Machine learning based mental health disorder diagnosis

A Dissertation Submitted for the Degree of Master of Computer Science

D. D. K. M. Gunathilaka

University of Colombo School of Computing

2021

UCSC

DECLARATION

I hereby declare that the thesis is my original work, and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis. This thesis has also not been submitted for any degree in any university previously.

Student Name: D. D. K. M. Gunathilaka

Registration Number: 2018mcs025

Index Number: 18440253

____

Signature of the Student & Date

This is to certify that this thesis is based on the work of Mr. /Ms. ________________________ under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Certified by,

Supervisor Name:

____

29/11/2021

Signature of the Supervisor & Date

I would like to dedicate this thesis to

my beloved parents,

my wife,

who encouraged me to be the man who I am today,

&

academic and non-academic staff

of

University of Colombo School of Computing,

who educated me and enabled me

to reach at this level.

ACKNOWLEDGEMENTS

I would like to take this opportunity to express my sincere gratitude to my research supervisor Dr. Ajantha Atukorale, Head of the Department - Department of Computation and Intelligent Systems of the University of Colombo, School of Computing, for guiding me throughout this research project, whom without, this effort has not been possible and fruitful. Finally, I am forever grateful towards all the academic staff for their immense efforts on delivering knowledge throughout these two years.

ABSTRACT

In the present society we live in, most people suffer from mental health disorders like depression, anxiety, addictions, loneliness, stress etc. due to various reasons such as complex and competitive lifestyles, personal problems, insecurities, genetic or other illnesses. Even though it’s burning issue among the society, most people do not like to talk about their mental health problems, share them with counsellors or get treatments due to various reasons like high consultation cost, busy lifestyle, ignorance, or even some people are ashamed of the situation, thinking it is a cause of public humiliation or harm for their social status etc. Because of these above-mentioned reasons, people tend to suffer alone or even ended up committing suicide. Some people with the access and exposure of internet tend to search their symptoms and try to find answers on the internet. But people cannot always trust and rely on the information they found on the internet. There is a problem of confidentiality of the data that submitted by the users for these web sites and there are problems of accuracy and reliability of the information that users received by these web sites. There arises a problem of lack of accurate and reliable digital mental health diagnosis platform that people can use prior seeking professional help. According to literature on mental health disorder diagnosis, Mental health disorders can be traced back to simple behavioral symptoms like feeling nervous, tired, stressful, panic, sweating, overreacting, having nightmares, having suicidal thoughts etc. These behavioral symptoms can be easily identified, analyzed, and can be used as a base to determine whether a person need a professional assistance or not. This research is focused on giving user an accurate and reliable information on their mental health status based on a survey consist with above mentioned behavioral measures and symptoms with the help of machine learning and classification techniques such as Linear Multiclass Classification, Neural Networks, Naïve Bayes, Decision Trees and Decision Forest. The main goal of this paper would be to suggest a system which can accurately classify whether the user have normal mental health condition, or whether they belong to any category of mental health disorder such as anxiety, depression, loneliness, or stress based on behavioral measures and finally suggest them further steps needed to be taken. Finally, this research will critically evaluate correlation of these behavioral symptoms with demographical factors such as age, gender, race, education level, residential area of users etc. and cross validate the accuracy and relevance of the trained model within the Sri Lankan context with the help of opinions of mental health experts, psychiatrist within Sri Lanka.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Motivation

In the present society we live in, most people suffer from mental health disorders like depression, anxiety, eating disorders, personality disorders, addictions, loneliness, or stress due to complex and competitive lifestyles, personal problems, insecurities, genetic or other illnesses. According to statistics of World Health Organization, in 2017, estimated total cases suffering from depression in the world estimated to be 4.4% of global population (WHO, 2017). According to weekly epidemiological report published by Ministry of Health, Nutrition, and Indigenous Medicine in Sri Lanka, estimated total cases suffering from depression is 802,321 and its 4.1% of total population in Sri Lanka (Epid.gov.lk, 2017). From that amount majority is among females and within older age groups.

Even though mental health disorders like depression are burning issues among the society, most people do not like to talk about their mental health problems, share them with counsellors or get treatments due to various reasons like high consultation cost, busy lifestyle, ignorance, or even some people are ashamed of the situation, thinking it is a cause of public humiliation or harm for their social status etc. Because of these above-mentioned reasons, people tend to suffer alone or even ended up committing suicide. But some people with the access and exposure of internet tend to search their symptoms and try to find answers on the internet. But there are few downfalls are there as well. People cannot always trust and rely on the information they found on the internet. There is a problem of confidentiality of the data that submitted by the user for particular site and there are problems of accuracy and reliability of the information that received by the user from these sites.

1.2 Statement of the problem

Lack of accurate and reliable digital mental health diagnosis platform that people can use prior seeking professional help.

This research is focused on giving user an accurate and reliable information on their mental health status based on a survey consist with behavioral measures and symptoms which can identify mental health disorders maintaining their privacy, classify whether they have normal mental health condition, or whether they belong to any category of mental health disorder such as anxiety, depression, loneliness, or stress, suggest them further steps needed to be taken. So, they can get the professional help that they needed.

According to research on mental health, mental health disorders can be traced back to simple behavioral symptoms like feeling nervous, tired, stressful, panic, breathing rapidly, sweating, overreacting, having nightmares, having suicidal thoughts etc. (Harvey, M., Luiselli, J. and Wong, S., 2009), (Carleton, R et al., 2017). These behavioral symptoms can be easily identified, analyzed, and can be used as a base to determine whether a person need a professional assistance or not.

1.3 Research Aims and Objectives

This research is focused on identifying set of generic behavioral symptoms which can clearly classify mental health status, develop a predictive model, and web based application which can predict user's health status.

1.3.1 Aim

Aim of this research is to give user an accurate and reliable information on their mental health status based on a survey consist with behavioral measures and symptoms which can identify selected set of mental health disorders, classify whether they have normal mental health condition, or whether they belong to any category of mental health disorder such as anxiety, depression, loneliness, or stress and finally suggest them further steps needed to be taken.

1.3.2 Objectives

- Identify set of mental health statuses that predictive model going to support.
- Identify set of features/ behavioral symptoms that can clearly classify mental health status.
- Design a framework that can process large datasets, incorporate the data analytics and classification techniques to generate a predictive model that can be used for mental health disorder diagnosis.
- Build a prototype web application which consist with survey of behavioral measures and uses this predictive model, that can be used by end users to get an insight on their mental health.

1.4 Scope

- There will be in-depth literature review carried out on mental health disorder diagnosis using behavioral measures and data analytics techniques that can be used to derive an accurate predictive model.
- Mental health disorder classification types scoped down to five most commonly occurred categories which are Normal, Anxiety, Depression, Loneliness and Stress. Other types like eating disorders, personality disorders, addictions will be out of scope and will not be considered.
- There can be various behavioral measures\ symptoms that can identify mental issues such as feeling nervous, tired, stressful, panic, breathing rapidly, sweating, overreacting, having nightmares, having suicidal thoughts etc. Only limited set of features will be selected based on literature review and dataset availability.
- User selection of initial predictive model will support up to limited demographics due to behavioral differences of people in different age groups, people residing in different countries.

1.5 Structure of the Thesis

First chapter gives a brief introduction to thesis machine learning based mental health disorder diagnosis. It describes the problem in hand, statement of problem, motivation of the research, its aim, objectives, and scope. Second chapter describes an in-depth literature review on mental health disorder diagnosis using behavioral measures and machine learning techniques that can be used to derive an accurate predictive model. It critically evaluates commercial applications currently available in mental health diagnosis industry, research carried out on machine learning based approaches in mental health disorder diagnosis, there advantages and limitations while identifying research gap. In methodology section, data set and feature selection, algorithm selection, the training process which was followed to implement a predictive model using behavioral measures and multiclass classification techniques, and the process followed to implement the prototype web application are explained. In forth chapter, evaluation process, primary evaluation metrices used, and comparison and a critical evaluation between the multiclass classification algorithms used in implementation of the predictive model are explained. In final section, conclusion of the system designed, its current achievements and the further improvements that can be done to improve the overall accuracy of the current application, its applicability to Sri Lankan context are described.

CHAPTER 2

LITERATURE REVIEW

2.1 A Literature Review

Since the beginning of 20th century, there were a dramatical change in treatments for the mental health disorders in outpatient clinics and mental hospitals by analyzing the individual's behavior. In early 20' people came up with online consultation platforms with human intervention to reach out to people whom suffering from mental health disorders. Later with the advancements of artificial intelligence and machine learning techniques, numerous research were presented related to early detection of mental health disorder with considerable developments and limitations.

Origin of online mental health counselling was assumed as a psychotherapy session using linked computers in 1972, by the staff of Stanford and UCLA. The first online mental healthcare "Dear Uncle Ezra" was established by Cornell University in 1986. It was a question and answer mechanism where people can repeatedly discuss the issues regarding mental health. Over the years, "Dear Uncle Ezra" has developed rapidly and after the two decades it has posted around twenty thousand of answers for the questions from the people worldwide. Main limitations of these sessions were that it was not clear who was answering the questions and if that therapist was a licensed therapist (Skinner, A. and Zack, J., 2004).

In 1995, John Grohol created a public mental health chat, which had ultimately developed a popular mental health psych central advertisement and convention. First mental health counselling service started by therapist Leonard Holmes where participants can donate money as an option. After that, other fee-based mental health services followed that theory but none of them are active in the present (Grohol, J., 1999).

Talkspace is an online therapy platform that instantly expands to give endless messaging therapy, begun by Roni and Oren in 2012. This system covers virtually every category in mental health. In earlier stages, the platform has given the clients a more powerful experience than exchanging emails with the counsellor. Talkspace can be used the people beyond age 13 via desktops, smartphones (both android and iPhone) and tablets. Using smartphone apps clients could send text, voice messages, and video whenever they wanted unconditionally. First to identify the clients' therapy needs, there will be a chat with a relevant therapist. So, this has overcome the main limitation of the “Dear Uncle Ezra” and clients can get treatment from licensed therapist. Secondly the client has to choose the payment plan according to his budget. Finally, it will help the client to find the best therapist and then the treatment session will begin. They have used machine learning and artificial intelligence related tools to data analysis in these sessions. The Talkspace is HIPPA (Health Insurance Portability and Accountability Act) compliant and uses SSL encryption for the protection of health records and data (Talkspace.com, 2021).

BetterHelp is another online counselling system founded by Alon Matas in 2013 and Danny Bragonier also has supported for the development. It is a web-based platform and online portal, which enables clients to associate with the counsellor through the internet in a private web-based facility like audio and video meetings, live chatting, and private messaging. This system provides live chat and video sessions via both desktop and mobile devices. The website has given end users a devoted “room” which acts as a secure and private online entrance for communication. The “room” is open 24 hours in whole week and can be accessed from any device with proper internet connection and also from any physical location. The BetterHelp has nearly 3800 licensed and professional therapists, the same as Talkspace. For security and privacy, the system uses SSL encryption, and it is also HIPPA (Health Insurance Portability and Accountability Act) compliant (Betterhelp.com, 2021).

oDoc is recently introduced Sri Lankan based online consultation platform. But it’s not only focused on mental health counseling. They provide services of over 900 SLMC (Sri Lanka Medical Council) registered doctors including dermatologists, cardiologists, and specialists for mental health disorders such as depression. It is a web and mobile based consultation system which provides facilities to channel doctors through audio and video appointments and get medical advice. They also provide other third-party facilities like home visit laboratory tests and medicine deliveries (oDoc.life, 2021).

These web-based consultation applications are convenient, and they are great platforms for people who are comfortable talking about things out loud. But they all have some common drawbacks as well. Sometimes, people hesitate to go directly to therapy or counseling session without a self-evaluation first. They actually need to come into a realization that they need professional help prior, directly going to any of above-mentioned platforms and seek help. Sometimes, suggested therapist may not be the best fit for you.

Even though human touch is an essential in mental health diagnosis, some people do not easily open up and share their problems due to reasons like shyness, ignorance, counselling process didn't feel right, or communication styles were off etc.

Therefore, there should be an early detection solution to address those issues without human intervention, and just by filling out a survey trained using machine learning techniques. It should be able to classify whether they have normal mental health condition, or whether they belong to any category of mental health disorder such as anxiety, depression, loneliness, or stress and finally suggest them further steps needed to be taken. So, they can get the professional help that they needed. There was multiple research carried out on machine learning based approaches to early detect different types of mental health disorders such as anxiety disorders, eating disorders, post-traumatic stress disorder and depression etc.

"Screening for major depressive disorder in a tertiary mental health center using Early Detect: A machine learning-based pilot study" was a recent research carried out on early detection of major depressive disorder. Research was conducted in a single tertiary mental health center in western Canada using questionnaires, face to face assessments and interviews. They have used machine learning algorithms to analyze data, detect patterns and make more confident predictions on patient's health status. Even though they were able to identify key predictive factors (like family history of mental illness, stressful events happened) and make more accurate predictions, those results were subjective to that specific center. As main limitations of this research they have identified that, Patients were assessed using partially improvised psychiatric evaluations, which was subjective to each patient and there was potential bias for self-reporting particularly with depressed patients due to negative thinking (Liu, Y., Hankey, J., Cao, B. and Chokka, P., 2021).

"Machine learning-based discrimination of panic disorder from other anxiety disorders" is another research conducted by Department of Psychiatry, Gachon University College of Medicine, Republic of Korea. Research data was collected from medical charts recorded between 2011 January and 2017 December. The inclusion criteria were patients aged between 20 - 65 years, diagnosed with a panic disorder. Training set and test set were taken as 70% and 30% and multiple machine learning algorithms such as artificial neural network, logistic regression, random forest, gradient boosting machine, and support vector machine were applied. The primary measurement of performance of the trained models was overall accuracy. Specificity, Sensitivity, F1-score, and Matthew's correlation coefficient were used as secondary measures. As per limitations of this research they have identified the limited sample size and cross-sectional design which might cause overfitting (Na, K., Cho, S. and Cho, S., 2020).

"Machine Learning Based Diagnosis of Binge Eating Disorder Using EEG Recordings" is a research conducted by Dominik Raab, Hermann Baumgartl and Ricardo Buettner from Aalen university, Germany. As data set, they have used an Electroencephalography (EEG) publicly available dataset provided by Max Planck Institute Leipzig, consist of 203 healthy participants in total including their psychological assessment through their completion of cognitive tests or questionnaires covering several mental behaviors like eating disorder. They have used random forest and K-fold cross validation where k set to 10 as the machine learning techniques for their classification. Limitations they faced included limited sample size, lack of a clear identification within the provided dataset as to whether a participant is affected by a Binge Eating Disorder or not, noises generated in EEG dataset such as line noises, radio, and electrical interference. The biggest limitation in that study was that they only choose explicitly healthy individuals as participants in the study which leads their study less reliable, lack of robustness and overfitted to that dataset (Raab, D., Baumgartl, H. and Buettner, R., 2020).

2.2 Research Gap

Following gaps were identified while going through literature on machine learning based mental health disorder diagnosis and their applications. There are couple of market leading online consultation platforms like BetterHelp, Talkspace and oDoc etc. They still provide consultation facilities on time-based subscriptions through audio/video appointments and they are yet to adopt machine learning and AI technologies. There is a huge potential to grow machine learning based applications in these consultation platforms.

It was identified that researchers in the field were able to discriminate patients specific to certain disorder such as diagnosis of Binge eating disorder using Electroencephalography recordings, discrimination of panic disorder from other mental health disorders etc. Limited number of patients in focused data samples used for classification process, potential bias, or overfitting towards selected patient data set (explicitly healthy or extremely biased towards specific to certain disorder), and limitations in data collection processes/ selection criteria were another major research gaps found in above literature.

Even though researchers were able to come up with classification solutions specific to certain mental health disorder, they were not yet able come up with generic classification solution where they can classify users among set of selected general mental health disorders. According to literature on mental health, mental health disorders have strong correlation with demographical factors such as age, gender, race, education level, residential area etc. as well as behavioral symptoms such as rapid breathing, sweating, feeling angry and tired etc. But most of the literature found were focused on behavioral or physical symptoms without considering demographics factors. And there was less research carried out focusing mental health patients in Sri Lankan context.

This research tries to evaluate behavioral factors and demographical factors that can used as basis of mental health disorder classification and come up with a general solution for selected set of mental health disorders. This research is focused on addressing that gap and provide reliable digital mental health disorder diagnosis platform that people can use prior seeking professional health.

CHAPTER 3

METHODOLOGY

In this section, the process, which was followed to feature selection, dataset selection, algorithm selection, high-level architecture of the mental health disorder diagnosis application and its training process and implementation are explained.

This research tries to evaluate behavioral factors, physical factors and demographical factors that can used as basis of mental health disorder classification and come up with a general solution for selected set of mental health disorders.

3.1 Feature Selection

According to research on mental health, mental health disorders can be categorized into different significant categories such as normal, anxiety disorder, depression disorder, loneliness, post-traumatic stress disorder, eating disorder, addictions etc. As per the scope of this project and to reduce the classification complexity, disorder types were scoped down to five mental health disorder categories and other types were not considered in the classification process. Top five most commonly occurred mental health disorders according to World Health Organization were selected in this process. Selected mental health disorder types were,

Normal	Anxiety	Depression	Loneliness	Stress

Table 1: Selected Mental Health Disorder Types

In traditional eye to eye consultancy process, the mental health expert or psychiatrist go through series of session with the patients and try to identify symptoms and root causes for specific mental health disorder.

But these disorders can be traced back to simple behavioral symptoms like feeling nervous, tired, stressful, panic, breathing rapidly, sweating, overreacting, having nightmares, having suicidal thoughts etc. These behavioral symptoms can be easily identified, analyzed, and can be used as a base to determine whether a person need a professional assistance or not. Based on literature review, considering correlation with mental health disorder types and dataset availability on Kaggle, 24 features (behavioral symptoms) were selected and used for the classification process.

Selected behavioral symptoms were,

Feeling nervous	Having sudden panic attacks	Breathing rapidly
Sweating excessively	Having troubles in concentration	Overreacting
Having troubles with work	Feeling hopeless	Feeling angry
Having suicidal thoughts	Changes in eating habits	Feeling tired
Having a close friend	Addiction to the social media platforms	Rapid weight gain
Material possession	Experiencing emotional flashbacks of stressful memories	Feeling shy
Having nightmares	Avoiding people or activities	Feeling negative
Having troubles in sleeping	Having troubles in focusing	Blaming yourself

Table 2: Selected Behavioral Symptoms

3.2 Dataset Selection

Mental disorder symptoms dataset is a publicly available labeled data set on Kaggle. It was used as a primary source of labeled data set for this classification, and it is clearly labeled on above mentioned selected five mental health disorder types. Based on the features of this dataset, a survey was constructed to present for patients. Initially, patients or users of the system will be asked to fill out a survey consist with questions related to above mentioned behavioral symptom. In this survey they will be rated on each behavioral symptom on scale of 1 to 10 where 1 being the lowest and 10 being the highest where patients display or having these symptoms. These numeric input on each feature will be used as inputs of predictive model and user will be classified into one five mental health status based on it.

3.3 Multiclass Classification Algorithm Selection

In following section, definitions of selected multiclass classification algorithms and their features are briefly described.

3.3.1 Linear Multiclass Classification

Linear multiclass classifier trains a linear model to solve multiclass classification problems. It classifies classes based on a value which is a linear combination of characteristics or features of objects. In a multiclass classification scenario assuming that the number of classes is m and number of features is n. It assigns the c^{th} class a coefficient vector $w_c \in R^n$ and a bias $b_c \in R$, for c=1, …m. given a feature vector $x \in R^n$, the c^{th} class's score would be $y^c = w_c^T x + b_c$.

3.3.2 Naïve Bayes Multiclass Classification

Naive Bayes classification is a probabilistic approach that can be used for solving multiclass classification problems. Using Bayes' theorem, it calculates the conditional probability for a sample belonging to a class based on the sample count for each feature combination groups.

$$P(A|B) = \frac{P(B|A).P(A)}{P(B)}$$

Equation 1: Bayes Theorem

Naïve bayes algorithm assumes independence among the presence of features in a particular class even though they may be dependent on each other, converting equation into naïve and simpler.

3.3.3 Maximum Entropy Multiclass Classification

Maximum Entropy classifier also trains a linear model to solve multiclass classification problems assuming that the number of classes is *m* and number of features is *n*. It assigns the c^{th} class a coefficient vector $w_c \in R^n$ and a bias $b_c \in R$, for c=1, …m. given a feature vector $x \in R^n$, the c^{th} class's score would be $P(c|x)=e^{y\wedge c}/\sum_{c'=1}{}^{m} e^{y\wedge c'}$, where $y^{\wedge c}=w_c{}^T x+b_c$. Both Linear multiclass classifier and Maximum Entropy multiclass classifier trains a linear model where only difference is between the probability distribution function of weights distribution across the classes.

3.3.4 Multiclass Decision Forest Classification

Multiclass Decision Forest is an ensemble learning algorithm where learning process use multiple models instead of a single model. Decision forest classifier builds up multiple trees and then vote on the most popular or confident output class. Voting is an aggregation mechanism which, each tree generated in a multiclass decision forest populates a non-normalized frequency histogram of labels, then sums up the histograms and normalizes the result to get the final probabilities for each label. When assembling the final decision tree, trees which have high prediction confidence will have a greater weight. Because of that, Multiclass Decision Forest algorithm is very resilient algorithm in the presence of noisy features.

3.3.5 Multiclass Neural Network

Multiclass Neural Network is a supervised learning mechanism which consist of set of inter-connected layers. The input layer is the initial layer and it's linked to output layers by an acyclic graph consist of weighted edges and nodes. Between the input and output layers, there can have a one or more hidden layers depending on classification problem. Most of the predictive tasks can be easily achieved with a one or few hidden layers. In Multiclass Neural Network classifier, output is calculated by calculating the weighted sum of the values of each node from the previous layer and then applying an activation function on top of that weighted sum.

3.4 High-level Architecture

Following diagram shows the high-level architecture of mental health disorder diagnosis application.

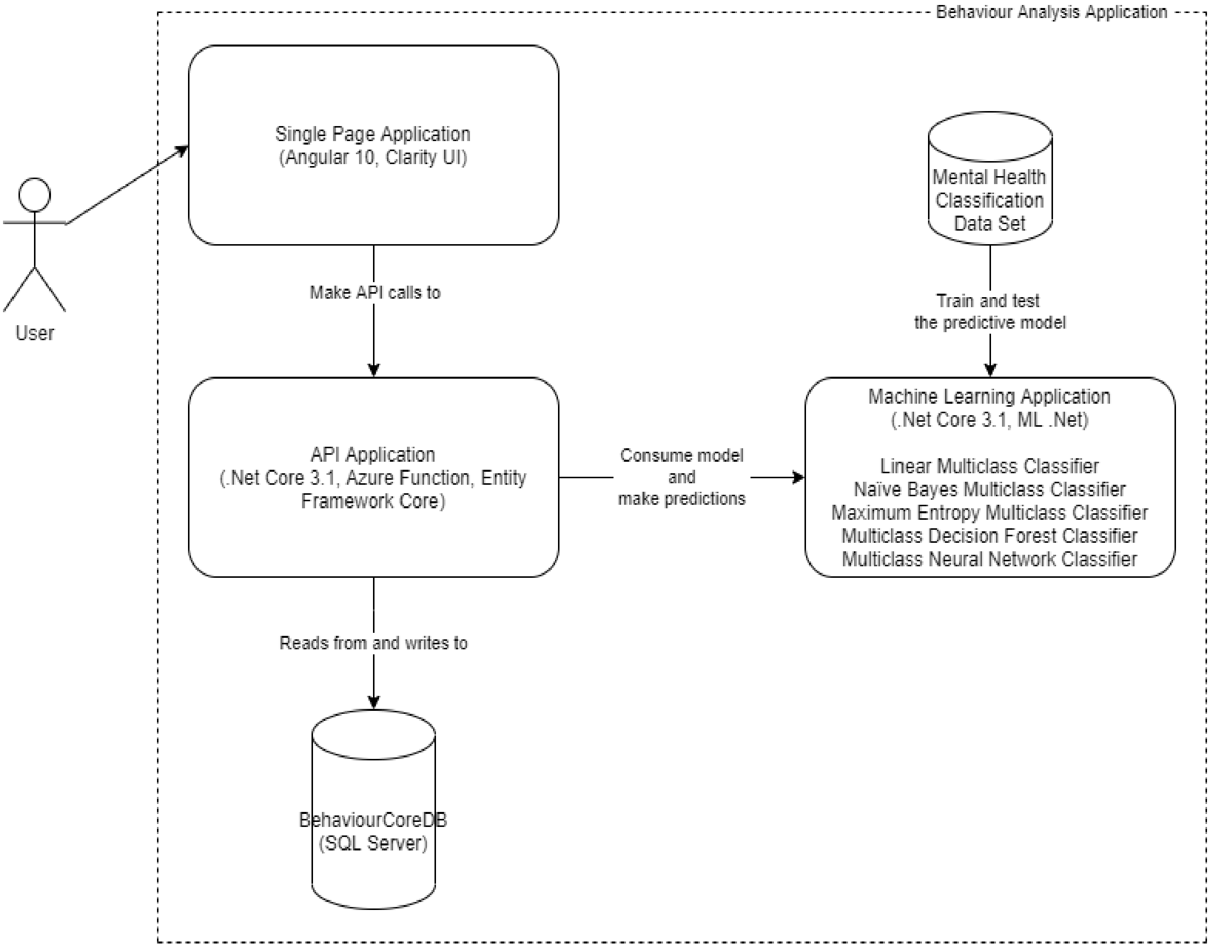


Figure 1: High-level Architecture

Prototype mental health disorder diagnosis application solution consists with three major components/ applications as shown in above high-level architecture diagram. Following is a brief explanation of each component.

1. Single page frontend application where users can interact, provide answers to surveys, and get insight on their mental health status. It was implemented using Angular and Clarity UI framework.
2. Backend API application where business logics such as survey creation, getting user inputs, passing user inputs to the predictive model, consuming model and making predictions, and saving responses etc. are handled. It was implemented using .Net Core framework and Entity framework. Relational database on Microsoft SQL server was used to persist user data such as survey responses.
3. Machine learning application that trains the predictive model which takes user survey responses as its input and predict their mental health status. Each of above-mentioned multiclass classification algorithm was implemented using ML .Net and Azure machine learning studio.

3.5 Training Process

Following diagram shows the high-level abstract view of the training process followed in the machine learning application.

As shown in below figure of training process, initial Mental Disorder Symptoms dataset which found from Kaggle was preprocessed prior using in classification process. Duplicate records and noisy tuples were removed to avoid overfitting and increase the accuracy of the model. Then it was separated (Three to One) to training set and testing set in order to evaluate and assess the accuracy of the trained predictive model.

Multiclass classification algorithms such as Linear Multiclass Classification, Naïve Bayes, Maximum Entropy, Multiclass Decision Trees and Multiclass Decision Forest and Multiclass Neural Network were implemented and trained iteratively on training data set to develop multiple predictive models, and each of them were evaluated and scored against the testing set.

In the evaluation process, macro accuracy, micro accuracy, precision, recall, log loss and log loss reduction metrics were calculated based on the confusion matrix of each trained predictive model and compared against to choose the best multiclass classification algorithm applicable to mental health disorder classification problem depend on the dataset.

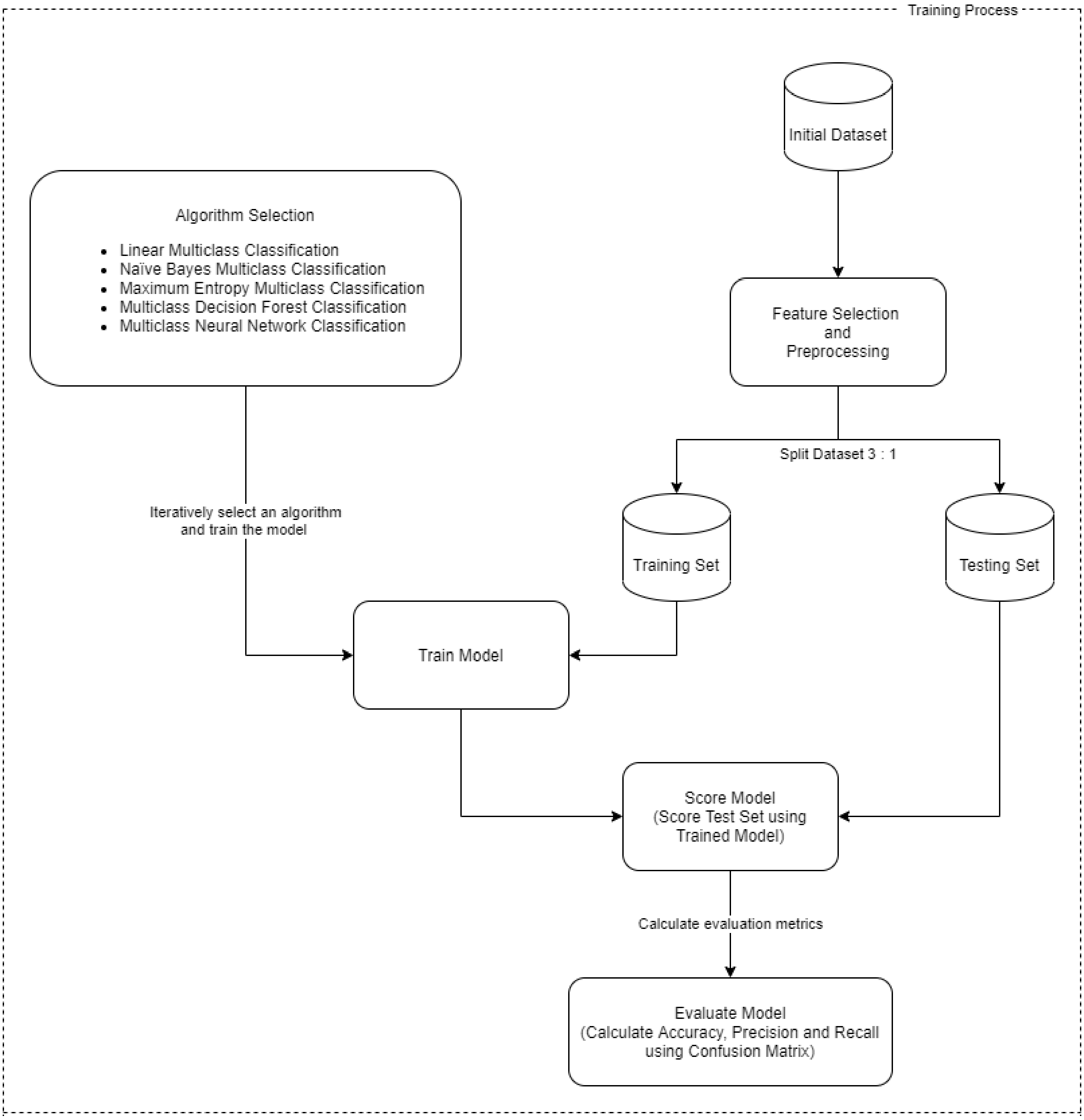


Figure 2: Training Process

3.6 Implementation

In following section, implementation process of selected multiclass classification algorithms in machine learning application is briefly described.

Implementation of machine learning application mainly consist with four steps, which starts with preprocessing and loading data, then building training pipelines and training the predictive models iteratively using selected multiclass classification algorithms, evaluating models against selected evaluation metrics and finally make prediction consuming the most accurate trained model. Multiclass classification algorithms were implemented by using ML .Net which is open-source machine learning library for C# and Python and Azure machine learning studio which is a data science platform created by Microsoft. Machine learning application will expose a trained predictive model as a model file, and it will be loaded into memory and consumed in backend application upon making predictions.

Following is a brief explanation of above-mentioned main steps of the implementation of machine learning application with their respective code snippets. Following figure of Create Model main method sums up the whole model building process, calling respective methods of building training pipeline, evaluating and cross validating, and finally saving the model.

```
1 reference | Kasun Gunathilaka, 170 days ago | 1 author, 3 changes
public static class ModelBuilder
{
    private static string TRAIN_DATA_FILEPATH = @"C:\Users\KasunG\Desktop\Behaviour Analysis\Project\behavior-analysis\BehaviourAnalysis\Resources\processed_data_training.csv";
    private static string TEST_DATA_FILEPATH = @"C:\Users\KasunG\Desktop\Behaviour Analysis\Project\behavior-analysis\BehaviourAnalysis\Resources\processed_data_testing.csv";
    private static string MODEL_FILEPATH = @"C:\Users\KasunG\Desktop\Behaviour Analysis\Project\behavior-analysis\BehaviourAnalysis\BehaviourAnalysisML.Model\MLModel.zip";
    // Create MLContext to be shared across the model creation workflow objects
    // Set a random seed for repeatable/deterministic results across multiple trainings.
    private static MLContext mlContext = new MLContext(seed: 1);

    1 reference | Kasun Gunathilaka, 170 days ago | 1 author, 2 changes
    public static void CreateModel()
    {
        // Load Data
        IDataView trainingDataView = mlContext.Data.LoadFromTextFile<ModelInput>(
                                        path: TRAIN_DATA_FILEPATH,
                                        hasHeader: true,
                                        separatorChar: ',',
                                        allowQuoting: true,
                                        allowSparse: false);

        IDataView testingDataView = mlContext.Data.LoadFromTextFile<ModelInput>(
                                        path: TEST_DATA_FILEPATH,
                                        hasHeader: true,
                                        separatorChar: ',',
                                        allowQuoting: true,
                                        allowSparse: false);

        // Build training pipeline
        IEstimator<ITransformer> trainingPipeline = BuildTrainingPipeline(mlContext);

        // Train Model
        ITransformer mlModel = TrainModel(mlContext, trainingDataView, trainingPipeline);

        // Evaluate quality of Model
        EvaluateUsingCrossValidation(mlContext, trainingDataView, trainingPipeline);
        EvaluateUsingTestData(mlContext, testingDataView, mlModel);

        // Save model
        SaveModel(mlContext, mlModel, MODEL_FILEPATH, trainingDataView.Schema);
    }
```

Figure 3: Create Model Main Method

As shown in the above figure, after preprocessing and manually removing duplicate and noisy tuples, training data set and testing dataset are loaded into memory separately. Training data set is used in training pipeline with different multiclass classification algorithms to create predictive models iteratively. Testing set is used in evaluation process to calculate primary evaluation metrics based on confusion matrix of each trained model. Once datasets are loaded, multiple training pipelines are created separately for each multiclass algorithm selected with training data set.

```
1 reference | Kasun Gunathilaka, 170 days ago | 1 author, 2 changes
public static IEstimator<ITransformer> BuildTrainingPipeline(MLContext mlContext)
{
    // Data process configuration with pipeline data transformations
    var dataProcessPipeline = mlContext.Transforms.Conversion.MapValueToKey("Disorder", "Disorder")
                              .Append(mlContext.Transforms.Concatenate("Features", new[] { "feeling_nervous", "panic", "breathing_rapidly", "sweating", "trouble_in_concentration",
                                  "having_trouble_in_sleeping", "having_trouble_with_work", "hopelessness", "anger", "over_react", "change_in_eating", "suicidal_thought",
                                  "feeling_tired", "close_friend", "social_media_addiction", "weight_gain", "material_possessions", "introvert", "popping_up_stressful_memory",
                                  "having_nightmares", "avoids_people_or_activities", "feeling_negative", "trouble_concentrating", "blamming_yourself" }))
                              .Append(mlContext.Transforms.NormalizeMinMax("Features", "Features"))
                              .AppendCacheCheckpoint(mlContext);
    // Set the training algorithm

    //Maximum Entropy Multiclass Classifier
    Console.WriteLine("=============== Maximum Entropy Multiclass Classifier ===============");
    var trainer = mlContext.MulticlassClassification.Trainers.SdcaMaximumEntropy(labelColumnName: "Disorder", featureColumnName: "Features")
                              .Append(mlContext.Transforms.Conversion.MapKeyToValue("PredictedLabel", "PredictedLabel"));

    //Linear Multiclass Classifier
    //Console.WriteLine("=============== Linear Multiclass Classifier ===============");
    //var trainer = mlContext.MulticlassClassification.Trainers.SdcaNonCalibrated(labelColumnName: "Disorder", featureColumnName: "Features")
    //                          .Append(mlContext.Transforms.Conversion.MapKeyToValue("PredictedLabel", "PredictedLabel"));

    //Maximum Entropy Multiclass Classifier trained with L-BFGS method
    //Console.WriteLine("=============== Maximum Entropy Multiclass Classifier trained with L-BFGS method ===============");
    //var trainer = mlContext.MulticlassClassification.Trainers.LbfgsMaximumEntropy(labelColumnName: "Disorder", featureColumnName: "Features")
    //                          .Append(mlContext.Transforms.Conversion.MapKeyToValue("PredictedLabel", "PredictedLabel"));

    //Naive Bayes Multiclass Classifier
    //Console.WriteLine("=============== Naive Bayes Multiclass Classifier ===============");
    //var trainer = mlContext.MulticlassClassification.Trainers.NaiveBayes(labelColumnName: "Disorder", featureColumnName: "Features")
    //                          .Append(mlContext.Transforms.Conversion.MapKeyToValue("PredictedLabel", "PredictedLabel"));

    var trainingPipeline = dataProcessPipeline.Append(trainer);

    return trainingPipeline;
}
```

Figure 4: Build Training Pipelines

In the training pipeline separate model will be trained upon feature set in training data set using multiclass classification algorithm selected. Once predictive models are created, Primary evaluation metrics such as Micro Accuracy, Macro Accuracy, Precision, Recall Log Loss, and Log Loss Reduction are calculated against the confusion matrix of each selected multiclass classification algorithm as shown in the below figure.

```
1 reference | Kasun Gunathilaka, 170 days ago | 1 author, 2 changes
public static void PrintMulticlassClassificationFoldsAverageMetrics(IEnumerable<TrainCatalogBase.CrossValidationResult<MulticlassClassificationMetrics>> crossValResults)
{
    var metricsInMultipleFolds = crossValResults.Select(r => r.Metrics);

    var microAccuracyValues = metricsInMultipleFolds.Select(m => m.MicroAccuracy);
    var microAccuracyAverage = microAccuracyValues.Average();
    var microAccuraciesStdDeviation = CalculateStandardDeviation(microAccuracyValues);
    var microAccuraciesConfidenceInterval95 = CalculateConfidenceInterval95(microAccuracyValues);

    var macroAccuracyValues = metricsInMultipleFolds.Select(m => m.MacroAccuracy);
    var macroAccuracyAverage = macroAccuracyValues.Average();
    var macroAccuraciesStdDeviation = CalculateStandardDeviation(macroAccuracyValues);
    var macroAccuraciesConfidenceInterval95 = CalculateConfidenceInterval95(macroAccuracyValues);

    var logLossValues = metricsInMultipleFolds.Select(m => m.LogLoss);
    var logLossAverage = logLossValues.Average();
    var logLossStdDeviation = CalculateStandardDeviation(logLossValues);
    var logLossConfidenceInterval95 = CalculateConfidenceInterval95(logLossValues);

    var logLossReductionValues = metricsInMultipleFolds.Select(m => m.LogLossReduction);
    var logLossReductionAverage = logLossReductionValues.Average();
    var logLossReductionStdDeviation = CalculateStandardDeviation(logLossReductionValues);
    var logLossReductionConfidenceInterval95 = CalculateConfidenceInterval95(logLossReductionValues);

    Console.WriteLine();
    Console.WriteLine($"*************************************************************************************************************");
    Console.WriteLine($"*       Metrics for Multi-class Classification model      ");
    Console.WriteLine($"*------------------------------------------------------------------------------------------------------------");
    Console.WriteLine($"*       Average MicroAccuracy:    {microAccuracyAverage:0.###}  - Standard deviation: ({microAccuraciesStdDeviation:#.###})  - Confidence Interval 95%: ({microAccuraciesConfidenceInterval95:#.###})")
    Console.WriteLine($"*       Average MacroAccuracy:    {macroAccuracyAverage:0.###}  - Standard deviation: ({macroAccuraciesStdDeviation:#.###})  - Confidence Interval 95%: ({macroAccuraciesConfidenceInterval95:#.###})")
    Console.WriteLine($"*       Average LogLoss:          {logLossAverage:#.###}  - Standard deviation: ({logLossStdDeviation:#.###})  - Confidence Interval 95%: ({logLossConfidenceInterval95:#.###})");
    Console.WriteLine($"*       Average LogLossReduction: {logLossReductionAverage:#.###}  - Standard deviation: ({logLossReductionStdDeviation:#.###})  - Confidence Interval 95%: ({logLossReductionConfidenceInterval95:#.##
    Console.WriteLine($"*************************************************************************************************************");

}

5 references | Kasun Gunathilaka, 171 days ago | 1 author, 1 change
public static double CalculateStandardDeviation(IEnumerable<double> values)
{
    double average = values.Average();
    double sumOfSquaresOfDifferences = values.Select(val => (val - average) * (val - average)).Sum();
    double standardDeviation = Math.Sqrt(sumOfSquaresOfDifferences / (values.Count() - 1));
    return standardDeviation;
}

4 references | Kasun Gunathilaka, 171 days ago | 1 author, 1 change
public static double CalculateConfidenceInterval95(IEnumerable<double> values)
{
    double confidenceInterval95 = 1.96 * CalculateStandardDeviation(values) / Math.Sqrt((values.Count() - 1));
    return confidenceInterval95;
}
```

Figure 5: Primary Evaluation Metrics Calculation

Based on the overall accuracy, best trained predictive model is selected and consumed in backend application to make prediction based on inputs provided by the user. As a further step, based on the accuracy results of multiclass classification techniques, predictive model with a single algorithm or a combination of most accurate algorithms will be used in the classification process.

```
1 reference | Kasun Gunathilaka, 170 days ago | 1 author, 2 changes
public static PredictionEngine<ModelInput, ModelOutput> CreatePredictionEngine()
{
    // Create new MLContext
    MLContext mlContext = new MLContext();

    // Load model & create prediction engine
    string modelPath = @"C:\Users\KasunG\Desktop\Behaviour Analysis\Project\behavior-analysis\BehaviourAnalysis\BehaviourAnalysisML.Model\MLModel.zip";
    ITransformer mlModel = mlContext.Model.Load(modelPath, out var modelInputSchema);
    var predEngine = mlContext.Model.CreatePredictionEngine<ModelInput, ModelOutput>(mlModel);

    return predEngine;
}
```

Figure 6: Create Prediction Engine

Once the best algorithm for the mental health disorder classification problem is selected and the predictive model is finalized, prediction engine instance will be created loading final predictive model into the memory as shown in the above figure.

```
1 reference | Kasun Gunathilaka, 171 days ago | 1 author, 1 change
public class PredictionService : IPredictionService
{
    2 references | Kasun Gunathilaka, 171 days ago | 1 author, 1 change
    public string PredictDisorder(Questionnaire questionnaire)
    {
        ModelInput sampleData = new ModelInput()
        {
            Feeling_nervous = questionnaire.Nervous,
            Panic = questionnaire.Panic,
            Breathing_rapidly = questionnaire.BreathingRapidly,
            Sweating = questionnaire.Sweating,
            Trouble_in_concentration = questionnaire.TroublesInConcentration,
            Having_trouble_in_sleeping = questionnaire.TroublesInSleeping,
            Having_trouble_with_work = questionnaire.TroublesWithWork,
            Hopelessness = questionnaire.Hopeless,
            Anger = questionnaire.Angry,
            Over_react = questionnaire.OverReacting,
            Change_in_eating = questionnaire.ChangesInEating,
            Suicidal_thought = questionnaire.SuicidalThoughts,
            Feeling_tired = questionnaire.Tired,
            Close_friend = questionnaire.CloseFriend,
            Social_media_addiction = questionnaire.SocialMediaAddiction,
            Weight_gain = questionnaire.WeightGain,
            Material_possessions = questionnaire.MaterialPossession,
            Introvert = questionnaire.Shy,
            Popping_up_stressful_memory = questionnaire.StressfulMemories,
            Having_nightmares = questionnaire.Nightmares,
            Avoids_people_or_activities = questionnaire.AvoidingPeople,
            Feeling_negative = questionnaire.NegativeThoughts,
            Trouble_concentrating = questionnaire.TroublesInConcentration,
            Blamming_yourself = questionnaire.BlamingYourself
        };

        ModelOutput predictionResult = ConsumeModel.Predict(sampleData);
        return predictionResult.Prediction;
    }
}
```

Figure 7: Consume Model

As shown in the above figure backend application consumes the trained model with questionnaire answers provided by user to predict their mental health disorder category. Finally, predicted mental health disorder will be returned to frontend application to display to the user.

CHAPTER 4

EVALUATION AND RESULTS

In this section, evaluation process, metrices used to assess the predictive model of mental health disorder diagnosis application and its result are explained.

As primary evaluation approach experiment-based evaluation is selected and as a further step it is planned to cross validate model using expert opinions to validate the accuracy of the developed model within Sri Lankan context.

4.1 Evaluation Metrices

In primary evaluation approach, selected primary evaluation metrices were confusion matrix, overall accuracy, macro accuracy, micro accuracy, precision, recall, log loss and log loss reduction. In following section, definitions of selected evaluation metrices is briefly explained.

4.1.1 Confusion Matrix

Confusion matrix is a N-by-N matrix, which is a tabular representation of actual values vs model predictions. Each column, row is dedicated to one class. Using this confusion matrix, we can calculate important metrices such as Accuracy, Precision and Recall. Following is a representation of 2x2 confusion matrix. It can be extended by number of classes for multiclass classification scenario.

		Predicted Class	
		Positive (P)	Negative (N)
Actual Class	Positive (P)	True Positives (TP)	False Negatives (FN)
	Negative (N)	False Positives (FP)	True Negatives (TN)

4.1.2 Accuracy

Accuracy of a trained model is calculated as number of correct predictions (total number of true positives and true negatives) divided by the total number of predictions.

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ samples}$$

$$Accuracy = \frac{(True\ Positives + True\ Negatives)}{(True\ Positives + True\ Negatives + False\ Positives + False\ Negatives)}$$

Equation 2: Accuracy Calculation

4.1.3 Macro Accuracy

Macro Accuracy is calculated by treating all classes equally and compute the metric independently for each class and then taking the average.

4.1.4 Micro Accuracy

Micro Accuracy is calculated by aggregating the contributions of all classes to compute the average metric.

4.1.5 Precision

Precision depicts the correct proportion of positive identifications given by the model. It’s a measure of result relevancy. Precision of a trained model is calculated as the sum of true positives divided by the sum of true positives and false positives across all classes.

$$Precision = \frac{True\ Positives}{(True\ Positives + False\ Positives)}$$

Equation 3: Precision Calculation

4.1.6 Recall

Recall depicts the correct proportion of actual positives that was identified by the model. It measures how many truly relevant results are returned. Recall of a trained model is calculated as the sum of true positives divided by the sum of true positives and false negatives across all classes.

$$Recall = \frac{True\ Positives}{(True\ Positives + False\ Negatives)}$$

Equation 4: Recall Calculation

4.1.7 Log-Loss

Log Loss calculates the accuracy of a classifier by penalizing false classifications. Log loss metric represents the amount of uncertainty or vagueness of our prediction based on how much it varies from the actual. Minimizing this function can be seen as maximizing the classifier's accuracy. Log-loss closer to 0 indicates a better model.

$$Log\ Loss = -\frac{1}{N}\sum_{i=1}^{N}(y_i\ \log(y_i{}') + (1 - y_i)\log(1 - y_i{}'))$$

Equation 5: Log-Loss Calculation

4.1.8 Log-Loss Reduction

Log-Loss Reduction is called as reduction in information gain. It gives a measure of how much a model improves on a model that gives random predictions. Log-loss reduction closer to 1 indicates a better model.

$$Log\ Loss\ Reduction = \frac{Log\ Loss(prior) - Log\ Loss(classifier)}{Log\ Loss(prior)}$$

Equation 6: Log-Loss Reduction Calculation

4.2 Experiment-based Evaluation

In experiment-based approach, multiple predictive models were trained separately on training set iteratively using each of selected multiclass classification algorithm and evaluated on testing set to assess the macro accuracy, micro accuracy, precision, recall log loss and log loss reduction based on their confusion matrixes.

Following figures will show the individual trainer outputs with above mentioned metric calculations of Linear Multiclass Classification, Naïve Bayes Multiclass Classification, Maximum Entropy Multiclass Classification, Multiclass Decision Forest Classification used in the training process and generated decision trees in Decision Forest algorithm. By comparing each of these trainer outputs most accurate predictive model will be selected and used in classification process.

```
Microsoft Visual Studio Debug Console
=============== Linear Multiclass Classifier ===============

=============== Training  model ===============
=============== End of training process ===============


=============== Cross-validating to get model's accuracy metrics ===============

*************************************************************************************************************
*       Metrics for Multi-class Classification model
*------------------------------------------------------------------------------------------------------------
*       Average MicroAccuracy:    1  - Standard deviation: ()  - Confidence Interval 95%: ()
*       Average MacroAccuracy:    1  - Standard deviation: ()  - Confidence Interval 95%: ()
*       Average LogLoss:            - Standard deviation: ()  - Confidence Interval 95%: ()
*       Average LogLossReduction: 1  - Standard deviation: ()  - Confidence Interval 95%: ()
*************************************************************************************************************


=============== Validating with testing dataset to get model's accuracy metrics ===============

************************************************************
*    Metrics for multi-class classification model
*-----------------------------------------------------------
    MacroAccuracy = 0.9299, a value between 0 and 1, the closer to 1, the better
    MicroAccuracy = 0.9281, a value between 0 and 1, the closer to 1, the better
    LogLoss = 1.6348, the closer to 0, the better
    LogLoss for class 1 = 2.8041, the closer to 0, the better
    LogLoss for class 2 = 2.828, the closer to 0, the better
    LogLoss for class 3 = 1.3734, the closer to 0, the better
    LogLoss for class 4 = 0.0494, the closer to 0, the better
    LogLoss for class 5 = 1.8982, the closer to 0, the better
************************************************************


Confusion table
              ||========================================
PREDICTED     ||     0 |     1 |     2 |     3 |     4 | Recall
TRUTH         ||========================================
0. Loneliness || 1,392 |     3 |    34 |    62 |    24 | 0.9188
1.     Stress ||    27 | 2,261 |    42 |    92 |    48 | 0.9154
2.    Anxiety ||    29 |    16 | 1,811 |     9 |    22 | 0.9597
3.     Normal ||    91 |   128 |    20 | 2,757 |    23 | 0.9132
4. Depression ||     6 |    45 |    21 |    51 | 2,008 | 0.9423
              ||========================================
Precision     ||0.9010 |0.9217 |0.9393 |0.9280 |0.9449 |
```

Figure 8: Linear Multiclass Classification

```
Microsoft Visual Studio Debug Console
=============== Naive Bayes Multiclass Classifier ===============

=============== Training  model ===============
=============== End of training process ===============


=============== Cross-validating to get model's accuracy metrics ===============

*************************************************************************************************************
*       Metrics for Multi-class Classification model
*------------------------------------------------------------------------------------------------------------
*       Average MicroAccuracy:    1  - Standard deviation: ()  - Confidence Interval 95%: ()
*       Average MacroAccuracy:    1  - Standard deviation: ()  - Confidence Interval 95%: ()
*       Average LogLoss:         .006  - Standard deviation: (.001)  - Confidence Interval 95%: (.001)
*       Average LogLossReduction: .996  - Standard deviation: ()  - Confidence Interval 95%: ()
*************************************************************************************************************


=============== Validating with testing dataset to get model's accuracy metrics ===============

************************************************************
*    Metrics for multi-class classification model
*-----------------------------------------------------------
    MacroAccuracy = 0.9298, a value between 0 and 1, the closer to 1, the better
    MicroAccuracy = 0.928, a value between 0 and 1, the closer to 1, the better
    LogLoss = 0.7147, the closer to 0, the better
    LogLoss for class 1 = 0.8404, the closer to 0, the better
    LogLoss for class 2 = 0.9171, the closer to 0, the better
    LogLoss for class 3 = 0.4676, the closer to 0, the better
    LogLoss for class 4 = 0.697, the closer to 0, the better
    LogLoss for class 5 = 0.6349, the closer to 0, the better
************************************************************


Confusion table
              ||========================================
PREDICTED     ||     0 |     1 |     2 |     3 |     4 | Recall
TRUTH         ||========================================
0. Loneliness || 1,392 |     3 |    34 |    62 |    24 | 0.9188
1.     Stress ||    27 | 2,261 |    42 |    92 |    48 | 0.9154
2.    Anxiety ||    29 |    16 | 1,811 |     9 |    22 | 0.9597
3.     Normal ||    91 |   128 |    20 | 2,757 |    23 | 0.9132
4. Depression ||     6 |    45 |    21 |    52 | 2,007 | 0.9418
              ||========================================
Precision     ||0.9010 |0.9217 |0.9393 |0.9277 |0.9449 |
```

Figure 9: Naïve Bayes Multiclass Classification

```
Microsoft Visual Studio Debug Console
=============== Maximum Entropy Multiclass Classifier ===============

=============== Training  model ===============
=============== End of training process ===============


=============== Cross-validating to get model's accuracy metrics ===============

*************************************************************************************************************
*       Metrics for Multi-class Classification model
*------------------------------------------------------------------------------------------------------------
*       Average MicroAccuracy:    1  - Standard deviation: ()  - Confidence Interval 95%: ()
*       Average MacroAccuracy:    1  - Standard deviation: ()  - Confidence Interval 95%: ()
*       Average LogLoss:         .005  - Standard deviation: ()  - Confidence Interval 95%: ()
*       Average LogLossReduction: .997  - Standard deviation: ()  - Confidence Interval 95%: ()
*************************************************************************************************************


=============== Validating with testing dataset to get model's accuracy metrics ===============

************************************************************
*    Metrics for multi-class classification model
*-----------------------------------------------------------
    MacroAccuracy = 0.9299, a value between 0 and 1, the closer to 1, the better
    MicroAccuracy = 0.9281, a value between 0 and 1, the closer to 1, the better
    LogLoss = 0.7584, the closer to 0, the better
    LogLoss for class 1 = 0.9127, the closer to 0, the better
    LogLoss for class 2 = 1.0051, the closer to 0, the better
    LogLoss for class 3 = 0.5478, the closer to 0, the better
    LogLoss for class 4 = 0.6538, the closer to 0, the better
    LogLoss for class 5 = 0.6976, the closer to 0, the better
************************************************************


Confusion table
              ||========================================
PREDICTED     ||     0 |     1 |     2 |     3 |     4 | Recall
TRUTH         ||========================================
0. Loneliness || 1,392 |     3 |    34 |    62 |    24 | 0.9188
1.     Stress ||    27 | 2,261 |    42 |    92 |    48 | 0.9154
2.    Anxiety ||    29 |    16 | 1,811 |     9 |    22 | 0.9597
3.     Normal ||    91 |   128 |    20 | 2,757 |    23 | 0.9132
4. Depression ||     6 |    45 |    21 |    51 | 2,008 | 0.9423
              ||========================================
Precision     ||0.9010 |0.9217 |0.9393 |0.9280 |0.9449 |
```

Figure 10: Maximum Entropy Multiclass Classification

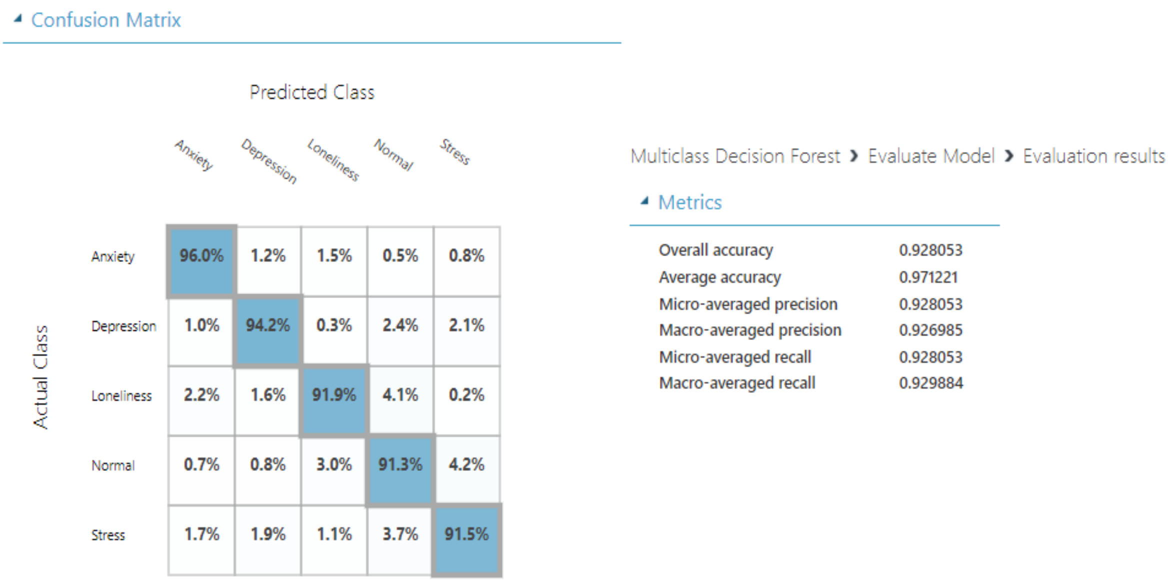


Figure 11: Multiclass Decision Forest Classification

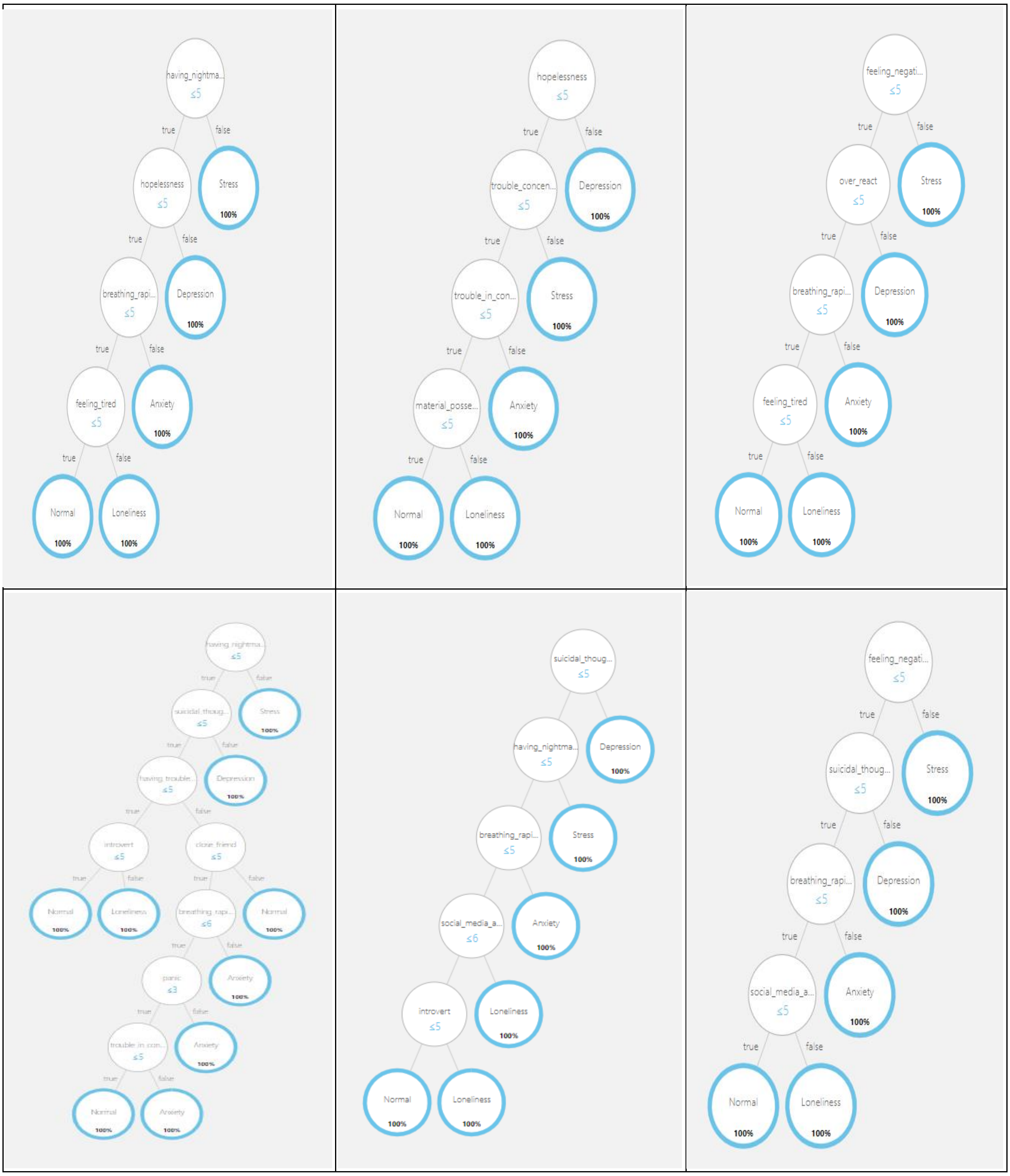


Figure 12: Generated Decision Trees in Decision Forest

4.3 Critical Evaluation

Following is a summarization of above-mentioned trainer output results with macro accuracy, micro accuracy, precision and recall calculations for each selected multiclass classification algorithms.

		Linear Multiclass Classification	Naïve Bayes Multiclass Classification	Maximum Entropy Multiclass Classification	Multiclass Decision Forest Classification
Macro Accuracy		0.9299	0.9298	0.9299	0.9712
Micro Accuracy		0.9281	0.9280	0.9281	0.9280
Precision	Loneliness	0.9010	0.9010	0.9010	0.9448
	Stress	0.9217	0.9207	0.9217	0.9448
	Anxiety	0.9393	0.9392	0.9393	0.9396
	Normal	0.9280	0.9277	0.9280	0.8950
	Depression	0.9449	0.9494	0.9449	0.9261
Recall	Loneliness	0.9188	0.9100	0.9188	0.9600
	Stress	0.9154	0.9388	0.9154	0.9420
	Anxiety	0.9597	0.9468	0.9597	0.9190
	Normal	0.9132	0.9128	0.9132	0.9130
	Depression	0.9423	0.9418	0.9423	0.9150

Table 3: Evaluation Metrics Summary

According to above summary, it can be seen that almost all classification algorithms used in the training process give a similar level of accuracy, precision and recall above 90%. Linear Multiclass Classification algorithm and Maximum Entropy Multiclass Classification algorithm gives almost same outputs since both algorithms trains a linear classification models where only difference is between the probability distribution functions of weights distribution across the classes. Linear Multiclass Classification algorithms provides easier, straightforward, memory and computationally efficient mechanism to implement compared to other algorithms and provide comparatively good outcome in mental health disorder classification based on behavioral measures.

Naïve Bayes algorithm also gives a similar but slightly lesser outcome compared to other algorithms used. This algorithm may not be the optimal choice in certain scenarios. Naïve bayes algorithm assumes independence and equal contribution of the features towards final classification output into the classes, even though there may be behavioral measures that dependent on each other and unequal weighted contribution towards final mental health diagnosis classification. Some behavioral measures such as having suicidal thoughts, sudden panic attacks, rapid weight gain etc. can have higher impact on mental health conditions like anxiety and depression rather than stress or loneliness. Naïve bayes heavily depends on training data set as well. It performs well when mental health disorder classes are uniformly distributed within the dataset. Naïve bayes would not work if there is no occurrence of certain mental health disorder class or behavioral measure in the training data set since posterior probability resulting in zero. These can be seen as reasons for slightly lesser accuracy, pression and recall metrics of naïve bayes algorithm compared to others.

Furthermore, it can be seen that the Multiclass Decision Forest algorithm gives a slightly better accuracy, pression and recall metric outcomes compared to other algorithms used in mental health disorder classification context, since it builds up multiple decision trees and then vote on the most accurate output class. Also, it’s a very computation effective, memory efficient and noise resilient algorithm in presence of noisy features compared to other algorithms used. Since Multiclass Decision Forest algorithm builds up multiple decision trees and then vote on the most popular or confident output class, trees which have high prediction confidence will have a greater weight when assembling the final decision tree. Therefore, it performs well compared to other algorithms when different behavioral measures have differently weighted impact towards final mental health status.

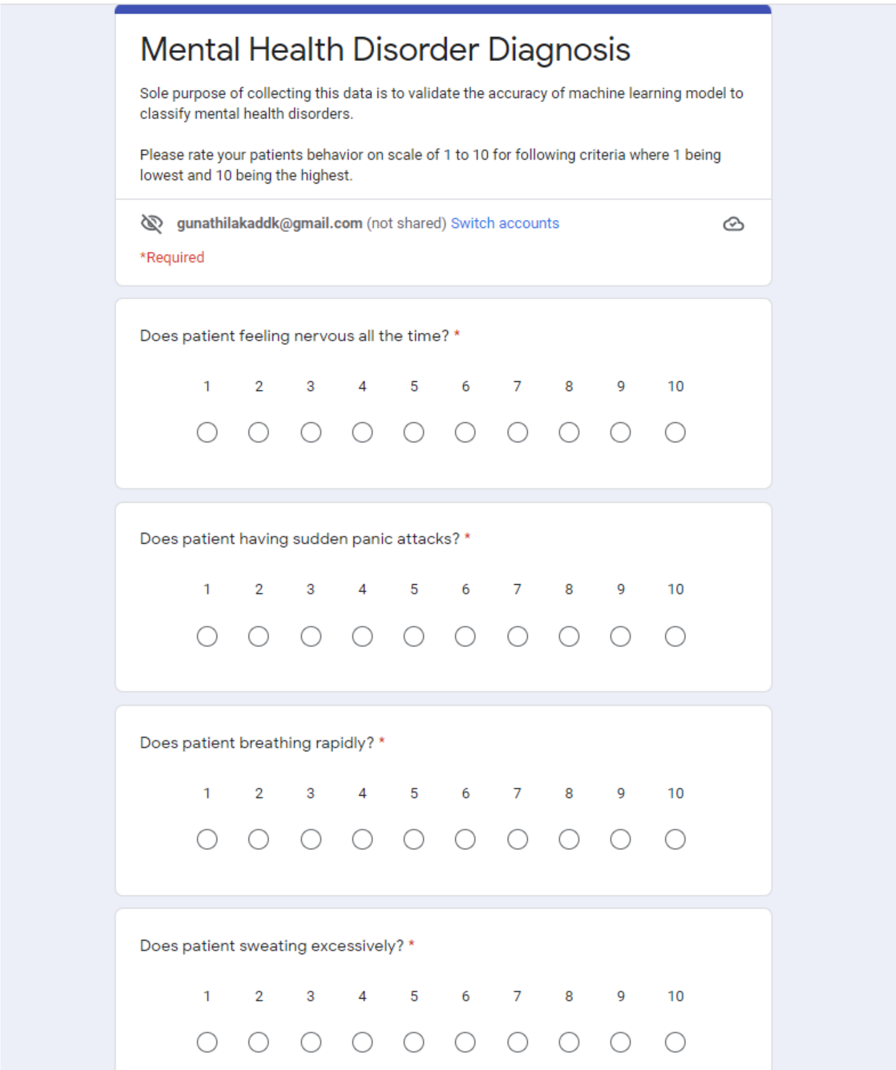
4.4 Cross Validating Model using Expert Opinions

According to literature on mental health, mental health disorders have strong correlation with demographical factors such as age, gender, race, education level, residential area etc. as well as behavioral symptoms such as rapid breathing, sweating, feeling angry and tired etc. For instance, some age groups, some races, people residing in developed countries can think of some behavioral symptoms as luxuries to have rather than necessities, or outcome of some other cause (natural causes etc.) or even as negligible things.

For example, social media addiction can be seen as a usual habit among younger generation rather than symptom of mental health disorder. Sweating excessively can be seen as a usual outcome of weather conditions of topical countries.

Therefore, to cross validate the accuracy and the relevance of trained model within the Sri Lankan context we can use the expert opinions as a secondary evaluation approach. Idea is to seek help from mental health experts, psychiatrist and hospitals within Sri Lanka and collect their patients' behavioral data and mental health conditions along with patient's consent to use these data in training a machine learning model.

By analyzing and applying these patient data against the trained predictive model we should be able to come up with accuracy, precision and recall metrics. Based on these results, further improvements should be done to improve the accuracy and the relevance of the predictive model within Sri Lankan context. Google forms can be used as data collection medium for this phase. Following is a sample google form designed for this purpose.



Mental Health Disorder Diagnosis

Sole purpose of collecting this data is to validate the accuracy of machine learning model to classify mental health disorders.

Please rate your patients behavior on scale of 1 to 10 for following criteria where 1 being lowest and 10 being the highest.

gunathilakaddk@gmail.com (not shared) Switch accounts

*Required

Does patient feeling nervous all the time? *

1 2 3 4 5 6 7 8 9 10

Does patient having sudden panic attacks? *

1 2 3 4 5 6 7 8 9 10

Does patient breathing rapidly? *

1 2 3 4 5 6 7 8 9 10

Does patient sweating excessively? *

1 2 3 4 5 6 7 8 9 10

Figure 13: Data Collection Form to Collect Expert Opinions

CHAPTER 5

CONCLUSION AND FUTURE WORK

In this section, conclusion of the system designed and the further improvements that can be done to the current implementation are explained.

This research was focused on identifying set of generic behavioral symptoms which can clearly classify mental health status, develop a predictive model, and web based application which can predict user’s health status. Based on selected dataset, behavioral symptoms, and multiclass classification algorithms it was able to successfully implement a predictive model and a prototype web application which can categorize user’s mental health status into one of five different statuses (normal, anxiety, depression, loneliness, or stress) with overall accuracy above 90% in all multiclass classification algorithms used.

Based on the evaluation, it was found that Multiclass Decision Forest algorithm gives a slightly higher accurate outcome compared to other algorithms selected, since it builds up multiple decision trees and then ensembles the most accurate decision tree based on prediction confidence. Also, it was identified that Decision Forest algorithm was a very computation effective, memory efficient and noise resilient algorithm in presence of noisy features compared to other algorithms used in the classification process. It was found that Linear Multiclass Classification algorithms provides easier, straightforward, and comparatively good outcome in mental health disorder classification based on behavioral measures. Naïve Bayes algorithm also provide comparatively good classification results but there can be few limitations based on its primary assumption of independence and equal contribution of the features towards final classification output into the classes as described in evaluation section.

In this research, Mental health disorder classification types were scoped down to five categories. (Normal, Anxiety, Depression, Loneliness and Stress) and other types like eating disorders, personality disorders, addictions were considered as out of scope. Also, this study only considered a limited set of behavioral measures (24 measures) as feature set. As a future step machine learning application can be further extended to support more mental health disorder types and behavioral measures. In order to support more disorder types and behavioral measures we may need to come up with extended datasets and data collection mechanisms as well.

In the machine learning application, only five supervised learning multiclass classification algorithms were implemented and used for comparisons. But there can be better classification methodologies exist for this mental health disorder type classification. As a further step, comparison and integration of more multiclass classification algorithms, integration of single highest accurate classifier or use a combination of most accurate classifiers into prototype web application can be done to improve the overall accuracy.

In the training process of the predictive model, only a single dataset which was publicly available on Kaggle, with nearly 45,000 records was used. As a further step of accuracy improvement, we can try to integrate more datasets with different real life patient history records and re-train the predictive model.

According to literature on mental health, mental health disorders have strong correlation with demographical factors as well as behavioral symptoms. As further step we can extend this research with a combination of both these factors, try to find correlations of mental health disorders with the combination of behavioral and demographical factors and come up with a predictive model. This cross validating, improving the accuracy and validity of the developed predictive model within Sri Lankan context with the help of mental health experts, psychiatrist was postponed and considered as future work due to the COVID-19 pandemic situation within the country.

With the rise of global pandemic situations, application like this in commercial environment with reach of thousands of users around the globe will help to overcome limitations of traditional face to face consultations and time-based consultation subscription facilities through audio/video appointments. There is a huge potential to grow machine learning based applications in online digital consultation platforms. Hopefully, with the research and technology advancement in machine learning and artificial intelligence application like this will help the mankind to get accurate and reliable information on their mental health condition and get the necessary professional help they need.

REFERENCES

[1] World Health Organization. 2017. Depression and Other Common Mental Disorders. [online] Available at: https://www.who.int/mental_health/management/depression/prevalence_global_health_estimates/en/ [Accessed 4 February 2021].

[2] Epid.gov.lk. 2017. WEEKLY EPIDEMIOLOGICAL REPORT, Vol. 44 No. 15: Depression and Other Common Mental Health Disorders. [online] Available at: https://epid.gov.lk/web/images/pdf/wer/2017/vol_44_no_15-english.pdf.

[3] Harvey, M., Luiselli, J. and Wong, S., 2009. Application of applied behavior analysis to mental health issues. Psychological Services, 6(3), pp.212-222.

[4] Carleton, R., Afifi, T., Turner, S., Taillieu, T., Duranceau, S., LeBouthillier, D., Sareen, J., Ricciardelli, R., MacPhee, R., Groll, D., Hozempa, K., Brunet, A., Weekes, J., Griffiths, C., Abrams, K., Jones, N., Beshai, S., Cramm, H., Dobson, K., Hatcher, S., Keane, T., Stewart, S. and Asmundson, G., 2017. Mental Disorder Symptoms among Public Safety Personnel in Canada. The Canadian Journal of Psychiatry, 63(1), pp.54-64.

[5] Eisenberg, D., Nicklett, E., Roeder, K. and Kirz, N., 2011. Eating Disorder Symptoms Among College Students: Prevalence, Persistence, Correlates, and Treatment-Seeking. Journal of American College Health, 59(8), pp.700-707.

[6] Skinner, A. and Zack, J., 2004. Counseling and the Internet. American Behavioral Scientist, 48(4), pp.434-446.

[7] Grohol, J., 1999. The Insider's Guide to Mental Health Resources Online. CyberPsychology & Behavior, 2(2), pp.173-173.

[8] Talkspace.com. 2021. Talkspace Psychiatry - Connect with A Psychiatrist. [online] Available at: https://www.talkspace.com/psychiatry/ [Accessed 4 February 2021].

[9] Betterhelp.com. 2021. About Us - The Largest Online Counseling Provider | Betterhelp. [online] Available at: https://www.betterhelp.com/about/ [Accessed 4 February 2021].

[10] oDoc.life 2021. Services • oDoc. [online] Available at: https://www.odoc.life/services/ [Accessed 5 February 2021].

[11] Liu, Y., Hankey, J., Cao, B. and Chokka, P., 2021. Screening for major depressive disorder in a tertiary mental health centre using EarlyDetect: A machine learning-based pilot study. Journal of Affective Disorders Reports, 3, p.100062.

[12] Na, K., Cho, S. and Cho, S., 2020. Machine learning-based discrimination of panic disorder from other anxiety disorders. Journal of Affective Disorders, 278, pp.1-4.

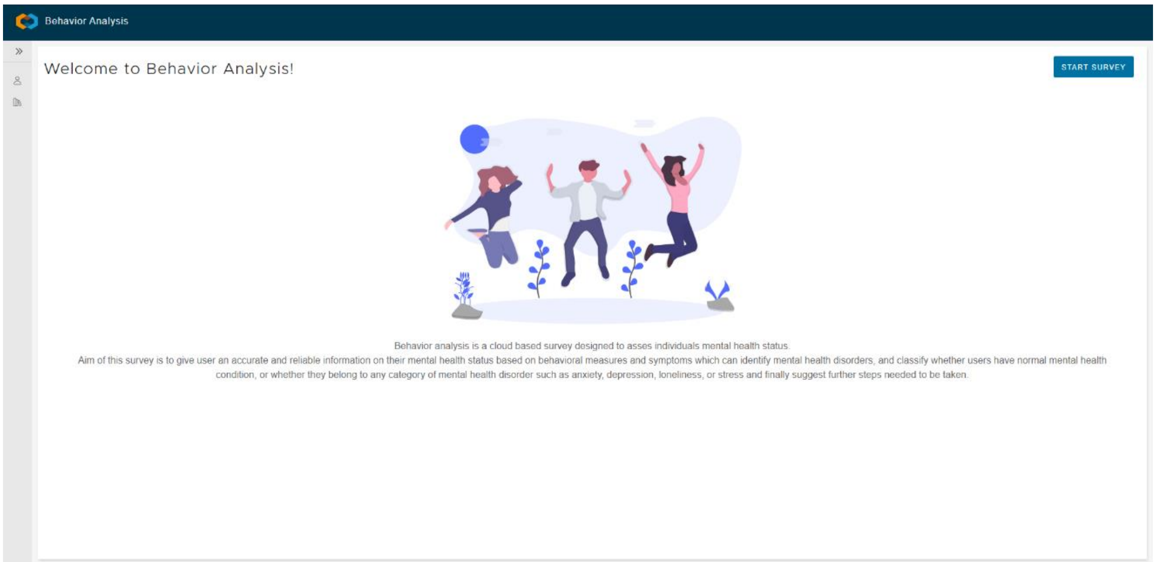
[13] Raab, D., Baumgartl, H. and Buettner, R., 2020. Machine Learning Based Diagnosis of Binge Eating Disorder Using EEG Recordings.

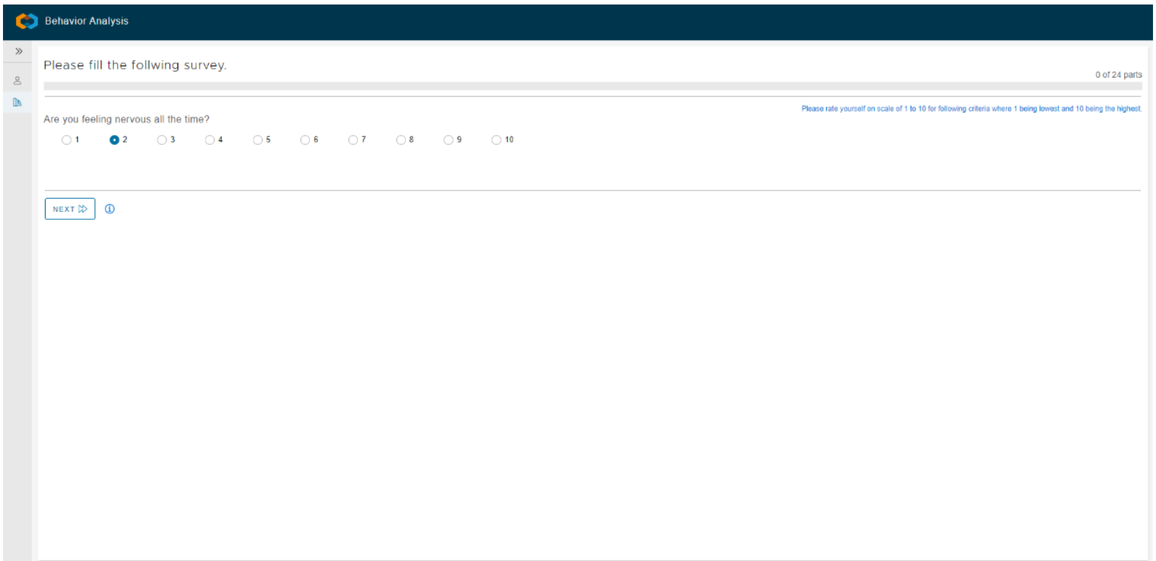
[14] Shatte, A., Hutchinson, D. and Teague, S., 2019. Machine learning in mental health: a scoping review of methods and applications. Psychological Medicine, 49(09), pp.1426-1448.

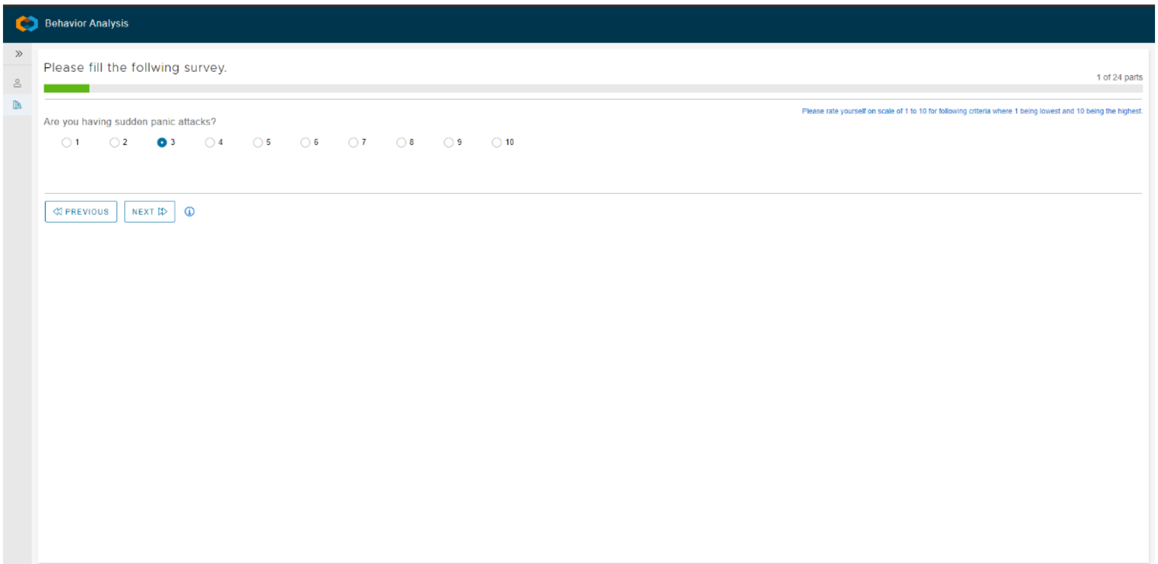
[15] Mohr, D., Zhang, M. and Schueller, S., 2017. Personal Sensing: Understanding Mental Health Using Ubiquitous Sensors and Machine Learning. Annual Review of Clinical Psychology, 13(1), pp.23-47.

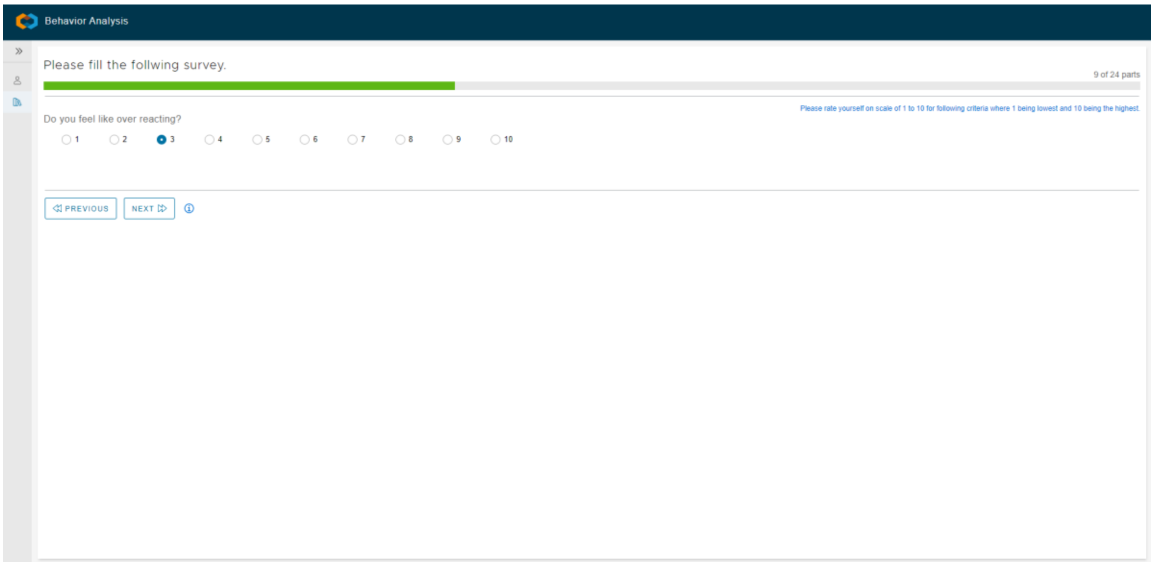
APPENDICES

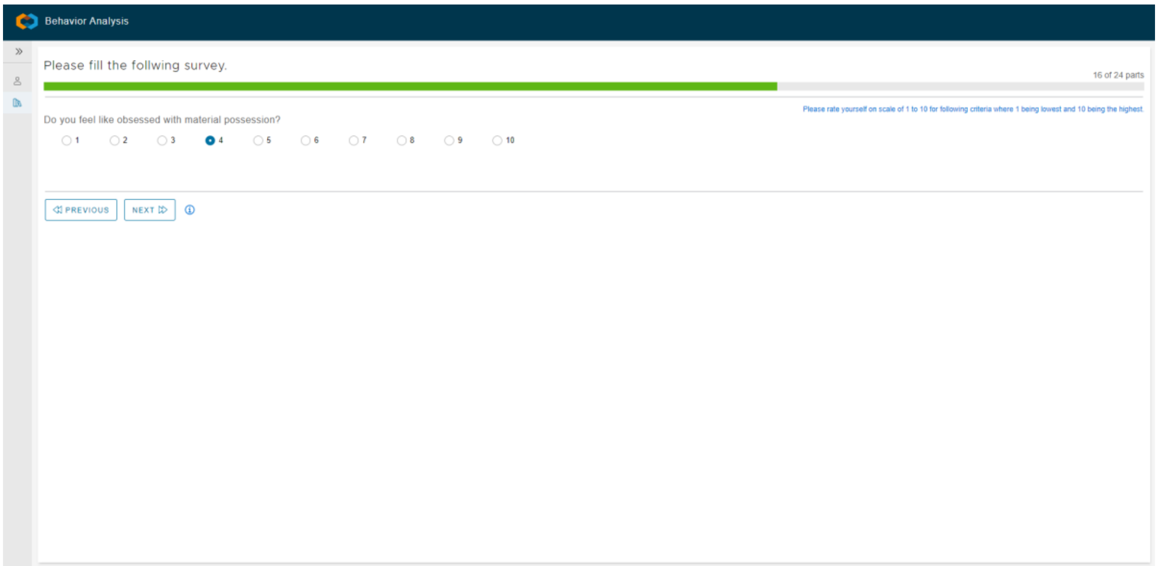
Appendix A: User Interfaces of prototype web application

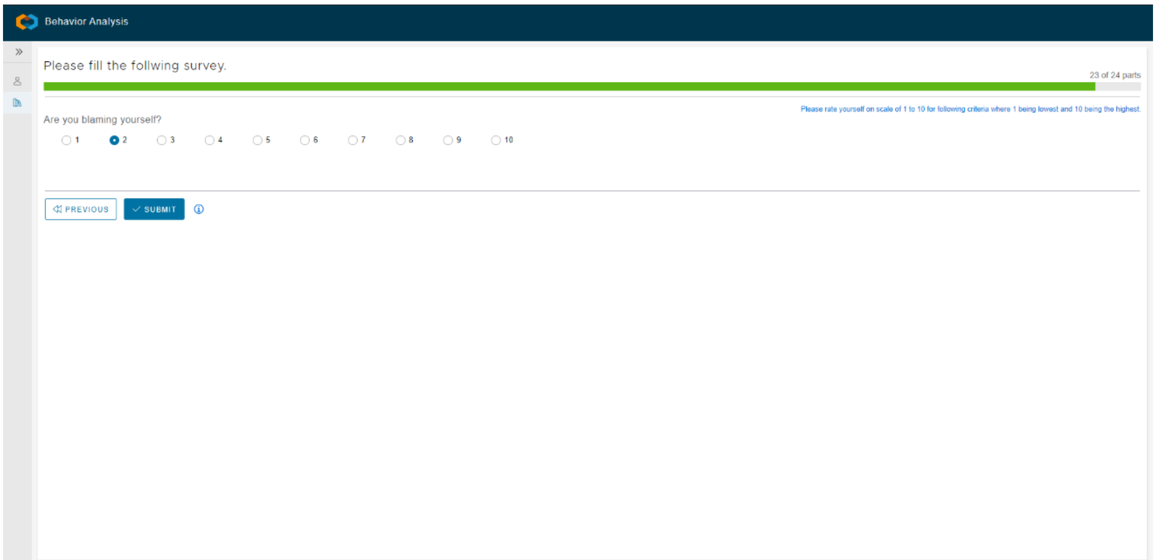


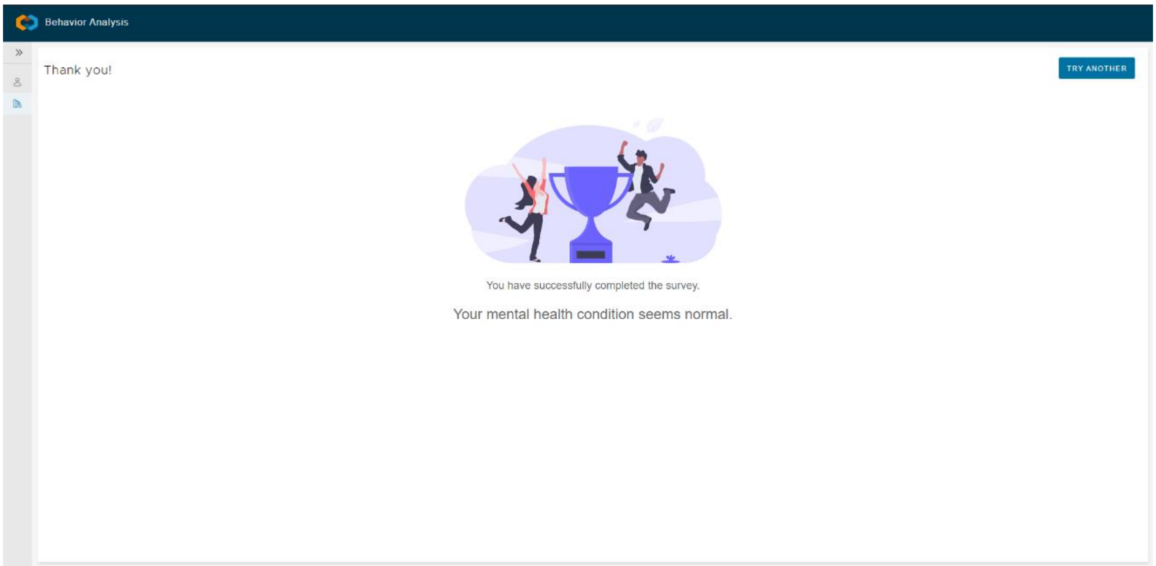


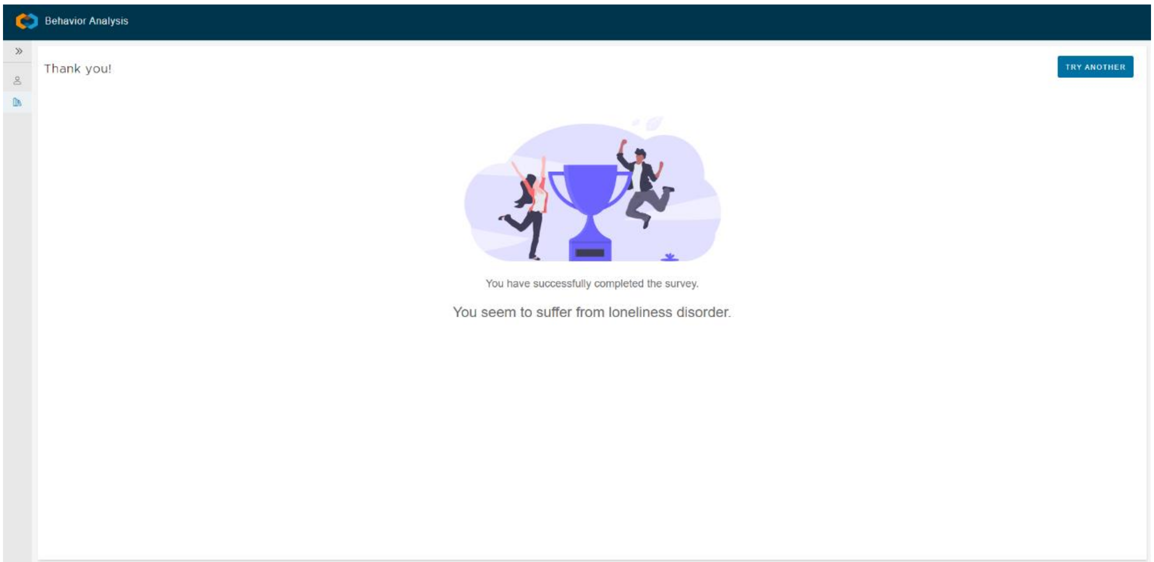


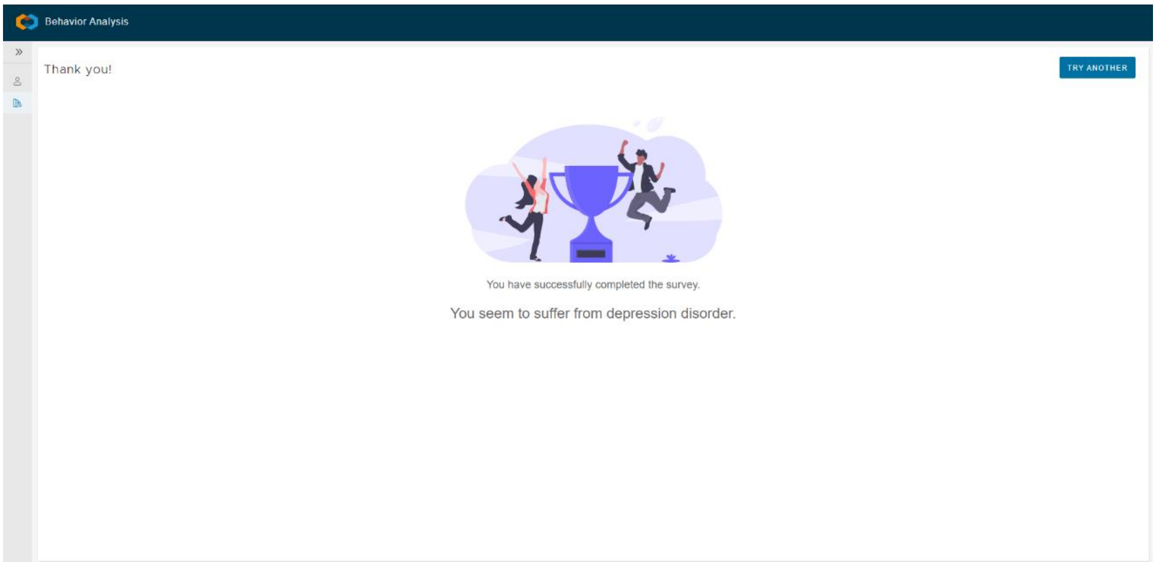

Behavior Analysis
Please fill the follwing survey.
9 of 24 parts
Please rate yourself on scale of 1 to 10 for following criteria where 1 being lowest and 10 being the highest
Do you feel like over reacting?
1 2 3 4 5 6 7 8 9 10
PREVIOUS
NEXT

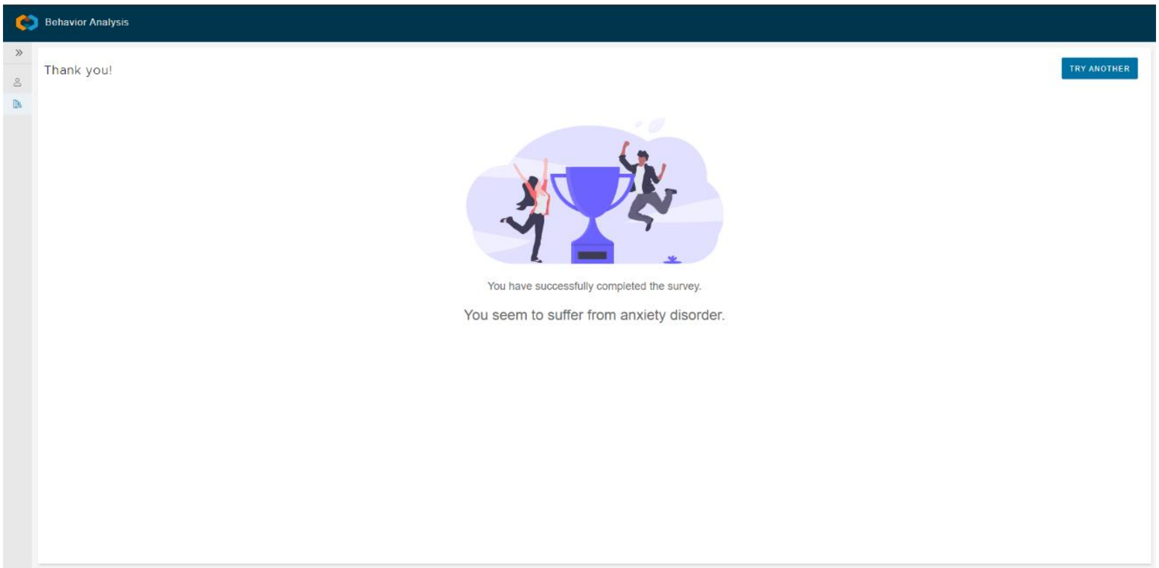

Behavior Analysis
Please fill the follwing survey.
16 of 24 parts
Please rate yourself on scale of 1 to 10 for following criteria where 1 being lowest and 10 being the highest
Do you feel like obsessed with material possession?
1 2 3 4 5 6 7 8 9 10
PREVIOUS
NEXT

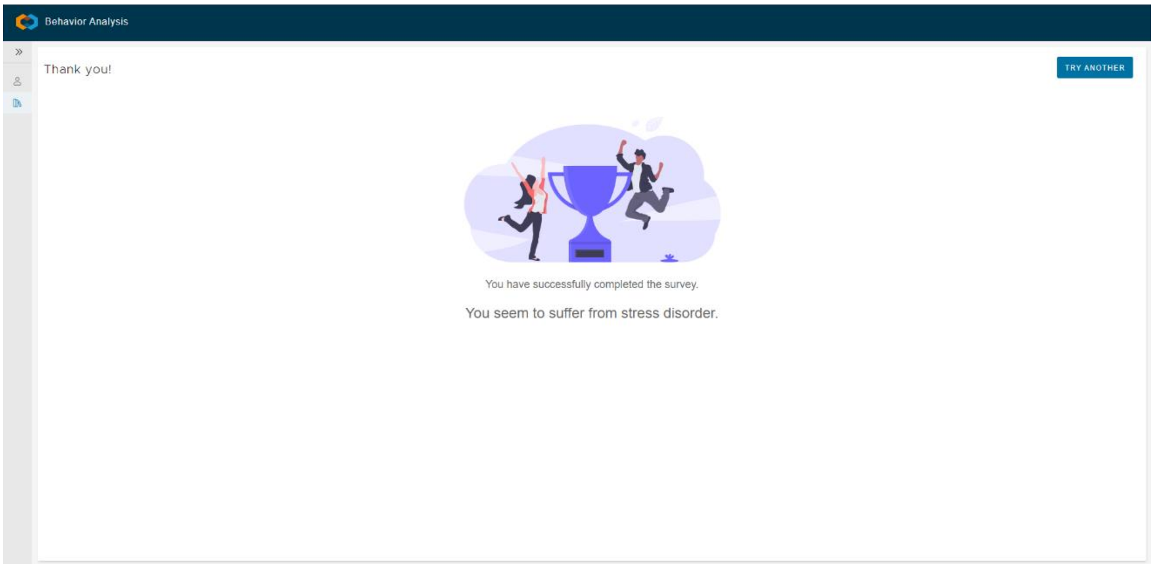

Behavior Analysis
Please fill the follwing survey.
23 of 24 parts
Please rate yourself on scale of 1 to 10 for following criteria where 1 being lowest and 10 being the highest
Are you blaming yourself?
1 2 3 4 5 6 7 8 9 10
PREVIOUS
SUBMIT


Behavior Analysis
Thank you!
TRY ANOTHER
You have successfully completed the survey.
Your mental health condition seems normal.


Behavior Analysis
Thank you!
TRY ANOTHER
You have successfully completed the survey.
You seem to suffer from loneliness disorder.


Behavior Analysis
Thank you!
TRY ANOTHER
You have successfully completed the survey.
You seem to suffer from depression disorder.





Appendix B: Expert opinion data collection form

https://forms.gle/X9vzw1Hi394VBnar9

Appendix C: Source code Git repository

https://github.com/kasungunathilaka/BehaviourAnalysis.git