Symptom Based Disease Prediction using Machine Learning

1st Kyle Jerome Just
Computer Science Department
California Polytechnic University
Pomona, USA
kjjust7@gmail.com

2nd Joel Joshy
Computer Science Department
California Polytechnic University
Pomona, USA
joeljoshy3243@gmail.com

3rd Musa Waghu
Computer Science Department
California Polytechnic University
Pomona, USA
musawaghu@gmail.com

4th Joshua Furman
Computer Science Department
California Polytechnic University
Pomona, USA
jafbird52@gmail.com

5th Anita Mehrazarin
Computer Science Department
California Polytechnic University
Pomona, USA
mehrazarin.a@gmail.com

***Abstract*—Machine learning has shown great promise in disease prediction using clinical and demographic data. In this paper, we propose a novel approach for disease prediction using machine learning techniques and evaluate its effectiveness in predicting several diseases using real-world healthcare datasets. Each diagnosis will be based on collected data from a patient that is then applied to our classification model to give a single, final diagnosis as well as treatment options and more official tests for an identified illness.**

***Index Terms*—Disease, Symptoms, Automated Diagnosis, Decision Tree Classifier, Random Forest Classifier, Naive Bayes Classifier, Healthcare**

I. Introduction

The healthcare system, especially here in the USA is highly expensive and often avoided by people for this reason. Often people dismiss their symptoms to be as trivial illnesses and refuse to see a professional. Unfortunately, if these symptoms point to a more serious illness, it may fester until treatment may be much more difficult or even impossible. Depending on the accuracy of each model, we may decide to use a random forest model, Naive Bayes classifier, or decision tree to classify each illness. A dataset of symptoms with their subsequent diagnosis will be used with these models to attempt to predict the illness of a patient. This patient will submit their symptoms, and the severity of each, to the model and receive a preliminary diagnosis along with resources to follow up the preliminary with a more definitive test or doctor's visit.

This will be able to provide medical information and treatment to people without the doctors that can diagnose their specific illnesses. A simple internet connection is all that will be needed to provide help to the people most in need. It can also help divert trivial illnesses from doctors by providing safe, at-home remedies and allowing people with more pressing illnesses to see a doctor sooner.

II. Datasets

The datasets that we will be using include samples of a multitude of diagnosed conditions and the symptoms that were experienced before the diagnosis. This dataset includes about 9 features of symptoms per diagnosis with about 5000 data points. Each unique illness has 120 data points, resulting in a very balanced dataset. This dataset also contains data with a patient's reported severity of symptoms as well [3]. A second dataset that we are using includes very similar data, but also includes common medications, as well as common tests to detect such an illness. With these two datasets in tandem, we want to create a model to diagnose and provide advice to patients at risk for the diseases we cover.

Most of our data will be sourced from kaggle.com and other data repositories as we are not able to collect our own data on the problem. These datasets will be named and linked within the references section at the end of this report.

III. Methodology

Initially, we will be training a decision tree to attempt to classify each disease based on symptoms and severity. However, due to the nature of the problem a Naive Bayesian model, neural network, non-linear SVG, random forest, or K-nearest neighbor may suffice as well.

Analytics Vidhya in their article discusses the situations in which a Random Forest model application is more suited than a Decision Tree: "Random Forest is suitable for situations when we have a large dataset, and interpretability is not a major concern". Towardsdatascience.com moves towards a similar conclusion in their article called "Why Random Forests Outperform Decision Trees" by calling Random Forest "more accurate than a single Decision Tree". It seems to be a trend of websites and users favoring the use of a Random Forest application because real-world applicable scenarios are higher stakes and want more accurate results. Unsurprising enough, our project falls into a similar category. We found Random

Forest on its own after multiple testing phases to be the best practice given our domain. In our results/predictions, we need total accuracy over a long period of time with a larger dataset. ”Interpretability is not a major concern” and will never be in our project because of what we are trying to do The only number that we care about is that accuracy percentage. Trying to understand the causes and effects certain data has on our model can lead us to dead ends in the field of medicine. We are less concerned with how the system works and more concerned with what the system returns to us. As an example in the article titled ”Why random forests outperform decision trees,” there is a GIF of a graph with a decision boundary perfectly splitting the data. It proves the accuracy increase of stability as more trees are added to the boundary.

To make our data operable for a decision tree, we will be using one-hot-encoding. Encoding each type of symptom in an array and using a positive value in a single position to differentiate between each symptom.

In the goal to diagnose diseases, false positives are a safer option than false negatives. A false negative could lead to a sense of security and negligence to illnesses that may need attention immediately. To reduce the number of false negatives given by the model, we have decided to maximize recall rather than accuracy or precision. We may even use a weighted metric with accuracy and recall selected.

We will be using a multi-class classification model, so each disease will be assumed to be mutually exclusive. This may not be the case in the real world, however, to keep results simple and readable we will assume that the best match of the symptoms will be the proposed rather than all of the diseases they apply to. If we were to give all the applicable possible diseases that would also be a slight burden on the user. Add that to our false positive over false negative mentality when building our model and soon all our results would be scaring away users. We want our model to return the least likely disease/illness and that is it. If there are other diseases very close in probability we can have a flag to also look out for that disease. Again, the model is purely for the user’s knowledge and to make checking for these illnesses/diseases more attainable, it is not an end decision. Licensed professionals make that call.

IV. Results

We tested a multitude of models to find the best fit for our data. This included: Decision tree, K-Nearest Neighbor, Neural Network, Random Forest, and a Support Vector Machine. The K-Nearest Neighbor and SVM were both tested with One vs One Ensemble models and One vs Rest models in addition to their single counterparts. Each of these models performed very well on our data.

The Decision Tree Performed very well using the entropy criterion, getting near perfect predictions, but the single decision tree is prone to biases in the data. Which lead us to try the Random Forest as well which performed even better, consistently scoring 100% in cross validation to ensure that this was no error, we decreased the training size to only 40

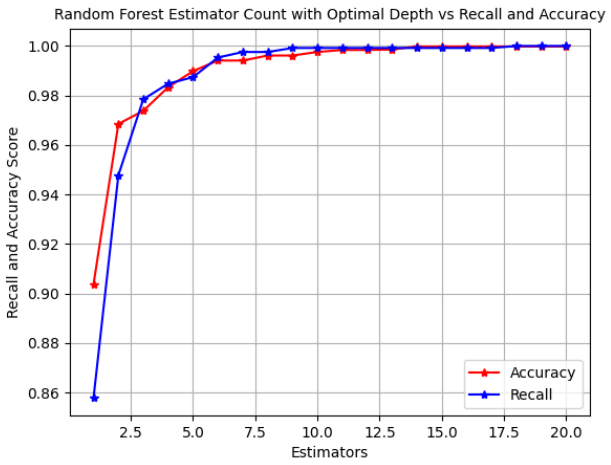

Fig. 1. Estimators vs Recall and Accuracy

samples, the model can still predict at 93% accuracy. This well performing random forest was using 14 estimators and a depth of 16 each and building its estimators with the logarithmic loss criterion. The K-Nearest Neighbors had very similar progress, predicting very well once the data had been pre-processed and encoded, hitting 99% and 100% once optimized with 2 to 3 . The Neural Network had the most issues originally mostly to do with the format of our data, but after proper encoding and a parameter sweep of the various activation functions and learning rates, it also began to achieve the 99th and 100th percentile. The Neural Net seemed to enjoy a hyperbolic tangent activation function, stochastic gradient-based optimizer to adjust the weights and an adaptive learning rate. The SVM that we used also scored very high when cross validated, sitting at the 99th and 100th percentiles. This model seemed to perform the best using a radial basis function kernel to transform the data, and a penalty of 3 for interfering points.

V. Concerns

The abundance of high scores in this project was extremely worrying to us. As high scores are normally a sign of overfitting a model. We have a few ideas on why this pattern arose across multiple models and various training sizes. The data that we used was from Kaggle, a site known for its easy and workable data to learn with. This data was superb and plentiful boasting 5000 data points of perfectly formatted and clean data. However, it was absent of much of the noise that comes with less refined data. We believe that this allowed our models to be so consistently accurate. Without more data, we are not able to test the robustness of this data in real world scenarios. Despite all of this, there is no doubting that the research done here is valuable information for the field and future machine learning models that set out to do the same task.

VI. Related Works

That being established, it is clear that the Random Forest model is popular among medical-related domains. After doing

some preliminary research, that became all the more true. Especially specific to disease/symptom-based predictions. Here are three similar projects/papers to ours that utilize a different but practical methodology.

Similar to our project, ScienceDirect has a similar paper on a "Random Forest-based predictor for Medical Data Classification Using Feature Ranking" by Zahangir Alam, Saifur Rahman, and Sohel Rahman [1]. The paper discusses using a feature ranking approach to filter the highly ranked features for the final implementation of the Random Forest classifier. There was a total of ten data sets used and they were able to get ten very accurate reads for the ten different diseases tested. That is the main difference of this paper from ours. They used a multitude of feature ranking strategies before running the final classification of the Random Forest. They claim this methodology can be used in any other domain that "exhibits similar characteristics of features". Their data was obtained from UCI's Machine Learning Repository. As for their model construction, they started with a ranker algorithm. That would determine their feature ranking which goes into feature selection. Upon removing all values of unranked features, the classification models were run. It is important to note that during classification there was also a ten-fold cross-validation, much like our approach in six separate groups. Their results consisted of Diabetes, Bupa, Hepatitis, Heart Diseases, Parkinsons', etc. The paper highly recommends using ranker algorithms before using a final classifier, which is something we did not do.

Another paper that uses the Random Forest method in healthcare was done by Snehith Raja, M. Anurag, Ch. Prachetan Reddy, and Nageswara Rao Sirisala. Titled, "Machine Learning Based Heart Disease Prediction System" [4], this paper discusses the epidemic of heart attack deaths worldwide, after touching on the idea that if found early it is possible to have better treatment for preventative measures. The model uses old patient records in the form of CSV files. Using their data and a multitude of classification models/algorithms, the group had the goal in mind of finding a near-perfect model. The final three of an original pool of six-seven classifiers were ID3, Decision Tree, and Random Forest. In that order from least to most accurate, Random Forest stood with 99.67, 99.87, and a surprising 100 percent. The goal of the paper was similar to ours, in the sense of streamlining better access to medical attention. By entering the reported value by the user they can get an accurate reading and proceed to get a consultation if need be.

The final paper we will discuss is one that isn't human-related but is still disease-related. This paper titled, " Plant Diseases Detection and Classification using Machine Learning Models: by Poojan Panchal, Vignesh Charan Raman, and Shamla Mantri [2] discusses the problem of getting an expert's opinion on physical analysis. In this work, they use two different classifiers, the K-means clustering and HSV-dependent classification for "recognizing infected part of the leaf and feature extraction using GLCM" otherwise known as Gray Level Co-occurrence Matrix. Final classification was tested with four different approaches: Random Forest, Decision Tree, KNN, and SVM. Again, much like our project, the Random Forest classifier was the best. They tested their classifiers on four different categories: Bacterial Spot, Early Blight, Late Blight, and Normal. For Early Blight, the Random Forest classifier was 100 percent accurate.

After delving into the different approaches groups take in order for quick and accurate models regarding healthcare, it is apparent that the Random Forest classifiers are typically the preeminent choice. This is due to their ability to make quick and accurate decisions from large data sets, which anything disease related is always going to have. Random Forest is simply a good algorithm as its robust, resistant to outliers, handles high dimensional data, avoids overfitting, and is capable of handling class imbalance. All of these traits make it an almost always ideal fit for any model in the realm of healthcare. At the very least the Random Forest classification method is used in my final classifications to fine-tune the results.

VII. Conclusion

The final model that resulted in the highest cross-validation recall and accuracy was the Random Forest ensemble classifier. This model was chosen not only for its scores, but for its reputation for robustness in the face of noise and outliers, which would be common in the medical field, especially with patients self reporting symptoms. Our model seemed to perform the best with 14 estimators and a depth of 16 each. The estimators were built using the logarithmic loss criterion. With this combination of methods, the final model, across 6 folds in cross-validation, reached an average recall of 100%. After classifying a particular disease, a recommendation can be given on treatment options and follow-up protocols. We hope that this exploration into the medical field with machine learning can provide insight for other researchers and analysts looking to perform this on a much larger scale that we did.

References

[1] Md. Zahangir Alam, M. Saifur Rahman, and M. Sohel Rahman. A random forest based predictor for medical data classification using feature ranking. *Informatics in Medicine Unlocked*, 15:100180, 2019.

[2] Poojan Panchal, Vignesh Charan Raman, and Shamla Mantri. Plant diseases detection and classification using machine learning models. In *2019 4th International Conference on Computational Systems and Information Technology for Sustainable Solution (CSITSS)*, pages 1–6, 2019.

[3] Pranay Patil. Disease symptom prediction. https://www.kaggle.com/datasets/itachi9604/disease-symptom-description-dataset, May 2020.

[4] M. Snehith Raja, M. Anurag, Ch. Prachetan Reddy, and Nageswara Rao Sirisala. Machine learning based heart disease prediction system. In *2021 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1–5, 2021.