Implementation of Self-report mHealth Application and Data Analysis

By

Zhongwei Teng

Thesis

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

MASTER OF SCIENCE

in

Electrical Engineering

December 16, 2017

Nashville, Tennessee

Approved:

Richard Alan Peters, Ph.D

D. Mitchell Wilkes, Ph.D

To my advisor, partners and my family.

Thanks for your helps.

ACKNOWLEDGMENTS

The data used in this thesis was obtained from the U.S. Department of Agriculture's Economic Research Service.

TABLE OF CONTENTS

Page

DEDICATION ii

ACKNOWLEDGMENTS iii

LIST OF TABLES vi

LIST OF FIGURES vii

Chapter

1 INTRODUCTION 1

1.1 mHealth Technology 1

1.2 Self-report Collection Application 2

1.3 Self-report Data Analysis 3

1.4 Thesis Overview 4

2 FUNDAMENTAL TECHNOLOGY OF COLLECTING AND ANALYSIS ON SELF-REPORTED DATA 5

2.1 Technology Related to Application 5

2.1.1 Authentication Schema 5

2.1.1.1 Authentication Based on Username and Password 5

2.1.1.2 Authentication Based on OAuth-2 Framework 7

2.1.2 QR Code 9

2.1.3 JSON Web Token 10

2.2 Technology Related to Data Analysis 10

2.2.1 Data Cleaning 10

2.2.1.1 Missing Data 11

2.2.1.2 Outliers Detection 12

2.2.2 Basic Machine Learning Module 13
2.2.2.1 Random Forest 13
2.2.2.2 Boosted Tree 14
2.2.2.3 Model Ensemble 14
2.2.3 Cross-entropy Loss 15
3 IMPLEMENTATION OF QR-CODE BASED AUTHENTICATION ON PAIN CHECK APPLICATION 16
3.1 Overview 16
3.2 Authentication Implementation 16
3.3 Implementation on Pain-check Application 19
4 ANALYSIS AND PREDICTION ON E&H DATASET 21
4.1 Analysis on E&H Dataset 21
4.2 Obesity Prediction on E&H Dataset 21
4.2.1 Evaluation of Random Forest 21
4.2.2 Evaluation of Boosted Tree 21
4.2.3 Evaluation of Model Ensemble 23
4.2.4 Discussion 23
5 CONCLUSION AND FUTURE OUTLOOK 27
BIBLIOGRAPHY 29

LIST OF TABLES

Table Page

4.1 Parameters Description of Random Forest Model 22

4.2 Parameters Description of Boosted Tree Model 24

4.3 Ensemble Model Performance Evaluation 25

LIST OF FIGURES

Figure Page

2.1 Schematic diagram of username and password authentication 6

2.2 Schematic diagram of OAuth2 Authentication 8

3.1 Schematic diagram of QR-Code based authentication 17

3.2 Generating Step of QR-Code Based Authentication 17

3.3 Registration Step of QR-Code Based Authentication 18

3.4 Access Step of QR-Code Based Authentication 19

3.5 Screenshot of Pain-check Application 20

4.1 Correlation Map of E&H Dataset 22

4.2 Entropy Loss of Random Forest Model 23

4.3 Entropy Loss of Boosted Tree Model 24

4.4 Diagram of implemented Model Ensemble 25

Chapter 1

INTRODUCTION

In the healthcare field, subjective reports from patients play an important role, in treatments such as pain management. In fact, analysis has shown that patients' self-reported outcomes provide critical statistics for aiding the diagnosis of conditions, such as cancer [1] and emotional disorders [2]. Decades ago, the data collection relied on paper questionnaire. There were large labor costs for inquiry, questionnaire organization and data input. Procedures for collecting and analysys of subjective reports are necessary for approach.

With the development of mobile technology, innovations in healthcare have grown quickly in recent years. Mobile health (mHealth, a technology that combines mobiles devices with healthcare information systems) has been adopted for many tasks, such as patient monitoring and diagnostics. It is noteworthy that previous research has demonstrated that mHealth can alleviate specific health system limitations that hinder effective coverage of health interventions [3].

To help overcome such shortcomings this paper presents a complete process for mHealth applications. Two components, a self-report collecting application and a data-analysis model, are essential for deriving accurate diagnoses from patients' self-reports.

1.1 mHealth Technology

With the rapid growth of mobile technology, such as mobile device (smart phone) and cloud computing technology, researchers in the fields of health care and public health are discovering the benefits such technology can bring. In recent years,more research results and clinical evidence [4] have shown that mHealth is a very useful tool for patient health management and clinical diagnosis.

According to a comprehensive review of mHealth from Free [5], mHealth technology has been applied to the following disorders: diabetes, hypertension, asthma, eating disorders, HIV treatment, smoking cessation, body weight loss, reducing alcohol consumption, sexually transmitted infection prevention, and testing and data collection.

Despite the advantages of mHealth and its versatility in different situation, researchers still encounter barriers to its adoption. Obstacles, such as knowledge, policy, and infrastructure [6], still prevent mHealth from being widely adopted by healthcare providers.

1.2 Self-report Collection Application

With mobile devices, self-report collections can be made simpler and more efficient for researchers as compared with conventional paper-based questionaires. According to a literature survey by Marie-pierre Gagnon on mHealth applications [7] perceived usefulness and ease of use were the most recurrent factors (19 and 6 elements, respectively) among the 150 elements that emerged as barriers or facilitators to m-health adoption by healthcare providers. The survey included 4096 relevant papers. Based on those results, research is required to find maximally useful self-reporting applications.

Recently, researchers and frontline healthcare workers have been successfully designing user-friendly interfaces for mHealth applications that collect self-reports from patients. Authentication schema, however, essential for mobile application and network security, have frequently been overlooked.

Network security is an essential issue that must be considered with mobile technology. Especially in mHealth, patients' sensitive data must be protected carefully. When a patient is enrolled in an mHealth program it is critical to ensure that the patient's mobile device is connected to the correct medical records, in the same way that wristbands are carefully checked and cross-referenced to establish the physical identity of the patient with their medical record identity. Further, once this relationship has been established between a user's mobile devices and their medical record, rigorous authentication and security mechanisms, such as data-at-rest and data-in-transit encryption, need to be applied.

This paper proposes a novel QR-Code based authentication schema for enhancing patients' user experience. One motivating example, a pain check application, has been tested in the hospital to collect postoperative pain information. By adopting the QR-code authentication schema, this application becomes more secure and alleviates the worries of both providers and patients that the data is well protected. Therefore, We also compare our schema to more conventional methods to asses both security and convenience.

1.3 Self-report Data Analysis

Self-reports collected by an mHealth application are considered to be raw data prior to first analysis. If the raw data is discarded after the analysis important information can be lost. Thus, a "bridge" (pipeline) needs to be established between raw dataset and final diagnoses (or prediction). All raw data are cleaned, analyzed and fitted to appropriate machine model in the pipeline to get an accurate prediction.

This paper introduces the concept and characteristics of “dirty data”. Some key points which may affect data clearing and corresponding solutions are also discussed.

A suitable data model is necessary once the data is prepared. In the field of data analysis mainstream algorithms like gradient boosting [8] and random forest [9] have proven useful for prediction. Many researchers have worked on optimizing current algorithms or proposing new ones to increase the performanace and accuracy of prediction. For example, loss function optimization or distributed optimization have been demonstrated to be a fast and effective ways to enhance model performance.

It is noteworthy that the dataset itself is vital to a model's performance. The performance of the same machine learning model can differ greatly between datasets. For example, Boyd [10] found that model selection through different datasets has a direct effect on performance. Model ensemble/stack, a technology in which training data is analyzed with different strong models in sequence, has been proposed and applied in recent years.

To test and analyze the methods proposed here, the Eating Health (E&H) Module of the American Time Use Survey (ATUS) is used. This self-reported dataset is germane to our purpose. Since the relationship between peoples' obesity and their daily behaviors may be derived from the dataset.

The ATUS Eating & Health (EH) Module was fielded from 2006 to 2008 and again in 2014 to 2016. The EH Module data files contain additional information related to eating, meal preparation and health. E&H dataset comprises three files: a respondent file, an activity file, and a replicate weights file.

The EH Respondent(EHR) file contains a record of 37 variables for each EH module respondent. This dataset reflects the status of respondent, like general health information. The EH Activity(EHA) file documents secondary eating and activity through five related variables. The EH Replicate Weights(EHRW) file contains 161 variables that weight the various EH parameters.

1.4 Thesis Overview

This paper proposed a novel QR-Code based authentication schema and applied it on the pain-check application in the first part. Then to predict obesity based on E&H dataset, we evaluate performance between single model and model ensemble.

The rest of this paper is organized as follows: Section II describes related technology used in this paper. Section III details proposed QR-Code authentication and a practice-based scenario, pain-check application. Evaluations of different machine learning models on E&H dataset are discussed in Section IV. We made summary in Section V.

Chapter 2

FUNDAMENTAL TECHNOLOGY OF COLLECTING AND ANALYSIS ON SELF-REPORTED DATA

2.1 Technology Related to Application

2.1.1 Authentication Schema

We will introduce some conventional authentication schema which have already been applied in most of applications in this section.

2.1.1.1 Authentication Based on Username and Password

In the schema of traditional username/password authentication, each patient need a username and a secret password. The database will maintain a table to store corresponding credentials once a new patient register on client side. To send request to server, patients are required to provide username and password. After authentication, server will send a secret token back to client side. Further request will be accepted via valid token. Sketch map is shown in Fig. 2.1.

Credentials and medical record identity are relatively independent in this schema. mHealth application database can't recognize authorized user when patients logging into system first time. Patients need to enter their identity/profile manually.

Outcome reports are connected with updated profile and personal identity confirmation is relied on patients themselves. Medical record identity is not linked with application's user directly. When patients create username/password on client side, server will not save password into database directly. To enhance security, possible secret keys are prepared by server. Original password will be hashed via complex algorithms and only hashed password will be stored in the database.

During the authentication process, server will encode received password in the same

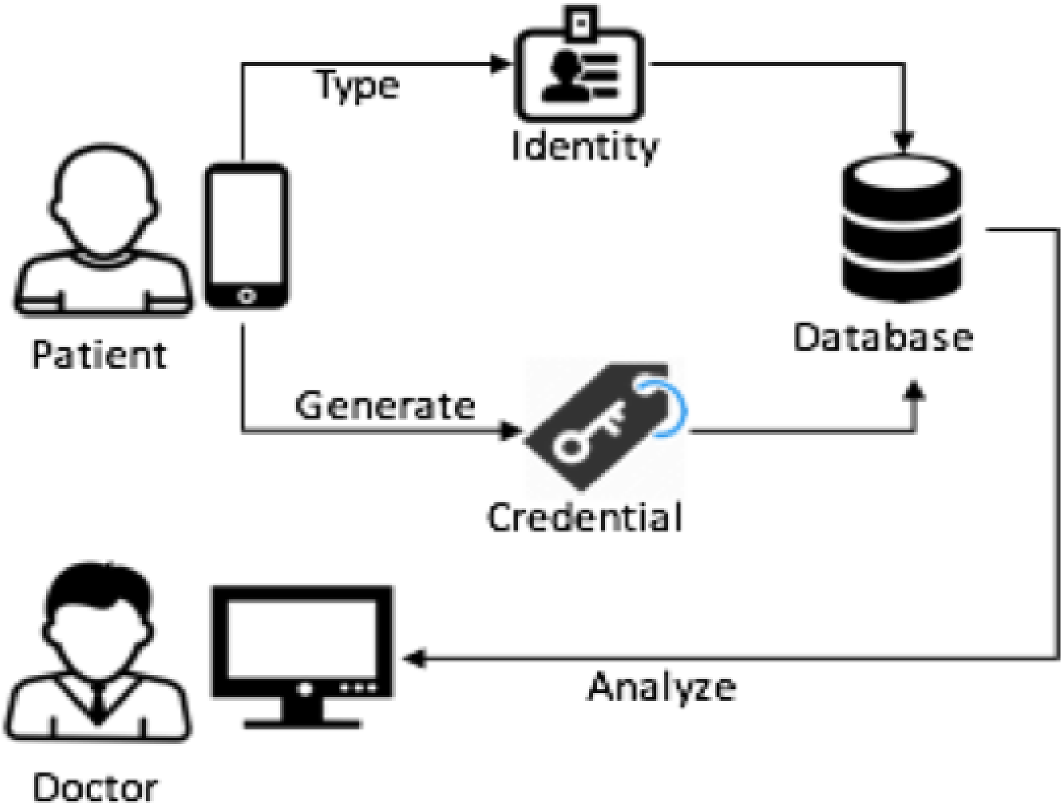


Figure 2.1: Schematic diagram of username and password authentication.

way and search corresponding username/password in the credential table. If correct records are found, a web token then be created and send back to user.

Since the only credential is username in traditional authentication schema, patients can manage credential on device as long as they can provide valid username/password. Generally, patients are allowed to login/log out, switch different accounts or change secret password on same device. On pain check application, if patients want to change password, verification via email is required. Meanwhile, same account can be authorized on different device and patients are available to create different accounts.

Generally, accounts are permanent in username/password authentication even though revocation are available by deleting accounts record in database manually before patient login into application. On mHealth application, considering the mobile platform, the logging process is typically required only when current token expired. Once patients are authorized, they can continually use application until they need a new token.

For proposed application, patients can make pain level reports at any time as long as they have accounts. Patients are encouraged to use application during observation period while credential will not revoke after they recover.

2.1.1.2 Authentication Based on OAuth-2 Framework

OAuth 2 framework [11] is the one of the most popular and safe authentication methods in recent years. Authorization for users is supported by other mainstream sites, such as Google and Facebook. Users can visit various applications using just one credential. Using OAuth2, users benefit from the removal complex and duplicated usernames and passwords and can start to use a new application quickly. OAuth2 provides an additional way for patients besides creating a new username/password pair on pain check application. Registration is not necessary using OAuth2. Sketch map is shown shown in Fig. 2.2. Third party application can provide protected resources for mHealth applications. Basic information, like name and email address, will be stored in the database when user login into appli-

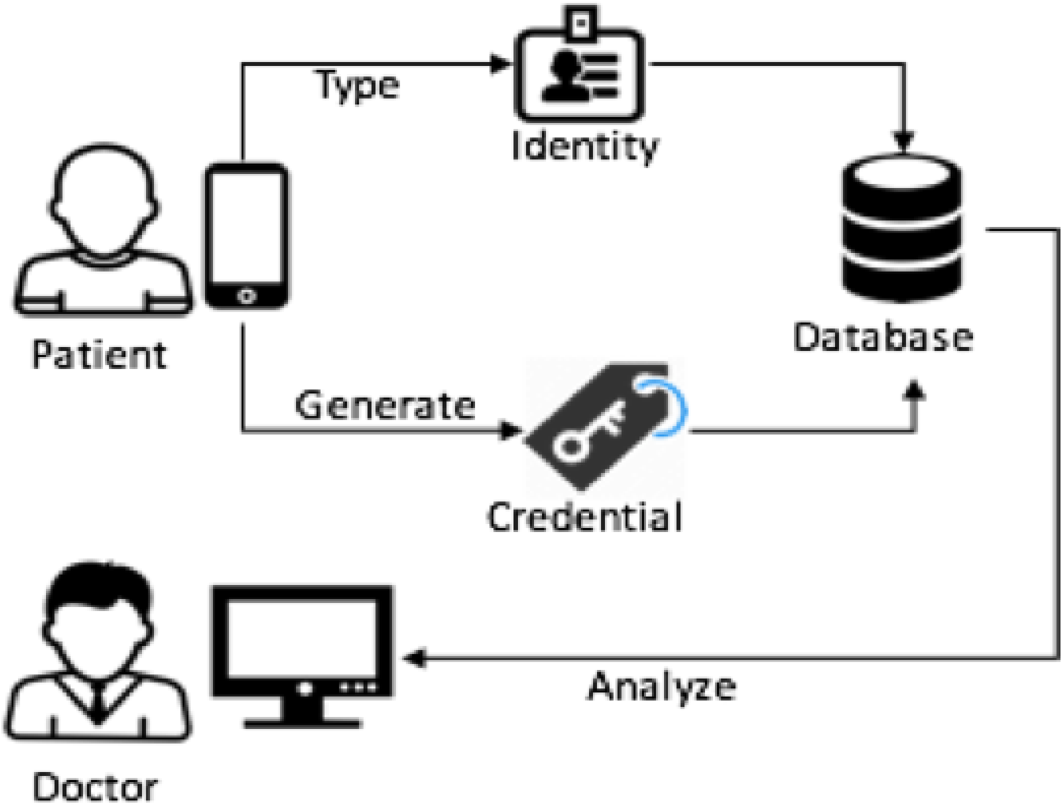


Figure 2.2: Schematic diagram of OAuth2 Authentication.

cation first time. Dependents on patients' profile on the third-party application, mHealth applications may not require further identity.

For proposed pain check application, besides basic personal information, each patient will have their own unique identity. When patient login into application via third party application, they need to input identity for identity confirmation.

When user visit mHealth application first time, client will redirect user to authorization server of third party applications. Users need to authenticate client in this step. After authentication, authorization server send authorization code back to server of mHealth application. Application server then request access token/ refresh token from authorization server by sending authorization code.

Client can catch protected resources with the help of access token. Access token is not permanent. Expiration time is decided by third party applications.

Under OAuth2 framework, mHealth applications will not store users' original username/password. Credential management is responsible for third party application. As long as patients have valid accounts, they can switch these accounts on same device.

Patients can change personal information without limitation on pain check application. Since their submissions are only connect with identity showing on their profile, they can bind multiple accounts with same identity.

For mHealth application which applied authentication based on OAuth2, it only keeps the access token or refresh token that is generated by other applications. Any credential change need to be done on third party application rather than our client side. Meanwhile, we can't revoke authorized user's credential.

2.1.2 QR Code

QR-Code, acronym of quick response code, is a two-dimensional barcode which is designed by a Japan company in 1994. The fast readability and storage capacity make it a better choice when applied in the mobile application compare to standard one-dimensional barcode. When inputs are numeric-only, it can contain up to 7089 characters (4296 when inputs are alphanumeric). Error correction is based on ReedâĂŞSolomon error correction algorithm and with highest error correction level, 30% of codewords can be restored.

Since QR-Code is not a human-readable medium, related QR-Code reader is necessary. Luckily nowadays, mainstream mobile operating systems (iOS and Android) have fully supported straight QR scanning which makes related technology practical. The only needed hardware is a cellphone.

2.1.3 JSON Web Token

JSON, regarded as JavaScript Object Notation, is used to transmit data in a human-readable format. JWT [12] is a JSON-based open standard which has been widely used in modern authentication schema to transmit access token in a compact way securely. In this case, JWT can be send through multiple ways, like HTTP header or URL parameters, in fast speed due to its compact size. Meanwhile, it also can contain sufficient information encoded in a JWS (JSON Web Signature) and/or JWE (JSON Web Encryption) structure.

JWT is consisted of three parts: header, payload and signature. Header is defined the hashing algorithm used to encode JSON object. Payload contains message which need to be transmitted. It also claimed the its expiration time which can be used as used to revoke authentication. The signature is used to verify the accuracy of token.

2.2 Technology Related to Data Analysis

2.2.1 Data Cleaning

When building our model, we always want our data to be perfect so we can be more concentrated on optimize model itself. While real world is not as perfect as we expected, data collection is always affected by noise [22]. Unprocessed data is usually called "dirty data". Feeding dirty data into machine learning model will mislead us to make wrong decision, even though corrupted data may even give us a temporary better result. We will introduce some common problems in data cleaning.

2.2.1.1 Missing Data

The literature on the statistical analysis of data with missing value has flourished since early 1970s [13]. When respondents reply to questionnaires, they may refuse to answer some questions. Sensitive questions, like household income, is highly possible to be rejected. It's also possible that respondents just forget the answer. It happens for either category features or continuous features. In modern data analysis, whole dataset is constructed as a matrix. Each row represents an unique observation or respondent. Features are represented by columns. Thus, the identification of missing value are often codes as "Unknown" or "-1" in the raw dataset. For category data, like âĂIJHave you drink soda in the past week", it often maps different answer to simple integer and provide an additional value for unknown number. This may not work when handling continuous data, missing continuous values are often set to be null in the dataset.

Solutions according to the problem can be categorized as three parts: deleting observations, average imputation and classification Imputation.

When talking about deleting observations, as the most straightforward way of cleaning missing data, deleting observations will also bring researchers lot of troubles, especially when there are not enough data. While it still deserves us to detect each observation before dive into other imputation methods. Considering invalid observations, that most of features are set to null, will affect the accuracy of imputation, we need to delete them from our dataset as the first step.

We can apply average imputation when a continuous feature has enough valid observation, we can aggregate existing data and calculate their average value. The null value within dataset will be replaced by average value as an assumption about generally case. The average imputation can be easily affected by outliers. Any outliers will extremely increase/decrease average value. In this case, this method will mislead the learning model and try to tell the model a wrong story.

Average imputation is not suitable for category feature. The idea of classification was then proposed. Null value in the dataset should be predicted based on other features. Classifier is used during the process of data cleaning. This method splits one classification problem to multiple problems. The new value is considered as the most possible choice for current person.

2.2.1.2 Outliers Detection

As we mentioned before, data is not perfect and like real world, there are always noise in the raw dataset besides missing data. For example, when people did data entry work, they might make mistakes. The corresponding dataset may contain uncertain values. Type errors, like letters occurs in the integer type, is easy to be found and be cleaned.

Outliers we try to detect is usually not caused by human error. Due to some personal reasons, single observation may have much better experimental value that others and this unpredictable value will harm performance of data model. A simple example for our EH dataset is that you may found a people with perfect BMI, while he has fast food every day. This observation is trying to tell our model that fast food is good for keeping you weight which is different with other observations. That's why we need to clear outliers.

Outliers is usually obvious in the distribution graph. When dealing with small dataset, the most effective way is to analyze features in the dataset one by one. By detecting outliers manually, a threshold can be made as a filter. This work can be extremely tedious with the growth of size of features and observations. Meanwhile, it's hard to make a perfect threshold when ambiguous boundaries on the graph.

Probabilistic noise identification [14] is a methods proposed by Kubica to solve this problem more precisely. Points are generated using three distinct models: the clean model, the noise model, and the corruption model. Predicted data are generated by probabilistic model which is considered as noise system. After detection of corrupted data, corrected value will also be predicted.

2.2.2 Basic Machine Learning Module

The goal of this paper is to evaluate the performance of several given machine learning models and corresponding model ensemble for given E&H dataset. In this section, the characteristics of different learning models which are used will be discussed.

2.2.2.1 Random Forest

Random Forest was proposed by Breiman [9] first time in 2001. Andy Liaw and Matthew Wiener implemented and proved the high performance of random forest for both classification and regression example [15]. Implementation and practical usage of this model are also stated in the paper.

Generally, the idea of random forest take the advantage of boosting and bagging. Variable selection is an important factor [16] to build a reliable data model. According to previous research [9], not all features are benefit for our prediction. Feeding too much features to our model without consideration may mislead our algorithm. Variable importance measuring is an critical topic for random forest.

In random forest, different features are chosen randomly and combined as a new dataset. Dataset with different variables are called trees. All trees need to be trained individually at the beginning. Based on the results, trees with better performance will be given higher weight when making final prediction. In the end, a simple majority vote is taken for prediction [15].

Currently, one of the most widely used interface of random forest is âĂIJscikit learn" [17]. We will use it to evaluate our dataset. When creating âĂIJRandom Forest Classifier" in âĂIJscikit learn" package, several parameters will hugely affect model performance. We will briefly discuss these features before dive into implement it.

2.2.2.2 Boosted Tree

The idea of boosted tree was first proposed by Friedman [8] to improve tree-based supervised learning. Rather than random forest, Gradient boost of regression trees produces competitive, highly robust, interpretable procedures for both regression and classification, especially appropriate for mining less than clean data [8].

The core concept of boosted tree is tree ensemble. When we get a cleaned dataset, a set of classification and regression trees [18] can be derived from it. Leafs in classification and regression trees contains prediction score which is better for optimization in probability prediction model. One thing need to be pointed out is the difference between boosted tree and random forest. They are both ensemble tree, the difference is the way we train them.

XGBoost is an open source package implemented by Tianqi Chen [19]. This portable and reusable package support multiple languages and has solved problems beyond billions of examples. Meanwhile, it also provides a fast way to solve problem in distributed system.

2.2.2.3 Model Ensemble

The idea of multi-model ensemble has successfully shown its performance in many studies. For a given dataset, different model shows different performance limitation by adjusting model parameters. While when combined with weaker model, the overall ensemble model can outperform better than any participating single model. Weigel proposed that multi-model ensembles can indeed locally outperform a ’best-model’ approach, but only if the single-model ensembles are overconfident [20].

Meanwhile, the performance also affected by the way we ensemble weaker model. Specifically, the way how single model make a prediction matrix for the next layer will hugely affect the overall loss value. On the other hand, higher layer depth will give us a better performance and it may cause overconfident problem.

2.2.3 Cross-entropy Loss

The cross-entropy (CE) method is a new generic approach to combinatorial and multi-extremal optimization and rare event simulation [21]. Instead of using precision rate or recall rate for our classifier, we can use cross-entropy loss to evaluate probability outputs as a probability estimator. Loss function of each sample can be defined as [21]:

$$L_log(y,p) = -(ylog(p) + (1-y)log(1-p)) \tag{2.1}$$

Chapter 3

IMPLEMENTATION OF QR-CODE BASED AUTHENTICATION ON PAIN CHECK APPLICATION

3.1 Overview

The idea of user authentication using QR Code was proposed [22] in recent years. Based on a QR Code scanning and JWT approach, this authentication method can provide short-term credentials for patients. Specifically, when doctors want to introduce pain check application to patients, they can show QR code on their phone and let patients scan this to start to use application. We will test our schema on pain-check application in the section. Overview diagram is shown in Fig. 3.1.

3.2 Authentication Implementation

Taking advantages of QR-Code, JWT and mobile device, the schema is designed to provide secure and handiness. The authentication process can be divided into three parts: generating QR-Code, patients' registration and credentials validation. Entire authentication process is accomplished by resource provider(RP), doctors' clients(DC) and patients' clients(PC).

The QR-Code in the process varies according to different patients. Thus, QR-Code generator is located in the doctor's mobile device. The generating phrase is described as follow. Graph is shown in Fig. 3.2.

1. DC choose target patient and send patient's Id and their own id to RP.

2. RP validate doctor's credentials and received doctor's id. If they are not matched, request will be rejected.

3. RP encrypt doctors' and patients' information in the form of JWT then send registration token back to DC. The registration token should expire soon.

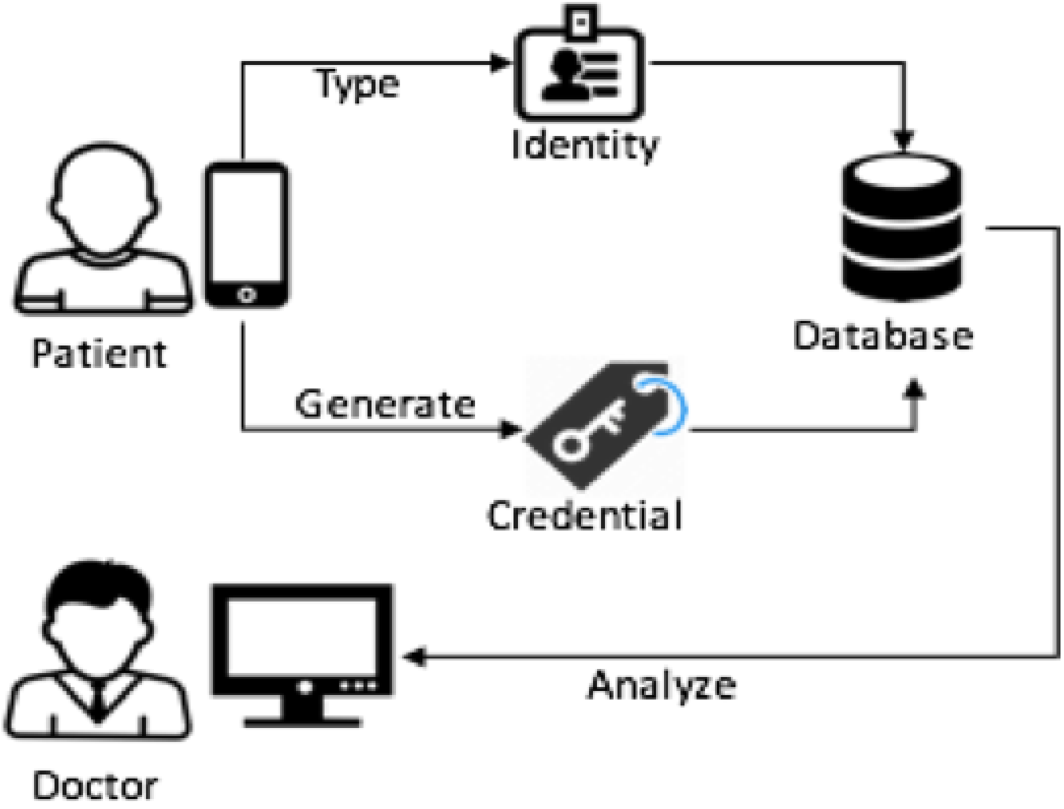


Figure 3.1: Overview of QR-Code based authentication.

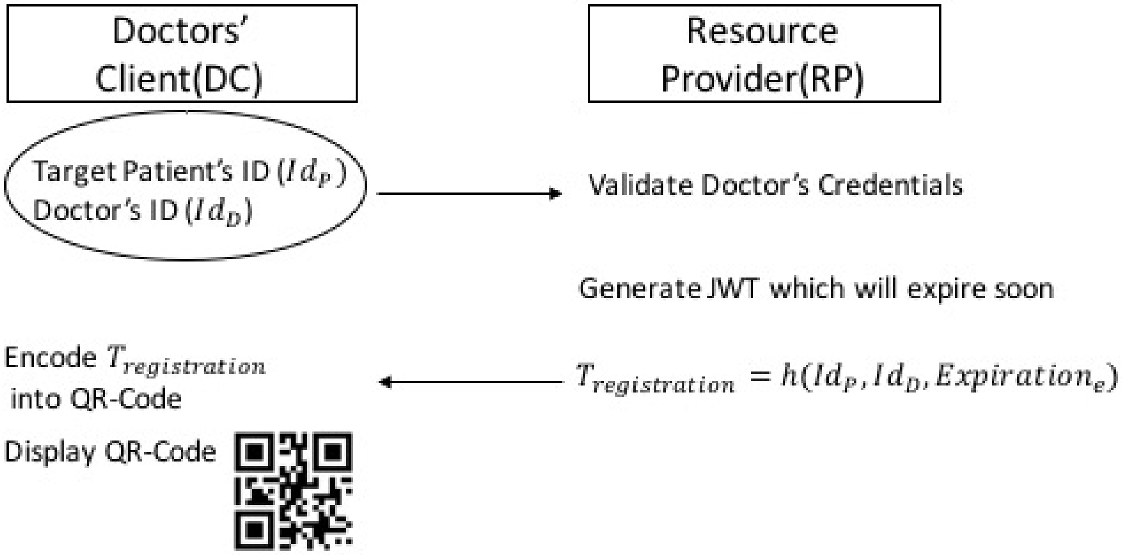

Doctors' Client(DC)
Resource Provider(RP)
Target Patient's ID (Id_P)
Doctor's ID (Id_D)
Validate Doctor's Credentials
Generate JWT which will expire soon
Encode $T_{registration}$ into QR-Code
$T_{registration} = h(Id_P, Id_D, Expiration_e)$
Display QR-Code

Figure 3.2: Generating Step of QR-Code Based Authentication.

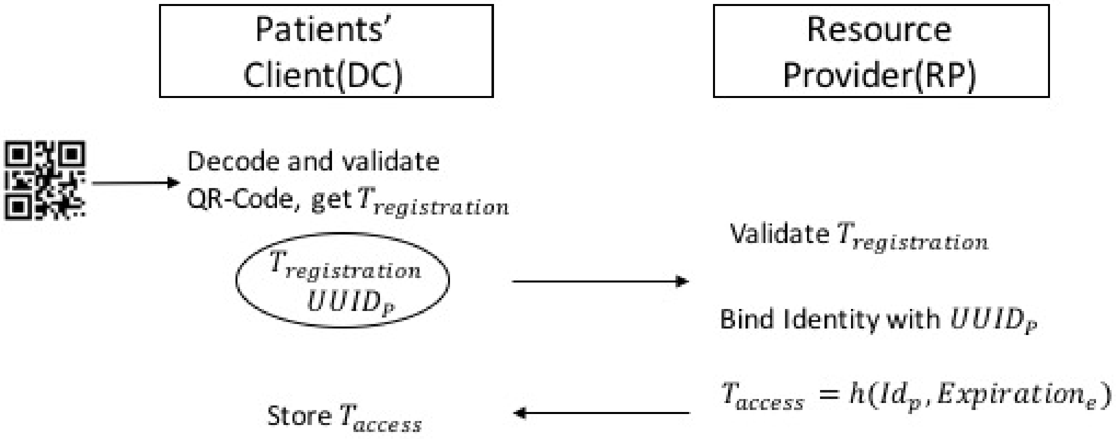

Patients' Client(DC)
Resource Provider(RP)
Decode and validate QR-Code, get $T_{registration}$
$T_{registration}$ $UUID_p$
Validate $T_{registration}$
Bind Identity with $UUID_p$
$T_{access} = h(Id_p, Expiration_e)$
Store T_{access}

Figure 3.3: Registration Step of QR-Code Based Authentication.

4. DC encode registration token as JavaScript object notation (JSON) into QR-Code.

Patients' registration is used to bind PC and medical identity in the database. The registration phase can be described as below. Graph is shown in Fig. 3.3.

1. PC scan QR-Code to get JSON and corresponding registration token. If detected JSON is in the wrong format, then reject it.
2. PC send registration token and its own universally unique identifier (UUID) to SP.
3. If registration token has expired or it's invalid, SP will send rejection to PC. Otherwise, SP will bind patient's identity with received UUID.
4. SP generate access token (JWT) and send it back to PC.

Patients' access phrase can be described as below. Graph is shown in Fig. 3.4.

1. PC send access token and UUID to RP.
2. RP valid if current UUID match data in the database. If matched, PC are allowed to access private data.

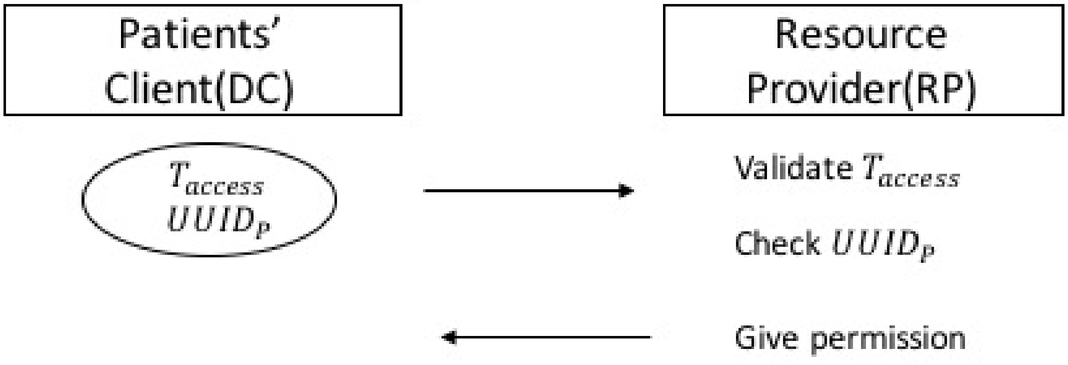


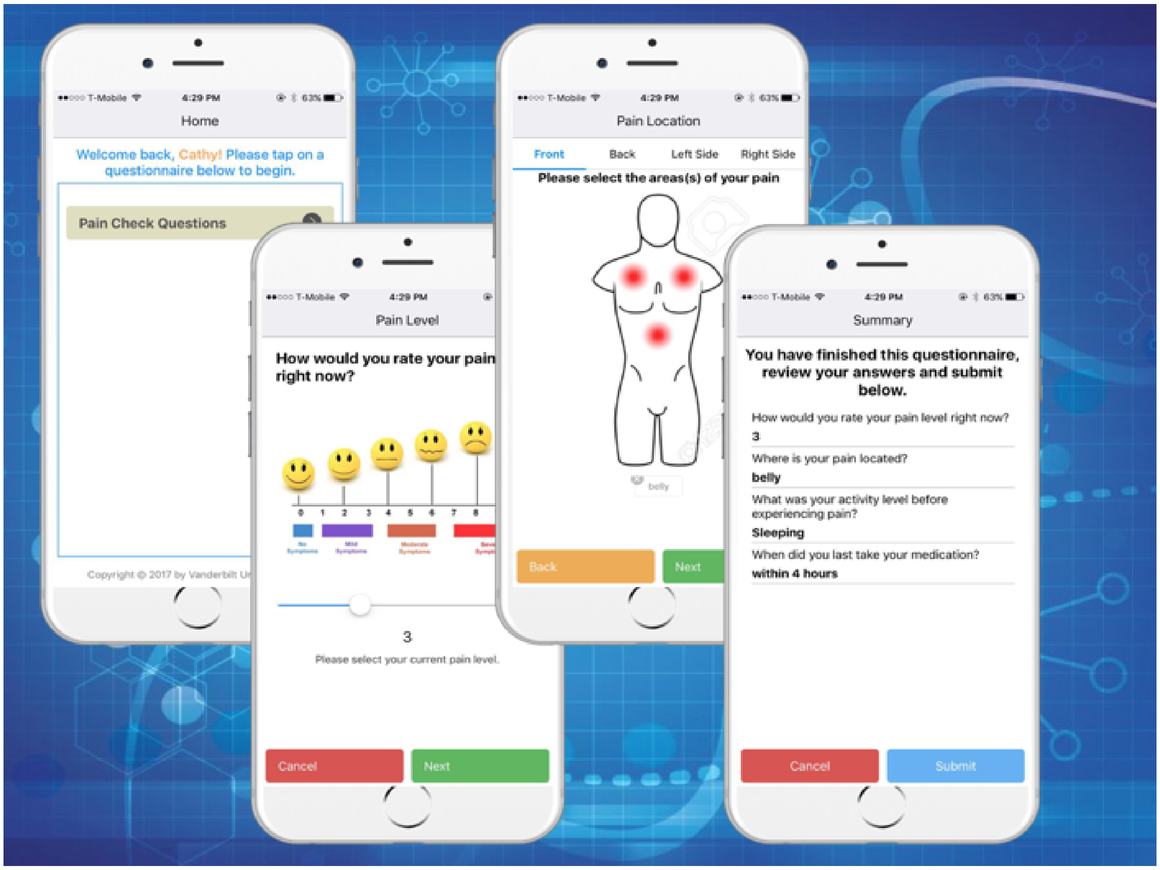
Figure 3.4: Access Step of QR-Code Based Authentication.

3.3 Implementation on Pain-check Application

One practical scenario which is fit for applying QR-Code based authentication is self-report pain management application.

Pain management plays critical role in healthcare program from decades ago. Evidence extracted from published data shows that concise postoperative pain measurement has comparable positive influence to pain management strategy [23]. The measurement of subjective pain intensity continues to be important to both researchers and clinicians [24].

Target patients of pain management often suffer from postoperative pain. Meanwhile, the patient's age is also a factor we have to consider. The handiness and the convenience brought by proposed authentication can effectively reduce the barriers for patients to use this in a practical way. The application's screenshots are shown in Fig. 3.5.



Home

Welcome back, Cathy! Please tap on a questionnaire below to begin.

Pain Level

How would you rate your pain right now?

3

Please select your current pain level.

Pain Location

Please select the areas(s) of your pain

Summary

You have finished this questionnaire, review your answers and submit below.

How would you rate your pain level right now?
3

Where is your pain located?
belly

What was your activity level before experiencing pain?
Sleeping

When did you last take your medication?
within 4 hours

Figure 3.5: Screenshot of Pain-check Application.

Chapter 4

ANALYSIS AND PREDICTION ON E&H DATASET

4.1 Analysis on E&H Dataset

Features in this dataset can be categorized into two parts, behavior and characteristics. In this thesis, behavior means response to the questionnaires from respondent, for example, the time they spending in primary drinking and eating which is subjective answer. Characteristics is the objective features of respondent, like body mass index (BMI). We choose correlation map to take a glance of the relationship between behavior and characteristics. Correlation map is shown in Fig. 4.1.

It's clear to see from the map that most features won't affect others directly. Some features show high correlation with others, like the eating behavior with meat and milk. High correlation may have side effect to our model.

4.2 Obesity Prediction on E&H Dataset

4.2.1 Evaluation of Random Forest

In this section, we evaluate random forest model by adjusting 4 parameters. Plot below shows when parameters changed, the value of cross-entropy loss. The parameters are described in Table. 4.1.

Results are shown in the Fig. 4.2.As we can see, with the increase of tree number and depth of tree, the loss become smaller and smaller. The log loss tends to be around 0.3.

4.2.2 Evaluation of Boosted Tree

For boosted tree model in XGB package, the parameters that we can adjust is shown in the Table. 4.2.

By modify parameters and document corresponding cross-entropy loss, we plot figures shown in Fig. 4.3. Even though loss in the training set decrease when we try to increase

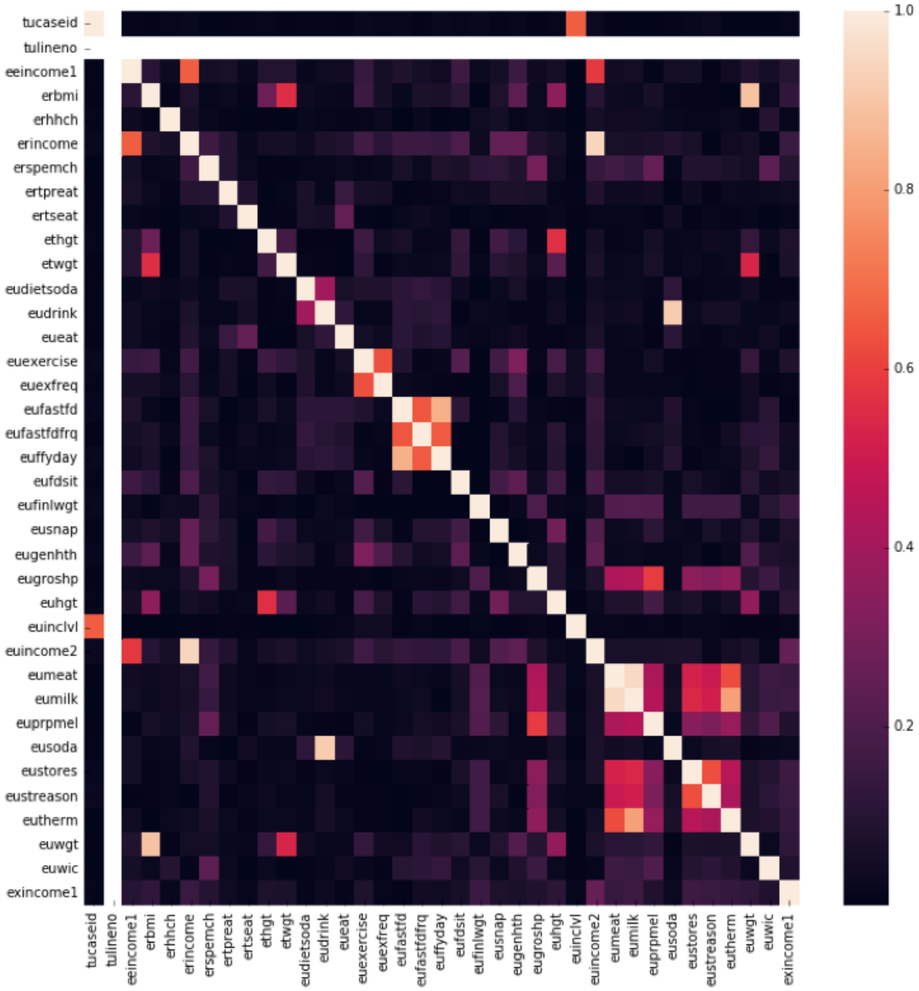


Figure 4.1: Correlation Map of E&H Dataset

Table 4.1: Parameters Description of Random Forest Model

Parameters	Description
n_estimators	The number of trees in the forest
max_features	The number of features to consider when looking for the best split
max_depth	The maximum depth of the tree
min_samples_split	The minimum number of samples required to split an internal node

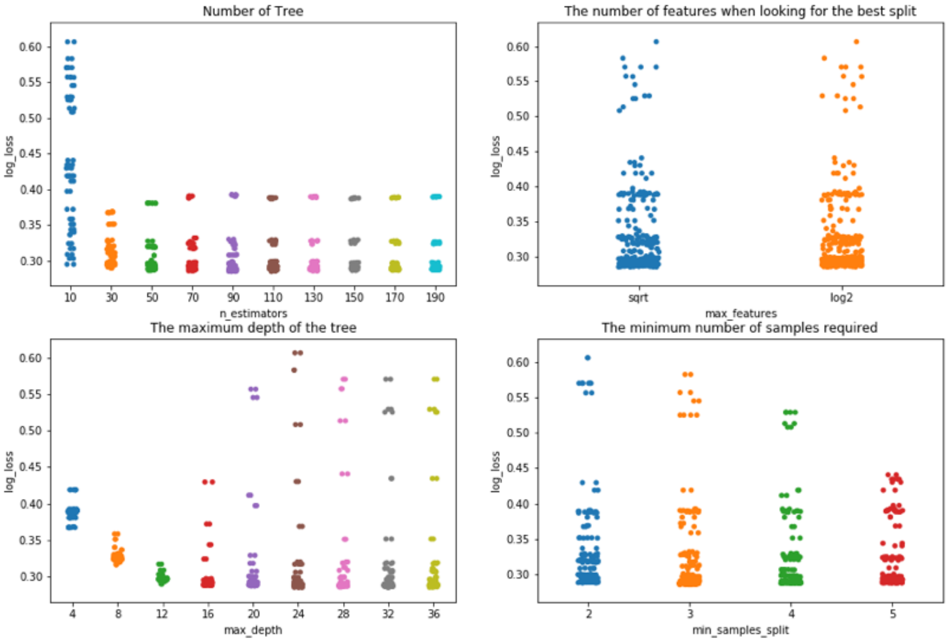


Figure 4.2: Entropy Loss of Random Forest Model

max_depth or subsample, we can see the loss of test set in the figure that increase. The overall loss tends to be around 0.25 to 0.26.

4.2.3 Evaluation of Model Ensemble

Besides random forest and boosted tree, we will also apply weaker single model implemented in Scikit-Learn package, like logistic regression classifier, Gaussian process classification. The schema of ensemble is shown in Fig. 4.4:

We compared best entropy-loss of single model in different layer and documented in the Table. 4.3

4.2.4 Discussion

We are surprised to find that compared with feeding data directly into single model, the performance of second layer and third was dropped down by features of prediction

Table 4.2: Parameters Description of Boosted Tree Model

Parameters	Description
Eta	step size shrinkage used in update to prevents overfitting
Max_depeth	maximum depth of a tree, increase this value will make the model more complex to be overfitting.
Subsample	subsample ratio of the training instance.
Colsample_bytree	subsample ratio of columns when constructing each tree.

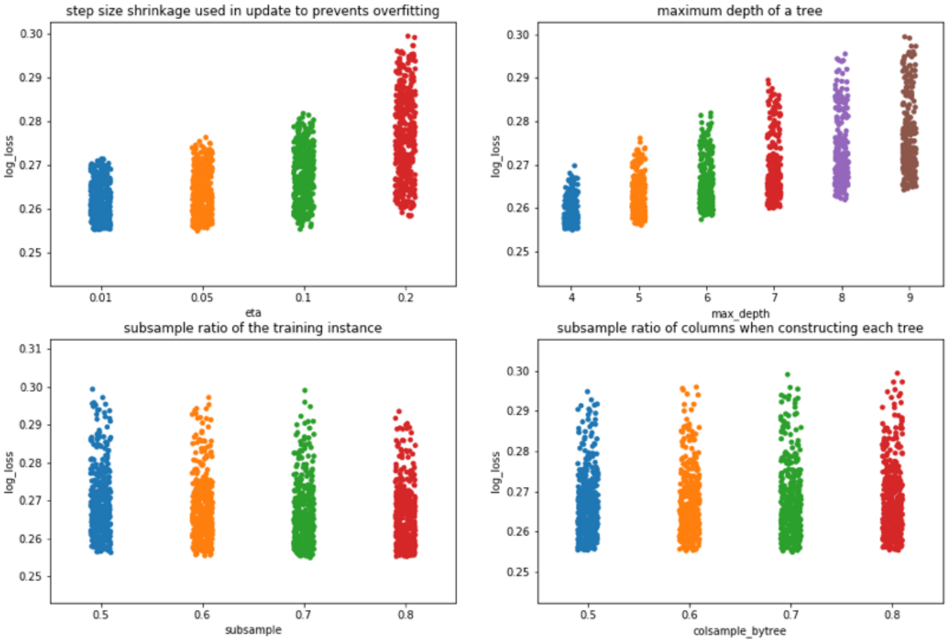


Figure 4.3: Entropy Loss of Boosted Tree Model

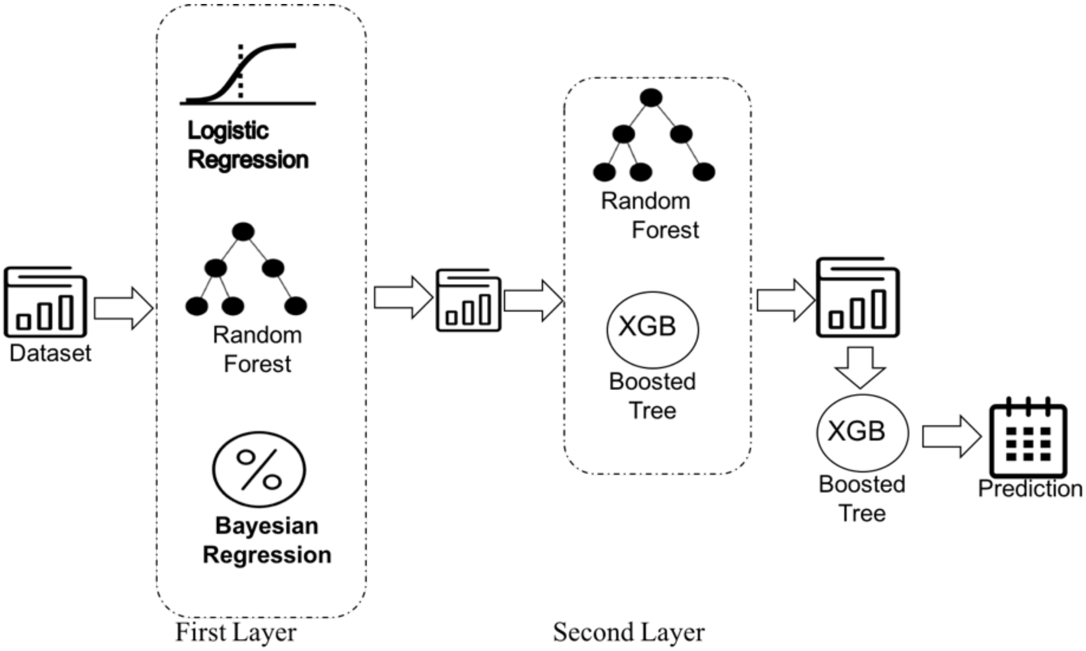


Figure 4.4: Diagram of implemented Model Ensemble

Table 4.3: Ensemble Model Performance Evaluation

First Layer	Cross-Entropy Los
Logistic Regression Classifier	0.6215
Decision Tree Classifier	0.4602
Random Forest Classifier	0.2970
NaÃŕve Bayes Classifier	0.5719
Second Layer	
Random Forest Classifier	0.3369
Boosted Tree Classifier	0.3465
Third Layer	
Boosted Tree Classifier	0.3482

in the first layer. This is obviously different with our assumption that performance of an appropriate ensemble should be better than any single model. But as we can see from the table 3, the loss of next layer is more like the average of previous layer. Thus, our question in this section is that why we get a different result in our experiments.

According to the theory proposed by Weigel[20], it is questionable that model ensemble can always enhance prediction skill. Assuming each single model has best performance in the combination, only when participating models are overconfident, the ensemble model can outperform the best participating model and this conclusion holds regardless of which combination algorithm is applied[20]. When combination of single models is well-dispersed, overall performance can be dropped down.

Chapter 5

CONCLUSION AND FUTURE OUTLOOK

In this paper, we discussed the collections of patients' self-report in the field of health care and how to clean and analyze self-report dataset.

Taking the advantages of zero threshold entry and simplicity, approaches, like questioning and questionnaire survey, are more acceptable for patients. However, it come with the tedious work of medical staffs. On the other hand, due to the manual data entry, data itself is not reliable. Researchers need to clean data before analyze any problem.

The usage of mobile applications can solve above-mentioned defects. While considering complicacy of patients group, any product usage barriers, like complex workflow or tiny text on screen, may chill patients' enthusiasm for the mobile application. The key factor is how to make mobile application have similar experience with traditional survey.

By inspecting the workflow of such applications, we found the authorization of first experience is comparably time-consuming. For patients who usually have unpredictable medical condition, it especially unreasonable to ask them to type text on small screen concisely. For optimization, we analyzed several authorization methods of mobile application and presented our approach using QR code. Based on proposed evaluation metrics, we found our method deduced both guiding time and authorization time. Patients are more likely to their pain level self-report.

Nevertheless, we need to talk about the trade-off between long-term credentials and short-term credentials. Even though we optimized the experience when patients are using application at first time, when authentication expires, patient still need the help of hospital staffs to re-login. This happens when a doctor want to extend target patient's observation period.

In the latter part of this thesis research, a pipeline for cleaning and analyzing mHealth self-report dataset is proposed. Taking E&H dataset as motivating example, we analyzed the feature distribution. Several characteristics of dirty data are introduced and tested in the dataset. We also cleaned the dataset for training them in the next section.

Evaluation of different machine learning models are made in the end of this research. By implementing algorithms, like random forest, boosted tree, using open source package, we found the performance limitation of single model when predicting if a person is obesity by his personal living habitation. Beside comparison of single model, we also compare the performance of ensemble model. Based on the entropy-loss, the graph shows that model ensemble has actually decrease our overall performance. The performance of ensemble model is more like the weighted average of all participated model. According to previous research, model ensemble only works when single model shows overconfident when making prediction.

In the future, extension of current pain check application is estimable. Rather than single functionality, a platform for self-report will further optimize data collection from both doctor and patients side. For such a platform, whether we can connect to hospital database and use patients' data directly instead of file importing is a key factor to increase compatibility. This platform should provide a more convenient way for doctors to check patients' status. the postoperative condition can be predicting by collecting enough real pain-level data. Similar pipeline we mentioned in this research can be applied.

BIBLIOGRAPHY

[1] Donna L Berry, Brent A Blumenstein, Barbara Halpenny, Seth Wolpin, Jesse R Fann, Mary Austin-Seymour, Nigel Bush, Bryant T Karras, William B Lober, and Ruth McCorkle. Enhancing patient-provider communication with the electronic self-report assessment for cancer: a randomized trial. *Journal of clinical oncology*, 29(8):1029–1035, 2011.

[2] Michael D Robinson and Gerald L Clore. Belief and feeling: evidence for an accessibility model of emotional self-report. *Psychological bulletin*, 128(6):934, 2002.

[3] Alain B Labrique, Lavanya Vasudevan, Erica Kochi, Robert Fabricant, and Garrett Mehl. mhealth innovations as health system strengthening tools: 12 common applications and a visual framework. *Global Health: Science and Practice*, 1(2):160–171, 2013.

[4] Heather Cole-Lewis and Trace Kershaw. Text messaging as a tool for behavior change in disease prevention and management. *Epidemiologic reviews*, 32(1):56–69, 2010.

[5] Caroline Free, Gemma Phillips, Lambert Felix, Leandro Galli, Vikram Patel, and Philip Edwards. The effectiveness of m-health technologies for improving health and health services: a systematic review protocol. *BMC research notes*, 3(1):250, 2010.

[6] Misha Kay, Jonathan Santos, and Marina Takane. mhealth: New horizons for health through mobile technologies. *World Health Organization*, 64(7):66–71, 2011.

[7] Marie-pierre Gagnon. A systematic review of factors associated to m-health adoption by health care professionals. In *Medicine 2.0 Conference*. JMIR Publications Inc., Toronto, Canada, 2014.

[8] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[9] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[10] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

[11] Dick Hardt. The oauth 2.0 authorization framework. 2012.

[12] Michael Jones, John Bradley, and Nat Sakimura. Json web token (jwt). Technical report, 2015.

[13] Roderick JA Little and Donald B Rubin. *Statistical analysis with missing data*. John Wiley & Sons, 2014.

[14] Jeremy Kubica and Andrew W Moore. Probabilistic noise identification and data cleaning. In *ICDM*, pages 131–138, 2003.

[15] Andy Liaw, Matthew Wiener, et al. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.

[16] Carolin Strobl, Anne-Laure Boulesteix, Achim Zeileis, and Torsten Hothorn. Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC bioinformatics*, 8(1):25, 2007.

[17] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12(Oct):2825–2830, 2011.

[18] Wei-Yin Loh. Classification and regression trees. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(1):14–23, 2011.

[19] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794. ACM, 2016.

[20] Andreas P Weigel, MA Liniger, and C Appenzeller. Can multi-model combination really enhance the prediction skill of probabilistic ensemble forecasts? *Quarterly Journal of the Royal Meteorological Society*, 134(630):241–260, 2008.

[21] Pieter-Tjerk De Boer, Dirk P Kroese, Shie Mannor, and Reuven Y Rubinstein. A tutorial on the cross-entropy method. *Annals of operations research*, 134(1):19–67, 2005.

[22] Kuan-Chieh Liao and Wei-Hsun Lee. A novel user authentication scheme based on qr-code. *JNW*, 5(8):937–941, 2010.

[23] SJ Dolin, JN Cashman, and JM Bland. Effectiveness of acute postoperative pain management: I. evidence from published data. *British journal of anaesthesia*, 89(3):409–423, 2002.

[24] Mark P Jensen, Paul Karoly, and Sanford Braver. The measurement of clinical pain intensity: a comparison of six methods. *Pain*, 27(1):117–126, 1986.