islington college

(इस्लिङ्टन कलेज)

Module Code & Module Title

CS6P05 Final Year Project Computing

Autumn 2019

E-Checkup

Final Year Project (Proposal)

Rishav Dahal

Group: C8

London met ID: 17031113

External Supervisor: Ishwor Shrestha

Internal Supervisor: Subeksha Shrestha

Submission: 22nd November

Word Count: 1722

Contents

Introduction

My idea for Final Year project is E-Checkup. It is a ML program that analyzes the symptoms provided by the user to predict the probable disease. The aim of this project is to use the dataset and apply machine learning techniques to predict the type of disease based on the symptoms.

Problem statement

When we type in the basic symptoms that we might be observing for few days in Google or Yahoo or any other search engines, we often get through unnecessary conclusions or predictions that mostly ends in death. Search engines are vague and provide both authorized and unauthorized results.

E-Checkup is intended to solve that particular problem by pre-analyzing the symptoms to conclude the possibility of one or more particular disease so that we can proceed into further treatment consulting medical professionals.

Aims and Objective

- This project is based on Deep Learning.
- Its aim is to get certain symptoms from user and classify them into common disease with minimal error rate.
- The main objective is to incorporate the deep learning algorithms to access data and derive a meaning.
- To know where your symptoms are leading to and proceed with the necessary checkups and meet the specific medical professionals.

Project outcome and deliverables

- The dataset for this project is taken from a study conducted at Columbia University.
- There are three conclusions drawn using three algorithms: Decision Tree, Random Forest and Naïve Bayes.
- Tkinter is used for the GUI design.
- This helps to improve efficiency and minimize error of the program.

Risk and Contingency

- Major challenge is to obtain proper outcomes based on multidimensional user symptoms.
- Limitation is that this project only predicts diseases limited on the dataset.
- Dataset can be further appended and machine can be trained to add other predictions.

Resource Requirement and Implementation:

The basic requirement of this project is a computer system which allows python applications to run. This project is based up on Python3, so the computer must support Python program to be executed successfully. Data set and training of the model is also major requirement of this project. The data set is divided into two parts: Training data set and Testing data set. The training data set is used to train the model while the testing data set is used to test the model. The accuracy obtained is approximately 95%.

- Functional requirements of E-Checkup can be:
    - The User should open the web application
    - The system should ask for the Name of the user and 5 symptoms as input
    - The user should click on predictions to create 3 predictions based on 3 different algorithms
    - The user should be able to understand the predictions and understand them for further processing and also save the data to save the user info in the system.
    - Python3 and libraries as numpy, tkinter and pandas.

Similarly, some of the Non-functional requirements are:

- The data must be in CSV format to train the model
- The format of data must be in correct order as it is places
- The GUI must be designed correctly to provide correct output

The Implementation of this project had all to deal with mapping user inputs i.e. symptoms into probable disease with an acceptable error rate. The Classification method ML was used including three different algorithms: Decision Tree, Random Forest and Naive Bayes. These all algorithms have their different complexity measure and accuracy level. Working with three algorithms

helped to compare the outputs on a given set of input. The main challenge was to obtain accurate result based on the trained model using the data set.

A **Decision Tree** is a map of the possible outcomes of a series of related choices. It allows an individual or organization to weigh possible actions against one another based on their costs, probabilities, and benefits. They can be used either to drive informal discussion or to map out an algorithm that predicts the best choice mathematically.

The **Random Forest** algorithm combines multiple algorithm of the same type i.e. multiple decision trees, resulting in a forest of trees, hence the name "Random Forest". The following are the basic steps involved in performing the random forest algorithm:

1. Pick N random records from the dataset.
2. Build a decision tree based on these N records.
3. Choose the number of trees you want in your algorithm and repeat steps 1 and 2.
4. In case of a regression problem, for a new record, each tree in the forest predicts a value for Y (output). The final value can be calculated by taking the average of all the values predicted by all the trees in forest. Or, in case of a classification problem, each tree in the forest predicts the category to which the new record belongs. Finally, the new record is assigned to the category that wins the majority vote. (Ponit, 2015)

A **Naive Bayes** classifier is an algorithm that uses Bayes theorem to classify objects. Naive Bayes classifiers assume strong, or naive, independence between attributes of data points. Popular uses of naive Bayes classifiers include spam filters, text analysis and medical diagnosis.

A naive Bayes classifier is not a single algorithm, but a family of machine learning algorithms that make uses of statistical independence. These algorithms are relatively easy to write and run more efficiently than more complex Bayes algorithms. The formula used for naive Bayes is:

$$P(c|x) = (P(x|c) * P(c)) / P(x) \text{ (machinelearningplus, 2016)}$$

These three algorithms are implemented in the project with their functions to predict the disease.

Resources to complete the project

Software	Reasons
Adobe Photoshop	For graphical design and visual illustration
Jupyter Notebook	For editing code
MySQL	For database interaction
MS Word	For planning, management and visual illustration

Hardware	Reasons
PC	For testing the web client

Methodology

I have chosen to utilize iterative model for my project. My tasks don't endeavor to begin with full particular of necessities. Rather improvement starts by determining and execute portions of the application. Every one of web customers will be assessed so as to improve and recognize further prerequisites in each stage. The procedure is then repeated, creating better server and the customers. My undertaking is evaluated to be done in 24 weeks.

(Klinger, n.d.)
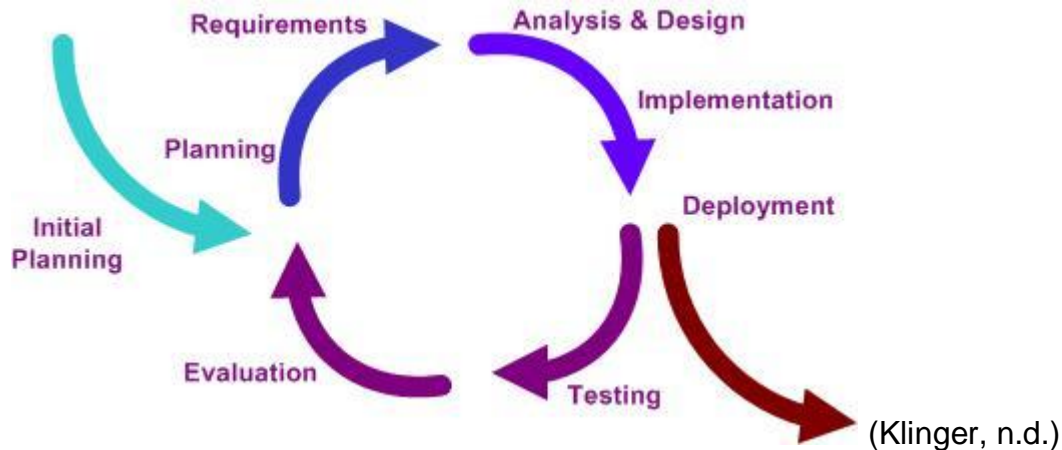
Considering an iterative life cycle model, it consists of repeating the following four phases:

- Requirements phase
- Design phase
- Implementation Phase
- Review Phase

Initial Planning

In this stage, all the arranging of the project is finished. Initially, I will make arrangements and guidelines to finish my activities in due time. I will set time period for each piece of the tasks. I will recognize my weakness in topic and dispense more opportunity to improve it. I will likewise investigate about the subjects required for the application advancement. I will also research about the algorithm used for analysis of symptoms. (Solutions, n.d.)

Requirement Phase

In this stage, requirements for the software are accumulated and broke down. I will begin with essential requirements for web customer. I will begin by coding straightforward working code which can recognize diseases. With every Iteration requirement will help to grow bigger that give more highlights.

Design Phase

This stage starts with basic program plan and database structure. UI and UX design of server and customers is executed when the necessity is made. In this stage, requirements are outlined in configuration structure. This might be another program structure, or an augmentation of a prior plan. Every cycle stage will give reliable and straightforward plan.

Implementation

In this stage, execution of design and requirement are finished by coding. Most importantly, server is composed utilizing Python. Web customers are composed when the requirement and design is made. Testing is additionally performed is this stage. In this stage customers are incorporated with the server and tested. With every cycle, server and customers are made bug-less and stable.

Review Phase

A Review stage begins by assessing the server and customers written in execution stage. This phase also reviewed the current requirements that is weather the feature made in requirement phase is implemented or not. If the feature are advanced and is not possible to implement, then the requirements are changed. Nature of code is additionally assessed in this stage. Iteration makes the survey increasingly powerful and productive.

Deployment Phase

This is the final steps where the final output of a project can be find. In this phase, users can find the stable and final server and client that can be run in their devices.

Cycle Diagram:
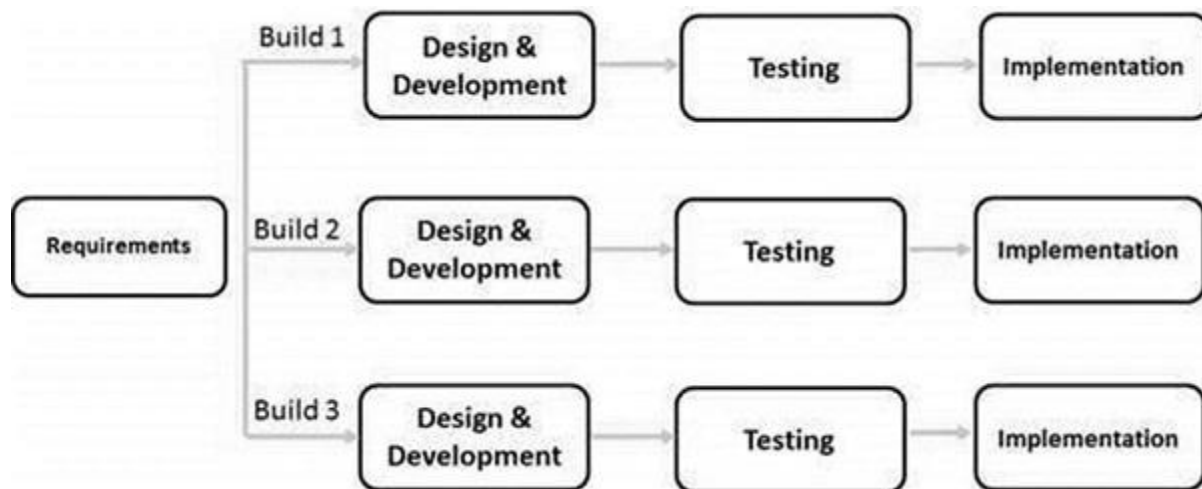
Conclusion:

E-Checkup is a Disease prediction symptom based on ML classification technique. This project is developed for personal use and creates no medical evidences. It is not a substitute for professional medical advice, diagnosis or treatment. It is like a pre-requisite for medical

professional. This project measures an accuracy of around 95% with very less error rate. Going through this project, one can know about where their symptoms are taking them towards. The three algorithms are compared and there different predictions which include two normal predictions and one final prediction.

Also, there is a data saving option which allows user to save data and store in a file including information as Name of the user, symptoms and predictions that are made. In this way, many other predictions can be extracted from the project and we can be aware of our probable diseases.

Bibliography

Klinger, C. (n.d.) *Iterative Software Development Approact* [Online]. Available from: https://wiki.nci.nih.gov/display/CommonProjects/Iterative+Software+Development+Approach [Accessed 05 April 2017].

machinelearningplus. (2016) *machinelearningplus* [Online]. Available from: https://www.machinelearningplus.com/predictive-modeling/how-naive-bayes-algorithm-works-with-example-and-full-code/ [Accessed 2019].

Ponit, T. (2015) *Tutorials Ponit* [Online]. Available from: https://www.tutorialspoint.com/r/r_random_forest.htm [Accessed 2019].

Solutions, B. (n.d.) *What is a software development process* [Online]. Available from: http://www.selectbs.com/analysis-and-design/what-is-a-software-development-process [Accessed 05 April 2017].

studentprojectguide. (2018) *studentprojectguide* [Online]. Available from: https://www.studentprojectguide.com/php/online-hospital-management-system/.

Tyagi, G. (n.d.) *6 stages of software development process* [Online]. Available from: https://www.synapseindia.com/6-stages-of-software-development-process/141 [Accessed 05 April 2017].