A MOBILE HEALTH PLATFORM FOR AUTOMATED DIET MONITORING USING CONTINUOUS GLUCOSE MONITORS AND CONTEXT-AWARE MACHINE LEARNING

A Thesis

by

SORUSH OMIDVAR

Submitted to the Graduate and Professional School of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Bobak Jack Mortazavi
Committee Members,	Farzan Sasangohar
	Theodora Chaspari
Head of Department,	Scott Schaefer

May 2022

Major Subject: Computer Science

ABSTRACT

Automated diet monitoring, an important tool in preventing healthy individuals and those with pre-diabetes from developing Type 2 Diabetes, requires automatic eating detection and estimation of the macronutrient contents of ingested food. While signals from continuous glucose monitors may track the post-prandial glucose response (glucose response after eating) and use this for estimation of nutritional information, the proper identification and segmentation of these periods of eating require additional sensing modalities and contextual information. In this work, we developed a framework for machine learning modeling to detect eating periods, properly segment post-prandial glucose responses, and estimate nutritional content from these segments in real-world environments using data captured from a continuous glucose monitor and augmented with contextual data from smartwatch wearable sensors. Using a custom-developed platform, we conduct a human subject study where participants were free to eat what they wished, when they wished, logging data and wearing a set of sensors. To aid future, just-in-time diet monitoring applications, we found that contextual data improved eating moment detection and thus enables real-time macronutrient estimation.

DEDICATION

I want to dedicate this work to my wife Saba, mother Zahra, father Abolfazl, brother Sepehr and friend Nima who were always in my corner no matter of what happened. This work was utterly impossible without their financial and emotional support.

ACKNOWLEDGMENTS

I want to express my genuine appreciation for the help from Amin Hamiditabar for helping in the development of the mobile application, Sicong Huang for assisting in the project, Arash Pakbin for inspiring me in the theoretical framework and all other friends and lab mates for their support. Moreover, my gratitude goes to Dr. Bobak Moratzavi, who believed in me and provided me the opportunity to serve in the STMI lab. Additionally, I would like to thank the Texas A&M University Graduate and Professional School for allowing me to construct this LaTeX thesis template.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supervised by Professor Mortazavi, and with the help of committee consisting of Professors Sasangohar and Chaspari of the Department of Computer Science & Engineering. All work for the thesis was completed independently by the student.

Funding Sources

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the NSF Engineering Research Center for Precise Advanced Technologies and Health Systems for Underserved Populations (PATHS-UP; Award #1648451) and NSF-IIS award #2014475.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 Outline

This chapter introduces the topic and declares the study's goals; the second chapter defines the context and reviews the literature for how context is defined and used, its definition, and application specifically in relation to biosensing and health monitoring. The third chapter introduces our methodology and establishes a framework for data collection, providing details and applications used for this study in the experiments conducted. The fourth chapter is dedicated to experiments and results, and the final chapter concludes this study and explores potential future directions.

1.2 Objectives

There are 1.4 (0.5) billion people in the world with a body-mass index of > 25 (> 30) suffering from being overweight (obese). Obesity can increase the chance of diabetes mellitus, a medical condition where the body cannot regulate blood glucose properly, which is a leading cause of death (based on the World Health Organization report). Additionally, it serves as a primary co-morbidity to a number of other serious medical complications, including indirect deaths in COVID-19 pandemic Caballero et al. [2020]. Therefore, significant research has investigated the prescription of diet control and exercise to prevent healthy individuals and those with pre-diabetes from developing type 2 diabetes, which depends on correctly logging meal times and macronutrient content (such as carbohydrate, fat, etc. and protein). Therefore, in this study, I aim to perform an accurate eating detection and subsequently estimate the macronutrient of such detected meals.

1.2.1 Eating Detection

This diet and exercise monitoring involves monitoring the frequency, quantity, and quality (macronutrient content) of the diet, as well as periods of intense exercise and of inactivity Solis et al. [2019], Akbari et al. [2020]. However, the consistent logging and monitoring of this information is an arduous task and is often accompanied by user error in the process of data insertion (wrong values, types, or time) and can cause a lot of issues for machine learning models Cordeiro et al. [2015], Chung et al., Sajjadi et al. [2021]. We can break the task into detecting eating moments and estimating meal nutrition. However, some methods can help both of them. Lots of efforts were made to detect the eating periods automatically, such as placing a sensor to detect chewing and swallowing sound near the throat Gao et al. [2016], putting cameras in glass lenses or frames to detect the eating and the content and size of a meal Gemming et al. [2015], monitoring the tensions and contractions in temporalis muscles to monitor muscle Zhang and Amft [2018]. All of the methods above are invasive to the user's privacy; on the other hand, with the advent of smartwatch, hand movements can be monitored and tracked to detect eating gestures which relieve the user from the burden of wearing extra and unorthodox sensors. Such a detection was previously done using medical-grade wearable but the model suffers high number of false positives Dong et al. [2014], Sharma et al. [2020]. Similar efforts were made using continuous glucose monitor (CGM) sensor mostly for diabetic participants Staal et al. [2019]; however, such a detection is accompanied by a lag (required for food metabolism). Therefore, it will be of interest to make an effort to provide some contextual information for CGM models (such as hand motion) to improve the model performance.

1.2.2 Macronutrient Estimation

Macronutrient estimation is additionally challenging Cordeiro et al. [2015], Chung et al., even when correct periods of meal detection occur. To help the user reduce errors and automate the process, we can utilize extra sensors (such as CGM) to discern the fat, protein, and carbohydrate content of a meal as their post-prandial glucose response (PPGR) has a different peak time, amplitude, and area under the curve Zeevi et al. [2015]. Several studies have attempted to estimate the nutrient content of a meal based on CGM readings for non-diabetic participants in a laboratory controlled environment with meals of known macronutrient quantities and fixed periods of fasting and inactivity Sajjadi et al. [2021], Paromita et al. [2021], Huo et al. [2019]. Even with such limitations results suffer from low performance. The reason behind such a poor performance can be misinformation (lack of data) or disinformation (biased data), which can be alleviated using contextual information provided by wearables and nearables.

1.3 Context Awareness

Unique and novel wearable sensors enable a breadth of personal health monitoring applications. Physiological and behavioral data collected by wearables, especially in uncontrolled environments, is affected by noise, motion artifact, and external stimuli such as daily activities. Therefore, understanding the contextual information surrounding wearable data is paramount. Context data provides additional information on who, what, where, when, why, and how of remote data collection. However, context is not a universal definition for analytic models on sensing systems. By evaluating context, sensing systems can capture health and behavior from very discrete actions to long-term longitudinal health monitoring. The ADL recognition, along with other behavioral information, can further be used for detecting high-level lifestyle information such as diet, which itself could provide contextual information when looking into physiological data for health monitoring. Accordingly, context is the key to unlocking the potential of sensing and bio-sensing for health and wellness applications as dicussed in the next chapter.

1.4 Conclusion

Finally, by combining the methodology from eating detection studies, macronutrient estimation methods, and introducing contextual information, I build a multi-modal model to identify periods of eating in the wild correctly. Additionally, I set the stage ready for the next level, which is the macronutrient estimation of the meal based on PPGR and contextual information of such segmented meal times.

2. DEFINING CONTEXT AWARENESS FOR REMOTE HEALTH MONITORING

2.1 Introduction

2.2 Context for Lifestyle Monitoring: Complex Behavior Modeling

Contextual information can aid the capture of data that effectively monitors health and wellness Hurley et al. [2020]. This context, the who, what, where, when, why, and how of remote data collection allow models to understand and interpret changes in biomarker data. This section highlights the importance of utilizing contextual information for lifestyle monitoring. Here, actions are not specified to windows of time or locations but may be more complex in nature, duration, and overlap with others. We then demonstrate this in the context of health and behavior, namely, tracking activity and diet monitoring, which itself then provides invaluable context for health and wellness monitoring applications, such as diabetes management, hypertension control, and other chronic conditions where lifestyle interventions aid recovery and outcome. Whereas the prior section provided methods by which automated exercise monitoring is possible Das et al. [2021], Mortazavi et al. [2014], we demonstrate this context definition, identification, and use through diet monitoring to complete a holistic lifestyle monitoring review.

2.2.1 Need for Context for Lifestyle Monitoring

Blood glucose and diet monitoring management (through meal macronutrient estimation) are essential for tracking lifestyle and behavior choices of individuals, particularly those with or at risk of developing type 2 diabetes. Continuous glucose monitoring (CGM) are a great example of sensors that track biometric information which both enable health and wellness applications but also need significant context to frame the data capture. A study by Zeevi et al. [2015] showed that by monitoring the postprandial glucose response (PPGR), individualized responses to specific meals can be identified. In particular, it identifies what causes the glucose responses to become elevated, stay elevated and how to avoid hypoglycemic events through meal choices. Exercise can also impact glucose responses, where exercise can cause reductions in glucose excursions Cockcroft et al.

[2019], MC et al. [2019].This information is extremely important in helping participants make the right diet and exercise choices.

Deciphering the personalized response from CGM sensors to automate and interpret diet, however, is an arduous task. Using 8 hours of glucose data and constraining participants to single meals and no exercise, it becomes possible to recover diet information automatically from sensing devices Das et al. [2021], Huo et al. [2019], Sajjadi et al. [2021], Yang et al. [2021]. However, removing those restrictions quickly increases the difficulty of this task. Moreover, the same person might not respond to the same meal equally as his/her body response highly depends on environmental and personal factors such as psychological mood, energy expenditure (sports), and medications Oviedo et al. [2017]. With the advent of sophisticated machine learning models, there is an opportunity to augment the blood glucose time series with contextual information to enhance prediction accuracy. Therefore, in the last two decades, a growing number of studies leveraged contextual data along with blood glucose readings for diet monitoring Akbari and Chunara [2019], Oviedo et al. [2017], Rabby et al. [2021]. Contextualizing and personalizing glucose response from additional sources of data, such as multi-omics data, can aid longitudinal monitoring of choices and their health effects Zhou et al. [2019].

Indeed, one of the most important contextual data for smart diet monitoring is automated detection of eating moments. Not only could this context help with more accurate computations, but it can also facilitate nutrition logging and data annotation for the end users by sending them timely reminders. Placing acoustic sensors for detecting chewing and swallowing sound near the throat region Amft et al. [2005], Gao et al. [2016], Pasler and Fischer [2014], cameras or smart glasses to visually detecting the food Gemming et al. [2015], Hodges et al. [2006], Sun et al. [2014], electromyograph on temporalis muscles to monitor muscle contractions Huang et al. [2017], Zhang and Amft [2016, 2018], and wearables on the hand to detect the hand movement Dong et al. [2012], Luktuke and Hoover [2020] have all been explored. This information provides a foundation for context information in lifestyle monitoring, identifying when certain choices are being made. Using wearable to detect eating moments is the most orthodox method in the literature as the eating gesture is usually patterned and periodic. Wearables can be categorized into the medical grade (such as Empatica, Actigraph, and Shimmer), which are typically available to researchers with more battery life, higher price, higher functionality, or commercial ones (such as Apple Watch, Samsung, and Fitbit). These wearables can record the participants' hand movements and gestures 1) using an accelerometer and gyroscope or 2) measuring and calculating the current location of the hand with respect to the body or hand's attitude (such as InertiaCube3). The collected time series can be fed to machine learning models or simpler algorithms (such as naive Bayes classifier) for feature extraction.

2.2.2 Context Pattern Extraction for Lifestyle Monitoring

An integral part of diet management comes from estimating the macronutrient content of a meal for health information. Currently, this work remains burdensome for the user, requiring manual logging which can be facilitated using automatic detection methods Mortazavi and Gutierrez-Osuna [2021]. Some techniques exist to aid in providing context to meal logging, defining context for diet monitoring applications.

Food crushing and swallowing can have a distinct sound that can be used toward detecting a meal. To do so, a microphone should be placed near the mouth to listen to the process. The eating process usually has three distinct stages of tearing, crushing, and swallowing. (Amft et al., 2005) Amft et al. Amft et al. [2005] placed a microphone in the ear channel (which is usually used for hearing aids) to listen to these three stages and recognized the chewing part among four test subjects for the period of 3827 seconds (in total) eating a predefined food (potato chips, apple, lettuce, pasta, and rice). Using spectral analysis (Fourier transform) and applying the cut-off of 5 dB, they recognized the chewing and the meal type by 99% and 80-100%. To make this methodology more accessible Gao et al. [2016] proposed to use Bluetooth headset, which provided 95% and 77% to 94% accuracy in the laboratory (where there is a low amount of noise) and in a realistic setup outside the lab.

The visual feed is usually used as a reminder to report and log the meal rather than an automatic way of detecting the eating moment or the meal content. Although the camera sensors can be used toward automatic detection of eating, due to privacy issues, battery life, low performance, and being inconvenient, they are used as a way to refresh the participant memories. Hodges et al. Hodges et al. [2006] developed the SenseCam platform initially to help patients with amnesia and Alzheimer's disease, but it can be used toward smart dieting as well. Gemming et al. Gemming et al. [2015] analyzed social and environmental parameters using the SenseCam worn by 40 participants for four days. Sun et al. Sun et al. [2014] utilized a camera mounted at the chest to detect food and estimate the macro-nutrient contents using a nutrition database. This approach gives satisfactory results in the macro-nutrient content detection in 85% of the time (for 100 tested meals) with an error of less than 30% if the images are not occluded.

Similar to auditorial detection of the chewing sound, we can consider the muscle tension as contextual data for eating. The jaw and temporalis muscles contract and relax periodically during the chewing process or tearing the food. The chewing process can be recognized by planting myelography and vibration sensors near the temporalis muscle Zhang and Amft [2016]. By planting a myelography sensor into an eyeglass, Zhang et al. Zhang and Amft [2018] were able to detect the eating moments and classify the food hardness with the accuracy of 95% and 94%. In a similar approach, Huang et al. Huang et al. [2017] reached the accuracy of 96% detection of a chewing cycle and 91% of the food type.

Dong et al. Dong et al. [2009] tried to estimate the number of bites using an InertiaCube3 sensor capable of recording the wrist motions (yaw, pitch, and roll orientations) with the frequency of 60 hertz. Using such a sensor and methodology on ten participants eating a meal (using hands or utensils), they reached the recall of 91%. In a follow-up, Dong et al. Dong et al. [2014] successfully detected the eating moments outside the laboratory environment (not controlled) with an accuracy of 80%. In this study, the accelerometer and gyroscope time-series were collected from an iPhone placed on participants' hands for about 12 hours with the frequency of 15 Hz for a batch of 30 participants. In a similar trial, Luktuke et al. Luktuke and Hoover [2020] classified the eating gestures among 276 participants at the Clemson university cafeteria using IMU data with the frequency of 15 Hz with an accuracy of 75 % to 85% (for different motions such as drinking and biting). It should be noted that although the earlier studies have higher accuracy, the eating tonalities and food had lower diversity than the newer ones; therefore, it was a simpler task with higher accuracy.

2.2.3 Using Context Framework for Lifestyle Monitoring

Considering the importance of contextual information, several studies and trials tried to include it in their datasets. For example, the Ohio T1DM dataset, which is a dataset compiled at the University of Ohio, contains not only the continuous glucose monitoring time series but also the self-reported time of the meal, exercise, sleep, work, stress, and illness as well as sensor data such as accelerometer, gyroscope, body temperature, heart rate, galvanic skin response and step count Marling and Bunescu [2020]. Also, the D1NAMO dataset, which consists of twenty-nine participants (twenty normal and nine with type-1 diabetes), for about five days contains glucose time series as well as ECG, breathing, accelerometer, and meal pictures Dubosson et al. [2018]. So, one might think about augmenting the CGM readings with contextual information to help the prediction in smart dieting. Although to our knowledge, there is no study to do such an integration, a few recent works benefited from context to enhance predicting blood glucose which are discussed below.

Rabby et al. Rabby et al. [2021] used the Ohio T1DM dataset to predict participants' blood glucose using both CGM and the context data using a deep learning model. Considering that the insulin bolus and meal carbohydrate content are discrete events, the authors proposed a semi-analytical formulation to spread them over time and convert them to a continuous variable. By doing that, all of the contextual variables (insulin, sleep, galvanic response, heart rate, steps, carbohydrate) become continuous and ready for feature extraction. The extracted features from the contextual information are fed into a stacked LTSM model for temporal decoding and finally to a deep learning model for classification. Using contextual information can lower the prediction error by 8% on average. It should be noted that similar efforts have been made by other studies Akbari and Chunara [2019], Martinsson et al. [2020].

Considering the improvements achieved in blood glucose prediction using contextual information, one might find the future of this field in doing the same for macro-nutrient prediction task as well as taking the game to the next level by making it cheaper, more thorough, convenient, and accessible. With the advent of commercial smartwatches and their sophisticated sensors, the context collection paradigm is shifted drastically. Previously, the collection of heartbeat, body temperature, ECG, and galvanic skin response time series were limited to only medical-grade wearables. However, nowadays, these data can be collected cheaper (the commercial smartwatch prices are one-third of the medical-grade ones), more accessibly (many of us are wearing them in day-to-day life), and more conveniently (wearing a smartwatch is much easier than wearing an accelerometer on the hip or a breathing band). Therefore, the next generation of smart dieting trials and datasets should benefit from these accessible, cheap, and accurate devices in addition to continuous glucose monitor readings.

2.3 Conclusions

Context is not a ubiquitous definition for analytic models on sensing systems. Contextual information obtained from wearable sensors can serve as building blocks to help the system to interpret the environmental parameters more efficiently. By evaluating context as a hierarchy of information, sensing systems can capture health and behavior from very discrete actions to long-term longitudinal health monitoring. Health and wellness monitoring with sensors can range from applications that require instant recognition of sensing (heartbeat to heartbeat for example) to long term context and trends. As a result, context recognition for sensing systems cannot be considered a single paradigm, but rather a hierarchical concept that builds and integrates. Context-driven classification helps improve systems for health and wellness applications because knowing the specific contextual state can at least focus: 1) sensor (and axis) selection; 2) features for recognition of a variety of lifestyle monitoring applications; and 3) enable longitudinal health and wellness modelling. Context is the key to unlocking the potential of sensing and biosensing for health and wellness applications.

3. FRAMEWORK

3.1 Introduction

Manual meal announcement and content estimation is an arduous, faulty process and cannot be trusted as it is often forgotten and accompanied by significant errors as big as 28 grams for carbohydrates on average Rhyner et al. [2016]. Therefore, it is essential to automate this process. Several attempts with different tonalities have been made in this regard, as discussed in the previous chapter Amft et al. [2005], Hodges et al. [2006], Dong et al. [2014], Sun et al. [2014], Pasler and Fischer [2014], Gemming et al. [2015], Gao et al. [2016], Zhang and Amft [2016], Huang et al. [2017], Zhang and Amft [2018], Luktuke and Hoover [2020]. In addition to privacy concerns of some methods such as monitoring visual and auditorial signals (for food images and chewing sound), these methods cannot estimate the content of the food accurately; on the other hand, models based on CGM readings can be helpful in this regard Zeevi et al. [2015], Staal et al. [2019]. Although PPGR can reflect the content of an eaten meal Zeevi et al. [2015], it is insightful only after a significant delay required for macronutrient absorption. Literature suggests that context-aware models perform better in smart health monitoring Rabby et al. [2021], Marling and Bunescu [2020], Akbari and Chunara [2019], Bertrand et al. [2021], specifically for the eating detection module hand gestures captured by wearables are of interest Dong et al. [2014], Sharma et al. [2020]. Therefore, in this study, I augment the contextual information of wearables to CGM readings to predict an eating moment sooner and more accurately. Additionally, I cater to the need for a macronutrient estimation model by providing the data required for estimating the content of a meal in the wild. In the absence of a dataset containing CGM and contextual information, we performed a pilot trial to cater to our needs, as explained later. Despite studies where medical-grade smartwatches were used Dong et al. [2014], Sharma et al. [2020], Rabby et al. [2021], we made the study more inclusive and financially more available by using typical off-the-shelf smartwatches. As discussed later, such a modification required us to develop applications for the data collection process.

In the following subsections, I explain the necessity of designing a new study and its key characteristics, the developed applications, and the methodology for processing the data.

3.2 Study Design

The role of contextual information in smart dieting has not received enough attention until recent years. Several recent studies made an effort to include wearables and other wearables for diabetic patients with the hope of augmenting data for a better prediction, including D1-NAMO Dubosson et al. [2018] and Ohio T1-D Marling and Bunescu [2020]. D1-NAMO consists of twenty healthy and nine type-I diabetic participants who logged their meals in addition to blood glucose, acceleration, ECG, and breathing measurements. Ohio T1-D contains twelve type-I diabetic participants who recorded their insulin, meal times and carbohydrate contents, EDA, heart rate, steps, and temperature. Although these two datasets contain some contextual information, they fail to provide the hand gesture features necessary for eating detection as explained by Dong et al. Dong et al. [2014]. Also, they are more focused on type-I diabetic participants whose body metabolism and PPGRs behave differently from normal healthy ones. Therefore, to cater to our needs, we designed a new study where participants wear a smartwatch and a CGM in addition to logging meal timing and macronutrient contents. Although most of the commercial CGM sensors are reliable and offer comparable collection frequency and accuracy, choosing the right type of smartwatch can be challenging as it should 1) be commercially and financially available, 2) contains an accelerometer and gyroscope for measuring hand gestures, and 3) offers the option for easy and robust application development for data collection. Below, I compare some of the most common wearables to choose the most appropriate one for our trial.

Several wearables offer contextual information, including Empatica, Apple, Fitbit, and Shimmer. Table 3.1 compares some of their specifications. Considering the studies done by Dong et al. Dong et al. [2014], the presence of gyroscope and accelerometer readings can help detect hand gestures during eating periods. Therefore, Empatica E4 is not a viable option as it suffers from not having a gyroscope sensor. Similarly, Empatica Embrace 2 does not currently offer rotation data, although it has a gyroscope sensor. Fitbit Sense measures both acceleration and rotation, but it does not save them, so a custom application should be developed to collect such data. Upon further investigation, I realized that the watch is not capable of buffering such data with a high-frequency (> 1 Hz) for more than a couple of minutes, mainly due to the absence of a powerful operating system and low internal memory storage; therefore, it is not a viable option. Shimmer 3 IMU unit is a promising option and has been used by other studies Dong et al. [2014]; however, it is not commonly available among participants (in comparison to off-the-shelf smartwatches such as Apple watch or Fitbit), and also it does not offer the health data (such as burnt calorie, heart rate, and temperature) which might be helpful in macronutrient estimation Rabby et al. [2021]. Also, its price is drastically higher (two or three times) than other wearables. Therefore, I decided to use the Apple watch as the primary smartwatch as it gives both motion and health data, it is commonly used, comfortable to wear, financially accessible, well-documented, and tested. To include more health data, I decided to use Emaptica E4 as the secondary watch for further exploration and measuring the impact of its precise high-frequency data in the eating detection model.

Unite	Acclerometer	Gyroscope	Common	Health
Empatica E4	✓	✗	✗	PPG, EDA, Thermo
Shimmer3 IMU	✓	✓	✗	✗
Empatica Embrace 2	✓	✗*	✗	Thermo, EDA
Apple Watch 6	✓	✓	✓	HR,Act, Cal, Step
Fitbit Sense	✗*	✗*	✓	Cal, Step, HR, Act

Table 3.1: Motion and health sensor for Empatica E4, Empatica Embrace 2, Apple Watch 6, Fitbit Sense, Shimmer 3 IMU wearables. PPG, EDA, Thermo, HR, Act, Cal, Step denotes photoplethysmography, electrodermal activity, thermometer, heart rate, activity type/summary, burnt calories, steps. The stared cells are not commercially available or not providing data with the required resolution.

3.3 Developed Applications

Contextual information, including hand gestures, metabolism, and energy expenditure, can be important to smart dieting studies. As discussed previously, in smart diet trials, the meal timing, macronutrient composition, and participant activities should be logged and recorded. Also, the contextual information from wearable sensors such as temperature, heart rate, electrodermal activity, hand acceleration, and rotation can be helpful. Therefore, we developed two applications to allow 1) collecting participant inputs (meals and activities) and gathering smartwatch sensor readings as well as 2) helping analysts to match meal information with its pictures.

3.3.1 iPhone-iWatch Context Collector Application

The application should collect motion and health data from smartwatch sensors and transfer it to the Cloud robustly and accurately. The term robust refers to being able to buffer the data and handle Internet outages (even up to a day), while being accurate means to align the sensor data with its corresponding time and save it with enough precision. The application consists of a phone module and a companion one for the wearable. The companion wearable module should be standalone; otherwise, any disruption on the phone module (i.e., if the user accidentally closes the application on the phone) negatively affects the data collection process. Also, the companion should be able to communicate with the phone module to temporarily store the data (in the case of no Internet coverage) and transfer it to the Cloud due to the following reasons. Routing the data from the watch to the phone and subsequently to the Cloud avoids further battery consumption in the smartwatch, considering that its battery is severely limited compared to phones. Also, it prevents the necessity of the watch from being directly connected to the Internet through cellular (which is more expensive due to the required monthly subscription) or WiFi (extra battery drainage). Such a architecture is shown in Figure 3.1

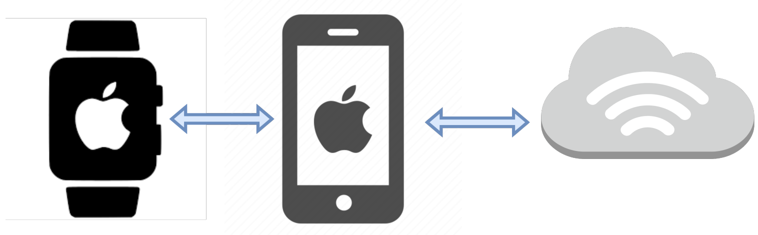


Figure 3.1: Schematic of application modules showing the rout of data from the watch to the Cloud.

This application was designed and developed using SWIFT language in collaboration with Amin Hamiditabar and its source code can be found under our GitHub repository. The next subsections explain it features and architecture.

3.3.1.1 Phone Module

The phone module should provide the option to log the user meals and activities in the front-end side as well as handle the watch-phone and phone-Cloud connections on the back-end. Figure 3.2 shows the application GUI which allows users to log various types of activities (exercise, travel, sleep, work and other annotations) as well as meals with its macronutrient information (calories, carbohydrate, fat, protein, ingredients and eaten portion). We enforced the insertion of meal pictures to help us checking and retrieving lost macronutrient information in subsequent processing. Although logging meals and activities in real-time can help avoid forgetting it, we provided the option to log an old input/activity.

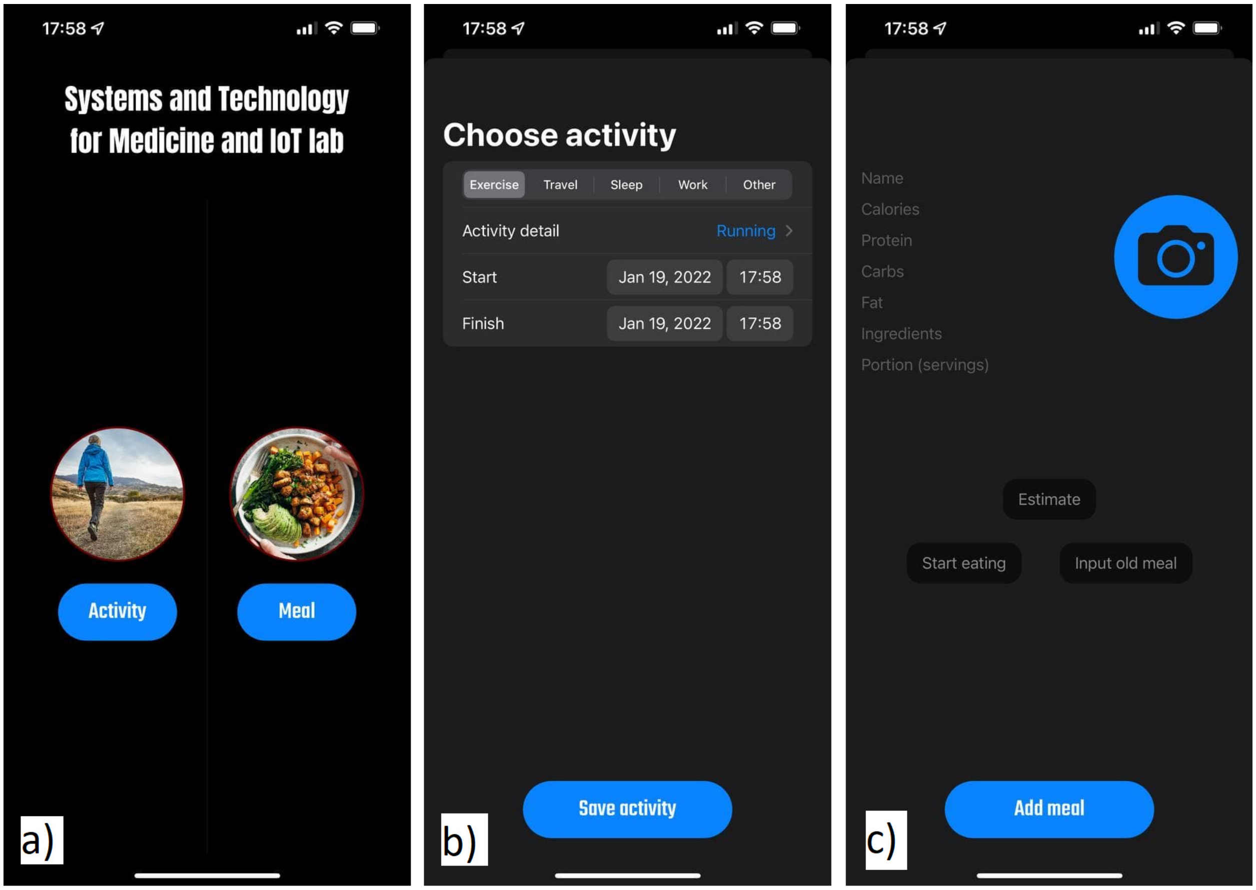


Figure 3.2: Phone module snapshots a) main menu, b) activity logger and c) meal logger.

Just-in-time interventions and real-time processing of data is of interest in smart dieting. To cater for this need and monitoring the trials (to ensure that participants are in compliance to the study instructions), I designed the application to send the data packets received from the watch to the Cloud with a minimal lag. We used FireBase service developed by Google as it offers free, reliable and secured connection (to satisfy HIPPA and IRB requirements) to phones where the data can be stored for further processing. Considering that several participants are recruited for each round of a trial where all of them should be able to use this application at the same time, it should be able to correctly recognize and classify participant's data; therefore, an identification method is required which should not allow revealing the identity of the participants. Hence, I used the participant ID for this matter and the FireBase agent classified the data upon receiving them. The participant ID should be entered only once after the installation of the application on the phone (by the recruiter).

3.3.1.2 Wearable Module

The data collection frequency is crucial as it can severely affect the battery life of the watch and eating detection and macronutrient estimation process. Although Apple watch offers the core motion data with the frequency of up to 64 Hz, doing so limit the battery life to less than a day, which negatively affects the trial (considering that the nocturnal data are not needed, participants can charge up the battery and phone every night). Therefore, to balance the battery life and data precision, I chose the frequency of 10 Hz as the similar value is suggested by the literature Dong et al. [2014]. Although the developer can control the core motion frequency, Apple iOS directly handles the frequency of healthkit data reports (heart rate, calories, and steps are reported sporadically depending on tonality and severity of the physical activity).

Considering the limited battery life on wearables, operating systems take radical actions to assure killing abusive applications that use resources (i.e., battery, RAM, CPU) tremendously. This posed an extra challenge to our application as it should monitor the participant motions for a significant portion of a day (10+ hours) which iOS counts as abusive usage. This resulted in our application to get a forced closure after two to three hours. The iOS shows some flexibility and allows applications to run longer during sessions (i.e., workout, focus); however, this option is limited to six hours. So, the participant has the burden of manually putting the application into session mode. To mitigate this issue, I could avoid iOS closing the application by making inquiries about the location because this makes the application counted as a navigation one that is permitted to run for an extended amount of time. Therefore, although the location is frequently acquired, it is not saved (this explains the arrow sign on the watch screen when the application is collecting data).

3.3.2 Meal Match Graphical Application

Extracting the timing and contents of a meal in a smart-dieting trial is crucial, arduous and prone to errors. To alleviate this burden, I designed and developed an application to ease this process through a mix-and-match graphical interface. This application can be used alongside of the previous one (meal logger) to both help modify the content of a meal as well as filling the missed meals through pictures captured by the user.

The application GUI should be flexible and robust to allow easy, fast, and accurate matching meals with their pictures. The interface of the application can be seen in Figure 3.3, and it contains the following elements: 1) meal information (red box), 2) meal pictures (blue box), and 3) control module (green box).

A summary of each meal information alongside its start/end timing and some controlling modules are offered for each meal as shown in the red box. The "Start Pic" and "End Pic" columns are initially empty and shall be filled by the analyst using the meal picture table (blue box) through drag and drop. "Ratio" is the amount of the eaten meal estimated by the analyst visually. "Note" is an editable column and allows the analyst to leave notes for further processing about each meal. The "Modify" column allows changing/adding constituents which will be discussed later.

The meal pictures in the blue box allow a visual detection by the analyst to match meal pictures and their names. After recognizing the meal and matching it to its picture, the analyst can simply drag and drop the meal picture from the blue box to a corresponding row in the red box. This process sets the start and end times of the meal in the back-end side of the application and updates the CSV file. To ease the matching process, all of the match pictures are grayed-out as shown in Figure 3.3.

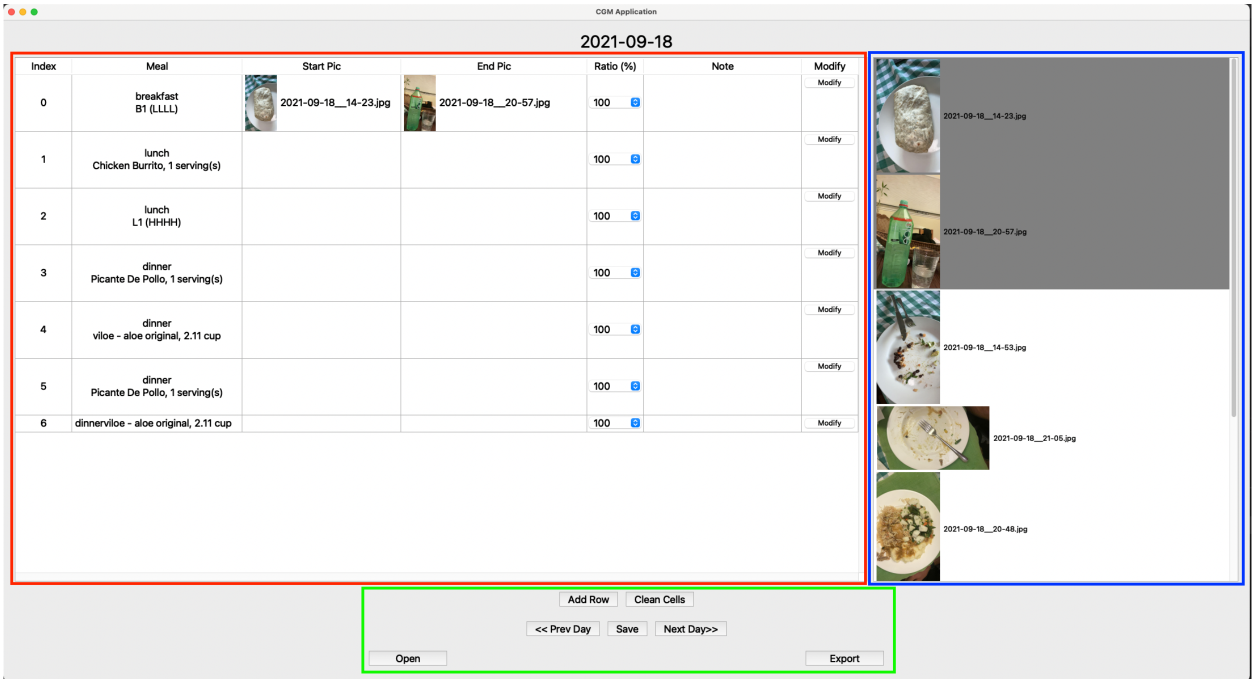


Figure 3.3: Schematic of the application GUI, which is color-coded as red, blue, and green for meal information, meal pictures containing the time of the meals, and control module.

The control box (green box) gives control to the analyst to add/fix records. Most of the logged meals are recorded during the trial by the participant; however, the analyst might need to access and modify their details. There are some exceptional cases where the analyst should manually insert or delete a meal. This can be done by using "Add Row" and "Clean Cells," where the first one makes a new row entry while the latter one cleans selected cells in the red box. To make the visual detection easier, we filter the pictures in the blue and red boxes by dates, and the analyst can navigate through different dates using "« Prev Day" and "Next Day» buttons. The application automatically saves the changes when moving to the next or previous date to avoid data loss. Finally, "Open", "Save" and "Export" buttons are used to open a new CSV file for processing, temporarily saving the changes and finishing the file processing.

The application should be flexible and offer an option to cater to the need for manual modifications by the analyst in exceptional cases. I designed and placed the "Modify" column, containing a "Modify" button for each row. By clicking the button, a new window pops up and offer meal name, type, and macronutrient in editable text boxes, as shown in Figure 3.4.

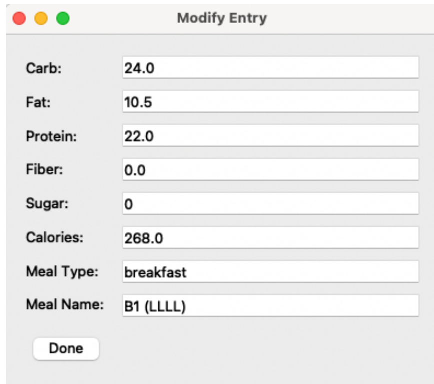


Figure 3.4: Modifying option to change and correct meal information including name, type, carbohydrate, fat, protein, fiber, sugar and calorie of each meal.

This application is developed using Python language and PyQt5 package and can be found under GitHub repository.

3.4 Methodology

Eating detection and macronutrient estimation models use the past readings to predict future meals and their contents. In such supervised modeling, the extracted features from CGM and contextual time-series are fed to the models to predict the labels by a regression model to handle the carbohydrate, fat, and protein content of the meal and a binary classifier to discern eating from non-eating moments. Sensor data often suffer from collection noise that requires attention and filtration. An additional challenge is to correctly map the sensor data to labels where the temporal relation of the data is not lost. My approach to analyze the CGM readings and contextual information consist of three main steps: data pre-processing, data windowing, feature extraction, and modeling.

3.4.1 Pre-processing

Sensor data and user inputs collected in the field environment are subject to various noises, errors, and degradation in quality; therefore, proper smoothing, correction, and filtration are often required. Literature suggests that CGM errors can benefit from the Kalman filter as they are often accompanied by normal noise Staal et al. [2019], Rabby et al. [2021]. Therefore, I applied the Kalman filter on the CGM time series and interpolated the results up to a minute. Similarly, I used the exponential move mean average technique over the acceleration and gyroscope data with the window of one second as suggested by Dong et al. [2014]. Although users were strongly encouraged to log their meals and activities right after, several instances required further attention before getting analyzed as the participant inserted the wrong time (AM vs. PM) or date. Unlike the clinical trials where the participants are limited to eating only at some time of the day or to having a predetermined meal decomposition Huo et al. [2019], Sajjadi et al. [2021], Paromita et al. [2021], this trial did not limit the participants in any way. Therefore, it is expected to see more variation and unexpected results such as mealtime longer than usual (i.e., a participant has a glass of oat milk in 45 minutes or snack while working). After reviewing the data, I had to exclude two participants' data due to missing meal logs and smartwatch data loss.

3.4.2 Data Windowing

Eating moment detection models depend on the temporal variation of BG and hand motion; therefore, a proper representation should be able to capture and resolve such changes. As an example, Figure 3.5 shows a general profile of PPGR after consuming a meal where BG starts to rise after a lag and then drops as the meal is absorbed and metabolized. Deep learning models can address this characteristic and resolve the temporal relation in BG variations using LSTM (long short-term memory) or GRU (gated recurrent unit) layers; however, such a technique gives robust results only in the presence of an abundant amount of accurately labeled meal. On the other hand, for cases with a sparse dataset, the sliding window can be used as denoted in the literature Mishra et al. [2020], Minor et al. [2017], J. et al. [2016].

The sliding window method slices a time series into several sub-samples where each one contains a label and is treated as an independent observation for further statistical analysis. Activity and BG variations have a temporal scale of minutes, while accelerometer and gyroscope vary abruptly in the order of a second. Therefore, two sliding windows with different temporal resolutions should be defined to capture the micro (minute) and macro (hour) resolutions. The outer window acquires the CGM data and decides about the eating meal flag. On the other hand, the inner window, which is shorter and located inside the outer one, inquires about the core motion and health data. For example, if the outer window denotes 12:40 to 13:40, the inner window loops through it minute by minute and extract the contextual data as shown in Figure 3.5.

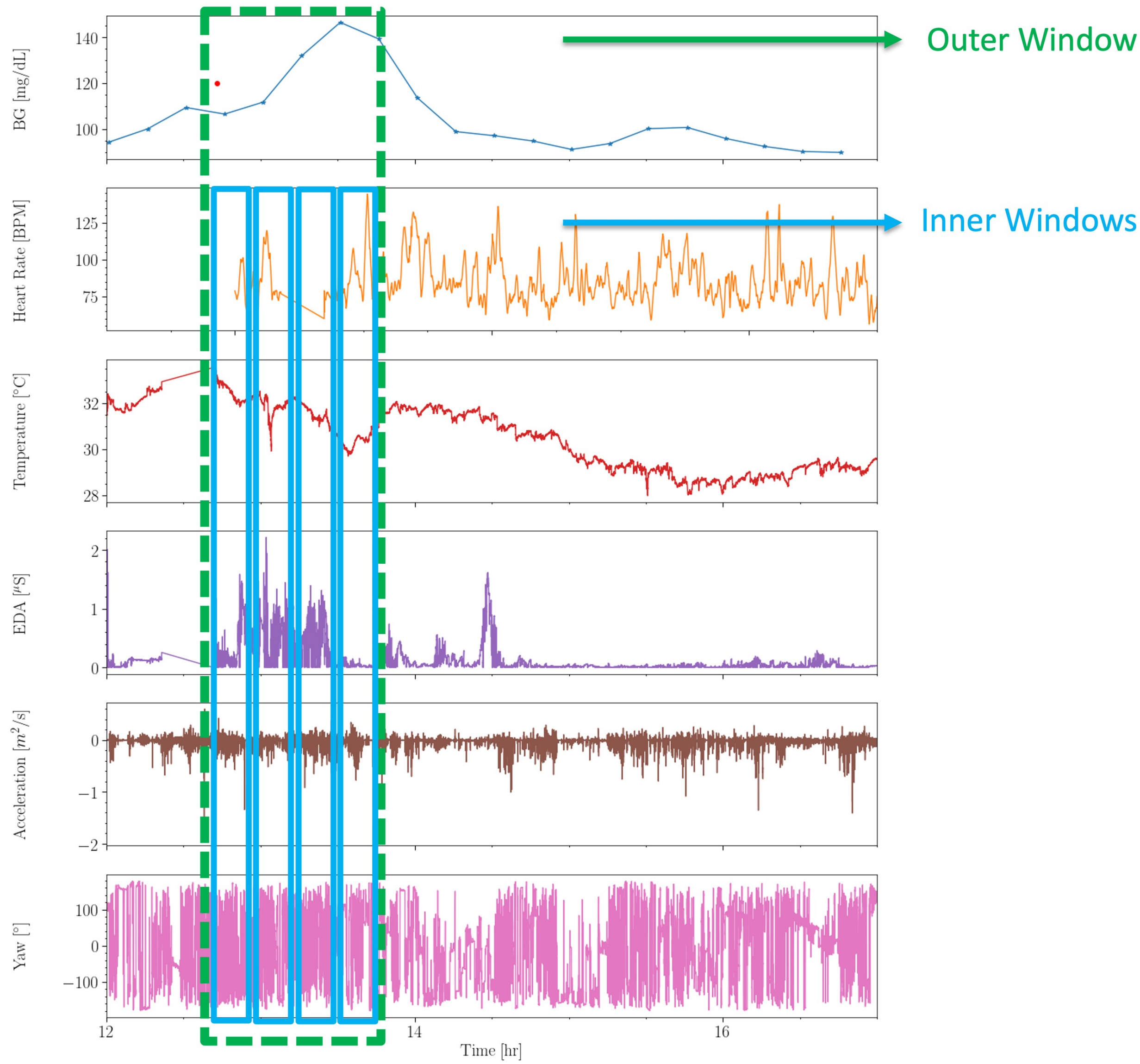


Figure 3.5: The schematic shows the sliding window technique with the outer window (dashed green) and inner ones (dashed light blue) on CGM (top panel) and smartwatch data (bottom five panels) where the outer window is about 1 hour and the inner one is 25 minutes long. For the sake of better visualization, I made the inner window longer. The inner window slides inside the outer one, as shown. The red dot in the top panel shows the meal time for the participant and CGM signals start to rise with about 45 minutes lag after the meal.

The outer window should slide through time to capture the daily picture of participant activity and diet. To avoid any complications in labeling the window, the overlap is avoided between outer windows. The positive ones start before a meal and are followed by negative ones until the next positive window is met. In other words, if a positive outer window begins at 12.40 and lasts for an hour, then a negative window starts at 13.40 and repeat every hour till the next meal as shown in Figure 3.6 (the next meal is later in the day and not shown in the figure).

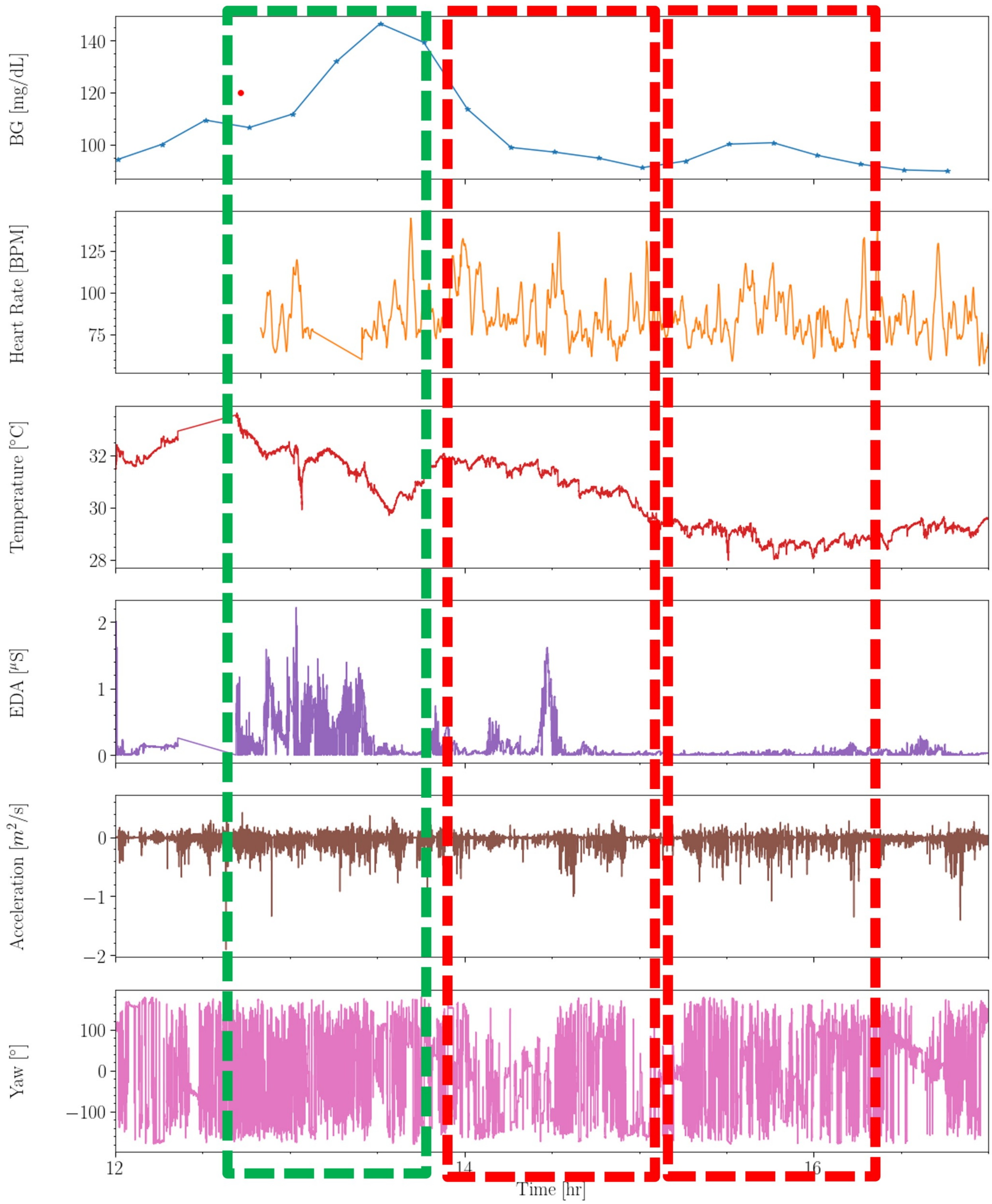
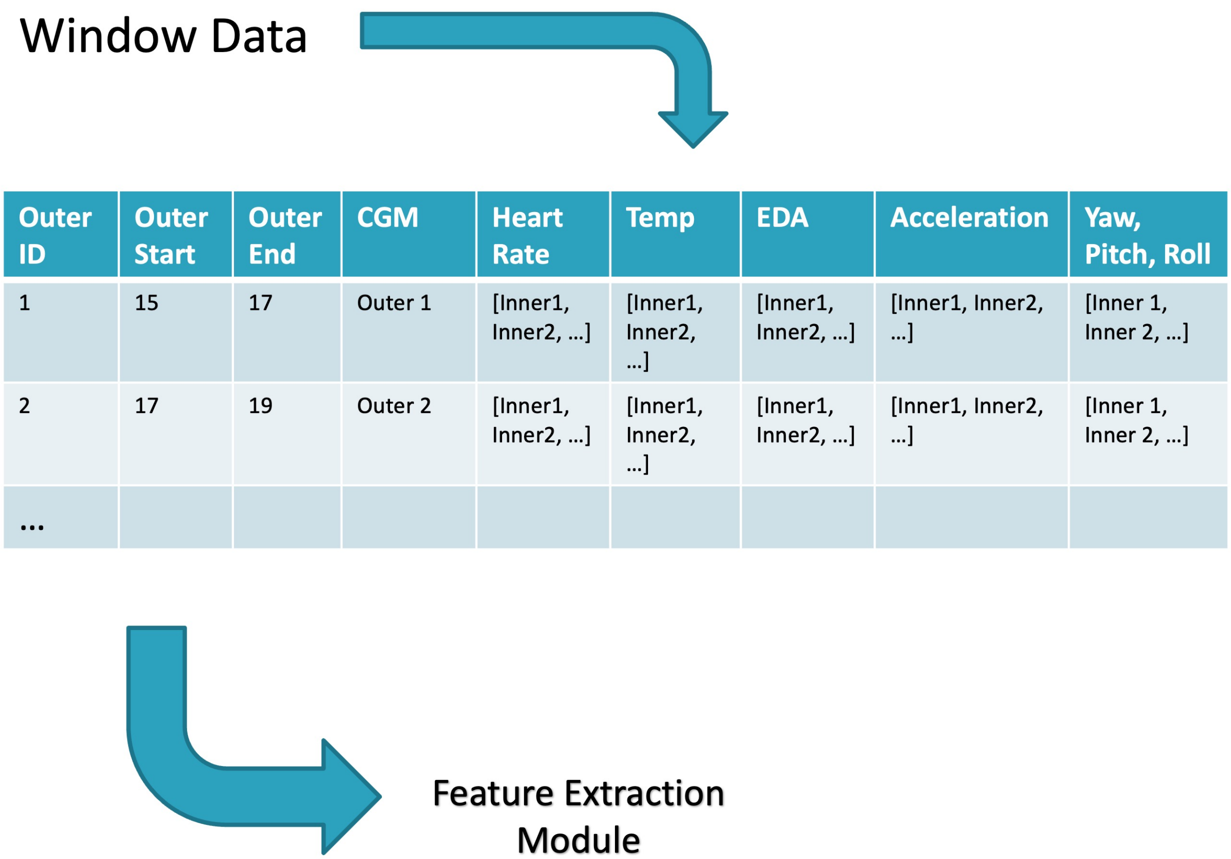


Figure 3.6: The sliding of outer window through time with positive (green dashed lines) and negative (red dashed lines).

After acquiring the information of each window, one can form a table of window information containing all the motion, health, BG data and eating/non-eating labels as shown in Figure 3.7. After creating such a table, it should be submitted to the feature extraction module to be presented to the model.



Outer ID	Outer Start	Outer End	CGM	Heart Rate	Temp	EDA	Acceleration	Yaw, Pitch, Roll
1	15	17	Outer 1	[Inner1, Inner2, …]	[Inner1, Inner2, …]	[Inner1, Inner2, …]	[Inner1, Inner2, …]	[Inner 1, Inner 2, …]
2	17	19	Outer 2	[Inner1, Inner2, …]	[Inner1, Inner2, …]	[Inner1, Inner2, …]	[Inner1, Inner2, …]	[Inner 1, Inner 2, …]
…								

Figure 3.7: Each row of the table shows the extracted data for an outer window. Each row contains several sub-lists of different sensors such as CGM, heart rate, temperature, EDA, accelerometer and gyroscope.

Instead of directly feeding the model with raw accelerometer and gyroscope data, the literature suggests translating them into parameters that represent the dynamics of the motion more clearly. Therefore, the core motion data extraction module translates the acceleration and rotation data based on the formulation offered in Equation set 3.1 Dong et al. [2014], Sharma et al. [2020].

Unlike the core motion, health and CGM data do not show significant variations during the length of a minute (inner window length), and they are directly understandable for the model. Therefore, I took the averaged value for heart rate, EDA, and temperature for each inner window and passed it to the feature extraction module.

$$\begin{aligned} \text{Rotational to linear acceleration feature } &= \sum^{W} \frac{|S_\phi| + |S_\theta| + |S_\psi|}{|S_x| + |S_y| + |S_z|} \\ \text{Linear acceleration feature} &= \sum^{W} |S_x| + |S_y| + |S_z| \end{aligned} \tag{3.1}$$

where W, S_ϕ, S_θ, and S_ψ are window size, yaw, pitch, roll and S_x, S_y, S_z are linear acceleration.

3.4.3 Feature Extraction

Although the sliding window can break a lengthy time series into smaller sub-samples, the representation is still not complete as they should be summarized through population informant metrics. Therefore, inner window raw data are passed to the feature extraction module, and the extracted features are fed into the models. Figure 3.8 shows the relation between modules and the pipeline of data and feature extraction.

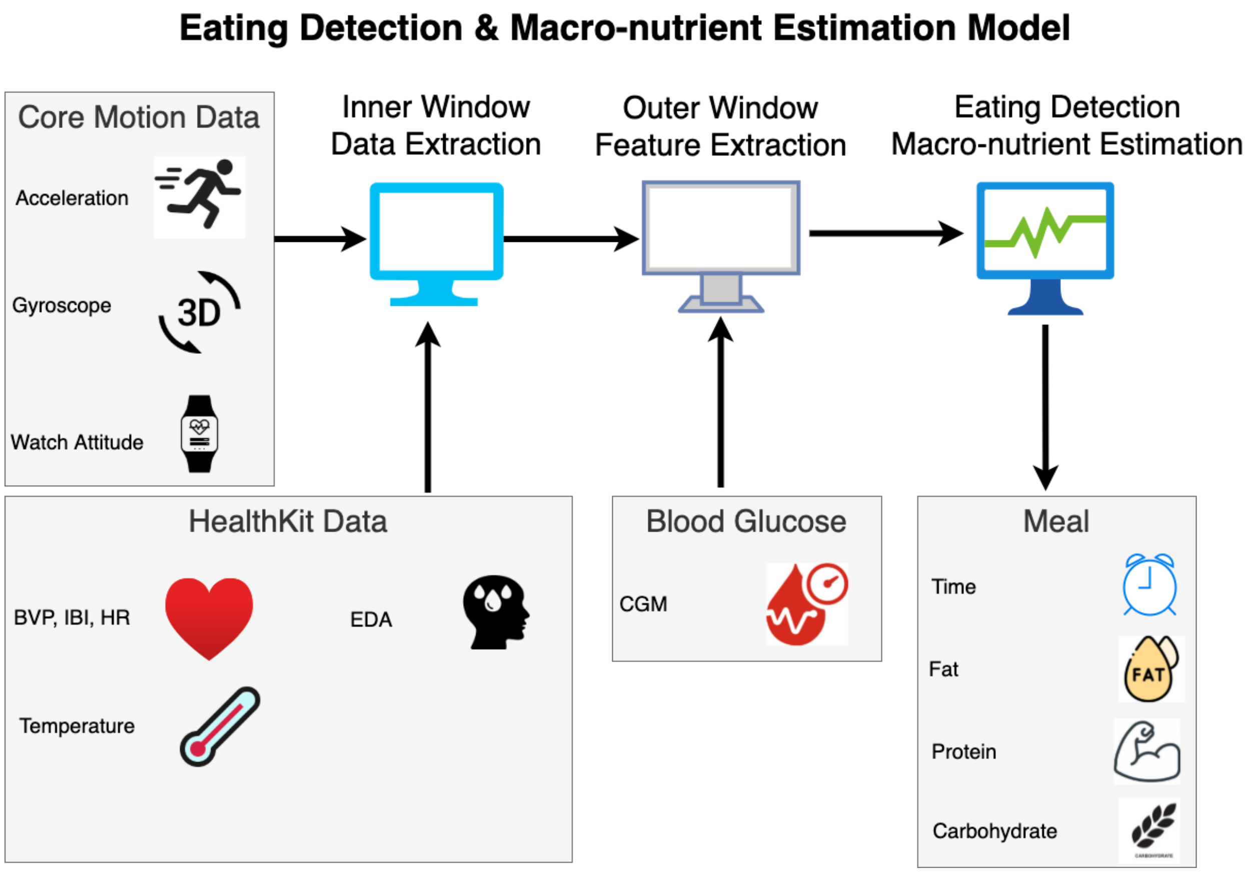


Figure 3.8: Schematic of the eating moment detection and meal macro-nutrient estimation.

3.4.4 Model

Considering the nature of the dataset and its sparsity, tree-based models are a plausible choice. For the hand motion, I benefited from the existing dataset to train a model to detect eating from hand gestures and subsequently transfer it to our collected context-aware dataset. Additionally, there is a need for another multi-modal model to consider both hand motion as well as CGM and health data. The sections below discuss the details of each model thoroughly.

3.4.4.1 Eating Detection Using Hand Motion

Pre-trained models and domain generalization can be interesting when the collected dataset is small or significantly imbalanced, and the literature offers a significant amount of labeled data. To benefit from the studies done by Sharma et al. [2020], I made an effort to build a model trained by previously collected contextual information and test it on my dataset. Such a domain generalization requires standardization and normalization to remove any difference between populations.

I exploited an XGBoost classifier (hereinafter XG1) to perform binary classification (eating vs. non-eating) on sliding window data captured from the Clemson dataset with satisfactory results. The Clemson data is publicly available, which can be accessed at Adam Hoover's project website and explained in Sharma et al. [2020]. It contains 354 participants who wore a shimmer unit for one day in a free-style living and manually logged the beginning and end of the meals. The dataset offers the start and ends timestamps of meals, acceleration in X, Y, and Z directions, and yaw, pitch, and roll. I applied the sliding window technique with the length of one minute on the data and labeled windows that occurred during a meal as positive and the rest of them as negative. To make the model extendable to all participants, I normalized the extracted features of windows by removing the mean and dividing by the standard deviation for each participant.

To avoid overfitting, I performed 5-fold cross-validation where 30 % of each fold train data was dedicated to validation through a stratified splitter (to assure fair and balanced division). The model was tuned (against the validation set) for various values of the number of estimators (50 to 400) and max depth (3 to 7). The best model chosen based on the ROC-AUC has the number of estimators and max depth of 150 and 5 on validation data. To tackle the imbalance ratio of positive to negative windows (1 to 17.2), I employed both the Synthetic Minority Over-Sampling (SMOTE) technique and higher positive class weights which had similar results. Figure 3.9 shows the XG1 ROC curve. Repeating the experiment with Random Forest resulted in slightly lower performance.

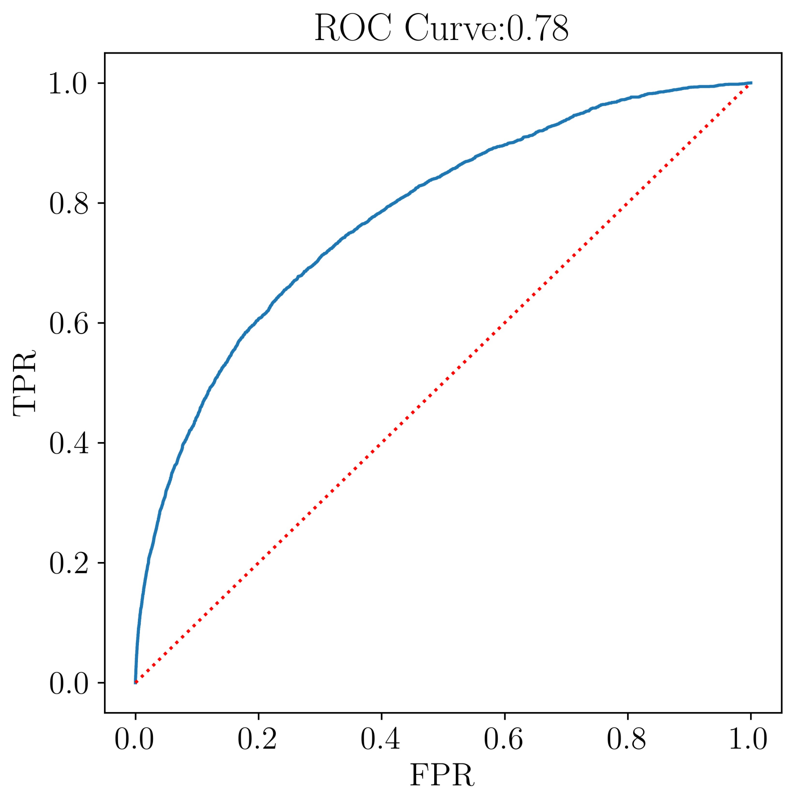


Figure 3.9: The ROC curve of XG1 trained and tested on Clemson dataset with the area under the curve of 0.78.

Having a high value for ROC-AUC guarantees that there is a threshold at which the model can provide satisfactory metrics. Testing XG1 on Clemson test fold gives the averaged accuracy-weighted, recall, specificity, precision-weighted, and F1-weighted score of 71%, 66%, 75%, 93%, and 81% respectively. Figure 3.10 provides the confusion matrix. Considering that the dataset is highly imbalanced, a weighted version of precision, accuracy, and F1-score are of interest to provide a robust picture Dong et al. [2014]. Such a performance is similar and comparable with the literature Dong et al. [2014], Bertrand et al. [2021].

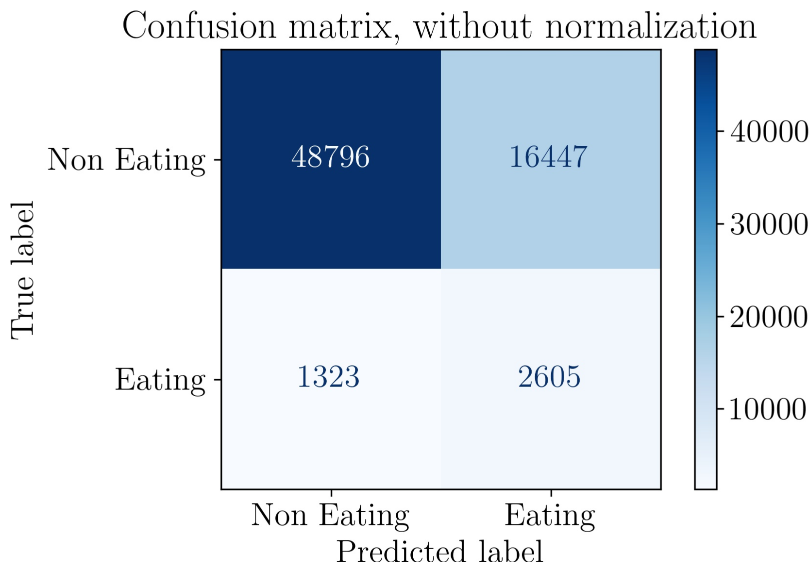


Figure 3.10: The confusion matrix of XG1 trained and tested on the Clemson dataset. The eating and non-eating labels are provided for each row and column. The ground truth (True label) is shown in rows, and predicted ones are denoted as columns.

3.4.4.2 Multi-Modal Eating Detection Using Hand Motion, CGM and Health Data

To benefit from the existing datasets on contextual information, I augmented the XG1 model with CGM and health data in a new XGBoost model (hereinafter XG2). I developed XG2 to account for CGM and health data (temperature, heart rate, and EDA). XG2 is fed by the top consecutive five minutes prediction probability from XG1 and extracted features from CGM and health data. Considering that XG2 does not have enough positive data to afford hyperparameter tuning and the minimal sensitivity of XG1 to hyperparameters, I skipped the tuning phase and used a typical XGBoost model with the number of estimators and max depth of 200 and 3, respectively, with a 5-fold cross-validation for each participant. Please note that the hand motion model XG1 is a general one as participants share similar gestures while XG2 is a personalized model to account for BG variation and PPGR differences as suggested by Sajjadi et al. Sajjadi et al. [2021]. To test the efficiency of each sensor in the detecting process, I defined several combinations and compared their predictions as discussed in the next chapter.

4. EXPERIMENTS, RESULTS & DISCUSSION

4.1 Introduction

In this section, 1) I review the meal information of recruited participants, and 2) the performance of eating detection model in different case scenarios and combinations.

4.2 Pilot Trial

We conducted a 10-day data collection trial on healthy participants who wore an Abbott FreeStyle Libre Pro CGM (which samples interstitial glucose every 15 minutes), a Dexcom G6 Pro CGM (samples interstitial glucose every 5 minutes), an Empatica E4 smartwatch, and an Apple Watch Series 6, and were asked to log all activities and context changes on a custom smartphone application. This study was approved under Texas A&M IRB 2019-0793 and was piloted on eight healthy participants, aged average (standard deviation) of 26.8 (3.8) years and less than 35, with the body mass index < 35, resting heart rate < 120 per minute, blood pressure $< 140/90$, and no known history of cardiovascular disorders or diabetes. Out of eight participants, three were female. We explained the objectives, process, limitations, and benefits of the project to the recruited participants. Participants were asked not to postpone the meal and activity logging as it can cause errors in timing and content. The custom-developed application (discussed in the previous chapter) was installed on their iPhone (two Android-user participants benefited from the lab loaner iPhone) on the first day of trial during the CGM insertion. Throughout the trial, recruiters were in touch with the participants to ensure a successful experience. Participants were asked to wear both smartwatches during the day, put them into charge (sync mode for E4) right before going to bed, and put them back on first thing in the morning. We dropped the first day of each round as the CGM readings were inaccurate (warming up). Also, we had to discard two participants' data due to lack of adherence, resulting in 1,392 hours of data. Table 4.1 offers some information about meals consumed by each participant.

ID	Num of Meals	Duration (std)	Calories (std)	Carb (std)	Fat (std)	Protein (std)
p1	35	14.8 (17.4)	401.7 (219.0)	45.6 (25.5)	13.6 (12.6)	17.9 (12.9)
p3	44	17.7 (32.3)	328.8 (289.4)	5.6 (5.4)	21.1 (19.1)	21.5 (29.2)
p5	21	18.9 (9.1)	477.7 (369.5)	52.9 (38.8)	21.1 (21.2)	23.9 (23.3)
p6	42	23.8 (18.6)	731.1 (503.2)	84.9 (68.6)	30.8 (28.0)	36.5 (35.7)
p7	55	12.2 (13.5)	308.4 (219.2)	46.0 (36.2)	12.8 (20.7)	10.7 (10.8)
p8	57	9.3 (5.9)	441.7 (306.3)	46.5 (31.6)	15.8 (16.2)	22.7 (22.2)

Table 4.1: Participants' meal information and variation are categorized by the number of meals, duration, calories, carbohydrate, fat, and protein. For all columns (except ID and Num of Meals) the value is an average while the parenthesized values denotes standard deviation. The values for meal duration are in minutes while carbohydrate, fat and protein are measure in grams.

4.3 Eating Detection Results

This section discusses the results of XG1 and XG2 models in different case scenarios and analyzes the efficiency of sensor combinations in the eating detection process. In this chapter, I make an effort to analyze the results and model performance by answering the following questions.

1. Can the watch identify eating moments?
2. Is the eating detection using CGM more accurate than motion?
3. How is the detection by motion and CGM affected by the length of the outer window?
4. Does the CGM model benefit from motion and health data augmentation?
5. Can the motion help shorten the CGM detection period?

Answering these questions requires a thorough analysis where all combinations of sensors and environmental parameters are considered. Therefore, I defined the following sensor combinations: CGM, CM, CGM+CM, CGM+CM+Health, where CM and Health are core motion and health data (heart rate, EDA, and temperature), respectively and outer window length of 15 to 90 minutes with the step of 5 minutes.

4.3.1 Eating Detection Using Motion

XG1, a pre-trained model on the Clemson University dataset (as explained in the previous chapter), detects eating moments solely using features extracted from hand motion obtained from smartwatches. By normalizing the data collected in the trial and feeding them to XG1 model, I was able to perform a successful domain adaptation. High ROC-AUC values in Figure 4.1 confirms this assertion. Obtaining high ROC-AUC values assures that there is a threshold at which the data can properly get classified and categorized. Based on Figure 4.1 most of the participants can benefit from the core motion data in the eating detection as ROC-AUC is ≥ 0.75 (aside from P7 with ROC-AUC of 0.68).

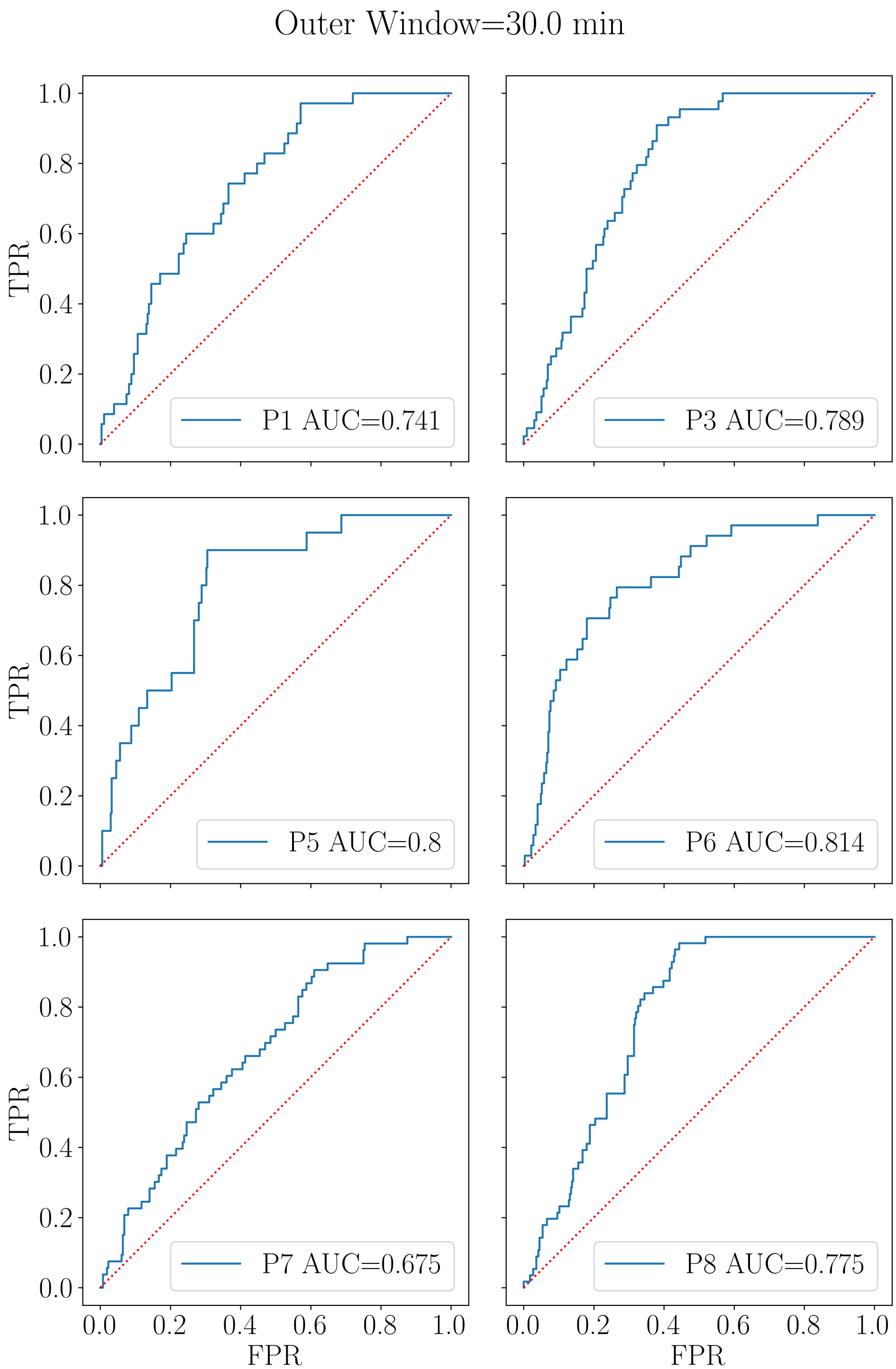


Figure 4.1: The ROC curve of XG1 tested on our context-aware dataset for each participant.

4.3.2 Motion vs CGM in Eating Detection

Identifying eating moments through motion can be more challenging than CGM for meals with an extended eating time. In day-to-day life, we usually don't finish a meal in a rush; instead, we often enjoy watching TV or listening to the radio while eating, which extends a meal (i.e., 17 minutes instead of 5 as shown in Table 4.1). This absence of hand gestures during such time can deteriorate the detection through motion and make it more challenging. On the other hand, the PPGR will still respond robustly (though with a more considerable lag) to the consumed macronutrients. Therefore, we expect to see a better detection using CGM than motion in such a setup. Figure 4.2 can be helpful in this regard as it compares the eating detection recall using motion and CGM against each other. Figure 4.2 confirms such an assertion as the lines fitted to CGM mostly land above the CM except for P3 and P6. However, it should be noted that P3 was on a Ketogenic diet which prohibits the consumption of carbohydrates and instead motivates using fat and protein. As discussed earlier, such an anomaly suppresses the BG abrupt rises, which is the most prominent way of eating detection for XG2 using CGM data. Therefore, it is expected to see a low performance for P3 using CGM.

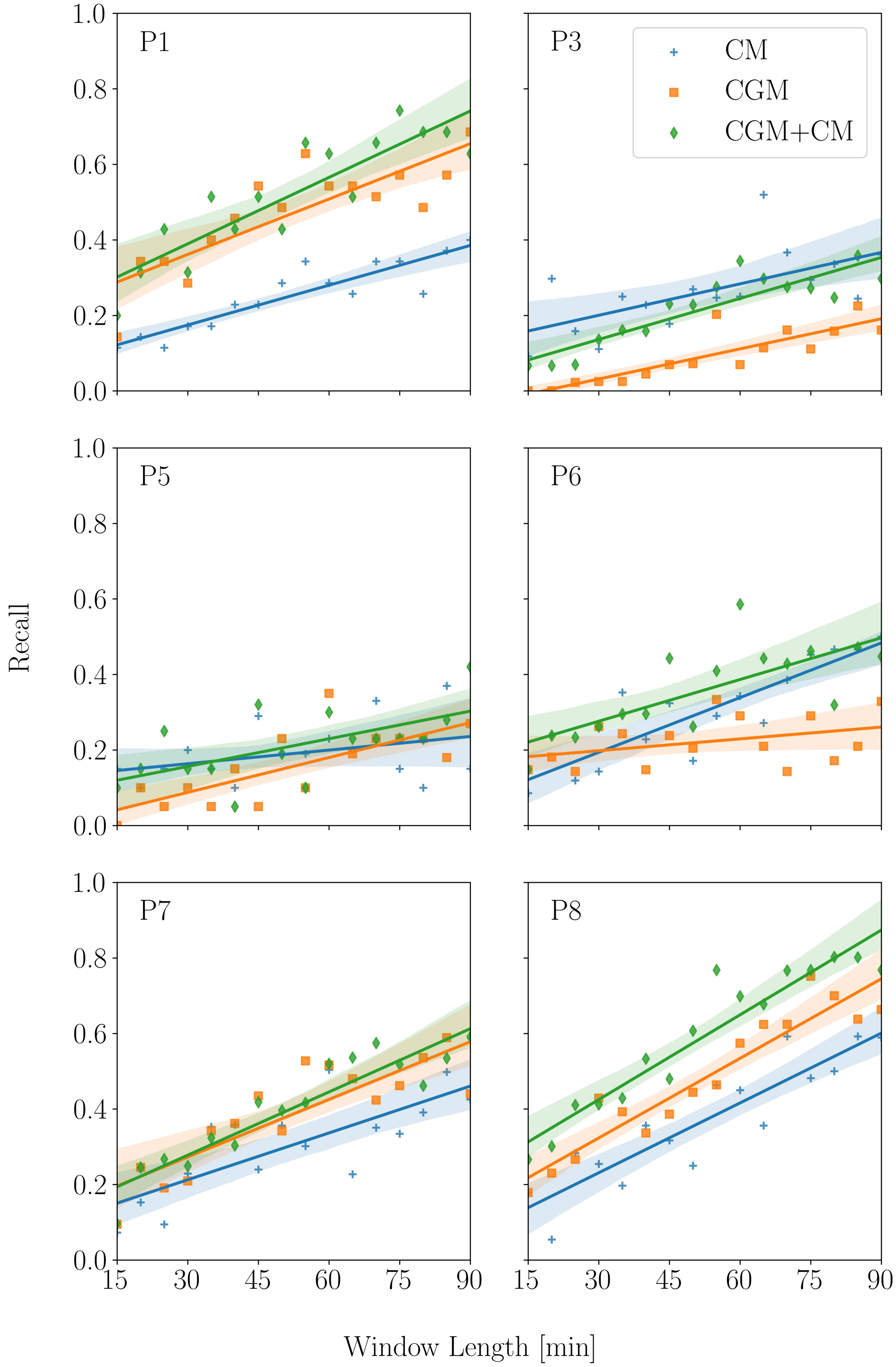


Figure 4.2: XG2 recall for all participants across different outer window length. Each panel contains CGM, CM and CGM+CM recall curves in orange, blue and green colors. The lines and their shades demonstrate the linear regression and 95% of confidence interval. The CGM+CM+Health combination is not plotted as it has similar values to CGM+CM.

4.3.3 Role of Length of Retrospective Information in Eating Detection by Motion

It is interesting to analyze the effects of retrospective information length (outer window duration) on the ROC curve of eating detection using core motion data. To answer this question, I ran the model with a variety of outer window durations as shown in Table 4.2 which suggests high ROC-AUC values for all participants/window lengths. The variation among participants' ROC-AUC is higher when the window length is small (i.e., 15 minutes); however, as the window becomes bigger, participants tend to have closer AUC and converge toward the Clemson dataset baseline (0.78). In other words, for participants with significantly higher AUC, such as P5, widening the outer window lowers the AUC. On the other hand, for ones on the other side of the spectrum, such as P7, this causes an improvement.

Retrospective [min]	P1	P3	P5	P6	P7	P8	STD
15	0.73	0.79	0.84	0.79	0.68	0.78	0.056
30	0.74	0.79	0.80	0.81	0.66	0.76	0.055
45	0.76	0.74	0.78	0.82	0.70	0.79	0.042
60	0.76	0.77	0.79	0.83	0.70	0.78	0.042
75	0.79	0.75	0.74	0.83	0.72	0.77	0.039

Table 4.2: ROC-AUC of eating detection using motion for participants with various length of retrospective (outer window) information. The last column denotes the standard deviation of the row.

4.3.4 Motion and Health Data Augmentation to CGM

Knowing that motion data is informative in making decisions about eating moments, I contributed the probability predicted by XG1 in the XG2 model. Intuitively, one can expect to see better results by adding informative sensors. So, I measured XG2 performance against the combinations mentioned above (CGM, CM, CGM+CM, CGM+CM+Health) to test this hypothesis.

Although there is a lot of variability among participants' recall, the general trend is CGM+CM > CGM, CM, which aligns well with our hypothesis. On the other hand, the presence of Health data does not help the prediction task, and the model completely ignores them. Having the recall of a model does not necessarily provide the big picture as the model might act more sensitive toward positive observation (thus higher recall value). Therefore, it is essential to show that although the recall is growing, the model remains specific and avoids bias toward positive instances. Figure 4.3 can be helpful in this regard as it provides the model precision, which is effectively the ratio of true positives to predicted positives. Adding the growing trends in Figure 4.3 to the ones in Figure 4.2 assures that the model is getting better in detection eating moments while remaining specific.

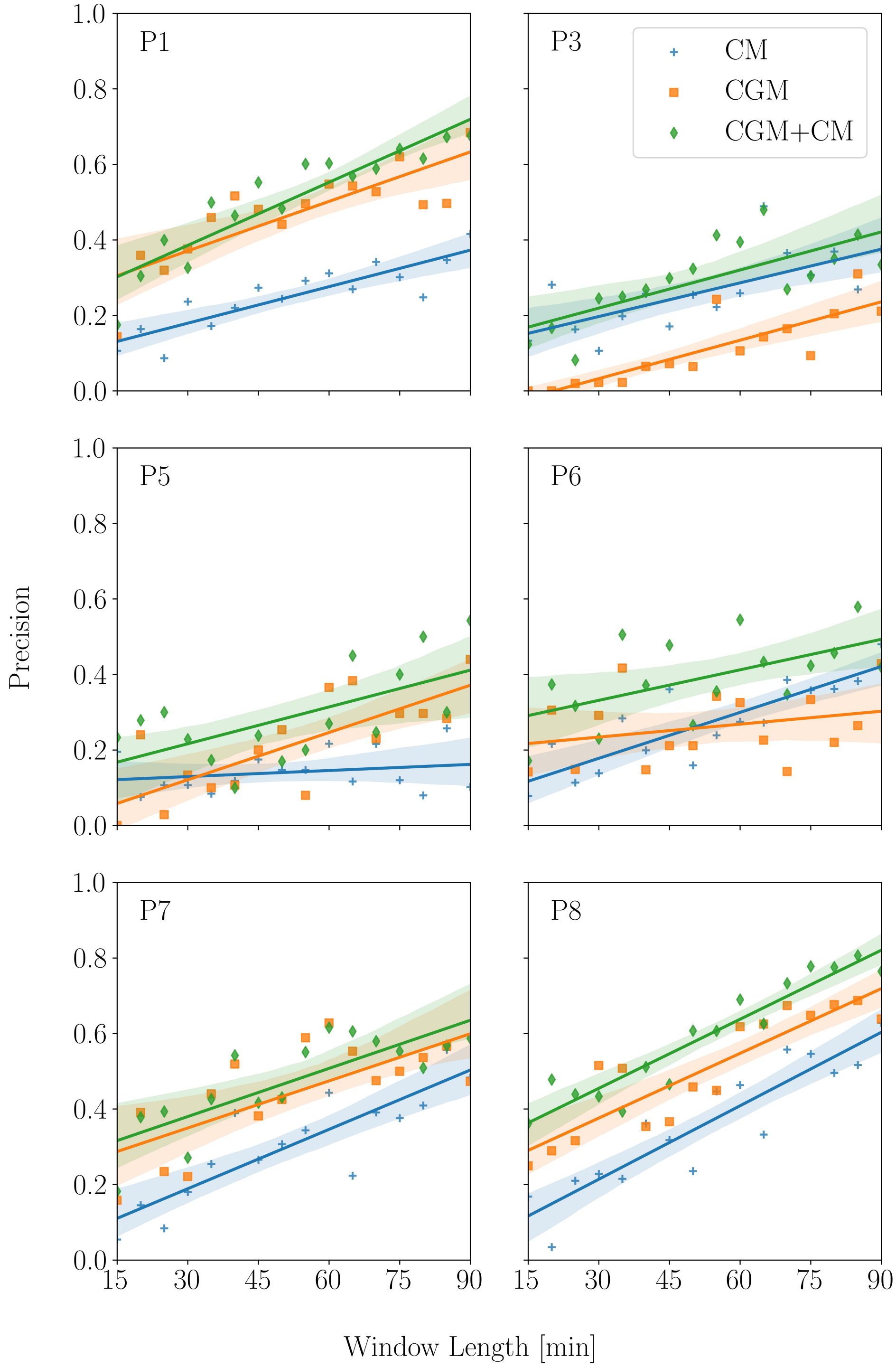


Figure 4.3: Similar to Figure 4.2 but for XG2 precision.

Additionally, there is a semi-linear improvement in recall and precision as the outer window grows for most participants. Making the outer window longer provides more retrospective information and makes the eating detection easier both for the watch and CGM sensors. Therefore, it is expected to see improvement in the detection as the window becomes bigger. This assertion holds for all participants with all combinations (CM, CGM, CM+CGM) except for P5, where the recall and precision do not follow any discernible pattern. To unmask the reason behind such an anomaly, I refer you to the number of logged meals by P5 (Table 4.1) which is almost half of the others. Further analysis suggested that there are several days for which the logs are scarce (one meal or none) which put the P5 logging into question. Such an issue is common in an uncontrolled environment as mentioned in the literature Cordeiro et al. [2015], Sajjadi et al. [2021] and is one of the main goals behind this study to remove the burden of logging from participant's shoulders. Therefore, it is legitimate to discard P5 from this analysis and claim that the bigger windows provide better results for eating detection.

4.3.5 Shortening the Eating Detection Period by Fusing CGM and Motion Data

Although the previous subsections proved the efficiency of motion augmentation with CGM, one might want to demonstrate such an enhancement quantitatively. One way to answer this quest is to measure how much time can be saved in the eating detection by fusing CGM and motion data. In another word, what is the relation between T_{CGM+CM} and T_{CGM} for a constant recall of R% where T_{CGM} and T_{CGM+CM} are the length of retro respective information for CGM and CGM+CM. To answer this question, we can benefit from the slopes and intercepts acquired from the linear regression process in Figure 4.2. Table 4.3 shows such information for the arbitrary values of $T_{CGM} = 45$ and 60 minutes for all participants, which corresponds to $\approx 40\%$ reduction of time to achieve a similar recall value by fusing CGM and motion data.

Participant	T_{CGM+CM}, 45 [min]	T_{GGM+CM}, 60 [min]
P1	37 (-8)	50 (-10)
P3	12 (-33)	23 (-37)
P5	20 (-25)	39 (-21)
P6	13 (-32)	17 (-43)
P7	42 (-3)	56 (-4)
P8	30 (-15)	44 (-15)

Table 4.3: First column denotes the participant ID while T_{CGM+CM}, 45 and $T_{CGM+CM} = 45$, 60 suggest the time by CGM+CM to reach the same recall of CGM alone at 45 and 60 minutes respectively. The parenthesized values are time reduction in minutes by fusing motion and CGM data.

4.4 Conclusion

This section explored several case scenarios with different measurement tonalities and analyzed the effects of retro respective information duration in the eating detection process. The list below summarizes the findings in this section:

- CGM data can be very informative in discerning eating moments from non-eating ones, especially for participants who are not avoiding carbohydrates in their diet.
- Hand motion can play a pivotal role in detecting eating moments while it might result in false positives or false negatives when the meals are extended due to secondary activities such as watching TV.
- CGM data can be more effective for an extended set of meals as PPGR acts robustly.
- Providing longer retrospective information enhances the recall and precision at the same time for eating detecting using hand motion and CGM.
- Fusing the motion data to CGM can increase the model's performance by removing false positives.

- Temperature, EDA, and heart rate data do not help the eating detection process.
- Adding the motion data to CGM can reduce the detection period approximately by 40% for the window of 45 and 60 minutes.

5. CONCLUSION & Future Work

In this study, we conducted a context-aware trial that contains consumed meal timing and content, performed activities and exercise timing and type, blood glucose, and hand motion time series. Additionally, two applications were developed and explained, which are crucial for the data collection and analysis process. I used a data-driven model to detect eating moments with and without context and compared their performance against each other. It was shown that adding hand motion to blood glucose data can help the model to detect eating moments faster and more accurately. On the other hand, the addition of temperature, electrodermal activity, and heart rate, categorized under health data, does not help the model performance. Although the macronutrient estimation of the detected models was not done in this study, the current platform is capable of such extension, as discussed in future work. The study highlights are listed below.

5.1 Thesis Highlights

- Recent studies have made an effort to enhance the smart health monitoring models by incorporating contextual information.

- Most of the currently available datasets with blood glucose are focused on type-I diabetic participants and suffer from the absence of contextual information (especially hand acceleration and rotation).

- To fill this gap, we designed a context-aware study and ran a pilot version with eight healthy participants where they logged their activities and meals and wore CGM and two smartwatches.

- In the absence of a robust lab-on-a-wrist platform to collect needed contextual information, I relied on the commercially typical available smartwatches for this regard. Considering the required sensors, robust data collection, capability to stream the data to the cloud, and battery life, I decided to use Apple Watch 6.

- Apple watch does not automatically save the required data; therefore, I developed a custom application to collect hand motion and gestures as well as a GUI for logging meals and activities.
- To explore the efficiency of health data, I added Empatica E4 as the second watch.
- I pre-processed the Apple watch, E4, and BG data and pipelined them for the modeling phase.
- I trained an XGBoost classifier model (XG1) to discern eating from non-eating moments on an existing dataset that contains more than 1000 meals and 354 participants.
- I successfully transferred the XG1 model to our context-aware dataset and tested its performance.
- I trained another XGBoost classifier (XG2) to identify eating moments considering blood glucose and health data and the eating probability provided by XG1 for hand motion.
- Analysis suggested that blood glucose and hand motion data are informative for eating detection among most participants.
- XG2 model suffers from low performance for participants consuming a very low amount of carbohydrate as the model relies on abrupt glucose excursion caused by carbohydrate mainly.
- Fusing the motion and CGM data helps the model remove false negatives and find more positive instances (higher recall and precision simultaneously).
- Providing more retrospective information helps the model detect eating faster and more confidently.

5.2 Future Work

Although the current model provides satisfactory results in eating detection using blood glucose and hand motion, it relies on past information to make such a decision. Such a dynamic is not enough for just-in-time prevention and interventions where the model should provide a recommendation (or possibly prompt the user) with a robust prediction about the near future. The current data collection platform allows for such an extension, but more efforts should be made to enhance the modeling module.

Detecting and segmenting the eating moments is the first step in smart dieting, where more analyses are required to backtrack the macronutrient of a detected consumed meal. Although the literature suggests accurate prediction in this regard for a controlled environment, transferring and extending this task to a free-living style has not been achieved yet. Such a transition can benefit from the developed platform in this study as it considers not only the blood glucose data for annotated meals (including the macronutrient concentration) but also contextual information such as heart rate, electrodermal activity and temperature, which can be helpful in this regard.

A. E. Caballero, A. Ceriello, A. Misra, P. Aschner, M. E. McDonnell, M. Hassanein, L. Ji, J. C. Mbanya, and V. A. Fonseca. COVID-19 in people living with diabetes: An international consensus. *Journal of Diabetes and its Complications*, 34(9):107671, sep 2020. ISSN 1056-8727. doi: 10.1016/J.JDIACOMP.2020.107671.

Roger Solis, Arash Pakbin, Ali Akbari, Bobak J Mortazavi, Roozbeh 2019 Jafari, and Roozbeh 2019 Jafari. A human-centered wearable sensing platform with intelligent automated data annotation capabilities. In *Proceedings of the International Conference on Internet of Things Design and Implementation*, pages 255–260. ACM, apr 2019. ISBN 9781450362832. doi: 10.1145/3302505.3310087. URL https://dl.acm.org/doi/abs/10.1145/3302505.3310087.

Ali Akbari, Roger Solis Castilla, Roozbeh Jafari, Bobak J. Mortazavi, Roger Solis Castilla, Roozbeh Jafari, and Bobak J. Mortazavi. Using intelligent personal annotations to improve human activity recognition for movements in natural environments. *IEEE Journal of Biomedical and Health Informatics*, 24(9):2639–2650, sep 2020. ISSN 21682208. doi: 10.1109/JBHI.2020.2966151.

Felicia Cordeiro, Elizabeth Bales, Erin Cherry, and James Fogarty. Rethinking the mobile food journal: Exploring opportunities for lightweight photo-based capture. *Conference on Human Factors in Computing Systems - Proceedings*, 2015-April:3207–3216, apr 2015. doi: 10.1145/2702123.2702154. URL http://dx.doi.org/10.1145/2702123.2702154.

Chia-Fang Chung, Elena Agapie, Jessica Schroeder, Sonali Mishra, James Fogarty, and Sean A Munson. When Personal Tracking Becomes Social: Examining the Use of Instagram for Healthy Eating. *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. doi: 10.1145/3025453. URL http://dx.doi.org/10.1145/3025453.3025747.

Seyedhooman Sajjadi, Anurag Das, Ricardo Gutierrez-Osuna, Theodora Chaspari, Projna Paromita, Laura E. Ruebush, Nicolaas E. Deutz, and Bobak J. Mortazavi. Towards The Development of Subject-Independent Inverse Metabolic Models. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 2021-June, pages 3970–3974. IEEE, 2021. doi: 10.1109/icassp39728.2021.9413829.

Yang Gao, Ning Zhang, Honghao Wang, Xiang Ding, Xu Ye, Guanling Chen, and Yu Cao. IHear Food: Eating Detection Using Commodity Bluetooth Headsets. In *Proceedings - 2016 IEEE 1st International Conference on Connected Health: Applications, Systems and Engineering Technologies, CHASE 2016*, pages 163–172. Institute of Electrical and Electronics Engineers Inc., aug 2016. ISBN 9781509009435. doi: 10.1109/CHASE.2016.14.

Luke Gemming, Aiden Doherty, Jennifer Utter, Emma Shields, and Cliona Ni Mhurchu. The use of a wearable camera to capture and categorise the environmental and social context of self-identified eating episodes. *Appetite*, 92:118–125, sep 2015. ISSN 0195-6663. doi: 10.1016/j.appet.2015.05.019.

Rui Zhang and Oliver Amft. Monitoring Chewing and Eating in Free-Living Using Smart Eyeglasses. *IEEE Journal of Biomedical and Health Informatics*, 22(1):23–32, jan 2018. ISSN 21682194. doi: 10.1109/JBHI.2017.2698523.

Yujie Dong, Jenna Scisco, Mike Wilson, Eric Muth, and Adam Hoover. Detecting Periods of Eating During Free-Living by Tracking Wrist Motion. *IEEE Journal of Biomedical and Health Informatics*, 18(4):1253–1260, 2014. ISSN 21682194. doi: 10.1109/jbhi.2013.2282471.

Surya Sharma, Phillip Jasper, Eric Muth, and Adam Hoover. The Impact of Walking and Resting on Wrist Motion for Automated Detection of Meals. *ACM Transactions on Computing for Healthcare*, 1(4):1–19, 2020. doi: 10.1145/3407623.

Odd Martin Staal, Steinar Salid, Anders Lyngvi Fougner, and Oyvind Stavdahl. Meal estimation from Continuous Glucose Monitor data using Kalman filtering and hypothesis testing. In *2019*

IEEE 58th Conference on Decision and Control (CDC). IEEE, 2019. doi: 10.1109/cdc40024.2019.9028912.

David Zeevi, Tal Korem, Niv Zmora, David Israeli, Daphna Rothschild, Adina Weinberger, Orly Ben-Yacov, Dar Lador, Tali Avnit-Sagi, Maya Lotan-Pompan, Jotham Suez, Jemal Ali Mahdi, Elad Matot, Gal Malka, Noa Kosower, Michal Rein, Gili Zilberman-Schapira, Lenka Dohnalová, Meirav Pevsner-Fischer, Rony Bikovsky, Zamir Halpern, Eran Elinav, and Eran Segal. Personalized Nutrition by Prediction of Glycemic Responses. *Cell*, 163(5):1079–1094, nov 2015. ISSN 0092-8674. doi: 10.1016/J.CELL.2015.11.001.

Projna Paromita, Theodora Chaspari, Seyedhooman Sajjadi, Anurag Das, Bobak J Mortazavi, and Ricardo Gutierrez-Osuna. Personalized Meal Classification Using Continuous Glucose Monitors. 2021. URL `http://ceur-ws.org`.

Zepeng Huo, Bobak J. Mortazavi, Theodora Chaspari, Nicolaas Deutz, Laura Ruebush, and Ricardo Gutierrez-Osuna. Predicting the meal macronutrient composition from continuous glucose monitors. In *2019 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*. IEEE, may 2019. ISBN 9781728108483. doi: 10.1109/bhi.2019.8834488.

Nathan C Hurley, Erica S Spatz, Harlan M Krumholz, Roozbeh Jafari, and Bobak J Mortazavi. A survey of challenges and opportunities in sensing and analytics for risk factors of cardiovascular disorders. *ACM Transactions on Computing for Healthcare*, 2(1):1–42, 2020.

Anurag Das, Seyedhooman Sajjadi, Bobak Mortazavi, Theodora Chaspari, Projna Paromita, Laura Ruebush, Nicolaas Deutz, and Ricardo Gutierrez-Osuna. A sparse coding approach to automatic diet monitoring with continuous glucose monitors. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2021-June:2900–2904, 2021. doi: 10.1109/ICASSP39728.2021.9414452.

Bobak Jack Mortazavi, Mohammad Pourhomayoun, Gabriel Alsheikh, Nabil Alshurafa, Sunghoon Ivan Lee, and Majid Sarrafzadeh. Determining the single best axis for exercise repetition recognition and counting on smartwatches. In *2014 11th International Conference on Wearable and Implantable Body Sensor Networks*, pages 33–38. IEEE, 2014. ISBN 1479949590.

Emma J. Cockcroft, Bert Bond, Craig A. Williams, Sam Harris, Sarah R. Jackman, Neil Armstrong, and Alan R. Barker. The effects of two weeks high-intensity interval training on fasting glucose, glucose tolerance and insulin resistance in adolescent boys: a pilot study. *BMC Sports Science, Medicine and Rehabilitation*, 11(1), dec 2019. doi: 10.1186/S13102-019-0141-9. URL `/pmc/articles/PMC6900855//pmc/articles/PMC6900855/?report=abstracthttps://www.ncbi.nlm.nih.gov/pmc/articles/PMC6900855/`.

Riddell MC, Pooni R, Yavelberg L, Li Z, Kollman C, Brown RE, Li A, and Aronson R. Reproducibility in the cardiometabolic responses to high-intensity interval exercise in adults with type 1 diabetes. *Diabetes research and clinical practice*, 148:137–143, feb 2019. ISSN 1872-8227. doi: 10.1016/J.DIABRES.2019.01.003. URL `https://pubmed.ncbi.nlm.nih.gov/30641168/`.

Michael Yang, Projna Paromita, Theodora Chaspari, Anurag Das, Seyedhooman Sajjadi, Bobak J. Mortazavi, and Ricardo Gutierrez-Osuna. A Metric Learning Approach for Personalized Meal Macronutrient Estimation from Postprandial Glucose Response Signals. pages 1–5. Institute of Electrical and Electronics Engineers (IEEE), aug 2021. doi: 10.1109/bhi50953.2021.9508528.

Silvia Oviedo, Josep Vehí, Remei Calm, and Joaquim Armengol. A review of personalized blood glucose prediction strategies for T1DM patients. *International Journal for Numerical Methods in Biomedical Engineering*, 33(6):e2833, 2017. doi: 10.1002/cnm.2833. URL `http://hdl.handle.net/10256/13497`.

Mohammad Akbari and Rumi Chunara. Using Contextual Information to Improve Blood Glucose Prediction. pages 1–17, 2019. URL `http://arxiv.org/abs/1909.01735`.

Md Fazle Rabby, Yazhou Tu, Md Imran Hossen, Insup Lee, Anthony S. Maida, and Xiali Hei. Stacked LSTM based deep recurrent neural network with kalman smoothing for blood glucose prediction. *BMC Medical Informatics and Decision Making*, 21(1):1–15, dec 2021. ISSN 14726947. doi: 10.1186/s12911-021-01462-5. URL https://doi.org/10.1186/s12911-021-01462-5https://link.springer.com/articles/10.1186/s12911-021-01462-5https://link.springer.com/article/10.1186/s12911-021-01462-5.

Wenyu Zhou, M. Reza Sailani, Kévin Contrepois, Yanjiao Zhou, Sara Ahadi, Shana R. Leopold, Martin J. Zhang, Varsha Rao, Monika Avina, Tejaswini Mishra, Jethro Johnson, Brittany Lee-McMullen, Songjie Chen, Ahmed A. Metwally, Thi Dong Binh Tran, Hoan Nguyen, Xin Zhou, Brandon Albright, Bo-Young Hong, Lauren Petersen, Eddy Bautista, Blake Hanson, Lei Chen, Daniel Spakowicz, Amir Bahmani, Denis Salins, Benjamin Leopold, Melanie Ashland, Orit Dagan-Rosenfeld, Shannon Rego, Patricia Limcaoco, Elizabeth Colbert, Candice Allister, Dalia Perelman, Colleen Craig, Eric Wei, Hassan Chaib, Daniel Hornburg, Jessilyn Dunn, Liang Liang, Sophia Miryam Schüssler-Fiorenza Rose, Kim Kukurba, Brian Piening, Hannes Rost, David Tse, Tracey McLaughlin, Erica Sodergren, George M. Weinstock, and Michael Snyder. Longitudinal multi-omics of host–microbe dynamics in prediabetes. *Nature 2019 569:7758*, 569(7758):663–671, may 2019. ISSN 1476-4687. doi: 10.1038/s41586-019-1236-x. URL https://www.nature.com/articles/s41586-019-1236-x.

Oliver Amft, Mathias Stäger, Paul Lukowicz, and Gerhard Tröster. Analysis of Chewing Sounds for Dietary Monitoring. In *Lecture Notes in Computer Science*, pages 56–72. Springer Berlin Heidelberg, 2005. ISBN 0302-9743. doi: 10.1007/11551201_4.

Sebastian Pasler and Wolf-Joachim Fischer. Food Intake Monitoring: Automated Chew Event Detection in Chewing Sounds. *IEEE Journal of Biomedical and Health Informatics*, 18(1): 278–289, 2014. doi: 10.1109/jbhi.2013.2268663.

Steve Hodges, Lyndsay Williams, Emma Berry, Shahram Izadi, James Srinivasan, Alex Butler, Gavin Smyth, Narinder Kapur, and Ken Wood. SenseCam: A Retrospective Memory Aid. In

Lecture Notes in Computer Science, pages 177–193. Springer Berlin Heidelberg, 2006. ISBN 0302-9743. doi: 10.1007/11853565_11.

Mingui Sun, Lora E Burke, Zhi-Hong Mao, Yiran Chen, Hsin-Chen Chen, Yicheng Bai, Yuecheng Li, Chengliu Li, and Wenyan Jia. eButton. In *Proceedings of the The 51st Annual Design Automation Conference on Design Automation Conference - DAC '14*. ACM Press, 2014. doi: 10.1145/2593069.2596678. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4203446.

Qianyi Huang, Wei Wang, and Qian Zhang. Your Glasses Know Your Diet: Dietary Monitoring Using Electromyography Sensors. *IEEE Internet of Things Journal*, 4(3):705–712, jun 2017. ISSN 23274662. doi: 10.1109/JIOT.2017.2656151.

Rui Zhang and Oliver Amft. Bite glasses. In *Proceedings of the 2016 ACM International Symposium on Wearable Computers*. ACM, 2016. doi: 10.1145/2971763.2971799.

Yujie Dong, Adam Hoover, Jenna Scisco, and Eric Muth. A New Method for Measuring Meal Intake in Humans via Automated Wrist Motion Tracking. *Applied Psychophysiology and Biofeedback*, 37(3):205–215, 2012. doi: 10.1007/s10484-012-9194-1. URL http://europepmc.org/articles/pmc4487660?pdf=render.

Yadnyesh Y. Luktuke and Adam Hoover. Segmentation and Recognition of Eating Gestures from Wrist Motion using Deep Learning. *Proceedings - 2020 IEEE International Conference on Big Data, Big Data 2020*, pages 1368–1373, 2020. doi: 10.1109/BigData50022.2020.9378382.

Bobak J. Mortazavi and Ricardo Gutierrez-Osuna. A Review of Digital Innovations for Diet Monitoring and Precision Nutrition. *Journal of Diabetes Science and Technology*, 2021. doi: 10.1177/19322968211041356. URL https://doi.org/10.1177/19322968211041356.

Yujie Dong, Adam Hoover, and Eric Muth. A device for detecting and counting bites of food taken by a person during eating. In *2009 IEEE International Conference on Bioinformatics and*

Biomedicine, BIBM 2009, pages 265–268, 2009. ISBN 9780769538853. doi: 10.1109/BIBM.2009.29.

Cindy Marling and Razvan Bunescu. The OhioT1DM dataset for blood glucose level prediction: Update 2020. In *CEUR workshop proceedings*, volume 2675, page 71. NIH Public Access, 2020.

Fabien Dubosson, Jean-Eudes Ranvier, Stefano Bromuri, Jean-Paul Calbimonte, Juan Ruiz, and Michael Schumacher. The open D1NAMO dataset: A multi-modal dataset for research on non-invasive type 1 diabetes management. *Informatics in Medicine Unlocked*, 13:92–100, 2018. doi: 10.1016/j.imu.2018.09.003. URL `https://research.ou.nl/en/publications/77d64e49-34f7-4831-9e58-99a434c744f5`.

John Martinsson, Alexander Schliep, Björn Eliasson, and Olof Mogren. Blood Glucose Prediction with Variance Estimation Using Recurrent Neural Networks. *Journal of Healthcare Informatics Research*, 4(1):1–18, 2020. doi: 10.1007/s41666-019-00059-y.

Daniel Rhyner, Hannah Loher, Joachim Dehais, Marios Anthimopoulos, Sergey Shevchik, Ransford Henry Botwey, David Duke, Christoph Stettler, Peter Diem, and Stavroula Mougiakakou. Carbohydrate Estimation by a Mobile Phone-Based System Versus Self-Estimations of Individuals With Type 1 Diabetes Mellitus: A Comparative Study. *Journal of medical Internet research*, 18(5), may 2016. ISSN 1438-8871. doi: 10.2196/JMIR.5567. URL `https://pubmed.ncbi.nlm.nih.gov/27170498/`.

Lauriane Bertrand, Nathan Cleyet-Marrel, and Zilu Liang. The role of continuous glucose monitoring in automatic detection of eating activities. *LifeTech 2021 - 2021 IEEE 3rd Global Conference on Life Sciences and Technologies*, pages 313–314, mar 2021. doi: 10.1109/LIFETECH52111.2021.9391849.

Tejaswini Mishra, Meng Wang, Ahmed A. Metwally, Gireesh K. Bogu, Andrew W. Brooks, Amir Bahmani, Arash Alavi, Alessandra Celli, Emily Higgs, Orit Dagan-Rosenfeld, Bethany Fay, Susan Kirkpatrick, Ryan Kellogg, Michelle Gibson, Tao Wang, Erika M. Hunting, Petra Mamic, Ariel B. Ganz, Benjamin Rolnik, Xiao Li, and Michael P. Snyder. Pre-symptomatic detection of COVID-19 from smartwatch data. *Nature biomedical engineering*, 4(12):1208–1220, nov 2020. ISSN 2157-846X. doi: 10.1038/s41551-020-00640-6. URL `https://www.nature.com/articles/s41551-020-00640-6`.

Bryan David Minor, Janardhan Rao Doppa, and Diane J. Cook. Learning activity predictors from sensor data: Algorithms, evaluation, and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12):2744–2757, dec 2017. doi: 10.1109/TKDE.2017.2750669.

MortazaviBobak J., PourhomayounMohammad, LeeSunghoon Ivan, NyamathiSuneil, WuBrandon, and SarrafzadehMajid. User-optimized activity recognition for exergaming. *Pervasive and Mobile Computing*, 26:3–16, feb 2016. doi: 10.1016/J.PMCJ.2015.11.001. URL `https://dl.acm.org/doi/abs/10.1016/j.pmcj.2015.11.001`.