself to more accurately reflect the current number of appointments.

Potential Improvements:

One of the biggest improvements to our program would be to include additional site scrapers that would handle data from private institutions. Another would be to provide travel time to the destination from a given zip if not implementing a Google Maps interface. Over time our

model's accuracy will continue to increase due to the fact that we are constantly saving new data into our data folder to train the model on. This project's future will involve establishing a website to host the software and a dedicated server to collect and process the data.

Conclusions:

The government-run website, www.vaccinateri.org, is extremely inefficient when it comes to allowing people to sign up, especially when appointments are booked every minute and in high demand. For one, there are 10 pages to search through all with varying dates and locations. Two, the appointments that are listed as available may be being booked when you try to sign up leaving you with no choice but to start from page 1. Our project provides an easy way to navigate and sign up for appointments at the Rhode Island Government run facilities. This software will help people schedule appointments quickly before the appointments run out, make scheduling appointments easier, and provide users with valuable information regarding vaccine appointments.