GRADUATE SCHOOL
EAST TENNESSEE STATE UNIVERSITY

East Tennessee State University
Digital Commons @ East Tennessee State University

Electronic Theses and Dissertations

Student Works

12-2019

Design and Development of a Comprehensive and Interactive Diabetic Parameter Monitoring System - BeticTrack

Nusrat Chowdhury
East Tennessee State University

Follow this and additional works at: https://dc.etsu.edu/etd

Part of the Computer and Systems Architecture Commons, and the Health Information Technology Commons

Recommended Citation

Chowdhury, Nusrat, "Design and Development of a Comprehensive and Interactive Diabetic Parameter Monitoring System - BeticTrack" (2019). *Electronic Theses and Dissertations.* Paper 3646. https://dc.etsu.edu/etd/3646

This Thesis - unrestricted is brought to you for free and open access by the Student Works at Digital Commons @ East Tennessee State University. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of Digital Commons @ East Tennessee State University. For more information, please contact digilib@etsu.edu.

Design and Development of a Comprehensive and Interactive Diabetic Parameter Monitoring System - BeticTrack

A thesis

presented to

the faculty of Computer and Information Science

East Tennessee State University

In partial fulfilment of

the requirements of the degree

Master of Science in Computer and Information Sciences

by

Nusrat Jahan Chowdhury

December 2019

Ferdaus Kawsar, PhD. Chair

Phil Pfeiffer, PhD.

Brian Bennett, PhD.

Stephen Hendrix, M.S.

Keywords— Diabetic android application, Blood glucose tracker, meal tracker, wearable activity tracker, HealthCare IoT

ABSTRACT

Design and Development of a Comprehensive and Interactive Diabetic Parameter Monitoring System – BeticTrack

by
Nusrat Jahan Chowdhury

A novel, interactive Android app has been developed that monitors the health of type 2 diabetic patients in real-time, providing patients and their physicians with real-time feedback on all relevant parameters of diabetes. The app includes modules for recording carbohydrate intake and blood glucose; for reminding patients about the need to take medications on schedule; and for tracking physical activity, using movement data via Bluetooth from a pair of wearable insole devices. Two machine learning models were developed to detect seven physical activities: sitting, standing, walking, running, stair ascent, stair descent and use of elliptical trainers. The SVM and decision tree models produced an average accuracy of 85% for these seven activities. The decision tree model is implemented in an app that classifies human activity in real-time.

Copyright by Nusrat Jahan Chowdhury 2019

All Rights Reserved

DEDICATION

I dedicate this thesis to my friend, Osayamen Omigie, for his unconditional help, support and encouragement throughout the research.

I dedicate this thesis to my friend Faizan Hasan, for his constant support and showing positive attitude in all my decisions.

I dedicate this thesis to my friends Jean-Marie Nshimiyimana and Tahsin Rezwana, for believing in me more than myself.

ACKNOWLEDGEMENTS

I wish to express my earnest gratitude to my thesis advisor and committee chair Dr. Ferdaus Kawsar, for introducing me to this interesting area of research and for believing me worthy of this research. I am thankful for his constant support, feedback, guidance, and patience throughout the research.

I also wish to extend my deepest gratitude to my thesis advisor and committee member Dr. Phil Pfeiffer for his sincere concern and care about my research progress. I am thankful for his prompt and insightful responses in editing my thesis draft document.

I would like to express my appreciation to Dr. Brian Bennett for agreeing to serve on my thesis committee and as an editor for this report. I appreciate Dr. Bennett's support and patience throughout the thesis completion process. I am thankful for his prompt and thorough revisions on the thesis draft.

Special thanks to Dr. Asad Hoque and Stephen Hendrix for agreeing to serve on my thesis committee.

Finally, I express my gratitude to my family for being a constant support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

Diabetes is a chronic disease that damages a body's natural blood sugar/glucose processing. It can lead to life-threating conditions like heart failure, stroke, kidney failure, limb amputation, and blindness. In 2016, 2.2 million deaths were caused by high blood glucose worldwide, 1.6 million of which were due to diabetes [1]. According to the World Health Organization, the number of people with diabetes worldwide has quadrupled since 1980 [1].

Diabetes is a manageable and preventable disease. Maintaining a healthy diet, regular physical activity, and a normal body mass index can prevent the onset of type 2 diabetes (T2D). Additional factors like balanced blood sugar level and timely medication can help T2D patients avoid further complications.

Currently, most T2D patients fail to maintain the controlled and prescribed lifestyle that authorities recommend. According to [2], at least 45% of T2D patients fail to control blood glucose within the recommended range (HbA1c <7%). Reasons for poor treatment adherence include a lack of integrated healthcare in existing health care systems; demographics like age, education, and income; and the perceived burden in obtaining and taking medications [2].

Healthcare information technology interventions have been shown to reduce T2D patient failures to maintain healthy lifestyles. One such intervention, mHealth, is "is the generation, aggregation, and dissemination of health information via mobile and wireless devices and the sharing of that information between patients and providers" [3]. According to [4], smartphone users will account for roughly two-thirds of new global connections. The ease with which data can be collected through smartphone-accessible sensors and then accessed by health providers via the Internet promises to make mHealth an integral part of modern health care.

Contemporary apps, however, fail to enable mHealth-based diabetes interventions. Almost 100,000 health-related applications are available in the Google Play Store and the Apple App Store [5]. According to [6], more of these apps target diabetes than any other illness. Most diabetes-related apps focus exclusively on patient self-care management, even though communication between clinician and patient is important for chronic diseases like diabetes.

Incorporating telemedicine into mHealth protocols can address the lack of support for patient-clinician communication in contemporary apps. Telemedicine uses telecommunication to provide remote support to target patients. According to [7], telemedicine can support the treatment of diabetes by providing clinicians with real-time access to patient physiological data, including levels of blood glucose physical activity, and carb intake; laboratory and test data, including imaging data; and patient history, including medications, symptoms, and previous doctors' visits.

This research was concerned with the development of a system that integrates telemedicine and mHealth in the service of the following goals:

- Provide T2D patients with a way to record and review behavioral data related to diabetes management, including daily carbohydrate intake.
- Provide patients with a way to set and receive reminders about the actions they need to take to manage their illness, including what medication to take and when to take them.
- Provide patients with a means of recording physiological data in real-time, passively, through wearable insole devices and classifying activity dynamically.
- Provide physicians and other clinicians with secure, Internet-based access to their patients' medical records, so these physicians can monitor patient progress and communicate with their patients.

The resulting system, BeticTrack, enables patients to record their blood glucose and hemoglobin levels and daily carbohydrate intake in a secure, cloud-based database that their physicians can access. It allows patients to set reminders to measure and enter data in the system. It also represents a step towards the automated detection of patient activity and real-time logging of that activity, without patient intervention.

Among the goals of BeticTrack system, this research particularly focuses on building an automated physical activity module to track a patient activity. BeticTrack collects real-time sensor data via Bluetooth and classifies seven types of activities: sitting, standing, walking, running, stair ascension, stair descension, and using an elliptical. To classify the seven activities, this research initially collected real-time data from four human subjects and applied two machine learning techniques—support vector machines (SVM) and decision tree learning—on the dataset. The SVM provided an average accuracy of 80%, and the decision tree provided an average accuracy of 80%-90%. Because of its higher accuracy, the developed decision tree model is implemented in the BeticTrack system to classify real time activity of a human based on the seven activities. Using the implemented decision tree model within BeticTrack, further testing was done to measure classification time. From the analysis, it can be concluded that, among the seven activities, the system classifies sitting and running, classifies walking and using an elliptical with some degree of error, and fails to identify stair ascent, stair descent, and standing. The classification error for stair ascent, stair descent and standing may be reduced by collecting data for a longer time period from different age groups and regenerating the machine learning models from the larger dataset.

2 RELATED WORK

2.1 Self-Management Applications for Diabetes Intervention

The number of self-management applications for diabetic care management is increasing rapidly due to the availability of mobile devices and their ability to connect with IoT devices (viz. Table 1). Most assistive systems are described as self-monitoring, meaning that patients can provide their records and get feedback from the applications. Self-monitoring applications are often patient-centric and may not support communication among patients, doctors, and healthcare providers.

In [8], [9], and [10], Martinez et al., Brzan et al., and Viazie et al. review self-monitoring applications for iOS and Android devices. Of the more than one hundred such apps, these reviewers identify only a few that they deem as acceptable to some degree.

Micro sugar is a consultation and information app for diabetic patients in China. It focuses on blood glucose (BG) health. A patient can use the app to learn about diabetic health and consult a doctor if necessary. However, language barriers precluded identifying the app's communication standard [11].

The BlueStar Diabetes app provides dietary suggestion and a BG level alert. It supports connections to wearables and to electronic health records (EHR) [12,13].

mDiab primarily monitors BG and provides BG level alerts [14-17]. mDiab does not support patient activity detection, intervention schedule reminders, food carb estimation, and doctor-patient communication.

Dbees monitors BG, provides BG level alerts, adds dietary a calculator, and provides healthcare-related suggestions [18,19].

NexJ HealthCoach is a secured health app that enables consultation with a health coach. It does not account for any treatment parameters, like BG, exercise, and food intake [20,21].

Diabetic's Diary [22] and Diamedic [23] account for key treatment parameters, including BG, carb intake, and exercise. These apps require manual input of all these records. They do not, moreover, support communications with physicians or clinicians.

Table 1 – Features of a selection of commercially available mobile apps

Apps	Glucose	Food intake	Exercise	Physician Interaction	Medication Reminder	Other features
BlueStar [12,13]	Yes	None	Can sync with wearable devices	None	yes	
mDiab [14-17]	Yes	None	Manual input	Email options to interact	Yes	
Dbees [18,19]	Yes	Yes – calculates carbohydrates	None	Can share data and test result	Yes	
NextJ HealthCoach [20, 21]	Yes	Yes – keeps a log of meal	Step counter	None	None	Reward points based on improvement, notifications
Diabetic Diary [22]	Yes	None	None	None	Yes	Blood pressure, weight log, notifications
Glucose Buddy [24]	Yes	Yes	None	None	None	Weight, Blood pressure, A1C
Glucose Comp [25]	Yes	None	None	None	Yes	Weight, notes, reminders
Glucose Tracker26]	Yes	None	None	Via email	None	Records can be exported in .csv
GlucoSweet [27]	Yes	Yes – calculates carbohydrates	None	None	None	Data visualization can be exported in CSV
Glucose Wiz [28]	Yes	None	None	None	None	Data visualization, weight, data sharing via social media and email
SugarPal [29]	Yes	None	None	None	None	Data visualization and save option

Glucose Buddy is similar to Diary and Diamedic. It also supports activity detection based on another third party application [24].

Glucose Companion is intended for patient-centric self-management. It has features for BG and weight tracking, notes, and graphic visualization, and supports downloading files from the app [25]. Glucose tracker, GlucoSweet, Glucose wiz, and SugarPal are similar to Glucose Companion; all four focus on self-management [26-29].

Goyal et al. (2017) developed a self-management app, Bant, for adolescent type I diabetes patients [30]. The app's primary features include BG monitoring and providing interventions. Goyal et al. (2016) also describe Bant-II, a second app for type II diabetic patients [31]. Bant-II's key features include self-monitoring of BG, dietary intake, and physical activity. The authors proposed the use of a Bluetooth device to monitor BG, along with Jawbone UP24, a wearable monitor, and mobile phone sensors to detect patient inactivity. The target activity for Bant II is 'walking'. Bant-II treats more than 5000 steps on a given day as an adequate amount of walking, otherwise, it treats the user as sedentary. Bant–II also encourages patients to set progressive goals. Neither Bant nor Bant II supports patient-clinician communication.

A few apps enable physicians or coaches to interact with patients. One, ActiveAgeing, includes a setup phase that sets a degree of interaction between patient and doctor. A web app then allows an assigned doctor to monitor patient progress [32]. ActiveAgeing, however, does not collect any wearable sensor data for exercise or activity monitoring.

DiaFit supports BG monitoring, medication tracking, dietary intake, and the use of wearable devices to track activity. It supports the use of a cloud portal to store patient data, although no direct communication between patient-physician is reported in the research [33]

A French app, DIABEO, accounts for BG input, carbohydrate intake, and physical exercise. DIABEO requires active patient input for these data: it provides no support for direct input from wearable sensors. The app is modeled as a teleconsultation module so that secured messages can be sent to authorized clinicians [34].

One Drop, another healthcare app, tracks BG and reminds users about medication dosage. It uses wearable sensors (HealthKit, Google Fit) to collect information on users' activities. The authors fail to specify which activities are tracked and the extent of the tracking. One Drop does not support clinician-patient interaction. [35].

The DialBetics app includes options for entering data on blood glucose, blood pressure, body weight, and step count [36]. An evaluation module analyzes the data collected. DialBetics collects voice prompts as data input and can send prerecorded instructions based on the analysis module. The app's activity module focuses only on step counts and does not track other forms of physical exercise, all of which can be important for a diabetic patient.

Pustozerov et al. [37] describe an app that enables physicians to view patient input. The app supports the tracking of a patient's BG, diet, insulin dosage, and physical exercise, and stores laboratory test data. This app does not support the use of wearable IoT devices to collect patient activity data.

In [38], Ryan et al. describe Intelligent Diabetes Management (IDM), a system that incorporates a smartphone app and a website. IDM does insulin bolus calculations and serves as an electronic diabetes diary. It enables communication between patients and physicians, which, according to the authors, helps patients to change their management habits. IDM does not support the use of physical activity monitoring systems to gather data on patient activity.

DiaCert uses a pedometer device to track a patient's steps automatically. Physicians can use a DiaCert exercise record, along with a patient's BG, to suggest a required HbA1c level. DiaCert does not support the monitoring of dietary intake [39].

MyDay, an IoT based healthcare app, uses Bluetooth BG meters, and mobile sensors to collect real-time data on type 1 diabetes patients. It uses this information to provide patients with feedback. MyDay does not support clinician access to patient data [40].

2.2 Wearable Sensor Use in Activity Detection

2.2.1 Physical Activity in Diabetic Treatment

Physical activity is the movement of the human body by means of skeletal muscle contraction, e.g., walking, sports, and household chores. Physical exercise is a repetitive planned physical activity focused on improving physical fitness [41]. The standard measure of physical activity or exercise depends on the duration of activity and intensity: low, moderate, or high.

Moderate to high physical activities are typically prescribed for diabetic patients to maintain BG and to avoid complications like cardiovascular diseases. When diabetic patients' health and muscle strength are questionable, low-intensity exercises can avoid possible harms and stresses caused by vigorous activities [42,43]. Wen et al. [44] found that a low-intensity activity, such as a 15-minute walk per week, has a positive impact on T2D patients and can reduce the risk of mortality rate to 14%.

According to Hamasaki et al. [45], daily physical activity and movement can reduce T2D-related risks to patient health. Hamasaki et al. reviewed physical activities that incur less risk to T2D individuals, including mowing lawns, stair climbing, gardening, cleaning, cooking, and walking. These researchers found that walking has the most pronounced effect on a person's body. The effect of walking depends on the duration walked and the pace; according to [45] (Table 1, p. 4). the metabolic equivalent (MET) values for very slow, slow and moderate walking are 2, 2.8, and 3.5, respectively. The MET is a measure of the intensity of physical activity, expressed as the amount of oxygen intake relative to the normal resting position (sitting) [46].

Researchers have also established that inactivity can adversely affect T2D patient health. Biswas et al. [47] found that a sedentary lifestyle—one that includes longer periods of inactivity like sitting—can counteract the positive effects of physical exercise. According to the American Diabetes Association (ADA) [48], a diabetic patient should not sit more than 1.5 hours at a time. Similarly, physicians encourage patients to reduce sedentary hours in daily life.

2.2.2 Activity Recognition

Activity recognition is challenging due to the diverse and incomplete nature of data obtained from motion sensors and individual variances in patterns of human activity. Machine learning techniques have recently become effective in solving activity recognition problems due to their ability to extract information from devices that sense human motion.

State of the art research in machine learning is concerned with identifying the best machine learning practices in human activity detection (HAR). Traditional methods include models like the Random Forest Model (RF) [49], the Hidden Markov Model (HMM) [50], Support Vector Machines (SVM) [51], and the Decision Tree Model [52].

One of the two classes of models adopted in this research, SVM, is a supervised learning algorithm for classification or regression problems. The SVM algorithm classifies data in n-dimensional feature space by generating hyperplanes between pairs of classes. The objective is to choose hyperplanes with a maximum margin between the classes. SVM can be applied to linear (i.e., readily separable) datasets and nonlinear (i.e., hard-to-separate) datasets alike.

The other, decision tree algorithms, divide datasets into a hierarchical tree structure while eliminating factors that fail to impact any decision. Decision trees generate a set of decision strategies based on the dataset to provide a categorical outcome.

2.2.3 Insole Devices in Health Care Research

Andre et al. [53] conducted a HAR classification with a wearable insole device containing six ground contact force (GCF) pressure sensors. The authors studied six activities: walking on a level surface, walking upslope, walking downslope, sitting, ascending stairs, and descending stairs. Data were collected from 11 individuals for all activities. An RF learning model was used for the HAR. The model's accuracy ranged from 81.8% to 93.84%. This research only used one sole size (US men 8). The authors did not use the models they generated to classify activity data in real-time.

Chinimilli et al. [54] used a smart shoe containing four GCF sensors and four thigh-mounted inertial measurement units (IMUs) to detect six types of human activities: sitting, standing, walking, jogging, ascending stairs, and descending stairs. The authors developed an intelligent fuzzy inference algorithm using an RF model. The resulting model's accuracy ranged from 47.56% to 100%. While the authors' system does not provide any classification mechanism for real-time data, it identifies and notifies users about transitions between activity types.

Hegde et al. [54][55] conducted real-time activity detection experiments using a pair of Bluetooth Low Energy (BLE) insole devices. The devices consisted of three pressure sensors, three accelerometers, and three gyroscopes. The pressure sensors and accelerometers sampled data at a rate of 50 Hz. The authors collected sitting, standing, walking, and cycling data from 4 human subjects and fed this data to an SVM and Artificial Neural Network (ANN). The resulting machine learning models exhibited an accuracy of 96.9%. SmartStep, an Android application that the authors developed that incorporated these models, subsequently achieved an average real-time classification accuracy of 95.4%.

Nguyen et al. [56] conducted research to classify human activity data collected from a smart insole device. The device contains eight plantar-pressure sensors. The authors compared the HAR using three machine learning algorithms: SVM, Decision Tree, and K-Nearest Neighbor (KNN). They found that the KNN algorithm outperformed the other two, yielding classification accuracies of 98.11%, 98.11%, 99.73%, 100%, and 99.73% respectively for walking on flat surfaces, descending and ascending stairs, and descending and ascending inclines.

2.3 Summary

Existing applications provide a range of functions that support best practices for managing diabetic patient health. Most applications focus on blood glucose monitoring, calorie/carbohydrate intake, and/or physical activity. Some focus on patient-centric monitoring. Others incorporate physician participation. However, none of the commercially available mobile/web application systems for T2D patients that this review identified supported the five key elements of T2D treatment routines: i.e., the monitoring of BG, meals, and exercise, along with medication reminder and patient-physician interaction. Moreover, while researchers have used HAR to identify real-time human activities, no identified study implemented or integrated real-time HAR for specific patient groups, including T2D patients

3 METHODOLOGY

The goal of this study was to monitor physical activity in a manner that would help T2D patients communicate activities to their clinician. This research investigated the use of sensor data to monitor physical activity in T2D patients, as part of an effort to help those patients and their clinicians manage patient health. To this end, a system was developed, called BeticTrack, that uses sensors to collect patient movement data, processes the data in real-time to classify physical activity, calculates the duration of daily human activity, logs this information, and classifies subsequent physical activity in real-time.

The research primarily focused on developing several machine learning models to classify seven human activities real-time by collecting human data through wearable insole devices. The most effective of these was then incorporated into the app to classify and store real-time activity information.

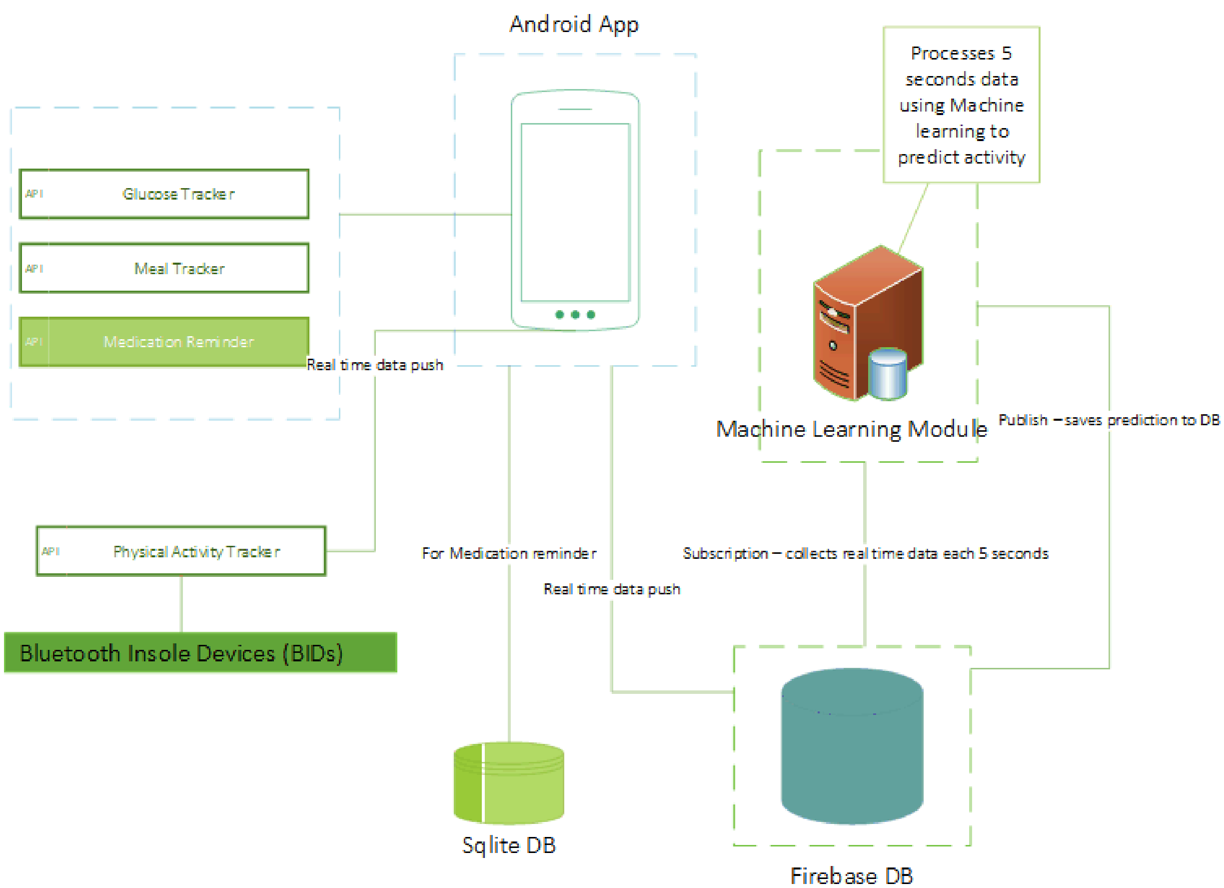

Figure 1 - System Architecture

3.1 System Design and Architecture

The BeticTrack system was developed as an Android mobile application with an SQLite database back end (Figure 1). The BeticTrack app, written in the Java programming language, supports all versions of Android. It consists of four modules: a carb intake tracker, which patients can use to log daily food intake; a blood glucose tracker, which records patient-provided glucose data (fasting and prandial); a medication reminder, which patients can configure and which, by default, contains no sensitive health data; and a physical activity tracker, which collects patient activity data from a BID via Bluetooth. Both the user interface and architecture for BeticTrack are original contributions resulting from this study.

3.1.1 Carb Intake Tracker (CIT)

Figure 2a shows the CIT module user interface (UI). This module enables users to enter their daily meals in the app. Carbohydrate levels are crucial for T2D patients because the body produces sugar from carbohydrates. Uncontrolled intake of carbohydrates can cause a BG level to rise at an alarming rate. Providing patients with information about their daily carbohydrate intake can help them control their diet. A patient can add his/her breakfast, lunch and dinner menus to the system through the form. Data are stored in real-time in Firebase cloud database. A future version of this module will include carbohydrate calculations for users.

3.1.2 Blood Glucose Tracker (BGT)

Figure 2b shows the BGT module UI. This module provides a platform for recording and tracking patient BG levels, along with A1C levels during postprandial and fasting periods. The BG measurement standard is the Hemoglobin A1c (HbA1c/A1c). According to [43], the level of A1C should be less than 7%. The records are stored in the Firebase cloud.

3.1.3 Medication Reminder (MR)

Electronic reminders have been found to help patients adhere to schedules for taking medication and taking BG readings: two interventions that are crucial for diabetes management [45]. To help patients adhere to their prescribed regimens, the MR module allows patients to set reminders for when to take medication and BG readings and update their BG record in the portal. The SQLite database is used to store medication and alarm task for these two modules.

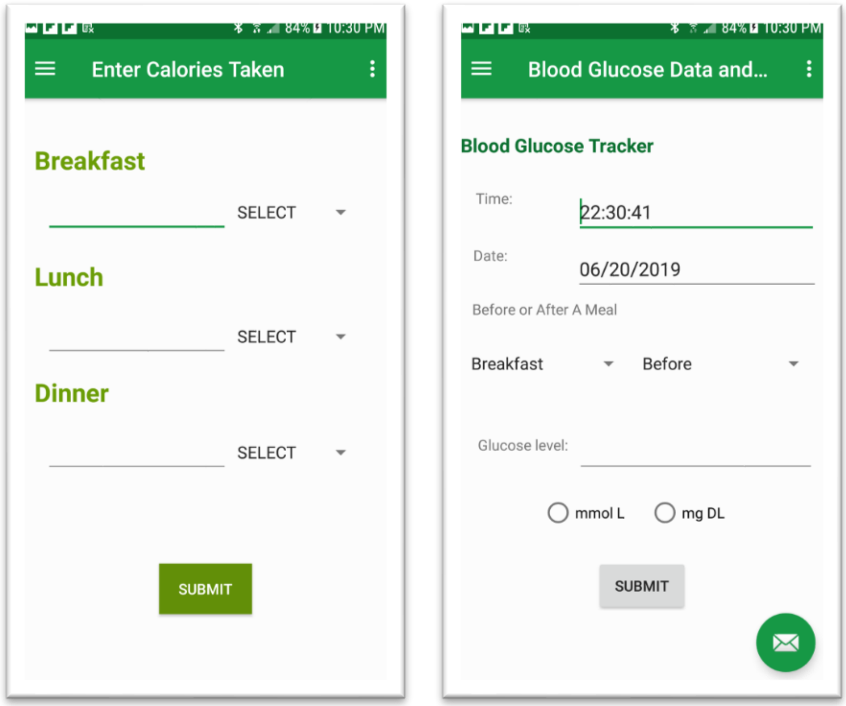

Figure 2 - a) Carbohydrate intake tracker and b) blood glucose tracker

3.1.4 Physical Activity Tracker (PAT)

Physical activity, an essential intervention for type 2 diabetes, helps T2D patients to control their BG level. T2D patients are insulin resistant: their body does not produce enough insulin to process blood glucose. In these patients, muscles can process glucose during exercise.

This research’s primary focus is to provide T2D patients with a means of tracking their physical activity in real-time. The PAT module tracks a patient’s level of activity over time, in order to calculate their caloric expenditure. It uses a pair of wearable insole devices to track patient activity data via Bluetooth. The app detects movements that impact a patient’s BG level and other health indicators.

3.1.5 Persistent Data Stores

The four Android modules are supported by SQLite and Firebase databases. SQLite, a small, simple, and fast relational database, is embedded in the BeticTrack app. BeticTrack's MR module, which communicates strictly with the patient, uses a local database to store medicine names and alarm tasks; all medication alarms are managed locally, in the Android device.

Firebase is a NoSQL database for mobile and web application development. It provides a cloud-based infrastructure that supports concurrent, real-time access to data sets by multiple applications and data synchronization between devices. It enables the fast querying of data sets, regardless of set size. It supports event-based subscription: applications accessing a Firebase database can subscribe to any data table and receive notifications when subscribed-to tables are updated. It also provides a module that authenticates different users, along with their user roles.

BeticTrack uses Firebase to store credentials for properly authenticating users and authorizing their actions. BeticTrack supports four types of user roles: one for patients; one for doctors; one for observers, who are authorized to view or monitor patient records; and one for administrators, who are authorized to read and modify data for the other three types of users. Figure 3 shows examples of BeticTrack screens for logging users into a system, using their e-mail addresses and passwords.

3.2 Tracking Physical Activity

The Android application and its supporting databases provided a platform for supporting this research's primary focus: the use of foot sensors to gather and interpret data about a patient's activity. This research sought to use sensors to identify seven types of patient activity: sitting, standing, walking, running, ascending stairs, descending stairs, and using an elliptical trainer. To

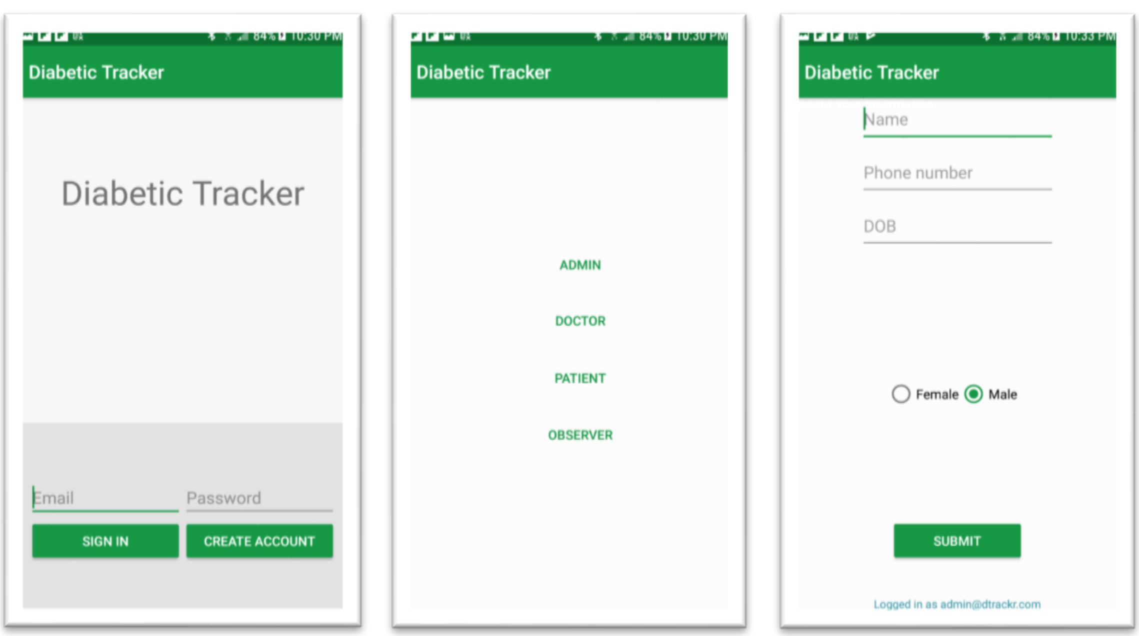

Figure 3 - User authentication screen, user type selection, and profile update

Figure 4 - Wearable Insole Device

this end, a machine learning module was created to analyze and classify activity based on real-time data.

3.2.1 Wearable Insole Device

A battery-operated and Bluetooth-enabled smart insole device was used to collect information about patient activity (Figure 4). Each Bluetooth Low Energy (BLE) Insole Device (BID) has a unique identifier name and a MAC address. These insoles come in EU shoe sizes 37-44. They can be worn in any shoe by replacing the original shoe insole. The smart insole provides four Soft Pressure Sensors (A, B, C, D) and three built-in accelerometers (a, b, c). Data from these sensors can be collected using any Bluetooth device with an operating system (OS) of iOS 8.4, Android 5.0, or above.

3.2.2 Application Development for Data Collection using Bluetooth

An Android application was developed to collect insole sensor data, using the insole provider's application program interface (API). The API provides data access functions and enables communication between the Bluetooth and Android devices.

BeticTrack's workflow for activating a BID is depicted in Figure 5. When the BeticTrack app starts, it scans for nearby BIDs until timing out. The app matches the scan's result with information in its database, attaching only user-owned devices to the session. When the app detects user-provided BIDs, it updates its UI, prompting for permission to connect the devices. Figure 6 shows screenshots of representative device scans.

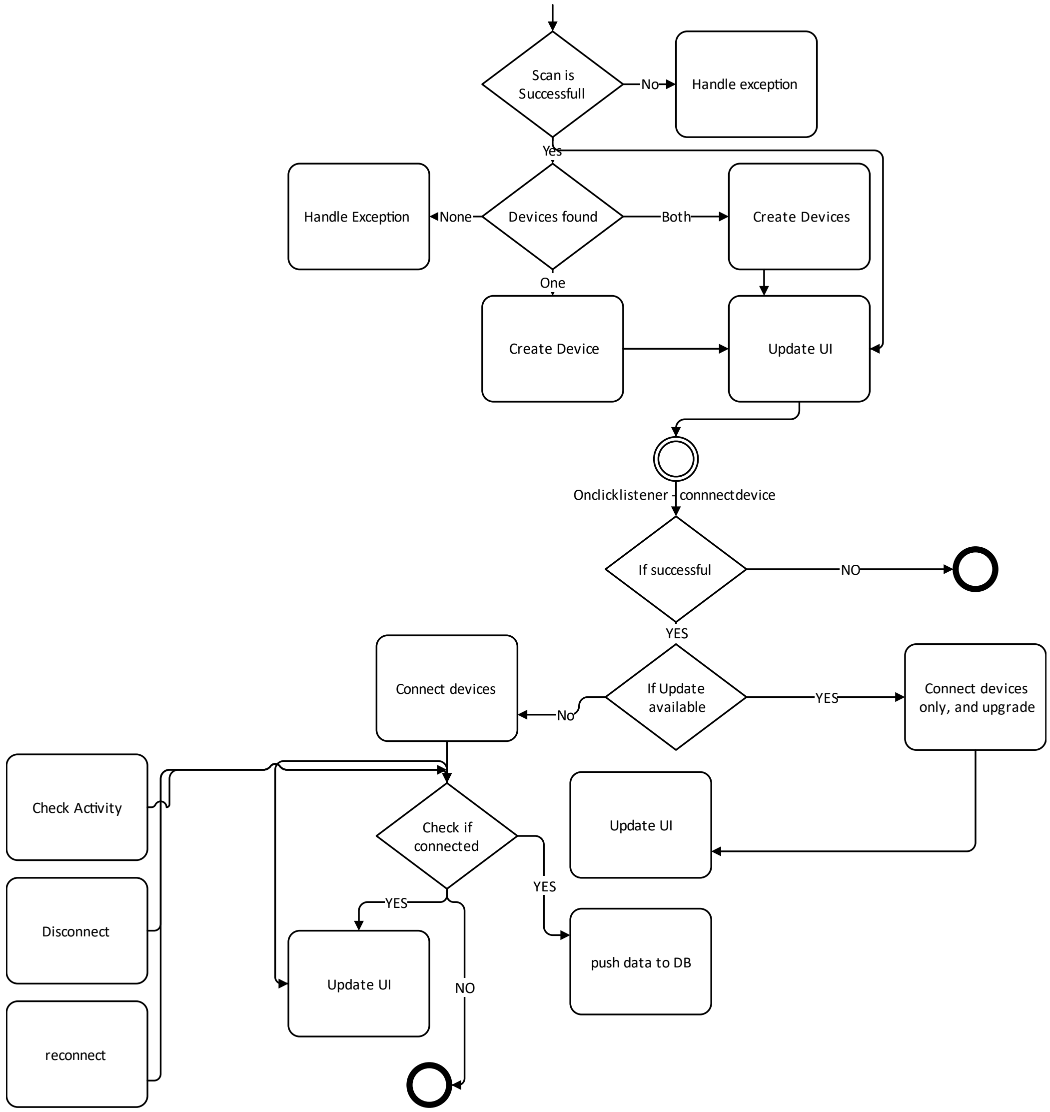

Figure 5 - The data collection application's work flow

If the scan finds both devices, the UI prompts the user with "connect left" and "connect right" buttons. If a user clicks to connect, the application first checks for BID firmware upgrades, per the manufacturer's requirements. The device then connects to the BIDs and pushes data real-time data to the database, updating the UI.

To ensure improved the app's performance and enhanced responsiveness in response to the BIDs real-time connection state change, a unique UI design was built using Java asynchronous programming. Figure 6 shows screenshots of the UI based on the BIDs'

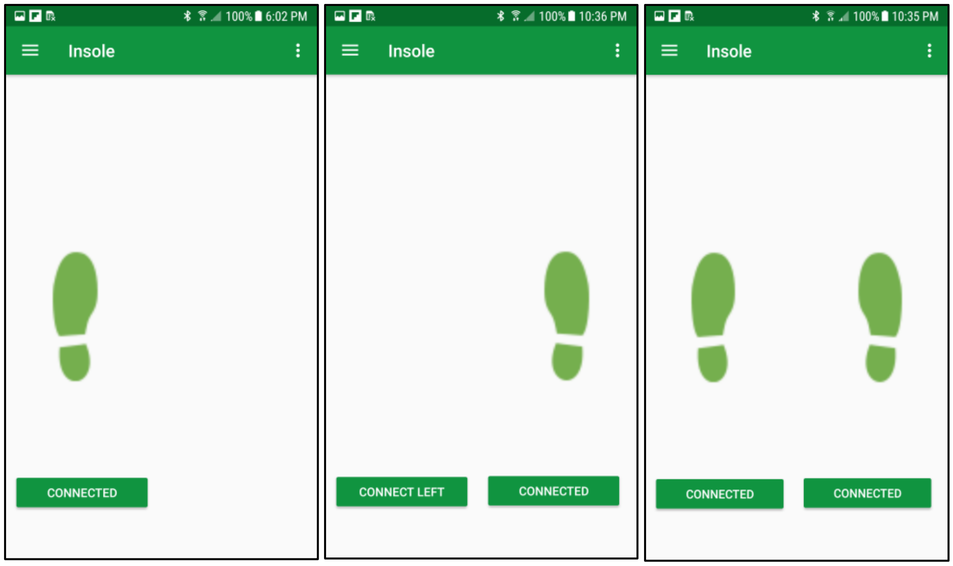

Figure 6 - UI with left BID connected, right BID connected, and both BIDs connected

connection states. The first screenshot represents a state where only the left BID is discovered by the device and connected. The second screenshot displays the state when both BIDs are discovered by the app but the only the right BID's connection succeeded. The third screenshot represents the normal state where both BIDs are discovered and connected.

Once connected, the BIDs can send two types of real-time data to the app. The one, continuous accelerometer data, characterizes device motion in three-space, using X-, Y- and Z-axes. This data is sent continuously, at a frequency of 12 Hz. The other type, event-based sensor data, is sent when the four pressure sensors sense any touch or pressure.

The app relays saved data collected from the BIDs in real-time to Firebase database. The Firebase data was used initially for data analysis. For this phase, the data were deleted from the database after each data collection phase.

3.3 Activity Detection

In keeping with work by Wen et al. [44] and Hamaski et al. [45], this research focused on routine, low-intensity activities like walking, stair ascending, and stair descending. For completeness, it included running as a moderate to a high form of activity and elliptical training as a form of equipment-based exercise. All five activities require foot movement, which supports

Figure 7 - Standing postures. a) normal b) cross legged

the use of BID as a basis for activity identification. Additionally, in keeping with findings by researchers like Biswas et al. [47], monitoring of patient activity was extended to sitting and standing: two sedentary activities that, in excess, are deemed risky to patient health.

3.3.1 Data Collection

Data were collected for seven types of activities:

- Walking on flat surfaces.
- Running on a running track and a treadmill.
- Sitting on standard couches, while maintaining normal sitting postures (Figure 7).
- Standing on an even surface in common standing positions, e.g., when switching legs or changing from straight-legged to cross-legged (Figure 8).
- Ascending straight, landing-free staircases, at varying rates of speed (Figure 9a).

Figure 8 - Sitting postures

Figure 9 - a) stair ascension, b) stair descension

- Descending straight, landing-free staircases, at varying rates of speed (Figure 9b).
- Using two different models of elliptical trainers (Figure 10).

Sensor data was collected from four volunteers, two males and two females, each of whom did all seven activities. Data collected while walking on landings between flights of stairs were disregarded to ensure the integrity of stair data. Data, as they were collected, were stored in the Firebase database. After each activity, the dataset was exported. Data was then deleted from the database to maintain the volunteers' confidentiality and to ensure no data was compromised.

Figure 10 - Volunteer using elliptical for data collection.

3.3.2 Data Preprocessing

Data preprocessing is the preparation of a raw data set for subsequent analysis by converting this data into a usable format. As the data collected were from different individuals in real-time, a cleaning step was needed to condition the raw data for analysis. After this step,

Table 2 – Collected data attributes

Name	Description
Timestamp	the timestamp of event in milliseconds
Ax, Ay, Az	X-, Y-, and Z-axis data for accelerometer A
Bx, By, Bz	X-, Y-, and Z-axis data for accelerometer B
Cx, Cy, Cz	X-, Y-, and Z-axis data of accelerometer C
A	heel sensor data in Boolean
B	feet palm sensor in Boolean
C	feet palm sensor in Boolean
D	feet palm sensor in Boolean
Impact	force exerted on the sole, scaled from 0-7 based on the force

feature extraction was done to reduce the dataset to a more manageable size. Table 2 shows the attributes included in each collected data set.

3.3.2.1 Data Cleaning

The primary considerations for data cleaning were removing outliers and irrelevant data, accounting for missing data (NaN values), converting strings to numeric values, and labeling the data based on activities.

Outliers. Outliers are data that deviate from the original data; e.g., before the start of the running activity, standing activity data can be considered as outliers. The first step of data cleaning involved the removal of the first and last few seconds of data from test runs. This eliminated sensor readings from before activities started and after activities ended. These irrelevant readings were collected due to the need to start recording data before visually confirming that both left and right insoles were sending data (Figure 11).

Irrelevant data. The BID SDK includes built-in data—e.g., sole data, varus—that were irrelevant to the research and removed from the dataset.

Missing data points. The shoe sole SDK returns an empty (NaN) value for sensors that fail to generate readings during a given 12 Hz reading cycle. For example, the A, B, C, and D sensors only generate values when a person is stepping. Figure 12 shares a snippet of a dataset containing NaNs. For this research, NaNs were retained for use in the machine learning model.

String to numeric conversion. True and false sensor values were converted to 1 and 0 for cleaning and machine learning purposes.

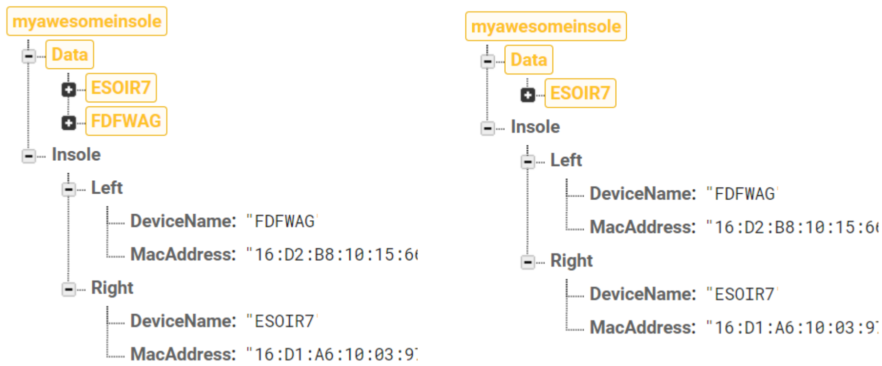

Figure 11 - Firebase console, a) both devices are b) one is - connected and active

3.3.2.2 Feature extraction

Feature extraction is the extraction of meaningful data representations from a raw and possibly redundant dataset. Periodic activities, like those sampled in these trials, typically exhibit repeating patterns. Feature extraction can introduce invariance and provide compact, quantitative characterizations of the patterns.

For this research, three types of features were extracted from sensor data recorded over ten-second (10s) intervals:

- Mean, mode, a median of accelerometer data. For each 10s interval, each accelerometer axis's mean, mode, and median values.
- The elapsed time between consecutive shoe sole sensor events. Time differences between consecutive sensor events; these were used to distinguish between activities like walking and running, the former having less frequent events than the latter.

1740	1551818359555	-2.0	-2.0	33.0	-2.0	-1.0	33.0	-2.0	-2.0	33.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	6X1CSV
1741	1551818359404	-2.0	-1.0	33.0	-2.0	-2.0	33.0	-2.0	-1.0	33.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	6X1CSV
1742	1551818359333	-3.0	-1.0	33.0	-2.0	-1.0	33.0	-2.0	-2.0	33.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	6X1CSV
1743	1551818359312	-2.0	-2.0	33.0	-2.0	-2.0	33.0	-2.0	-1.0	33.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	6X1CSV
1744	1551818359225	-2.0	-1.0	33.0	-3.0	-2.0	33.0	-2.0	-2.0	33.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	6X1CSV
1745	1551818488597	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0	0	0	0	6	6	0	102	2	BBJE8I
1746	1551818488304	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1	0	0	0	6	6	0	102	2	BBJE8I
1747	1551818488204	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0	0	0	0	2	12	0	104	7	6X1CSV
1748	1551818487910	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0	1	1	0	2	12	0	104	7	6X1CSV
1749	1551818487718	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0	0	0	0	6	6	0	101	2	BBJE8I
1750	1551818487378	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1	1	0	0	6	6	0	101	2	BBJE8I

Figure 12 - Timestamps containing NaNs

- Mean, mode, the median of the elapsed times.

Feature extraction yielded 43 features. These included mean, mode, and median values for the A, B, and C sensors; the number of events for all sensors; the number of true events for the A, B, C, and D sensors (true when a sensor touches the ground); the mean, mode, and median impacts; and mean, mode, and median time differences for the A, B, C, and D sensors.

3.3.3 Machine Learning Technique

Support-vector-machine- and decision-tree-based-learning algorithms were used to analyze the preprocessed data. Both methods are well-suited to handling many-dimensioned data sets like the 43-dimensional data set in this research. Both are also more resistant to overfitting compared to other machine learning techniques. Overfitting is a modeling error that occurs in small datasets. Although overfitting classifies the correct result, when additional data is introduced, the classification model fails.

A multiclass classification mechanism was adopted since we collected data for seven activities. Although SVM is a binary classification approach, this binary nature can be extended for multiclass classification. Multiclass classification can be done by decomposing multiple classes into a combination of binary classes. Binary SVM can be conducted on the decomposed binary classes by doing the one-vs-one or one-vs-all [57, 58]. This research used the one-vs.-one (OVO) classification. For a K-way multiclass problem the OVO trains k(k-1)/2 binary classifiers (21 classifiers for seven classes based on seven activities). Each classifier contains binary classes from the original training set. Each classifier then did binary classification for a pair of classes.

3.3.4 Analysis Strategy

A preliminary review of the data determined that A/B/C/D sensor data and accelerometer data are equally important for activity detection. While some activities can be detected directly from sensor readings, activities like sitting that generate no sensor data require the use of accelerometer data to detect. Four combinations of activity were studied:

- two activities (walking and running): detection with accelerometer and sensor data
- two activities (walking and running): detection with only sensor readings
- seven activities (walking, running, sitting, standing, stair ascension and descension, elliptical use): detection with accelerometer and sensor data

- seven activities (walking, running, sitting, standing, stair ascension and descension, elliptical use): detection with only the accelerometer data.

3.4 Integrating the Machine Learning Model into the Application

To classify a patient's real-time physical activity and log activity to the system, an activity classification module was integrated into the PAT module. Figure 13 depicts the updated application's architecture and workflow. The machine learning model (MLM) listens for changes to the Firebase database. When the MLM is notified by Firebase that the database has received sensor data, the MLM activates the ML operation. The MLM collects sensor data at five-second intervals and deletes the data from the database as the data is read. The collected data is then preprocessed and passed to the ML algorithm. Once the ML classifies an activity based on the collected data, the ML saves its classification along with the classification time to the Firebase database. At the same time, the Android application's PAT module listens for classification event push events, displaying changes in activity classification data on the app's UI.

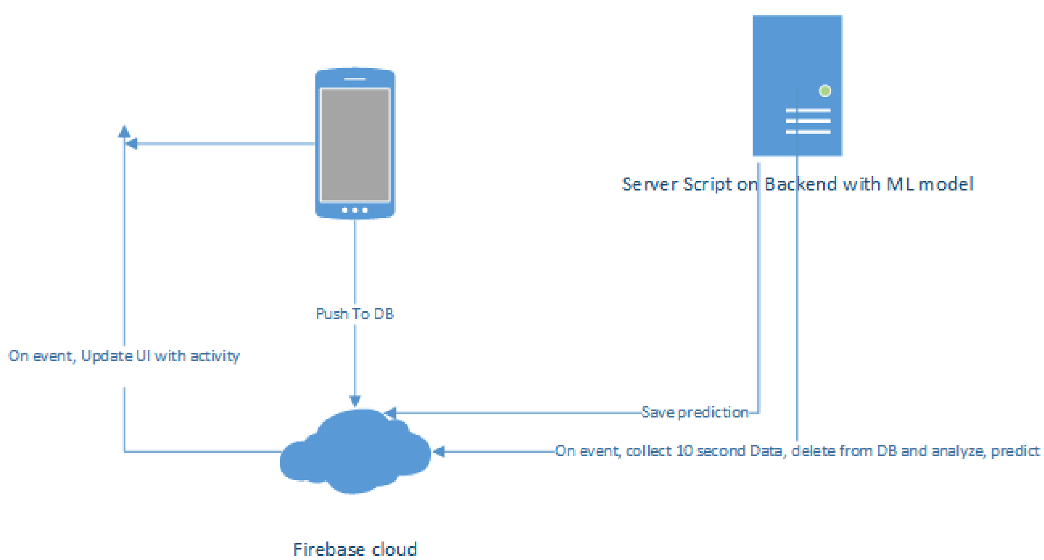

Figure 13 - Machine learning model integration to the application

4 RESULTS

BID data from the volunteers' activity trials were input into SVM and decision tree classification algorithms to generate models of volunteer activity. The resulting models were compared to identify which yielded more accurate characterizations of these activities. Finally, additional real-time data were used to assess the models' validity relative to data outside of the training sets.

Machine learning models were developed with the help of Python’s scikitlearn library, using test-to-train ratios of 10:90, 15:85, 20:80, 25:75 and 30:70 for the SVM and decision tree models alike. Different test-to-train ratios were used to determine which ratio yielded the most accurate results for both the SVM and decision tree models. Finally, two sets of models were produced. One was generated from all 43 features extracted from sensor and accelerometer data. The other, which was used to evaluate the accuracy of sensor-based activity detection, was generated from 16 sensor data features: i.e., number of total events; number of true events for sensors A, B, C, and D; mean, mode, and median impact; and mean and median time differences for sensors A, B, C, and D. The use of sensor-based data alone, without accelerometer data, was done to analyze the importance of accelerometer data for classifying activities that do not require foot movement (e.g., sitting). The resulting models were compared based on four sets of measures:

- The accuracy provided by the machine learning models (on a scale of 100)
- Classification accuracy based on error rate (in a scale of 100)
- Classification accuracy based on real-time data. For this analysis, we compared the real-time activity time compared to the actual activity done by a human subject.
- Data labeling was considered carefully; each shoe sole data was tagged with the shoe sole name to identify it.

4.1 Human Movement Data

To improve the quality of the machine learning models, sensor data from the trials was preprocessed to eliminate suspect data, as described in the "Methodology" chapter. For example,

- Data collection start and end times for each activity were monitored using timers, and the first and last few seconds of data were removed to eliminate possible outliers.

- Throughout a data collection session, the connectivity of both insole devices was monitored carefully. This addressed problems with firmware upgrades of the BID, which occasionally caused one of the devices to stop operating.
- NaN values from sensors were left intact throughout training. The extraction process ignored NaNs when computing means, modes, and medians, except for intervals that consisted entirely of NaN values, where NaN was returned instead.

The cleaned left shoe sole data was then plotted to obtain insights into BID patterns produced by different activity types. Figure 14, for example, shows a plot of BID walking data relative to time. All attributes described in Table 2 (below) were plotted against the events' timestamps. The red, blue, and green graphs show X-, Y-, and Z-axis data from the A, B, and C

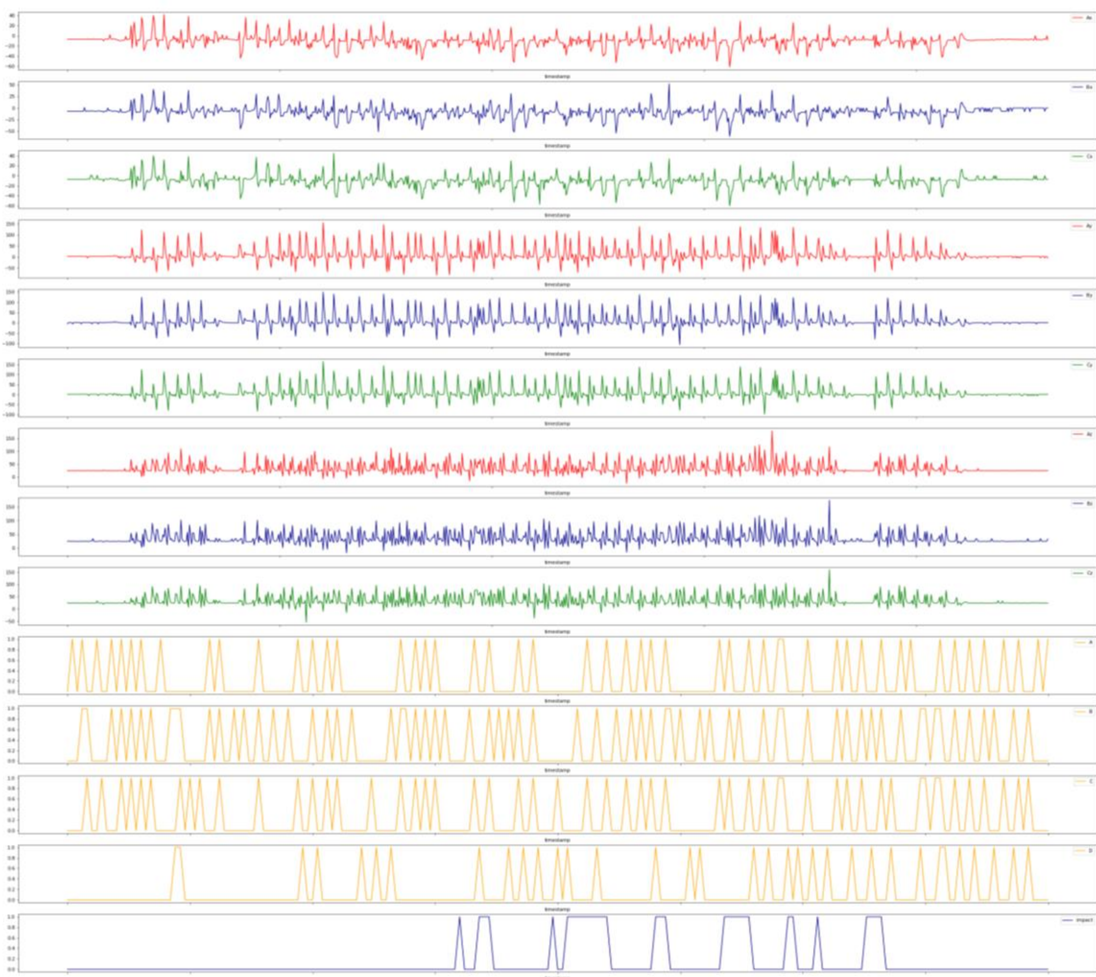

Figure 14 - Walking data visualization of one shoe sole

accelerometers, respectively. Orange plots show A, B, C, and C sensor data. The bottommost row in blue shows the ‘impact’ value.

The following observations were drawn from plots of four individuals' activities:

- Impact events and impact values for running are more pronounced than for walking. Accelerometer data varies for the two activities as well.
- Standing data included one sensor event. Sitting data included no such events.
- Stair ascent exercises all four sensors. Stair descent exercises three (A, B, and D).
- Elliptical data exercises two sensors and exhibits a rhythmic accelerometer pattern.

Additional plots generated from one of these individuals' trials are shown in Appendix A, Figures 24-29.

4.2 Human Activity Modeling

4.2.1 Training the Models

While training the model, different test train ratios (10:90, 15:85, 20:80, 25:75 and 30:70) were used to identify the ratio that generated the most accurate models. The best results— 90-100% accuracies—were obtained with test-to-train ratios of 15:85 and 20:80. Using these ratios, the machine learning models were built from two sequences of training exercises:

- two activities (walking and running) with accelerometer and sensor data
- seven activities (walking, running, sitting, standing, stair ascension and descension, elliptical use) with accelerometer and sensor data

Initially, the models provided accuracy results in a scale of 0-100. An accuracy of 96% means 96 results were classified correctly, while 4 classifications were incorrect. For each of the five test:train ratios, an average of 5 observations were performed on the classification models.

Figure 15 – Two activities and seven activities SVM result on four test train ratios

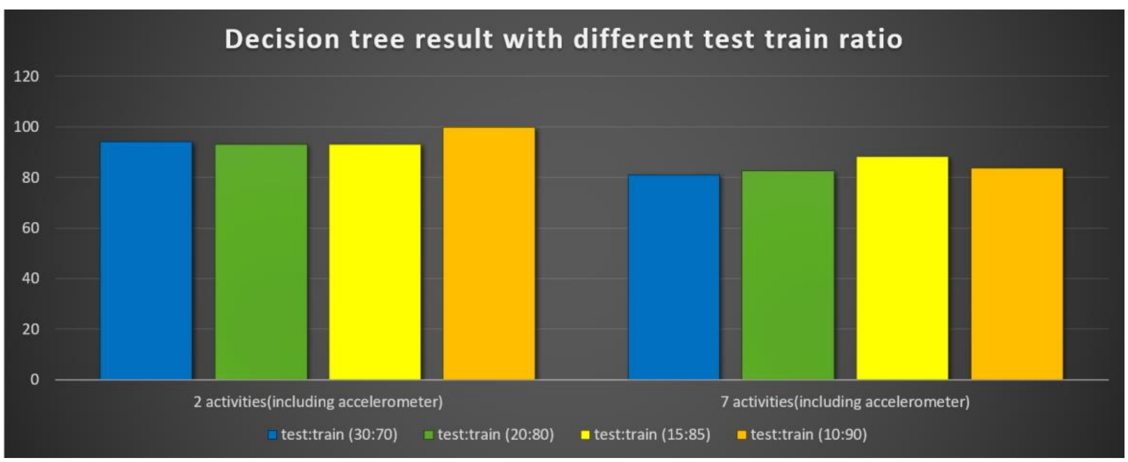

Figure 16 - Two activities and seven activities decision tree result on four test train ratios

Figure 15 and 16 depict an example of SVM and decision tree result on four test train ratios (30:70, 20:80, 5:85, 10:90).

4.2.2 SVM Models of Volunteer Activity

4.2.2.1 Walking Vs. Running

4.2.2.1.1 With Accelerometer and Sensor Data

A 96% classification accuracy was achieved when using accelerometer and sensor data to distinguish walking from running (Figure 17). From the 32-test samples of walking behavior, the

```
[[31  1]
 [ 1 26]]
             precision    recall  f1-score   support

          1       0.97      0.97      0.97        32
          2       0.96      0.96      0.96        27

avg / total       0.97      0.97      0.97        59

Accuracy of Linear Kernel SVM :0.966
```

Activity	Walking	Running
Walking	**31**	**1**
Running	**1**	**26**

Figure 17 - Walking and running classification with accelerometer and sensor readings

model classified 31 as walking and one as running. For running, only one sample was detected as walking, while 26 were detected as running.

4.2.2.1.2 With Sensor Data Alone

An 80% classification accuracy was achieved with data from the 15 sensor readings (Figure 18). The confusion matrix shows that from among 16 walking samples, eight were identified as actual walking and eight as running. For running, only one sample was detected as walking whereas all others were detected as actual running.

```
[[ 8  8]
 [ 1 28]]
             precision    recall  f1-score   support

          1       0.89      0.50      0.64        16
          2       0.78      0.97      0.86        29

avg / total       0.82      0.80      0.78        45

Accuracy of Linear Kernel SVM :0.800
```

Activity	Walking	Running
Walking	8	8
Running	1	28

Figure 18 - Walking and running classification with only the sensor reading

4.2.2.2 All Seven Activities

4.2.2.2.1 With Accelerometer and Sensor Data

In Figure 19, the seven activities are labeled as follows: 1: walking, 2: running, 3: sitting, 4: standing, 5: stair ascent, 6: stair descent, 7: elliptical use. The linear SVM produced an overall classification accuracy of 85%. The analysis from the confusion matrix was as follows:

Walking. Among 17 samples, 15 were classified accurately as walking, where two were classified as elliptical data. The precision-recall ratio is 0.75:0.88, denoting low precision and high recall: i.e., the model incurred no false positives but missed some of the walking samples.

Running. Among 28 samples, all were classified as running. The precision-recall ratio is 1:1, meaning that all of the model's classifications were accurate.

```
[[15  0  0  0  0  0  2]
 [ 0 28  0  0  0  0  0]
 [ 0  0 18  0  0  0  0]
 [ 0  0  3 10  1  0  1]
 [ 0  0  0  0  6  0  0]
 [ 1  0  0  0  3  3  0]
 [ 4  0  0  3  0  0 22]]

             precision    recall  f1-score   support

          1       0.75      0.88      0.81        17
          2       1.00      1.00      1.00        28
          3       0.86      1.00      0.92        18
          4       0.77      0.67      0.71        15
          5       0.60      1.00      0.75         6
          6       1.00      0.43      0.60         7
          7       0.88      0.76      0.81        29

avg / total       0.87      0.85      0.85       120

Accuracy of Linear Kernel SVM :0.850
```

Figure 19 - Result of linear SVM on seven activities with accelerometer and sensor data.

Sitting. Among 18 samples, all were classified as sitting. The precision-recall ratio is 0.86:1, denoting low precision and high recall: i.e., the model incurred no false negatives but incorrectly classified some other activities as sitting.

Standing. Among 15 samples, ten were identified correctly as standing, three were classified as sitting, and one each classified as stair ascension and elliptical. The precision-recall ratio is 0.71:0.67: i.e., it failed to classify many standing samples as standing.

Stair ascent. All six samples were classified as stair ascent. The precision-recall ratio is 0.60:1, meaning that other activities were also incorrectly detected as stair ascension.

Stair descent. Among seven samples, three were identified accurately as stair descent, three as stair ascent, and one as walking. The precision-recall ratio is 1:0.43: i.e., the model incurred no false positives but missed many samples, which were also stair descent.

Elliptical. Out of 29 samples, 22 were classified correctly, four were classified as walking and three were classified as standing. The precision-recall ratio is 0.80:0.76, which indicates that it misinterpreted a few samples as activities other than elliptical use.

```
[[ 6  0  0  0  1  0 12]
 [ 0  2  0  0  0  0 19]
 [ 0  0  0  0  0  0 12]
 [ 0  0  0  1  0  0 16]
 [ 0  0  0  0  1  0 17]
 [ 0  0  0  0  1  2  5]
 [ 0  0  0  0  0  0 25]]

             precision    recall  f1-score   support

          1       1.00      0.32      0.48        19
          2       1.00      0.10      0.17        21
          3       0.00      0.00      0.00        12
          4       1.00      0.06      0.11        17
          5       0.33      0.06      0.10        18
          6       1.00      0.25      0.40         8
          7       0.24      1.00      0.38        25

avg / total       0.64      0.31      0.24       120

Accuracy of Linear Kernel SVM :0.308
```

Figure 20 - Result of linear SVM on seven activities based on sensor data only

4.2.2.2.2 With Sensor Data Alone

Figure 20 shows the result of seven activity analysis without the accelerometer data.

Walking. Among 19 samples, six were identified accurately as walking, where 12 were classified as elliptical data. The precision-recall ratio is 1:0.32. High precision and low recall mean that although no other activities were classified as walking, a lot of walking activities were classified incorrectly.

Running. Among 21 samples, two were classified as running and rest were classified as elliptical data. The precision-recall ratio is 1:0.1, meaning that the probability of running to be classified as running was 10%, and the probability of other activity being classified as running is 0%.

Sitting. Among 12 samples, all were classified as elliptical data. The precision-recall ratio is 0:0; i.e., none of the classifications were correct. This validates the use of accelerometer data in the analysis: i.e., sitting does not generate any pressure sensor data. Eliminating accelerometer data from activity detection fails to identify sedentary movement.

No. of Activities	Used Accelerometer Data	Classification Accuracy
2	No	80% - 90%
2	Yes	100%
7	No	29%
7	Yes	80-90%

Figure 21 - Analysis result with decision tree method.

Standing. Among the 17 samples, one sample was classified correctly as standing. The rest were classified as elliptical data. The precision-recall ratio is 1:0.06; i.e., the model failed to classify a lot of standing samples as standing, while classifying no other activity as standing.

Stair ascent. One sample was identified correctly as stair ascent and the rest were elliptical. The precision-recall ratio is 0.33:0.06; i.e., the correct classified rate is too low, and other activities were also classified wrongly as stair ascent.

Stair descent. Two samples were classified accurately as stair descent, two as stair ascent and the other 17 as elliptical. Precision is 1, and recall is 0.25, meaning classification probability is just 25%.

Elliptical. Out of 25 samples, all were classified correctly, so the recall is 1. However, the precision is 0.24 because almost all other activities were wrongly classified as elliptical data.

4.2.3 Decision Tree Models of Volunteer Activity

Four sets of decision-tree models were generated from the test data. These models yielded the best results (100% accuracy) for walking and running detection with all 43 features. With 43 features, 7-activity detection accuracy ranged from 90 to 100%. With the 15 sensor data features alone, the 2-activity and 7-activity models yielded 80-90% and 29% accuracy rates, respectively. Figure 21 shows the analysis with the decision tree.

4.2.4 Implementation in BeticTrack

Based on this analysis of model quality, the Machine Learning Module (MLM) generated from all 43 features was integrated into BeticTrack. The model was then tested in real-time to determine its accuracy of classification. The measure for this classification was time. For example, a person conducted the seven activities during some period of time, and the real-time

```
walking      minutes: 3      seconds: 14
running      minutes: 0      seconds: 0
sitting      minutes: 4      seconds: 8
standing      minutes: 0      seconds: 0
stairup      minutes: 0      seconds: 0
stairdown      minutes: 0      seconds: 0
elliptical      minutes: 2      seconds: 17
```

Figure 22 - 4 minutes sitting, followed by 3 minutes standing, then 2.5 minutes walking

classification result was stored in the database. The classifications were collected and the cumulative time of each session of activity was calculated to check the accuracy of classifications.

Figure 22 shows the classification for two minutes of walking. In most cases (for 1 minute 45 seconds), the MLM classified walking as elliptical. It classified walking as stair ascension only for ten seconds.

A session was conducted with four minutes of sitting, followed by three minutes of walking and 2.5 minutes of standing. The MLM classified sitting accurately (4 minutes 3 seconds) (Figure 23). However, it did not classify any of the standing data as standing (it classified walking for 3 minutes and 14 seconds). Walking, moreover, was misclassified as elliptical data in most instances (for 2 minutes and 17 seconds). The model only classified 30 seconds of walking data originally as walking.

Figure 24 provides data from two sessions of 30-second stair descent. The stairs that were used for this test spanned three floors, with a landing between each flight of stairs. The classification identified the descent as well as the walking portions of this test as walking.

```
walking      minutes: 0      seconds: 0
running      minutes: 0      seconds: 0
sitting      minutes: 0      seconds: 0
standing      minutes: 0      seconds: 0
stairup      minutes: 0      seconds: 10
stairdown      minutes: 0      seconds: 0
elliptical      minutes: 1      seconds: 45
```

Figure 23 - Two minutes walking, real-time classification

```
walking      minutes: 0      seconds: 52
running      minutes: 0      seconds: 0
sitting      minutes: 0      seconds: 0
standing      minutes: 0      seconds: 0
stairup      minutes: 0      seconds: 0
stairdown      minutes: 0      seconds: 0
elliptical      minutes: 0      seconds: 11
```

Figure 24 - 30 seconds stair descent

To test running and elliptical real-time classification, we conducted an experiment that consisted of three minutes running, followed by 2.5 minutes of elliptical training (Figure 25). The model classified all running sessions as running. For the elliptical session, it classified 30 seconds data as elliptical and 1 minute 25 seconds data as walking. There was a sitting sample of ten seconds in between.

These data suggest that the model accurately classifies sitting and running but confuses elliptical and walking data. Also, the model failed to classify stair climbing, stair descent, and standing. In all these cases, the model was classifying walking for the most part. One possible explanation for the inaccuracy of the classification is that the interval on which the classifications were based was too long. The features generated were based on 10 seconds of data, whereas climbing stairs can take less than 10 seconds, depending on the number of steps.

For standing data, although the model classified the correct result, Figure 26 shows that the error rate of standing data classification is just 4.1% for standing. One possible reason for the erroneous classification could be the dataset's size. Using a model based on more data collected from different people over a longer period of time could improve the model's classifications.

```
walking      minutes: 1      seconds: 25
running      minutes: 3      seconds: 0
sitting      minutes: 0      seconds: 10
standing      minutes: 0      seconds: 0
stairup      minutes: 0      seconds: 0
stairdown      minutes: 0      seconds: 0
elliptical      minutes: 0      seconds: 33
```

Figure 25 - 3 minutes running followed by 2.5 minutes elliptical test

```
Activity     error rate
walking------------ 2.3255813953488373
running------------ 3.0303030303030303
sitting------------ 3.2
standing----------- 4.123711340206185
stair ascension---- 2.898550724637681
stair descension--- 2.272727272727273
elliptical--------- 3.592814371257485
```

Figure 26 - Error rate based on decision tree analysis.

The classification algorithm can classify sitting and running with almost 100% accuracy because of the nature of the shoe sole. The sitting data do not include any sensor data, compared to the other seven activities. On the other hand, the running data has the highest number of true events with frequent events while running. The sensor assumes a value as true if it senses the pressure. The pressure sensor has only two values (0 and 1), and for the case of running the sensor gets 1 if the sensor touches the ground with running pace. However, for other slow-to-medium pace activities like standing and walking or stair usage, the sensor only gets combinations of 0 values from the pressure sensor. The pressure sensor data has an impact on the failed classification. Another concern of the analysis is the overlap in classifying walking data and using elliptical data. Since the pattern of walking and using an elliptical instrument has similarities, removing elliptical data from the classification may produce better insight into classifying walking activity. This research focused on collecting five seconds of continuous data to generate features. Further analysis can be done to identify a time interval that can produce a better classification model.

5 CONCLUSIONS AND FUTURE WORK

BeticTrack's development was driven by the desire to ensure ease of use and ease of access to data for patients and physicians. There is no shortage of mobile applications for diabetes management in the marketplace. However, we incorporated physician feedback while designing the system, and we tried to capture all data that are important for diabetic management accurately. The overall goal of this research was to integrate all key parameters (blood glucose, carb intake, physical activity, timely reminders, and physician interaction) in one platform, to provide patients with a self-management portal, and provide clinicians with a remote patient monitoring platform. To these ends, this app included the use of wearable technology to collect patient information about their physical activity. The wearable IoT insole device collects physical activity data with no additional effort from its users. Currently, the app detects sitting, walking, and running with 85% accuracy, using SVM- and decision-tree-based learning algorithms. A future goal is to identify and track additional activities like cardio exercises. A second would be to improve the model's classifications for activities like stair climbing and standing. To make these classifications more accurate, this research should collect more data for a longer period from people of different ages and abilities. Also, future activity detection analysis should consider more than seven activities. K-fold cross-validation (K-fold CV) in the classifier sample could yield improved machine learning models. K-fold CV divides the complete dataset into k partitions to ensure that each partition is used as a test set over the learning iteration.

Currently, the app stores real-time activity along with its duration in the database. Future development includes creating a graphical representation of activity trends over time so that patients and physicians can track activity history. The detected activity pattern can be stored based on a timeline, which would allow doctors and patients to see these trends. Also, the balance of blood glucose could be calculated from the stored carbohydrate value, current BG record, and the exercise level. Future development may also include integrating this personal health record into an existing EHR system. For example, the medication reminder is accessible through a patient’s app. This could be made accessible to physicians via an EHR system, allowing them to update and check current medications prescribed to the patient.

In the next phase, our goal is to conduct a pilot study involving real patients and measure this application’s effectiveness in an actual setting. Future development also includes integrating all the modules into a HIPAA- (Health Insurance Portability and Accountability Act of 1996) compliant platform. According to the U.S. Department of Health and Human Services, standards for patient data management, any application dealing with sensitive human data must be secured [53]. The goal of HIPAA compliance is to protect an individual's health information while allowing the flow of health information necessary for the healthcare system. To ensure HIPAA compliance, this application will need to use technology such as Microsoft Azure's CosmosDB database. CosmosDB is a HIPAA-compliant server communicating over a secure socket and storing information on an encrypted database.

Although activity detection can accurately detect activities like running and sitting, there is a few seconds delay in display on the mobile device after an activity starts. Identifying the potential reason for the delay and revising the design to minimize the delay could be another goal for future research.

This research's long-term goal is to make the application available to patients for a pilot study and evaluate the application's effectiveness for type 2 diabetic patients. Future designs can be adapted, and apps created on the trial result and patient feedback. Depending on our findings from the deployment of the system for a pilot study involving real patients, this application has the potential to become a standard tool for diabetic care. If successful, this model can ideally extend to treat and manage other chronic diseases.

REFERENCES

[1] Who.int. (2019). Home. [online] Available at: https://www.who.int/ [Accessed 21 Apr. 2019].

[2] Polonsky, W. H., & Henry, R. R. (2016). Poor medication adherence in type 2 diabetes: recognizing the scope of the problem and its key contributors. Patient preference and adherence, 10, 1299.

[3] MHealth. (2018, May 30). Retrieved April 14, 2019, from https://www.himss.org/library/mhealth

[4] Business Insider. (n.d.). Retrieved April 22, 2019, from https://www.businessinsider.com/

[5] Tamony, P., Holt, R., & Barnard, K. (2015). The role of mobile applications in improving alcohol health literacy in young adults with type 1 diabetes: Help or hindrance? Journal of diabetes science and technology, 9(6), 1313-1320.

[6] Martínez-Pérez, B., De La Torre-Díez, I., & López-Coronado, M. (2013). Mobile health applications for the most prevalent conditions by the World Health Organization: review and analysis. Journal of medical Internet research, 15(6), e120.

[7] Klonoff, D. C. (2009). Using telemedicine to improve outcomes in diabetes—an emerging technology.

[8] Martinez, M., Park, S. B., Maison, I., Mody, V., Soh, L. S., & Parihar, H. S. (2017). iOS Appstore-Based Phone Apps for Diabetes Management: Potential for Use in Medication Adherence. JMIR diabetes, 2(2), e12.

[9] Brzan, P. P., Rotman, E., Pajnkihar, M., & Klanjsek, P. (2016). Mobile applications for control and self management of diabetes: a systematic review. Journal of medical systems, 40(9), 210.

[10] Veazie, S., Winchell, K., Gilbert, J., Paynter, R., Ivlev, I., Eden, K.B., Nussbaum, K., Weiskopf, N., Guise, J.M. and Helfand, M. (2018). Mobile Applications for Self-Management of Diabetes.

[11] Micro Sugar - Focus on Blood Sugar Health, Apple App Store 2017, ITunes. (n.d.). Retrieved from https://itunes.apple.com/us/genre/ios-medical/id6020, [Accessed 21 Apr. 2019].

[12] BlueStar Diabetes. Apple App Store. 2018. https://itunes.apple.com/us/app/bluestardiabetes/id700329056?mt=8.[Accessed 21 Apr. 2019].

[13] BlueStar Diabetes. Google Play. WellDoc, Inc.2017. https://play.google.com/store/apps/details?id=com.welldoc.platform.android&hl=en. [Accessed 21 Apr. 2019].

[14] mDiab Lite. Apple App Store. Infokom GmbH. 2014. https://itunes.apple.com/us/app/mdiablite/id604866236?mt=8. [Accessed 21 Apr. 2019].

[15] mDiab. Apple App Store. Infokom GmbH. 2017. https://itunes.apple.com/us/app/mdiab/id608181716?mt=8. [Accessed 21 Apr. 2019].

[16] mDiab Lite. Google Play. Infokom GmbH. 2013. https://play.google.com/store/apps/details?id=com.infokom.mdiab.lite&hl=en. Accessed [Accessed 21 Apr. 2019].

[17] mDiab. Google Play. Infokom GmbH. 2014. https://play.google.com/store/apps/details?id=co m.infokom.mdiab&hl=en. Accessed [Accessed 21 Apr. 2019].

[18] Dbees. Freshware. Amazon App Store. 2012. https://www.amazon.com/gp/product/?ie=UTF8&ASIN=B004XWHBIU&ref=mas_ty. [Accessed 21 Apr. 2019].

[19] Dbees. Freshware. Apple App Store. 2012. https://itunes.apple.com/us/app/dbees-comdiabetes undercontrol/id408492591?mt=8. [Accessed 21 Apr. 2019].

[20] https://play.google.com/store/apps. [Accessed 21 Apr. 2019].

[21] NexJ Health Coach +. Apple App Store. NexJ Health Inc. 2016. https://itunes.apple.com/ca/app/nexj-healthcoach/id1047247250?mt=8. [Accessed 21 Apr. 2019].

[22] IPhone, iPod & iPad Apps and Games. (n.d.). Retrieved April 22, 2019, from https://www.formidapps.com/

[23] (n.d.). Retrieved April 22, 2019, from http://myhealthapps.net/app/details/409/MoodKit

[24] Azumio Inc. (2008, October 28). Glucose Buddy Diabetes Tracker. Retrieved April 22, 2019, from https://itunes.apple.com/us/app/glucose-buddy-diabetes-tracker/id294754639

[25] Appadvice. (n.d.). Latest iOS News And Reviews. Retrieved April 22, 2019, from https://appadvice.com/appnn

[26] Keep track of your blood glucose the easy way! (n.d.). Retrieved April 22, 2019, from https://www.glucosetracker.net/

[27] (n.d.). Retrieved April 22, 2019, from http://sockii.com/products/

[28] Copyright @ OursSite.com - All Rights Reserved. (n.d.). Thelinklinks.com LINKLINKS LTD. Retrieved April 22, 2019, from http://thelinklinks.com.composesite.com/

[29] Easy grip on your health in diabetes type 1 with SugarPal. (n.d.). Retrieved April 22, 2019, from http://www.sugarpal.nl/Home.html

[30] Goyal, S., Nunn, C. A., Rotondi, M., Couperthwaite, A. B., Reiser, S., Simone, A., & Palmert, M. R. (2017, June 19). A Mobile App for the Self-Management of Type 1 Diabetes Among Adolescents: A Randomized Controlled Trial. Retrieved from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5495971/

[31] Goyal, S., Morita, P., Lewis, G. F., Yu, C., Seto, E., & Cafazzo, J. A. (2016). The systematic design of a behavioral mobile health application for the self-management of type 2 diabetes. Canadian journal of diabetes, 40(1), 95-104.

[32] Triberti, S., Bigi, S., Rossi, M.G., Caretto, A., Laurenzi, A., Dozio, N., Scavini, M., Pergolizzi, E., Ozzello, A., Serino, S. and Riva, G. (2018, January). The ActiveAgeing Mobile App for Diabetes Self-management: First Adherence Data and Analysis of Patients' in-App Notes. In International Symposium on Pervasive Computing Paradigms for Mental Health (pp. 129-138). Springer, Cham.

[33] Modave, F., Bian, J., Rosenberg, E., Mendoza, T., Liang, Z., Bhosale, R., Maeztu, C., Rodriguez, C. and Cardel, M.I. (2016). DiaFit: the development of a smart app for patients with type 2 diabetes and obesity. JMIR diabetes, 1(2), e5.

[34] Jeandidier, N., Chaillous, L., Franc, S., Benhamou, P. Y., Schaepelynck, P., Hanaire, H., & Reznik, Y. (2018). DIABEO app software and telemedicine versus usual follow-up in

the treatment of diabetic patients: protocol for the TELESAGE randomized controlled trial. JMIR research protocols, 7(4), e66.

[35] Osborn, C. Y., van Ginkel, J. R., Rodbard, D., Heyman, M., Marrero, D. G., Huddleston, B., & Dachis, J. (2017). One Drop| Mobile: An Evaluation of Hemoglobin A1c Improvement Linked to App Engagement. JMIR diabetes, 2(2), e21.

[36] Waki, K., Fujita, H., Uchimura, Y., Omae, K., Aramaki, E., Kato, S., Lee, H., Kobayashi, H., Kadowaki, T. and Ohe, K. (2014). DialBetics: a novel smartphone-based self-management support system for type 2 diabetes patients. Journal of diabetes science and technology, 8(2), 209-215.

[37] Pustozerov, E., & Popova, P. (2018, May). Mobile-based decision support system for gestational diabetes mellitus. In 2018 Ural Symposium on Biomedical Engineering, Radioelectronics and Information Technology (USBEREIT)(pp. 45-48). IEEE.

[38] Ryan, E.A., Holland, J., Stroulia, E., Bazelli, B., Babwik, S.A., Li, H., Senior, P. and Greiner, R (2017). Improved A1C levels in type 1 diabetes with smartphone app use. Canadian journal of diabetes, 41(1), 33-40.

[39] Bonn, S. E., Alexandrou, C., Steiner, K. H., Wiklander, K., Östenson, C. G., Löf, M., & Lagerros, Y. T. (2018). App-technology to increase physical activity among patients with diabetes type 2-the DiaCert-study, a randomized controlled trial. BMC public health, 18(1), 119.

[40] De Pinho André, R., Diniz, P. H. F., & Fuks, H. (2017, September). Bottom-up investigation: Human activity recognition based on feet movement and posture information. In Proceedings of the 4th international Workshop on Sensor-based Activity Recognition and Interaction (p. 10). ACM.

[41] Caspersen, C. J., Powell, K. E., & Christenson, G. M. (1985). Physical activity, exercise, and physical fitness: definitions and distinctions for health-related research. Public health reports, 100(2), 126.

[42] Sayer, A. A., Dennison, E. M., Syddall, H. E., Gilbody, H. J., Phillips, D. I., & Cooper, C. (2005). Type 2 diabetes, muscle strength, and impaired physical function: the tip of the iceberg?. Diabetes care, 28(10), 2541-2542.

[43] Cetinus, E., Buyukbese, M. A., Uzel, M., Ekerbicer, H., & Karaoguz, A. (2005). Hand grip strength in patients with type 2 diabetes mellitus. Diabetes research and clinical practice, 70(3), 278-286.

[44] Wen, C.P., Wai, J.P.M., Tsai, M.K., Yang, Y.C., Cheng, T.Y.D., Lee, M.C., Chan, H.T., Tsao, C.K., Tsai, S.P. and Wu, X (2011). Minimum amount of physical activity for reduced mortality and extended life expectancy: a prospective cohort study. The Lancet, 378(9798), 1244-1253.

[45] Hamasaki H. (2016). Daily physical activity and type 2 diabetes: A review. World journal of diabetes, 7(12), 243–251. doi:10.4239/wjd.v7.i12.243

[46] Gillen, J. B., Little, J. P., Punthakee, Z., Tarnopolsky, M. A., Riddell, M. C., & Gibala, M. J. (2012). Acute high-intensity interval exercise reduces the postprandial glucose response and prevalence of hyperglycaemia in patients with type 2 diabetes. Diabetes, Obesity and Metabolism, 14(6), 575-577.

[47] Biswas, A., Oh, P. I., Faulkner, G. E., Bajaj, R. R., Silver, M. A., Mitchell, M. S., & Alter, D. A. (2015). Sedentary time and its association with risk for disease incidence, mortality, and hospitalization in adults: a systematic review and meta-analysis. Annals of internal medicine, 162(2), 123-132.

[48] American Diabetes Association. (2014). Standards of medical care in diabetes—2014. Diabetes care, 37(Supplement 1), S14-S80.

[49] Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. R news, 2(3), 18-22.

[50] Krogh, A., Larsson, B., Von Heijne, G., & Sonnhammer, E. L. (2001). Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. Journal of molecular biology, 305(3), 567-580.

[51] Hsu, C. W., & Lin, C. J. (2002). A comparison of methods for multiclass support vector machines. IEEE transactions on Neural Networks, 13(2), 415-425.

[52] Magerman, D. M. (1995, June). Statistical decision-tree models for parsing. In Proceedings of the 33rd annual meeting on Association for Computational Linguistics (pp. 276-283). Association for Computational Linguistics.

[53] Chinimilli, P. T., Redkar, S., & Zhang, W. (2017, May). Human activity recognition using inertial measurement units and smart shoes. In 2017 American Control Conference (ACC) (pp. 1462-1467). IEEE.

[54] Hegde, N., & Sazonov, E. S. (2015, November). SmartStep 2.0-A completely wireless, versatile insole monitoring system. In 2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) (pp. 746-749). IEEE.

[55] Hegde, N., Melanson, E., & Sazonov, E. (2016, August). Development of a real time activity monitoring Android application utilizing SmartStep. In 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) (pp. 1886-1889). IEEE.

[56] Nguyen, N. D., Bui, D. T., Truong, P. H., & Jeong, G. M. (2018). Classification of five ambulatory activities regarding stair and incline walking using smart shoes. IEEE Sensors Journal, 18(13), 5422-5428.

[57] Huang, C., Davis, L. S., & Townshend, J. R. G. (2002). An assessment of support vector machines for land cover classification. International Journal of remote sensing, 23(4), 725-749.

[58] Halldorsson, G. H. (2003). Support vector machines in multisource classification. In Geoscience and Remote Sensing Symposium, 2003. IGARSS'03 s. 2054-2056.

APPENDIX: GRAPHS OF PREPROCESSED SENSOR DATA

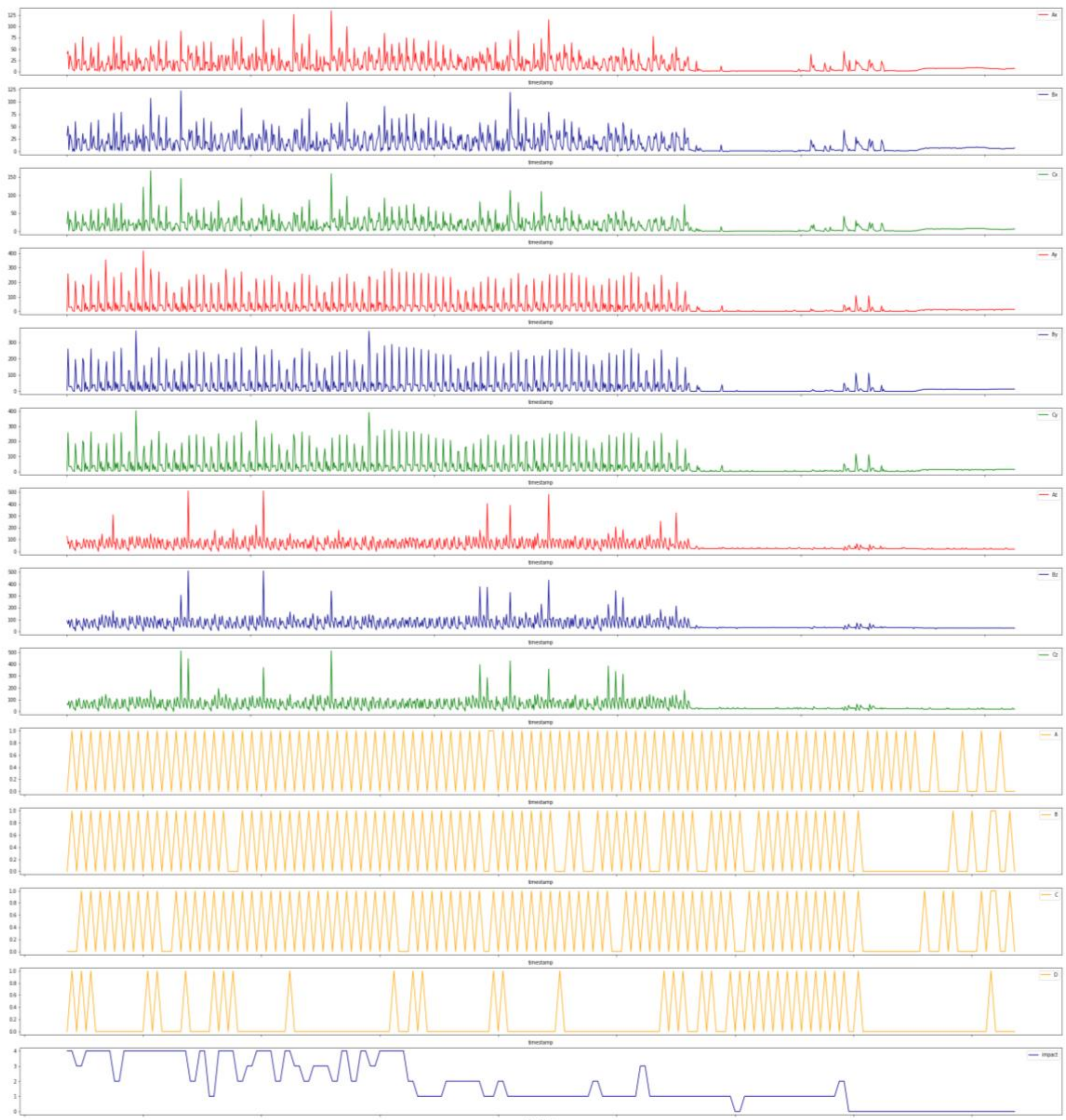

Figure 27 - Running data visualization of one shoe sole

Figure 28 - Sitting data visualization of one shoe sole

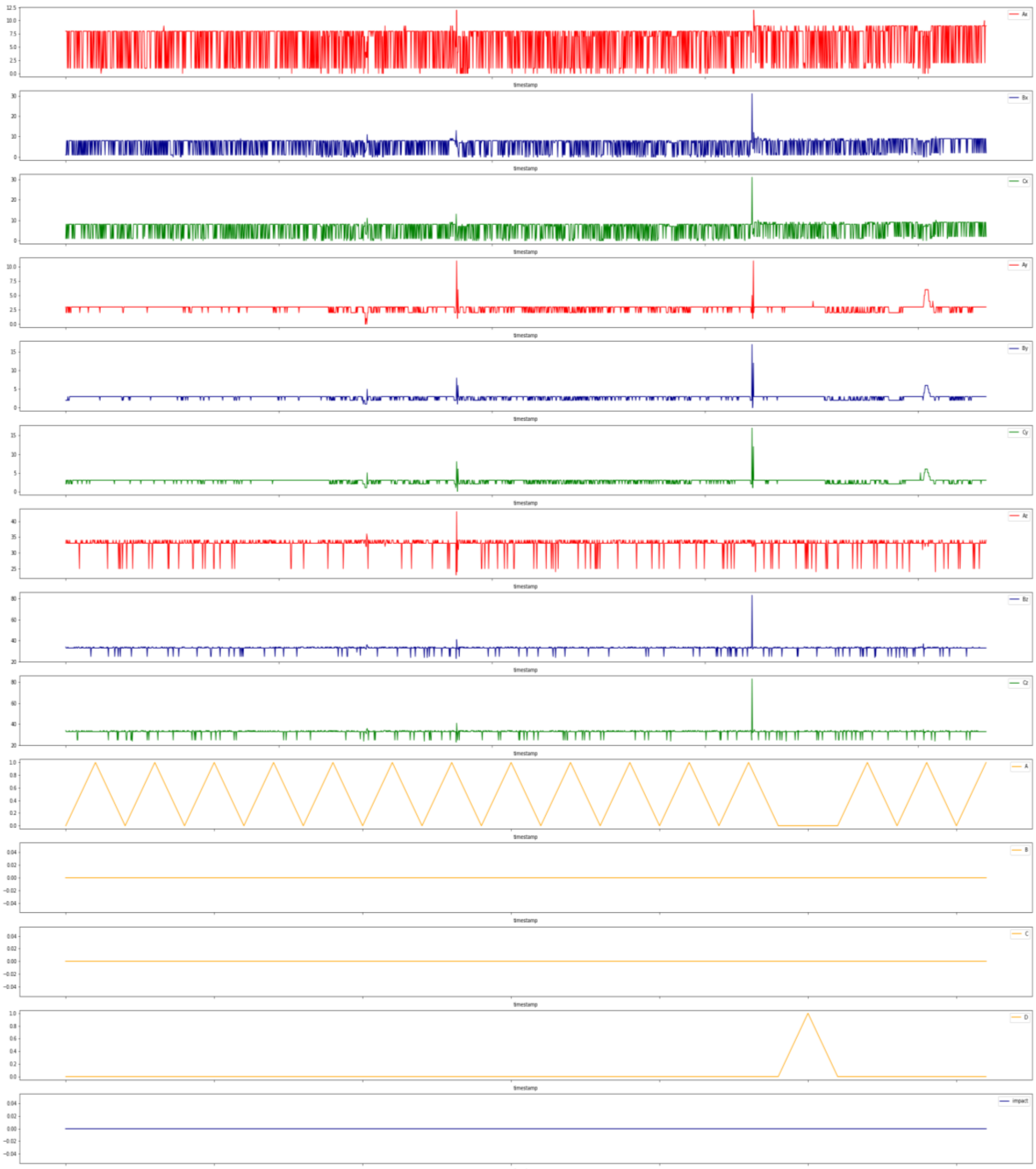

Figure 29 - Standing data visualization of one shoe sole

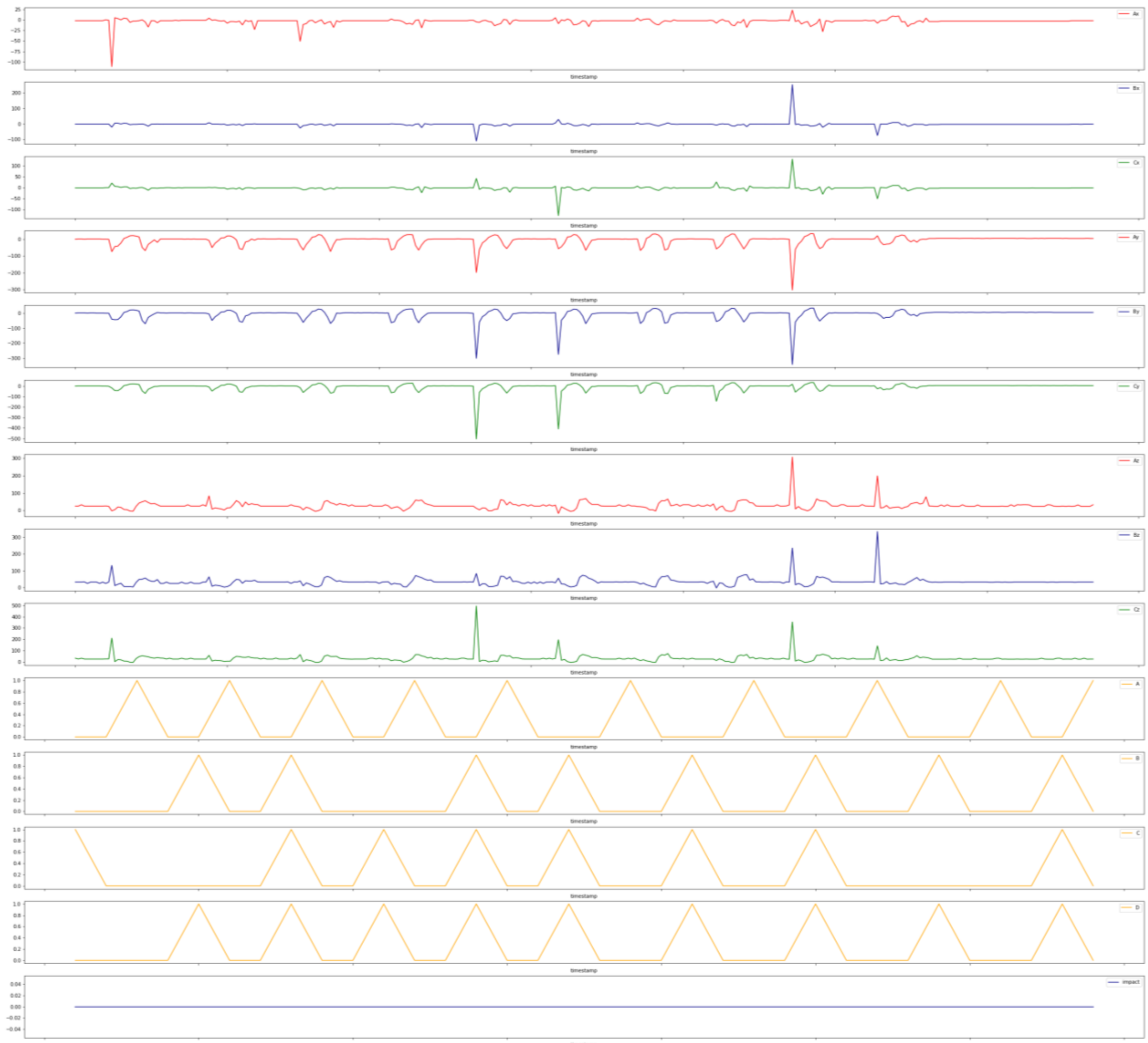

Figure 30 - Stair ascension data visualization of one shoe sole

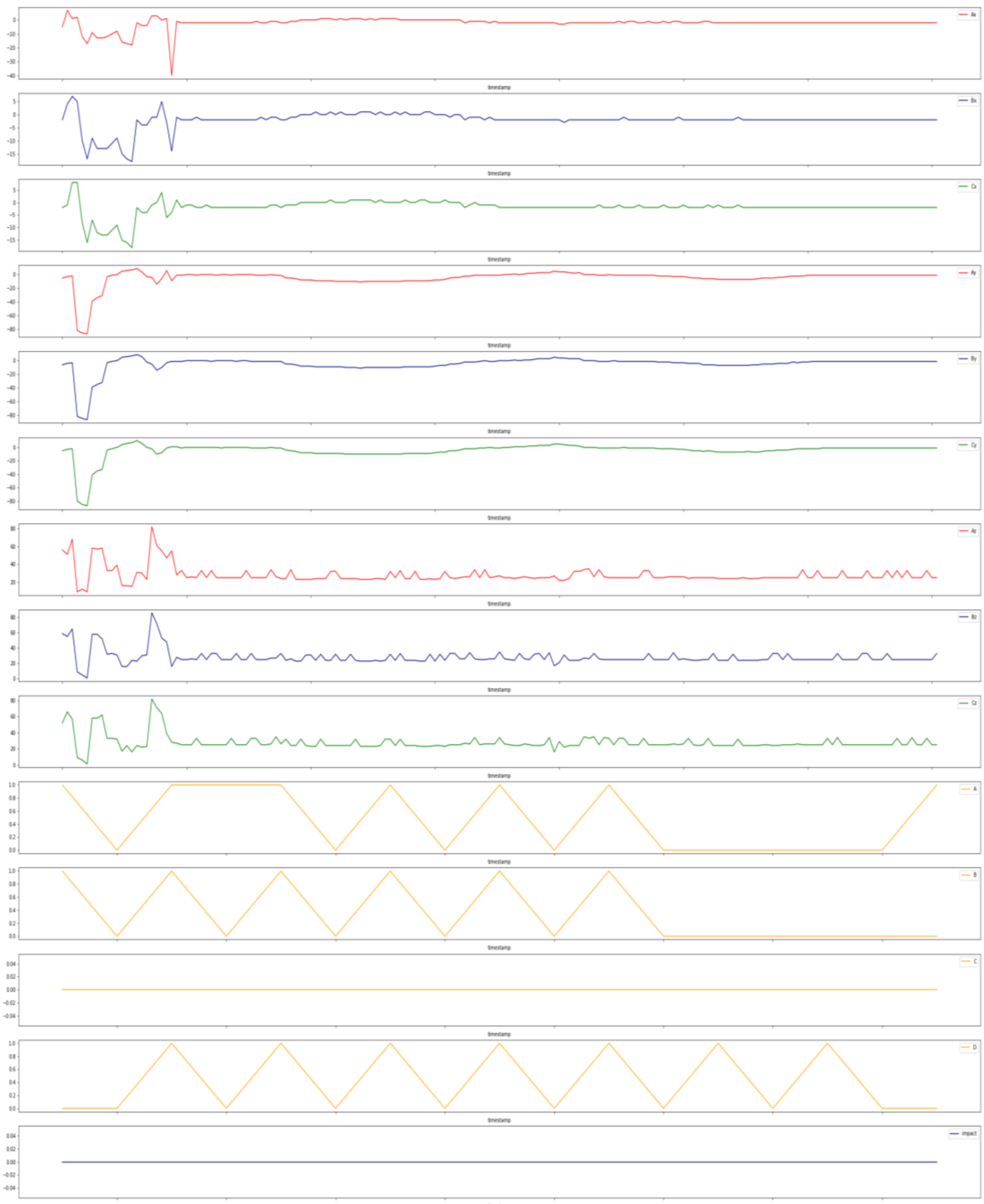

Figure 31 - Stair descent data visualization of one shoe sole

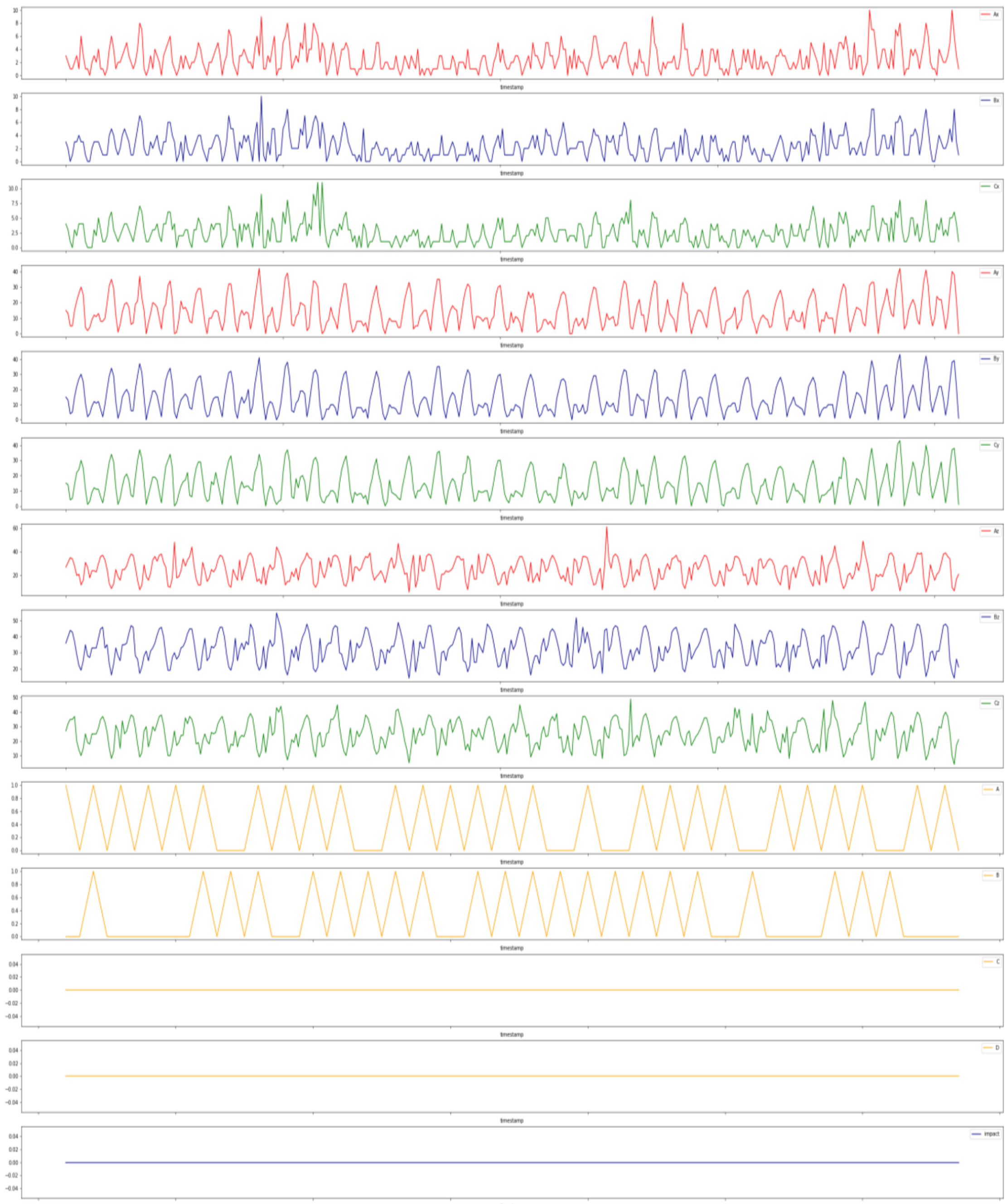

Figure 32 - Elliptical data visualization of one shoe sole

VITA

NUSRAT JAHAN CHOWDHURY

Education:	MS, Computer and Information Sciences, Department of Computing, East Tennessee State University, TN, USA, 2019
	B.Sc in Computer Science and Telecommunication Engineering, Noakhali Science and Technology University, Noakhali, Bangladesh, 2013
Experience:	Graduate Assistant, Department of Computing, East Tennessee State, University Johnson City, TN, 2017-2019.
	Lecturer, Cox's Bazar International University, Cox's Bazar, Bangladesh, 2014-2015.
	Intern (ROM), Grameen phone IT, Chittagong, Bangladesh, summer 2013
Publication:	Real-Time Alerting of Hazardous Driving Behaviour Using IoT and Connected Vehicles, Nusrat Chowdhury, Mohammad Hoque, Ferdaus Kawsar, Mohsen Kamrani, and Asad Khattak, Accepted for presentation at The 2018 ITS America Annual Meeting Detroit in Detroit, June 4 - 7, 2018.
	Intelligent Traffic Signaling System Based on Vehicle Demand for Excessively Populated Developing Countries, Iftekhar Mahmud Towhid, Nishu Nath, and Nusrat Jahan Chowdhury, International Journal of Scientific and Engineering Research, 6.4 (April - 2015) (1883 - 1890)