UNIVERSITY OF CALIFORNIA SAN DIEGO

Using Wearables and Machine Learning to Enable Personalized Lifestyle Recommendations to Improve Blood Pressure

A Dissertation submitted in partial satisfaction of the requirements
for the degree
Doctor of Philosophy

in

Electrical Engineering (Computer Engineering)

by

Po-Han Chiang

Committee in charge:

Professor Sujit Dey, Chair
Professor Jyoti Mishra
Professor Truong Quang Nguyen
Professor Ramesh R Rao

2022

The Dissertation of Po-Han Chiang is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2022

TABLE OF CONTENTS

Dissertation Approval Page iii

Table of Contents iv

List of Figures v

List of Tables vi

Acknowledgements vii

Vita ix

Abstract of the Dissertation xi

Introduction 1

Chapter 1 Offline and Online Learning Techniques for Personalized Blood Pressure Prediction and Health Behavior Recommendations 4

Chapter 2 Using Wearables and Machine Learning to Enable Personalized Lifestyle Recommendations to Improve Blood Pressure 27

Chapter 3 Optimal Use of Harvested Solar, Hybrid Storage and Base Station Resources for Green Cellular Networks 57

Conclusion 90

Bibliography 92

LIST OF FIGURES

Figure 1.1: Batch learning and online learning. 5

Figure 1.2: Block diagram of proposed personalized BP model using RFFS. 13

Figure 1.3: Different mechanisms of online learning. 14

Figure 1.4: MAE of SBP on different users. 20

Figure 1.5: Daily blood pressure of two users. 23

Figure 1.6: Time-averaged of MAE of SBP and DBP on different users. 25

Figure 2.1: Block Diagram of data storage and access. 34

Figure 2.2: System architecture of the proposed method. 34

Figure 2.3: Illustration of feature extraction and statistics. 36

Figure 2.4: Prediction error with different ratios of selected features. 37

Figure 2.5: SHAP values of features on two users. 48

Figure 2.6: Box plot of SHAP values of lifestyle features. 49

Figure 2.7: Pearson Correlation Heatmap on two users. 50

Figure 2.8: Example of personalized recommendation of subject 1. 51

Figure 3.1: Architecture of proposed system. 62

Figure 3.2: Battery level and buffer level of two users versus time. 83

Figure 3.3: Performance with uniformly distributed solar trace. 84

Figure 3.4: Performance with actual solar trace. 85

Figure 3.5: Required battery level and average buffer vs. V 86

Figure 3.6: Average grid power and aggregate utility consumption vs. $\frac{w_I}{w_G}$ 87

LIST OF TABLES

Table 1.1: Features and Target Variables ……..…….……...10

Table 1.2: MAE, RMSE, and MAPE of different models (offline learning) …….....……..20

Table 1.3: RMSE and MAE models using personal and aggregated dataset …..……..21

Table 1.4: Top features and the weekly change of top 3 features and the corresponding BP level ……..…….…..…22

Table 1.5: Time-averaged MAE, RMSE, and MAPE of different online learning models.....24

Table 2.1: Cohort Statistics ……...…….……..33

Table 2.2: Features and Target Variables …….…….……..36

Table 2.3: Prediction Error of RFSV and related models …..……..45

Table 2.4: MAE in different prediction horizons using RF, RF-ARIMA and RFSV MAE …46

Table 2.5: MAEs using different feature selection methods ……47

Table 2.6 Recommendations and BP of subjects in the experimental group before and after their personalized recommendation……...…52

Table 2.7: Summary of the BP changes in the experimental group and the control group … …...……53

Table 3.1: Summary of key notations ……..……63

Table 3.2: Simulation Parameters ……81

ACKNOWLEDGEMENTS

Foremost, I would like to express my sincere gratitude to my advisor Prof. Sujit Dey for his continuous support of my Ph.D. study and research. Under his careful supervision and guidance, I have learned a lot in scientific research and improved my writing and presentation skills in a systematic way.

Besides my advisor, I would like to thank the rest of my thesis committee Prof. Truong Quang Nguyen, Prof. Ramesh R Rao, and Prof. Jyoti Mishra Ramanathan, for their insightful comments and suggestions. I would also like to thank the collaborators who worked closely with me on this project: MD. Melissa Wong, Dr. Ranjini B. Guruprasad, and Mr. Jared J. Leitner. They not only enlightened me with academic knowledge but also gave me valuable advice whenever I needed it the most. I am also grateful to every member of the MESDAT lab at UCSD for giving me an exciting and productive experience.

Finally, I'd like to dedicate my work to my parents and sister who have been a constant source of support. Without their unconditional love and encouragement, this thesis paper cannot be accomplished. At the same time, my appreciation also goes to my girlfriend, who always accompany and inspire me with her fullest and truest heart.

Chapter 1, in part, is from the material as it appears in proceedings of IEEE conference on Healthcom 2018. Po-Han Chiang, Melissa Wong and Sujit Dey, and in IEEE Access 2019. Po-Han Chiang and Sujit Dey. The dissertation author was the primary investigator and author of the papers.

Chapter 2, in part, is from the material as it appears in proceedings of IEEE Journal of Translational Engineering in Health and Medicine 2021, Po-Han Chiang, Melissa Wong and Sujit Dey. The dissertation author was the primary investigator and author of the papers.

Chapter 3, in part, is from the material as it appears in proceedings of IEEE International Symposium on Personal, Indoor and Mobile Radio Communications 2015, Po-Han Chiang, Ranjini B. Guruprasad and Sujit Dey. and in IEEE Transactions on Green Communications and Networking 2018, Po-Han Chiang, Ranjini B. Guruprasad and Sujit Dey. The dissertation author was the primary investigator and author of the papers.

VITA

2010 Bachelor of Engineering, National Taiwan University

2013 Master of Science, National Taiwan University

2014-2022 Research Assistant, University of California San Diego

2022 Doctor of Philosophy, University of California San Diego

Publication

[1] P. Chiang, M. Wong and S. Dey. "Using Wearables and Machine Learning to Enable Personalized Lifestyle Recommendations to Improve Blood Pressure." *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 9, pp. 1-13, 2021

[2] J. J. Leitner, P. Chiang and S. Dey, "Personalized Blood Pressure Estimation Using Photoplethysmography: A Transfer Learning Approach," accepted in *IEEE Journal of Biomedical and Health Informatics*, 2021

[3] Y. Ku, P. Chiang and S. Dey, "Real-Time QoS Optimization for Vehicular Edge Computing with Off-Grid Roadside Units," in *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, pp. 11975-11991, Oct. 2020

[4] P. Chiang and S. Dey, "Offline and Online Learning Techniques for Personalized Blood Pressure Prediction and Health Behavior Recommendations," *in IEEE Access*, vol. 7, pp. 130854-130864, 2019

[5] J. Leitner, P. Chiang and S. Dey, "Personalized Blood Pressure Estimation using Photoplethysmography and Wavelet Decomposition," in Proc. of *IEEE Healthcom*, Bogotá, Colombia, 2019.

[6] Y. Ku, P. Chiang and S. Dey, “Quality of Service Optimization for Vehicular Edge Computing with Solar-Powered Road-Side Units,” in Proc. of *IEEE ICCCN*, Hangzhou, China, 2018.

[7] P. Chiang, R. Guruprasad and S. Dey, "Optimal Use of Harvested Solar, Hybrid Storage and Base Station Resources for Green Cellular Networks*," IEEE Transactions on Green Communications and Networking*, vol. 2, no. 3, pp. 707-720, Sept. 2018

[8] P. Chiang and S. Dey, "Personalized Effect of Health Behavior on Blood Pressure: Machine Learning Based Prediction and Recommendation," in Proc. of *IEEE Healthcom*, Ostrava, Czech, 2018

[9] P. Chiang, S. Chiluvuri, S. Dey and T. Nguyen, "Forecasting of Solar Photovoltaic System Power Generation using Wavelet Decomposition and Bias-compensated Random Forest," in Proc. of *IEEE GreenTech*, Denver, CO, 2017,

[10] P. Chiang, R. Guruprasad and S. Dey, "Renewable Energy-Aware Video Download in Cellular Networks," in Proc. of *IEEE PIMRC*, Hong Kong, 2015

[11] P. Chiang, P. -H. Huang, S. -S. Sun, W. Liao and W. -T. Chen, "Joint power control and user association for traffic offloading in heterogeneous networks," in Proc. of *IEEE Global Communications Conference*, Austin, TX, 2014

[12] P. Huang, P. Chiang and W. Liao, "Coverage and capacity aware cell scaling in green cellular networks," in Proc. of *IEEE Global Communications Conference,* 2013.

ABSTRACT OF THE DISSERTATION

Using Wearables and Machine Learning to Enable Personalized Lifestyle Recommendations to Improve Blood Pressure

by

Po-Han Chiang

Doctor of Philosophy in Electrical Engineering (Computer Engineering)

University of California San Diego, 2022

Professor Sujit Dey, Chair

Hypertension, or high blood pressure (BP), one of the most prevalent chronic diseases, affects 30% of American adults and contributes to over 410,000 deaths per year. Lifestyle factors such as sleep and exercise are proved to be highly correlated with BP. However, the individual effect of lifestyle, that is which of the lifestyle factors has most important effect on an individual's BP level, has not yet been studied. Using physiological and lifestyle data

collected by wearables, our goal is to investigate the relationships between BP and lifestyle factors and provide personalized and precise recommendations to improve BP, as opposed to the current practice of general lifestyle recommendations.

Firstly, we extract necessary and interpretable features from raw lifestyle data collected by wearables. To utilize temporal information from the BP series, we propose to extract new features based on ARIMA to enhance the accuracy of BP prediction. We propose a machine learning method to explore the personalized relationship between BP and lifestyle factors. The proposed system provides BP prediction as well as lifestyle recommendations. Furthermore, since BP and lifestyle data are collected and learned sequentially, the performance of prediction is prone to the existence of concept drifts and anomaly points. To solve this problem, we propose an Online Weighted-Resampling (OWR) technique to enhance RFFS in an online learning scenario. Thirdly, we propose a feature selection method using feature importance derived from Shapley Value, in order to remove redundant and irrelevant features and provide the personalized insight that may affect an individual's BP. We evaluate and show that our proposed technique outperforms other popular machine learning methods in terms of prediction error. We also show the effectiveness of personalized recommendations using a randomized controlled trial. Our research demonstrates prospects for reducing BP through precise lifestyle changes, either effectuated through personalized interventions by clinicians

In the final section, my earlier work in green communication, which utilizes renewable energy to reduce grid energy consumption of base station (BSs), is presented. We propose to utilize data buffer of user equipment as well as energy storage at the BS to better adapt the BS resource allocation and hence its energy consumption to the dynamic nature of RE. A low-complexity online control scheme based on Lyapunov optimization framework is presented.

Introduction

Hypertension, or high blood pressure (BP), is one of the most prevalent chronic diseases in the world, affecting 30% of American adults and contributing to over 410,000 deaths per year. Moreover, BP is an essential indicator for human health and is known to be greatly influenced by lifestyle factors, like activity and sleep. However, the degree of impact of each lifestyle factor on BP is unknown and may vary between individuals. In this dissertation, we propose to elucidate the complex relationships between BP and lifestyle factors at the level of the individual. Based on the continuous data collected from wearables of users, we aim to 1) provide a prediction of BP, which will give users a quick and reliable way to understand their health condition, and 2) provide personalized and actionable insight to users in order to control their BP by adjusting their lifestyle accordingly.

Machine learning (ML) has been widely used in various healthcare applications to provide better information to doctors at the point of patient care. ML models can be trained to analyze images, recognizing patterns, and identify abnormalities, thus improving the accuracy and efficiency of all these processes. On the other hand, continuous, large-scale, and high-quality data collected by wearables brings great potential for early diagnosis and prevention. However, the potential of wearables for medical studies has not been utilized due to the lack of connection to health condition. Using ML techniques as well as various physiological and lifestyle data through wearables, we can achieve one step closer to understanding proactive and personalized healthcare.

Note that in this dissertation, we use the terms “lifestyle” and “health behavior” interchangeably. In Chapter 1, we propose a personalized model to predict an individual’s daily BP and estimate the effect of the individual’s health behavior on his/her BP. The personalized BP

prediction model based on health behavior provides users timely insight about their BP. Moreover, its interpretable and personalized insight helps individuals and their healthcare providers improve BP through lifestyle changes. We showed that providing personalized insight to a user as to what are the most important health behavior factors that affect his/her BP can potentially positively influence the user to make lifestyle changes which in turn can positively impact the user's BP levels. Our proposed personalized modeling technique works well when sufficient training data samples have been already collected for individuals. However, for a new user, this means significant time lapse before the user's training data is collected, and the model and recommendations can be used reliably. Moreover, due to various external factors that cannot be captured by the proposed BP model (e.g., medication and measurement error), non-stationary characteristics of the data stream may deteriorate the prediction. To address the above challenges, we propose an online learning technique, which updates the model incrementally given the data stream. One can therefore improves the model gradually with more data while addressing evolving statistical distribution of data.

In Chapter 2, we examine the effect of varying degrees of intervention on BP control through a randomize controlled trial with real hypertensive patients. Our proposed system consists of automated data collection using home BP monitors and wearable activity trackers and feature engineering techniques to address time-series data and enhance interpretability. We propose Random Forest with Shapley-Value-based Feature Selection to offer personalized BP modeling and top lifestyle factor identification, and subsequent generation of precise recommendations based on the top factors. Our study results validate our system's ability to provide accurate personalized BP models and identify the top features which can vary greatly between individuals.

We also validate the effectiveness of personalized recommendations in a randomized controlled experiment.

In Chapter 3, we present the previous research in green communication, which utilizes renewable energy to reduce grid energy consumption of base station (BSs). We consider the scenario of downlink Orthogonal Frequency Division Multiple Access (OFDMA) networks with non-ideal hybrid energy supply (HES). To jointly optimize the energy consumption and the quality of service (QoS) of users, we adopt the weighted sum of users' utility of data rates and grid energy consumption as our performance metric. We propose a low-complexity online control scheme based on Lyapunov optimization framework. The proposed technique can provide asymptotically optimal performance bound without requiring the stochastic distribution information of RE arrival and channel state condition. The experimental results demonstrate the ability of the proposed approach to significantly improve the performance in terms of grid power consumption and user QoS compared with the existing schemes.

Chapter 1

Offline and Online Learning Techniques for Personalized Blood Pressure Prediction and Health Behavior Recommendations

1.1 Introduction

Hypertension, or high blood pressure (BP), one of the most prevalent chronic disease in the world, affects 30% of American adults and contributes to over 410,000 deaths per year [1]. Health behavior (e.g., exercise, nutrition and sleep) has been widely acknowledged to have a large influence on BP [2,3]. However, the relationship between health behavior and BP is only studied through clinical trials in ambulatory settings, whose scope is limited in terms of trial population and duration. On the other hand, wearables such as Fitbit, Apple Watch, and Samsung Galaxy Watch collect a large amount of high-granularity health behavior data such as the duration and quality of activities and sleep. However, the potential of wearables for early diagnosis and management of hypertension has not been fully utilized due to the lack of quantitative connection between health behavior and BP.

The primary metrics used to measure BP are systolic (SBP) and diastolic blood pressure (DBP), which are defined as the maximum and minimum blood pressure, respectively, during a pulse. They are measured in millimeters of mercury (mmHg). For accurate diagnosis and treatment of hypertension, constant BP measurement is necessary, which is difficult today outside of ambulatory care environment. Traditional cuff-based BP measurements require significant user effort [4], which limits the usability and increases the chance of measurement error. Although there

has been great attention to automatic and continuous BP estimation using electrocardiography and/or photoplethysmography (PPG) signal [5], the accuracy and cost limit the viability of such methods.

On the other hand, the prediction of BP based on historical BP and health behavior provides users an efficient way to manage their BP. Sleep and exercise are shown statistically correlated with BP in randomized controlled trials [2, 3]. However, what remains lacking in the literature is the effect of these behaviors on the individual's BP. Our study aims to provide personalize and actionable insight to users in order to improve their BP by adjusting their health behavior accordingly.

Random forest (RF) [6] is one of the most popular machine learning methods. RF averages multiple decision trees and works well on both classification and regression tasks [7]. Moreover, RF provides measures of the relative importance of the features (predictor variables) as one of its intrinsic properties. The importance score is shown as an efficient tool for identifying relevant variables [8]. Therefore, we choose RF as our candidate ML method. Like other ML methods, RF-based multivariate time series prediction faces a major challenge, that is, how to capture and leverage the dynamic dependencies among multiple variables. To better capture the temporal information in BP time series, we extract time series-related features of BP. Finally, utilizing feature importance obtained from RF, we propose a stable and consistent feature selection

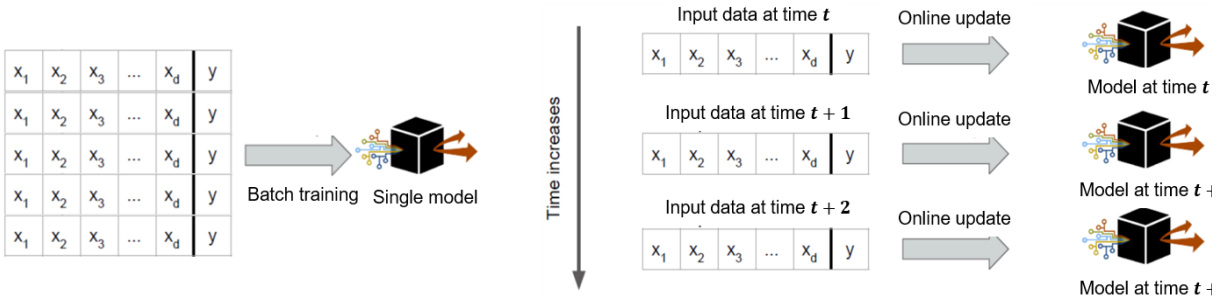

Figure 1.1 (a) left, Batch learning (b) right, Online learning

technique, namely RF with Feature Selection (RFFS), to remove redundant and/or irrelevant features and enhance prediction accuracy.

In our preliminary work [9], we proposed a personalized model using RFFS to predict daily BP using historical BP and health behavior and identify the most important health behavior to his/her BP prediction. We have shown that RFFS works well when sufficient data examples are already collected for individuals. However, there are two main limitations to be addressed: 1) Since BP and health behavior data are collected and used for training sequentially, it takes significant time for new users to collect training data in order to achieve satisfying performance, which limits the usability. 2) Due to various external factors that cannot be captured by the proposed BP model (e.g., diet, stress and measurement error), non-stationary characteristics in BP and health behavior time series may deteriorate the prediction. To extend RFFS with sequential and limited data examples, we propose a novel Online Weighted-Resampling (OWR) technique to alleviate the negative effect of concept drifts and anomaly points and thereby decrease the necessary time for data collection. OWR creates adaptive training dataset through the resampling process of RF when new examples arrive. The focus of our proposed OWR will be: 1) how to adjust the sample weights based on prediction performance of the current model and 2) how to update the RF model by resampling given the weights calculated in 1) in a way that examples with higher weights are more likely to be selected to train future models, and vice versa. The proposed approach is not only applicable to BP prediction but all applications in online learning scenarios.

The rest of the paper is organized as follows. In Section 1.2, we will introduce related work of BP prediction and online learning. In Section 1.3, data collection and representation, and time series feature extraction are presented. We then detail the proposed RF-based feature selection and online weighted-resampling technique. In Section 1.4, the performance of the proposed method is

evaluated. We also provide early validation of the effects of personalized health behavior recommendations suggested by the proposed RFFS model. Finally, we conclude the paper in Section 1.5.

1.2 Related Work

Previous work focused on blood pressure prediction is described in [10-16]. The authors in [10] use contextual data (e.g., age, weight and smoke habit) and the authors in [11] use heart rate and steps collected by wearables to diagnose hypertension. The above techniques provide an alternative way to diagnose hypertension, but they cannot provide a numerical prediction of BP. The authors in [12] predict SBP using demographic and contextual data using an artificial neural network (ANN). However, the prediction is based on a single BP measurement and does not consider the dynamics of BP. In [13,14], PPG signals are used to predict short-term BP with ensemble trees models [13] and ANN models [14]. However, PPG-based prediction is only applicable for a very short time horizon (~10 minutes) while our technique aims to predict BP one day ahead, which provides timely and actionable information to users. In [15], the 24-hour time series of BP and heart rate are trained with Extreme Learning Machine (ELM) to provide hourly BP prediction. However, the sample size in [15] is limited to a single day, and the prediction performance is only compared with other ELM variants. The authors in [16] propose to solve the temporal dependency between BP and contextual data by using Long Short-Term Memory (LSTM) models. Since the data in [16] is measured daily and then averaged on a monthly basis, the granularity of temporal dependency is therefore not fully utilized due to insufficient temporal resolution and information loss in the averaging process. None of the above studies uses exercise and sleep data collected by wearable devices to predict BP. Moreover, the above studies assume

offline learning scenario where sufficient data is always available, which may not be the case. In this work, both offline and online learning scenarios are considered.

One major challenge in BP prediction and other healthcare applications is the sequentially collected data stream may contain 1) anomaly points which may skew and mislead the training process, and 2) concept drifts where the statistical distribution of data evolves [17]. As training data accumulates, batch-training ML models may not adapt to concept drifts since a model trained on the old distribution of examples is not guaranteed to fit the new distribution of examples [18]. A widely adopted solution is windowing, which provides a simple forgetting mechanism by removing obsolete examples from the training set [19]. However, how to choose the optimal size of the window for given data poses a new problem (e.g., smaller window sizes are more responsive to concept drifts with the disadvantage of insufficient training data and vice versa). Another online learning method is to equip ML models with drift detectors [18]. Drift detectors track and monitor the performance of ML models or data distribution. When concept drift is detected, ML models will be rebuilt [20] or updated [21]. The drawback of this method is that the performance of such methods largely depends on the accuracy of drift detectors. As a result, the model cannot fully adapt to the dynamics of the evolving distribution of examples if the drift detector is not triggered.

Without explicit drift detection, the authors in [22] propose to use new decision trees as the base learners in RF, namely Mondrian trees. Since new split can be introduced anywhere in addition to leaf nodes in the trees, Mondrian trees support can support online learning. Online Stochastic Gradient Descent (SGD), proposed by [23], implements regularized linear models with SGD. The gradient of the loss is estimated sequentially and used to update the model with a decreasing learning rate. However, those incremental learning methods cannot respond to abrupt changes and anomaly points.

Finally, online ensemble learning methods [24] train new learners whenever a new example [25] or sequential blocks of the data streams [26] arrives instead of training all learners at a time. The prediction is the weighted average of the outputs generated by all learners. The learners' weights are then updated with respect to their accuracy. Similar to the boosting technique [27], the weights of previous training examples are updated based on the accuracy of incoming examples. The weights are then used to fit the new learner [28]. However, such a technique requires that the base learner can handle the sample weight information. In this work, OWR proposes to provide a dynamic resampling mechanism of previous examples to utilize the adaptive sample weight. OWR can be applied to any base learner in ensemble learning since it creates an adaptive training dataset with the same size as the original training dataset.

1.3 Proposed Method

In this section, we will first describe the collected data and the proposed feature extraction for time series prediction. We then introduce the proposed RF-based Feature Selection (RFFS) and Online Weighted-Resampling (OWR) methods.

1.3.1 Data Acquisition and Representation

The dataset used in this study is collected from self-tracking experiment on 8 participants for 90 days. The study is approved by the university's Institutional Review Board. The users (subjects) include four males and four females, and their age ranges from 25 to 79. Users who consented were then sent a Fitbit Charge HR and an Omron Evolv wireless BP monitor to collect their health behavior and BP. Since sleep and exercise data from Fitbit is measured in every minute

TABLE 1.1. Features and Target Variables

Target Variables	Original Features	Derived Features
SBP, DBP	Heart rate, calories burned, steps, distance, floors climbed, sedentary minutes, lightly active minutes, fairly active minutes, very active minutes, exercise calories, minutes asleep, minutes awake, awakening times, bedtime, wake up time	Historical BP ($BP_{t-1}, BP_{t-2} \dots, BP_{t-k}$), $EWMA_{BP}$, days (categorical), morning (categorical)

while BP and heart rate are measured twice (8-10 am and 7-9 pm) daily, the dataset consists of time series with different frequencies. To model the features and the target variables (SBP and DBP), we aggregate the health behavior data on a daily basis.

Our objective is to predict BP level one day ahead using one's past BP readings as well as exercise, sleep and heart rate on the previous day. In addition to steps count and floors climbed, we also include advanced fitness data, including calories burned and different levels of active time (sedentary, lightly active, fairly active, very active). The data is obtained from Fitbit based on heart rate and the metabolic equivalent [29], an intensity measure expressing the energy expenditure during various physical activities. Sleep data includes minutes asleep, minutes awake, awakening times, bedtime, and wake up time. Bedtime and wake up time represent the time users go to sleep and wake up respectively. Each example consists of one measurement of SBP and DBP in the morning or the evening. For data pre-processing, the examples with missing BP are removed, and the examples with missing exercise or sleep data are imputed using K Nearest Neighbor (KNN) imputation [30]. After filtering, there were at least 52 effective days for each user in the dataset, and totally 1326 examples are used.

In supervised learning, the dataset consists of observations (examples) of the target variable y and the respective features vector X. It is assumed that the examples are independent and identically distributed, which does not apply to time series. Time series prediction problems include a set of time-ordered observations $s_t = (X_t, y_t),\ t = 1,2 \dots T$ where X_t are the values of

X and y_t is the value of y observed at time t, and the task is defined as to predict the future values of y_u for time $u > t$ given $s_1, s_2 \ldots, s_t$. In our problem, we set $u = t + 1$. In addition to using X_t as features, time series features can be extracted from $y_1, y_2 \ldots, y_t$ to capture the temporal dependency of y, which is the BP time series in this study. The simplest features extracted from y are recent observations of y (e.g. $y_t, y_{t-1} \ldots, y_{t-T}$), which assumes that correlation exists between successive observations of the series. If T is properly chosen, such features can capture the underlying structure of the time series. Additionally, we use exponentially weighted moving average (EWMA) to capture the trend of BP time series. Two additional features, namely Day and Morning, are created to capture the daily and weekly periodicity of BP time series. We use one-hot encoding to perform binarization of the categorical features. The original and derived features are summarized in Table 1.1.

1.3.2 RF-based Feature Selection (RFFS)

Preliminaries: regression decision tree

Classification and Regression Tree (CART) [31] is a non-parametric model used in machine learning problems. CART is built by learning simple decision rules inferred from the training examples. CART consists of two types of nodes: 1) internal nodes, which use one of the features and the corresponding threshold to split the examples into binary branches, 2) leaf (terminal) nodes, where no more split is performed. If the target variable is continuous, the prediction of the target variable is the average of all training examples at that node. In the training phase, the topmost internal node (root node) contains all training examples. At each internal node, the feature and its split threshold are selected greedily to minimize the mean squared error of the prediction. In the prediction phase, the new example moves down from the root node to one of the

leaf nodes based on the splitting criteria along its path. The predicted value is then the average training example at that leaf node.

RF is an enhanced approach by aggregating a collection of decision trees in order to reduce overfitting of the data and the resulting high variance. RF contains two major components: bootstrap aggregation and features bagging. RF produces bootstrap datasets that are randomly and independently drawn with replacement from the training dataset. Each bootstrap dataset is the same size as the original training set. Bootstrap aggregation, or namely bagging, averages the prediction of decision trees trained from different bootstrap datasets [32]. In addition to bagging, RF introduces feature bagging, which randomly selects a subset of features when constructing each tree. Finally, because of its non-parametric property, RF generalizes well with both categorical and numerical features, with minimal parameter tuning required.

There may be redundant and irrelevant features in the proposed dataset either from original features (e.g., sleep and calorie burned) or derived features (e.g., EWMA). High-dimensional data will degrade the performance especially when the number of labeled examples is relatively small. Due to the non-linear relationship between BP and features as well as varying extents of correlation, it is challenging to select useful features with preset rules. Feature importance, computed by measuring how selected features reduce prediction error when building decision trees in RF, is often used to rank and select features. However, the feature importance generated by a single run of the RF model is usually inconsistent due to the stochastic nature of bootstrap aggregation and feature bagging. To obtain a consistent ranking of features, [33] trains multiple RF models and averages the feature importance generated by those RF models. Therefore, we propose a RF-based feature selection technique which uses an additional model consisting of

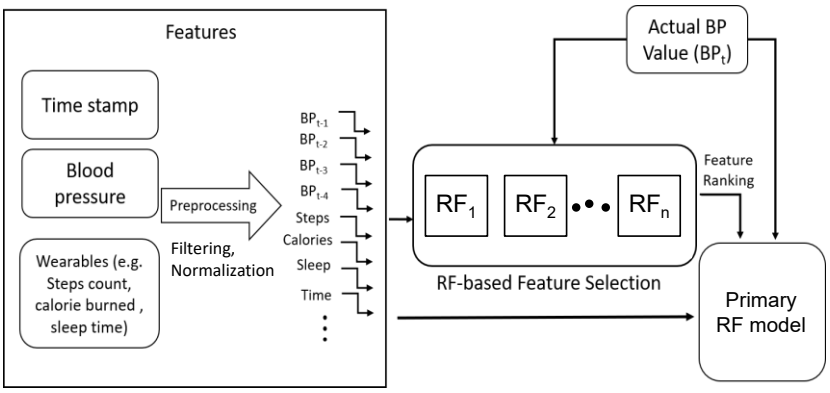

Figure 1.2. Block Diagram of proposed personalized BP model using RFFS

multiple independent RFs to rank and select important features. We define the feature importance vector in j^{th} run of RF as $I_{Xy}(j)$ where $j = 1,2 \ldots J$. The average feature importance $\bar{I}_{Xy}$ can then be calculated by

$$\bar{I}_{Xy} = \frac{\sum_{j=1}^{J} I_{Xy}(j)}{J} \ . \tag{1}$$

Only the features with higher feature importance in $\bar{\boldsymbol{I}}_{\boldsymbol{Xy}}$ will be selected to train the primary RF model. The optimal number of RFs and the threshold to select features may vary with different datasets and is out of the scope of this paper. Here we use the median value in $\bar{\boldsymbol{I}}_{\boldsymbol{Xy}}$ as the selection threshold. In our proposed model, we use 5 RFs with 100 trees in our feature selection model since any larger number of RFs and trees results in the same ranking of features. The block diagram of our proposed BP model integrated with RFFS is shown in Fig. 1.2.

Although RFFS can be applied to any kind of ML problems, it is designed to handle static dataset and batch-training. As motivated earlier in Section 2.1 simply applying RFFS in an online learning scenario may not solve the negative effect of concept drifts and anomaly points. Based on RFFS, we propose the Online Weighted-Resampling (OWR) method in the next section.

1.3.3 Online Weighted-Resampling (OWR) Methodology

We first define the online learning problem as one that examples arrive sequentially in the form of a data stream $s_1, s_2, \dots, s_t, s_t=(X_t, y_t)$. A learning algorithm creates a sequence of models $h_1, h_2 \dots, h_t$. h_t is a model trained with historical examples $(s_1, s_2, \dots, s_{t-1})$ and the previous model h_{t-1}. For new coming example X_t, h_t give a prediction of the target variable as $\hat{y}_t$. It is assumed that after some delay, the actual value of y_t is provided. The learning algorithm can update the model by comparing $\hat{y}_t$ and y_t. Example s_t then becomes a part of the training data and the above process is repeated.

The rationale of OWR is that previous examples have nonuniform contributions to predicting the target variable. Possible reasons for such nonuniformity are: 1) a single anomaly example behaving abnormally and 2) concept drifts that a continuous sequence of examples having a different statistical distribution from previous examples. Therefore, the objective of OWR is to provide a dynamic resampling mechanism of all previous examples while keeping all the advantages of RF. Compared to traditional online learning, and sliding-window based online learning, OWR can create training datasets by assigning different weights to previous examples. The concept of OWR is shown in Fig 1.3. The different shades of the green on previous training examples represent their importance in future models. The darker the shade, the more important

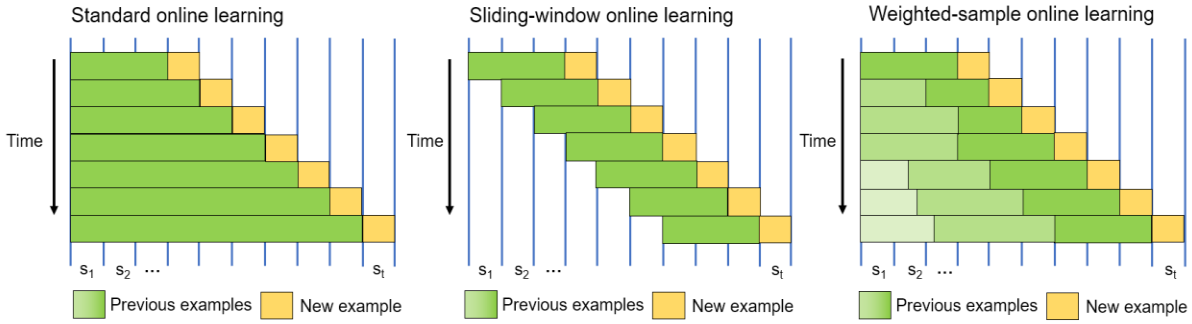

Figure 1.3. (a) left, standard online learning (b) center, sliding-window online learning (c) right, weighted-resampling online learning

the example in the current model. Unlike standard online learning (Fig. 1.3a) which uses all previous example to train and sliding-window learning [18] (Fig. 1.3b) which discards examples after a specific time, the objective of OWR is to adapt to anomaly and concept drifts flexibly, as shown in Fig. 1.3c.

In the traditional bootstrapping process, each example has uniform weight, which means each example is resampled with the same probability. In the proposed OWR, a vector of sample weight $\boldsymbol{W} = w_1, w_2 \dots, w_t$ is maintained and updated according to the prediction error of the incoming example. OWR consists of two parts: 1) Update the weight vector W and 2) Create bootstrap dataset based on W. We first introduce the weight update mechanism. To effectively update W, OWR compares the prediction error of the incoming example with prediction error of previous examples. We denote the absolute error of the arriving examples and mean absolute error (MAE) of previous examples by e_t and $\bar{e}_t$ respectively:

$$e_t = |\hat{y}_t - y_t|, \tag{2}$$

$$\bar{e}_t = \frac{\sum_{j=1}^{t-1} e_j}{t-1}, \tag{3}$$

OWR consists of three weight-update rules as the following:

Anomaly adaptation: If the absolute error of incoming example s_t is significantly larger than the MAE of the previous examples, OWR will initialize s_t to a weight $\alpha, \alpha < 1$. The weight w_t of new incoming examples

$$w_t = \begin{cases} \alpha, & \text{if } e_t > \varepsilon \bar{e}_t \\ 1, & \text{otherwise} \end{cases} \tag{4}$$

where ε is a constant and $\varepsilon > 1$.

Concept drift adaptation: The concept drift adaptation is designed in a way that a single spike of prediction error does not lead to a drastic change in weights of previous examples.

Therefore, W should be updated only when consistently high prediction errors are observed. Similar to [19], we define a warning and a drift threshold denoted as L_W and L_D respectively and $L_W < L_D$. When the error exceeds L_W at a certain time t_W, OWR enters the warning mode and records t_W. In the warning mode, if the error drops below the threshold L_W, OWR goes back to the normal mode. On the other hand, if the error reaches L_D at time t_D, a drift is confirmed and the weights of all examples before t_W will be reduced by a factor of β. To set a proper threshold L_W and L_D, we keep the minimum absolute error of previous examples e_t^{min} starting from the time of the beginning ($t = 0$) or previous drift t_D^*. We have

$$e_t^{min} = \min_{j=t_D^*,2\ldots t-1} e_j \,, \tag{5}$$

$$L_W = \epsilon_1 e_t^{min} \,, \tag{6}$$

$$L_D = \epsilon_2 e_t^{min} \tag{7}$$

where ϵ_1 and ϵ_2 are constants and $1 < \epsilon_1 < \epsilon_2$. After the weight is updated, OWR will reset t_W and e_t^{min}, and update t_D^* with the current time t_D.

Forgetting: Forgetting mechanism is implemented with a fading factor γ in OWR. Whenever a new example s_t arrives, the new weights w_j' of existing training examples $s_1, s_2 \ldots, s_{t-1}$ are:

$$w_j' = \gamma w_j \,,\ j = 1,2,..,t-1. \tag{8}$$

Note that standard online learning and sliding-window online learning in Fig. 1.3 can be implemented by properly adjusting fading factor γ.

To utilize the sample weight W, we modify the bootstrapping process of RF. At time t, each decision tree D_j $(j = 1,2 \dots, M)$ in RF is trained with bootstrap datasets sampled from the original dataset $S = s_1, s_2 \dots, s_{t-1}$. Based on W and $N = t - 1$, we present Weighted Bootstrapping Algorithm. The main idea of Weighted Bootstrapping algorithm is as follows. In step 1, a new bootstrap dataset for one decision tree is initialized. In step 2, the weights in W are mapped into the interval $\left[0,\ \sum_{j=1}^{N} w_j,\right]$ with subintervals $I_1, I_2, \dots\ I_N$. The length of the subintervals is proportional to the value of their weights. In step 3 to 7, the example is drawn using subintervals $I_1, I_2, \dots\ I_N$ and uniform distribution function. The process repeats $N = t - 1$ times where the size of all bootstrap datasets equals original dataset. Consequently, the example with higher weights is more likely to appear in the bootstrap dataset.

Weighted Bootstrapping Algorithm

Input: Training dataset $\boldsymbol{X} = \{x_i, i = 1,2 \dots N\}$, a sequence of N examples, and weights of the N examples $\boldsymbol{W} = \{w_i, i = 1,2 \dots N\}$, $w_i \in [0,1], \forall i$

1: Create a new dataset $\boldsymbol{X'}$ with the same size as $\boldsymbol{X}$

2: Partition the interval $\left[0, \sum_{j=1}^{N} w_j\right]$ into N subintervals $I_1 = (0, w_1)$, $I_2 = (w_1, w_1 + w_2)$, $\dots I_N = (\sum_{j=1}^{N-1} w_j, \sum_{j=1}^{N} w_j)$

3: **for** $i = 1$ **to** N **do**

4: Simulate $u \sim U\left(0, \sum_{j=1}^{N} w_j\right)$, U is uniform distribution function where the probability density of $U(a, b)$ is

$$f(x) = \begin{cases} \frac{1}{b-a}, & a \le x \le b \\ 0, & \text{otherwise} \end{cases}$$

5: Identify the interval $I_{j^*}, j^* \in \{1,2 \dots N\}$ such that $u \in I_{j^*}$

6: Add sample x_{j^*} to $\boldsymbol{X'}$

7: **end for**

Output: $\boldsymbol{X'}$

1.4 Results and Discussion

In this section, we will discuss the experiment settings in both offline learning and online learning scenarios. We will present results obtained by using the proposed RFFS and compare the results with existing ML models in an offline learning scenario. We will validate the effectiveness of personalized recommendations of health behavior generated by our personalized BP model using RFFS. Finally, we will compare the results obtained by using OWR and existing online learning methods in an online learning scenario.

We implement and evaluate our proposed methods in Python environment. We also use Scikit-learn library [34], Keras [35], and Crème [36] to implement other ML methods. We filter and impute the missing health behavior features using K Nearest Neighbor (KNN) with K = 5. Data is then standardized to zero mean and unit variance before training. MAE, root mean square error (RMSE) and mean absolute percentage error (MAPE) are calculated and used as our evaluation metrics. Their definitions are as follow:

$$MAE = \frac{\sum_{i=1}^{n}|\widehat{BP}^i - BP^i|}{n} \tag{9}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}\left(\widehat{BP}^i - BP^i\right)^2}{n}} \tag{10}$$

$$MAPE = \frac{nMAE}{\sum_{i=1}^{n}|BP^i|} \times 100\% \tag{11}$$

where $\widehat{BP}^i$ is the ith prediction of BP made by trained models and BP^i is the actual value of the ith BP.

1.4.1 Offline Learning

For offline learning, we use 5-fold cross-validation to randomly split our dataset into training (80%), and test (20%) sets five times and average the prediction results. Each person's model is trained with his/her own data. To show the effectiveness of RFFS, we compare the performance with several representative machine learning algorithms, including Support Vector Machine (SVM) [37], Gradient Boosting Decision Trees (GBDT), Multilayer Perceptron (MLP) [38], and Long Short-Term Memory (LSTM) [39]. We also compare the prediction using RF with only original features (termed as RF) and RF with derived time series features but without feature selection (termed as RFTS). For model setting details, we set the number of trees as 500 for all RF models. We set the maximum ratio of total features used in each tree as 0.33 and the minimum number of samples to split as 2. For SVM, the rbf kernel is used and the best γ and C are selected using cross-validation. For GBDT, the number of trees as 100 and the learning rate as 0.1. Both MLP and LSTM models were trained using 0.001 and 20 as the learning rate and batches size with Adam optimizer [40]. Due to the small number of the training examples, the total depth of fully connected layers in both MLP and LSTM models was set as 4 and the maximum neurons in each layer as 50. We also use early stopping and dropout layers in both models with dropout rate as 0.2 to avoid overfitting.

The MAE, RMSE, and MAPE of BP prediction of our proposed method and other methods are summarized in Table 1.2. Noted the values in Table 1.2 are the average MAE, RMSE, and MAPE over all the users, and we will discuss the performance of individual users later in the next paragraph. As shown in Table 1.2, RFFS achieves the lowest prediction error in terms of MAE, RMSE, and MAPE. We observe that the performance of both DL models (MLP and LSTM) is the worst of all methods. The worst prediction error is because the DL models may overfit the small

TABLE 1.2. Comparison of MAE, RMSE, and MAPE of different models (offline learning)

	SBP			DBP		
	MAE	RMSE	MAPE (%)	MAE	RMSE	MAPE (%)
SVM	6.30	8	5.26	5.30	6.69	6.74
MLP	9.06	11.11	7.48	7.57	8.05	9.04
LSTM	8.63	10.74	7.16	6.23	7.87	8.12
RF	5.75	7.6	4.80	5.08	6.78	6.69
GBDT	5.91	7.82	4.97	5.21	6.99	6.86
RFTS	5.94	7.67	4.94	4.82	6.51	6.30
RFFS	5.18	6.88	4.31	4.30	6.12	5.64

training dataset (~180 examples for each user). GBDT and RF perform similarly while RF has the lowest prediction error among existing models. Our proposed RFFS performs better than RF by 10.2% and 9.5% in terms of MAE and RMSE in SBP; 15.4% and 9.7% in terms of MAE and RMSE in DBP respectively. Note that although RFTS (using time series features but without feature selection) performs better than RF for DBP, it is worse than RF for SBP. Based on the comparison among RF, RFTS and RFFS, including time series features in prediction is beneficial only with the proposed feature selection technique.

Since SBP and DBP perform similar trends of prediction performance, we will use the MAE of SBP to illustrate the performance of personal prediction in Fig. 1.4. We have the following observations: 1) RFFS has the lowest MAE among most users except for user 4 where RFFS performs worse than RF by 6.4% and 6.8% in terms of RMSE and MAE, 2) the prediction performance greatly varies with different users. For example, the MAE of SBP ranges from 3.65

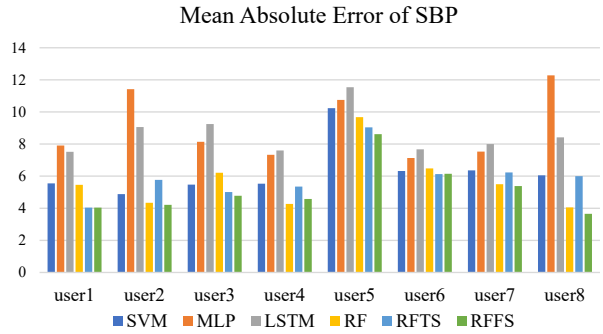

Figure 1.4. MAE of SBP on different users

to 8.62 with RFFS, and 3) RFTS leads to mixed results of SBP prediction. For example, RFTS performs better than RF for user 1, 3, 5, and 6 in terms of MAE, but it is worse for user 2, 4, 7, and 8. The above observation indicates that the effect of health behavior on BP and the temporal correlation of BP differ from person to person. For some users, including the historical value and trend of their BP improves the prediction. However, for other users it is not useful but increases the dimension of the data.

Next, we will show the benefit of the personalized model by comparing its prediction performance with the non-personalized model in the next paragraph. The comparison between the personalized model with an aggregated model is shown in Table 1.3. For the aggregate model, the data of all participants are concatenated into a single dataset with one additional feature specifying different users, which is then used to train a single aggregated RF model, as opposed to the case of personalized models, each of which uses a single person's data for training. Note that the RMSE and MAE of the personalized model shown are the averages of personalized predictions of all participants.

As shown in Table 1.3, the personalized model performs better than the aggregated model by 20.3% and 23.3% in terms of RMSE and MAE of SBP; 7.2% and 14.5% in terms of RMSE and MAE of DBP, respectively. Although the aggregated model is trained with a larger dataset and hence theoretically should have performed better than the personalized models, the latter

TABLE 1.3. Comparison of RMSE and MAE models using personal and aggregated dataset

	SBP		DBP	
	RMSE	MAE	RMSE	MAE
Personal model	6.88	5.18	6.12	4.30
Aggregated model	8.61	6.75	6.58	5.03

outperforms since the relationship between health behavior features and BP varies significantly across individuals.

1.4.2 Validation of Personalized Recommendation

To validate the effectiveness of health behavior recommendation suggested by the RFFS model, we observe whether users' BP will change after they change their top health behavior features (ones with the highest feature importance) based on feature importance calculated by RFFS. In this experiment, we exclude BP time series features derived in Section 1.2.1. We randomly select two users and provide each with personalized recommendations, in the form of the top 3 health behavior features in his/her BP prediction. We suggested them to change their top behaviors and observed their BP level. The week before the recommendation (week1) and the week after the recommendation (week2) are used for comparison.

In Table 1.4, we list the top 3 features and their normalized importance score of user 1 and user 2. We can observe that the top 3 features are very different for each user. BP is mainly correlated to exercise for user 1 and sleep for user 2. We suggested user 1 to increase exercise and suggested user 2 to go to bed earlier in week 2. We summarize the changes of their top features in

TABLE 1.4. Top features and the weekly change of top 3 features and the corresponding BP level

	User 1	User 2
Top 3 features (Normalized importance score)	1. Minutes sedentary (0.080611) 2. Calories burned (0.074208) 3. Minutes lightly active (0.070083)	1. Bedtime (0.109519) 2. Minutes awake (0.102392) 3. Time in bed (0.090492)
Change of the top features from week 1 to week 2	1. 21 minutes decrease in sedentary minutes 2. 152 calories increase in calories burned 3. 12 minutes increase in lightly active minutes	1. 58 minutes earlier in bedtime 2. 41 minutes increase in awake time 3. 66 minutes increase in time in bed
Average BP of the weeks (SBP/DBP)	Week 1: 130.71 / 82.57 Week 2: 110.57 / 70.86	Week 1: 106.67 / 68.33 Week 2: 102.83 / 63.83

Table 1.4. The daily BP level in the two weeks is shown in Fig. 1.5, and the average BP levels in week 1 and week 2 are shown in Table 1.4. We can observe that both SBP and DBP of user 1 decrease with more exercise, and with earlier bedtime and more sleep for user 2. Although there is no exact conclusion indicating a causal relationship between the top features of health behavior and the BP level of the users, the results show that changing the top personal features can potentially help users improve and control their BP levels. Moreover, the change of BP varies significantly between the two users. The possible reason is that the stableness of BP and its correlation to health behavior may differ among people.

1.4.3 Online Learning

Standard cross-validation does not apply to online learning since random splits cannot reflect the characteristics of the data stream. In this study, we use prequential evaluation [41] to evaluate the proposed OWR in the online learning problem described in Section 1.3. Whenever a new example is observed, the current model makes a prediction. After the actual BP is known, we evaluate the error and update the model using OWR. We compare the performance of OWR with two popular online learning methods: 1) Mondrian Forests (MF) [22] and 2) Stochastic gradient descent (SGD) [23]. To show the effectiveness of our proposed OWR, we also compare the results using RFFS with 1) all previous examples (namely RFFS-All) and 2) examples in a fixed-size

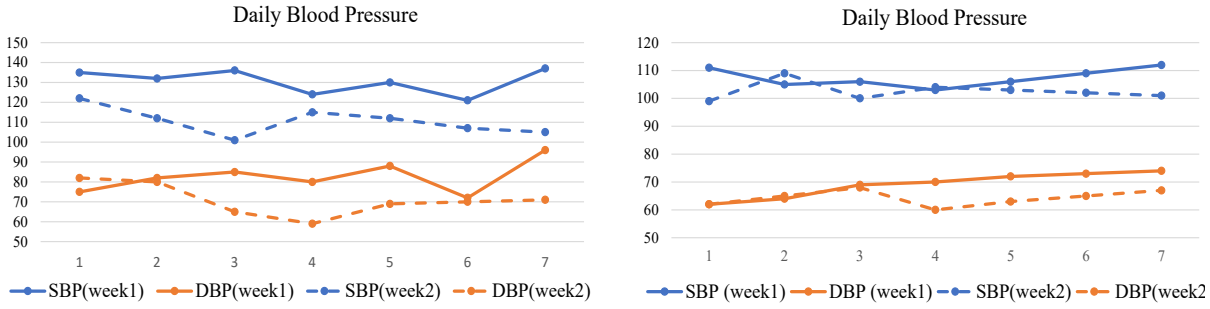

Figure 1.5. (a) left, daily BP of user 1 and (b) right, daily BP of user 2

TABLE 1.5. Comparison of time-averaged MAE, RMSE, and MAPE of different online learning models

	SBP			DBP		
	MAE	RMSE	MAPE (%)	MAE	RMSE	MAPE (%)
SGD	6.05	7.96	5.02	4.86	6.76	6.67
MF	6.12	8.02	5.08	4.90	6.62	6.71
RFFS-All	6.09	8.00	5.05	4.86	6.61	6.65
RFFS-Win	6.21	8.12	5.15	4.81	6.59	6.59
OWR	5.66	7.51	4.69	4.49	6.11	6.14

window (namely RFFS-Win). The hyperparameters for RF-based techniques are the same as the offline learning settings in Section 1.4.1. Cross-validation method is used to select the optimal parameters in OWR. The size of the window in RFFS-Win is 20, which is chosen based on the performance.

The MAE, RMSE, and MAPE of BP prediction of our proposed OWR method and other online learning methods described above are summarized in Table 1.5. Note the values reported in Table 1.5 are the time-averaged MAE, RMSE, and MAPE of all users in the prequential evaluation scheme. Table 1.5 shows that the proposed OWR method produces the best prediction of BP in terms of all the evaluation metrics used, MAE, RMSE, and MAPE. Among the compared methods, SGD performs slightly better than others in terms of SBP while RFFS-Win is the best in terms of DBP. Our proposed OWR performs better than SGD by 6.5% and 5.8% in terms of MAE and RMSE of SBP; 6.7% and 7.3% better than RFFS-Win in terms of MAE and RMSE of DBP respectively. Note that the prediction error is higher in online learning scenario than offline learning scenario. It is because time-averaged MAE, RMSE, and MAPE include the prediction error from the beginning when there are not enough training examples. Compared with RFFS-ALL, which uses all previous examples for training, OWR outperforms by 7.2% and 7.6% in terms of MAE of SBP and DBP respectively.

Time-averaged MAE of SBP and DBP of user 1 to user 4 are shown in Fig. 1.6. We remove the first 10 days of MAE since the MAE is too high and fluctuating to represent the actual performance. From Fig. 1.6, we can make the following observations: 1) As the training data accumulates, the MAE shows a downward trend for all users; 2) OWR consistently gives the best BP prediction except for user 1's DBP where OWR performs worse than RFFS-Win by 2.4% in terms of the last MAE; 3) Both the prediction performance and the required time to achieve the lowest MAE varies significantly with different users. For example, the minimum MAE is achieved on day 80 for user 1 and user 2 while it is day 38 for user 3. Moreover, brief periods of increasing trends are also observed, which indicates the existence of potential concept drifts and anomaly points. The above information can provide personal insight in addition to MAE, such as the required training examples and indicators of abrupt/gradual change of BP.

1.5 Conclusion

In this paper, we investigate the personal effect of health behavior on BP using data collected from wearables and home BP monitors. Our proposed method predicts BP one day ahead and provides the importance ranking of health behavior. We extract the time series feature from

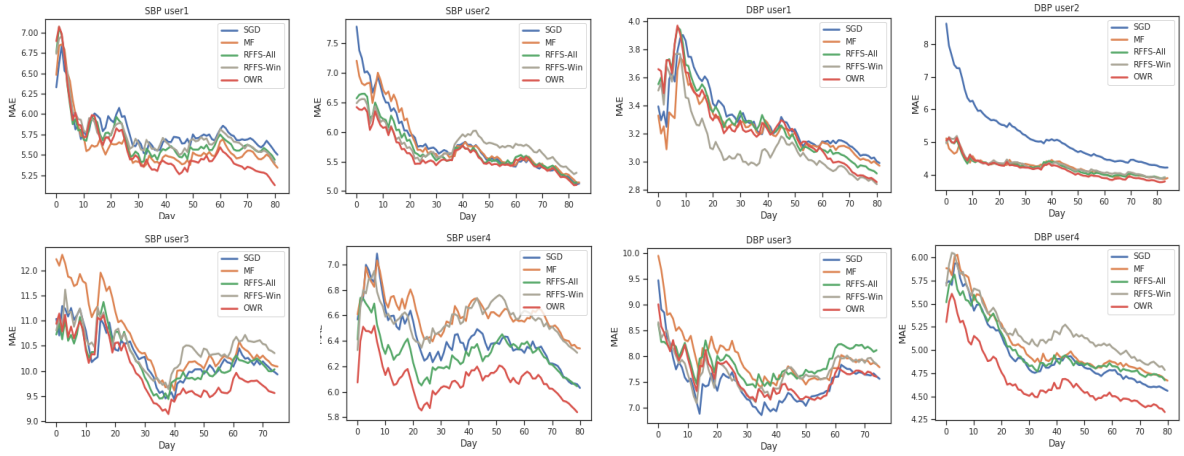

Figure 1.6. (a) left, time-averaged MAE (SBP) on four users and (b) right, time-averaged MAE (DBP) on four users

the raw data and apply the RF-based feature selection (RFFS) technique to enhance the prediction performance. In online learning scenario where BP and health behavior data are sequentially collected, we propose an Online Weighted-Resampling (OWR) technique to adapt the training dataset based on prediction performance. OWR can alleviate the negative effect of concept drifts and anomaly points.

The experimental results show that our techniques outperform other existing techniques in both offline learning and online learning scenarios. The accuracy of our method is comparable with the standard cuff-based BP measurement (MAE of BP below 5.0). Moreover, we show that significant changes in BP can be possible after users changed their health behavior features suggested by our model. In future work, we will extend our proposed method on a larger and more heterogeneous group of subjects in order to obtain a more representative result. We will solve the mismatch between the manual measurement of BP and high granularity of wearable data. This can be done either by better representation of features or incorporating continuous BP estimation techniques to increase the frequency of BP measurement.

Chapter 1, in part, is from the material as it appears in proceedings of IEEE conference on Healthcom 2018. Po-Han Chiang, Melissa Wong and Sujit Dey, and in IEEE Access 2019. Po-Han Chiang and Sujit Dey. The dissertation author was the primary investigator and author of the papers.

Chapter 2

Using Wearables and Machine Learning to Enable Personalized Lifestyle Recommendations to Improve Blood Pressure

2.1 Introduction

High blood pressure, or hypertension is one of the most prevalent chronic diseases in the world [1]. Stepwise management of hypertension begins with modifying lifestyle factors (e.g., activity, sleep) which, alone, can be effective in controlling BP [2,3,42,43]. What remains lacking in the literature is the individual effect of these lifestyle factors on BP. Traditionally, these relationships have been investigated through large-scale Randomized Controlled Trials (RCTs). However, the aggregate insights derived from RCTs are not necessarily tailored for individuals. That is, the impact of specific lifestyle factors on BP may differ across individuals due to an individual's unique genomic makeup. Secondly, the data in the RCTs are usually collected in healthcare settings or self-reported fashions. It is well-established that BP measurements obtained in healthcare settings are often unreliable [44], while self-reported data often falls short of accuracy and granularity.

In contrast, wearable activity trackers, or wearables, such as Apple Watch, Fitbit and Samsung Galaxy Watch, collect a great amount of lifestyle data in high granularity and continuity. As a result, a personalized model for BP and lifestyle factors can be built for each individual based on his/her data. To date, the potential of using wearables' data for BP management has not been

fully investigated due to the complex dependency between BP and lifestyle factors. In this study, we propose to use machine learning (ML) techniques to elucidate the complex relationships between BP and lifestyle factors at the level of the individual. Based on the continuous data collected from wearables of users, we aim to 1) build a predictive model of BP for individuals, which will give users a quick and reliable way to understand their health condition, and 2) utilize the above model to provide personalized and precise insight to users, as opposed to general lifestyle recommendations.

In our preliminary work [9], we used Fitbit Charge HR and Omron Evolv to collect lifestyle and BP data, respectively, of 8 volunteers. We then trained a Random Forest (RF) model [6] to predict the 24-hour-ahead BP for each volunteer using lifestyle data. We proposed a stable and consistent feature selection technique, namely Random Forest with Feature Selection (RFFS), to enhance the prediction accuracy of RF. Moreover, we used the relative importance of the features generated by RFFS to identify the most important lifestyle factors for his/her BP. The most important lifestyle factors were shared with selected subjects. We observed that the above subjects changed their lifestyle factors according to the shared information and their BP decreased from its previous level. In [45], we proposed an online ML technique to prioritize training samples based on the performance of prediction. The proposed technique addressed the challenge of concept drifts and anomaly points due to sequential data collection.

However, there were three main limitations to be addressed: 1.) The dataset consisted of a series of BP and lifestyle factors data with mixed sampling frequency. Extra feature engineering and modeling for time series were necessary to fully utilize the potential of temporal dependency. 2) The selection of the feature in RFFS was based on how each feature improved the prediction accuracy of BP; however, it did not imply how each feature is affecting BP. 3.) The

recommendation was only given to two subjects, and the duration of observation after the recommendation was only one week. The lack of a control group and short observation time made it challenging to reach a significant conclusion.

To tackle the above challenges, we extract new features from raw data collected by wearables and BP monitors. We aggregated the raw lifestyle data, which was mostly recorded every minute, into a summary of 1-hour, 24-hours, 48-hours and 72-hours before each BP reading and extracted features with the above non-overlapping and contiguous time windows. The improved granularity and representation of features extracted from wearables are not only improving the accuracy of BP prediction but also comprehendible for patients and physicians. Secondly, to capture the periodicity and the trend of previous BPs, we create new features based on Autoregressive Integrated Moving Average (ARIMA) model [46] to better represent the BP time series. To deal with unevenly spaced BP readings, we propose to transform the original BP time series into an evenly spaced time series by resampling and interpolation. To explore the best feature selection method, we evaluate multiple popular methods, and we choose Shapley value [47] based on its prediction performance and interpretability. Shapley value is a model-agnostic feature interpretation method derived from Game Theory. Given a set of feature values and a trained ML model, Shapley value can indicate how each feature contributes to the actual BP prediction from the mean prediction. We propose a feature selection method, namely RF with Shapley-Value-based Feature Selection (RFSV), which uses feature importance based on Shapley value to remove redundant and irrelevant features. Moreover, we use the top features selected by RFSV to provide the precise insight that may affect an individual's BP.

To evaluate the effectiveness of the proposed techniques, we conducted a randomized controlled experiment with patients who have Elevated BP or Stage I hypertension and were not

taking any antihypertensive medications. We collected BP and wearable data and trained the BP prediction model for each subject. Subjects were randomized to either receive personalized lifestyle recommendations based on their data (experimental group) or not receive lifestyle recommendations (control group). We compared and discussed the change of BP levels across the study period for both groups.

The rest of the paper is organized as follows. In Section 2.2, we will investigate the related work of BP prediction technique and BP studies using lifestyle intervention. In Section 2.3, we present the overall architecture of the BP prediction and recommendation system with the proposed RFSV. We then detail data collection and representation, ARIMA time series feature extraction and RFSV. In Section 2.4, the prediction performance of the proposed method is compared with other ML methods. Moreover, we will discuss the effectiveness of personalized lifestyle factors recommendations suggested by the proposed system. Finally, we conclude the paper in Section 2.5.

2.2 Related work

The authors in [12] predicted BP using demographic and contextual data (e.g., age, weight and smoking habit) with an artificial neural network (ANN). However, the prediction was based on a single BP measurement and did not consider the dynamics of BP. In [13,14, 48-50], PPG signals were used to predict short-term BP with ensemble trees models [13,14] and neural-network-based models [48-50]. However, PPG-based prediction is only applicable for a very short time horizon (~10 minutes), while our technique aims to predict BP in a longer time horizon, to provide actionable information to users. In [15], the 24-hour time series of BP and heart rate were trained with Extreme Learning Machine (ELM) to provide hourly BP prediction. However, the length of collected data in [15] was only a single day, and the prediction performance was not

compared with other ML methods. The authors in [16] proposed to predict BP using Long Short-Term Memory (LSTM) models [39] with additional contextual data (e.g., age, BMI and BP medication) layer. The data in [16] was averaged every month, so the temporal relationship of data was not fully utilized due to lower temporal resolution and information loss in the averaging process. All the above studies did not use physical activity and sleep data, which were the most relevant lifestyle data related to BP that can be collected by current technology. During physical activity, heart rate and stroke volume increase to meet the metabolic requirements of the muscles, which result in expansion of arteries and force exerted against the artery changes, which is translated into BP [51]. Although BP normally increases during physical activity, the inverse relationship between physical activity and BP has been shown in numerous observational studies and can be explained by the reduction of arterial stiffness through exercise [52]. Secondly, inadequate sleep, including issues of quantity and efficiency, also has a significant negative impact on BP, possibly by higher hypothalamic-pituitary-adrenal axis activation [53]. Besides activity and sleep factors, it has been known that dietary factors, like sodium intake, may also affect BP [2, 54]. Traditional methods assess food (nutrition) intake with self-report measures, such as food frequency questionnaires (FFQs) and photo-assisted dietary assessments [55]. However, the accuracy of dietary intake assessment remains a challenge. Moreover, no widely adopted technology can assess dietary intake automatically and accurately [55]. Therefore, we focus on only physical activity and sleep factors in our study. Based on heart rate and steps collected by wearables, the authors in [11] trained bidirectional LSTM models to diagnose various chronic diseases, including hypertension. However, the proposed methods focused on the diagnosis of hypertension and did not provide a numerical prediction of BP.

In addition to BP prediction, the other critical insight from BP analysis is how lifestyle factors such as physical activity and sleep affect an individual's BP. Although the effectiveness of lifestyle interventions on BP management has been proven in many studies [2-5], the insight on an individual level is absent. Long-term BP and the result of exercise treadmill stress tests were used for BP factor analysis in [57]. The authors compared different interpretable ML techniques and concluded that those techniques could derive different insights on the model behavior. In [58], a mobile app was designed to deliver behavioral recommendations on diet and exercise to manage hypertension. The authors in [58] collected biometric, demographic and engagement data from a mobile app, and they proposed ML models to predict participant completion of the intervention. The BP factors collected by the above studies were either from electronic health records or self-reported methods, so the accuracy and granularity of lifestyle factors were limited. In contrast, our method uses wearables to collect lifestyle data, which enhances the quality and granularity of the data. Therefore, our model can pinpoint the lifestyle factors responsible for an individual's BP. Moreover, the conclusions of previous studies are only drawn from ML models without validating the effectiveness of the recommendations. In our study, we provide recommendations based on Shapley Value and conduct a randomized experiment to validate the effectiveness of recommendations.

2.3 Method

In this section, we will first introduce the clinical study and data collection process. We will give an overview of the BP prediction and lifestyle recommendation system and discuss each step in more detail.

2.3.1 Clinical Study Cohort and System Architecture

Our clinical study (protocol #181405) was reviewed and approved by UC San Diego Human Research Protections Program, which operates Institutional Review Boards (IRBs) at UC San Diego. The study was in collaboration with UC San Diego Health, with patient enrollment, onboarding and management conducted by the Altman Clinical & Translational Research Institute at UC San Diego. Patients were screened for recruitment with UC San Diego Health System's electronic health record. The selection criteria included subjects who were pre-hypertensive or with Stage I hypertension (SBP between 120-140/ DBP under 90 per ACC/AHA 2017 guidelines [59]) and who were not taking any antihypertensive medications. Subjects who had consented were provided a Samsung Galaxy Watch and an Omron Evolv wireless BP monitor to collect their lifestyle factors and BP data for 90 days. Of the 36 consented subjects, data of 11 subjects were excluded since they withdrew from the study or failed to collect data for at least half the study duration (45 days) in the study period. The characteristics of the included cohort are shown in Table 2.1. Data was collected remotely through the application programming interfaces (APIs)

TABLE 2.1. Cohort Statistics (n =25)

Age (yrs +/-SD)	50.2 +/- 14.3
Male	16
Female	9
SBP (Mean +/- SD)	126.2 +/- 8.3
DBP (Mean +/- SD)	78.5 +/- 6.3
Average Resting Heart rate	69.6 +/- 10.1
Peak Heart rate	92.9 +/- 12.3

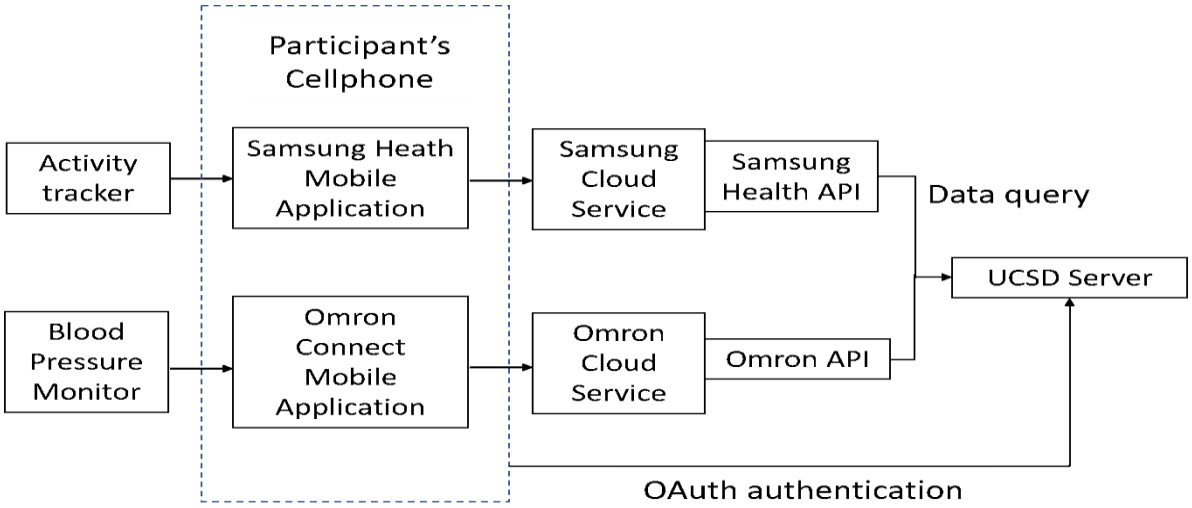

Figure 2.1. Block Diagram of data storage and access

provided by Samsung and Omron, as shown in Fig. 2.1. The primary metrics used to measure BP are systolic and diastolic blood pressure (SBP and DBP), which are defined as the maximum and minimum BP, respectively, during a pulse.

The objectives of our proposed system, shown in Fig. 2.2, are prediction of BP for an individual, identification of the most important features that impact the individual's BP trend and providing personalized and precise recommendations on lifestyle factors that will positively impact his/her BP trend. To achieve the objectives, we train a ML model to predict the current BP level using one's historical BP readings as well as activity, sleep and heart rate data collected from the Galaxy Watch. The raw data are then filtered, extracted and imputed as features. To better capture temporal information in BP time series, we extract time-series features of BP using

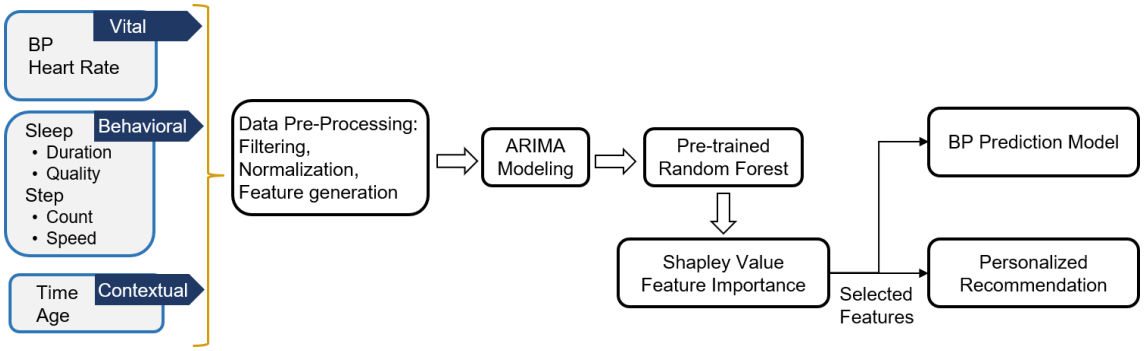

Figure 2.2. System architecture of the proposed method

ARIMA, as discussed in Sec. 2.3.3. The feature selection based on a pre-trained RF model and Shapley value is performed to remove redundant and/or irrelevant features in BP prediction. In addition to building a predictive model of BP, we will provide personalized lifestyle recommendations to our subjects by pointing out the most important factors affecting their BP based on Shapley value.

2.3.2 Data Characteristics and Features Extraction

The Galaxy Watch provides heart rate (HR), number of steps, walking/running speed, floors climbed, sleep duration and sleep stages of the user. Also, we discretize the activity data into different levels of active time (sedentary, lightly active, very active) based on subjects' steps and heart rate every minute. Maximum HR (HR^{max}) of each subject is calculated as [59]:

$$220 - age. \tag{1}$$

Three HR zones (zone 1, 2, and 3) are defined as [27]:

$$Z * HR^{max}, Z \in [.5, .7, 1]. \tag{2}$$

We define three active levels as follows: sedentary (steps < 10 or HR in zone 1), lightly active (steps ≥ 10 and HR is in zone 2), and very active (steps ≥ 10 and HR in zone 3). Sleep data includes sleep duration, bedtime, wake-up time and sleep stages. Bedtime and wake-up time represent the time subjects go to sleep and wake up, respectively. Sleep stages include light sleep, REM sleep and deep sleep. We also define the average heart rate during sleep as slpHR.

Data from the Galaxy Watch is mostly recorded every minute while BP is measured by subjects twice per day, so the data consists of time series with different frequencies. Moreover, although the guideline for BP measurement in this study is to measure in the morning (8-10 am) and at night (7-9 pm), there are missing values, time deviations (e.g., measurement in the afternoon)

Features	Average	Range
Steps_24	6258.3	1100-17620
Speed_24 (m/s)	4.2	1.3-8.6
MaxHR_24 (beats/minute)	148.2	110-200
Floors_24	6.2	0-50
Sedentary_24 (%)	45.3	40.5 - 56.6
Lightly_active_24 (%)	10.2	2.2 - 14.3
Very_active_24 (%)	1.6	0 – 5.1
Bed_time_24 (HH:MM)	22:03	20:57-2:01
Up_time_24 (HH:MM)	7:20	05:51-9:21
Sleep_duration_24 (minutes)	453	297-560
Light_sleep_24 (%)	39	31.1-46.5
REM_24 (%)	35	32.8-41.2
Deepsleep_24 (%)	4	3.2 – 8.0
Sleep_HR_24 (beats/minute)	64.2	59.1-78.4
SBP_arima	126.4	118.4-155.6
DBP_arima	78.2	69.1-91.3

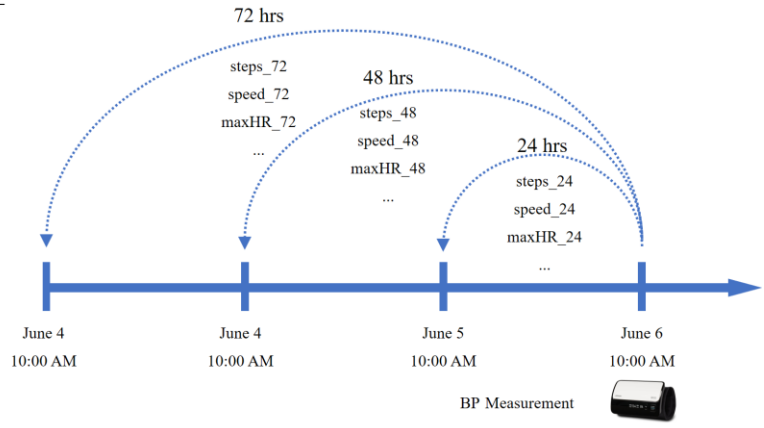

Figure 2.3. Left: Statistics of representative features. Right: Illustration of feature extraction

and redundant values (e.g., two-morning measurements at 7 am and 9 am, respectively). Thirdly, most of the lifestyle factors such as sleep and activity, have a daily cycle. Based on the above observations, we extracted the lifestyle factors data as a summary of 24-hours, 48-hours and 72-hours before each BP reading and extracted features using the above non-overlapping and contiguous time windows. For example, for each pair of (SBP, DBP), the feature "steps_24" was defined as the total number of steps in the previous 24 hours before the measured BP and "steps_48" was the average of the total daily steps in the previous 48 hours. Note that instead of summation, HR and walking/running speed were averaged over the previous 24/48/72 hours and MaxHR is the maximum HR over the previous 24/48/72 hours. Finally, "measure_time" denotes the time in

TABLE 2.2. Features and Target Variables

Target Variables	Original Features	Derived Features
SBP, DBP	heart_rate_24, maxHR_24, steps_24, speed_24, floors_24, bed_time_24, up_time_24, sleep_duration_24 (sleep_24), light_sleep_24 (Lsleep_24), REM_24, deepsleep_24 (Dsleep_24), heart_rate_48, maxHR_48, steps_48, speed_48, floors_48, bed_time_48, up_time_48, sleep_duration48 (sleep_48), light_sleep_48 (Lsleep_48), REM48, deep_sleep48 (Dsleep_48), heart_rate_72, maxHR_72, steps_72, speed_72, floors_72, bed_time_72, up_time_72, sleepduration72 (sleep_72), light_sleep_72 (Lsleep_72), REM72, deep_sleep_72 (Dsleep_24), measure_time	SBP_arima, DBP_arima, sedentary_24, lightly_active_24, very_active_24, slpHR_24, sedentary_48, lightly_active_48, very_active_48, slpHR_48, sedentary_72, lightly_active_72, very_active_72, slpHR_72,

a day when BP was measured. The statistics of the representative features over the previous 24 hours and the method of feature extraction are shown in Fig. 2.3. The original and derived features are summarized in Table 2.2.

2.3.3 ARIMA Feature Extraction from BP time series

Time series prediction problems include a set of time-ordered observations $s_j = (X_j, y_j)$, $j = 1,2 \dots J$ where X_j are the values of features X and y_j is the value of target y observed at time j, and the task is defined as predicting the future values of y_u for time $u > j$ given $s_1, s_2 \dots, s_j$. In addition to using X as features, time-series features can be extracted from $y_1, y_2 \dots, y_j$ to capture the temporal relationship of y. In this paper, we use ARIMA [46] to capture the temporal pattern of BP series.

Three parameters (p, d, q) are used to construct the ARIMA model, and (p, d, q) stands for the order of the autoregressive model, the order of differencing, and the order of the moving average model, respectively, and the prediction y_j can be expressed by:

$$(1 - S)^d y_j = \delta + \alpha_1 y_{j-1} + \alpha_2 y_{j-2} + \cdots + \alpha_p y_{j-p} + \varepsilon_1 - \beta_1 \varepsilon_{j-1} - \beta_2 \varepsilon_{j-2} - \cdots - \beta_q \varepsilon_{j-q} \quad (3)$$

where S stands for the backward shift operator for $S(y_j) = y_{j-1}$, δ is the constant, $\alpha_1, \alpha_{2,} \dots, \alpha_p$ are the autoregressive parameters, ε_j is the random error at time t and $\varepsilon_j \sim N(0, \sigma^2)$, and $\beta_1, \beta_2, \dots, \beta_q$ are the moving average parameters. To cope with seasonality, the authors in [60] proposed Seasonal ARIMA (SARIMA). In SARIMA, additional seasonal AR and MA terms are used for prediction using values at times with lags that are multiples of pre-defined periods T (the span of the seasonality). In this paper, we set $T = 1$ days. To determine these three parameters (p, d, q), we run an exhaustive search to determine the best ARIMA model for each subjects' BPs

and the corresponding set of optimal parameters. After the model is developed, one-step forecasts from the ARIMA model are defined as additional features, namely SBP_arima and DBP_arima.

As described in the previous section, the BP series is not evenly spaced due to manual measurements. For example, a subject may measure his/her BP at 7 am, 3 pm and 9 pm on one day while measuring his/her BP only at 6 pm on another day. However, ARIMA can only model evenly spaced time series. To address this issue, we transform the BP data into evenly spaced observations by resampling and linearly interpolating the closest two BP readings before and after each resample point. Note that the resampled BP series is not the actual BP measurement and is used only to generate ARIMA features.

2.3.4 Predictive Modeling using Random Forest (RF)

To select the best ML methods for our task, we evaluate popular machine learning techniques, including Random Forest, Support Vector Machine [37], Gradient Boosting Decision Trees [61], LSTM [39], and ARIMA [46]. Although neural network-based approaches outperform in unstructured data like image and language, tree-based ensemble ML models constantly have the best performance in structured data where data is essentially in tabular form [7]. Moreover, neural networks are highly prone to overfitting when the underlying data sizes are small and no domain-specific insight can be used to design the architecture of the underlying neural network [62]. In this study, the number of BP samples for each subject is less than 180 (subjects are requested to measure their BP twice per day for 90 days) and the data is structured for interpretation purposes, which is best suited for tree-based ensemble ML models. Among the ML models, we find that RF gives the best performance through the evaluation in Sec. 2.4.2 (Table 2.3). Therefore, RF is used to model BP and lifestyle factors in this study.

RF is an ensemble predictor of several decision tree predictors. We will first introduce the

decision tree and its application in ML tasks, Classification and Regression Tree (CART) model. CART [33] is a non-parametric method used to build decision tree predictors in ML problems. CART arranges a sequence of questions (decision rules) based on input features into a tree-lie structure. A decision tree consists of two types of nodes: 1) internal nodes, which split the samples into two sub-trees or leaf nodes based on decision rules. Each internal node is labeled with a single input feature and a corresponding split threshold of that feature. 2) leaf nodes, where no more split is performed. In regression tasks, the target variable is continuous, so the prediction of the target variable is the average of all training samples at that node. In the training phase, the topmost internal node (root node) contains all training samples. At each internal node, the feature and its split threshold are selected to minimize the mean squared error of the prediction. In the prediction phase, the new sample moves down from the root node to one of the leaf nodes according to the splitting criteria along its path. The predicted value is then the average training sample at that leaf node.

RF is an enhanced approach by aggregating a collection of decision trees to reduce overfitting of the data and the resulting high variance of the prediction [6]. Compared with CARTs, RF introduces two major enhancements: bootstrap aggregation (bagging) and feature bagging. RF produces bootstrap datasets that are randomly and independently drawn with replacement from the training dataset. Each bootstrap dataset with the same size as the original training set is used to train a decision tree. Bootstrap aggregation in RF averages the prediction of decision trees trained with bootstrap samples, which greatly reduces the variance of prediction from a single decision tree. Moreover, since individual trees generated in the bagging process are identically distributed, the expected prediction of RF is the same as the expected prediction of individual trees. As a result, RF has a lower variance than individual trees, while its bias remains the same [30]. In addition to

bootstrap aggregation, RF further reduces the correlation between its member decision trees by introducing feature bagging, which randomly selects a subset of features when constructing each tree.

2.3.5 Feature Importance with Shapley Value

Although RF performs well on classification and regression tasks, high-dimensional data will degrade the performance, especially when the number of samples is relatively small. There may be redundant features, which provide no more information than the currently selected features, or irrelevant features, which may introduce noise instead of any useful information.

Feature selection techniques improve the prediction accuracy and reliability by removing irrelevant or redundant features across the datasets. In our study, the candidate feature selection method should not only improve the prediction performance but also measure the relevance between BP and the features. With the relevant information, the most relevant (important) feature can be used for personalized and precise recommendations. Based on the above objectives, we choose four representative feature selection methods, namely, Pearson Correlation-based Feature Selection [64], Information Gain-based Feature Selection [65], Random Forest Feature Importance (mean decrease impurity) [33], and Shapley Value Feature Importance [66, 67]. All four methods provide a numerical importance or relevance measure for each feature, which can be used to select the features for ML tasks and provide recommendations based on the importance score. Based on the empirical evaluation of prediction accuracy, which is detailed in Sec. 2.4, we select Shapley Value Feature Importance to select the features.

Shapley value, derived from Game Theory, assumes that each feature in a data sample is a ‘player’ in a game, and the prediction is the payout [11]. The Shapley value aims to distribute the payout among the features based on their contribution. To calculate feature importance for each

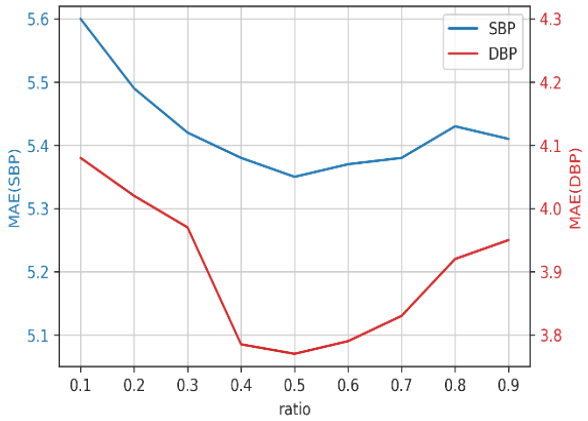

Figure 2.4. Prediction error with different ratios of

feature x_k , $k = 1,2 \dots K$, based on Shapley value, the model is evaluated over all possible feature value combinations with and without x_k. The Shapley value is calculated by [40]:

$$\phi_k = \sum_{S \subseteq X \backslash \{x_k\}} \frac{|S|!(p-|S|-1)!}{p!} \left(f(S \cup \{x_k\}) - f(S)\right) \tag{4}$$

where S represents all possible feature sets $S \subseteq X \backslash \{x_k\}$ and X is the set of all features. p is the number of the features in X and $|S|$ is the number of features in S. $f(S \cup \{x_k\}) - f(S)$ is calculated by the marginalized prediction using the model trained with feature set $S \cup \{x_k\}$ minus the prediction using model trained with feature S.

The complexity to compute the exact form of ϕ_k is prohibitively high since the number of possible sets S in (4) is 2^n where n is the number of features. In [41], the authors propose Tree Shapley Additive exPlanation (SHAP) algorithm to approximate Shapley value in polynomial time for tree-based ML models. This algorithm has been used in this work to calculate the feature importance. By averaging the absolute value of all Shapley values across all training samples, we can get the average contribution of a feature to the prediction of our pre-trained model. We define the feature importance vector for the j^{th} sample as $I_{Xy}(j) = [\phi_1, \phi_2 \dots, \phi_k]$ where $j = 1,2 \dots J, k = 1,2 \dots K$. The average feature importance can then be calculated by

$$\bar{I}_{Xy} = \frac{\sum_{j=1}^{J} |I_{Xy}(j)|}{J} \ . \quad (5)$$

2.3.6 RF with Shapley-Value based Feature Selection (RFSV) and Personalized Recommendations

To select the best features for BP prediction, we first train a RF model using all features of training samples and calculate the feature importance $\bar{I}_{Xy}$ for all features. Based on $\bar{I}_{Xy}$, we select a subset of features with higher feature importance. To decide the selection ratio of total features, we compare the performance between different ratios. Fig. 2.4 shows the BP prediction performance, measured by mean absolute error (MAE) of the final RF models trained with features under different selection ratios. We can observe that the ratio of 0.5 performs the best in terms of MAE. Based on the empirical results, we select 0.5 as the ratio of feature selection. The resulting BP prediction model is the RF model re-trained with only the selected features based on $\bar{I}_{Xy}$.

In addition to prediction performance, Shapley value suggests how each feature contributes to the deviation of BP prediction from the average BP prediction among the dataset. Therefore, we select the top three lifestyle factors with the highest Shapley importance for each person as his/her personalized and precise recommendation. Note that in the recommendation, we exclude measure_time, heart rate and BP time-series features derived in Sec. 2.3.2, even if they are selected as the top factors. The rationale is that those factors are not actionable for subjects although they might contribute to BP prediction.

2.4 Results and Discussion

In this section, we will first discuss the experiment settings. We will present the results obtained by using the proposed RFSV and compare the results with existing ML models. Secondly, we will validate the effectiveness of personalized and precise recommendations of lifestyle factors generated by our BP model using RFSV.

2.4.1 Experiment setting

Of the 25 subjects, we sorted out the 13 subjects to train and evaluate the BP-lifestyle model based on the quality, length, and availability of their data. Each person's model is trained with only his/her data. The other 12 subjects had sufficient BP data but less than 45 effective days of continuous lifestyle data. However, their BP data was included in Sec. 2.4.3 as in the control group to evaluate the effectiveness of personalized recommendation. We implement and evaluate our proposed methods in the Python environment. We also use the Tree SHAP [69], Scikit-learn library [34], Keras [35] and Auto.arima [70] to implement RFSV and other ML models. MAE, root mean square error (RMSE), mean absolute percentage error (MAPE) and Coefficient of determination (R^2) are calculated and used as our evaluation metrics. Their definitions are as follows:

$$MAE = \frac{\sum_{i=1}^{n}|\widehat{BP}^i - BP^i|}{n} \tag{6}$$

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}\left(\widehat{BP}^i - BP^i\right)^2}{n}} \tag{7}$$

$$MAPE = \frac{nMAE}{\sum_{i=1}^{n}|BP^i|} \times 100\% \tag{8}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}\left(\widehat{BP}^i - \overline{BP}^i\right)^2}{\sum_{i=1}^{n}\left(BP^i - \overline{BP}^i\right)^2} \tag{9}$$

where $\widehat{BP}^i$ is the ith prediction of BP made by trained models and BP^i is the actual value of the ith BP.

We use 5-fold cross-validation to randomly split our dataset into training (80%), and test (20%) sets five times and average the prediction results. To show the effectiveness of RFSV, we compare the predictive performance with several representative ML algorithms referenced earlier in Sec. 2.2, including Support Vector Machine (SVM), Gradient Boosting Decision Trees (GBDT), Long Short-Term Memory (LSTM), and ARIMA. We also compare our performance against a regressor (termed as SimpleMean), which always predicts the mean of the training data. The rationale is that the prediction error may largely depend on the underlying BP fluctuation of the subject. By comparing SimpleMean and other ML algorithms, we can exclude the dependency of the underlying fluctuation. In ARIMA, we take SBP_arima and DBP_arima, the ARIMA forecasts introduced in Sec. 2.3.3. For setting details of other models, we set the number of trees to 500 for all RF models. We set the maximum ratio of total features used in each tree as 0.33 and the minimum number of samples to split as 2. For SVM, the RBF kernel is used, and the best γ and C are selected using cross-validation. For GBDT, the number of trees as 500 and the learning rate as 0.05. LSTM was trained using 0.001 and 20 as the learning rate and batch size with Adam optimizer [40]. The total depth of the fully connected layers in LSTM models was set to 4 and the maximum neurons in each layer to 50. We also use early stopping and dropout layers with a dropout rate of 0.2 to avoid overfitting.

2.4.2 BP Prediction using RFSV

The MAE and RMSE of BP prediction of the proposed method and other methods are summarized in Table 2.3. Note that the values in Table 2.3 are the average MAE and RMSE over all the users. As shown in Table 2.3, most of the methods outperform SimpleMean, which suggests

the prediction power of lifestyle factors. The possible reason why LSTM performs the worst of all methods is LSTM may overfit the small training dataset (~180 samples for each user). GBDT and SVM perform similarly while GBDT has a slightly better prediction error. ARIMA is the second-best method based on MAE for SBP and MAE and RMSE for DBP. The possible reason is the temporal dependency in historical BP contains enough information, that with proper modeling, it outperforms ML models only based on lifestyle factors. However, worse RMSE for SBP using ARIMA may suggest overfitting to the SBP series. Among all methods, our proposed RFSV model achieves the lowest prediction error in terms of MAE, MAPE and RMSE. Our proposed RFSV performs better than ARIMA by 10.1% and 6.2% in terms of MAE for SBP and DBP; 10.9% and 7.5% in terms of MAPE for SBP and DBP; 14.4% and 10.4% in terms of RMSE for SBP and DBP, respectively (RMSE of SBP is compared with GBDT). In terms of R^2, RFSV achieve 0.51 and 0.52 for SBP and DBP, which means the most proportion of the variance is explained by RFSV compared to other methods.

We carry out a Paired Student's t-test [71] separately for each subject to assess the statistical significance of the difference in estimation errors between our method RFSV and two methods, ARIMA and GBDT, which achieve the closest performance to our method shown in Table 2.4. The null hypothesis of the Paired Student's t-test is that there is no difference between the performance of two ML models. We then calculate the p-value using the method in [49] for

TABLE 2.3. Prediction Error of RFSV and related models

	SBP				DBP			
	MAE	RMSE	MAPE (%)	R^2	MAE	RMSE	MAPE (%)	R^2
SimpleMean	6.87	10.32	5.46	0.29	4.70	7.86	5.99	0.27
SVM	6.15	9.67	4.87	0.46	4.26	7.21	5.40	0.44
GBDT	6.03	9.64	4.71	0.46	4.11	7.01	5.22	0.45
LSTM	8.21	13.10	6.64	0.21	6.01	10.23	7.54	0.19
ARIMA	5.94	9.98	4.70	0.41	4.05	6.68	5.24	0.47
RFSV	5.34	8.24	4.19	0.51	3.80	6.05	4.83	0.52

TABLE 2.4. MAE in different prediction horizons using RF, RF-ARIMA and RFSV MAE (SBP/DBP)

	Current	12hr	24hr	48hr
RF	6.15/4.26	6.89/5.16	6.45/4.83	6.79/5.01
RF-ARIMA	5.79/3.95	6.61/4.82	6.21/4.48	6.51/4.60
RFSV	5.34/3.80	6.32/4.5	6.01/4.21	6.25/4.38

each subject and compare it with a significance level α, the probability of rejecting the null hypothesis given that it is true ($\alpha = 0.05$ is used in most studies). If the p-value is smaller than α, the null hypothesis is rejected. Therefore, the results statistically provide convincing evidence that two ML models perform differently. For 16 out of the 25 subjects, the performance difference between RFSV and ARIMA is statistically significant at the level $\alpha = 0.05$ for both SBP and DBP. Similarly, for 20 out of the 25 subjects, the performance difference between RFSV and GBDT show statistical significance at the level $\alpha = 0.05$ for both SBP and DBP.

Besides the prediction of current BP, we will discuss the effect of applying ARIMA prediction of BP and Shapley-based feature selection for different prediction time horizons. The BP predictions of current BP (the MAEs in Table 2.3), 12 hours, 24 hours and 48 hours ahead are summarized in Table 2.4, comparing our proposed RFSV with: 1) RF, which does not include SBP_arima and DBP_arima and feature selection, and 2) RF-ARIMA, which includes SBP_arima and DBP_arima but without feature selection. As shown in Table 2.4, we can make the following observations: 1) RFSV consistently gives the best BP prediction, which shows the effectiveness of ARIMA feature extraction and Shapley-based feature selection. 2) For each method, the MAE worsens as the prediction horizon expands, except for 12-hours ahead prediction, which is the

TABLE 2.5. MAEs using different feature selection methods

	No Feature Selection	PCFS	IGFS	RFFI	RFSV
SBP	5.79	5.64	5.71	5.36	5.34
DBP	3.95	3.87	3.92	3.83	3.80

worst performer. The result indicates that the accuracy of the prediction based on lifestyle factors and historical BP decreases with time. The worst performance for 12-hours ahead prediction suggests that the proposed technique may work better when the prediction horizons are multiples of 24 hours.

Finally, we compare the average MAE of RFSV (which uses Shapley value for feature selection) with three other feature selection methods introduced in Sec. 2.3.5, namely Pearson Correlation-based Feature Selection (PCFS), Information Gain-based Feature Selection (IGFS), and Random Forest Feature Importance (RFFI). As shown in Table 2.5, all feature selection methods result in lower MAE than the prediction without feature selection. While RFSV and RFFI perform significantly better than PCFS and IGFS, RFSV has the lowest MAE. We also carry out a Paired Student's t-test to assess the statistical significance of the difference in estimation errors between feature selection methods. Between RFSV and No Feature Selection, 12 out of the 25 subjects show statistical significance at the level alpha = 0.05 for both SBP and DBP. However, only 4 out of the 25 subjects show statistical significance when we compare RFSV and RFFI. This is consistent with the observation that feature selection can reduce the MAE, and RFSV performs just slightly better than RFFI in terms of MAE. We decide to use RFSV because of its lowest MAE and strong interpretability base on Game Theory.

2.4.3 Personalized and Precise Recommendation

In Fig. 2.5, we illustrate the contribution from each feature to an increase (or decrease) in SBP prediction for two subjects using SHAP [41]. Each dot represents the Shapley value for the feature listed on the Y-axis to the BP prediction of a sample. The placement on the X-axis represents the amount of positive/negative contribution to BP prediction. The color represents the actual value of the feature (red is high while blue is low). The feature list is sorted by contribution to the model from most to least. For example, from heart_rate_1 of subject 1 we can observe most blue dots (lower heart rate) are associated with higher BP prediction and most red dots (higher heart rate) are associated with lower BP. On the contrary, heart_rate_1 of subject 2 has the opposite relationship with his/her SBP. Moreover, the order of the top features from the two subjects is very different. The above observations confirm that different lifestyle factors may affect the BP of different individuals differently, with the top factors different for different individuals, and hence

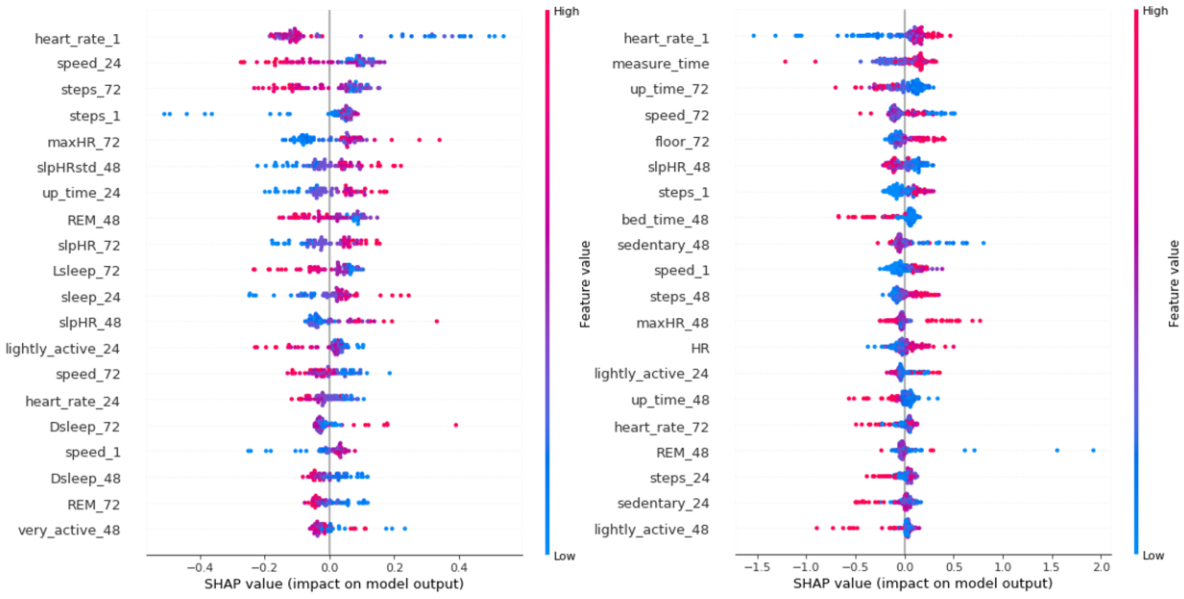

Figure 2.5. Left: SHAP values of features on SBP for subject 1. Right: SHAP values of features on SBP for subject 2.

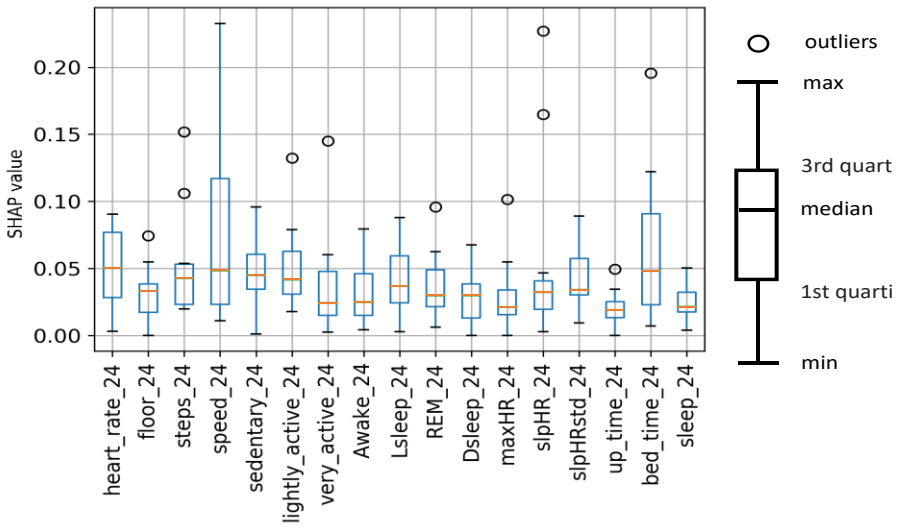

Figure 2.6. Box plot of SHAP values of lifestyle features (over the previous 24 hours)

the motivation to provide personalized recommendations based on his/her data. With high granularity of lifestyle data collected from individuals and interpretation by Shapley values, patients and doctors can understand how lifestyle factors affect BP in a more precise and personalized fashion. In addition to using two subjects as examples to discuss the SHAP results above, we next expand the discussion to all subjects in this study. We first calculate the mean absolute value of SHAP values (which are the dots plotted in Fig. 2.5) of each feature. Based on the mean SHAP values of each subject, we provide a box plot of representative features over the previous 24 hours used in Sec. 2.3.2 to show the minimum, the maximum, the median, and the first and third quartiles of the SHAP values among all subjects, as shown in Fig. 2.6. We can observe that heart_rate24, speed_24, and bed_time_24 (the time when subjects go to sleep) have the highest median SHAP values while sleep_24 (total sleep duration) and up_time_24 (the time when subjects wake up) have the lowest median. Among the 17 features in Fig. 2.6, heart_rate24 has the highest SHAP values in 3 of 25 subjects; speed24 has the highest SHAP values in 7 of 25 subjects and bed_time_24 has the highest SHAP values in 5 of 25 subjects. For the other 10

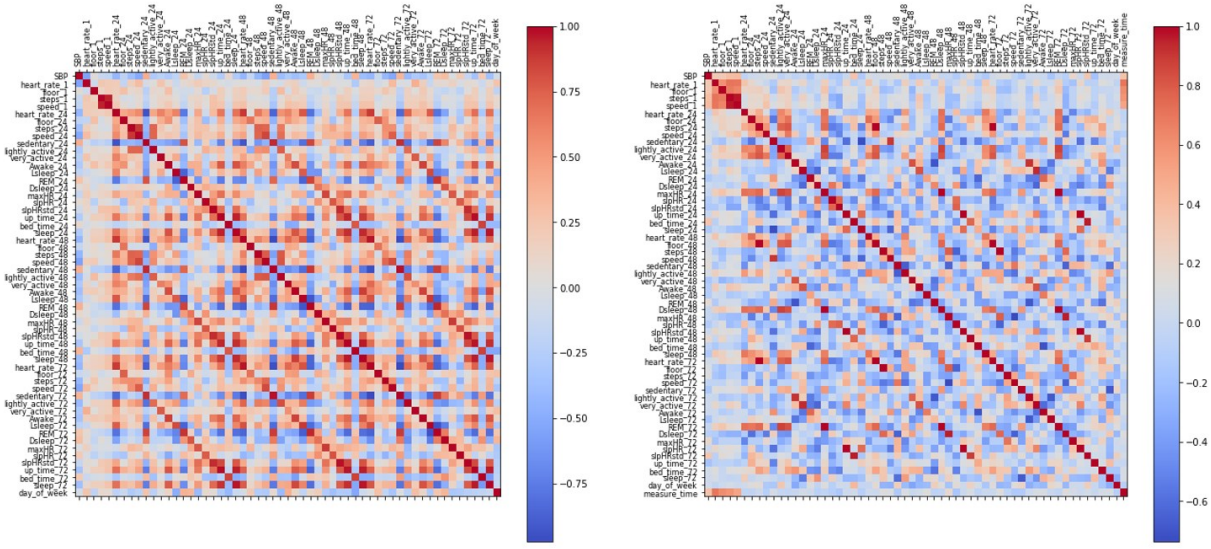

Figure 2.7. Left: Pearson Correlation Heatmap for subject 1. Right: Pearson Correlation Heatmap for subject 2

subjects, their top features are not the three features with the highest median SHAP values shown. The above result validates our motivation to provide recommendations based on each subject's SHAP values. However, the statistical analysis of SHAP values from all subjects may still provide valuable insights for designing population health management solutions.

To further validate the correlation between BP and the features, Fig. 2.7 displays a heatmap of the Pearson correlation between all features and SBP for the two subjects shown in Fig 2.5. For subject 1, the top three factors based on SHAP are heart_rate_1, speed_24 and steps_72, and all of them are negatively correlated to SBP. For subject 2, the top three factors based on SHAP are heart_rate_1 (positive correlation), measure_time (positive correlation) and up_time72 (negative correlation). We can observe the top features selected based on SHAP are largely consistent with the correlation heatmap, in terms of both direction and intensity.

To validate the effectiveness of the lifestyle recommendations suggested by the RFSV model, we randomly selected 6 of the 13 eligible subjects to form the experimental group which would receive personalized recommendations. That is, one month before the end of the study, we

Your BP report from 3/30-7/1

Summary of your SBP/ DBP (254 readings)

average	125/80
lowest	94/57
highest	159/96

Based on your blood pressure and lifestyle data, our analysis shows your blood pressure has the greatest correlation with the following:

- Daily average walking/running speed; increasing walking/running speed may positively influence (reduce) your future BP

The following graph(s) depict your daily blood pressure as compared to the following variable(s):

- Average walking/running speed

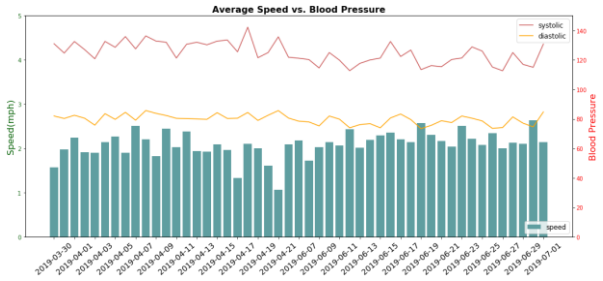

Figure 2.8. Example of personalized recommendation of subject 1

sent each subject in the experimental group an email consisting of 1) basic statistics of their BP during the study period, including the average, minimum and maximum blood pressure during the trial and 2) top lifestyle features for his/her BP prediction model based on Shapley value. The design of recommendation language for each feature is done in consultation with the physician collaborator in our research team. Lastly, we plot the figures which show the daily values of BPs and the corresponding top features to serve as subjects' reference. An example of the recommendation for subject 1 is shown in Fig. 2.8. Although heart_rate_1 is the top feature for subject 1 (Fig. 2.5), it is not an actionable factor. Therefore, we recommend the next top feature, walking/running speed (speed_24), as the top factor. From Fig. 2.5, we can observe most red dots (higher speed) of speed_24 are associated with lower BP, so our personalized recommendations suggest the subject increase his walking/running speed. Following this email feedback, we collect BP data for a month for the experimental group and compare it with the control group.

The other 7 subjects are assigned to be in the control group which did not receive any feedback. As mentioned in Sec. 2.4.1, the other 12 subjects complete BP measurements across the

TABLE 2.6. Recommendations and BP of subjects in the experimental group before and after their personalized recommendation

Subject ID	Before recommendation (SBP, DBP)			Recommendations	After recommendation (SBP, DBP)		
	Mean BP	Max BP	slope		Mean BP	Max BP	slope
1	129, 81	142, 90	-0.12, -0.04	speed, steps	123, 80	139, 85	-0.27, -0.14
2	128, 82	137, 89	-0.04, -0.06	wake up time	120, 75	132, 79	-0.12, -0.21
3	117, 69	134, 79	0.05, 0.03	steps, floors	117, 68	122, 72	-0.08, 0.21
4	134, 83	157, 100	-0.04, -0.03	steps, speed	133, 80	142, 86	-0.27, -0.13
5	127, 82	152, 94	0.03, 0.01	steps, speed, sleep duration	121, 81	127, 85	-0.41, -0.32
6	122, 79	127, 91	0.04, 0.07	wake up time, floors	120, 78	126, 83	-0.08, -0.07

study period, but they did not have enough lifestyle factor data collected from wearables. Those subjects are assigned to the control group since they had the same treatment (not receive feedback) as the 7 subjects in the control group and we only focus on their BP measurements. In Table 2.6, we list the top features (recommendations) of the subjects in the experimental group and their BP statistics before and after receiving the recommendations. For the same type of feature in different time windows, we give the same recommendation without mentioning the time windows. For example, if user 1's top feature is steps_24 and user 2's top feature is steps_48, we will give both users the same recommendation as steps. The rationale is that features extracted based on different time windows may be useful to enhance predictive accuracy, but they are not intuitive for people to follow. From Table 2.5, we can observe that the top features can be very different for different subjects. For example, BP is mainly correlated to activity-related features for some subjects (1, 3, 4), sleep-related features for others (2, 5). To evaluate the change in BP, we calculate the mean and maximum of daily BP in the first week and the last week of the study since BP fluctuates with time and each single measurement may not reflect the actual BP of an individual. Additionally, we use longitudinal linear regression to calculate the linear slope of BP trend before and after receiving the recommendation to further understand their BP changes.

TABLE 2.7. Cohort Statistics and summary of the BP changes in the experimental group and the control group

Measures	Experimental group (n = 6)		Control group (n = 19)	
Age	50.1 +/- 15.0		52.9 +/- 13.1	
Female ratio (%)	33%		37%	
	SBP	DBP	SBP	DBP
Initial BP (Mean +/- SD)	125.4+/-5.9	80.8+/-6.7	127.2+/-7.3	80.0+/-5.9
Mean BP change	-3.8	-2.3	-0.3	-0.9
Subjects with decreasing mean BP (%)	5 (83%)	6 (100%)	9 (47%)	10 (53%)
Max BP change	-10.5	-8.8	-3.3	-2.5
Subjects with decreasing max BP (%)	6 (100%)	6 (100%)	12 (63%)	11 (58%)
Trend slope in the last 30 days	-0.26	-0.13	-0.04	0.01

In Table 2.6, we show that the mean and maximum BP of most subjects in the experimental group improved (decreased) after the recommendation, except for subject 3 whose mean SBP remained the same. The average changes in mean BP for all subjects were -3.8 and -2.3 for SBP and DBP, respectively, and the average changes of maximum BP were -10.5 and -8.8 for SBP and DBP, respectively. For the slope of BP trend, we observe the BP trend turns from slightly increasing to decreasing for subjects 5 and 6 and from slightly decreasing to a steeper decreasing trend for subjects 1, 2 and 4. The exception is the DBP trend for subject 3. The change in BP varies significantly among the subjects where the change ranges from -10 to 0 points for mean BP and -25 to -1 points for maximum BP, from the first week to the last week of the study. One possible reason for such variation is that the stableness of BP and its correlation to lifestyle factors may differ among people. Finally, we discuss subject 3, whose BP remained mostly unchanged during the study. Although his BP records satisfied the initial subject screening criteria (SBP between 120-140 and DBP under 90), his measured BP was mostly recorded to be under 120/80 during the study. Therefore, lifestyle recommendations may have less effect on his BP which is already in a normal range.

Next, we compare the change of BP between the control group and the experimental group, as shown in Table 2.7. For consistency, we use the same method to derive the mean BP, max BP, and BP trend slopes for subjects in the control group. The average mean and max BPs of subjects in both groups decreased by the end of the study, suggesting a positive effect of awareness through only using the wearable device and measuring BP daily. However, the decreases in mean BPs (-3.8 and -2.3 for SBP and DBP) and max BPs (-10.5 and -8.8 for SBP and DBP) for the subjects in the experimental group are meaningfully greater than subjects in the control group, which are (-0.3, -0.9) and (-3.3, -2.5) for mean BPs and max BPs respectively. A two-sided Student's t-test [49] is done to compare the reduction of mean BPs and max BPs for the two groups. The null hypothesis is that the mean BPs and the max BPs for the two groups have no difference. The p-values for mean BPs are 0.15 and 0.22 for SBP and DBP, and the p-values for max BPs are 0.07 and 0.05 for SBP and DBP respectively. The result does not reject the null hypothesis for significance level $\alpha = 0.05$ except for max DBP. One possible reason for higher p-values is the impact of random error due to the smaller sample size, especially for the experimental group. Since the average changes cannot fully represent the individual effect, we also calculate the ratio of subjects in each group who improved (reduced) their mean and max BP. In the experimental group, 83% (5 of 6 subjects) and 100% (6 of 6 subjects) improved their mean SBP and DBP, respectively, compared to only 47% (9 of 19 subjects) and 53% (10 of 19 subjects) of the control group. Similarly, all subjects in the experimental group improved their max SBP and DBP, respectively, compared to only 63% and 58% of the subjects in the control group, respectively. Finally, in the last 30 days, the BP trend slope of subjects in the control group is relatively flat, while a decreasing trend is observed in the experimental group. In summary, subjects who received personalized recommendations about their lifestyle factors and BP were more likely to have demonstrated a

decrease in their mean and max BP by the end of the study. Furthermore, the magnitude of this decrease in BP was greater in this experimental group compared to the control group.

Limitations to this experiment include, by definition, the relatively small number of subjects who had complete BP and lifestyle data for analysis by study end (13 of the original 36 subjects enrolled), due to early participant drop-off and missing data. The lower ratio of eligible subjects demonstrates the universal challenge of keeping patients engaged in their health and the need to create more automated and convenient means of remote health monitoring. In addition, while this snapshot in time showed promising results, the lasting effect of any intervention is best demonstrated over longer study periods. In summary, while the results presented above are encouraging, future studies with a greater number of participants monitored over a longer duration are needed.

2.5 Conclusion

In this paper, we investigate the personal effect of lifestyle factors on BP using data collected from wearables and home BP monitors, on 25 subjects in a clinical trial conducted in collaboration with UC San Diego Health and Altman Clinical and Translational Research Institute. Our proposed approach includes developing a personalized BP model for each individual using BP and lifestyle data for that individual, identify the most important lifestyle attributes that impact an individual's BP trend and provide precise recommendations to improve the individual's BP. Specifically, we propose a RFSV personalized model which we demonstrate can outperform other existing ML techniques in terms of prediction accuracy - by 10.1% and 6.2% in terms of MAE for SBP and DBP; 10.9% and 7.5% in terms of MAPE for SBP and DBP; 14.4% and 10.4% in terms of RMSE, for SBP and DBP respectively, and also achieving the highest R^2. We also propose a method based on Shapley value to identify the top features which affect the BP for each individual

and provide personalized recommendations. Using a randomized control experiment, we show that significant improvement in BP can be achieved with personalized lifestyle recommendations. After receiving recommendations, the subjects in the experimental group decreased their BPs by 3.8 and 2.3 for systolic and diastolic BP, compared to a decrease of 0.3 and 0.9 for the subjects who did not receive recommendations.

In the next chapter, we present the previous research [110] in green communication, which utilizes renewable energy to reduce grid energy consumption of base station (BSs). We propose a low-complexity online control scheme based on Lyapunov optimization framework. The proposed technique can provide asymptotically optimal performance bound without requiring the stochastic distribution information of RE arrival and channel state condition.

Chapter 2, in part, is from the material as it appears in proceedings of IEEE Journal of Translational Engineering in Health and Medicine 2021, Po-Han Chiang, Melissa Wong and Sujit Dey. The dissertation author was the primary investigator and author of the papers.

Chapter 3

Optimal Use of Harvested Solar, Hybrid Storage and Base Station Resources for Green Cellular Networks

3.1 Introduction

The proliferation of mobile traffic will lead to drastically increasing energy consumption in future cellular networks. The total energy consumption and carbon dioxide equivalent (CO_{2e}) emission of mobile cellular networks globally for 2020 has been estimated to more than 120TWh and 179 million tons (Mt) [72]. According to [73], base stations (BSs) consume 80% of the total power in cellular networks. Therefore, the need to reduce the grid power consumption of the BSs is crucial for energy-efficient cellular networks. There has been significant research of energy-efficient BSs [74], ranging from physical layer approaches involving power and spectral resource allocation [75, 76] and RF chain switching [77] to network level techniques wherein active BS selection and user association [78] is performed. In addition to the above techniques focusing on reduction of BS energy consumption, powering BSs with renewable energy (RE), which may increase BS energy consumption while reducing grid energy consumption of BSs, is a promising solution [79, 80]. Though the last few years have seen tremendous growth in the use of RE in several commercial and industrial sectors, its adoption in cellular networks has been limited. The primary challenge in utilizing RE energy for BSs is the highly intermittent, unreliable and variable nature of RE availability across time and space, leading to mismatch between RE generation and loads [81]. In this paper, we will mainly focus on solar energy, however, our insights and proposed approach will apply to wind and other intermittent RE sources.

One approach to overcome the challenges of intermittency and variability of RE availability is the use of high capacity batteries [82]. However, high CAPEX of such systems limits the economic viability for operators and growth of RE-powered BSs. Therefore, using RE in conjunction with grid energy (i.e., hybrid energy supply (HES)) is a viable approach to save grid energy [83-87]. The other challenge is the non-ideal behavior of batteries. For example, lead-acid batteries are widely used in telecommunication power systems as backup power supply and energy storage. From [82], the efficiency of lead-acid battery depends on the charging/discharging rate and state of charge and is lower than 75%. Batteries are assumed to be ideal in most of the previous green communication studies [83-86]. In this paper, we will incorporate the non-ideal characteristic of batteries and propose the corresponding charging and discharging decision.

In our preliminary work [88], we propose a RE-aware technique to adapt time resource of the BS and data buffers at user equipments (UEs) depending on the amount of harvested RE at the BS, channel condition and buffer level of UEs. We have demonstrated that the technique increases the utilization of RE and hence decreases the grid energy consumption without requiring energy storage. However, there are two main limitations: 1) Excessive solar energy is wasted if the harvested solar energy is larger than maximum power consumption of BS or data buffer of all users is full, 2) depletion of UE data buffer may limit the ability to adapt the BS resource and hence grid energy saving. Therefore, we propose the use of energy storage to complement the RE-aware BS resource allocation technique in this work.

In this work, our objective is to minimize grid energy consumption of BSs as well as maximize QoS of users while optimally utilizing harvested solar power. The focus of the proposed approach will be to modulate BS energy consumption with the use of both energy storage at the BS and data storage of the UEs in order to mitigate the mismatch between harvested solar energy

and BS energy consumption. Though the proposed approach is applicable to any application that utilizes data storage of users, we consider the problem for video download/streaming, which will play a large proportion of mobile traffic and hence the BS energy consumption.

The rest of the chapter is organized as follows. In section 3.2, we will investigate the related work in wireless cellular communication using renewable energy. In section 3.3, the system model is described and the problem formulation is presented. We present the proposed Lyapunov-based solar power-aware BS resource (L-SPAR) allocation methodology and algorithm in section 3.4. In section 3.5, the feasibility and performance bounds are derived. The performance of the proposed algorithm is evaluated via simulation in section 3.6. Finally, we conclude the paper in section 3.7.

3.2 Related Work

In [83, 87], the authors focus on a single BS-UE link of the HES BS and propose an optimal BS resource allocation technique using the two-stage water filling (WF) policy to minimize grid energy consumption. Unlike the above techniques which consider only a point-to-point link with fixed transmission rate requirement, our approach adapts transmission rate and BS resource allocation of multiple BS-UE links. In [89, 90], the authors propose an energy cooperation scheme where BSs trade and transfer energy via smart grid based on RE availability and traffic load. However, the technique requires BSs to be fully connected with two-way power grid to transfer energy while we focus on shaping the power consumption of the BS to realize grid energy reduction. BSs with Non-direct energy transfer schemes are proposed to minimize the grid energy cost either by traffic offload [84] or cognitive spectrum sharing [91]. However, these approaches require frequent inter-cell coordination to adapt cell size or spectrum sensing while our technique is applicable to a single cell and do not require to change cell size. Approaches have also been developed for cellular networks with HES which also use battery storage [83-86]; however, the

above approaches assume the battery to be ideal, while our work does consider battery imperfection and shows it to have significant impact in HES communication system. The authors in [87, 92] consider non-ideal behavior of batteries with a threshold-based charging/discharging strategy, but their research effort focuses on cellular network throughput optimization instead of energy saving.

The other key challenge of RE-powered communication systems design is how to efficiently utilize given channel side information (CSI) and energy side information (ESI). In conventional communication systems, BS power consumption is minimized thorough optimizing energy efficiency (as measured in bits/J) with given CSI. In [6], BS power consumption in multi-user OFDMA systems is proposed as a function of the transmission power, the signal processing power and the fixed circuit power to optimize system energy efficiency, which is ratio of the achieved sum throughput and the energy consumed. However, in RE-powered systems, ESI should be also considered, especially when the charge and discharge behavior of batteries is non-ideal. Markov decision processes (MDP) are widely applied to relevant online optimization problems with statistical knowledge of CSI and ESI [93]. However, MDP suffers from the curse of dimensionality with exponential dimension of system states. Lyapunov optimization technique, which has advantages such as low-complexity, computable theoretical bounds and little requirement of prior statistical knowledge, was first applied in RE-powered communication systems in [94]. Lyapunov optimization framework is applied to solve subcarrier power allocation in [95] and user association problem in [86] to minimize power consumption given CSI and ESI as stochastic processes. However, they do not consider the non-ideal battery behavior which will significantly affect the dynamic of batteries and the resulting optimal solution. In [96], the authors maximize the network utility in multi-hop wireless networks with imperfect batteries and limited

RE. The above works do not consider utilizing the data storage of UEs and adapting the transmission accordingly, which will further enhance the energy and QoS performance.

In this work, we focus on mobile video, which is estimated to contribute to over two-thirds of mobile data traffic by 2018 [97]. Hence it is critical to reduce the BS energy consumption while satisfying the QoS requirements of users during video download/streaming. Techniques which adapt streaming quality like dynamic adaptive streaming over HTTP (DASH) [98] have been studied and applied to shape the download traffic. Most previous work addressing energy consumption during video download/streaming [99, 100] focuses on power saving of mobile devices by shaping the traffic transmitted to users and extending the periods of no transmission or idle periods of mobile devices, but do not address reducing BS energy consumption during video download. The authors in [101] propose to schedule transmission given the channel state predictions for wireless video download and adapt the video bitrate to minimize BS energy consumption while the use of RE and UE buffer is not considered. The authors in [102] propose to adapt compression ratio of video traffic to the amount of harvested RE. However, the technique requires perfect distribution information of harvested RE and traffic demand of UEs.

3.3 System Model and Problem Formulation

In this section, we will first present the system model comprising of network, channel, traffic demand, BS energy consumption, energy storage and data buffer models. Then, we formulate the weighted sum optimization problem to address both grid energy consumption and QoS of users with constraints of UE data buffer, energy storage and BS utilization. For ease of reference, we list the key notations of our system model in Table 3.1.

3.3.1 Network and Channel Model

Consider downlink communication in a OFDMA cellular network system with a set of BSs $\boldsymbol{B}$, each with subcarriers set $\boldsymbol{K} = \{1,2 \dots K\}$. Without loss of generality, we will consider one BS, $b \in B$ and its associated set of users $I = \{1,2 \dots I\}$. For the sake of notational brevity, henceforth, we will drop the subscript b from the BS related variables. Also, we will use the terms energy storage and battery interchangeably. Fig. 1 shows a single BS, with energy flows from different energy sources (PV panel, grid and battery) to different energy sinks (the BS and the battery), and data links between the BS and I users. Transmission time is equally divided into n time slots of duration λ, which is normalized to one in our paper for ease of analysis. We assume perfect channel state estimation including path loss, multi-path fading, shadowing and other factors if any at both the transmitters (BSs) and the receivers (UEs). Each subcarrier $k \in K$ can only be used by one user

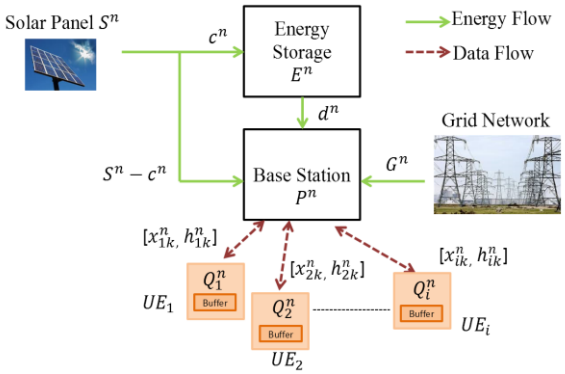

Figure 3.1. Architecture of proposed system

and subcarrier allocation is performed at the beginning of each time slot. We denote subcarrier allocation by the binary matrix $X^n = \{x_{ik}^n\}_{i\in I,k\in K}$ where x_{ik}^n is defined as

$$x_{ik}^n = \begin{cases} 1, \text{if subcarrier } k \text{ is assigned to user } i \text{ in } n^{th} \text{ slot} \\ 0, \text{otherwise} \end{cases} \quad (1)$$

Let $H^n = \{h_{ik}^n\}_{i\in \boldsymbol{I},k\in \boldsymbol{K}}$ be the channel gain matrix where h_{ik}^n is the channel gain of user i on subcarrier k. The channel gain h_{ik}^n of each BS-UE link is assumed to be statistically independent and identically distributed (i.i.d) and remains constant during each slot. We denote P_k^n the downlink transmit power allocated on each subcarrier k of the BS in the n^{th} slot and is assumed to be fixed within each time slot. Since joint optimization of subcarrier and power allocation is proved to be NP-hard [33] and there is no standard method for optimal solution, our proposed technique only decides subcarrier allocation X^n and it can work with any existing power allocation techniques. Let r_{ik}^n denote the achievable transmission rate from the BS to user i on subcarrier k in the n^{th} slot and is given by

$$r_{ik}^n = W\log_2\left(1 + \frac{|h_{ik}^n|^2 P_k^n}{\sigma N_0}\right) \quad (2)$$

TABLE 3.1: Summary of Key Notations

Notation	Description
$\boldsymbol{K}$	Index set of subcarriers of BS b
$\boldsymbol{I}$	Index set of the UEs associated with b
x_{ik}^n	Subcarrier allocation indicator in n^{th} slot
h_{ik}^n	Channel gain of user i on subcarrier k
r_{ik}^n	Achievable transmission rate of user i on subcarrier k
R_i^n	Total achievable transmission rate of user i
Q_i^n	Buffer level of user i
δ_i^n	Buffer consumption rate of user i
$U_i(\delta)$	Utility function of user i
P^n	Power consumption of the BS in n^{th} slot
E^n	Battery level in n^{th} slot
c^n	Charging energy in n^{th} slot
d^n	Discharging energy in n^{th} slot
θ_0	Perturbation parameter of battery level
θ_i	Perturbation parameter of buffer level of user i

where W, N_0 and σ are the bandwidth of each subcarrier, noise power density and the nominal spectral efficiency in (bit/s)/Hz respectively. Note that r_{ik}^n is clipped within $[r^{min}, r^{max}]$ to account for practical modulation orders. Let R_i^n denote the total achievable transmission rate from the BS to user i over all subcarriers and is given by

$$R_i^n = \sum_{k=1}^{K} x_{ik}^n r_{ik}^n \tag{3}$$

3.3.2 UE QoS Model

As discussed earlier, we focus on mobile video download/streaming. Video contents can be transmitted to UEs and stored at the UEs' buffer, and then the transmission can be paused without stalling if there is enough data for playing. Therefore, we are interested in the data buffer level of the users. We define the buffer level Q_i^n of user i in n^{th} slot as the sum of the buffer level in $n-1^{th}$ slot and the data accumulated in the buffer λR_i^n subtracted by the data used for video playback $\lambda \delta_i^n$. Note that λ is assumed to be 1 in our case. Therefore, buffer level in the n^{th} slot Q_i^n is given by

$$Q_i^{n+1} = Q_i^n + R_i^n - \delta_i^n, \forall i, \forall n, \tag{4}$$

We assume $Q_i^0 = 0, \forall i$ and δ_i^n is the buffer consumption rate satisfying

$$0 \leq \delta_i^n \leq \delta^{max} \tag{5}$$

with a finite δ^{max} at all time. For smooth playback δ_i^n and R_i^n should be decided in a manner that the video buffer does not overflow or underflow. Hence for each user we have

$$0 \leq Q_i^n \leq Q_i^{max}, \forall i, \forall n, \tag{6}$$

where $Q_i^{max} \in (0, \infty)$ is the maximum number of bits that can be stored at user i, which may depend on the video client and network policy. In DASH [24], the video content is segmented into

small HTTP-based files. Video segments are then pre-encoded in multiple versions with their "quality levels," specifying specific video bit rate and resolution [98, 105]. During video download, the streaming adaptation engine at UE i selects the quality level of requested video segments based on throughput estimation and media playout conditions [105]. The better QoS UEs have, the higher the buffer consumption rate. To provide a measure of user QoS during video download, we denote utility function[1] $U_i(\delta_i^n)$ for each user i. Every $U_i(\delta)$ is assumed to be positive, increasing, strictly concave and differentiable for $\delta \in [0, \delta^{max}]$ [25, 35]. For convenience, we denote β_i as the maximum first derivative of $U_i(\delta)$ according to the property of strictly concave functions.

3.3.3 Base Station Power Consumption Model

According to [75, 76, 81], the BS power consumption can be modeled as a constant power term plus a radio frequency (RF) related power term. Secondly, the RF related power term can be modeled as a linear function of the number of active subcarriers. The total power consumption of the BS in time slot n is:

$$P^n = \sum_{i=1}^{I} \sum_{k=1}^{K} x_{ik}^n \left(\frac{P_{tx}}{\Delta} + P_{sp}\right) + P_0 \tag{7}$$

where P_{tx} is the constant transmit power level per subcarrier and Δ is power amplifier efficiency. P_{sp} denotes the signal processing power per subcarrier and P_0 denotes fixed circuit power consumption of the BS such as the baseband processor, the converter and the cooling system. Finally, the energy consumption of the BS in time slot n is $\lambda P^n$and again λ is omitted since λ is assumed to be 1 in our case.

[1] Network utility is first proposed by authors in [35] as a measure of user satisfaction when getting a data rate from the network. Increasing, differentiable, and concave utility functions $U(\cdot)$ (e.g., proportional-fair utility functions) are widely adopted in network utility maximization (NUM) problems

3.3.4 Energy Storage

Let the amount of harvested solar at the beginning of the n^{th} slot be S^n. We assume that S^n is immediately available for use in n^{th} slot and takes values in some finite set $S^n \in [0, S^{max}]$ and there exists $S^{max} < \infty$. We also assume that S^n is i.i.d. among different time slots. Although the i.i.d. process cannot fully represent the non-linear and non-stationary solar arrival, it captures the intermittent nature of solar and has been widely adopted in previous studies [86, 93, 94]. The BS stores the harvested solar in the battery and let $E^{max} \in (0, \infty)$ denote the battery capacity. At the beginning of the n^{th} time slot, the transmitter harvests and stores the c^n units of energy. It then draws d^n units of energy from the battery to power the BS. We assume $E^0 = 0$ and model the battery level E^n as

$$E^{n+1} = \varphi E^n + \eta c^n - d^n, \forall n, \tag{8}$$

where c^n and d^n are two non-negative numbers denoting the amount of energy used to charge and discharge the battery in the n^{th} slot respectively. To characterize the imperfection of batteries, we firstly use $\varphi \in (0,1)$ as storage efficiency. This implies that during each time slot, $(1 - \varphi)$ portion of the energy stored in the battery will be lost due to energy dissipation. We use $\eta \in (0,1)$ to denote the charging efficiency of the battery. When c^n units of energy are used for charging, only ηc^n can be stored in the battery for future use and $(1 - \eta)c^n$ is lost due to charging loss[2].

We assume that $E^0 = 0$ and E^n is constrained by energy causality and limited capacity of the battery. Hence, E^n should satisfy

$$0 \leq E^n \leq E^{max}, \forall n. \tag{9}$$

As show in Fig. 1, the grid energy consumption G^n in the n^{th} slot is given by the BS energy

[2] Practically, energy loss occurs during both charge and discharge, and the efficiency depends on factors such as temperature, charging/discharging rate and battery level. For simplicity, these two losses are combined into one and the efficiency is assumed fixed in this work.

consumption P^n subtracted the energy drawn from the battery d^n and the portion of solar energy $S^n - c^n$

$$G^n = P^n - d^n - S^n + c^n \geq 0, \forall n. \tag{10}$$

Note that G^n can never be negative, i.e., there is no transfer of energy back to the grid from the BS for general power grids. c^n is constrained by

$$0 \leq c^n \leq S^n, \forall n, \tag{11}$$

Therefore, d^n is constrained by

$$0 \leq d^n \leq P^n, \forall n. \tag{12}$$

Problem Formulation

To minimize grid energy consumption while maximizing the utility of users, the weighted sum of the above two objectives is used as our objective. Given the solar energy S^n and the channel conditions $\boldsymbol{H}^n$, the buffer availability of each user Q_i^n and the battery level E^n, our objective is to determine subcarriers allocation $\boldsymbol{X^n}$, the buffer consumption rate δ_i^n, charging energy c^n and discharging energy d^n in each time slot to optimize the objective function while satisfying the buffer constraints of users, the battery constraint and the BS utilization constraint. Therefore, the optimization problem **P1** can be formulated as

$$\textbf{P1:} \min_{\boldsymbol{X}^n, \delta_i^n, c^n, d^n} \lim_{N \to +\infty} \frac{1}{N} \sum_{n=0}^{N-1} \mathbb{E}[w_G G^n - w_I \sum_{i=1}^{I} U_i(\delta_i^n)] \tag{13}$$

$$s.t. \quad \text{(5), (6), (9)-(12)}$$

$$\sum_{i=1}^{I} x_{ik}^n \leq 1, \ \forall i, \forall n, \forall k, \tag{14}$$

Constraints (14) states that each subcarrier is exclusively assigned to a single user. Note that since our objective is to maximize aggregate utility of users, we will take negative terms for all $U_i(\delta_i^n)$ in the objective function. Let w_G and w_I denote the weights of the grid energy consumption and aggregate user utility respectively. By properly adjusting w_G and w_I, solving **P1** can effectively

minimize grid power consumption with any QoS requirement, depending on the network policy.

The problem **P1** is a stochastic optimization problem. As shown in (6) and (9), the feasible actions set (charging, discharging, the buffer consumption rate and BS resource allocation) is confined by the current UE buffer level and the battery level. Also, the state transition can be described as a function of actions $\{\boldsymbol{X}^n, \delta_i^n, c^n, d^n\}$ and their states $\{E^n, Q_i^n\}$ in the previous time slot given the probability distribution of the solar energy S^n and the channel conditions $\boldsymbol{H}^n$. Such problems can be modeled by MDP and theoretically solved by linear programming (LP) or dynamic programming (DP) techniques [83,87]. However, the performance of such techniques depends on accurate statistical estimation of solar arrival and channel condition. Furthermore, the offline solution provided by MDP requires exponential number of states to characterize the system, so the process is practically infeasible due to computation complexity. For example, if we have J states of solar arrival, M states of channel gain and L states of buffer level for each user i and T states of battery levels, we need to solve the MDP problem with JM^iL^iT possible states.

Inspired by Lyapunov optimization framework developed in recent works [94, 103], we will next present a low-complexity online L-SPAR algorithm with the following advantages:

• The proposed algorithm provides an online solution requiring no prior knowledge of the probability distribution of the wireless channel or solar arrival processes.

• The proposed algorithm minimizes objective function considering only the current data buffer and the energy storage state, which greatly reduces the complexity that standard MDP/LP solutions would have faced to solve **P1**.

• Although the proposed algorithm does not result in exact optimal solution, the performance can achieve arbitrarily close to the optimal solution by adjusting the penalty parameter [94] in the Lyapunov optimization framework which will be discussed in Section 3.4.

3.4 L-SPAR Methodology and Algorithm

In this section, we will first describe the proposed Lyapunov optimization framework and propose a per-time slot problem **P2**. To solve **P2**, we then propose an online algorithm which determines the BS subcarrier allocation, the amount of energy charging/discharging the battery, and buffer consumption rate of users. In section 3.5, we will show that the proposed algorithm provides a feasible and asymptotically optimal solution for **P1**.

Per-time Slot Problem and L-SPAR Algorithm

In constraints (6) and (9), there exists time-dependent coupling between the state of battery and data buffer and the decision of charging/discharging, BS resource allocation and buffer consumption rate across time slots, which makes the optimization challenging. The principle we apply Lyapunov optimization here is to decouple such dependency by transforming such constraints of the battery level E^n and the data buffer level Q_i^n into a set of virtual queues. Based on Lyapunov optimization framework, the objective function in **P1** is defined as penalty function. By greedily minimizing a weighted function of Lyapunov drift, which is the sum of the squares of the current queue backlogs, and the penalty function, the objective function can be optimized with the long-term average constraints satisfied. Traditional Lyapunov optimization can only guarantee to satisfy long-term averaged constraints. To ensure deterministic bounds on all queue sizes derived from (6) and (9), we use the similar method as in [103] to introduce perturbation parameter $\boldsymbol{\theta} = \{\theta_0, \theta_1, \ldots \theta_I\}$ and define the virtual queues, which represent the shifted version of original battery level E^n and data buffer level Q_i^n. The physical meaning of $\boldsymbol{\theta}$ is the convergence value of data and batteries buffer and was chosen carefully to satisfy the original queue constraints

$$\tilde{E}^n = E^n - \theta_0, \tag{15}$$

$$\tilde{Q}_i^n = Q_i^n - \theta_i \ , \ i = 1,2,..,I, \tag{16}$$

where

$$w_G V + \frac{P^{\max}}{\varphi} \leq \theta_0 \leq \frac{w_G V}{\eta} + \frac{E^{\max} - S^{\max}}{\varphi}, \tag{17}$$

$$w_I V \beta_i + \delta^{\max} \leq \theta_i, i = 1,2,..,I, \tag{18}$$

where V denotes the non-negative weight parameter in Lyapunov optimization where larger V will place more emphasis on penalty minimization over the queue stability. Note that although the battery and the data buffer levels are always non-negative according to constraints (6) and (9), the virtual queues $\tilde{E}^n$ and $\tilde{Q}_i^n$ can be negative.

We now define the per-time slot problem **P2**, which minimizes the weighted sum of the drift of virtual queues and the penalty function with all constraints except (6) and (9). We will later show that (6) and (9) are indeed satisfied by L-SPAR in section 3.5.

$$\mathbf{P2}: \min_{\boldsymbol{X}^n, \delta_i^n, c^n, d^n} \sum_{i=1}^{I} \tilde{Q}_i^n \left(\sum_{k=1}^{K} x_{ik}^n r_{ik}^n - \delta_i^n\right) + \tilde{E}^n [\eta c^n - d^n - (\tilde{E}^n + \theta_0)(1-\varphi)] +$$

$$V[w_G(P^n - S^n + c^n - d^n) - w_I \sum_{i=1}^{I} U(\delta_i^n) \tag{19}$$

$$s.t. \ (5), (10) - (12), (14)$$

After integrating and generalizing Lyapunov optimization framework to propose **P2,** we present the online L-SPAR algorithm. The objective of L-SPAR is to stabilize the battery and data buffer levels around the perturbed level $\boldsymbol{\theta}$ and meanwhile minimize the penalty function. We assume that I users are scheduled in each time slot and the channel state information and buffer level

Algorithm 1 L-SPAR Algorithm

Initialization: Choose a pair of $(\boldsymbol{\theta}, V)$ which satisfies (17) - (18).

1: At the beginning of each time slot n, obtain solar energy S^n, virtual battery level queue $\tilde{E}^n$, channel gain $\boldsymbol{H}^n$ and virtual buffer level queue $\tilde{Q}_i^n \ \forall i \in \boldsymbol{I}$.

2: Decide optimal action set $\{\boldsymbol{X}^{n*}, \delta_i^{n*}, c^{n*}, d^{n*}\}$ by solving **P2**

3: Update $\tilde{E}^n$ and $\tilde{Q}_i^n$ according to (4), (8), (15), (16)

4: Set $n = n + 1$

information of users are periodically reported to the BS using Channel Quality Indicator (CQI) and mechanisms similar to Buffer Status Report (BSR) during uploading as in 3GPP specification [36]. Based on the above information and the amount of generated solar energy and the current battery level, the BS will run L-SPAR in every slot. In other words, given a pair of $(\boldsymbol{\theta}, V)$ and by observing the current state of random processes $\{S^n, \boldsymbol{H}^n\}$ and queues $\{\tilde{E}^n, \tilde{Q}_i^n\}$, L-SPAR will determine an optimal action set $\{\boldsymbol{X}^{n*}, \delta_i^n, c^{n*}, d^{n*}\}$ as a solution for **P2**. In the next section, we will focus on solving the per-time slot problem.

Solving the per-time slot problem

Next, we will focus on solving **P2**. After rearranging **P2** and using Eq. (8), the BS energy consumption model, the objective function in (19) can be written as

$$\underset{\boldsymbol{X}^n, \delta_i^n, c^n, d^n}{\text{Min}} \sum_{i=1}^{I} \tilde{Q}_i^n \left(\sum_{k=1}^{K} x_{ik}^n r_{ik}^n\right) - \sum_{i=1}^{I}\left[\tilde{Q}_i^n \delta_i^n + w_I V U(\delta_i^n)\right] + w_G V \sum_{i=1}^{I}\sum_{k=1}^{K} x_{ik}^n \left(\frac{P_{tx}}{\Delta} + P_{sp}\right) + c^n(\eta \tilde{E}^n + w_G V) - d^n\left(\tilde{E}^n + w_G V\right) + C$$

where C represents the constant term $\tilde{E}^n\left(\tilde{E}^n + \theta_0\right)(1-\varphi) + w_G V(P_0 - S^n)$ in n^{th} slot, which can be omitted in the optimization process. We will decouple the problem into three parts: 1) Charge and discharge of the battery, 2) BS resource allocation, 3) UE buffer consumption rate.

Charge and discharge of the battery: To decide c^{n*} and d^{n*}, we first solve the problem as a simple threshold-based structure

$$\min_{c^n, d^n} c^n(\eta \tilde{E}^n + w_G V) - d^n\left(\tilde{E}^n + w_G V\right) \tag{20}$$

$$s.t.\ \text{(10), (11), (12).}$$

Case 1: $\eta \tilde{E}^n + w_G V > 0; -\left(\tilde{E}^n + w_G V\right) \leq 0$. L-SPAR will discharge as much as possible and will not charge. Since d^{n*} has to satisfy $P^n - S^n + c^n - d^n \geq 0$ and $c^{n*} = 0$ in this case, we have $d^{n*} = max\{P^n - S^n, 0\}$.

Case 2: $\eta\tilde{E}^n + w_G V > 0$; $-\left(\tilde{E}^n + w_G V\right) > 0$. L-SPAR will neither charge nor discharge in n^{th}slot. Therefore, $c^{n*} = d^{n*} = 0$.

Case 3: $\eta\tilde{E}^n + w_G V \leq 0$; $-\left(\tilde{E}^n + w_G V\right) > 0$. L-SPAR will charge as much as possible and will not discharge at n^{th}slot. Since c^{n*} has to satisfy constraint (12), we have $c^{n*} = S^n$ and $d^{n*} = 0$.

Case 4: $\eta\tilde{E}^n + w_G V \leq 0$; $-\left(\tilde{E}^n + w_G V\right) \leq 0$. Case 4 will not happen since it contradicts with our assumption $0 < \eta < 1$ and $V > 0$.

Note that if $S^n > P^n + c^n$, in this case, the portion of harvested solar $S^n - P^n - c^n$ can not be utilized either to charge the battery or power the BS and will be wasted. Furthermore, when $-V \geq \tilde{E}^n \geq \frac{-V}{\eta}$, we have $c^{n*} = d^{n*} = 0$, which may lead to a "static zone" where there is no further charge and discharge of the battery. However, there is a $(1-\varphi)$ portion of the energy stored in the battery which will be lost due to leakage, so the battery will not be trapped in the static zone in our algorithm.

BS resource allocation: After solving the charge and discharge problem as a function of P^n, we will solve the BS resource allocation $\boldsymbol{X}^n$ based on the three possible cases derived from the charge and discharge decision.

Case 1: We have $c^{n*} = 0$ and $d^{n*} = max\{P^n - S^n, 0\}$. We will first solve $\boldsymbol{X}^n$ assuming $d^{n*} = P^n - S^n$. Rewriting **P2** and omitting the constant terms, we want to solve

$$\min_{\boldsymbol{X}^n} \sum_{i=1}^{I} \tilde{Q}_i^n \left(\sum_{k=1}^{K} x_{ik}^n r_{ik}^n\right) - \tilde{E}^n \sum_{i=1}^{I}\sum_{k=1}^{K} x_{ik}^n \left(\frac{P_{tx}}{\Delta} + P_{sp}\right)$$

$$= \min_{\boldsymbol{X}^n} \sum_{i=1}^{I}\sum_{k=1}^{K} x_{ik}^n \left(\tilde{Q}_i^n r_{ik}^n - \tilde{E}^n\left(\frac{P_{tx}}{\Delta} + P_{sp}\right)\right)$$

$$\overset{(a)}{\Leftrightarrow} \sum_{k=1}^{K} \min_{\boldsymbol{X}^n} \sum_{i=1}^{I} x_{ik}^n \left(\tilde{Q}_i^n r_{ik}^n - \tilde{E}^n\left(\frac{P_{tx}}{\Delta} + P_{sp}\right)\right) \quad (21)$$

s.t. (14)

where (a) is because multiple subcarriers can be allocated to one single user. Therefore, the minimization problem can be viewed as the sum of K minimization problems in each subcarrier. For simplicity, we define $y_{ik}^n = (\tilde{Q}_i^n r_{ik}^n - \tilde{E}^n(\frac{P_{tx}}{\Delta} + P_{sp}))$. To minimize the per-subcarrier problem, the solution for the optimal BS resource allocation is to select one user with the minimal and negative y_{ik}^n. If all y_{ik}^n are non-negative on subcarrier k, the BS will not allocate subcarrier k to any user. The optimal solution x_{ik}^{n*} is given as

$$x_{ik}^{n*} = \begin{cases} 1, & \text{if } y_{ik}^n \le \min\{0,\ y_{k,min}^n\} \\ 0, & \text{otherwise} \end{cases} \tag{22}$$

where $y_{k,min}^n = \min_{i \in \boldsymbol{I}, k'=k} y_{ik'}^n$, $k = 1,2,..,K$. If the resulting $P^{n*} < S^n$ given x_{ik}^{n*}, there is no feasible solution for $d^{n*} = P^n - S^n$. We can set $d^{n*} = 0$ and solve $\boldsymbol{X}^n$ as the same method in Case 2 where $c^{n*} = d^{n*} = 0$.

Case 2 & 3: We have $c^{n*} = 0$ and $c^{n*} = S^n$ in Case 2 and Case 3 respectively while $d^{n*} = 0$ in both cases. Note that no matter $c^{n*} = 0$ or $c^{n*} = S^n$, the $c^n(\eta\tilde{E}^n + w_G V)$ term in **P2** remains a constant in n^{th} slot which does not affect the decision of $\boldsymbol{X}^n$. After rewriting **P2** and omitting the constant terms, we want to solve

$$\min_{\boldsymbol{X}^n} \sum_{i=1}^{I} \tilde{Q}_i^n (\sum_{k=1}^{K} x_{ik}^n r_{ik}^n) + w_G V \sum_{i=1}^{I} \sum_{k=1}^{K} x_{ik}^n (\frac{P_{tx}}{\Delta} + P_{sp})$$

$$= \min_{\boldsymbol{X}^n} \sum_{i=1}^{I} \sum_{k=1}^{K} x_{ik}^n (\tilde{Q}_i^n r_{ik}^n + w_G V(\frac{P_{tx}}{\Delta} + P_{sp})) \tag{23}$$

s.t. (14).

Similar as Case 1, if we set $y_{ik}^{n'} = (\tilde{Q}_i^n r_{ik}^n + w_G V(\frac{P_{tx}}{\Delta} + P_{sp}))$, we have the optimal solution x_{ik}^{n*} given as

$$x_{ik}^{n*} = \begin{cases} 1, & \text{if } y_{ik}^{n'} \le \min\{0,\ y_{k,min}^{n'}\} \\ 0, & \text{otherwise} \end{cases} \tag{24}$$

where $y_{k,min}^{n'} = \min_{i \in \boldsymbol{I}, k'=k} y_{ik'}^{n'}, k = 1,2,..,K$. For each subcarrier k in all three cases, the user with minimal $\tilde{Q}_i^n r_{ik}^n$ will be selected as potential candidate to serve. The physical interpretation is that among the users whose data buffer levels are lower than their perturbed levels θ_i, the BS will serve the user with the largest product of achievable transmission rate and the gap to their predefined buffer level θ_i in order to fill the gap. On the other hand, if the buffer levels of all users are larger than θ_i, BS will not allocate subcarriers or serve the user with smallest product of achievable transmission rate and excess data compared to their perturbed buffer level θ_i to avoid buffer overflow. After selecting the potential candidate, the algorithm compares $\tilde{Q}_i^n r_{ik}^n$ with either $\tilde{E}^n(\frac{P_{tx}}{\Delta} + P_{sp})$ or $-w_G V(\frac{P_{tx}}{\Delta} + P_{sp})$, depending on whether the BS is powered by the battery in Case 1 or grid in Case 2 & 3 respectively. We can observe that if $\tilde{E}^n$ is larger (the battery level is higher) in Case 1 or V is smaller (L-SPAR focuses more on stability of queues over performance) in Case 2 & 3, the BS is more likely to allocate subcarriers to users and hence consumes more energy.

UE buffer consumption rate: To obtain the optimal UE buffer consumption rate δ_i^n, we solve the problem

$$\min_{\delta_i^n} - \sum_{i=1}^{I} [\tilde{Q}_i^n \delta_i^n + w_I V U(\delta_i^n)] \tag{25}$$

$$s.t. \ (5).$$

As we prove below, the optimal solution δ_i^{n*} is given as

$$\delta_i^{n*} = \min\{\delta^{\max}, \ U_i'^{-1}\left(\frac{-\tilde{Q}_i^n}{w_I V}\right)\} \tag{26}$$

where $U_i'^{-1}(\cdot)$ is the inverse function of $U_i'(\delta)$ and satisfies $U_i'^{-1}(U_i'(\delta)) = \delta$ for $\delta \in [0, \delta^{\max}]$.

Proof: The minimization problem can be viewed as the sum of I minimization problems for each user. The objective function for each user is a strictly convex function for $\delta \in [0, \delta^{\max}]$ since it is the negative sum of a linear function and a strictly concave function $U_i(\delta)$. Moreover, if the

derivative of a strictly convex function is zero at some point which is $\delta_i^n = U_i'^{-1}\left(\frac{-\tilde{Q}_i^n}{w_I V}\right)$ in our case, then that point is a global minimum. For $\tilde{Q}_i^n \leq -w_I V U_i'(\delta^{\max})$, the optimal point $\delta_i^n = U_i'^{-1}\left(\frac{-\tilde{Q}_i^n}{w_I V}\right)$ is within $[0, \delta^{\max}]$. For $\tilde{Q}_i^n > -w_I V U_i'(\delta^{\max})$, $U_i'^{-1}\left(\frac{-\tilde{Q}_i^n}{w_I V}\right)$ is not within $[0, \delta^{\max}]$ and hence not a feasible solution. Moreover, since $U_i'(\delta)$ is positive and decreasing for $\delta \in [0, \delta^{\max}]$, the first derivative of objective function $-\tilde{Q}_i^n - w_I V U_i'(\delta_i^n)$ is always negative for $\delta \in [0, \delta^{\max}]$. Therefore, the objective is a monotonically decreasing function for $\delta \in [0, \delta^{\max}]$ and thereby $\delta^{\max}$ is the optimal solution.

In each time slot, the computational complexity of the L-SPAR algorithm comes from the BS resource allocation where sorting y_{ik}^n or $y_{ik}^{n'}$ of I users on each subcarrier requires $O(I \log I)$ time. The complexity of BS resource allocation is then bounded by $O(KI \log I)$ where K is the numbers of subcarriers. As we described in the previous section, the complexity is independent of the complexity of system states (e.g. harvested solar energy, channel state, battery state, UE buffer state) and the choice of $(V, \boldsymbol{\theta})$.

3.5 Performance Analysis

In this section, we will show that the L-SPAR algorithm satisfies all constraints in **P1** and provides a theoretical performance bound of L-SPAR. Furthermore, we will discuss the relation between the performance and the choice of the predefined parameters $(V, \boldsymbol{\theta})$ in L-SPAR.

3.5.1 Feasibility Analysis

In the proposed L-SPAR algorithm, the UE data buffer constraint (6) and the battery capacity and energy causality constraint (9) are ignored. It is important to show that for given

pair of $(V, \boldsymbol{\theta})$, solving the per-time slot problem **P2** will produce feasible solutions of **P1** under the constraints (6) and (9).

Proposition 1: Under the L-SPAR algorithm, the battery level E^n is confined within $[0, E^{\max}]$.

Proof: We first prove E^n is lower bounded by 0. Firstly, we have $E^0 = 0$ from assumption. From L-SPAR we know that $d^{n*} = 0$ when $\tilde{E}^n = E^n - \theta_0 \leq -w_G V$. Suppose $0 \leq E^n \leq \theta_0 - w_G V$, we have $E^{n+1} = \varphi E^n + \eta c^n \geq 0$. On the other hand, if $E^n > \theta_0 - w_G V$, $E^{n+1} \geq \varphi E^n - S^n + P^n \geq \varphi E^n - P^{\max}$. Since $\theta_0 \geq w_G V + \frac{P^{\max}}{\varphi}$ by constraint (17), $E^{n+1} > \varphi(w_G V + \frac{P^{\max}}{\varphi} - w_G V) - P^{\max} \geq 0$.

Next, we will show that E^n is upper bounded by $E^{\max}$. Suppose $\frac{-w_G V}{\eta} + \theta_0 \leq E^n \leq E^{\max}$, from L-SPAR we know the optimal $c^{n*} = 0$. Therefore, $E^{n+1} = \varphi E^n - d^{n*} \leq E^n \leq E^{\max}$. Otherwise, if $E^n < \frac{-w_G V}{\eta} + \theta_0$, $E^{n+1} \leq \varphi E^n + S^{\max} < \varphi\left(\frac{-w_G V}{\eta} + \theta_0\right) + S^{\max}$. Since $\theta_0 \leq \frac{w_G V}{\eta} + \frac{E^{\max} - S^{\max}}{\varphi}$ by constraint (17), $E^{n+1} \leq E^{max}$.

Lemma 1: Given $(V, \boldsymbol{\theta})$ satisfying (17) and (18), we have $U_i'(Q_i^n) \leq U_i'\left(U_i'^{-1}\left(\frac{-\tilde{Q}_i^n}{w_I V}\right)\right), Q_i^n \in [0, \delta^{max}]$

Proof: See Appendix A in [110].

Since $U_i(\delta)$ is concave and differentiable for $\delta \in [0, \delta^{\max}], Q_i^{n+1} \geq Q_i^n - \delta_i^{n*} \geq Q_i^n - U_i'^{-1}\left(\frac{-\tilde{Q}_i^n}{w_I V}\right) \geq 0$ if and only if $U_i'(Q_i^n) \leq U_i'\left(U_i'^{-1}\left(\frac{-\tilde{Q}_i^n}{w_I V}\right)\right)$, which is derived in Lemma 1. Therefore, with the assumption that $Q_i^0 = 0, \forall i$, we can prove that $Q_i^n \geq 0$. Next, we will show that Q_i^n is upper bounded by Q_i^{max}.

Lemma 2: $Q_i^n \le \theta_i + \frac{\mu}{r^*}(\frac{P_{tx}}{\Delta} + P_{sp}) + Kr^*$, where $r^* = \max_r \frac{\mu}{r}(\frac{P_{tx}}{\Delta} + P_{sp}) + Kr,\ \mu = \max\{E^{\max} - \theta_0, -w_G V\}, r \in [r^{min}, r^{max}]$

Proof: See Appendix B in [110].

According to Lemma 2, given the size of the available data buffer Q_i^{max}, we derive the upper bound of the control parameter θ_i

$$\theta_i \le Q_i^{max} - \frac{\mu}{r^*}(\frac{P_{tx}}{\Delta} + P_{sp}) - Kr^* \forall i \tag{27}$$

where constraint $Q_i^n \le Q_i^{max}$ can be satisfied.

The above two propositions together imply that the proposed per-timeslot L-SPAR algorithm with proper selection of $(V, \boldsymbol{\theta})$ can always yield a feasible control policy satisfying constraints (6) and (9) under any arbitrary stochastic process of solar energy S^n and channel conditions $\boldsymbol{H}^n$.

3.5.2 Performance Analysis

We will next show that L-SPAR algorithm yields an asymptotically near-optimal solution. By the definition of the drift-plus-penalty function defined in [23], we define Lyapunov function as the total sum of virtual queues length

$$L(n) = \frac{1}{2}\left[\sum_{i=1}^{I}(\tilde{Q}_i^n)^2 + \left(\tilde{E}^n\right)^2\right]. \tag{28}$$

Next, the Lyapunov drift is defined as

$$\Delta(n) = \mathbb{E}[L(n+1) - L(n)]. \tag{29}$$

The Lyapunov drift-plus-penalty function is then defined as

$$\Delta_V(n) = \Delta(n) + V\{w_G \mathbb{E}[G^n] - w_I \sum_{i=1}^{I} \mathbb{E}[U_i(\delta_i^n)]\}.$$

Lemma 3: For arbitrary feasible decision variables $\{\boldsymbol{X}^n, \delta_i^n, c^n, d^n\}$, $\Delta_V(n)$ is upper bounded by

$$\Delta_V(n) \leq \mathbb{E}\Big\{ \sum_{i=1}^{I} \tilde{Q}_i^n (\sum_{k=1}^{K} x_{ik}^n r_{ik}^n - \delta_i^n) + \tilde{E}^n [\, \eta c^n - d^n - (\tilde{E}^n + \theta_0)(1-\varphi)\big] +$$

$$V[w_G(P^n - S^n + c^n - d^n) - w_I \sum_{i=1}^{I} U(\delta_i^n)]\} + C_1 \tag{31}$$

where the constant term C_1 equals to

$$C_1 = \frac{\max\{(\eta S^{\max})^2, [P^{\max} + (1-\varphi)E^{\max}]^2\}}{2} + I\,\frac{\max[(R^{\max})^2, (\delta^{\max})^2]}{2}.$$

Proof: After subtracting $\boldsymbol{\theta}$ on both sides of (4) and (8), we have

$$\tilde{Q}_i^{n+1} = \tilde{Q}_i^n + R_i^n - \delta_i^n, \tag{32}$$

$$\tilde{E}^{n+1} = \tilde{E}^n + \eta c^{n-1} - d^{n-1} - (\tilde{E}^n + \theta_0)\,(1-\varphi). \tag{33}$$

By squaring both sides of (32) and (33), and summing up the equalities, we have

$$\begin{aligned}
\Delta(n) &= \frac{1}{2}\left[\sum_{i=1}^{I} (\tilde{Q}_i^{n+1})^2 + \left(\tilde{E}^{n+1}\right)^2\right] - \frac{1}{2}\left[\sum_{i=1}^{I} (\tilde{Q}_i^{n})^2 + \left(\tilde{E}^{n}\right)^2\right] \\
&= \frac{1}{2}\sum_{i=1}^{I}\left(\sum_{k=1}^{K} x_{ik}^n r_{ik}^n - \delta_i^n\right)^2 \\
&\quad + \frac{1}{2}[\eta c^n - d^n - (\tilde{E}^n + \theta_0)\,(1-\varphi)]^2 \\
&\quad + \sum_{i=1}^{I} \tilde{Q}_i^n \left(\sum_{k=1}^{K} x_{ik}^n r_{ik}^n - \delta_i\right) \\
&\quad + \tilde{E}^n\big[\, \eta c^n - d^n - \big(\tilde{E}^n + \theta_0\big)(1-\varphi)\big] \\
&\leq \sum_{i=1}^{I} \tilde{Q}_i^n \left(\sum_{k=1}^{K} x_{ik}^n r_{ik}^n - \delta_i\right) \\
&\quad + \tilde{E}^n\big[\, \eta c^n - d^n - \big(\tilde{E}^n + \theta_0\big)(1-\varphi)\big] + C_1.
\end{aligned}$$

The inequality holds since $x_{ik}^n r_{ik}^n, \delta_i^n, \eta c^n$ and $d^n + \big(\tilde{E}^n + \theta_0\big)(1-\varphi)$ are all non-negative. We then add the penalty function $V[w_G(P^n - S^n + c^n - d^n) - w_I \sum_{i=1}^{I} U(\delta_i^n)]$ and take expectation on both sides to obtain the desired result. We then define an auxiliary problem **P3**. In **P3**, the constraints (6) and (9) are replaced by the corresponding time-average version (36) and (37).

$$\textbf{P3:} \min_{\boldsymbol{X}^n,\delta_i^n,c^n,d^n} \lim_{N\to+\infty}\frac{1}{N}\sum_{n=0}^{N-1}\mathbb{E}\left[w_G G^n - w_I\sum_{i=1}^{I}U_i(\delta_i^n)\right] \tag{34}$$

$$s.t. \quad (5), (11)\text{-}(13), (15)$$

$$\lim_{N\to+\infty}\frac{1}{N}\sum_{n=0}^{N-1}\mathbb{E}[R_i^n - \delta_i^n] = 0, \forall i, \tag{35}$$

$$0\leq \lim_{N\to+\infty}\frac{1}{N}\sum_{n=0}^{N-1}\mathbb{E}[\,\eta c^n - d^n] \leq (1-\varphi)E^{max}, \tag{36}$$

Proposition 4: **P3** is the relaxation of **P1** where any feasible solution in **P1** satisfies (35) and (36).

Proof: By summing up both sides of (4) and (8) for $n = 1,2,\dots N$, taking the expectation, divide both sides by N and let N go to infinity, we have

$$\lim_{N\to+\infty}\mathbb{E}\left[\frac{Q_i^N}{N}\right] = \lim_{N\to+\infty}\mathbb{E}\left[\frac{Q_i^0}{N}\right] + \lim_{N\to+\infty}\frac{1}{N}\mathbb{E}[R_i^n - \delta_i^n], \forall i, \tag{37}$$

$$\lim_{N\to+\infty}\frac{(1-\varphi)}{N}\sum_{n=1}^{N}\mathbb{E}[E^n] = \lim_{N\to+\infty}\mathbb{E}\left[\frac{E^0}{N}\right] + \lim_{N\to+\infty}\frac{1}{N}\mathbb{E}[\eta c^n - d^n]. \tag{38}$$

Since both $Q_i^N < \infty$ and E^n is bounded within $[0, E^{max}]$, (37) and (38) are satisfied.

Let Y^{opt} and $\tilde{Y}^{opt}$ be the optimal value of penalty function of **P1** and **P3** respectively. $Y^{opt} \geq \tilde{Y}^{opt}$ since every feasible solution in **P1** satisfies **P3**.

Lemma 4: For arbitrary $\varepsilon > 0$, there exist a stationary control policy $\boldsymbol{\Pi} = \{\boldsymbol{X}^{\Pi}, \delta_i^{\Pi}, c^{\Pi}, d^{\Pi}\}$ which observes $\{S^n, \boldsymbol{H}^n\}$ for each slot n and independently choose a control action in **P3** and satisfies

$$\mathbb{E}[G^{n\Pi} - \textstyle\sum_{i=1}^{I}U_i\left(\delta_i^{n\Pi}\right)] \leq \tilde{Y}^{opt} + \varepsilon, \tag{39}$$

$$\left|\mathbb{E}\left[R_i^{n\Pi} - \delta_i^{n\Pi}\right]\right| \leq \varepsilon, \forall i, \tag{40}$$

$$(1-\varphi)E^{max} \geq \mathbb{E}[\,\eta c^{n\Pi} - d^{n\Pi}] \geq 0. \tag{41}$$

Proof: The proof is similar to Theorem 4.5 in [94], which is omitted for brevity.

Next, we will derive the worst-case performance of L-SPAR algorithm with the auxiliary problem **P3.**

Theorem 1: The objective function achieved by L-SPAR is upper-bounded by $Y^{opt} + \frac{C_2}{V}$ where C_2 is given by

$$C_2 = (1-\varphi)E^{\max}\max\{\theta_0, (E^{\max} - \theta_0)\} + C_1.$$

Note that Y^{opt} is the optimal value of **P1** under any feasible control algorithm, even if which relies on future knowledge of random process $\{S^n, \boldsymbol{H}^n\}$.

Proof: See Appendix C in [110].

According to Theorem 1, the gap between the solution achieved by L-SPAR and the optimal solution Y^{opt} is decided by $\frac{C_2}{V}$, and $C_2 = (1-\varphi)E^{\max}\max\{\theta_0, (E^{\max} - \theta_0)\} + \frac{\max\{(\eta S^{\max})^2, [P^{\max} + (1-\varphi)E^{\max}]^2\}}{2} + I\frac{\max[(R^{\max})^2, (\delta^{\max})^2]}{2}$ is a constant with given system parameters. Therefore, we can make the objective function arbitrarily close to theoretical optimal solution Y^{opt} by letting $V \to \infty$. However, increasing V comes with a cost of the increasing convergence level of data buffer $w_I V\beta_i + \delta^{\max}$ and batteries buffer $w_G V + \frac{P^{\max}}{\varphi}$, according to equations (17) and (18). In other word, the decision of V is a tradeoff between the performance (in terms of grid power consumption and user utility) and longer convergence time and higher buffer requirement. By setting proper values of (w_G, w_I, V), we can adjust the priority of L-SPAR algorithm to meet with different system requirements.

3.6 Simulation Framework and Results

In this section, we will discuss the developed simulation framework and results obtained by using the proposed L-SPAR algorithm and compare the results with existing methods during mobile video download.

3.6.1 Simulation Framework

We have developed a Matlab based simulation framework which consists of PV harvesting model, BS power consumption model, and traffic demand model of UEs. The framework allows us to implement different video download techniques and evaluate the grid power consumption for temporally varying harvested solar energy and channel conditions. We will briefly describe the above models and the related simulation parameters, as listed in Table 2.

In our simulation study, we assume the harvested solar energy S^n is uniformly distributed between 0 and S^{max} =200W. To show that L-SPAR does not depend on the assumption on the random processes and holds for non-i.i.d cases, we will also include the actual solar irradiance trace in [88] in performance comparison. We assume imperfect batteries at the BS, with storage efficiency $\varphi = 0.99$, and charging efficiency $\eta = 0.8$ and maximum capacity 300J. The linear power consumption model elaborated in section 3.3 is used with the parameters obtained from [107]. For the network and channel model, we assume the BS has 20 subcarriers with equal

TABLE 3.2. Simulation Parameters

Cell radius	150m
Simulation time	1hour
Path-loss(dB)	$140.7 + 36.7\, log_{10}\, R$, R is the distance between user and BS and is in kilometers
Noise power	-105.86dBm
Bandwidth BW	10MHz
Max transmit power	4W
Maximum BS power	140W
Static BS power	35W
Number of users	10
UE Buffer size	500 MB

bandwidth. The cell radius, transmit power, noise power, system bandwidth and channel gain parameters recommended in Long Term Evolution (LTE) specifications [108, 109] are listed in Table 3.2. We assume users are randomly distributed within a 150-meter radius with the total number of concurrent users $I = 10$. The channel gains at each time slot are exponentially distributed with mean equal to path-loss model given in Table 2. Different users download videos of different bitrates with utility function $U_i(\delta) = ln(1 + \delta_i)$. We assume each user has maximum buffer size of 500 MB and maximum buffer consumption rate 10MB/s. For the performance metrics, we assume equal weights of the grid power consumption w_G and aggregate user utility w_I in our simulation.

We compare the proposed technique with two existing relevant techniques, [85] and [95], listed below. For convenience, we will refer to them as Approach 1 [85] and Approach 2 [95] respectively. In Approach 1, which makes greedy decisions to minimize the objective function, the UEs first choose the largest possible buffer consumption rate considering its available downloaded data. Secondly, Approach 1 arranges the UEs in an ascending order with respect to their remaining data in their buffers. The BS then gives higher priority for using the harvested RE and only allocates the subcarrier with the best channel gain to the UE which buffer level is below a minimal level, which is the maximum buffer consumption rate δ^{max} in our simulation. In other words, the BS only allocates the minimum required resources (determined by the buffer consumption rate and buffer level) to the users while UEs try to maximize their utility. Approach 2 uses standard Lyapunov optimization framework while perfect batteries are assumed. The method is similar to L-SPAR instead of two key differences: 1) instead of using data buffer to halt or reduce data transmission, the BS allocates enough subcarriers to meet the required buffer

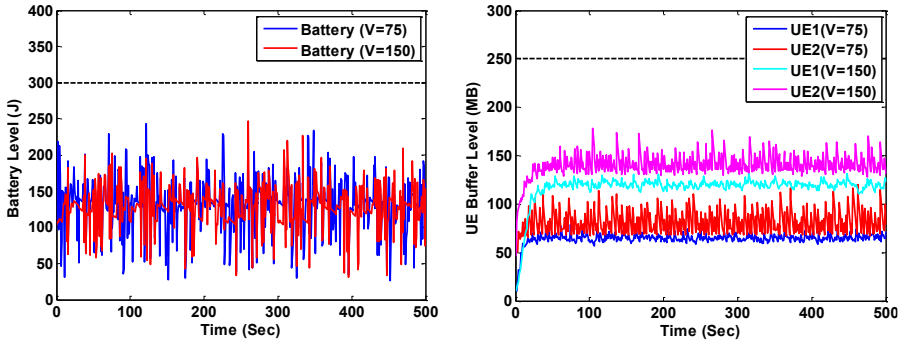

Figure 3.2. (a) left, battery level versus time (b) right, buffer level of two users versus time

consumption rate of UEs in each time slot, 2) the optimization process does not consider the effect of battery imperfection.

3.6.2 Simulation Results

The simulation results consist of two parts: We will first verify the feasibility of L-SPAR algorithm by examining the battery level of the BS and the UE buffer level. Secondly, we will compare L-SPAR with the other methods using the weighted sum of grid energy consumption and aggregate user utility defined in (13) and the corresponding required battery and UEs' buffer. In our simulation, the value of θ_0 and θ_i are chosen as the LHS of (17) and (18) respectively, which means the minimum value of θ_0 and θ_i are chosen with given V, the non-negative weight parameter in Lyapunov optimization where larger V emphasizes more on objective minimization over the queue stability. The rationale is that the larger the perturbation parameter θ_0 and θ_i, L-SPAR tends to maintain unnecessarily higher battery level and buffer level of UEs. We choose $V = 75$ and $V = 150$ as two examples to discuss how different values of V affect the dynamics of the battery and the UEs' buffer level, as shown in Fig. 2. Note we only show the first 500 seconds of the simulation since both targets converge and stabilize within the first 500 seconds.

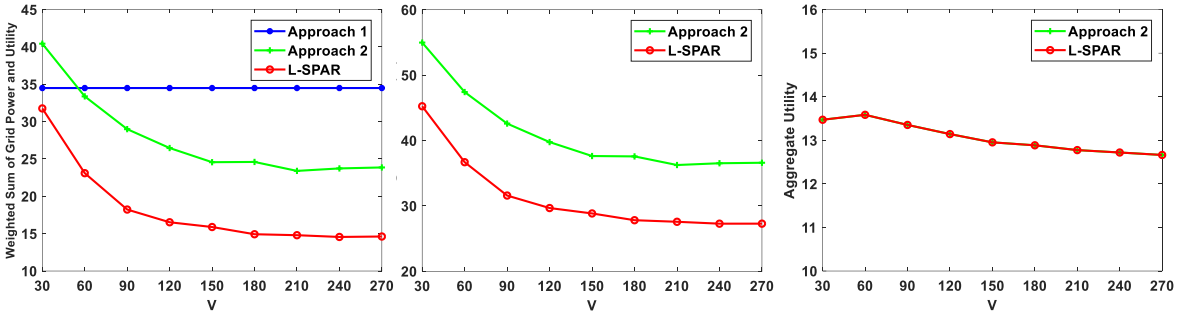

Figure 3.3. Performance with uniformly distributed solar trace (a) left, weighted sum of grid power consumption and aggregate user utility vs. V, (b) center, average grid power consumption vs. V and (c) right, aggregate utility of UEs vs. V

In Fig. 3.2(a), we can observe that the battery level in both cases fluctuates with the maximum charge ηS^{max} and the maximum discharge P^{max} and the variation is within the range between 40J and 250J. The major difference between the two cases is the frequency of charging and discharging in $V = 75$ is higher than $V = 150$. When V increases, the gap between charging and discharging threshold also increases, which makes charging and discharging less likely to occur. This also implies that when V increases, L-SPAR charge/discharge the battery less frequently to avoid charging energy loss. In Fig. 3.2(b), the buffer of two users with their distance to the BS equals to 100m (UE1) and 50m (UE2) are shown. Firstly, we can see that since the average channel gain of UE2 is better than UE1, the buffer level of UE2 converges faster and the convergence level is higher than UE1. This can be explained in the per-time slot problem that the larger the achievable transmission rate of UE i, the more likely the BS will allocate the subcarriers to UE i. Moreover, when V increases to 150, convergence of the buffer of both UEs to a higher level takes longer time, which is the tradeoff between performance matrix and queue size in Lyapunov optimization framework. As shown in Fig. 3.2, we can conclude that the proposed L-SPAR algorithm is feasible in terms of battery and data buffer.

In Fig. 3.3, we compare the performance metric, which is the weighted sum of grid energy consumption and aggregate user utility, between the three methods with different values of V. Note

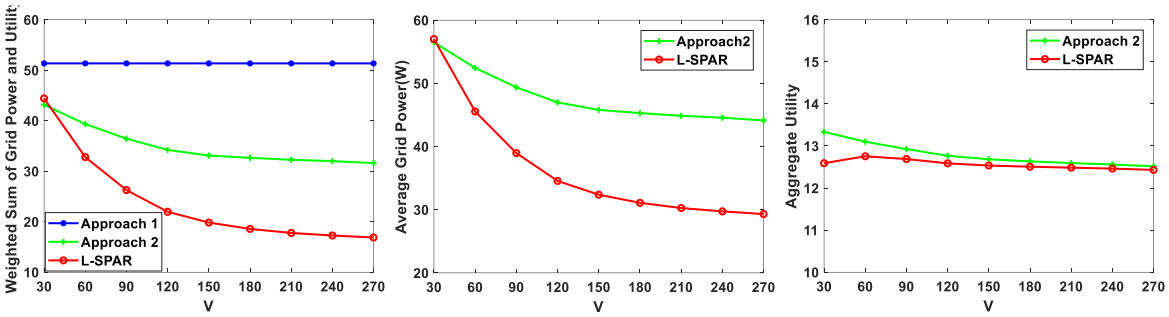

Figure 3.4. Performance with actual solar trace (a) left, weighted sum of grid power consumption and aggregate user utility vs. V, (b) center, average grid power consumption vs. V and (c) right, aggregate utility of UEs vs. V

that Approach 1 is independent of V. We can see that the weighted sum is inversely proportional to V, which verifies the asymptotic optimality of Theorem 1. Fig. 3(a) further shows that L-SPAR can produce consistently better performance than the other two methods. For example, when $V = 270$, L-SPAR improves the performance by 57.6% and 38.8% compared with Approach 1 and Approach 2 respectively. In Fig. 3.3(b) and 3.3(c), we compare the average grid power consumption and aggregate utility of UEs respectively between L-SPAR and Approach 2 (Approach 1 is omitted since it is independent of V). We can see that power consumption decreases as V increases for both methods while the aggregate utility of UEs decreases. Moreover, L-SPAR consistently consumes less power than Approach 2 with the same buffer consumption rate. As an example, when $V = 270$, L-SPAR consumes 25.5% less average power than Approach 2. As shown in Fig. 3.3, L-SPAR effectively reduces the energy consumption by 37.7% while the aggregate utility of UEs only decreases by 5.9%. In the scenario with actual solar irradiance trace, we assume the solar module associated with the BS uses typical crystalline solar cells with 15% conversion efficiency, and measurement of the solar power profile used in [88] for the simulation is from 8AM to 4PM. The performance is generally worse for all three methods because of non-stationary and non-linear solar profile. However, we can observe similar trend as the case with

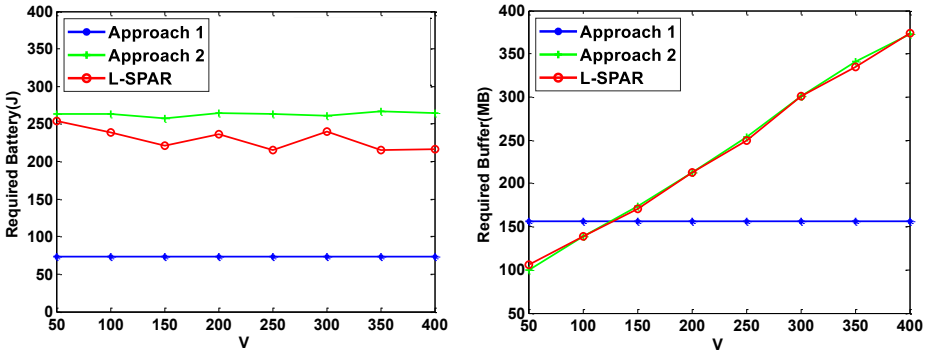

Figure 3.5. (a) left, required battery level vs. V and (b) right, average buffer vs. V

uniformly distributed solar power, as shown in Figure 4. When $V = 270$, L-SPAR improves the performance by 68.1% and 46.8% compared with Approach 1 and Approach 2 respectively.

In Fig. 3.5, we simulate the maximum battery level and user buffer needed for different values of V. In Fig. 5(a), the maximum battery level in Approach 2 approximately remains the same as V increases while it slightly decreases in L-SPAR. The observation is different from what is discussed in Sec. 3.5 where the required batter capacity should increase with the increment of θ_0 as V increases. The reason is that the maximum battery level in L-SPAR decreases as V increases is L-SPAR takes charging efficiency into account. In L-SPAR algorithm, L-SPAR will not charge if $\tilde{E}^n \geq \frac{-V}{\eta}$. Applying $\theta_0 = V + \frac{P^{\max}}{\varphi}$ as the setting in our simulation, we have the above threshold as $E^n = \tilde{E}^n + \theta_0 \geq \frac{\eta-1}{\eta}V + \frac{P^{\max}}{\varphi}$. Since $\frac{\eta-1}{\eta} < 0$, the charging threshold in L-SPAR decrease as V increases, showing that the actual required battery capacity is less than the theoretical bound in (17). In Fig. 3.5(b), we can observe that the required buffer level of both Approach 2 and L-SPAR is proportional to V. This verifies the performance analysis and the growth of required data buffer of UEs and longer convergence time observed in Fig 5(b), which becomes the main tradeoff to achieve better grid energy consumption and aggregate user utility.

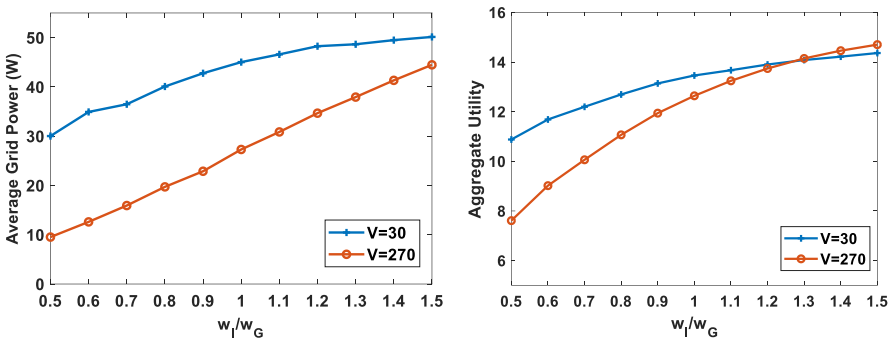

Figure 3.6. (a) left, average grid power consumption and (b) right, aggregate utility of UEs vs. $\frac{w_I}{w_G}$

We will next discuss the effect of choosing different w_G and w_I, the weights associated with grid energy consumption and user utility in equation (13). Grid power consumption and aggregate user utility versus $\frac{w_I}{w_G}$ with $V = 30$ and $V = 270$ are shown in Fig.3.6. As $\frac{w_I}{w_G}$ increases, both grid power consumption and user utility increases because the BS tends to consume more power to transmit data to UEs to increase aggregate utility. Secondly, the grid power consumption grows linearly while aggregate utility grows logarithmically with increasing $\frac{w_I}{w_G}$ which indicates the tradeoff between grid power and aggregate utility is not uniform. Thirdly, different values of V result in different tradeoffs between grid power and aggregate utility. For example, from $\frac{w_I}{w_G} = 0.5$ to $\frac{w_I}{w_G} = 1.5$, the ratio of increased aggregate utility to increased grid power is 0.17 when $V = 30$ and 0.20 when $V = 270$, respectively. In conclusion, the parameter set (w_G, w_I, V) in L-SPAR can be chosen to meet the data buffer constraint of UEs and arbitrary priority of grid power consumption and utility of UEs as desired by a specific service provider or network operator.

3.7 Conclusion

In this paper, we propose a renewable energy (RE)-aware BS resource allocation technique which aims to better utilize intermittent harvested renewable energy to reduce grid power consumption of hybrid energy supply (HES) BSs and enhance QoS of UEs. We utilize the data buffer of UEs together with energy storage of the BS to adapt the BS resource. Our technique optimizes charging and discharging of batteries, subcarrier allocation of the BS and buffer consumption rate according to given CSI and ESI and a imperfect battery model.

To avoid the performance degradation due to imperfect prediction of CSI and ESI and reduce the computation complexity, we propose a Lyapunov optimization-based online algorithm in a RE-aware manner. To minimize grid energy consumption while maximizing utility of users, the weighted sum of the above two targets is used as our objective function. To satisfy the causality and capacity constraints of batteries and UEs' buffer, we generalize the Lyapunov optimization technique and propose an online L-SPAR algorithm based on current state of energy arrival, channel condition, battery level and buffer level of UEs. We then prove the feasibility and performance bound of L-SPAR algorithm. The simulation results show that L-SPAR provides a feasible solution and effectively reduces grid power consumption compared to conventional non-RE schemes and existing Lyapunov-based techniques.

Jointly solving power and subcarrier allocation in each time slot can enable full utilization of spatial/temporal diversity of OFDMA networks and can potentially lead to better optimization opportunities and hence better performance. In the future, we plan to explore addressing this more generalized problem, while also addressing its significantly increased complexity. Finally, we plan to extend this research to a cooperative BSs scheme to incorporate both temporal and spatial variation of harvested RE in multi-BSs scenario.

Chapter 3, in part, is from the material as it appears in proceedings of IEEE International Symposium on Personal, Indoor and Mobile Radio Communications 2015, Po-Han Chiang, Ranjini B. Guruprasad and Sujit Dey. and in IEEE Transactions on Green Communications and Networking 2018, Po-Han Chiang, Ranjini B. Guruprasad and Sujit Dey. The dissertation author was the primary investigator and author of the papers.

Conclusion

In this dissertation, we investigate the relationships between BP and lifestyle factors and provide personalized and precise recommendations to improve BP, as opposed to the current practice of general lifestyle recommendations. We have collected real patient data from their BP monitors and wearable activity trackers remotely. Using the collected data, we train a personalized BP model based on random forest (RF), which can predict individual's BP using health behavior and historical BP, and identify the most important factors in predicting an individual's BP. We propose RFFS, which performs RF-based feature selection to enhance the prediction. Since BP and health behavior data are collected and learned sequentially, the performance of prediction is prone to the existence of concept drifts and anomaly points. To solve this problem, we propose an Online Weighted-Resampling technique to enhance RFFS in an online learning scenario. Lastly, we propose Random Forest with Shapley-Value-based Feature Selection to offer personalized BP modeling and top lifestyle factor identification, and subsequent generation of precise recommendations based on the top factors. We conducted a clinical study, applying our system to 25 patients with elevated BP or stage I hypertension for three consecutive months. Our study results validate our system's ability to provide accurate personalized BP models and identify the top features. We also validate the effectiveness of personalized recommendations in a randomized controlled experiment.

In the future, we would like to extend our research in the following directions. Firstly, a one-time recommendation was used in this study. However, the recommendation may change with time due to patient's compliance and/or contextual factors that cannot be collected in our study. Therefore, the proposed method should update and deliver the recommendation to patients in a timely basis. Although, the proposed Online Weighted-Resampling technique can improve the

performance, the effect on the recommendation is unknown. We plan to collect more BP and compliance data from the patient after the recommendation. Based on the information, we can adjust the future recommendations in a more interactive way.

Secondly, while the results of lifestyle intervention in this study are encouraging, a larger and more heterogeneous group of subjects over a longer duration are needed in order to obtain a more representative result. Since the study requires continuous engagement of patients, more automated and convenient means of monitoring and recommendation are also needed to keep patients engaged. For example, automated text reminder can be sent when patients forget to measure their BP or synchronize their data.

Last but not least, it has been known that dietary factors, like sodium intake, may also significantly affect BP. Traditional methods assess food (nutrition) intake with self-report measures, such as food frequency questionnaires (FFQs) and photo-assisted dietary assessments. However, the accuracy of dietary intake assessment remains a challenge. Moreover, no widely adopted technology can assess dietary intake automatically and accurately. By combining manual input from users (in text or picture) and computer vision, we plan to explore an efficient way to collect dietary factors in the future.

Bibliography

[1] C. Fryar, Y. Ostchega, C. Hales, G. Zhang, and D. Kruszon-Moran, "Hypertension prevalence and control among adults: United States, 2015–2016," *NCHS data brief*, no.289, 2017.

[2] N. Covassin and P. Singh, "Sleep duration and cardiovascular disease risk: epidemiologic and experimental evidence," *Sleep medicine clinics*. vol. 11, no. 1, pp. 81-89, 2016.

[3] V. Cornelissen and N. Smart, "Exercise Training for Blood Pressure: A Systematic Review and Meta-analysis," *Journal of the American Heart Association*, 2013.

[4] G. Odedegbe and T. Pickering, "Principles and techniques of blood pressure measurement," *Cardiology Clinics*, vol. 28, no. 4, pp. 571–586, 2010.

[5] R. Mukkamala, Hahn JO, Inan OT, Mestha LK, Kim CS, Töreyin H, Kyal S., "Toward ubiquitous blood pressure monitoring via pulse transit time: Theory and practice," *IEEE Trans. Biomed. Eng.,* vol. 62, no. 8, pp. 1879–1901, Aug. 2015.

[6] L. Breiman, "Random Forest," *Machine Learning*, vol. 45, no.1, pp. 5–32, 2001

[7] M. Fernández-Delgado, E. Cernadas, S. Barro, D. Amorim, "Do we need hundreds of classifiers to solve real-world classification problems?" *The Journal of Machine Learning Research*, vol. 15, no. 1, pp.3133-3181, 2014.

[8] R. Genuer, J. Poggi and C. Tuleau-Malot, "Variable selection using random forests," *Pattern Recognition Letters*, 2010.

[9] P. Chiang and S. Dey, "Personalized Effect of Health Behavior on Blood Pressure: Machine Learning Based Prediction and Recommendation," *in Proc. of IEEE International Conference on E-health Networking, Application & Services (Healthcom'18),* Ostrava, Czech, 2018.

[10] H. Golino, L. Amaral, and S. Duarte, "Predicting Increased Blood Pressure Using Machine Learning," *Journal of Obesity*, 2014.

[11] B. Ballinger, Hsieh, J., Singh, A., Sohoni, N., Wang, J., Tison, G.H., Marcus, G.M., Sanchez, J.M., Maguire, C., Olgin, J.E. and Pletcher, M.J., "DeepHeart: Semi-Supervised Sequence Learning for Cardiovascular Risk Prediction," i*n Proc. of the 32nd AAAI Conference on Artificial Intelligence*, New Orleans, LA, USA, 2018.

[12] T. Wu, G. Pang, and E. Kwong, "Predicting Systolic Blood Pressure Using Machine Learning," *in Proc. of International Conference on Information and Automation for Sustainability*, Colombo, Sri Lanka, 2014.

[13] B. Zhang, Z. Wei, J. Ren, Y. Cheng, and Z. Zheng, "An Empirical study on Predicting Blood Pressure using Classification and Regression Trees," *in IEEE Access*, pp. 21758-21768, 2018.

[14] P. Su, X. Ding, Y. Zhang, J. Liu, F. Miao and Ni Zhao "Long-term Blood Pressure Prediction with Deep Recurrent Neural Networks," *IEEE EMBS International Conference on Biomedical & Health Informatics (BHI),* Las Vegas, NV, 2018, pp. 323-328.

[15] Abrar, S., Tahir, G.A., Kakudi, H.A. and Loo, C.K., "A Personalized Blood Pressure Prediction Model Using Recurrent Kernel Extreme Reservoir Machine." *Future of Information and Communication Conference.* Springer, Cham, 2019.

[16] X. Li, S. Wu, & L. Wang, "Blood Pressure Prediction via Recurrent Models with Contextual Layer," *in Proc. of the International Conference on World Wide Web*, Perth, Australia, 2017.

[17] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachi, "A survey on concept drift adaptation." *ACM computing surveys (CSUR)* vol. 46, no.4, pp. 44, 2014.

[18] G. Ditzler, M. Roveri, C. Alippi and R. Polika, "Learning in nonstationary environments: A survey," *IEEE Computational Intelligence Magazine*, vol. 10, no.4, pp. 12-25, 2015.

[19] Bifet, Albert, and Ricard Gavalda. "Learning from time-changing data with adaptive windowing." *In Proc. of 2007 SIAM international conference on data mining*, Minneapolis, Minnesota, USA,2007.

[20] J. Gama, P. Medas, G. Castillo, P. Rodrigues, "Learning with drift detection," *in Brazilian symposium on artificial intelligence,* vol. 3171, pp. 286-295, 2004

[21] Minku, L.L., and Yao, X., 2011. DDD: A new ensemble approach for dealing with concept drift. *IEEE transactions on knowledge and data engineering*, *24*(4), pp.619-633.

[22] B. Lakshminarayanan, D. Roy, and Y. Teh, "Mondrian forests: Efficient online random forests." *In Advances in neural information processing systems*, pp. 3140-3148, 2014.

[23] T. Zhang, "Solving large scale linear prediction problems using stochastic gradient descent algorithms." *in Proc. of the twenty-first international conference on Machine learning*, Banff, Canada, 2004.

[24] Krawczyk, B., Minku, L.L., Gama, J., Stefanowski, J. and Woźniak, M., "Ensemble learning for data stream analysis: A survey." *Information Fusion*, vol. 37, pp. 132-156, 2017.

[25] J. Z. Kolter and M. A. Maloof, "Dynamic weighted majority: an ensemble method for drifting concepts," *Journal of MachineLearning Research,* vol. 8, pp. 2755–2790, 2007.

[26] Brzezinski, Dariusz, and Jerzy Stefanowski. "Reacting to different types of concept drift: The accuracy updated ensemble algorithm." *IEEE Transactions on Neural Networks and Learning Systems* vol. 25.1 pp. 81-94, 2013.

[27] Schapire, Robert E. "The boosting approach to machine learning: An overview" in *Nonlinear Estimation and Classification*, pp. 149-171. Springer, New York, NY, 2003.

[28] N. C. Oza, “Online bagging and boosting,” in Proc. of *IEEE international conference on systems, man and Cybernetics*, pp. 2340–2345, Oct. 2006.

[29] Fitbit, “What are active minutes?” https://help.fitbit.com/articles/en_US/Help_article/1379/

[30] G. E. Batista and M. C. Monard, “An analysis of four missing data treatment methods for supervised learning,” *Applied Artificial Intelligence*, vol, 17, no. 5-6, pp. 519–533, 2003

[31] W. Loh, “Classification and regression trees.” *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(1), pp.14-23.

[32] B. Efron and R. Tibshirani, “An introduction to the bootstrap,” *CRC Press*, 1994.

[33] Behnamian, A., Millard, K., Banks, S.N., White, L., Richardson, M. and Pasher, J., “A Systematic Approach for Variable Selection with Random Forests: Achieving Stable Variable Importance Values: Achieving Stable Variable Importance Values,” *in IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 11, pp. 1988-1992, Nov. 2017.

[34] “scikit-learn Machine Learning in Python,” 2015, http://scikit-learn.org/.

[35] F. Chollet, “Keras,” 2015.

[36] M. Halford, “Crème,” https://creme-ml.github.io/index.html

[37] Suykens, Johan AK, and Joos Vandewalle. "Least squares support vector machine classifiers." *Neural processing letters,* vol. 9, no. 3, p.p. 293-300, 1999.

[38] D. Rumelhart, G. Hinton, and R. Williams, "Learning representations by back-propagating errors." *Cognitive modeling*, vol. 5, no. 3, 1988.

[39] S. Hochreiter, and J. Schmidhuber, "Long short-term memory," *Neural computation,* vol, 9, no. 8, p.p. 1735-1780, 1998

[40] D. Kingma and J. Ba, “Adam: A method for stochastic optimization.” *arXiv preprint arXiv:1412.6980*, 2014

[41] J. Gama, R. Sebastião, and P. Rodrigues, “Issues in evaluation of stream learning algorithms." in *Proc. of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, Paris, France, 2009.

[42] Appel, L.J., Champagne, C.M., Harsha, D.W., Cooper, L.S., Obarzanek, E., Elmer, P.J., Stevens, V.J., Vollmer, W.M., Lin, P.H., Svetkey, L.P. and Young, D.R., “Effects of comprehensive lifestyle modification on blood pressure control: main results of the PREMIER clinical trial,” *Journal of the American Medical Association,* vol *289*, pp. 2083–2093, 2003

[43] K. Doughty, N. Del Pilar, A. Audette, D. Katz, “Lifestyle Medicine and the Management of Cardiovascular Disease,” *Current cardiology reports*, vol. 19, no. 11, pp. 116, 2017

[44] A. Siu, "Screening for high blood pressure in adults: US Preventive Services Task Force recommendation statement," *Annals of internal medicine*. vol. 163, no. 10, pp. 778-786, 2015.

[45] P. Chiang and S. Dey, "Offline and Online Learning Techniques for Personalized Blood Pressure Prediction and Health Behavior Recommendations," *in IEEE Access*, vol. 7, pp. 130854-130864, 2019

[46] G. Box, G. Jenkins, G. Reinsel, and G. Ljung, "Time series analysis: forecasting and control," *John Wiley & Sons*, 2015

[47] L. Shapley, "A value for n-person games." *Contributions to the Theory of Games*, vol. 2, no. 28, pp. 307-317, 1953.

[48] N. Hasanzadeh, M. M. Ahmadi and H. Mohammadzade, "Blood Pressure Estimation Using Photoplethysmogram Signal and Its Morphological Features," in *IEEE Sensors Journal*.

[49] G. Slapničar, M. Luštrek, and M. Marinko, "Continuous Blood Pressure Estimation from PPG Signal," *Informatica*, vol. 42, no. 1, 2018.

[50] S. Shimazaki, H. Kawanaka, H. Ishikawa, K. Inoue and K. Oguri, "Cuffless Blood Pressure Estimation from only the Waveform of Photoplethysmography using CNN," in *IEEE International Conference of the Engineering in Medicine and Biology Society (EMBC)*, Berlin, Germany, 2019.

[51] JE. Sharman and A. LaGerche "Exercise blood pressure: clinical relevance and correct measurement." *Journal of human hypertension* vol. 29, no. 6 pp. 351-358, 205.

[52] W. Sacre, G. Jennings, and B. Kingwell. "Exercise and dietary influences on arterial stiffness in cardiometabolic disease." *Hypertension*, vol. 63, no. 5: pp.888-893, 2014.

[53] M. Gold, I. Dziobek, K. Rogers, A. Bayoumy, P. McHugh, and A. Convit. "Hypertension and hypothalamo-pituitary-adrenal axis hyperactivity affect frontal lobe integrity." *The Journal of Clinical Endocrinology & Metabolism,* vol 90, no. 6, pp.3262-3267, 2005.

[54] Sacks, F.M., Svetkey, L.P., Vollmer, W.M., Appel, L.J., Bray, G.A., Harsha, D., Obarzanek, E., Conlin, P.R., Miller, E.R., Simons-Morton, D.G. and Karanja, N., "Effects on blood pressure of reduced dietary sodium and the Dietary Approaches to Stop Hypertension (DASH) diet," *New England journal of medicine*, vol. 344, no.1, pp. 3-10, 2001.

[55] McClung, H.L., Ptomey, L.T., Shook, R.P., Aggarwal, A., Gorczyca, A.M., Sazonov, E.S., Becofsky, K., Weiss, R. and Das, S.K., "Dietary intake and physical activity assessment: current tools, techniques, and technologies for use in adult populations," *American Journal of Preventive Medicine,* vol. 55, no. 4, pp 93-104, 2018.

[56] Carey, Robert M., Paul K. Whelton, and 2017 ACC/AHA Hypertension Guideline Writing Committee, "Prevention, Detection, Evaluation, and Management of High Blood Pressure in Adults: Executive Summary: A Report of the American College of Cardiology/American Heart

Association Task Force on Clinical Practice Guidelines," *Journal of the American College of Cardiology*, vol. 71, no. 19, pp. 2273-2275, 2018

[57] Elshawi R, Al-Mallah MH, Sakr S., "On the interpretability of machine learning-based model for predicting hypertension." *BMC medical informatics and decision making,* vol. 19, no. 1, pp. 146, 2019.

[58] N. Guthrie, M. Berman, K. Edwards, K. Appelbaum, S. Dey, J. Carpenter, D. Eisenberg and D. Katz, "Achieving Rapid Blood Pressure Control With Digital Therapeutics: Retrospective Cohort and Machine Learning Study," *JMIR Cardio,* vol. 3, no. 1, pp. e13030, 2019

[59] American Heart Association, "Target Heart Rates Chart," https://www.heart.org/en/healthy-living/fitness/fitness-basics/target-heart-rates

[60] A. Guin, "Travel time prediction using a seasonal autoregressive integrated moving average time series model." in *IEEE Intelligent Transportation Systems Conference*, pp. 493-498, Toronto, Canada, 2006.

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

[62] S. Wang, C. Aggarwal, and H. Liu. "Using a random forest to inspire a neural network and improving on it." In *Proceedings of the SIAM international conference on data mining*. Houston, TX, USA, 2017.

[63] W. Loh, "Classification and regression trees." *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1, no. 1, pp.14-23.

[64] M. Hall, "Correlation-based feature selection for discrete and numeric class machine learning," *Proceedings of the Seventeenth International Conference on Machine Learning*, pp. 359–366, 2000.

[65] S. Lei, "A feature selection method based on information gain and genetic algorithm," in *International Conference on Computer Science and Electronics Engineering*, vol. 2, pp. 355-358, 2012.

[66] S. Cohen, E. Ruppin and G. Dror, "Feature Selection Based on the Shapley Value," in *International Joint Conferences on Artificial Intelligence*, vol. 5, pp. 665-670. 2005.

[67] S. Lundberg and S. Lee, "A unified approach to interpreting model predictions," in *Advances in neural information processing systems*, pp. 4765-4774. 2017.

[68] C. Molnar, "Interpretable machine learning. A Guide for Making Black Box Models Explainable", 2019. https://christophm.github.io/interpretable-ml-book/.

[69] Lundberg, Scott M., Gabriel G. Erion, and Su-In Lee. "Consistent individualized feature attribution for tree ensembles." *arXiv preprint arXiv*, pp.1802.03888, 2018.

[70] R. Hyndman and Y. Khandakar, “Automatic time series forecasting: the forecast package for R,” *Journal of Statistical Software*, vol. 26, no. 3, pp. 1–22, 2008

[71] Welch, Bernard L, "The generalization of student's' problem when several different population variances are involved," *Biometrika*, vol.34, no.1, p.p. 28-35, 1947.

[72] GeSI, “SMARTer2020 report,” 2013, http://gesi.org/SMARTer2020.

[73] Nokia, “Technology Vision 2020: Flatten network energy consumption,” 2013, http://networks.nokia.com/file/flatten-network-energy-white-paper.

[74] D. Feng, C. Jiang, G. Lim, L. J. Cimini Jr., G. Feng, and G.Y. Li, "A Survey of Energy-Efficient Wireless Communications," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 167-178, First Quarter 2013.

[75] Z. Xu, C. Yang, G. Y. Li, S. Zhang, Y. Chen and S. Xu, "Energy-Efficient Configuration of Spatial and Frequency Resources in MIMO-OFDMA Systems," *in IEEE Transactions on Communications*, vol. 61, no. 2, pp. 564-575, February 2013.

[76] Q. Wu, W. Chen, M. Tao, J. Li, H. Tang and J. Wu, "Resource Allocation for Joint Transmitter and Receiver Energy Efficiency Maximization in Downlink OFDMA Systems," in *IEEE Transactions on Communications*, vol. 63, no. 2, pp. 416-430, Feb. 2015.

[77] R. Guruprasad, K. Son, and S. Dey, “Power-efficient Base Station Operation through User QoS-Aware Adaptive RF Chain Switching Technique,” *IEEE International Conference on Communications (ICC)*, June 2015.

[78] K. Son, H. Kim, Y. Yi, and B. Krishnamachari, “Base Station Operation and User Association Mechanisms for Energy-Delay Tradeoffs in Green Cellular Networks,” *IEEE Journal Selected Areas of Communication*., vol. 29, no. 8, pp. 1525–1536, Sept. 2011.

[79] M. L. Ku, W. Li, Y. Chen and K. J. Ray Liu, "Advances in Energy Harvesting Communications: Past, Present, and Future Challenges," in *IEEE Communications Surveys & Tutorials*, vol. 18, no. 2, pp. 1384-1412, Secondquarter 2016.

[80] Q. Wu, G. Y. Li, W. Chen, D. W. K. Ng and R. Schober, "An Overview of Sustainable Green 5G Networks," *in IEEE Wireless Communications,* vol. 24, no. 4, pp. 72-80, 2017.

[81] M. Patterson, N. F. Macia, and A. M. Kannan, “Hybrid microgrid model based on solar photovoltaic battery fuel cell system for intermittent load applications,” *in IEEE Transactions of Energy Conversion*, vol. 30, no. 1, pp. 359–366, Mar. 2015..

[82] Alcatel-Lucent Bell Labs, “Alternative Renewable Power Concepts for Alcatel-Lucent Outdoor Wireless Small Cell Solutions,” Jan. 2011.

[83] J. Gong, S. Zhou, and Z. Niu, “Optimal Power Allocation for Energy Harvesting and Power Grid Coexisting Wireless Communication Systems,” *IEEE Transactions on Communications*, vol.61, no.7, pp.3040-3049, July 2013.

[84] X. Huang and N. Ansari, "Energy sharing within EH-enabled wireless communication networks," *in IEEE Wireless Communications*, vol. 22, no. 3, pp. 144-149, June 2015.

[85] T. Zhang, W. Chen, Z. Han and Z. Cao, "A cross-layer perspective on energy harvesting aided green communications over fading channels," *in Proc. of IEEE INFOCOM*, Turin, 2013, pp. 3235-3230.

[86] Y. Mao, J. Zhang and K. B. Letaief, "A Lyapunov Optimization Approach for Green Cellular Networks with Hybrid Energy Supplies*," IEEE Journal on Selected Areas in Communications*, vol. 33, no. 12, pp. 2463-2477, Dec. 2015.

[87] K. Tutuncuoglu, A. Yener, and S. Ulukus, "Optimum policies for an energy harvesting transmitter under energy storage losses," *IEEE Journal of Selected Areas of Communication*., vol. 33, no. 3, pp. 467–481, Mar. 2015

[88] P. Chiang, R. Guruprasad and S. Dey, "Renewable Energy-Aware Video Download in Cellular Networks," *IEEE Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Hong Kong, Sept. 2015

[89] M. J. Farooq, H. Ghazzai, A. Kadri, H. ElSawy, and M.-S. Alouini, "A Hybrid Energy Sharing Framework for Green Cellular Networks," *in IEEE Transactions on Communications*, vol. 65, no. 2, pp. 918-934, Feb. 2017.

[90] M. J. Farooq, H. Ghazzai, and A. Kadri, "Energy sharing framework for microgrid-powered cellular base stations", *in Proc. of IEEE GLOBECOM*, Washington, D.C., USA, Dec. 2016.

[91] X. Huang, T. Han and N. Ansari, "On Green-Energy-Powered Cognitive Radio Networks," *in IEEE Communications Surveys & Tutorials*, vol. 17, no. 2, pp. 827-842, Second quarter 2015.

[92] B. Devillers and D. Gunduz, "A general framework for the optimization of energy harvesting communication systems with battery imperfections," *Journal of Communications and Networks*., vol. 14, no. 2, pp. 130–139, Apr. 2012.

[93] J. Yang and S. Ulukus, "Optimal packet scheduling in an energy harvesting communication system," *IEEE Transactions on Communications*, vol. 60, no. 1,pp. 220–230, Jan. 2012.

[94] M. Neely, "Stochastic network optimization with application to communication and queueing systems." *Synthesis Lectures on Communication Networks,* 3.1, pp. 1-211, 2010.

[95] J. Yang, Q. Yang, Z. Shen, and K. S. Kwak, "Suboptimal online resource allocation in hybrid energy supplied OFDMA cellular networks," *IEEE Commun. Letter.,* vol. 20, no. 8, pp. 1639–1642, 2016

[96] X. Wang, T. Ma, R. Zhang and X. Zhou, "Stochastic Online Control for Energy-Harvesting Wireless Networks with Battery Imperfections," *IEEE Transactions on Wireless Communications*, vol. 15, no. 12, pp. 8437-8448, Dec. 2016.

[97] CISCO, “Cisco visual networking index: Global mobile data traffic forecast update, 2013-2018,” 2014.

[98] Dynamic Adaptive Streaming Over HTTP (DASH)—Part 1: Media Presentation Description and Segment Formats, Int. Std. TS 26.247 V11.2.0, Apr. 2012, ISO/IEC.

[99] M.A. Hoque, M. Siekkinen, and J.K. Nurminen, “Energy Efficient Multimedia Streaming to Mobile Devices — A Survey,” *IEEE Communications Surveys & Tutorials*, vol.16, no.1, pp.579-597, First Quarter 2014.

[100] R. Guruprasad and S. Dey, “Rate adaptation and base station reconfiguration for battery efficient video download,” *in Proc. of IEEE WCNC*, Shanghai, China, Apr. 2013.

[101] H. Abou-zeid, H.S. Hassanein, and S. Valentin, “Energy-Efficient Adaptive Video Transmission: Exploiting Rate Predictions in Wireless Networks,” *IEEE Transactions on Vehicular Technology*, vol.63, no.5, pp.2013-2026, June 2014.

[102] A. Kwasinski and A. Kwasinski, “Traffic management for sustainable LTE networks,” *in Proc. of IEEE GLOBECOM*, Austin, Texas, USA, 2014.

[103] L. Huang and M. J. Neely, “Utility optimal scheduling in energy harvesting networks,” *IEEE/ACM Transactions on Networking*, vol. 21, no. 4, pp. 1117–1130, Aug. 2013.

[104] Y. F. Liu and Y. H. Dai, "On the Complexity of Joint Subcarrier and Power Allocation for Multi-User OFDMA Systems," *in IEEE Transactions on Signal Processing*, vol. 62, no. 3, pp. 583-596, Feb.1, 2014.

[105] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hoßfeld and P. Tran-Gia, "A Survey on Quality of Experience of HTTP Adaptive Streaming," in *IEEE Communications Surveys & Tutorials*, vol. 17, no. 1, pp. 469-492, 2015.

[106] F. Kelly, A. Maulloo and KH Tan, "Rate control for communication networks: shadow prices, proportional fairness and stability." *Journal of the Operational Research Society*, vol. 49.3, pp. 237-252, 1998.

[107] O.Arnold, F. Richter, G. Fettweis, and O. Blume, “Power consumption modeling of different base station types in heterogeneous cellular networks,” *IEEE Future Network and Mobile Summit*, 2010.

[108] 3GPP, “Physical channels and modulation,” Evolved Universal Terrestrial Radio Access (E-UTRA), TS 36.211, 2011

[109] H. Sun, B. Wang, R. Kapoor, S. Sambhwani, and M. Scipione, "Introducing heterogeneous networks in HSPA." *IEEE International Conference on Communications (ICC)*, Ottawa, Canada, 2012.

[110] P. Chiang, R. Guruprasad and S. Dey, "Optimal Use of Harvested Solar, Hybrid Storage and Base Station Resources for Green Cellular Networks," *IEEE Transactions on Green Communications and Networking*, vol. 2, no. 3, pp. 707-720, Sept. 2018.