Udacity Machine Learning Engineer Capstone Proposal

Christopher Weaver

8/17/2021

Domain Background

The past couple of years have presented us with a unique opportunity to utilize increasingly accessible hardware and sophisticated software to enhance individual health outcomes and knowledge. Smart phones continue to increase, not only in computational power, but also in the array of sensors these devices contain. With the release of the Apple Watch Series One, Apple made a marketing and engineering commitment to empowering users to monitor their own health through technology. In recent years, wearables such as the Apple Watch have been fitted with sensors to help users monitor heart rate, blood oxygen levels, and even check for cardiac arrhythmia. As these sensors become more prevalent, it opens up opportunities for software engineers to find new ways to use these sensors. Monitoring heart disease, physical activity, and environments that can lead to hearing loss have already taken shape; but there is more that can be done. Companies such as Cardiogram are banking on just that. Research conducted in partnership with the University of California found that utilizing LSTM neural networks and "off the shelf" wearable heart rate sensors, medical conditions such as diabetes, high cholesterol, high blood pressure, and sleep apnea could all be detected with high accuracy[1]. I propose that similar techniques can be extended beyond just these diseases into a whole host of other ailments and conditions.

Problem Statement

In the spring of 2020, individual cities, counties, and states began implementing restrictive mitigation strategies around the novel corona virus SARS-C0V-2. Many feared either contracting or spreading of the virus and took to quarantining at home for weeks or months at a time. Many institutions scrambled to find ways to resume semi-normal activity while at the same time keeping all involved safe and comfortable. Different testing solutions were rolled out as a mechanism for helping individuals know if they were infected with the virus so that they could make better informed decisions. But many of these tests are slow, expensive, inconvenient, and too inaccurate for individuals to take on a regular basis. What is needed is an easier way to test and monitor for Covid-19 that empowers more people to make responsible health decisions. The goal is is to build a classification machine learning model that can accurately diagnose Covid-19 using bio-metrics gathered either by api's to Apple's

---

[1] AAAI Conference on Artificial Intelligence, Feb 2018 (AAAI-18). https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16967/15916

Health app or from user input to specific questions such as “Are you currently experiencing a cough?”. Inputs would include numeric values (body temperature, pulse, and Sp02) to output a class output of either 1 or 0, with 1 representing a Covid-19 diagnosis and 0 representing no indication of infection with Covid-19. A further model would be developed which takes binary classification values such as the existence of a cough, sore throat, and headache and again outputs a class value of 1 or 0, with 1 representing a Covid-19 diagnosis and 0 representing no indication of infection with Covid-19. Each feature in the second model will be represented as a 1 if the user is experiencing the symptom or a 0 if they are not.

Dataset and inputs

After more than a year and a half of scientific scrutiny over Covid-19, we now have a pretty good idea how the virus impacts individual human health and the symptoms most common to be presented. The first step to finding a solution to the problem statement above is a good data set. I will be utilizing two different datasets to help tackle the issue. The first dataset contains biometric data usually referred to as vitals, such as temperature, pulse rate, and Sp02 levels. Each row of the dataset contains one measurement for each and a binary classification column for either currently being infected with Covid-19 or not at the time of the measurement. The dataset can be found here:

https://www.kaggle.com/rishanmascarenhas/covid19-temperatureoxygenpulse-rate

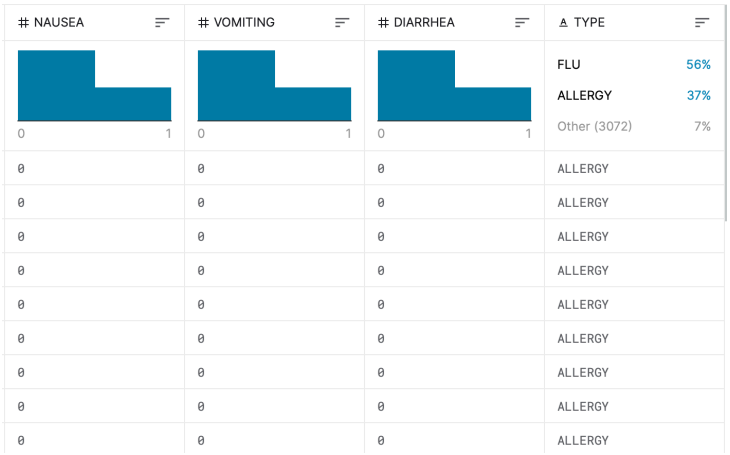
ID Identifier	Oxy Oximeter values (SpO2)	Pulse Pulse rate reading (BPM)	Temp Body Temperature (F)	Result Result
 0 9999	 85 100	 40 130	 95 105	Negative Positive
0	98	65	95	Negative
1	96	92	95	Negative
2	95	92	99	Negative
3	97	56	96	Negative
4	88	94	98	Positive
5	94	100	103	Positive
6	88	81	104	Positive

Preliminary evaluation of this first dataset shows 10k unique examples to train on. Final classification results which we are training our model to predict are evenly distributed

between Positive and Negative. Both Sp02 and Pulse features have a pretty even distribution of values between minimum and maximum values. The temperature feature has a significantly uneven distribution above 105 degrees F. Considering the even distribution of target values, this suggests very high fevers will be a significant indicator. Further correlation analysis will be needed to verify.

The second dataset to be utilized contains many more biometrics in binary classification form. These include cough, muscle aches, tiredness, and sore throat, among many others. Each value is represented as a 1 if the user had the symptom and a 0 if they did not. Each row also contains a value for diagnostics which could be Flu, allergy, cold, or Covid-19. The dataset can be found here:

https://www.kaggle.com/walterconway/covid-flu-cold-symptoms



# NASEA	# VOMITING	# DIARRHEA	A TYPE
0 1	0 1	0 1	FLU 56% ALLERGY 37% Other (3072) 7%
0	0	0	ALLERGY
0	0	0	ALLERGY
0	0	0	ALLERGY
0	0	0	ALLERGY
0	0	0	ALLERGY
0	0	0	ALLERGY
0	0	0	ALLERGY
0	0	0	ALLERGY
0	0	0	ALLERGY

Preliminary investigation into this dataset shows that it does not possess an even distribution among the possible target values. The Flu is significantly higher than any other target. The Flu and Allergy make up more than 90% of all values for our target. Our unbalanced dataset could result in a model that looks very accurate, but does poorly when it comes to predicting Covid-19 as the model found learning about this target not relevant to obtaining a low error rate. Most likely, we will have to rebalance the dataset by removing large portions of flu and allergy records. Luckily, with over 44k records, 2k of which are covid records, we can rebalance and still have a significant number of records for testing.

Solution Statement

The solution for our problem statement is two-fold. The first is to build an iOS app that utilizes biometric data taken from both an Apple Watch and iPhone and run them against a model that monitors for Covid-19. Utilizing the first dataset mentioned above,

we will build a model that takes in temperature, pulse, and SpO2 metrics and outputs a probability that the user has Covid-19 or not. This first model can monitor vitals throughout the day using an iOS application running background services and making api requests to Apple HealthKit. Unfortunately, the model in itself will not be sufficient for predicting infection with Covid-19. This is because other infectious diseases such as Influenza will also result in similar symptoms that can lead our first model to predicting a false positive for Covid 19. To overcome this, a second model will be built that takes as input, user responses as binary yes or no's to a list of questions regarding symptoms the user may be experiencing. This second model will use the second dataset provided above in order to help differentiate if the user has Covid-19 or perhaps a different underlying issue such as the cold or allergies. The second model will require users to answer questions within the app and would only be requested with the first model outputs a sufficiently high probability that the user has Covid-19. Together, both models may be able to predict with high accuracy if the end user has Covid-19 at any given time with little cost and effort.

Benchmark Model

The most common test for Covid-19 is a PCR test, usually done by a nasal swab. Some studies suggest that the sensitivity of a PCR test may be as low as 72% for those displaying symptoms and only 58.1% for those not (**https://www.healthline.com/health/how-accurate-are-rapid-covid-tests#how-accurate-is-it**). False positives are almost non-existent for PCR tests which means they have an extremely high specificity. Since our solution involves using measurements of user symptoms to diagnose and predict Covid-19, we will only benchmark against PCR tests when displaying symptoms. Because we are attempting to find better and easier ways to help people identify if they are infected with Covid-19, common tests such as PCR are a great benchmark to try and beat. Realistically, because our solution to the problem is significantly cheaper (assuming the user already owns an iPhone and Apple Watch) and far more convenient, we would probably be justified in having a lower accuracy target than the PCR test. It is worth noting that our dataset may have been generated using PCR tests themselves, which means our own model will suffer from the same inaccuracy issues as the PCR tests themselves. But again, due to the large benefits of our method of testing for Covid-19, slightly worse accuracy results may be an expectable tradeoff. Due to ethical concerns, we cannot test the accuracy of our models through experimentation; so we will have to assume the data from our dataset will be accurate as we test against that.

Evaluation Metrics

Standard evaluation metrics can be used for our solution. The two we will most focus on are sensitivity and specificity. When it comes to Covid-19, we much prefer a user gets a false positive than a false negative. False negatives will lead to users potentially acting in a manner more likely to infect others which goes against the major point of this project. Standard train test splits will be used to help us evaluate our models, with

particular attention paid to both precision and recall when testing against the test set. Depending on the model of choice, some models will likely output a continuous value between zero and one which we will interpret as its confidence percentage of a diagnosis. A standard log loss evaluation metric will also be helpful in evaluating and further training the model.

Project Design

To start, a Jupyter notebook will be used to start simple data wrangling and exploratory analysis of the datasets. We will evaluate correlations between different features in our dataset to help us get a better idea of the relationship between data points. Matplotlib will be utilized to help display, in graphical form, what the data looks like and how best to use it. We will look for things such as a poor distribution of feature options that may lead to bias in our models. Such bias would keep our models from generalizing well to the real world.

Next, we will need to preprocess the data in order for it to be usable by our models. We anticipate regularization will be necessary for some of the features such as temperature which contain higher numeric values that can lead to models over-emphasizing the importance of the feature. We will build out multiple models and evaluate which ones perform best for the classification task. Among the model we will try are SciKitLearns linear regression and decision tree models. We will also try a Pytorch neural network.

Finally, a simply iOS application will be built out that can query for bio-metric data in the Apple's HealthKit to run the first model. If the first model predicts the user has Covid-19, the app will next notify the user and ask them to answer a bunch of questions about symptoms that the second model can use to further identify what the user might be infected with.