AUTOMATED MOBILE SYSTEMS FOR PERSONALIZED HEALTH FEEDBACK

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Mohammod Mashfiqui Rabbi Shuvo

May 2016

AUTOMATED MOBILE SYSTEMS FOR PERSONALIZED HEALTH FEEDBACK

Mohammod Mashfiqui Rabbi Shuvo, Ph.D.
Cornell University 2016

In recent years, we have seen a prolific rise of mobile and wearable computing in healthcare and fitness. Rich user interfaces have made manual logging easier, and sensors have made tracking effortless. However state-of-the-art feedback technologies are still limited to either providing an overall status, lucrative visualization of data or simple tailoring based on age, gender or overall health status. It is possible to go beyond these paradigms and take advantage of more fine-grained information contained in the data. To this end, we created MyBehavior, a smartphone application that takes a novel approach to generate deeply personalized health feedback. MyBehavior automatically learns a user's physical activity and dietary behavior, and strategically suggests changes to those behaviors for a healthier lifestyle. The system uses a sequential decision making algorithm, Multi-armed Bandit, to generate suggestions that maximize calorie loss and are easy for the user to adopt. In addition, the system takes into account users preferences to encourage adoption using the pareto-frontier algorithm. A 14-week study with MyBehavior shows statistically significant increases in physical activity and decreases in food calorie when using MyBehavior compared to a control condition. I also discuss several lessons learnt and implications for future design of similar health feedback systems.

This document is dedicated to my loving parents, lovely siblings and
my friends/colleagues.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Personalized health recommendation: A unexplored space for mobile computing

Recently mobile and wearable computing have ushered in a new era of health research by enabling data collection in-situ. Modern sensors in smartphones and wearable devices are capable to unobtrusively track an individual's physical activity, social interactions and life contexts [88][127]. The variety of the sensors are also increasing, and in the process enabling the capture of even more details of a user's activities (e.g., eating moments [1][153], heart rate [1], attention [2]). Information that can not be captured with sensors; e.g., complex exercises, emotions [121] or food intakes [111]; can be manually recorded with easy-to-use apps. In conjunction to these ongoing data collection efforts, research has also progressed to analyze the collected data, and find behavioral patterns that relate to health and well-being. Rabbi et al. [124], for instance, showed the amount of daily face-to-face interaction from phone audio can inform depressive symptoms. Wang et al. [155], in their StudentLife project, demonstrated that different sensor and self-reported measures can correlate to physical activity, emotional state, productivity and performance of students.

Although the existing works to data collection and analysis are necessary to understand health conditions, it is also important to close-the-loop and provide feedback based on the data. However if we look at the existing literature then we will see that the mobile technologies for health feedback are still at its infancy. There is no system, prior to this dissertation, to provide personalized and actionable recommendation on ***what to do***, once a user's data is collected and health conditions are understood. i.e., if a user is less active, for instance, then current health feedback systems can not suggest specific exercises or physical activity that are applicable for the user given her lifestyle. If we look at domains outside of health then we will see a different picture; there are systems that can suggest ***what to do***. For example, Google can suggest webpages based on a search query that also matches a user's past preferences. Similarly, Pandora and Netflix can recommend songs and movies respectively which match a user's past behaviors.

To this end, this thesis explored the creation of an automated system to personalize health recommendations from phone data. The system is called MyBehavior and it works in the following steps: first, MyBehavior monitors a user's physical activity and food intake. Then MyBehavior *understands* common physical activity and dietary behavior by grouping similar activities. For instance, walking in the office or consumption of similar foods (e.g., pizzas) would be grouped together. Subsequently, MyBehavior issues *personalized* suggestions that relate to specific user behaviors: e.g., to walk more near the office or to avoid eating pizza.

The later chapters of this dissertation will largely focus on the details of the design and implementation of MyBehavior. In this chapter, I take the opportunity to discuss the motivation and feasibility of personalizing health suggestions with mobile data. Specifically I will discuss (1) "Are personalized recommendations necessary for health feedback?" I will contend that personalization is in fact necessary for health recommendations by grounding my argument in relevant literature in psychology and social science (2) I will argue that mobile data provide a novel opportunity of personalizing health recommendations that is different from prior related works (3) Then I will briefly outline on how to create and evaluate such a personalized health suggestions generation system (4) Finally, I will conclude by giving a brief description of MyBehavior as an instance of personalized health recommender system.

1.2 Importance of personalized health recommendation

> *"Its far more important to know what person the disease has than what disease the person has." —– Hippocrates [4]*

Personalization is generally a desired feature for any user data driven system. However personalization can be relevant to health feedback for two reasons. First, each individual is unique with distinct characteristics and behaviors. Personalized feedback can address such individual uniqueness and hence has the potential to be more effective. Secondly, personalized feedbacks can be easy to adhere to, since they relate to distinct individual needs and behaviors. In the following, I will describe these two points in detail. I will also provide relevant theories from the literature to support the two points. Subsequent chapter 5 and 6 will provide stronger quantitative and qualitative evidence of the efficacy of personalization.

1.2.1 Uniqueness of individuals

Proponents of small data [48], personal (precision) medicine [4][54] and N-of-1 interventions [141][65] contend that each individual is unique and heterogeneous because of their culture, age, gender, childhood development and contexts in life. Therefore one-size-fits-all interventions/treatments may not fit or apply to everybody. Since the goal of any intervention or treatment is to maximize individual benefits, it is important to consider individual differences. To this end, N of 1 intervention/treatments first collect evidence based information for an individual. The information is then analyzed to create interventions that work best for the individual.

Evidence of such individual differences is not unique to medical literatures. Other disciplines also support the notion of inter-personal variability. For instance, *idiographic* research in psychology pertains to the studies of an individual's unique agency and life, whereas *nomothetic* describes studies of classes of population where a subject is seen as an exemplar [64]. Similar to psychology, *logical positivists* in social science seek for objective laws, properties or principles in society while *constructivists* argue that there is no objective truth in society: social life or contexts are relative and subjective [129].

Acknowledging the importance of individual uniqueness in health intervention, personalization can tap into the individual uniqueness and can provide solutions that optimize individual needs. Importantly, modern mobile or wearable devices can collected deeply personal health data, and it is possible to provide unique personalized recommendation for each individual. Such personalization can be potentially highly applicable to the user.

1.2.2 Increased adherence

Any personalization scheme needs to learn a user's behaviors/preferences first, and subsequently personalize experiences that reflect the learnt behaviors/preferences. For instance, if Google Now [112] sees a user to browse American Football or Presidential debate related news then it will start recommending related articles. A personalized health recommender systems for health can do the same and suggest actions that already relate to a user's behavior. For instance, if a user already walks near his office then a personalized suggestion could be to walk more or walk faster near office. Similarly a suggestion can be issued to avoid a unhealthy food if the system sees the user to eat the food repeatedly.

There is an additional benefit of relating suggestions to existing user behaviors: the suggestions often involve activities that the user is already doing or can do with small changes. Therefore the suggestions are potentially easy-to-follow since they relate to the user's daily routine. Specifically, in comparison to generic non-personalized health suggestions namely "walk 30 minutes to be active", "go for a movie to reduce boredom" or "eat fish for dinner", personalized suggestions are arguably more actionable and, if acted on, can provide desired health outcome with less effort. Furthermore, the concepts of low barrier or low effort are well documented in behavior change literature. BJ Fogg's behavior model [52] argues that if a system wants to persuade its users to undertake an action then the action needs to be low effort or easy. Similar to Fogg's behavior model, Health belief model [68] and theory of planned behavior [13] use the concept of "barrier" and advocate to remove barriers in order to make a behavior change. Since personalized suggestions already relate to a user's behavior or routine, there is less barrier to entry which can aid the behavior change process.

1.3 Health feedback and mobile computing

Despite the significance of personalizing health feedback discussed above, there is no system that can automatically personalize health suggestions in order to reach a daily goal or maintain a healthier lifestyle. Current health feedback systems are limited to either providing overall health status, attractive visualization of data or generic one-size-fits-all recommendations. However it is possible to go beyond these paradigms, and extract patterns of behaviors in mobile data, and subsequently personalize suggestions at an individual level. In this section, I will discuss several examples of behavioral patterns that can be extracted from mobile data. These behavioral patterns can be subsequently used to personalize suggestions that relates to the user's life. Before going to that discussion, I will describe prior works in health feedback, which will help situate the novelty of my proposed personalized health recommendation scheme.

1.3.1 Prior work

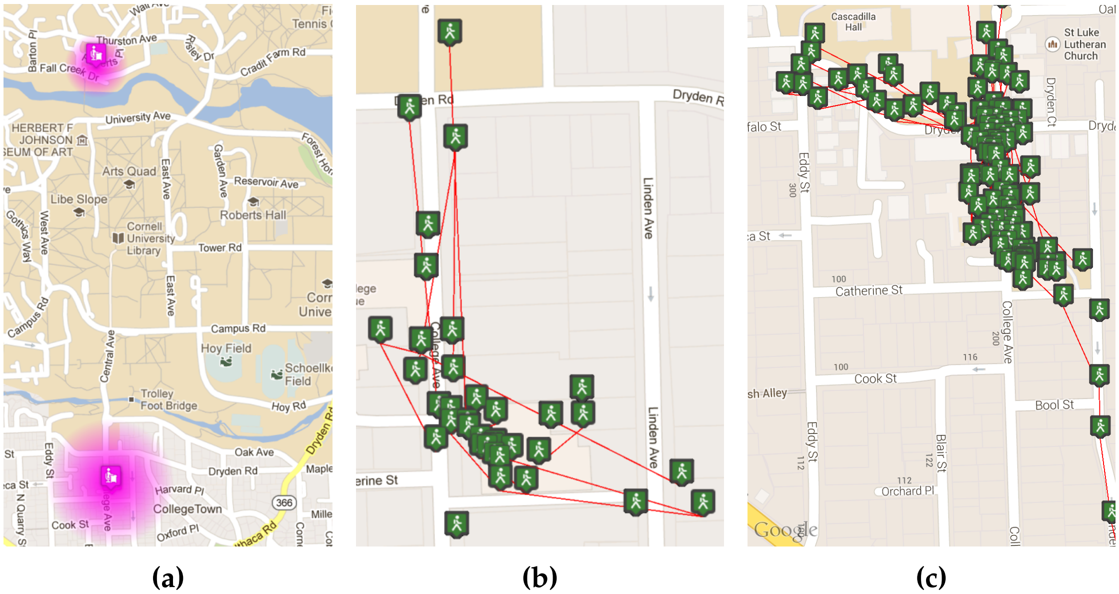
Existing health feedback technologies for mobile phones can be broadly characterized into three categories. They are as follows:

- **Overall feedback:** This category of feedback technology converts all the collected data across health dimension in one data point (e.g., step-count). The Ubifit app by Consolvo et al. [38], for instance, provides visual feedback on overall progress of daily physical activity on the phone wallpaper. BeWell [89] provides health feedback for overall physical activity, social interaction and sleep. Functionally, the overall feedback primes the user to achieve the daily goals (e.g., ten thousand steps). However, they fall short in giving specific recommendations that can tell how to reach the goals.

- **Visualization of almost entire data:** Another category of feedback technology relies on the visualization of the tracked data [106][51]. These technologies rely on users to interpret the data and find actionable insights and to-dos. However, visualizing the entire data can potentially create an information overload, and user can misinterpret the data without appropriate domain knowledge (e.g., the knowledge of making a successful behavior change, or trying something hard/unusual that they can not sustain). Furthermore since the variety of sensors are increasing, the manual interpretation of multi-dimensional data visualization will continue to get harder. Therefore it is questionable whether only visualizing data is a workable strategy as we move into a future with more sensors and multi-dimensional data.
- **Generic non-personalized suggestions:** Some existing feedback technologies provide suggestions on what to do, but the suggestions are the same for the entire population or a segment of the population [119][93]. For instance, suggestions are often tailored based on age, gender, ethnicity or users overall lifestyle (e.g, whether a user still figuring out what actions to take vs. user already has a well-maintained active lifestyle [122]). e.g., at early stages of a smoking cessation program, negative messages often work better than positive messages [37]. However these methods ignore individual differences, which can be improved with personalization.

1.3.2 Personalizing health recommendation from mobile data

Does mobile data contain information of human behaviors that can be used for actionable suggestions? In the following, three cases of mobile data will be described, where we will demonstrate that mobile data can contain meaningful behavioral patterns. Subsequently these behavioral patterns can be utilized for actionable personalized health suggestions to improve well-being.

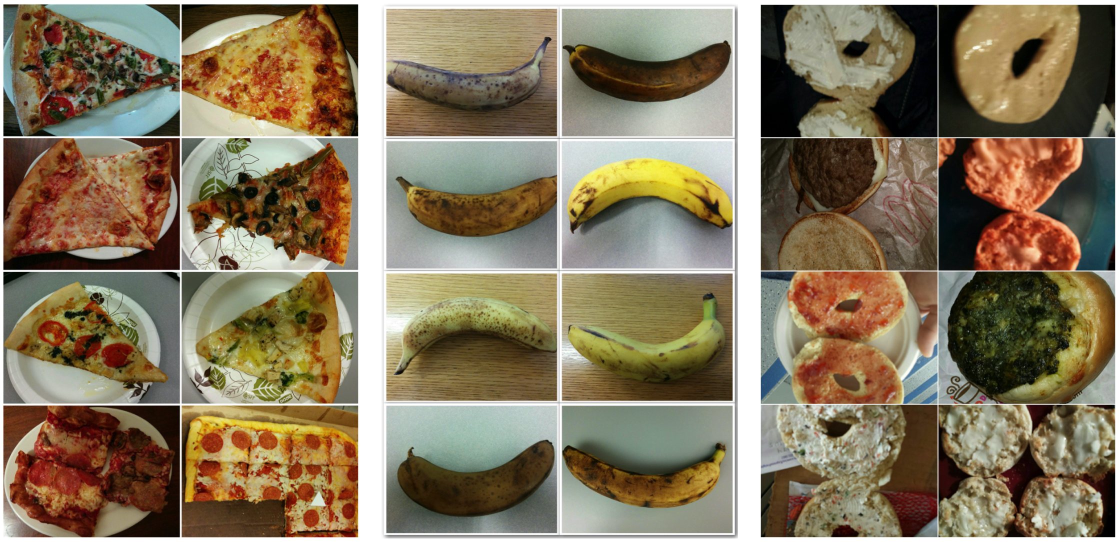


(a) (b) (c)

Figure 1.1: Visualization of user behaviors over a week (a) Heatmap of places a user stayed stationary (b) Location traces of frequent walks for the same user (c) Location traces of frequent walks for another user.

Physical Activity

Figure 1.1 shows a few physical activity behavior for two different users. Figure 1.1(a-b) show locations of behaviors of one user's stationary locations and a route of frequently taken short walks. One of the stationary location is the users's office and the other is the user's home. The walking traces represent the user's walks near her office. Figure 1.1c shows another walking behavior over a week for a different user. These examples demonstrate that physical activity traces from phones can inform behavior at an individual level. Such behaviors can be used to create personalized suggestions that can make users active easily (e.g., continue or increase walking near office).



(a) (b) (c)

Figure 1.2: Three separate dietary behaviors. (a) pizza eating behavior for a user (b) banana eating behavior for the same user (c) bagel eating behavior for another user

Food

Similar to physical activity, Figure 1.2 shows three separate dietary behaviors found in food journals of two different users. Figure 1.2a points a unhealthy behavior of repeated pizza eating. A personalized suggestion can be issued to avoid such unhealthy pizza eating behavior. On the other hand, Figure 1.2b's banana eating behavior is a healthy one and can be suggested to be continued. Finally, the repeated bagel eating behavior in Figure 1.2c can be suggested to be avoided.

Social interactions

Socialization is an important predictor of mental well-being. Earlier works have shown that the level of socialization is negatively correlated with depressive symptoms [124][155]. Furthermore, socialization and social support are important coping mechanisms for stress and depression [90].

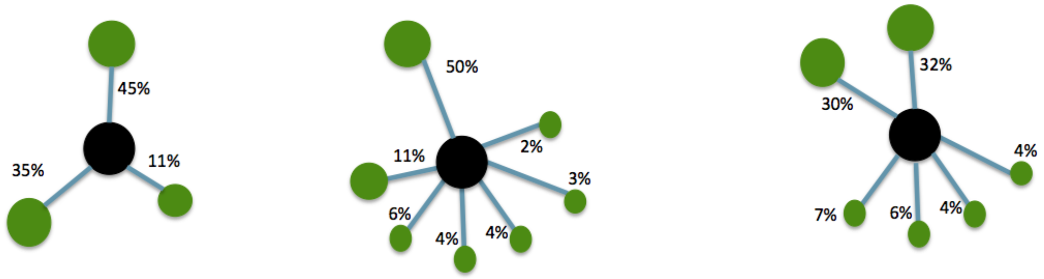


Figure 1.3: Distribution three users' SMS pattern

Our phones can capture our socialization level and can identify who we commonly communicate with [11][160]. Figure 1.3 shows a representative example of socialization, where the ego network of SMS pattern for 3 users are shown. The data is collected by Aharony et al. [11]. The black nodes represent users and the green nodes represent the receivers that the users have communicated over SMSes. The edge weight shows the percentage of a user's overall SMS that were sent to a receiver. The total edge weights of a user is near 80%, which means the user sent 80% of her SMS to the receivers included in the figure. In other words, the included receiver nodes can be considered as proxies to the social circle with whom the user largely (80%) communicates with. With such information, a personalized system for stress coping can suggest specific friends to talk to for social support [123].

1.4 The challenges of creating a health recommender system

The three evidences in the above section demonstrate encouraging examples of the potential of mobile data for personalizing health recommendation. Nevertheless, the creation of a personalized health recommender system requires careful design decisions, system building and evaluation. The creation process can often span multiple years of research and development. In the following, I will give an overview of the necessary steps to build a personalized health recommender system.

1.4.1 Designing health recommendation from raw data

The biggest hurdle of a health recommender system is to transform the raw personal data into suggestions that can positively affect a health outcome. For instance, an app to promote physical activity needs to transform milli-second level accelerometer data into actionable suggestions. At a high level, the transformation can be divided into two stages (Figure 1.4). The first stage is to mine a user's behavioral patterns from raw data stream. The second stage involves the personalization of health suggestions that relate to a user's behaviors. I will describe these two stages in more detail below.

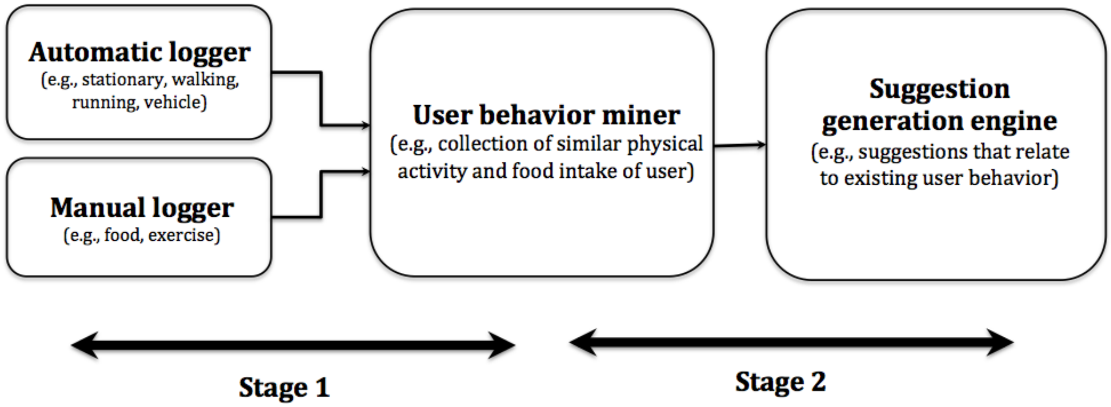

Automatic logger
(e.g., stationary, walking, running, vehicle)
Manual logger
(e.g., food, exercise)
User behavior miner
(e.g., collection of similar physical activity and food intake of user)
Suggestion generation engine
(e.g., suggestions that relate to existing user behavior)
Stage 1
Stage 2

Figure 1.4: Stages of transforming data into health feedback

Stage 1: From raw Data to Behavior

At the lowest level, phones record raw data that represent user activities with timestamps. e.g., a raw data point can be a 5 minute walk near office at

Nov, 3 2015 3:15PM. If the user has a habit or behavior of walking near the office then similar data points will appear multiple times in the raw data stream. Simple unsupervised clustering can group these repeated similar user behaviors. Figure 1.1 shows a few walking and stationary behaviors that are found by grouping repeated occurrences of similar activities from raw physical activity streams. Figure 1.5 shows grouping of similar food intakes.

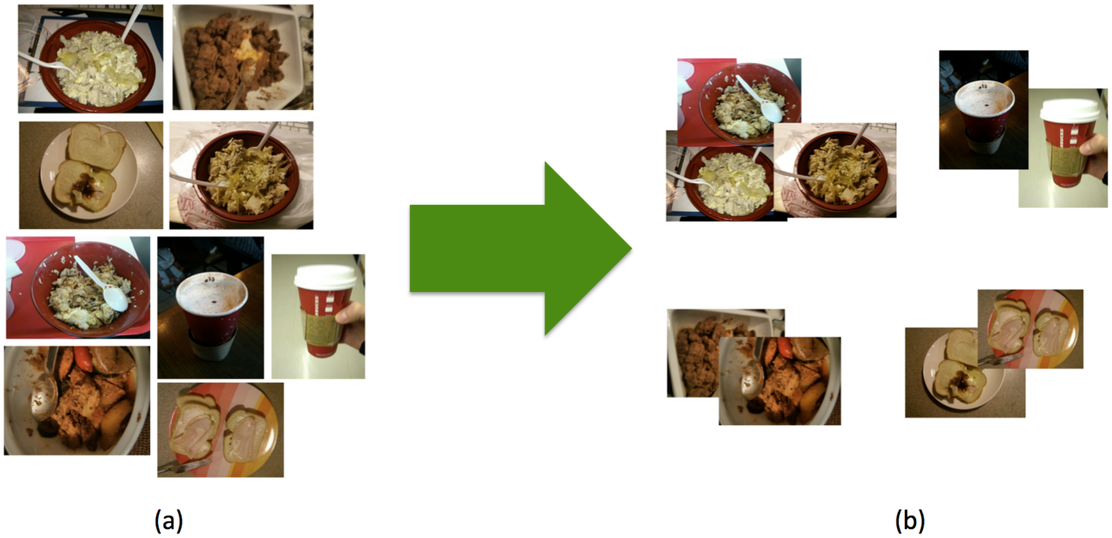
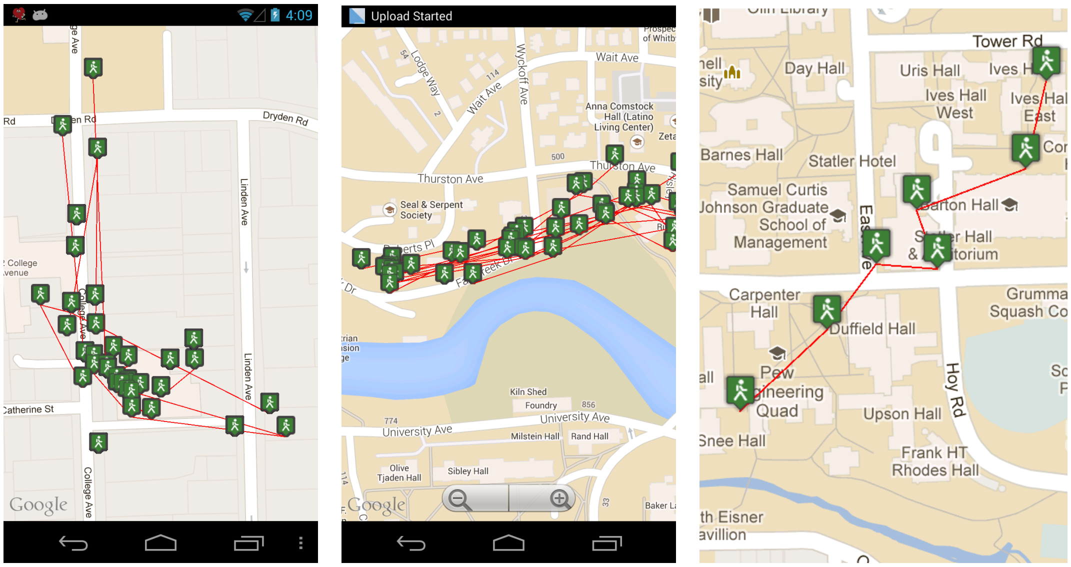


Figure 1.5: An example of transforming raw food logs in ‘a’ to eating behaviors in ‘b’

Although clustering similar activities can inform common user behaviors, the distance metric for clustering will depend on the kind of data (e.g., food, walking or SMS). For instance, if a user’s foods are tagged with food-ingredients then similar foods will have similar tags. A distance metric to determine similarity based on words, as commonly used in Natural Language Processing or Information Retrieval, can be useful to group foods that are similar [75]. On the other hand, in order to determine similar walking instances, a distance metric is needed to quantify similar walking trajectories. One such measure can come from hand-writing recognition literature, where trajectories of hand-written letters are matched with some canonical letters’ trajectories [147].

Stage 2: Recommendations that relate to user behaviors

After finding the relevant behaviors for different users, personalized recommendations can be generated that relate to the user behaviors. However many different behaviors can be found from a user's mobile data (e.g., all the different places a user might walk and all the foods the user might eat). Not all of these behaviors are good enough to be related to while providing suggestions. Therefore the suggestion generation process needs to focus on the behaviors that may accelerate the path of reaching a desired health outcome. e.g., a food recommender system can focus on suggestions that a user can do with small changes to existing food habits, like changing pizza eating behavior. In addition, not all the provided suggestions will be followed and the recommendation algorithm also needs to manage/co-ordinate a pool of effective suggestions that are followed upon.



(a) (b) (c)

Figure 1.6: Visualization of walking behaviors (a) walks near office (b) walks near home (c) a non-frequent walk

Choosing the appropriate behaviors for suggestions depends on the problem domain (e.g., from relevant literatures and theories from the domain). For instance, let us consider an app that wants to promote more physical activity. The app can suggest to undertake an activity which is low-effort or easy, as informed by theories in persuasion and behavior change [52][68]. Easy-to-do activities are often behaviors that user already do often or can do with small changes [86]. For example, assume the app found three separate walking behaviors of a user as shown in Figure 1.6. Figure 1.6a, 1.6b, and 1.6c respectively show frequent walking behaviors near office, near home and a non-frequent walk. The health feedback can prioritize the suggestions to walk more near office and home more, compared to Figure 1.6c, since they are lower-effort.

Theories can help to choose the recommendation, but one additional step is necessary to operationalize the theory into an app. The choice or ranking criteria needs to be embedded inside an algorithm. A way of doing so is to quantify the prioritization criteria as a ranking function of a decision making algorithm. Choosing an appropriate decision making algorithm is often easy, since there is already a collection of such algorithms in the decision theory, economics, reinforcement learning and recommender systems literature.

1.4.2 System implementation for real world use

The health recommender system will be implemented as a smartphone application. However, smartphones have limited batteries, therefore the recommendation generation algorithm needs to run efficiently that drains less battery. Less computationally heavy algorithms will be essential for commonly done operations. Since most of these algorithms often do machine learning, the online or incremental learning can be used for low computation and higher battery efficiency [32]. This is because, online algorithms only need to analyze the latest data point to learn, which in turn requires less computation. Furthermore, such online learning can often perform similar to a computationally heavy batch learning scheme. In fact, recent literature shows that online learning can be as good as batch learning if appropriate strategies are used for making incremental update to the classifier [22].

1.4.3 Iterative design and evaluation

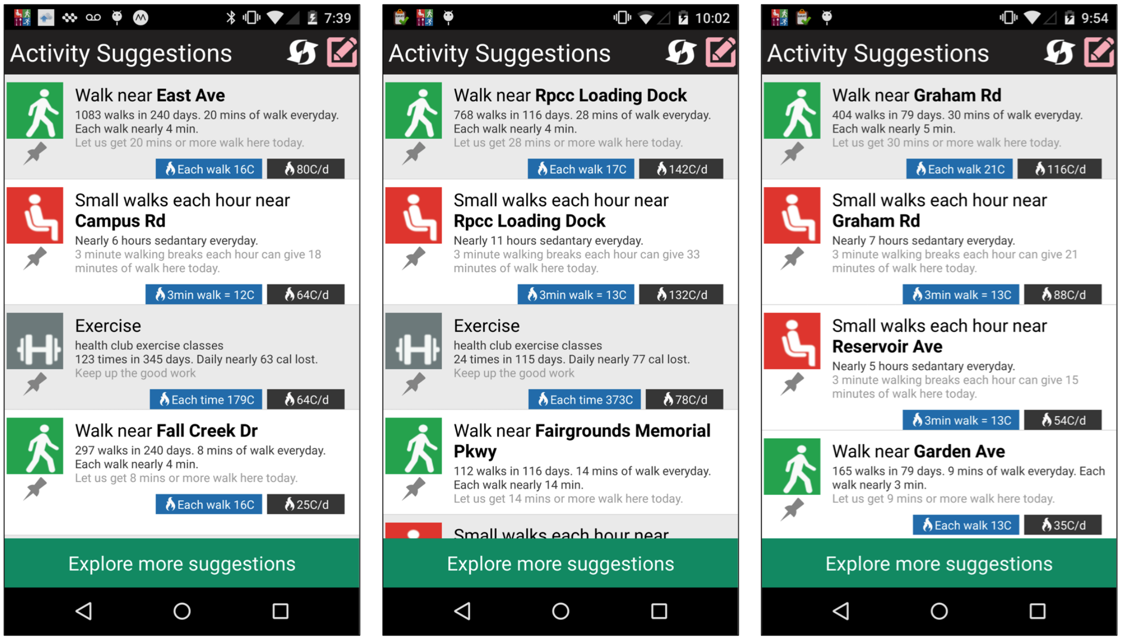
Personalized health recommender systems need to interact with its users. Thus the application needs to ensure smooth user experience. However good usability can be hard for data-driven personalized systems, since the machine learning optimization may not directly translate into information that users can understand. Typically several iterations of user-centered design and development are required to make a highly usable app.

For a novel behavior change application, Klasnja et al. [80] recommend to conduct several small scale pilot studies to polish user experience. Once the usability glitches are addressed in the pilot studies, the application may be evaluated quantitatively in a user study. However a proper large scale randomized controlled trail may be hard because of limited resources. As a result, a smaller scale “single case experiment”[42], with a within subject design, may be more suitable for early quantitative evaluation. Once early efficacy of the app is confirmed quantitatively with single case experiments, a larger scale randomized controlled trail can be run to formally demonstrate efficacy.

1.5 MyBehavior: A case study of automated health recommendation

MyBehavior is a smartphone application that takes a novel approach to generate deeply personalized health recommendations. It combines state-of-the-art behavior tracking with algorithms that are used in recommender systems. MyBehavior automatically learns a user's physical activity and dietary behavior, and strategically suggests changes to those behaviors for a healthier lifestyle. The system uses a sequential decision making algorithm, Multi-armed Bandit, to generate suggestions that maximize calorie loss and are easy for the user to adopt. In addition, the system takes into account a user's preferences to encourage adoption using the pareto-frontier algorithm. MyBehavior is developed as an Android application and the app was used by nearly 40 people in real life situations. Several pilot studies were run to test early efficacy of MyBehavior and to improve user experience. Finally, a 14-week validation study was done that showed statistically significant increase in physical activity and decrease in food intake.

Figure 1.7 gives a few examples of personalized feedback generated by MyBehavior. The three subfigures show distinct personalized physical activity suggestions for three different users. They show the frequency of different user behaviors and places where those activities happened. More frequent and calorie burning activities are ranked higher since they are easy-to-do while also can entail high calorie loss.



(a)

7:39

Activity Suggestions

Walk near **East Ave**
1083 walks in 240 days. 20 mins of walk everyday.
Each walk nearly 4 min.
Let us get 20 mins or more walk here today.
Each walk 16C | 80C/d

Small walks each hour near **Campus Rd**
Nearly 6 hours sedantary everyday.
3 minute walking breaks each hour can give 18 minutes of walk here today.
3min walk = 12C | 64C/d

Exercise
health club exercise classes
123 times in 345 days. Daily nearly 63 cal lost.
Keep up the good work
Each time 179C | 64C/d

Walk near **Fall Creek Dr**
297 walks in 240 days. 8 mins of walk everyday.
Each walk nearly 4 min.
Let us get 8 mins or more walk here today.
Each walk 16C | 25C/d

Explore more suggestions

(b)

10:02

Activity Suggestions

Walk near **Rpcc Loading Dock**
768 walks in 116 days. 28 mins of walk everyday.
Each walk nearly 4 min.
Let us get 28 mins or more walk here today.
Each walk 17C | 142C/d

Small walks each hour near **Rpcc Loading Dock**
Nearly 11 hours sedantary everyday.
3 minute walking breaks each hour can give 33 minutes of walk here today.
3min walk = 13C | 132C/d

Exercise
health club exercise classes
24 times in 115 days. Daily nearly 77 cal lost.
Keep up the good work
Each time 373C | 78C/d

Walk near **Fairgrounds Memorial Pkwy**
112 walks in 116 days. 14 mins of walk everyday.
Each walk nearly 14 min.
Let us get 14 mins or more walk here today.

Small walks each hour near

Explore more suggestions

(c)

9:54

Activity Suggestions

Walk near **Graham Rd**
404 walks in 79 days. 30 mins of walk everyday.
Each walk nearly 5 min.
Let us get 30 mins or more walk here today.
Each walk 21C | 116C/d

Small walks each hour near **Graham Rd**
Nearly 7 hours sedantary everyday.
3 minute walking breaks each hour can give 21 minutes of walk here today.
3min walk = 13C | 88C/d

Small walks each hour near **Reservoir Ave**
Nearly 5 hours sedantary everyday.
3 minute walking breaks each hour can give 15 minutes of walk here today.
3min walk = 13C | 54C/d

Walk near **Garden Ave**
165 walks in 79 days. 9 mins of walk everyday. Each walk nearly 3 min.
Let us get 9 mins or more walk here today.
Each walk 13C | 35C/d

Explore more suggestions

Figure 1.7: Physical activity suggestions for different users

1.6 Organization of the dissertation

This dissertation will mainly describe the design and development of MyBehavior. In the end, several implication and lesson learnt will be described. Specifically rest of this dissertation is organized as follows. Chapter 2 will discuss the psychological theories that underpin the suggestions generation scheme of MyBehavior. Chapter 3 will detail the data logging mechanism and transforming data into user behaviors. Chapter 4 and 5 will contain the process of suggestion generation and related pilot studies. Finally the details of a quantitative study with MyBehavior will be described in Chapter 6. I will conclude with my reflection on designing MyBehavior and potential future work in Chapter 7.

CHAPTER 2

THEORIES TO DESIGN HEALTH FEEDBACK

MyBehavior's uses machine learning on mobile data to personalize health recommendations. The goal of the recommendations is to suggest actions that aid the behavior change process. In order to do that effectively, MyBehavior can borrow from the psychology theories which has already analyzed and researched the behavior change process for decades. For instance, Social Cognitive Theory [17] argues that an individual should acquire self-efficacy or mastery over certain behaviors/actions (e.g., going to the gym) for sustained change. A way to acquire mastery over an action is to repeat the action often [86]. MyBehavior can follow Social Cognitive Theory and prioritize suggestions that encourages repetitions of actions to build mastery.

In fact, designing feedback technologies by grounding the design in psychology theories are quite common. Such grounding ensures that the design takes advantage of the insights of that are known to work for behavior change, and not to reinvent the wheel. In the following, I will describe several theories that are relevant for MyBehavior. Subsequent chapter will operationalize these theories to build the MyBehavior system. Before that, I will first give a brief description of the role of theory in behavior change technologies. Then I will specifically describe theories that are used to guide MyBehavior design. Finally, I will acknowledge other relevant theories that are utilized in prior works, but not used in MyBehavior.

2.1 Role of theory to design health feedback

For behavior change applications like MyBehavior, it is quite common to ground the underlying feedback mechanism into well known behavior change principles. A large number of prior health messaging applications relied on the Trans-theoretical Model of behavior change [122], health belief model [68] or Cognitive behavioral therapy [144] theory etc. For instance,Yun et al [163] used Health Belief Model to design SMS messages to improve asthma symptoms for children. Granholm et a. [61] utilized cognitive behavior therapy theory and created text-message interventions for medication adherence, socialization and auditory hallucinations for Schizophrenia treatment. Grimes et al. [63] created a game to increase nutrition or food related knowledge that follows the Trans-theoretical Model.

Evidence of these examples invokes the question on why theories are important to heath feedback design. Furthermore if theories are indeed important then what is a formal way to incorporate theories into health feedback design. The primary reason to ground health feedback design in existing theories is to incorporate the already-working knowledge of psychological theories; there is no need to spend time on different strategies and subsequently reinventing what has been already known in the psychology theories. While deciding on incorporating theory into design, behavioral psychologist Lawrence Greens mentioned the following [144],

> *"Each theory or model is an approximation of a different slice of reality, or a different perspective of the same slice. The framework should emerge* ***inductively*** *from personal experience in practice (and research). Theories and*

> *their variables can then be attached to the practitioner's (researchers) personal framework* ***deductively*** *by asking of each theory how it fits, what it adds or what it explains among the cause-effect relationships one presumes to be operating in ones practice (or research)."*

According to the above quote, theories can be used to design in two ways: inductively and deductively. First a deductive step can be taken by creating a first version of the application that directly follows a theory. Once the app is deployed, new problem specific observations will arise. These new observations then would be analyzed for patterns/themes. Theories can be used inductively to underpin and solidify the reasons behind the patterns/themes. A new version of the application then can be created and the process will iterate until a usable version is achieved. Such an iterative approach is also analogous in HCI literature as iterative design and in psychology as pragmatists mixed method [151].

In our work, we will first deductively use theory to guide the first version of MyBehavior in Chapter 4. Then we will analyze some observations from the first version's deployment and ground these observations in theories as an inductive step. Subsequently a new version will be developed in Chapter 5. Finally, we will not focus on just one theory. One theory can't explain all the facets of the change process. We will merge together several theories to pragmatically build MyBehavior.

2.2 Behavior change theories for MyBehavior

A successful behavior change may mean the build up of a few healthy dietary and exercise habits. In MyBehavior, the purpose is to aid the build of a few such habits. Now the process building habits can be divided into two steps. First, the user needs to be persuaded to do the actions that constitute the habit. However persuading only is not sufficient, since persuading without a strategy will not result into sustained behavior change. The existing theories in behavior change and habit forming can inform the strategy on how to persuade such that successful behavior change happens. In that vein, I will split the discussion in two parts. First, I will describe the theories of persuasion. Then I will describe how to persuade to build habits.

2.2.1 Momentary change: Theory of persuasion and low-effort

Persuasion is the process of convincing people to undertake desired actions. From ancient philosophers Plato [104] to more contemporary Robert Cialdinis work on influence [35], the topic of persuasion received a lot of attention. In wikipedia, a more up-to-date definition of "persuasion" is given as follows [158]:

> *"persuasion is a process aimed at changing a person's (or a group's) attitude or behavior toward some event, idea, object, or other person(s), by using written or spoken words to convey information, feelings, or reasoning, or a combination thereof."*

The definition defines persuasion as the an interplay among different concepts namely “reason”, “feelings”, “information”, “attitude or intentions” and “behaviors”. All these words themselves are broad research topics by their own. Therefore it is difficult to cover this broad definition, and subsequently operationalize the different concepts into an app. As a result, we will use a simpler model for persuasion by BJ Fogg [52]. Fogg’s model is meta-analysis kind of model that summarizes different elements of prior theories into a simpler form. Furthermore, the model has been widely implemented in computer based solutions.

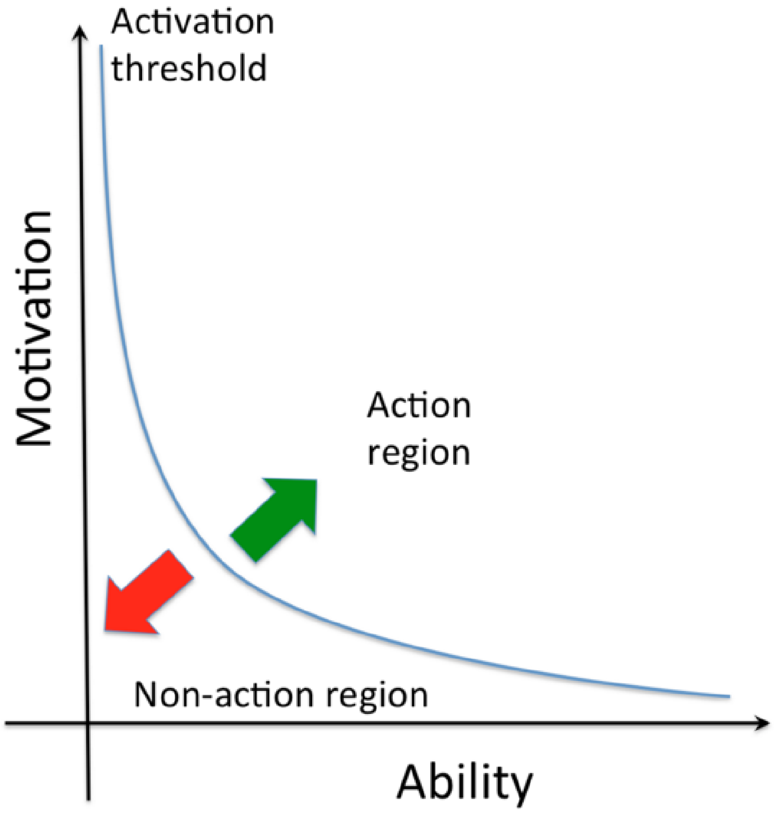


Figure 2.1: BJ Fogg’s Behavior model

BJ Fogg [52] model, as commonly known as Fogg’s behavior model or FBM, advocates the following: in order to persuade anybody to an action, the action is either easy to do (i.e., low-effort) or the user is motivated to take the action. Fogg uses the conceptual diagram in Figure 2.1 to describe his model. As can be seen from the figure, if somebody is highly motivated then s/he might take a high effort action. For instance, before a marriage ceremony or significant life event, people generally become highly motivated to look good in pictures, and often commit to high effort activities, namely starting to go to the gym. However, if an action is easy to do then often people can undertake an action while being less motivated. For example, introverts often find it difficult to go out and socialize with new people. They often go home after office and watch a movie, which requires less effort, even though they are not highly motivated to watch the movies. The "Nudge" literature is another example along the same line [91]. A classic example of nudge is the restaurant case, where healthier foods were stored into front positions and users consumed the food from the front. A possible explanation to this nudging behavior is that users took the easy option of taking the food from the front: they are easy to get irrespective of whether they want to eat healthy or not. Therefore Fogg argues that designers need to take advantage of low-effort more than the motivational approaches, which often requires conscious engagement [53]. However, majority of past literature in mobile health feedback technologies focused on motivating users to be healthy, rather than making things easy.

With the simple framing of persuasion by FBM [52], a large amount of behavior can be explained. Nudging behavior is an excellent example where the FBM can be used to design that promote healthier choices. Furthermore, FBM elegantly shows that the role of motivation in persuasion and its interplay with effort. However, the model is naive in considering motivation and effort as disjoint and mutually independent/orthogonal entities. For instance, knowing that a task is going to be easy might change somebody's motivation of taking the action.

The interplay between low-effort and motivation is better explained by Vroom's expectancy theory. Vroom's theory basically originated from Marketing research [113]. The theory says that the undertaking of a voluntary action adheres to the following steps:

$$\text{Effort} \implies \text{Performance} \implies \text{Reward} \implies \text{Valence}$$

First the individual asks whether she can give the effort and whether the effort will result into the intended performance. For instance, if somebody wants to start giving some effort for running a 5 mile race then she should ask whether she will be able to run (or perform) the 5 mile race. If she can do the 5 mile race then what reward is waiting for her. Finally what does she will feel (i.e., valence: happy or sad) about the behavior and reward. Therefore given this model, effort is related to end reward or motivation - if the action can be done within manageable effort then the user might be more motivated to take the action.

Although FBM and Vroom's expectancy model explore the role of effort or motivation in persuasion, these models are too general and are applicable for a broad range of persuasive applications. These models can explain persuasion or taking actions in the moment. e.g., buying a product or making a click on a website. However health behavior change is a long term process where persuasion needs to happen many times. Furthermore, there are elements of human behaviors that are beyond conscious engagement that persuasion requires. In other words, persuasion often assume conscious engagement towards taking an action (e.g, thinking about reward and personal feelings). However recent habit theory and dual process theory demonstrate that human actions can happen both unconsciously and consciously [46][76]. In fact, human mind can have only one conscious thought at a moment, and majority of the other actions happen habitually or unconsciously. Such habits are especially important for health behavior change, since if a few of these habits are healthy dietary or physical activity habits then people can be healthy with nearly no effort. FBM or Vroom's theory falls short to explain these unconsciousness behaviors.

In the following, I will address these limitations of simple persuasion. I will give a brief description of health behavior change theories that will address the long term engagement with persuasion which will result into forming habits.

2.2.2 Long term change: Theory of self-efficacy and habits

Forming new healthy habits, e.g., eating vegetables, can mean a long term behavior change. However, the question is what characterizes habits. Habits are actions that happen automatically and unconsciously whenever a cue or context is present [46]. For instance, I was going to a new restaurant with a friend few days ago. I started talking and got really engaged in a deep conversation while driving on the street. Then without thinking I took a road that I commonly take to go home, which is not on the road towards our destination restaurant: my hands on my steering wheel automatically turned towards that roads direction. This means that my head was unconsciously responding to context (i.e., road to home) and started taking my habitual actions. While I was taking the wrong turn, my conscious mind was engaged in the deep conversation with my friend.

This presence of unconsciousness in human psyche is well-known and documented in psychology. The great Sigmund Freud is often attributed to introduce the concept unconsciousness in psychology [62]. Nobel prize winner behavioral economist Daniel Kahneman's famous book thinking fast and slow [76], started with the “two systems theory” where he argues that human brain can support two kinds of processing (1) fast unconscious processing: this system works parallelly with other actions. (2) slow conscious system when we have to think and consciously make decisions; The slow system support serial processing only, and it is not possible to consciously think about two things at the same time. Therefore in an ideal scenario, if an individuals mind is clouded (e.g., highly stressful situations) with other thoughts then it is likely that the individual’s other actions are likely to be unconscious or habitual. From a health behavior change perspective, this means that if a stressful situation occurs then it is unlikely that somebody will continue to consciously/mindfully motivate themselves to continue the activities. Given our modern information age, it is more and more likely that our minds will increasingly be clouded by more information [98]. Therefore often modern human behavior researchers argue for more automatic and unconscious means of influence or change. For instance, Robert Cialdini, in his best selling book of social influence, mentioned the following regarding unconscious behavior [35]:

> *“The evidence suggests that the ever-accelerating pace and informational crush of modern life will make this particular [unconscious] form of unthinking compliance more and more prevalent in future.”*

Similar argument has been put up by Gardner [57]. Gardner contends that habits are in fact more sustainable than motivation based techniques, and designers should focus more on habit building for promoting healthier lifestyle. He argues that habits require less conscious engagement and likely to happen even when the mind is consciously engaged to something else. Therefore, the question is whether we can create some unconscious behavior (i.e., habits), given the limitation of slow and serial conscious processing of the human mind.

Literature suggests that context is an essential component of habits and habits only occur automatically in a given context [86][56]. For instance, if somebody has a walking habit in the office then the walking will start automatically while in the office. But if the office is switched to a new location then early habits will be gone and new habits will need to be formed in the office location. Another example can be driving a car. If we are inside our car, that we are driving for days, our actions of pressing the pedal or steering the wheel become automatic (i.e., we dont have to think about it anymore). However, if we buy a new car then automatic behaviors in our older car will not transfer and we have to consciously engage to drive the new car for a few days. Gardner [56] described similar connection between context and habit with the following anecdote:

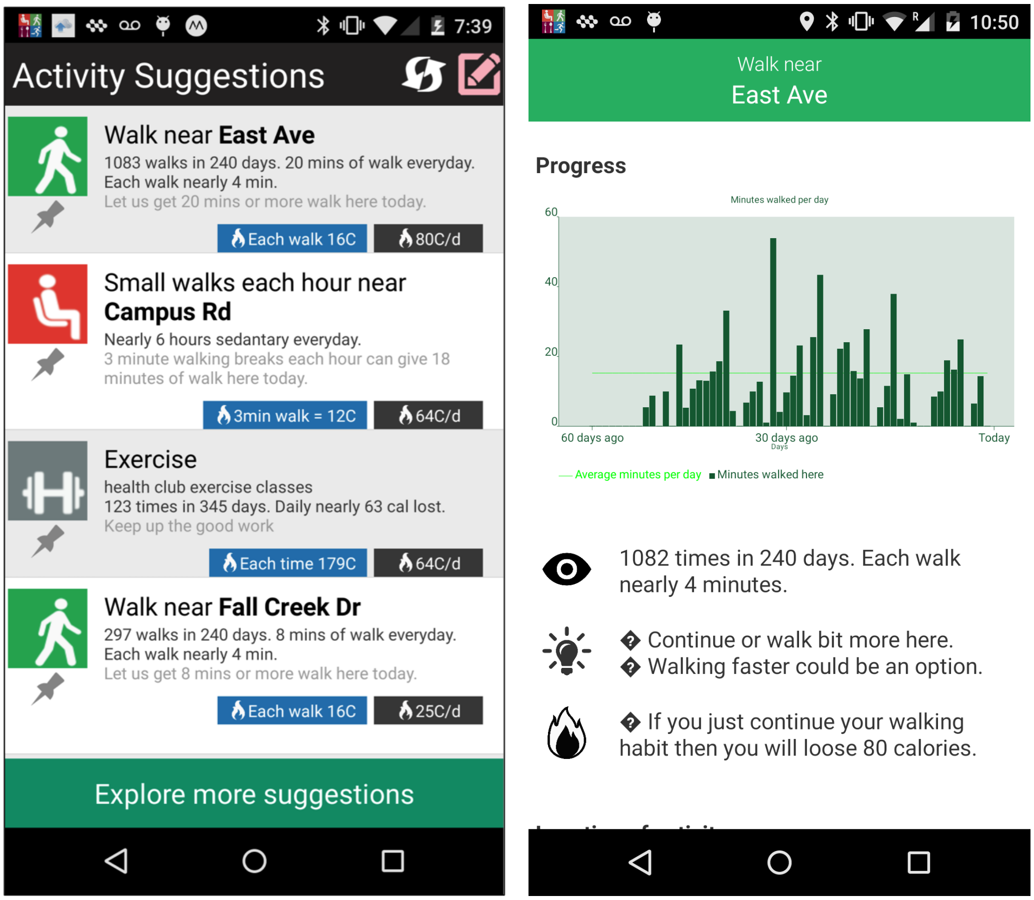
> *"There is a story of a practical joker, who, seeing a discharged veteran carrying home his dinner, suddenly called out, 'Attention!' whereupon the man instantly brought his hands down, and lost his mutton and potatoes in the gutter. (Huxley, 1866, cited in James, 1890, p120)"*

However, context only is not enough to form habits. The action needs to be repeated often to form habits. For instance, if an individual wants to be habituated of driving a new car then just being in the car would not magically create the habitual driving; the individual have to drive the car couple of times. More formally, Lally et al. [86] shows that if same behavior is repeated often then users acquire a sense of automaticity, and a higher automaticity is the predictor of habit formation.

Finally, it is not always easy to repeat a behavior to a point where the behavior starts happening unconsciously and habitually. For instance, I am going the gym regularly for the past 2 year. However, my gym works is yet to turn into a unconscious habit. According to Lally et al. [86], habit forming may take time and it depends on the task. Also, there are individual variabilities on how long it takes to form different habits.

Given that unconscious habits might take time to form, repetitions may still be useful. For instance, a number of theories, namely social cognitive theory [17], health belief model [68], and theory of planned behavior [13], argue the following. Social cognitive theory [17], from Albert Bandura, argues that an individual has to acquire self-efficacy to change her behavior. A big component of self-efficacy is the mastery feeling. Mastery of performing an action often comes through skills which are often acquired through repetitions. Thus if an individual keeps on repeating to go to the gym then the individual is acquiring the skills or mastery, even though habits are not immediately formed. This mastery of skill or a sense of control of executing an action is also well founded in other behavior change theories. For instance, theory of planned behavior [13] states that, the undertaking of a voluntary action depends on a sense of control of executing the action. This is referred to as "perceived behavior control belief".

In summary, if the same action is done repeatedly in a given context (e.g., place) then habits might form. If habits are not formed then at least an individual gains self-efficacy through repetitions, which is an essential component of behavior change. Given these insights, MyBehavior will strategically suggest repeated actions in a given place (or context). Furthermore to persuade users to take the actions, MyBehavior will ensure the actions are easy to do as suggested by FBM. In the following, I will give a brief preview on how these theories are incorporated in MyBehavior's recommendations. Later chapters will include details on how these recommendations are created.



Walk near **East Ave**
1083 walks in 240 days. 20 mins of walk everyday. Each walk nearly 4 min.
Let us get 20 mins or more walk here today.
Each walk 16C | 80C/d

Small walks each hour near **Campus Rd**
Nearly 6 hours sedantary everyday.
3 minute walking breaks each hour can give 18 minutes of walk here today.
3min walk = 12C | 64C/d

Exercise
health club exercise classes
123 times in 345 days. Daily nearly 63 cal lost.
Keep up the good work
Each time 179C | 64C/d

Walk near **Fall Creek Dr**
297 walks in 240 days. 8 mins of walk everyday. Each walk nearly 4 min.
Let us get 8 mins or more walk here today.
Each walk 16C | 25C/d

Walk near
East Ave

Progress

1082 times in 240 days. Each walk nearly 4 minutes.

� Continue or walk bit more here.
� Walking faster could be an option.

� If you just continue your walking habit then you will loose 80 calories.

(a) **(b)**

Figure 2.2: Screenshot of MyBehavior design (a) activity suggestions (b) tracking progress of suggestions over time

2.3 A preview of theories inside MyBehavior design

MyBehavior incorporates the ideas of low-effort and habit forming in its interface design. Figure 2.2 shows a few representative screenshots of MyBehavior, where the concepts of low-effort and habits are incorporated. Figure 2.2a shows a set of highest rates suggestions where two elements of low effort are included. First, the suggestions ask to continue or make small changes to already existing behaviors (e.g., continue walking or gym exercise). Since the suggestions already fits a users routine, they are easy to follow. Secondly, more frequently repeated behaviors are prioritized in the suggestion ranking; the idea is that the more a suggestion is done, the easier it gets. Finally, Figure 2.2b shows the progress of a suggestion over time. The progress prompts the user to repeat the suggestions often, in a given place or context (e.g., East Ave in the figure). Recall, repeating actions in a context helps to build habits.

2.4 Other theories of persuasion and behavior change

I conclude this chapter by describing a few well-known theories of persuasion and behavior change that are not used in MyBehavior. Note that, these theories are not included because it is hard to operationalize them in the initial version of MyBehavior, which is the first app to bridge the gap between health feedback and mobile data with actionable recommendations. In later iterations of MyBehavior, these theories will be operationalized to make MyBehavior more sophisticated.

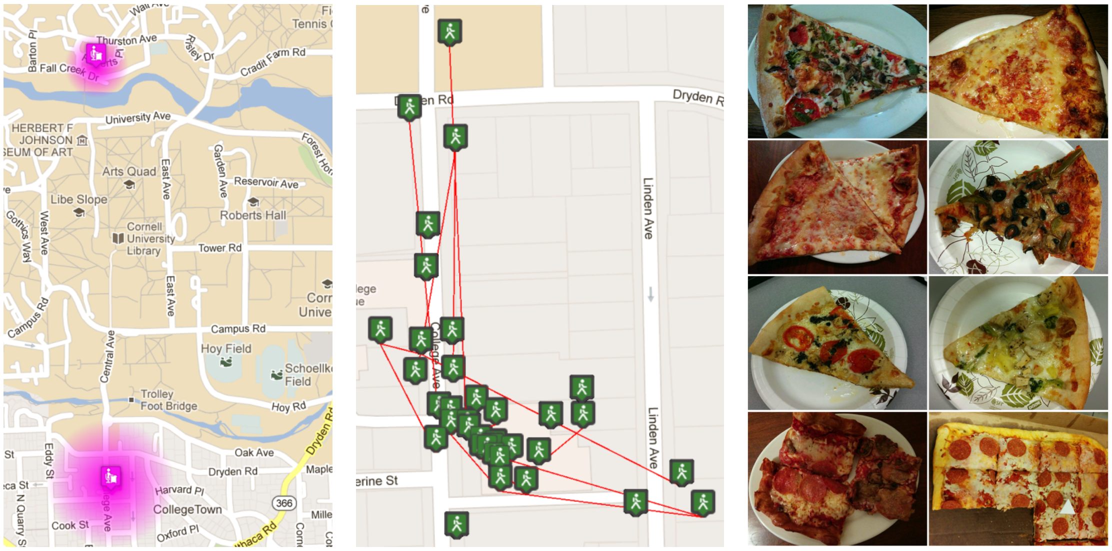
The first major missing theoretical aspect of MyBehavior is the ignoring of motivational side of persuasion, and prioritizing low-effort. Note that, low-effort activities are prioritized since they are actionable in both low and hight motivation states. However, extra motivation makes any action/recommendation more actionable. And this extra motivation can be created in different ways, and the earlier literature on health behavior change explored them extensively. For instance, gamification compares a user with other similar users, and can increase the motivation to exercise [99]. Social support can also increase motivation, and they pervasive in modern fitness apps. However, there is little evidence of the efficacy of social support in fitness apps [161]. Goal setting (e.g., step counting) is another way to increase motivation. Ubifit [38] and BeWell [87] both use the live wall-paper in the phone to increase user motivation to reach physical activity, socialization or sleep goals. Nonetheless these works in motivation are orthogonal to low-effort, as suggested by Fogg's behavior model [52]: low-effort actions can be more actionable with extra motivation. e.g., if a user goes to the gym regularly then going with friends can increase motivation, which in turn makes going to the gym more actionable.

In addition to motivational aspect of momentary short term changes, a well-used theory to model the long-term aspect of behavior change is the Trans-theoretical model, which is also ignored in MyBehavior [122]. The trans-theoritcal model, or TTM in short, is a stage based model. TTM argues that an individual moves through several stages while making a behavior change. The first stage is the *pre-contemplation* stage where the individual is not intending or taking actions to change behaviors. The second stage is the *ready* stage where the individual is intending to change behaviors but not taking actions. The third stage is the *action* stage where the individual is taking actions, but needs to strengthen the commitment and form habits. In the fourth stage, referred to as the *maintenance* stage, the individual is taking actions, and has already formed and maintaining habits. The principal idea of TTM based intervention is to tailor health messages or treatments based on TTM stages [84]. Such TTM based tailoring has found success over non-tailored counterparts over a number of areas; namely weight loss [72], stress management [50], smoking cessation [154] and depression management [92]. Despite these successes, there are several limitations of TTM. Firstly, a number of studies found no effect of TTM over non-stage based model [159]. Secondly, the discrete categorization of TTM is not well-defined [94][156][110]. Furthermore, it is not clear how the division would work where day-to-day data are available; e.g., if we have step-count data per day then what does it mean to be in ready or maintenance stage based on the every day step-count data (e.g., "Does walking 10K steps for a week mean an individual is in maintenance stage?"). It is for this potential ambiguity in how to operationalize TTM stages with data, we avoided to implement TTM in the first version of MyBehavior.

CHAPTER 3

FROM MOBILE DATA TO BEHAVIOR

In this chapter, the first step to personalize health recommendation, i.e., to log data and find patterns of user behaviors, is discussed. First, I describe the food and physical activity logging techniques, along with the optimization to ensure long term logging, e.g., reducing phone battery usage and low-burden food journaling with pictures. Then these logs are analyzed to find patterns of dietary intake and physical activity, e.g., walking behaviors near office or pizza eating behavior. In subsequent chapters, personalized suggestions will be created that relate to these behaviors.



(a) (b) (c)

Figure 3.1: Visualization of user behaviors over a week (a) Heatmap of places a user stayed stationary (b) Location traces of frequent walks for the same user (c) Repeated pizza eating behavior of another user.

3.1 Necessity of behavior mining

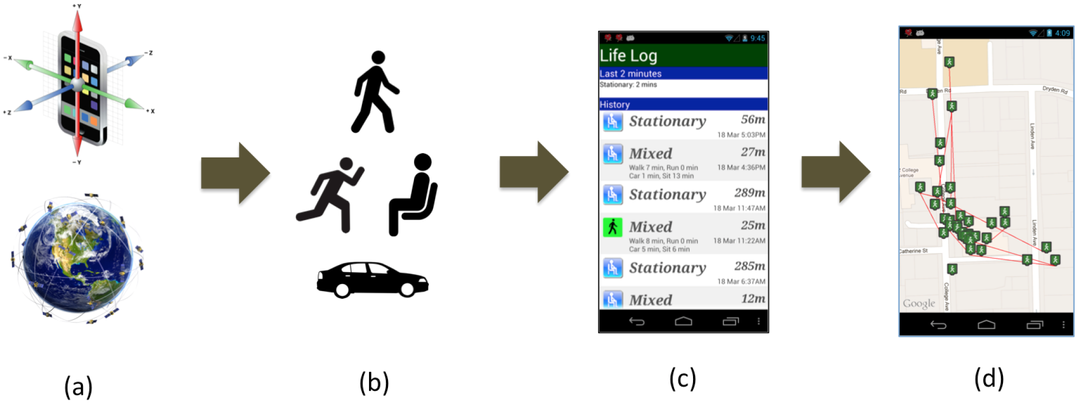
This chapter concerns with the mining of repetitive behavioral patterns. But a general question is why finding repetitive behavioral patterns are even important. In this paragraph, I provide a few anecdotal evidence of behavioral patterns that are mined from phone data of actual users. Then I argue how these behavioral patterns can be used for easy-to-do behavior change steps. Figure 3.1 shows three different examples of behaviors. Figure 3.1a shows the stationary behaviors of a user, which shows the home and office locations where the user stays sitting. Figure 3.1b is the walking locations of the same user near the office. Figure 3.1c shows repeated pizza eating behavior of another user. It can be readily seen that these behaviors point to cases which can be continued or changed towards a healthier lifestyle. Finally, since the behaviors are already part of a user's lifestyle, they are easy to change. And the good news is that these behaviors can be mined automatically from phone data. In the following, I detail how such phone data can be collected and how the behavior mining can be carried out subsequently to group similar phone data.

3.2 Mining physical activity behaviors

Physical activity behaviors are mined in a three stage process (Figure 3.2). First accelerometer and GPS data are logged to understand different activities, namely walking, stationary, running and in vehicle (Figure 3.2a-b). Then activities are grouped together in a chronological list that we refer to as Lifelog (Figure 3.2b-c). In the final step, the lifelog is analyzed to find patterns of physical activity behaviors (Figure 3.2c-d). These three stages are described below

3.2.1 Activity recognition

MyBehavior does physical activity recognition as a background process using accelerometer and location sensors. Accelerometer sensors are sampled at



Life Log

Last 2 minutes

Stationary: 2 mins

History

Stationary 56m 18 Mar 5:03PM

Mixed 27m Walk 7 min, Run 0 min Car 1 min, Sit 13 min 18 Mar 4:36PM

Stationary 289m 18 Mar 11:47AM

Mixed 25m Walk 8 min, Run 0 min Car 5 min, Sit 6 min 18 Mar 11:22AM

Stationary 285m 18 Mar 6:37AM

Mixed 12m

Figure 3.2: Stages of mining physical activity behaviors

the maximum available rate. Time domain features (such as mean, variance and zero crossing rate) and frequency domain features (such as energy distribution at different frequencies) are extracted directly on the phone. We use a Gaussian mixture model to classify the data into different activities: walking, running, stationary, driving, etc. The sensing and inference module of MyBehavior builds on the data normalization and inference techniques developed and tested in [96]. In order to re-validate the performance of the system, we collected accelerometer data from 20 participants. The accuracy of the activity inference system is shown in Figure 3.3, and they are comparable to the reported performance in [96].

MyBehavior employs a simple duty-cycling scheme to save phone battery. Sensing of the accelerometer is triggered every 20 seconds using the Android alarms functionality. Android alarms are efficient since they can be triggered without keeping the CPU awake. After every trigger, the CPU is turned on for 10 seconds to sense the accelerometer and perform inference.

All stationary, walking and running activities are location tagged. Since location sensors consume more power, locations are polled only when significant

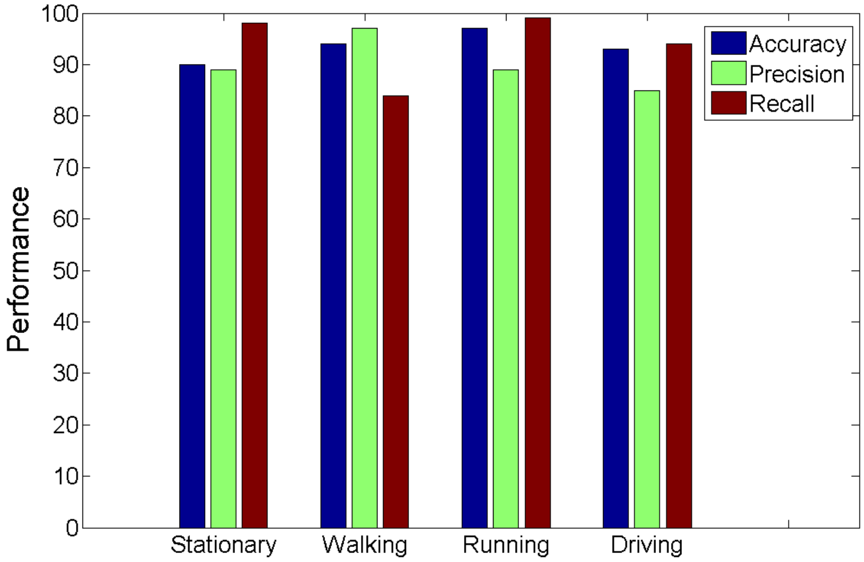
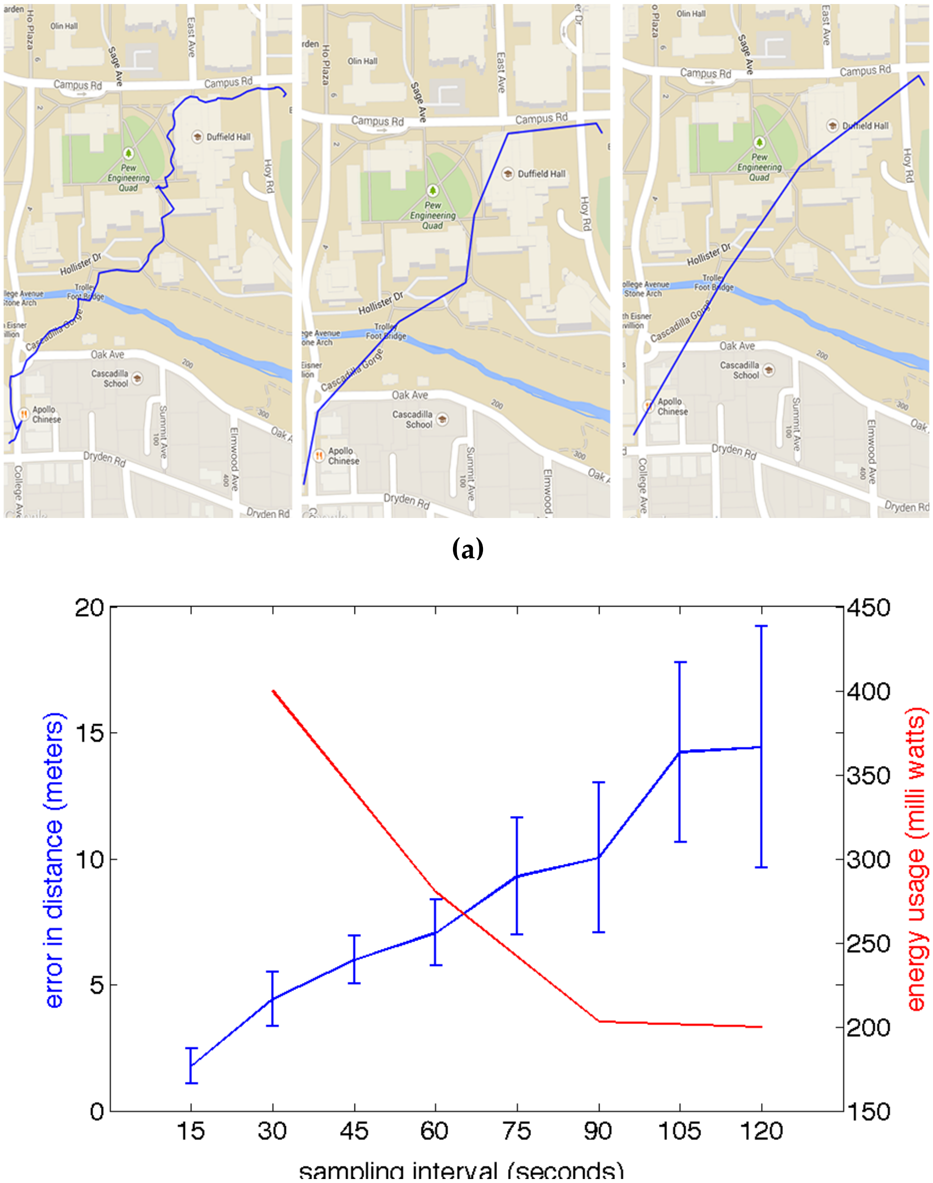


Figure 3.3: Performance of different activity classifiers

movement are recognized within a certain time window (a minute). As long as the movement is persistent, locations are polled periodically. We sample gps once a minute, and the decision for "one minute" sampling is done as follows. First, as noted, continuous sampling of location data (e.g, GPS) would quickly drain the battery of the smartphone, which would result in low usability. On the other hand, if we sample less frequently, then we have a poor estimate of the paths. Figure 3.4 illustrates this point more. Figure 3.4 shows 3 paths generated from three different sampling rates for the GPS. The left most path is the original path taken with dense GPS sampling at **10 Hz**. The paths in middle and right are subsampled at 1 and 2 minutes respectively. As can be observed from the figure, the one-minute sampling tracks the original path fairly closely. Figure 3.4 shows this fact more quantitatively. We plot the average distance of the original path and subsampled path for different rates of subsampling. A graph with GPS power loss with different subsampling strategy is also shown. We can see that 1 minute sampling of GPS can achieve a reasonable battery life



(a)

(b)

Figure 3.4: (a) Left is the original path. Paths in middle and right figures are sampled at 1 and 2 minutes (b) Average distance and GPS energy usage for different subsampling.

loss with an average of 8 meters distance loss due to subsampling.

With this simple duty-cycling strategy, the official Android Galaxy Nexus phone with a 1850mAh battery lasts 32.1 hours, which is a 50% improvement over a non–duty-cycling approach (21.4 hours). Note that finding the optimal power-saving scheme is not the focus of our research. Our goal was to implement a scheme that would allow users to run our application, while using their phone regularly without requiring a mid-day battery recharge.

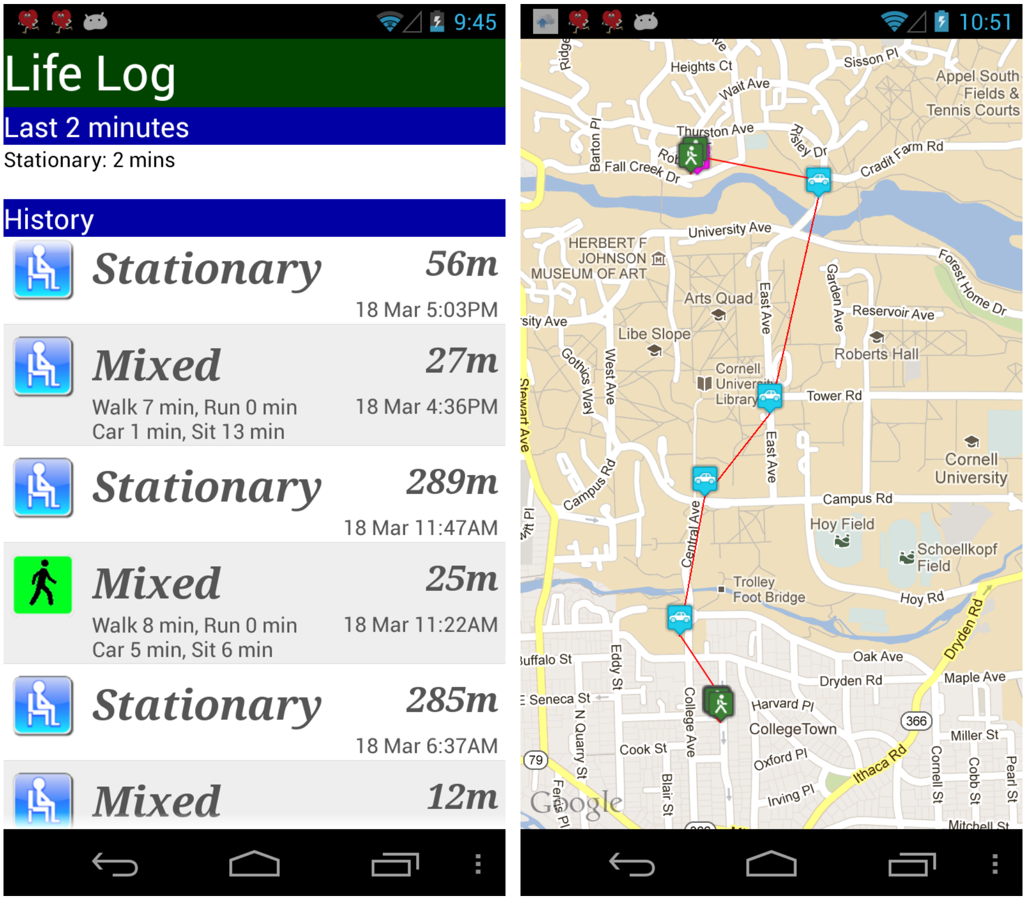
Finally, we expect that users would not be able to carry the phone at all times and certain activities would not be inferred accurately by our classifier [38][118]. To handle these types of omissions and errors, we gave users the option to manually input physical activity from a list of 800 different activities [12] and record the time and duration of the activity. Once the activity type and its duration are known, the calorie loss (kilo-calorie or *kcal*) of the activity is automatically computed using equation 3.1. MET or Metabolic equivalent [5] is a constant for specific exercises (the MET value of walking is 2 and running is 8).

$$kcal \equiv MET \ * body\ weight\ (kg) * hours\ of\ activity\ (meter) \tag{3.1}$$

3.2.2 Lifelog: A journal of daily physical activity

MyBehavior summarizes the activity entries in a *life-log* chronologically. The activity entries are generated automatically and require filtering in order to create concise and meaningful activity entries. The filtering is done in two stages: (i) a fixed-window mode filter is used to replace the instantaneous activity predictions within a one-minute window using the mode of the predicted labels, which smoothes out spurious noisy predictions; and (ii) contiguous activities with the same label are grouped into one episode to create lifelog entries. For example, if a user is stationary for 50 minutes, the system will generate 50 one-minute *stationary* labels and then aggregate them together into one 50-minute *stationary* entry.

MyBehavior does some additional processing than simple aggregation of



9:45

Life Log

Last 2 minutes

Stationary: 2 mins

History

Stationary *56m*
18 Mar 5:03PM

Mixed *27m*
Walk 7 min, Run 0 min
Car 1 min, Sit 13 min
18 Mar 4:36PM

Stationary *289m*
18 Mar 11:47AM

Mixed *25m*
Walk 8 min, Run 0 min
Car 5 min, Sit 6 min
18 Mar 11:22AM

Stationary *285m*
18 Mar 6:37AM

Mixed *12m*

(a) **(b)**

Figure 3.5: (a)-Screen shot of Lifelog (b) Visualization of a lifelogged event.

similar contagious activities. For some common behaviors like driving a car or taking a bus results in many short activity clusters in the log (e.g., walking to the bus, sitting stationary at each bus stop, and driving between stops). For these cases, MyBehavior generates a “mixed” activity entry in the lifelog. In addition to the automatically-sensed activities, manually-entered activities are shown in the lifelog as distinct entries. Furthermore, lifelog summarization happens realtime within the phone. Figure 3.5 shows an instance of a lifelog and map visualization of a entry in the lifelog.

3.2.3 Mining physical activity behaviors

MyBehavior use the lifelog to group similar physical activities together. For instance, all walking near an office would be grouped together. Such groups represent specific physical activity behavior. Manually tracked activities are first clustered based on the type of activities. For example, all types of yoga or gym exercises are grouped together. Automatically tracked activities with location tags are clustered by places they occur. Clusters are found using unsupervised machine-learning techniques to identify similarity. For Stationary activities, we first compute a distance between two stationary point s_1, s_2 using the following havershine equation.

$$\boldsymbol{d}(s_1, s_2) = 2rsin^{-1}\left(\sqrt{sin^2\left(\frac{\phi_2 - \phi_1}{2}\right) + cos(\phi_1)cos(\phi_2)sin^2\left(\frac{\lambda_2 - \lambda_1}{2}\right)}\right) \tag{3.2}$$

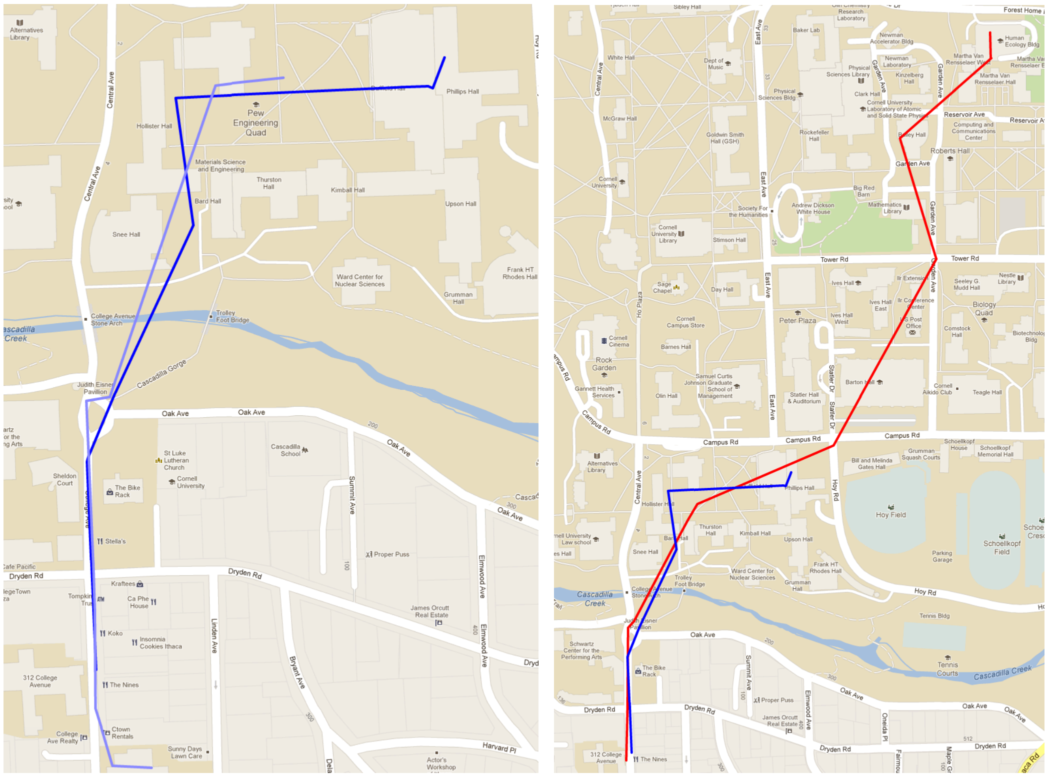
where $s_1 \equiv (\phi_1, \lambda_1)$, $s_2 \equiv (\phi_2, \lambda_2)$ are location points, and ϕ's are latitudes and λ's are longitude values in radian units. If the distance $d(s_1, s_2)$ fall within 150 meters of each other then s_1, s_2 are put in the same cluster. We choose 150 meters as threshold, since stationary events often happen in indoors, and the indoor localization are often accurate up to 150 meters []. As an example for stationary clustering, a users stationary activities in the office are typically in close proximity to each other (i.e., may be within 150 meters). Therefore, MyBehavior would cluster these office locations into a single cluster that would represent the users stationary behavior in the office.

Walking and running activities are more difficult to cluster because MyBehavior needs to determine whether two activity trajectories are similar and whether they happen in a similar location. To group similar walking or running events, MyBehavior uses an algorithm derived from the literature on handwriting recognition called Fréchet distance [15]. In handwriting recognition, the task is to find a canonical letter that matches the shape or trajectory of a handwritten letter. The analogous task in MyBehavior is to find whether a new walking trajectory (e.g., office to coffee shop) matches previous walking trajectories.

A common way to explain Fréchet distance is using the case of a dog owner taking her dog for a walk. Although the owner and the dog take the same path, each can choose their own trajectory. Given the trajectory of the owner and the dog, Fréchet distance computes the minimum length of the leash required to support these trajectories. Thus, if the two trajectories are very different, then the Fréchet distance would be comparatively high. In our case, if the user walks or runs multiple times in a similar path for a similar amount of time, then the Fréchet distance would be low. However, traditional Fréchet distance computation is expensive [15]. Instead, we use a discretized version of Fréchet distance [47]. More specifically this algorithm finds the distance in the following way. If $M_1 \equiv (M_1^1, M_1^2, M_1^3, ..., M_1^m)$ and $M_2 \equiv (M_2^1, M_2^2, M_2^3, ..., M_2^n)$ represent two walking (or running) trajectories where each $M_i^j \equiv (\phi_i^j, \lambda_i^j)$ is a sampled location. Fréchet distance between M_1 and M_2 is computed using the following equation 3.3:

$$\boldsymbol{fd}(i,j) = \begin{cases} \boldsymbol{d}(M_1^i, M_2^j)\ ; \quad \text{when } i = 1 \text{ and } j = 1 \\ \max(\boldsymbol{fd}(i, j-1),\ \boldsymbol{d}(M_1^i, M_2^j))\ ; \quad \text{when } i = 1 \text{ and } j > 1 \\ \max(\boldsymbol{fd}(i-1, j),\ \boldsymbol{d}(M_1^i, M_2^j))\ ; \quad \text{when } i > 1 \text{ and } j = 1 \\ \max(\boldsymbol{fd}(i-1, j),\ \boldsymbol{fd}(i-1, j-1),\ \boldsymbol{fd}(i, j-1), \boldsymbol{d}(M_1^i, M_2^j))\ ; \text{ when } i, j > 1 \end{cases} \tag{3.3}$$

Thus Fréchet distance at (i, j) is computed incrementally as the highest Fréchet distance in the local neighborhood (i.e., $(i-1, j-1), (i-1, j), (i, j-1)$) and the

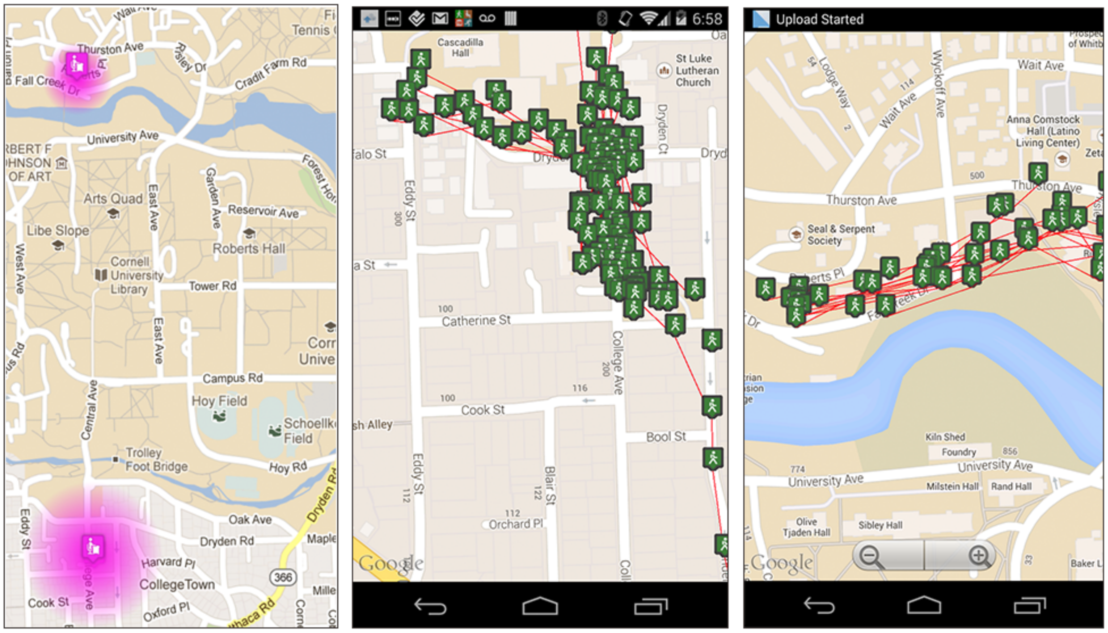


(a) **(b)**

Figure 3.6: (a) Two paths assigned to the same cluster by the Fréchet distance clustering; (b) Two paths not assigned to the same cluster by the Fréchet distance clustering.

distance between M_1^i, M_2^j. At the start, $\boldsymbol{fd}(m, n)$ is the Fréchet distance between M_1, M_2. With this definition, if two trajectories are very different then the Fréchet distance would be high in comparison to similar trajectories. Equation 3.3 is efficient with a runtime of $O(nm)$. Figure 3.6 shows two cases where two paths in the left figure are found similar by Fréchet distance measures whereas the paths in the right side figure are found dissimilar. Finally, MyBehavior uses a threshold based clustering on the Fréchet distance, similar to grouping stationary behaviors, to group walking and running behaviors.

Figure 3.7 shows some clusters generated by this technique. The image on the left represents a users stationary episode in the office and home, whereas the middle and right-hand images show two walking clusters generated by two different users. The middle image represents a users walks near the office, while the cluster in the right-hand image represents another users daily walks from home to a bus stand.



(a) (b) (c)

Figure 3.7: Clusters generated from user activities: (a) locations where user A stayed stationary, (b) location traces for user B where he walked around his office, and (c) walking traces of user A from his house to the bus stop.

3.2.4 Pros and cons of the mining techniques used for physical activity behaviors

Other than the grouping of manually logged activities, the clustering of stationary, walking or running are performed based on locations. We used simple threshold based clustering to group similar activities. However, there are other methods to group significant locations of an individual's life from location traces. Ashbrook and Starner [16] used k-means clustering in conjunction with markov models to predict a user's most significant locations. DBSCAN, on the other hand, used a probability density based estimate to group important locations from user location traces [166][69]. A major limitation of these works is that they only apply to stationary locations, and not applicable for clustering walking or running trajectories. Furthermore, k-means or density based clustering applies clustering on the whole data to group similar data points. However, such clustering may be computationally expensive in comparison to threshold based clustering [165]. Nonetheless, it can be argued that a computationally intensive approach may yield better accuracy. But this also requires more battery. In-the-cloud and out-of-phone computation of the clustering can be an option, but the locations need to exported to a server which may undermine privacy. The final reason for not using k-means or density based approach is that infrequent locations are not considered as significant behaviors. However, a non-significant behavior may be important for later and should not be thrown away. As we will see later in the next chapter, the suggestion engine will deliberately use infrequent behaviors to diversify a user's existing healthy habits. e.g., an infrequent long walk from office to home can be suggested to a user, and the user may do the activity after seeing the suggestion and make it a habit in future. All these reasoned combined, we decided to not to use k-means or DBSCAN in MyBehavior.

Regarding the clustering of trajectories, there are some existing research in past literature. Krumm [83] used hidden markov model and kalman filters to cluster similar location traces. However, a major problem with such latent variable approach is the associated computational complexity; i.e., computing hidden values require expectation-maximization which is an iterative technique and is computation intensive [26]. In comparison, Fréchet distance is less computationally intensive. In fact, Fréchet distance has been used before to cluster similar locations. Shah et al. [142] used Fréchet distance to compute similar trajectories, but the purpose was to detect identical driving behavior.

Finally, only the location based clustering is not enough to fully characterize user behaviors. There are other aspects to human behaviors. For instance, office locations are not accessible during weekends, and the behaviors of staying sitting in the office do not apply during the weekends. As a result, MyBehavior reliance on location still remains a limitation to the behavior mining approach. Despite the limitation, location is an important contextual variable. Also, location is well-used for advertising and other context based recommendations frequently [79][55][10]. Therefore, considering location is a reasonable practical choice to consider in the first version of MyBehavior. Later versions can utilize other contextual variables to further pin-point and contextualize health recommendations.

3.3 Dietary behavior

A user's dietary behavior can be understood by analyzing and finding similar patterns of eating in the food log. The process is shown pictorially in Figure 3.8 and works in the following steps. First, a user logs a food by taking a picture of the food (Figure 3.8a). Then the food is analyzed by a crowd-source workers (Figure 3.8b). We have developed a hybrid technique that uses a combination of machine learning and crowd-sourcing to label foods at fraction of a cost compared to earlier literature [111]. Then these foods are grouped based on the similarity of the labels (Figure 3.8c). In the following, I describe each of these steps in more detail.

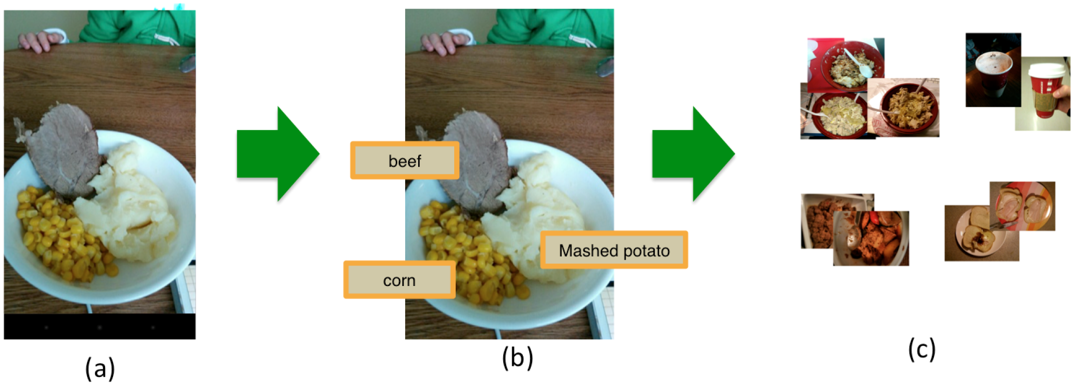


Figure 3.8: Stages of mining dietary behaviors

3.3.1 Crowd-sourcing dietary food logging

MyBehavior uses a picture based logging approach to record dietary intakes. Once the picture is taken, a common way to annotate the food is to manually search and enter every food ingredient with their portion size to determine the calorie. In the first version of MyBehavior, we used a manual logging approach to log food where users select food ingredients from a The United States Department of Agriculture (USDA) [67] database containing more than 8000 types of food. However, we realized that manual logging approach is too cumbersome and high-effort to continue for extended period of time. Earlier works have reported similar findings for manual logging [40][39].

However, it is hard to understand users' dietary behavior and personalize suggestions without food logs. Therefore we had to find a way to increase adherence to food logging. One way to reduce the burden or high effort of logging foods is by crowd-sourcing food ingredient labeling and calorie intake. In the crowd-sourcing approach, the user takes a photo of the food and the photo is labeled by crowd-workers in the web with food ingredient and calorie information. Noronha et al. [111] in their PlateMate work already validated such an approach where Amazon Mechanical Turk [14] (AMT) workers are used to ascertain nutritional information. In the following, we describe an end-to-end mobile phone implementation of this crowd-sourcing approach. In addition, we describe a method that can reduce 50% cost of crowd food labeling compared to Platemate [111]. Finally, we discuss the efficacy of the crowdsourcing system which would be followed by the clustering process of grouping similar foods.

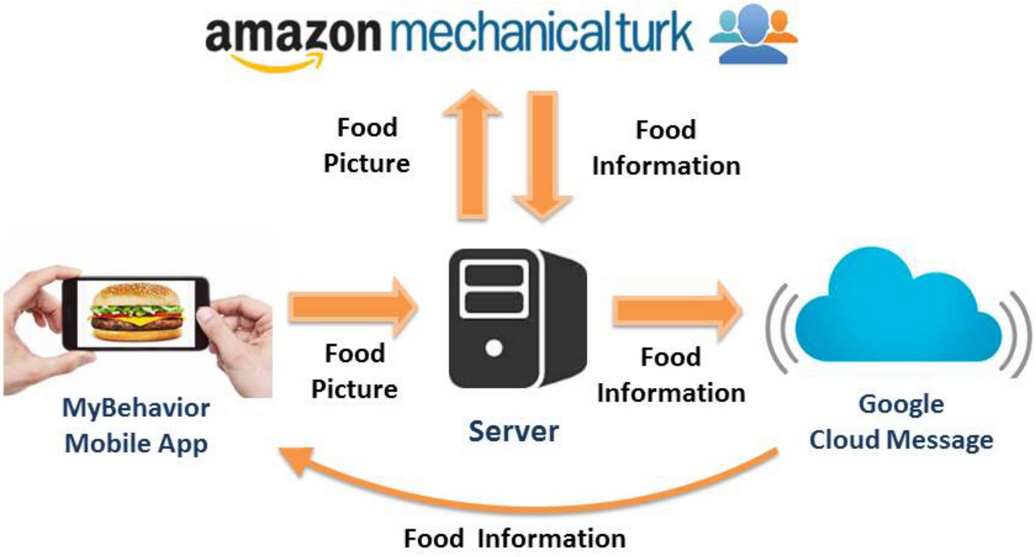
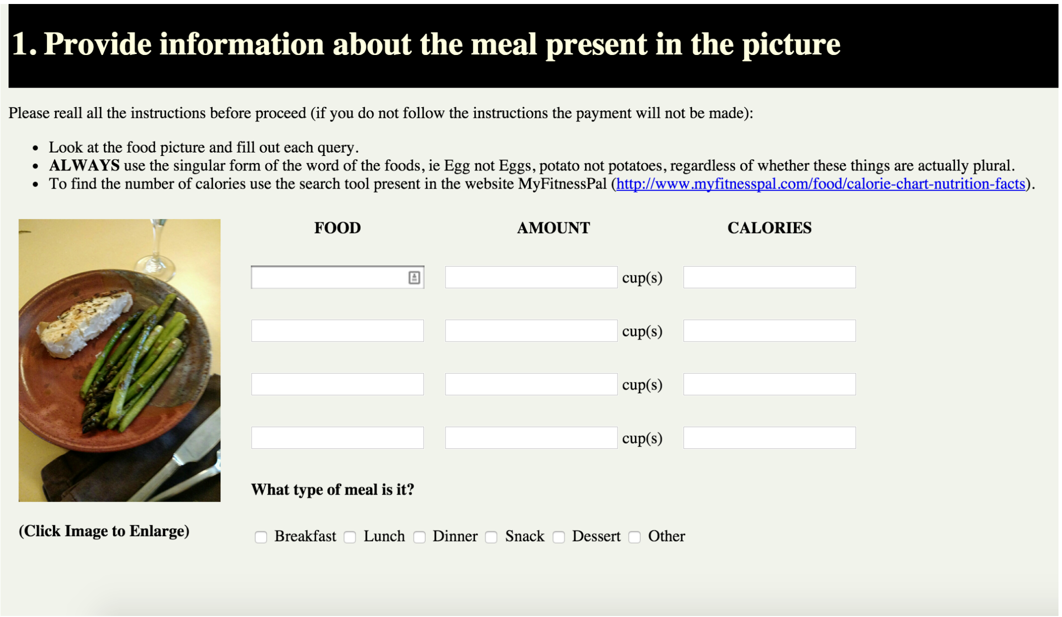


Figure 3.9: Food and calorie intake logging based on crowd sourcing using Amazon Mechanical Turk

A typical work flow of food logging works as follows: first a user takes a picture of the food using the application (Figure 3.11a). The picture is then sent to a server and a Human Intelligence Task (HIT) is created at Amazon Mechanical Turk (AMT), a popular crowd sourcing service. Then a set of crowd workers or ***turkers*** in AMT look at the picture, and provide the following information: food ingredient in the picture and an estimated calorie. Figure 3.10 shows a screenshot of the labeling interface that a turker uses to label a food. In order to make the task easier, we implement an auto-complete feature - when the worker starts typing the food name, some suggestions of food appear in a window below the field. If the user selects a food, the corresponding calorie information per portion size is automatically added to the form. Then the turker only needs to update the food portion sizes and the number of calories is updated accordingly. The food suggestions in auto-complete features contain 50 popular foods from MyFitnessPal [106], a popular fitness application for phones. We decided to use this approach because larger databases contain many occurrences for the same food name which is confusing for the worker. Once the crowd-worker fills out all the required information, s/he earns 4 cents for completing the task, and the food information is sent and stored in the server. The server then sends the food labels and the calorie information to the Google Cloud Messaging (GCM). GCM then sends the food information to the Android device from which the food picture was taken. Figure 3.9 shows the basic architecture of this application. Figure 3.11 shows the application in action. As shown in the figure, the app also gives users control to remove wrong labels by unchecking them.



1. Provide information about the meal present in the picture

Please reall all the instructions before proceed (if you do not follow the instructions the payment will not be made):

- Look at the food picture and fill out each query.
- **ALWAYS** use the singular form of the word of the foods, ie Egg not Eggs, potato not potatoes, regardless of whether these things are actually plural.
- To find the number of calories use the search tool present in the website MyFitnessPal (http://www.myfitnesspal.com/food/calorie-chart-nutrition-facts).

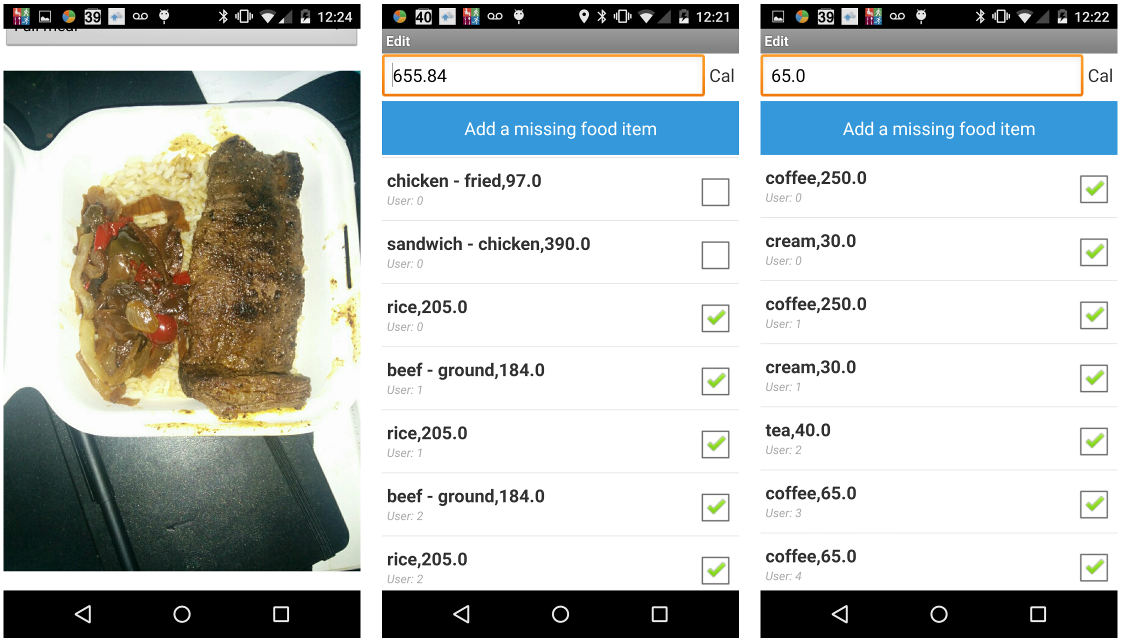
FOOD	AMOUNT	CALORIES
	cup(s)	
	cup(s)	
	cup(s)	
	cup(s)	

(Click Image to Enlarge)

What type of meal is it?

□ Breakfast □ Lunch □ Dinner □ Snack □ Dessert □ Other

Figure 3.10: Food and calorie intake logging based on crowd sourcing using Amazon Mechanical Turk



655.84 Cal

Add a missing food item

chicken - fried,97.0 User: 0	□
sandwich - chicken,390.0 User: 0	□
rice,205.0 User: 0	☑
beef - ground,184.0 User: 1	☑
rice,205.0 User: 1	☑
beef - ground,184.0 User: 2	☑
rice,205.0 User: 2	☑

65.0 Cal

Add a missing food item

coffee,250.0 User: 0	☑
cream,30.0 User: 0	☑
coffee,250.0 User: 1	☑
cream,30.0 User: 1	☑
tea,40.0 User: 2	☑
coffee,65.0 User: 3	☑
coffee,65.0 User: 4	☑

(a) (b) (c)

Figure 3.11: (a)Taking food picture in the app (b) Replies from turker about the food image. Each replies consists of a food ingredient label and a corresponding calorie amount (c) Replies from turkers on a Starbucks coffee.

3.3.2 Efficacy of crowd-sourcing dietary intake

We determined the improvements of MyBehavior crowd-sourcing calorie information with a pilot study. Users expressed interest to see the crowd-source based calorie content of the foods and were satisfied with the accuracy of food labeling and clustering. Specifically we counted the number of foods logged per day over 3 weeks. We have found that the number of foods recorded per day per user with crowd-based approach ($\mu = 4.2, \sigma = 2.5, q_{25} = 2.2, q_{50} = 4.1, q_{75} = 6.0$) is higher than the manual logging approach ($\mu = 2.4, \sigma = 1.7, q_{25} = 1.4, q_{50} = 2.0, q_{75} = 2.9$). This increase is also statistically significant (Wilcox ranksum test, $z = 2.5, p = 0.013$).

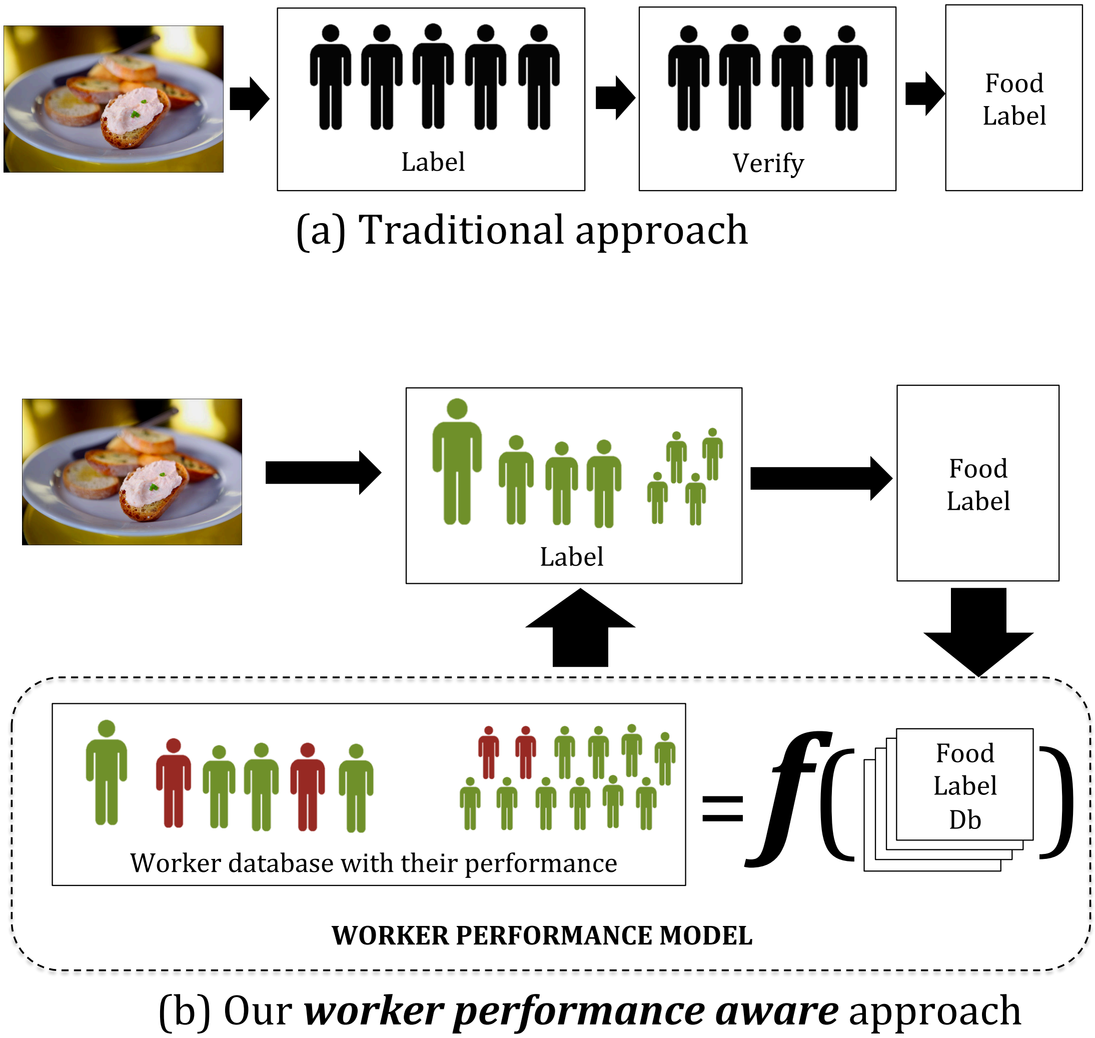
3.3.3 Limitations of crowd-sourcing dietary intake: cost

The increase of food logging with the crowd-sourcing approach is beneficial for understanding user behaviors in greater detail. However the increased adherence also means that the crowd-workers have to be paid. If the cost of hiring crowd-workers can be reduced then end-cost can be reduced largely. In the following, we propose a technique that would reduce the cost of crowd labeling of food by 50%. [126] describes more details of this approach.

3.3.4 Getting accurate crowd-sourced labels cheaply

In traditional crowd-sourcing approaches to food label, each image is labeled by a set of paid workers. Then another set of paid workers review if the labels are correct. The most correct food labels are then defined as final food labels (Figure 3.12a). Such a method is oblivious to a worker's past history of how well s/he performed in labeling tasks. However, some workers are consistently good at labeling while some are bad. Therefore, it is not necessary to verify every food label by another set of paid workers, since labeling quality is likely to be good if the image is labeled by high quality labelers. If costs for such verification workers are removed then the total end saving would be significant in large scale deployments where large number of images would be labeled.

Given this insight, we propose a *worker performance aware* approach (Figure 3.12b), where we maintain a set of high performing labelers to gain high labeling accuracy. We also construct a machine learning model that can determine high performing labelers automatically. The model can look at the reply patterns of labelers and predict their food labeling performance with no cost.



(a) Traditional approach

(b) Our ***worker performance aware*** approach

Figure 3.12: (a) Traditional methods doesn't distinguish between bad or good quality workers. These methods use paid workers to verify food labels to ensure good quality. (b) our approach keeps track of good quality workers and engages them only for good quality food labels.

Dataset

The machine learning model to predict turker performance would be constructed on a dataset of 3925 food images captured real users. 1801 unique turkers labeled these images for content with a total of 27784 turker replies (each reply corresponds to one HIT). Since not all turker replies were accurate and it is hard to verify how accurate all the 27784 turker replies, we select a representative sample of turker replies and reason about different turker's labeling performance. We uniformly select a random sample of 1200 replies out of 27784 replies and verified their quality as acceptable or not[1]. 2 independent verifiers of our research team judged the turker provided labels for acceptability. If there is a disagreement with a label's quality then another verifier judged for acceptability. A turker label is considered acceptable if 2 or more verifiers considered the label to be acceptable. A similar approach to our accepting or rejecting turker replies was used by Thomaz et al. [152]. For rest of the paper, the percentage of "acceptance" for a turker's replies is referred to as *ground truth accuracy*. After the accept/reject process, 70% of turker labels are found acceptable.

A model to identify worker performance

Reply patterns of high and low performing turkers are different. In this section, we propose a machine learning model that exploits the reply pattern differences to predict turker performance. We first introduce some notations. Then we describe the features and machine learning model which we later evaluate. We denote set of turkers as $T = \{t_1, t_2, t_3, ..., t_{|T|}\}$. The set images labeled by a turker t_i is denoted by $P_i = \{p_i^1, p_i^2, p_i^3, ..., p_i^{|P_i|}\}$. Note here the same image can be labeled by multiple turkers. The words replied in image p_i^j by turker t_i is denoted by set $r_i^j = \{r_i^j(1), r_i^j(2), r_i^j(3), ...\}$. For example, if a turker t_i replies to image p_i^j with steak, salad, and fries then $r_i^j(1), r_i^j(2), r_i^j(3)$ will be respectively "steak", "salad", and "fries". Finally, ground truth accuracy for turker t_i is denoted by y_i. Since replies from all turkers are not included in the subsample of 1200 replies in our dataset, y_i is not defined for every t_i.

[1] The labeling interface: http://goo.gl/uWabGy

Features

We introduce several metrics that quantify the difference between high and low performing workers' reply patterns. We also illustrate the efficacy of the metrics with visualization and verbal explanation. We then use these metrics as features for a predictive model.

Turker's reply frequency: Infrequently replying workers often provide highly accurate labels whereas the frequently replying workers can be high or low performing. Fig 3.13c shows the distribution of ground truth accuracy for infrequently replying turkers that replied less than 100 times. Large percentage of these turkers replied with high accuracy. Such turkers constitute nearly 83% of our total 1801 turkers in the dataset with 13% of 27784 total replies. For the remaining 87% replies from turkers with more than 100 replies each, the ground truth accuracy is mixed with both high and low performances as shown in Fig 3.13d. Another way to look the same phenomenon is the changes in cumulative ground truth accuracy as more frequently replying turkers are considered (Fig 3.13e). Mathematically, the Y-axis corresponding to the X-axis or number of replies by a turker is $\frac{\sum_{i=1}^{|T|} y_i * |P_i| * \mathbb{1}_{[|P_i| \leq x]}}{\sum_{i=1}^{|T|} |P_i| * \mathbb{1}_{[|P_i| \leq x]}}$. i.e., we include turkers that replied less than x times and weight the ground truth accuracy with frequency of replies to give more importance to highly replying turkers. From Fig 3.13e, we see turkers with less than 10 replies and 100 replies, the cumulative ground truth accuracies are around 93% and 85%. However, as more high replying turkers are considered the accuracy drops to 70%.

Match index: Replies from under performing turkers consistently do not match with replies from other turkers for same food images. However, the opposite is true for high performing labelers. This happens because for each image some

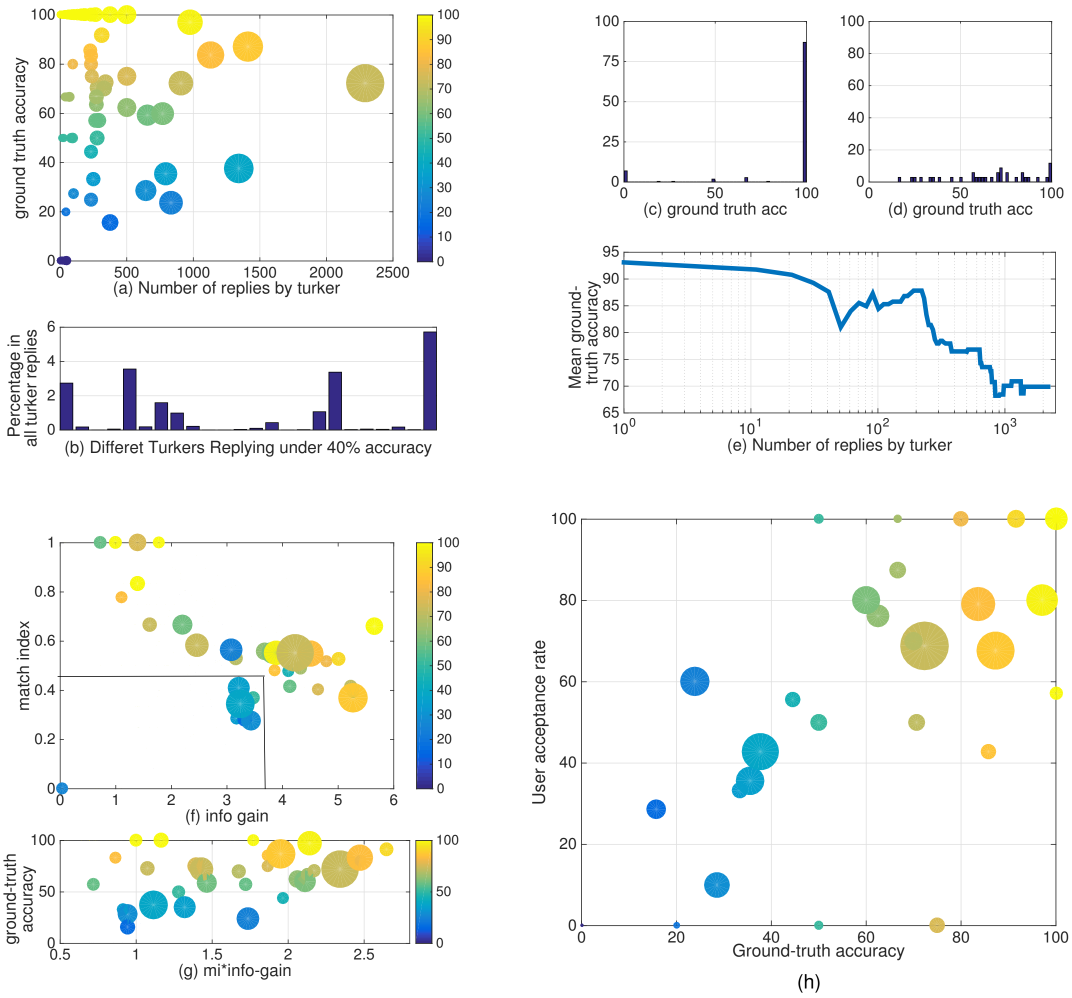


Figure 3.13: (a-b) Turkers replying to larger number of requests sometimes have low ground truth accuracy (c) Turkers who replied less than 100 HITs generally had high percentage correct labels (d) Ground truth accuracy varied widely for turkers who replied more than than 100 HITs (e) Cumulative accuracy as more higher replying turkers are added. Cumulative accuracy was nearly 85% for turkers replying less than 100 HITs (f) Distribution of match index and information gain (g) interaction of match index and information gain correlates with ground truth accuracy (h) Labeling accuracy for turkers replying to larger number of requests have high correlation with end user acceptance.

labels are accurate, since 70% of the replies in the labeled dataset are acceptable. Thus on the same image, accurate labels from a high performing would consistently match with other accurate labels from another high performing turker. However for low-performing turkers, their wrong labels would not match with accurate labels from high performing turkers. In fact, low-performing turkers wrong labels would not even match wrong labels of another low-performing turkers since low-performing turkers can not co-ordinate. With this intuition, we calculate a *match index* MI_{ij} for an image p_i^j a turker t_i replied to as follows: fraction of words in the t_i's reply to p_i^j that matches replies of other turkers to p_i^j. Before the matching, we preprocess the data with simple stemming from Natural Language Processing literature [25]. After stemming, 'Apple' and 'Apples' are considered identically. We then compute a match index MI_i for turker t_i by computing the average of all match index measures MI_{ij}s. The exact equation of match index MI_i is as follows:

$$MI_i = \frac{1}{|Pi|}\sum_{j=1}^{|Pi|} MI_{ij}, \textbf{ and } MI_{ij} = \frac{\sum_{k=1}^{|r_i^j|} \mathbb{1}_{[r_i^j(k)\in r_i^{j'}]}}{|r_i^j|}$$

where $r_i^{j'}$ is defined as the set words replied by turkers other than t_i for image p_i^j. Intuitively, the above equation means if the words of a turker t_i's reply is contained in replies from other turkers to the same image then the MI_{ij} would be high. Now accurate replies from a turker has a high chance to match replies other high performing turkers. Thus a high performing turker will have high MI_{ij}s which means the average of MI_{ij}s denoted as MI_i would also be high.

Information gain: Often under performing turkers use a small set food names for all replies. i.e., replies from under performing turkers lack diversity. On the other hand, replies from high performing turkers exhibit sufficient diversity, since food labels would be different if different users eat different foods. We compute this diversity with an entropy or information gain measure in the following two steps (1) we first construct a dictionary D with unique words $\{d_1, d_2, d_3, ..., d_{|D|}\}$ contained in aggregate replies from turkers. We pre-processed the words using stemming techniques from natural language processing [25] (2) then we compute entropy with the following equation:

$$entropy_i = -\sum_{j=1}^{|D|} p_{ij}\,\mathbf{log}\,p_{ij}$$

$$\textbf{where } p_{ij} = \frac{\sum_{l=1}^{|P_i|}\sum_{k=1}^{|r_i^l|}\mathbb{1}_{[r_i^l(k)=d_j]}}{\sum_{l=1}^{|P_i|}\sum_{k=1}^{|r_i^l|}1}$$

According to the above equation, if the food labels are predictable then entropy is low whereas if food labels are less predictable then entropy would be high. As a result, if turkers replied accurately then their replies would have high diversity and their entropies would be high compared to low-performing turkers.

Figure 3.13f shows the distribution of match index and information gain. In the figure, the large blue dots representing highly replying bad quality turkers are largely located where both information gain and match indices are low. Figure 3.13g shows the relation more prominently where ground truth accuracy positively correlates with the interaction [115] between match index and information gain ($r = 0.48, p = 0.001$).

End-user acceptance rate: Throughout the user study, we gave end-users to correct some of the labels provided by the turkers (see Figure 3.11(b-c)). We compute user acceptance rate of a turker as percentage of the turker's labels accepted by end-users. This user acceptance rate strongly correlates with ground truth accuracy ($r = 0.77, p < 0.0001$). Figure 3.13h shows this correlation of user acceptance rate with ground truth accuracy. However, user-corrections are relatively rare and only 1% (168/27784) turker replies were corrected by end users. Importantly, end-user corrections is more available for frequently replying turkers. 93% turkers replying more than 700 replies received some end-user corrections. In contrary, only 13% turkers under 700 replies received end-user correction.

Turker performance identification model

In this section, we formulate two machine learning regression models to identify turker performance or ground truth accuracy. One model uses the end-user acceptance rate feature and the other does not. The model without the user acceptance rate feature is as follows.

$$y_i' = \beta_0 + \beta_1 * MI_i + \beta_2 * entropy_i + \beta_3 * entropy_i * MI_i$$

where y_i' is the predicted ground truth accuracy. We add an interaction term between match index and information gain since such interaction correlates significantly with ground truth accuracy. Our second model uses end-user acceptance feature. However, we can use end-user acceptance features for only high replying turkers where the feature is more available. The second model is as follows:

$$y_i' = \begin{cases} \beta_0 + \beta_1 * MI_i + \beta_2 * entropy_i \\ +\beta_3 * entropy_i * MI_i, \textbf{ where } |P_i| \leq C \\ \gamma_0 + \gamma_1 * MI_i + \gamma_2 * entropy_i \\ +\gamma_3 * entropy_i * MI_i + \gamma_4 * \text{user\_acc}_i, \textbf{ otherwise} \end{cases}$$

where a separate linear regression model is used with end-user acceptance feature only when the turker t_i replied more than some constant C number of times. Finally, we fit the above regression model in piecewise manner [26] for different turker reply frequency interval. This was done based on earlier observation where turker performance was found to be dependent on frequency. For a window size of 100 in frequency or $|P_i|$ (i.e., 0-100, 100-200, 200-300 and so on), we formulate a different linear regression.

Accuracy of food content labeling

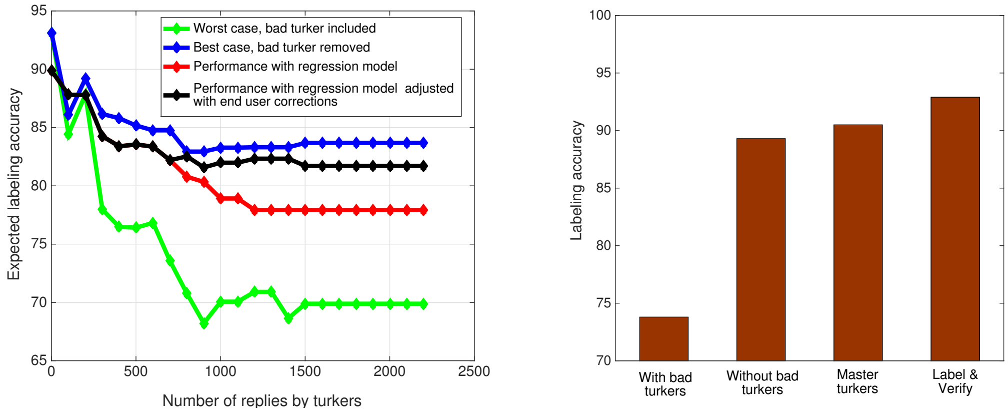
We evaluate the labeling performance of our models in two ways: (1) in an offline evaluation, where we count how many wrong labels are removed from our dataset if we discard low-performing turkers as predicted by our model (2) an online evaluation, where we label new images by turkers that exclude low-performing turkers as predicted by our model. The accuracy of the labeling is then compared with several control conditions.

Offline evaluation: We investigate the fraction of remaining accurate labels, if labels of low-performing turkers from our model are excluded from our dataset. Such a measure would indicate labeling accuracy without bad-labelers in an offline setting. We do the evaluation in two steps (i) we predict labeling performance of different turkers based on our model (ii) we estimate the percentage of remaining accurate labels after removing low-performing turkers from our model. To predict labeling performance of a turker t_i, we train a model with features from $T - t_i$ and predict the accuracy y'_i from the model for t_i. Such *leave-one-out* evaluation is commonly referred as cross-validation to demonstrate generalizibilty in machine learning. Furthermore, the model trained over $T - t_i$ has zero knowledge of the replies of t_i. Thus the prediction for t_i indicates the way our model would perform for an unknown turker. Given y'_i, the quantity $y'_i * |P_i|$ would indicate the expected amount of accurate labels from t_i with our model. Specifically we examine a metric called *expected labeling accuracy*, which is defined as $\frac{\sum_{i=1}^{|T|} y'_i * |P_i| * \mathbb{1}_{[|P_i| \leq x]}}{\sum_{i=1}^{|T|} |P_i| * \mathbb{1}_{[|P_i| \leq x]}}$, to demonstrate efficacy of our model. Here $\mathbb{1}$ is a indicator function and the denominator is a normalization term. Intuitively, the expected labeling accuracy means the cumulative labeling accuracy we can expect if only turkers replying to x number of queries are considered.

Figure 3.14a shows the results. We first define a worst and best case of expected labeling accuracy for different values of x (i.e., number of replies by turkers). The green shows expected labeling accuracy if no bad turkers are removed. This is the worst case scenario for our model. The blue curve on the other hand shows a upper limit or best case, where we discard all turkers with 60%[2] or less accuracy (i.e., ground truth accuracy) known in the labeled training set. Now the red curve shows the expected labeling accuracy if we exclude turkers with a predicted accuracy of less than 60% using the model without user acceptance rate. As can be seen that the model always performs better than worst case. Specially the expected labeling accuray is around 83% if the number of replies from a turker is less than 700. However after including turkers with 700 or more replies, the model performance deteriorates. A possible reason for deterioratation is the heavy tail nature of the turker reply distribution and there is not enough points to learn a reliable model with low number of turkers that replied higher than 700. On the other hand, if we use the model with user acceptance rate for $C = 700$ then we get the black line in Figure 3.14b. This model maintains accuracy of near 83% throughout where the possible upper limit model accuracy is around 84%.

<u>Online evaluation:</u> In online evaluation, we investigate labeling performance in a real-life deployment of our predictive algorithm. We upload a set of 30 images and get labels for these images. These image are randomly selected from our dataset. We check how many accurate food labels we receive if we remove low-performing turkers as predicted by our algorithm. Since a proper evaluation needs comparison with baseline conditions, we evaluate against three conditions when (a) no bad turkers are removed (b) against “master turkers”,

[2] 60% is chosen heuristically. Other accuracy numbers also achieve similar results



(a) **(b)**

Figure 3.14: (a) Predicted accuracy model. If turkers are chosen based on predicted accuracy. Turkers with less than 60% accuracy are discarded. (b) Accuracy of an online deployment of our system.

a list of high performing turkers in Amazon Mechanical Turk (c) with *label and verify* approach used in earlier work [111]. For label & verify, we recruited 5 turkers for labeling and 5 turkers for verifying. For each turker, we paid $0.04. For "master turkers", we could not get master turkers at $0.04. We paid $0.15 for each label from master turkers and recruited 5 master turkers for each food image.

2 independent evaluators in the research team judged how many of these labels were acceptable. If there is a disagreement then a 3rd evaluator judged the labels. Labels are considered acceptable if they are accepted by two or more times by the evaluators. A similar approach to our accepting or rejecting turker replies was used by Thomaz et al. [152]. Figure 3.14b shows percentage of acceptance. Acceptance after removing low-performing turkers with our method (88.7%) outperforms the control condition with no bad turkers removed (72.9%). Master turkers approach performed similarly to our approach (90.5%). Label and verify approach, where we considered food content labels that were accepted by majority of verification turkers, are 92.3% accurate. Although both

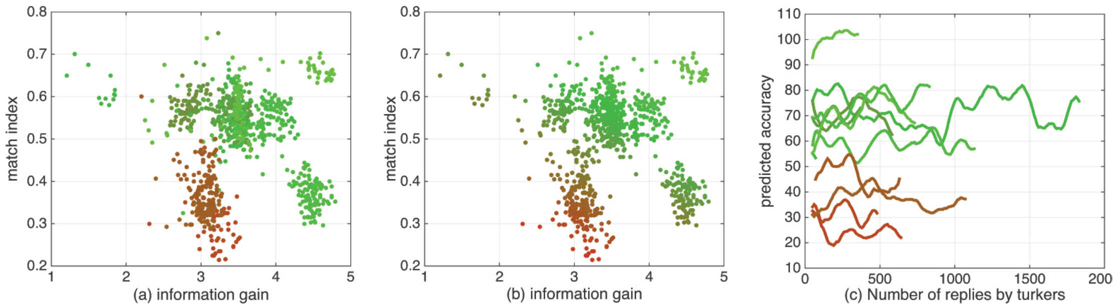


Figure 3.15: (a) distribution of match index and information gain on small windows of turker replies. Relatively red dots represent low quality turkers as defined by ground truth accuracy (b) same as 'a' but more red color of a dot is due to accuracy inferred by our model (c) evolution of inferred accuracy on small windows of turker replies.

master or label and verify approach performed with similar accuracy our approach, these approaches cost significantly more than our approach. We will discuss the cost proposition in more detail later.

Early detection of under-performing workers

So far, we built a model to identify a turker's performance from the turker's total history of replies. However an important problem is to detect low performing turkers early, so that they can be warned and eventually be removed before they adversely affect the labeling.

Previously discussed features also perform well for early detection of low-performing turkers. Figure 3.15a shows the match index and information gain for a window of 50 replies for turkers with more than 500 replies (i.e., both good and bad quality turkers)[3]. Dots from turkers with low quality replies are shown in more red colors (solid red is 0% ground truth accuracy) whereas a good quality turker is shown in more green color (solid green is 100% ground truth ac-

[3] We do not turkers with less than 500 replies since the number of turkers are large and they are generally high performing. Thus including them in the visualization and discussion will crowd the figured unnecessarily. However, in our evaluation we get very similar results for turkers with less than 500 replies.

curacy). The red dots, with low values of information gain and match index, are clearly separated from green dots. A linear regression model can learn these differences. Figure 3.15b shows the results of a 10 fold cross validation results (i.e., only predicted values are shown) of the linear regression model. Again we color code the dots similar to Figure 3.15a but with inferred accuracy rather than ground truth accuracy. We can see Figure 3.15b closely matches with Figure 3.15a.

In order to demonstrate early detection performance, figure 3.15c shows the predicted accuracy on windows of 50 replies over time. Each curved line in the figure is a turker and more green lines represent high performing turkers (i.e., with high ground truth accuracy). A line representing low performing turker is shown in a more red line. Predicted accuracy in Y-axis at a point x on X-axis represent predicted accuracy on the window of $[x-24, x+25]$ replies. i.e., predicted accuracy at $x = 500$ is computed from 476th to 525th replies. In figure 3.15c, more red color lines representing low-performing turkers consistently show low predicted accuracy (i.e., below 50% predicted accuracy) overtime. Thus within 50 replies, it is possible to understand which turker might be low performing. Furthermore, the green lines representing high-performing turkers also show high predicted accuracy, thus a high-performing turker would not be flagged as low performing with a cut off of 50% predicted accuracy.

Cost vs performance analysis

Our system can identify turker performance and early detect bad turkers. However, how much does our system cost to label foods? Furthermore, is the end accuracy acceptable? Our system requires \$0.2 to label a image where we recruit 5 turkers, at \$0.04 each. According to section 4.3.2, 88.7% of these turker labels are accurate. However if we deploy traditional *label and verify* [111] then we can get an accuracy of 92.3%, although it requires an extra $0.2 per image to filter wrong labels with additional turkers at $0.04 each. Finally, if we use master turkers - a set of high quality turkers maintained by Mechanical Turk - then we get 90.5% accuracy. However, assigning master turkers requires a steep price of $0.15 per label [152] with $0.75 per image.

The small accuracy loss would not affect end accuracy significantly, though our method costs significantly less. Our method incurs 50% and 73% less than all verify and master turker based approach respectively. We do so with an 1.8% less accuracy than master turkers and 3.6% less accuracy than earlier work [111]. The end effect 1.8% and 3.6% accuracy loss is minimal given we use majority voting [23] to decide final labels for an image: if 88.7% of turker labels are accurate then may be only 1 out of 5 labels received for an image can go wrong. Thus if we use a majority vote to select right labels then the wrong labels will not be likely included in the final set of labels. Therefore our approach can create accurate labels with 50% or less cost than competing techniques.

Limitations of worker aware approach

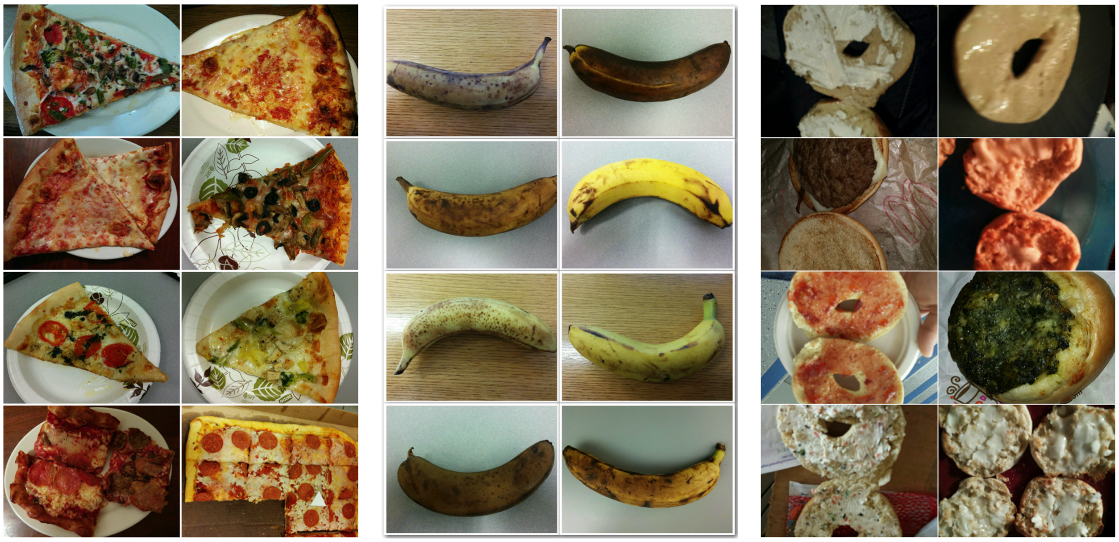
Lack of Long Term Field Evaluation: Despite the efficacy of our system in online and offline evaluation, it is unknown on how our solution will perform in real life longitudinal trials. However, it is not easy to work around our features. e.g., to get around information gain, a turker needs to consistently give garbage food labels that do not match with other labels to increase information gain. It is even harder to go around match index since wrong labels would not match with right labels by other high performing turkers.

Calorie Contents of Foods: In this work, we did not address the estimation of food calorie amounts. Measuring appropriate calorie amount from just food picture is unreliable, since it is hard to estimate size of foods from pictures. Furthermore, sometimes users do not consume the whole food in a picture. As a remedy, we give users control to adjust portion sizes. Then we use the portion size and calories per portion size to measure calories in the food.

Other Limitations: Although Amazon Mechanical Turk allows for incentivizing turkers, we do not incentivize high performing workers. It remains to be seen whether incentivizing the turkers increase turker performance even more. Furthermore, end user corrections are needed to reliably predict performance of frequently replying turkers. We can run similar correction step with other turkers and remove end-users dependence completely from the system. Finally, our approach advocates for selection and engineering of crowd-workers. Since crowd-workers are human labors, there are ethical concerns of our approach. However, we feel that our smart and cost-effective approach would create easier opportunities for larger scale data collection. Larger sized data then can enable completely automated food content labeling solution with machine learning that would not require crowd-worker manipulation.

3.3.5 Mining dietary behaviors

Once a user's dietary behaviors are understood from food logs, we do a simple food ingredient matching to group similar foods. For example, under this scheme burgers with similar ingredients will be grouped together. Figure 3.16 shows 3 different clusters found using this method.



(a) (b) (c)

Figure 3.16: Three separate dietary behaviors. (a) pizza eating behavior for a user (b) banana eating behavior for the same user (c) bagel eating behavior for another user

3.3.6 Pros and cons of the dietary behavior mining approach

The dietary behavior mining approach solves several practical problems namely using crowd-sourcing to increase logging and reducing cost. The human effort from crowd-sourcing was required, since the food detection task is still hard for a computer to automatically do.

However, the research on automated detection of objects in images is progressing rapidly due to the recent advances in deep learning [21]. Krizhevsky et al. [82] demonstrated breakthrough improvements in object recognition on the ImageNet dataset with Deep Convolutional Network [21]. The research classified the 1.2 million images into 1000 categories using a deep neural network with nearly 60 million parameters and 650,000 neurons. GPU computing was used to train the network. But it is hard to do the same at this point for day-to-day food image recognition, because it is hard to get a food database at the scale of million labeled images and perform the high amount of computation inside a phone. However a positive side is that, when a neural network are pre-trained at the scale of millions of images then the neural network can be reused for other image recognition tasks also. The plateclick [162] and im2calorie [100] works have used the pre-trained model from Krizhevsky et al. [82] to classify food images into categories and calories. Nonetheless, these automated food recognition research are still at a nascent stage, and the automated food categorization can categorize only stock food image photos or images from restaurant menus. But as more labeled food images become available, the deep networks will be able to better classify food images at nearly no cost.

3.4 System footprint

Excessive CPU cycles and memory usage can limit the battery life and user experience. In this section, we briefly describe how MyBehavior saves battery and memory during the behavior mining process. We discuss the performance of CPU and memory benchmarks for classifying physical activity and behavior mining. Note that, the food pictures taking and crowd-sourcing calorie estimates run 2-3 times per day and are not performance hogs.

	CPU use	Memory use	Duration
GUI only	0%	9MB	
Activity Inference	0%	9MB	continuous
Activity clustering	2%	19MB	1.5 sec

Table 3.1: CPU and Memory usage with duration for physical activity sensing and behavior mining

Table 3.1 shows a table of CPU and memory usage of different stages of behavior mining process. Since clustering and suggestion inferences are not continuously run and they are performed once a day. In order to make clustering more efficiently, we use an online clustering method called BIRCH [165].

Other than the clustering, the major power-hog in the system is the activity tracking since it is carried out continuously. However due to the duty cycling approach described earlier, we achieved a more than one day battery life on the Galaxy Nexus phone (32 hours). Furthermore, in modern smartphones the activity recognition is moved at the hardware level with low-power processing units to save battery [143]; e.g., iPhone's motion processor [157] and Android sensor hub [59] can infer physical activity at the chip level to save battery.

CHAPTER 4

HEALTH FEEDBACK AS A MULTI-ARMED BANDIT PROBLEM

In the last chapter, we discussed different techniques to collect and analyze physical activity and food data. We also described several techniques to find patterns of physical activity and dietary behaviors. In this chapter, we detail how MyBehavior utilizes the behavioral patterns to generate personalized food and physical activity suggestions. First, we provide an overview of creating suggestions with a Multi-armed bandit algorithm that is grounded in psychology theories of behavior. Then we provide specific technical details of the Multi-armed bandit formulation for physical activity and food suggestions, and how the clusters extracted in the earlier chapter are used. We end the chapter with a pilot study that used MAB and discuss the success and the lessons learned.

4.1 An overview of grounding suggestion in Multi-armed bandit and behavior change theories

MyBehavior's suggestions-generating strategy is grounded in contemporary behavioral science theories: (1) learning theory [97], (2) social cognitive theory [17], and (3) the Fogg Behavior Model (FBM) [52]. Behavior analysis applies learning theory first to assess whether a person has the skills needed to perform a behavior [28]. If so, the next step is to increase or decrease the target behaviors frequency by harnessing its antecedents (ie, its setting and cues) and consequences (ie, reinforcement). For example, if a health suggestion asks a user to swim but the user can't swim (i.e., he never acquired the skills), the user will not follow the suggestion. On the other hand, if a person has performed a behavior before, even if rarely, the skills can be assumed present. The Fogg Behavior Model applies theoretical principles to technology design by creating tools to prompt low-effort actions that can be triggered even when motivation is low [52]. Thus, MyBehavior suggests (ie, cues or triggers) a frequent behavior (eg, a particular walk) that the person often does in a particular life context. This small, low-effort change simply increases the frequency of a behavior that the person already does. Sometimes, instead, MyBehavior suggests an infrequent behavior (eg, bike ride) that would burn more calories and that the person has shown he/she can do, but does only rarely. Social cognitive theory [17], the most widely used behavioral theory, suggests that in order to voluntarily initiate an action, a person needs a sense of self-efficacy or confidence that he/she will be able to perform it. The more frequently the person can be triggered to ride a bike repeatedly in a certain context where bikes are accessible, the more self-efficacy increases, the less effortful the behavior becomes, and the more likely that bike riding becomes a habit.

MyBehavior operationalizes the above theory and formulates the suggestions generation process as a Multi-Armed Bandit model. The key idea behind this modeling strategy is the inherent exploration and exploitation trade-off that defines the task of making suggestions. The goal is to optimize longer-term heath benefits by making recommendations that are relevant and actionable. Exploitation would correspond to seeking high short-term gain by suggesting the most frequent healthiest activity that the user has already engaged in (or suggesting to eat the most healthy, previously eaten meal/snack). Exploration, on the other hand, would be to suggest activities/food that the user performs less frequently or is less healthy than the best activities/ food for the user (so appearing suboptimal), in order to explore if the user is likely to engage in these activities/foods. Exploration can lower short-term benefit, since these "suboptimal" activity/food suggestions are not lower effort by prior theoretical insight and does not have the same chance to be acted upon as the exploit suggestions. However exploration is necessary to accurately understand the space about the activities/food a user is likely to perform/eat in the long run. This knowledge helps craft a strategy that achieves the best long-term health outcome.

In addition to exploit-explore, an important side case that needs to be considered is the lifestyle. i.e., behaviors that were more common or more frequently repeated in the past may not be repeated. e.g., due to seasonal changes, playing outdoor soccer may be impossible to do in cold weather. The simple exploit/explore strategy described earlier would slowly adapt to such changed circumstances. There are faster ways to adapt to such changes. A subclass of MAB models called the *adversarial bandits* can tackle such changes and adapt to changed circumstances more quickly. In MyBehavior, we use an adversarial bandit that use exploit/explore when lifestyle does not change, but can also adapt to changes in lifestyle if that happens.

In the following, we explain in more details on how personalized suggestions are generated. We first introduce the general multi-armed bandit model. We then elaborate on how we use the multi-armed bandit model to create personalized suggestions by utilizing the activity and food clusters.

4.2 A brief overview to MAB

In a multi-armed bandit setting [130], a gambler faces a row of K slot machines $\{C_1, C_2, ..., C_K\}$, with unknown reward distributions $\{\nu_1, \nu_2, ..., \nu_K\}$ (with respective means $\{\mu_1, \mu_2, ..., \mu_K\}$). The gambler has to pull the arms of n slot machines in a sequence over time, including repetition. Let at time instance t the gambler chooses a slot machine C_t, and a random reward $\widehat{r}_{C_t,t}$ is given by drawing from the slot machine's reward distribution ν_t. Now, if $\mu^* = \max_i\{\mu_i\}$ is the maximal reward mean, then the optimal strategy would be to choose the associated slot machine every time. However the information of the best slot machine is unknown to the gambler and she can only approximate the reward distribution of different slot machines while receiving $\widehat{r}_{C_t,t}$ at different time instances. The principal challenge that the gambler faces is to choose a slot machine at time t so that the difference between the reward sum associated with optimal slot machine (i.e., with mean reward μ^*) and the sum of collected rewards is minimized. The difference is commonly referred to as *regret* ρ, and formally the gambler tries to minimize $\rho = n\mu^* - \sum_{t=1}^{n} \widehat{r}_{C_t,t}$.

Given the standard bandit setting described above, there are two kind of bandit problems (i) *stochastic bandits* where the underlying distribution for $\{\nu_1, \nu_2, ..., \nu_K\}$ is fixed (ii) *adversarial bandits* where the underlying reward distributions for slots $\{\nu_1, \nu_2, ..., \nu_K\}$ can change. Often the stochastic assumption is not sufficient for real world problem, where reward distribution of arms can change overtime. For instance, if MAB is used to model user preference for getting web clicks, where a category web link recommendation, e.g., NFL or NBA, is an arm and getting a click is equivalent to receiving an award, then user preference might change overtime for some category, e.g., NFL season nearing Super Bowl. Therefore an adversarial setting might fit the changing user preferences better than stochastic counter part. A common strategy to incrementally select slot machines is the $Exp3$ strategy. Exp3 works as follows: if there is no underlying change in the reward distribution then Exp3 exploits the most rewarding suggestions and seldom randomly explores other suggestions. Explore is necessary since at any point it is not known which are the most rewarding arm in the long run. However, if the underlying distribution of reward function changes then Exp3 adapts. Bubeck et al. [29] has more mathematical details on how the Exp3 adapts, but in the following we give some intuition. Exp3 maintains a probability distribution of selecting for the arms. Let us denote the probabilities as $\{p_1, p_2, ..., p_K\}$. Normally the most rewarding arms have the highest probability values to selected since there is high chance of maximizing rewards: i.e., for an arm C_i if the recent rewards of v_is are high then p_i would be also high. However, when the reward for an arm changes then two cases can happen (i) reward v_i of a previously less-rewarding slot C_i increases. In that case, p_i is low since v_is were low before. Exp3 recognizes this mismatch and increases p_i so that next time C_i is selected more (ii) reward of a previously high-rewarding slot C_i decreases. However p_i is high in that case since v_i was high before. Exp3 appropriately reduces the p_i value so that C_i is selected less.

4.3 Health suggestions as an MAB

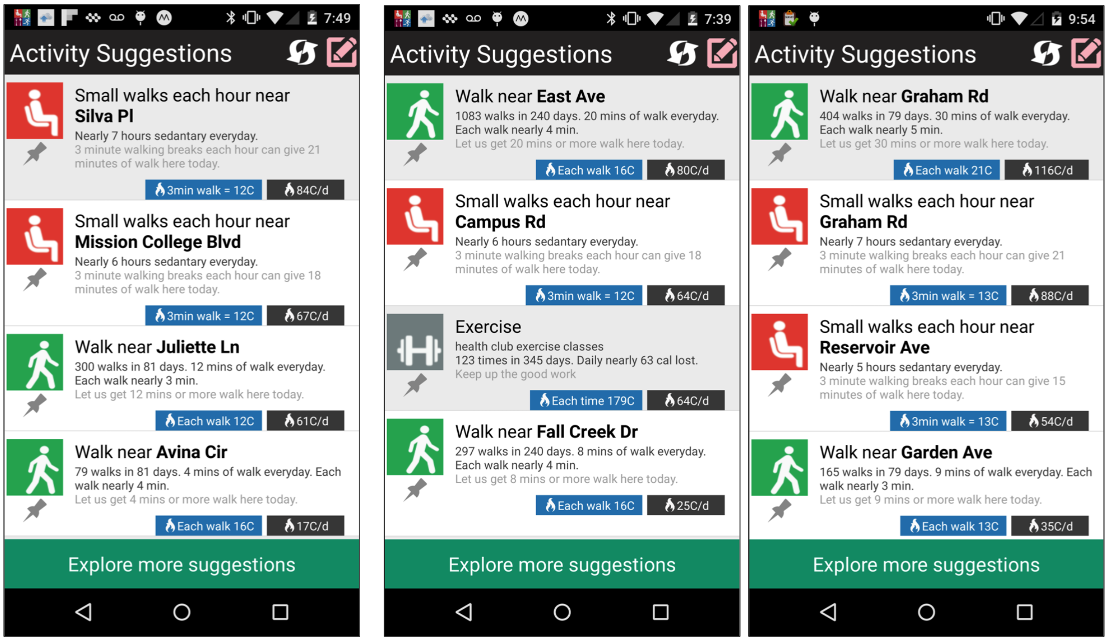
In the earlier section, we provided an overview of the suggestions generation process that utilizes health behavior change theories inside of an MAB. In this section, we provide more details of the MAB, and how it uses uses the behavioral patterns extracted in the earlier chapter.

4.3.1 Physical Activity Suggestion Generation

Let $\{C_1^A, C_2^A, ..., C_{k_A}^A\}$ denote all the activity clusters, where k_A is the total number of clusters. Each cluster C_i^A can be either a stationary, walking, running or manual input exercise type generated using the process described in the earlier chapter. An element in C_i^A, denoted by $c_{i,j}^A$, represents a real-life instance of an activity (e.g., a 5-minute walk to a coffee shop from the office), and each $c_{i,j}^A$ has an associated number of minutes, $minute(c_{i,j}^A)$, representing how many minutes the activity lasted. Using $minute(c_{i,j}^A)$ and the MET score associated with $c_{i,j}^A$'s activity type, we can compute the amount of calories, denoted by $\widehat{v}_{i,j}^A$, spent in doing $c_{i,j}^A$. And with a calorie count for each $c_{i,j}^A \in C_i^A$, we have a distribution $\widehat{v}_i^A$ of calories for the cluster C_i^A.

With the information about C_i^A and corresponding $\widehat{v}_i^A$, we can use a multi-armed bandit to suggest actions from user's past history. However, there are two more practical adjustments we make. First, even though we can suggest an action to be repeated for non-stationary activity clusters, we can not do the same for stationary activities. Many of our participants emphasized that they became aware of their long stationary periods after using MyBehavior and wanted to introduce small breaks between their ongoing daily activities. Therefore, we encourage users to make a small change in their activity habits by suggesting 3 minutes of walking for every hour that they were detected as being stationary. To accomplish this, for an activity $c_{i,j}^A$ representing a stationary episode, we adjust $\widehat{v}_{i,j}^A$ assuming $\left[\ 3 \times minute(c_{i,j}^A)/60\ \right]$ minutes of walking in $c_{i,j}^A$. As a result, we make suggestions that include incorporating small changes in sedentary patterns in addition to capitalizing on health activities that the users already engage in.

The second adjustment we make is based on the frequency of repetitions of an activity. This adjustment is made due to low-effort and self-efficacy theories described before. If we form a reward function for a bandit with the current definition of $\widehat{v}_i^A$, we would always choose the activity clusters that result in the



7:49

Activity Suggestions

Small walks each hour near **Silva Pl**
Nearly 7 hours sedantary everyday.
3 minute walking breaks each hour can give 21 minutes of walk here today.
3min walk = 12C | 84C/d

Small walks each hour near **Mission College Blvd**
Nearly 6 hours sedantary everyday.
3 minute walking breaks each hour can give 18 minutes of walk here today.
3min walk = 12C | 67C/d

Walk near **Juliette Ln**
300 walks in 81 days. 12 mins of walk everyday. Each walk nearly 3 min.
Let us get 12 mins or more walk here today.
Each walk 12C | 61C/d

Walk near **Avina Cir**
79 walks in 81 days. 4 mins of walk everyday. Each walk nearly 4 min.
Let us get 4 mins or more walk here today.
Each walk 16C | 17C/d

Explore more suggestions

(a)

7:39

Activity Suggestions

Walk near **East Ave**
1083 walks in 240 days. 20 mins of walk everyday. Each walk nearly 4 min.
Let us get 20 mins or more walk here today.
Each walk 16C | 80C/d

Small walks each hour near **Campus Rd**
Nearly 6 hours sedantary everyday.
3 minute walking breaks each hour can give 18 minutes of walk here today.
3min walk = 12C | 64C/d

Exercise
health club exercise classes
123 times in 345 days. Daily nearly 63 cal lost.
Keep up the good work
Each time 179C | 64C/d

Walk near **Fall Creek Dr**
297 walks in 240 days. 8 mins of walk everyday. Each walk nearly 4 min.
Let us get 8 mins or more walk here today.
Each walk 16C | 25C/d

Explore more suggestions

(b)

9:54

Activity Suggestions

Walk near **Graham Rd**
404 walks in 79 days. 30 mins of walk everyday. Each walk nearly 5 min.
Let us get 30 mins or more walk here today.
Each walk 21C | 116C/d

Small walks each hour near **Graham Rd**
Nearly 7 hours sedantary everyday.
3 minute walking breaks each hour can give 21 minutes of walk here today.
3min walk = 13C | 88C/d

Small walks each hour near **Reservoir Ave**
Nearly 5 hours sedantary everyday.
3 minute walking breaks each hour can give 15 minutes of walk here today.
3min walk = 13C | 54C/d

Walk near **Garden Ave**
165 walks in 79 days. 9 mins of walk everyday. Each walk nearly 3 min.
Let us get 9 mins or more walk here today.
Each walk 13C | 35C/d

Explore more suggestions

(c)

Figure 4.1: MyBehavior app screenshots (a) a set of activity suggestions for a user (b) a set of suggestions at a different time for the same user (c) a set of activity suggestions for a different user

highest calorie expenditure. For instance, if a weekly gym session or running on a treadmill (manually entered) is one of the activity clusters, the bandit would suggest that as a top action. But if the user takes many shorter walks near her workplace (e.g., 30 instances of 6-minute walks), s/he can potentially burn at least as many calories as a gym visit and more easily fit the activity into a daily routine. We incorporate this observation by defining a new function, $v^A_{i,j} = \widehat{v}^A_{i,j} \times |C^A_i|$, where $|C^A_i|$ is the number of activities similar in duration within C^A_i.

With these adjustments in place, we run a k_A-multi-armed bandit model with $\{C^A_1, C^A_2, ..., C^A_{k_A}\}$ arms with rewards distribution $\{v^A_1, v^A_2, ..., v^A_{k_A}\}$ (with respective means $\{\mu^A_1, \mu^A_2, ..., \mu^A_{k_A}\}$). We pick the 10 top suggestions (given $k_A \geq 10$) using the Exp3 strategy. 90% of the suggestions are exploited and the 10% are randomly chosen as explore suggestions. If $k_A < 10$ clusters are present, then we generate k_A suggestions. In our experiments, after a week of using the system, all of our users had provided sufficient data such that $k_A \geq 10$. Figure 4.1 shows different suggestions generated by MyBehavior. As seen in the screenshots, semantically meaningful messages are added with every suggestion. For suggestions generated by exploiting, MyBehavior asks the user to either continue positive activities (i.e., good calorie foods, walking, or exercise), make small changes in some situations (i.e., stationary activities). On the other hand, suggestions generated during exploration phase, the system asks the users to consider trying out the suggestions. All MyBehavior suggestions change overtime and are different for different users. Figure 4.1(a) and (b) are physical activity suggestions from the same user on different days. Figure 4.1(c) shows suggestions generated for a different user demonstrating the personalization capability of the system

4.3.2 Dietary Suggestion Generation

As mentioned earlier, we utilize a separate bandit with a similar reward function to construct food suggestions. We make a distinction between suggestions for meals and those for snacks (as the number of calories consumed are quite different for these two groups of food clusters), but the bandit optimization process is same for both. Considering $\{C^F_1, C^F_2, ..., C^F_{k_F}\}$ are the food clusters computed during clustering, and $c^F_{i,j}$ being a meal in C^F_i with calorie $cal(c^F_{i,j})$, we compute a function $v^F_{i,j} = cal(c_{i,j)}) \times |C^F_i|$. Similar to activity suggestions, we get a distribution of $v^F_{i,j}$ values for each C^F_i, which we denote as v^F_i. Considering $\{C^F_1, C^F_2, ..., C^F_{k_F}\}$ as slot machines with reward distribution $\{v^F_1, v^F_2, ..., v^F_{k_F}\}$ (with respective means $\{\mu^F_1, \mu^F_2, ..., \mu^F_{k_F}\}$), we run a multi-armed bandit process with $Exp3$. Top suggestions from the bandit prefer frequent meals with low calories compared to unhealthy ones. If a high-calorie food shows up as a suggestion—which can happen during bootstrapping—we still suggest the food but with a message indicating that it should be avoided.

4.4 Feasibility pilot study

We follow the guidelines of Klasnja et al. [80] to evaluate HCI technologies for behavior change, and evaluated the first version of MyBehavior with a small scale pilot trial. The authors in [80] argued that testing a novel HCI behavior change approach with a randomized control trial (RCT), the gold standard for quantitatively evaluating interventions, is not the best initial strategy. RCTs often treat the technology as a blackbox without looking into details of why it is working or whether there are opportunities for improvement. Klasnja et al. suggested researchers to conduct small initial pilot trials to verify whether the technology works in practice and if improvements can be made to maximize the impact of the technology.

To that end, we specifically wanted to evaluate the following research questions: (1) Are MyBehavior suggestions personalized and actionable? Is there evidence that users with personalized suggestions follow a healthier lifestyle? (2) Is there evidence of participants making use of multi-armed bandits exploit-explore strategy during the study? (3) What aspects of MyBehavior are working well? Does any aspect of MyBehavior need improvement before we deploy MyBehavior in a longitudinal trail?

In order to answer these questions, we conducted a 3-week pilot study of MyBehavior. We compared MyBehaviors personalized suggestions to prescriptive suggestions (i.e., pre-specified suggestions that are not personalized to users lifestyle) in a between-subject experiment. We collected a variety of data such as interviews, rating surveys, daily diaries and sensor tracked behavior to evaluate MyBehavior. Methodologically, we took a more pragmatist mixed-method [151] perspective. We triangulated the various data sources to understand whether MyBehavior is effective in providing users personalized and actionable suggestions. In this phase, the emphasis was on developing a viable automated approach to generating behavior change suggestions that is grounded in behavior change theory as well as decision theory and validate its computational feasibility, acceptance by users and future improvements.

4.4.1 Study procedure

To evaluate the early feasibility of MyBehavior, a small 3-week, two-group randomized control trial (RCT) was conducted. The team that supervised the trial included the builders of the MyBehavior app and authors of this paper. This team recruited participants through advertisements placed around the Cornell University campus. In the advertisement, we invited participants to test a new mobile app to help them stay on track for physical activity and food intake. Recruitment was restricted to participants who owned an Android mobile phone and had an interest in fitness. In the app, food was manually logged without the crowd-sourcing capability.

Prior to the study, the investigators arranged face-to-face meetings with the participants and acquired their informed consent. Participants also completed a brief survey to provide demographic data and information about their prior experience with mobile technologies and weight loss/fitness apps. All participants attended a training session, where they installed MyBehavior on their primary mobile phone and received basic instructions, including how to enter their gender, height, and weight and how to set up a weekly weight goal (i.e., lose weight, maintain weight, or gain weight). During the first week, users received a daily summary of their activities and food intake. This baseline week was intended to resemble many modern mobile health apps [106][105], where a calorie count and a lifelog is provided, without suggestions on what behaviors to change.

After the first week, the experimenters conducted an in-depth, semi-structured interview with participants about their experience to date and then randomized participants into control and experimental groups. A random number generator was used for randomization. Assignment was single blind, as the study participants did not know their condition, while experimenters had full knowledge about the assignments.

We provided MyBehaviors personalized context-sensitive suggestions to the experimental group while the control group received generic prescriptive recommendations generated from a pool of 42 suggestions for healthy living, such as walk for 30 minutes and eat fish for dinner. A certified fitness professional created these generic suggestions after following National Institutes of Health resources [109][108]. An external nutrition counselor also reviewed the suggestions to ensure that they were both healthy and achievable. The list of these 42 suggestions is added as Appendix 1 in this paper. For the following 2 weeks, participants continued to log behaviors and receive their respective suggestions on their mobile phones. During the entire study period, we asked participants to complete Web-based daily diaries to better understand their experience in following the suggestions. At the conclusion of the 3-week period, all participants were asked to complete a brief survey about the suggestions provided and were interviewed again face-to-face about their experience with the app.

4.4.2 Participants

We recruited 18 participants, 17 of whom completed the study. Of the 17 participants, there were 13 students (76%) and 4 professionals (24%), 8 females (47%) and 9 males (53%), all between the ages of 18 and 49 ($\mu = 28.3, \sigma = 6.96, q_{25} = 22, q_{50} = 26.3, q_{75} = 36$). All participants reported low-to-moderate levels of physical activity. The majority of participants were experienced mobile phone users - 9 participants (53%) had previous experience using a food diary, and 6 participants (35%) had previously kept an exercise log. After the randomization, participants in the groups were similar in terms of level of active lifestyle and experience with using mobile-based self-management tools. Our sample size was determined based on earlier literature [80][31][42] that suggested that small studies (n ≥ 4) are more suitable to test early feasibility of novel behavior change technologies like MyBehavior.

4.4.3 Outcome Measures

First, we used a suggestion-rating survey to evaluate user intentions to follow the suggestions. Participants completed this survey after the 3-week study concluded. Participants rated the suggestions, by indicating on a 1 to 5 scale, whether they would be willing and able to do the recommended action on an average day - 5 (Strongly Agrees that he/she can follow the suggestion), 1 (Strongly Disagrees). Each participant rated suggestions that she/he saw during the study in an online form. Experimental group participants rated 15 top-ranked - top 8 physical activity and top 7 food - personalized MyBehavior suggestions of their own. On the other hand, the control group participants rated 10 randomly chosen generic prescriptive suggestions. In addition, we quantitatively measured behavior change for all participants using logs of daily physical activity and dietary intakes.

The daily diary and the in-depth, semistructured interviews measured participant feedback regarding the suggestions. For the daily diaries, we queried (1) whether they looked at MyBehaviors suggestions, and (2) whether they made or wanted to make any changes after seeing the suggestions. The semistructured interviews covered users general overall experience with MyBehavior and the quality of the suggestions. Specifically, we inquired about awareness, behavior change, and of any software improvement they would like to see. In addition, in the interview, we asked clarifying questions that explained quantitative results observed from the data.

4.4.4 Analysis plan

Regarding the user's intention of following MyBehaviors suggestions, we gathered ratings for suggestions on a secure website and analyzed the data using RStudio. Since the ratings were in ordinal scale, we used a nonparametric Mann-Whitney U test [115] for statistical significance and effect size.

We measured behavior changes by analyzing activity and dietary logs for statistical significance using MATLAB (MathWorks, Inc) statistics toolbox and RStudio. For each user, we computed median walking length and calories per food item. We considered medians across entire weeks over other central measures since they are less susceptible to spurious noise or outliers (eg, occasional intake of very high-calorie food or atypical, unusually lengthy walk). We did not report changes in running and manually logged exercises in the data analysis as they often require higher effort and are tough to change within the 3 weeks of the experiment. In our analysis, we first considered the number of positive changes. A positive change is defined as a downward trend in median calories in meals, or an upward trend toward longer-length walks over the first week to the third week. We used the Fisher Exact Test [115] to measure the number of positive changes as an effect of MyBehavior. Because of small sample size, the Fisher Exact Test is used instead of the chi-square test for independence. We used a two-sample independent Students t test to measure statistical significance for total walk lengths and total food calories consumed per day. We computed differences in walking distances instead of total number of calories burned, since a walk of a fixed distance can result in a different amount of calories burned for different individuals [12]. We calculated the effect size of walking and eating behavior changes with Cohens d measure.

Finally, face-to-face, semistructured interviews were audio recorded and transcribed. Interview transcripts and daily diaries were then broken down into themes using thematic analysis [28].

4.4.5 Results

Adherence

A total of 17 participants completed the 3-week study, yielding almost 2.1 million recorded physical activity instances, amounting to more than 8000 hours of physical activity. During the same period, participants labeled nearly 850 images of food with annotations.

User Acceptance of MyBehavior Suggestions

In the suggestion-rating survey, the experimental group ($\mu = 3.4, \sigma = 1.2, q_{25} = 2.75, q_{50} = 3, q_{75} = 4$), with MyBehavior suggestions, intended to follow personalized suggestions more than the control group ($\mu = 2.5, \mu = 1.6, q_{25} = 1, q_{50} = 2, q_{75} = 4$) intended to follow the generic suggestions. A nonparametric Mann-Whitney U test [32] found this difference to be statistically significant ($P < .001, 95\%CI = [0 - 1.001]$, effect size = 0.99).

Physical Activity

Figure 4.2 shows the distribution, in the form of box plots, of walking lengths over time for the experimental (left-hand image) and the control (right-hand image) groups. For each week of the study, we computed these distributions for the different users. To ease interpretation, we joined the median per week with thick green or red lines for each user. A green line implies a positive change as discussed in the data analysis section. A red line indicates the reverse negative trend. We used a log scale for walking-length distribution since walking-length distributions have heavy tails [58].

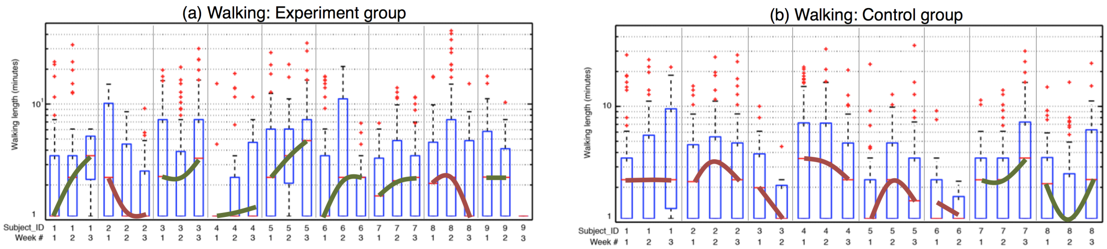


Figure 4.2: Box plots showing the distribution of walking lengths for the experimental group (a) and for the control group (b) over the 3-week study. We joined the medians of distributions and showed each trend as a thick green line (increasing trend) or red line (decreasing trend) for walking length.

For walking, 78% (7/9) of participants in the experimental group (Figure 4.2, left-hand image) showed positive trends, whereas 75% (6/8) of participants in the control group (Figure 4.2, right-hand image) exhibited negative trends. A Fisher Exact Test found this ratio in the number of positive changes between the experimental and control groups statistically significant [115] (P = .05). In addition, MyBehavior users walked an average of 10 minutes more per day within the experiment phase (i.e., from the first to the third week). However, we did not observe any change for the control group. A two-sample t test found this difference in change of walking duration to be significant ($t_{15} = 2.1, P = .055, 95\%CI[-0.23, 19.052], d = 0.9$).

Qualitative data from daily diary and face-to-face interviews largely supported this quantitative result. However we also observed some important subtleties. First, participants in the experimental group described the activity suggestions to be actionable and relevant to their lives. Control group participants appreciated that the generic suggestions reminded them of good habits. However, they often faced problems incorporating the suggestions into their daily lives. The following quotes were taken from the daily diaries of participants.

> Those suggestions are quite good, which reminds me not to sit too long in one place. [Experimental group participant 1]

> The exercise suggestions made me want to do some more activities and be less stationary. Seeing how long I have been stationary and the low frequency of activity made me want to make a change. [Experimental group participant 5]

> Try to get up from my desk more often...added walk" notes to my calendar. [Experimental group participant 2]

> I did some walking where I normally walk. The app now shows I walked

> there 26 times. The app makes me feel that I can do it again since I have done the same walk many times. [Experimental group participant 7]
>
> The suggestions encourage me to do/plan exercises for the near future...It reminds me that some foods are better than others. [Control group participant 1]
>
> They seem like good generic suggestions. The kind you would read...as tips in a health magazine or some such...[Control group participant 4]

Some MyBehavior users reported that even the non-frequent explore suggestions were actionable and expressed interest in acting on them. For instance, experimental group participant 7 said the following in his/her daily diary:

> I saw a walk to my nearest bus stand listed. Normally, I drive my car to go to my office. But looking at the extra walking I got while going to the bus stop makes me think about doing it often and making it a habit. [Experimental group participant 7]

Results from interviews also revealed that participants at various stages of active lifestyle reacted to suggestions differently [122]. For the experimental group, participants who were considering making changes expressed that they became more self-conscious about their behavior and they were eager to follow the suggested changes (e.g., starting to walk more near home, or continuing runs on treadmills). Comparatively, users likely maintaining an active lifestyle expressed that the suggestions reflected their current healthy behavior and considered them as good reinforcements. However, participants in the maintenance phase wanted to change their stationary behavior in the office with occasional small walks. For the control group, users were frustrated because the suggestions were not always feasible and did not blend with their routines and lifestyle. Control group users maintaining an active lifestyle were unaffected by generic suggestions and continued their regular behavior across weeks. For example, control group participants 7 and 8 were maintaining-participants and their behavior showed no negative trends in Figure 4.2 (right-hand image). Control group users who did not already have a maintaining lifestyle gradually became less active or made poorer food choices after the initial phase of the study.

Finally, on a few occasions, MyBehavior suggestions were hard to follow or did not reflect user preferences. For example, one user reported in the interview that he used to play soccer with his friends but his friends recently moved to a new location. He could no longer play soccer, which MyBehavior was suggesting. In addition, often user-preferred activities are not top MyBehavior suggestions. For instance, one user preferred to swim even though she did not do it often. Finally, experimental group participant 8 (subject 8 in Figure 4.2, left-hand image, with negative trends) reported an inability to follow MyBehavior suggestions because of a looming work deadline during the study.

Dietary Behavior

Figure 4.3 shows the distribution, in the form of box plots, of meal calories for the experimental group (left-hand image) and the control group (right-hand image). For each week of the study, we computed these distributions for different users. Similar to walking-behavior graphs, we joined medians across weeks to show positive or negative changes for each user.

For caloric intake, 78% (7/9) of participants in the experimental group showed positive trends (green lines in Figure 4.3, left-hand image), and 57% (4/7) of participants in the control group showed negative trends (red lines

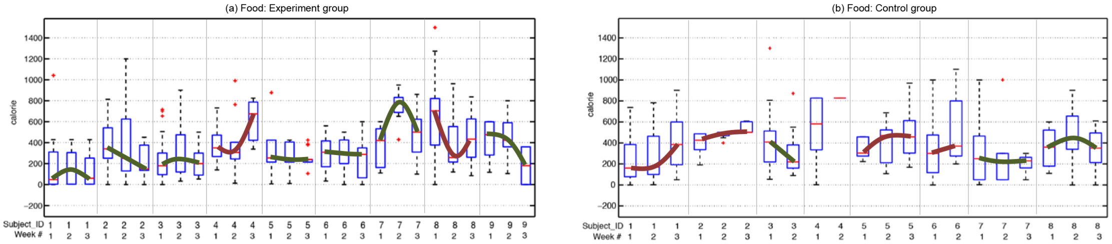


Figure 4.3: Box plots showing the distribution of food calories for the experimental group (a) and for the control group (b) over the 3-week study. We joined the medians of distributions and showed each trend as a thick green line (increasing trend) or red line (decreasing trend) for median food calorie intake.

in Figure 4.3, right-hand image1 participant had insufficient data). However, a Fisher Exact Test found this to be nonsignificant ($P = .15$). For control group participants, we also found their average median calories per day to increase by 211 calories ($\mu = 211.7, \sigma = 263.07, q_{25} = -31.25, q_{50} = 187.5, q_{75} = 429.35$) from the first week to the third week. Comparatively, the experimental group showed an average calorie per day decrease of nearly 100 calories ($\mu = -99.3, \sigma = 481.27, q_{25} = -527.83, q_{50} = -37.3, q_{75} = 87.5$) from the first week to the third week. This change was not significant in a two-sample t test ($t_{12} = 1.3234, P = .21, 95\%\ CI = [-201, 822.96], d = 0.72$).

In qualitative feedback, similar to physical activity suggestions, experimental group users found the suggestions to be more actionable and reported to make more changes compared to control group users who found the suggestions to be hard to work on. This feedback is illustrated in the following quotes from participants daily diaries.

> The pictures of my meals are very useful to keep track of what I've been eating in the past. People tend to forget about their habits, but pictures in this case are a nice way to bring your food history in front of your eyes. [Experimental group participant 9]

> The suggestions remind me that some foods are better than others. [Control group participant 1]

> It recommends me to eat stuff that I don't have at home. [Control group participant 4]

> These suggestions don't take into account my dietary restrictions. [Control group participant 5]

Similar to activity explore suggestions, MyBehavior users often found the explore suggestions to be actionable.

> I just wanted to see what it was...These ones [explore suggestions] seemed to pick up some "good" food habits. [Experimental group participant 4]

Finally, users reported manual food logging to be time-consuming in the interview. However, they also reported that this manual process made them more aware of their foods. Consequently, control group participants reported to make dietary changes without personalized suggestions.

4.5 Success and Lessons Learned for the Pilot Deployment

4.5.1 Principal Findings

To our knowledge, MyBehavior is the first system to automatically provide personalized suggestions that relate to users lifestyles. In the quantitative results, MyBehavior users demonstrated superior behavior changes compared to the control group. Qualitative measures from the face-to-face interviews and the daily diaries confirmed that the suggestions indeed were perceived to be personalized to their lives. This concordance of superiority in both quantitative behavior change and qualitative user perception makes MyBehaviors automated health feedback approach very promising and provides support for longitudinal studies and future investigations into automated personalization approaches.

Specifically, in the evaluation, users rated that they could follow MyBehavior personalized suggestions more than the control condition suggestions. Results also revealed a significant change in walking behaviors for MyBehavior users. In qualitative measures, users reported MyBehavior activity suggestions to be more actionable. Interestingly, although users qualitatively reported the dietary suggestions to be more actionable, dietary behavior changes were not found to be different between the groups. This finding could be due to the manual-logging nature of food intake being sufficient for behavior change alone. The manual process of food logging might produce self-awareness and reflection. Indeed, past research demonstrates that simple logging can improve ones food consumption behavior [145]. However, food logging is an arduous process and it is often hard to continue for an extended period.

Nonetheless, the pilot study with MyBehavior explores a unique space for health feedback. Earlier studies in this domain predominantly focused on overall behavior [38][89], tailoring [84], or self-tracking without deeper data analysis and personalization. MyBehavior takes a data mining approach to automatically find contextualized suggestions from logged data. This automated approach also relieves users from the burden of self-analyzing their data. Thus, MyBehavior is a marked departure from previous self-monitoring programs found in the literature, where users themselves decide on how to make changes on their own. MyBehavior suggestions relate to a users existing behaviors, making them actionable as the user is told where and when to act on them. Furthermore, unique sets of suggestions are generated for each user based on their routine and lifestyle. The literature on N-of-1 approaches [48] argue that such personalization should yield better efficacy than one-size-fits-all or tailored-suggestion approaches, where similar suggestions are provided to users with similar characteristics (e.g., age, gender, daily calorie intake, and loss).

4.5.2 Lessons Learned

Despite this promising direction, the automated data-driven personalization approach of MyBehavior brings its own challenges. Manual logging of food and exercise, in addition to automated logging, are necessary for proper functioning of MyBehavior. Qualitative interviews revealed that manual food and exercise logging were often burdensome. Future iterations of MyBehavior could use crowdsourcing-based semiautomated approaches to decrease the burden of manual food journaling [111]. Finally, interviews also highlighted the importance of considering contextual changes in users' lives and preferences. Thus, giving users control in deciding which suggestion they want to follow is required for well-accepted personalization.

4.5.3 Limitations

An important limitation is the short-term and small-scale nature of the study, which makes it difficult to make definitive conclusions. However, the study helped us to identify the potential efficacy of MyBehavior and pinpoint design improvements for future deployments. Indeed, Klasnja et al [80] argued that such short-term studies with similar evaluation goals as in our study are often more suitable for new and untested behavior change technologies like MyBehavior. Another limitation was that the non-personalized suggestions were sometimes too specific, for example, “walking with a dog”. In the daily diaries, some users reported that they could not follow this suggestion since they did not own a dog. While designing generic suggestions, we tried to find suggestions that most users could follow, without being overly generic. However, there will always be exceptions where a suggestion does not fit ones lifestyle.

4.6 System footprint

In this section, we describe the system foot print for running the multi-armed bandit algorithm. The memory footprint and CPU usage are both low in a test on Android Galaxy Nexus phone. Memory footprint is 19MB while 2% CPU is used to run the MAB. The MAB finished running within 1.5 seconds. Note here that the MAB runs once a day in MyBehavior to refresh the suggestions. Therefore the impact for running MAB is negligible to the phones battery performance.

The system footprint is low because MAB is an online learning technique. Unlike batch learning algorithms like Markov Decision Process (MDP), online learning does not require large amount of computation to learn and adapt. Only the latest data points need to processed to update the model. Furthermore, MAB have less parameters compared to other planning techniques like MDP. Therefore it is easy to learn and adapt MAB models from less amount of data.

4.7 Discussion and related work

We conclude this chapter with several related works to personalizing recommendation. We discuss different algorithms alongside the reasons of choosing or not choosing them for MyBehavior. In addition, since MyBehavior can be framed as both a recommender system and an application of reinforcement learning, we divide the related works into two parts; one part dedicated to recommender systems and the other part discussing reinforcement learning.

4.7.1 Related works in recommender systems

Recommender systems is one of the biggest successes of computer science in the past few decades [77]. These systems utilize a large amount user generated ratings and behaviors to learn patterns of preference and subsequently recommend contents that fit a user's likings and needs. It is not possible to mention all the research that has been done, but the most common techniques in recommender systems are discussed. Subsequently, the positioning of MyBehavior in the existing work on recommender systems is discussed.

A brief overview to common recommender systems algorithms

Collaborative filtering: Collaborative filtering (CF) is probably the most commonly used algorithm in recommender system [77]. CF's underlying assumption is that similar people have similar tastes/preferences. Technically, CF models the world as follows:

$$Users \times item \rightarrow rating$$

where $User$, *item*, and *rating* respectively represent users of the system, the items to be recommended [1] and the overall quality of the recommendation. CF does a nearest neighbor search to find users who rated similar items identically. Then CF recommends items to a user that other similar users rated highly. A potential advantage of CF is that there is no need to detailed understanding of the structure/patterns/values of users or items; similar users already have similar values or tastes which are embedded in their item ratings. Therefore, it is possible to give effective recommendations by simply matching tastes. There is no need of a detailed understanding of the reasons, which is often harder task, behind the common tastes and value. Another advantage of CF is the capability to recommend novel or serendipitous items; e.g., if a user has not seen movies that similar users highly preferred then CF can recommend those movies that are novel or serendipitous to the user. Model based techniques, which would be described later, can not provide such novel or serendipitous recommendations because they rely on data or movies that the user watched or rated before. Nonetheless, there are also a few disadvantages to CF. CF needs user rated data to building models, and users also need to rate a few items to get recommendations. Furthermore, CF is a black box technique since it employs no understanding of the structure of users or items. Also, CF assumes some users are similar, but users can have important differences according to N-of-1 or small data as discussed in Chapter 1. In addition, some knowledge of items or suggestions are necessary for health apps, since suggestions have to respect behavior change theories and promote change (Chapter 2). Finally, CF is not an adaptive algorithm that can learn and change based on user behavior and their response to suggestions. On the other hand, the purpose of health recommendation is to promote and model 'change'.

[1] items are same the suggestions in MyBehavior

Context-aware-recommendation: MyBehavior is a recommender system that falls into the category of context-aware systems. There are existing recommender systems work that apply context awareness [10][128]. These techniques however deploy a modification of collaborative filtering in the following way:

$$Users \times item \times context \rightarrow rating$$

where the goal is to find similar users who rated similar items in similar contexts identically. For instance, a movie recommender system can recommend movies that similar users liked that also fits for a weekend (here weekend is the context). Pertaining to contexts involving users location, a typical context aware recommender systems is a ″tourist guide″ where places-of-interests are suggested based on the current location [78]. However, collaborative filtering, with an extra dimension of context, means that even more data is needed than traditional collaborative filtering, because for each specific variety of context, we would need more data for each user and item combination. Furthermore, tourism or visiting a new place is different from making a user active within their lifestyle: people would prefer to visit most popular places when they are touring a new place, but people do not normally take tour into other people's lives or routines as part of daily activities. In other words, people are all different in their lifestyles, and it is necessary to consider the difference when suggesting small changes to a user′s routine.

Content-based-recommendation: Content based recommender (CBR) systems use a different reasoning compared to collaborative filtering in constructing recommendations. CBRs find some attributes or characteristics of the items that users preferred before. Formally, a CBR models the following function:

$$rank = f(\text{past user behavior, characteristics of items})$$

where f is the objective or ranking function that maps past user behavior of preferring different items and the inherent properties of the items to an order/ranking of user preference. The higher ranked items are more preferred. An example of CBR can be the following: a webpage recommender system wants to recommend pages that reflects a user's preference. The system analyzes a user's website browsing history or bookmarks to find similar websites the user visits, e.g., sports or entertainment news etc. One way to find similar items is to analyze the contents, e.g., tf-idf or topic model, of a webpage and find webpages with similar contents [75][27].

There are a few potential advantages to CBRs. The ratings can be constructed implicitly with domain knowledge, and without an explicit rating given by the user. "Click through rate" is a simple way to define ranking where a higher number of click is assumed as a implicit measure of user preference. Similarly an item might be suggested for purchase because they are available in a local store and therefore the item is easy to acquire without the shipping delay. A general problem with CBRs is that they can not introduce novelty or serendipity, because only users past data is used to make suggestions.[2]

MyBehavior is a content-based recommender system

MyBehavior is a content-based recommender system, where domain knowledge is used to construct the objective function. We utilized the domain knowledge of self-efficacy and low-effort theory from psychology to form an objective function from physical activity and food data collected from a user. Collaborative filtering is not used, because prior data from other users are needed to

[2] A hybrid recommender with CF and content-based mixed can give novel recommendations [30].

construct recommendations. Furthermore, collaborative filtering utilizes user similarity, which N-of-1 literature is in conflict with.

4.7.2 Related works in reinforcement learning

MyBehavior is a content-based recommender system. But, a content based recommender system can be built in several ways. e.g., a supervised learning can be used to learn, where a rank function is learnt as a function of past user behavior and item properties. However, supervised learning requires that some representative data are available ahead of time and there is no later adaptation necessary, because the world would not deviate from the representative data. Both of these assumptions are hard to satisfy for health recommender systems, because of two reasons (i) user data becomes available incrementally over time. Therefore, it is hard to learn the best suggestions/action without seldom exploration, (ii) user behavior would change as part of interventions. which means there is a need for adaptation. A ***reinforcement learning (RL)*** algorithm can fit the problem better, since they can learn from incrementally available data, create recommendations, and follow up or adapt based on the user reactions to the recommendations [149]. We will provide a brief description of the different reinforcement learning technique, followed by a discussion on the reinforcement learning algorithm, i.e., the multi-armed bandit, chosen for MyBehavior.

Associative vs Non-associative reinforcement learning: In RL, an agent takes an action (or recommends), evaluates the actions by receiving feedback/reward and updates its policy for future recommendations. One way to decide actions is make the actions depend on context or environment. For instance, a physical activity suggestion to walk outside is more appropriate in warm weather compared to a cold one. RL algorithms that take different actions depending on different contexts are called *associative* or *contextual* RL algorithms. Markov decision process (MDP) and contextual-bandits are *associative* reinforcement learning techniques [149]. However for *associative* RL algorithms, the modeling of context is a challenge because real world contexts are hard to understand and quantify [149]. Furthermore, for personalized systems, different people can experience same context differently, since understanding context can depend on personal interpretations of the situation and context. e.g., in an office space, there are lot of people, but most people are doing different things which therefore make their interpretation of the context of an office different. A simpler approach is to ignore the state or context all together and find the most useful actions. The category of RL algorithms that ignores contexts are called the *non-associative* reinforcement leaning algorithms. Multi-armed bandit is one of the most popular *non-associative* RL algorithms.

Stochastic (stationary) vs adversarial (non-stationary) reinforcement learning: Another categorization of RL algorithm deals with its capability to handle change over time if the user changes their behavior. Stochastic RL's assume that the users do not change overtime. On the other hand, adversarial RL considers that changes can happen in rewards received for the same actions across time.

MyBehavior is adversarial and non-associative

Due to the difficulty of accurately modeling context, which is partly due to the lack of any available context data apriori per person, MyBehavior's initial version uses a non-associative version of RL. i.e., there are no separate policies for different situations and context. Nonetheless, the results of MyBehavior show that such non-associative systems, in fact, can create effective suggestions. In future work, associative version of MyBehavior can be considered where large amount of user data is available and suggestions can be made as dependent on different user contexts. Finally, we choose adversarial RL, with the exp3 strategy, to adapt to lifestyle changes.

CHAPTER 5

HUMAN-IN-THE-LOOP

In the last chapter, an automated way to generate and rank suggestions using Multi-armed bandit (MAB) was described. However, a general problem of most automated ranking system is that they often do not capture all the complexities of human needs and preferences []. Two of such scenarios were already described in the last chapter, and for completeness I briefly describe them again below:

1. People are often highly motivated about certain activities that they do not regularly do or have done repeatedly in the past. For example, several users wanted, "going to the gym" as a top suggestion even though they did not frequently go to gym in the past. As a result, gym is not probably low-effort or easy-to-do.
2. Contexts and circumstances of a user's life can change. Therefore an actionable suggestion in the past may not stay actionable in new life circumstances. Although MyBehavior can dynamically adapt to lifestyle changes, on occasion MyBehavior took time to adapt.

MyBehavior addresses these two limitations by providing control to the users over the suggestions. Users can remove, up-vote or down-vote specific suggestions. However such user preference can destroy the low-effort optimization already done by the Multi-armed bandit from the earlier chapter. Therefore MyBehavior also balance between low-effort optimizations and users preference using a principled algorithm, called *Pareto-frontier*. The balancing criteria is also grounded in psychological theories of behavior change such that the resulting suggestions are highly actionable.

In the rest of the chapter, I describe details of the human controls in MyBehavior. First, I argue the necessity of keeping users-in-the-loop with detailed grounding into existing theories of behavior change. These theories would later also guide the ways the human inputs can be incorporated into a ranking algorithm.

5.1 Necessity of Human-in-the-loop

Above, we described two cases where incorporating human control can increase experience. Both of the these cases can be made stronger with theories from behavior change and context aware computing.

We start with the first observation, where people might prefer suggestions that are not easy-to-do and are not ranked higher by the MAB algorithm discussed in the last chapter. B.J. Fogg's behavior model (FBM) argues that an individual may undertake a hard-to-do action when they are motivated. According to Fogg, any health suggestion needs to ensure that users are sufficiently *motivated* and/or have high *ability* to implement the suggestion. Figure 5.1 shows a near redraw of the diagram of Fogg's model from Chapter 2. Note the diagram is conceptual, and the axes do not have any unit. A suggestion above the *activation threshold* has high enough motivation or ability or both to be actionable. Typically highly rated MAB suggestions are low-effort and thus located in the right (e.g., the green stars in the figure). However a not-so-low-effort suggestion, denoted by the red-star in the left-top of the figure 5.1, may be actionable since the user is highly motivated to follow the suggestion.

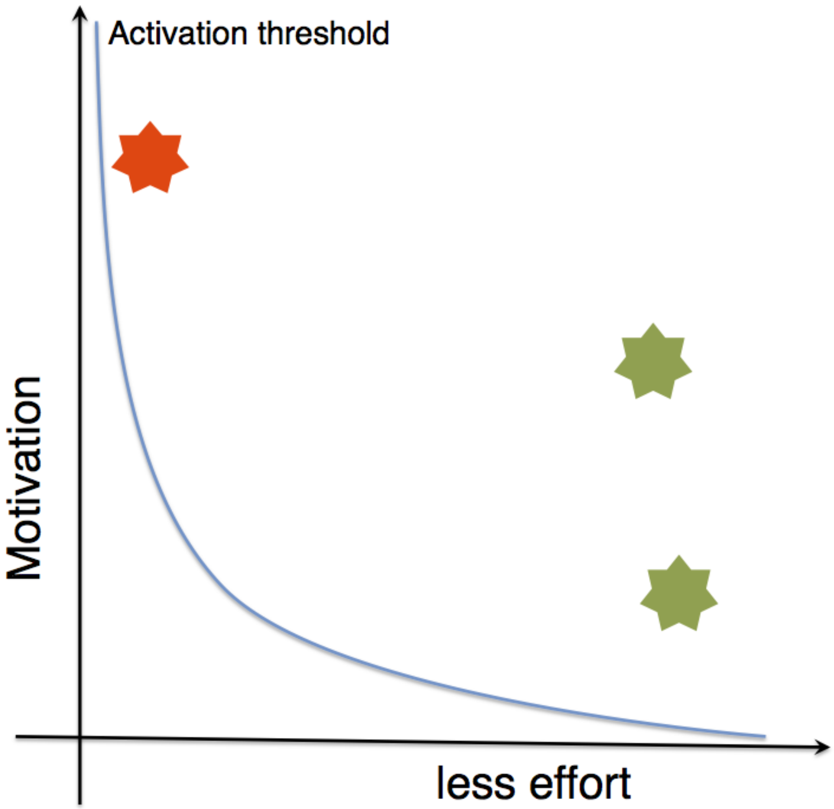


Figure 5.1: BJ Fogg's Behavior model

Given such insight of FBM, let us revisit the case where users wanted to go to gym even though gym is not highly ranked by MyBehavior's low-effort optimization. Since users think that they can go to gym (i.e., gym is actionable) even though they did not go to the gym repeatedly before, the users might be highly motivated to go to gym if we follow FBM. Therefore it is necessary to incorporate suggestions that users are motivated towards, since according to FBM these suggestions are actionable even though they are not low-effort, and rank them some how high inside Mybehavior.

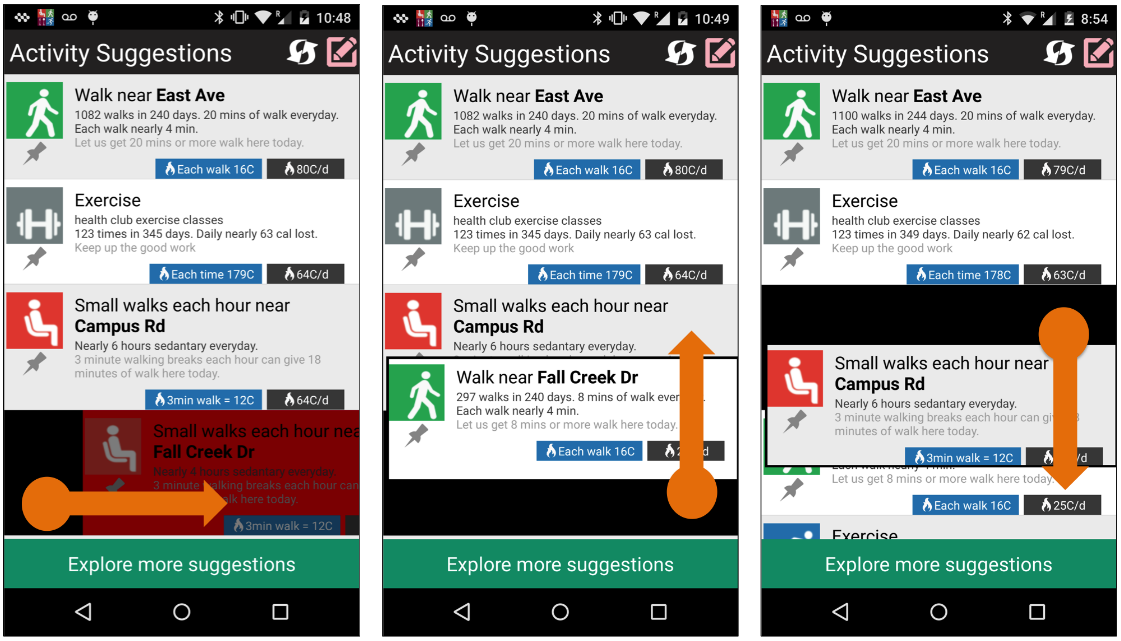
Regarding the second observation, where users could not follow a few suggestions because their life circumstances changed, MyBehavior was slow to adapt. The existing version of MyBehavior is tied to locations and past behavior, and there are other aspects of context or life that are hard to capture with the sensing technologies available. Dourish [45] in his book “Where the action is” and Lucy Suchman [148] in her work on “situated action” argue that it is hard completely understand context by automatic means. They argue that the meaning of context often depends on subjective observation, feelings and interpretation of the situation. These meanings can be partially dictated by culture, life circumstances etc. Such complexity to meaning is often hard to know objectively from sensors. For instance, when a user’s buddies left and he could not play soccer anymore then it is a context switch that is hard for MyBehavior to capture automatically. However, the user is aware of that the context switch happened. Hence a simple control to remove the suggestions would suffice to correct context changes that MyBehavior failed to understand.

5.2 Incorporating Human-in-the-loop

MyBehavior provides 3 different user interactions to incorporate user preferences. In the following, I first describe the interactions. Subsequently, I discuss how the user preferences are balanced with Multi-armed bandit generated rankings to create a final ranking of most actionable suggestions.

5.2.1 Interactions for human control

MyBehavior provides three different user controls/interactions to incorporate user preference. First interaction allows a user to remove the suggestions that the user does not want or is unable to follow due to a change in life circumstances. In terms of interaction, users can swipe from left to right and remove suggestions (Figure 5.2(a)); a removed suggestion is never considered in the



Activity Suggestions

Walk near **East Ave**
1082 walks in 240 days. 20 mins of walk everyday.
Each walk nearly 4 min.
Let us get 20 mins or more walk here today.
Each walk 16C | 80C/d

Exercise
health club exercise classes
123 times in 345 days. Daily nearly 63 cal lost.
Keep up the good work
Each time 179C | 64C/d

Small walks each hour near **Campus Rd**
Nearly 6 hours sedantary everyday.
3 minute walking breaks each hour can give 18 minutes of walk here today.
3min walk = 12C | 64C/d

Small walks each hour nea
Fall Creek Dr
Nearly 4 hours sedantary everyday.
3 minute walking breaks each hour can
lk here today.
3min walk = 12C

Explore more suggestions

(a)

Activity Suggestions

Walk near **East Ave**
1082 walks in 240 days. 20 mins of walk everyday.
Each walk nearly 4 min.
Let us get 20 mins or more walk here today.
Each walk 16C | 80C/d

Exercise
health club exercise classes
123 times in 345 days. Daily nearly 63 cal lost.
Keep up the good work
Each time 179C | 64C/d

Small walks each hour near **Campus Rd**
Nearly 6 hours sedantary everyday.

Walk near **Fall Creek Dr**
297 walks in 240 days. 8 mins of walk ever
Each walk nearly 4 min.
Let us get 8 mins or more walk here today.
Each walk 16C | 2 d

Explore more suggestions

(b)

Activity Suggestions

Walk near **East Ave**
1100 walks in 244 days. 20 mins of walk everyday.
Each walk nearly 4 min.
Let us get 20 mins or more walk here today.
Each walk 16C | 79C/d

Exercise
health club exercise classes
123 times in 349 days. Daily nearly 62 cal lost.
Keep up the good work
Each time 178C | 63C/d

Small walks each hour near **Campus Rd**
Nearly 6 hours sedantary everyday.
3 minute walking breaks each hour can gi 8
minutes of walk here today.
3min walk = 12C | /d

Let us get 8 mins or more walk here today.
Each walk 16C | 25C/d

Exercise

Explore more suggestions

(c)

Figure 5.2: Keeping human in the loop (a) dismissing a suggestion by removal (b) Moving a suggestion above (c) Moving a suggestions below

future. In addition, MyBehavior allows the users to re-sort the suggestions in order of their preference. Users can long-press a suggestion and move the suggestion above or below another suggestion (Figure 5.2(b-c)). For instance, if a user prefers to go to the gym even though s/he did not do it often before, the user can simply move the gym suggestion to the top.

5.2.2 Handling user changes using Pareto-frontier

Finding balance between user changes and low-effort

User customization creates a new ranking that reflects a user's preference. This ranking is an additional ranking to the ranking generated by the Multi-armed Bandit from chapter 4. Any change from the low-effort ranking introduced in the earlier chapter means that some suggestions are more preferred by or motivating to the user which are not lower-effort. FBM suggests that both low-effort and motivation are important for actionability, and it already describes how to balance between low-effort and motivation. Hence we can operationalize FBM to balance between low-effort, as suggested by MAB, and a user's motivation, as suggested by user preference ranking, and find a final ranking of actionable suggestions.

The final ranking or order of actionable suggestions are determined using FBM as follows [52]. In FBM, both motivation and low-effort (i.e. perceived effort level) are important factors in how actionable a suggestion would be. We illustrate what Fogg's behavior model would suggest with an example. Let us assume that there are three suggestions for a user: walking near the office, walking near the home, and going to gym. The user frequently walks near the office and prefers doing this. User also has a high preference for going to the gym, but is not good at gym work and goes infrequently. In addition, the user frequently walks near her house but is not keen on this activity. In this scenario, Fogg's behavior model would suggest that walking near the office is the most actionable. However, choosing between walking near home and going to gym would be a tie since one is easier to do while the other is more preferred.

An algorithm to operationalize FBM

Given the insight on how FBM can balance between low-effort and motivation, in this section we describe how the balancing can be done using an algorithm called Pareto-frontier. We first define the pareto-frontier algorithm [131]. Then we describe how Pareto-frontier operationalize FBM's balancing act as part of its algorithmic steps.

Pareto-frontier (PF) algorithm is a strategy for making decisions when there

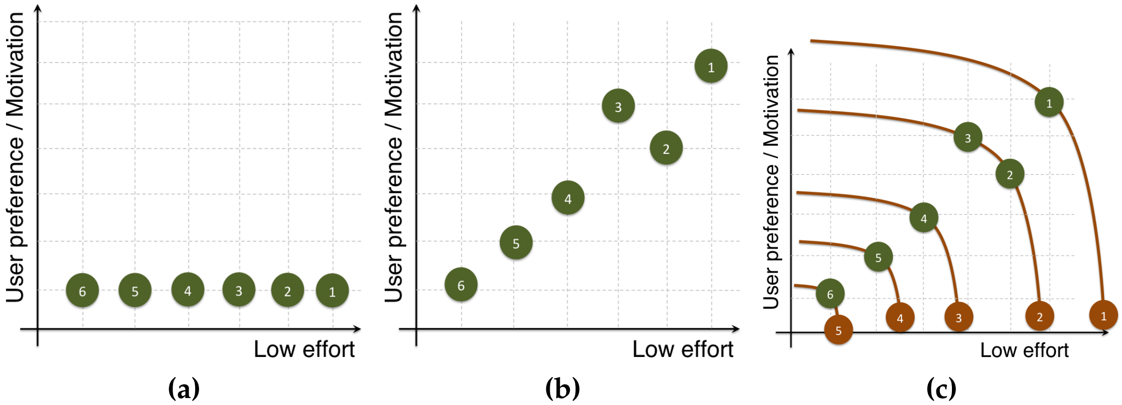


Figure 5.3: An example for pareto-frontier operation (a) before user preference (b) after adding user preference (c) pareto-frontier ranking

are multiple objectives. Specifically, let for input domain $x \in X$, we have objective functions $f_1(x), f_2(x), ..., f_n(x)$ that we have to maximize simultaneously. Now according to PF algorithm, $x_1 \in X$ optimizes these objective functions more than $x_2 \in X$ (also referred as x_1 *pareto-dominates* x_2) if

1. $f_i(x_1) \geq f_i(x_2)$ for all $i = 1, ...n$
2. $f_j(x_1) > f_j(x_2)$ for at least one $j = 1, ...n$

With the definition of pareto-froniter, let us visit an example in Figure 5.3 to show how pareto-frontier would work to operationalize FBM. Figure 5.3 shows the low-effort ranking of 6 suggestions, where the number in green circle 1 to 6 respectively denote the first to sixth ranked suggestions by the MAB. After the user interaction, let us assume a user swapped the order of 2 and 3: i.e., 3 is the second ranked suggestion in user preference ranking, while 2 is the 3rd user preferred rank. The user preserved the ranking of other suggestion. Now, if we consider user preference ranking as motivation then we will get the suggestion positioning in the combined low-effort and motivation space as shown in Figure 5.3b. This low-effort and motivation space is the same space as shown for FBM in Figure 5.1. Here, 1 is the highest ranked or most actionable since its low-effort and motivation level are higher than all. 2 and 3 are a tie since 2 is lower-effort while 3 is more motivating. 4 would be ranked next and higher than 5, since 4's low-effort and motivation levels are higher. Same holds for 5, which would be ranked higher than 6.

Let us if we run the pareto frontier algorithm (PF) on Figure 5.3b, we would get the same result as shown in Figure 5.3c. The dark brown colored lines are the classic representation of "frontiers" from pareto frontier literature. A frontier represents a set of suggestions that pareto-dominate the frontiers below. Suggestions on a pareto-frontier can not pareto-dominate each other. As can be seen from the Figure 5.3c, 2 and 3 are ranked the same in PF which is instructed by FBM. Other ranking by PF also holds similarly instructed by FBM. Therefore, running only the PF on the user-ranking and MAB ranking enacts the theoretical principles of FBM. In other words, MyBehavior can handle the human-in-the-loop in a theoretically grounded way using FBM, and MyBehavior can do the grounding by using a PF algorithm.

The algorithm

More formally PF is used to handle human-in-the-loop in the following way. First we introduce some notations. We denote the set of suggestions as X where an element $x_j \in X$ is a suggestion. For a suggestion x_j, v_j refers to its rank from MAB algorithm whereas p_j refers to its rank after users finishes reordering the suggestions (Figure 5.2(a-c)). Thus a higher rank or value of v_j or p_j means the suggestion is more low effort or more preferred respectively. With this notation, the pareto algorithm works as follows. Let us assume that for two suggestions x_i, x_j, preferences and low-effort ranks are p_i, p_j and v_i, v_j respectively. If x_i's both preference and low-effort ranks are higher than x_j then x_i ranks higher (or is more actionable) than x_j, and x_i *pareto-dominates* x_j. If x_i's preference is higher than x_j while the low-effort rank is lower than x_j (i.e., $p_i > p_j$ and $v_i < v_j$) or the other way around (i.e., $p_i < p_j$ and $v_i > v_j$) then x_i and x_j receive the same rank and the more actionable suggestions can not be decided. Note here, that pareto-frontier makes no assumption about scale of p or v and can still balance between them. Finally, the ranking process works iteratively as shown in Algorithm 1. It starts with a set of all available suggestions X. At every iteration, a set of suggestions X_i are selected that pareto-dominates rest of the suggestions. X_i are then ranked higher than the rest and are removed from the set of X. The process then repeats.

```
input : A set of suggestions $X$ annotated with user preference and
        caloric benefit used in MAB

Initialize an index value $i = 1$;
while $X$ is non-empty do
    - find subset $X_i$ in $X$ that pareto dominates $X - X_i$;
    - rank suggestion(s) in $X_i$ with $i$;
    - increment $i$ by one and remove $X_i$ from $X$;
end
```

Algorithm 1: Ranking suggestion with pareto-frontier

Finally a specific case that needs special attention in the pareto ranking is when a *new* suggestion x arrives with low-effort rank v and unknown preference p since the user never ranked it. In this case, a fair policy is adopted that acts as follows: If x_1 and x_2 are two other suggestions such that $v_1 > v > v_2$ and $p_1 > p_2$ then no matter what the unknown value of x's preference is, x would not be pareto dominated by x_2 since x has a higher low-effort rank than x_2. Since the value of p is unknown, it is fairly assumed that this unknown value to be less than a known value p_2. This would assign x the same rank as x_2 which is lower than x_1.

5.2.3 Effect of incorporating human input

We conducted a 3-week pilot to study with a new and improved version of MyBehavior that incorporated human input. To measure the benefit of incorporating human preference, we showed MyBehavior MAB generated suggestions to the users after the study. They were asked to rate 8 food and 8 activity suggestions between a scale of 1 to 5. This rating represents whether users liked the suggestion and would act on it on an average day (1 = disagree and 5 = agree). After users finished rating the default set of suggestions, they were instructed on the use of the remove and reorder functions to incorporate their preferences. Users on the average changed 3.5 suggestions out of 16 suggestions. When users finished providing their preferences, we ran the pareto-frontier algorithm and then showed users the revised suggestions. We asked the users to rate again. Ratings without incorporating the human preference are similar to results from the MyBehavior deployment from the earlier chapter ($\mu = 3.5, \sigma = 1.2$). However, after incorporating human preference using pareto-frontier algorithm, there is a statistically significant increase of almost 19% ($\mu = 4.2, \sigma = 1.1$).

5.3 Related work in Human-in-the-loop and multiple-objective optimization

In this chapter, three intuitive user interactions are used to incorporate human input/preference, and we subsequently merged the human input with the multi-armed bandit ranking described in the earlier chapter. We have posed the merging as a multiple-objective optimization problem and have used a well-known decision-making algorithm called *pareto-frontier* to find an overall ranking that is also grounded in psychology theory on persuasion. A few related works that are relevant to incorporating human control and optimizing multiple objectives are reviewed.

The decision theory and recommender systems literature on multiple objectives can be divided into 4 categories [8][70][117]. They are described below along with the reasons why they are/aren't used in MyBehavior:

1. **Multiple-attribute utility theory (MAUT) or value focused-methods** : MAUT methods assume that there is a global or overall utility that can be synthesized as a function of the different objectives (i.e., user preference or MAB ranking). Commonly, the global utility is modeled as a linear combination of the objectives [8][85]. For our problem, where we want to measure actionability, MAUT would construct a global actionability value, u, which is a linear combination of user preference, p, and MAB rank v; i.e., $u = ap + bv$, where a and b are some fixed constants. We avoided MAUT for a few reasons. First, estimating the values of a or b either requires data about overall ranking u, where the value of a and b are learnt, or some domain knowledge about relation between u, a, b, where an expert provides the value of a and b. Collecting the overall rankings require extra data from the user. On the other hand, it is difficult to determine the values of a and b even with domain knowledge, because user preference level p for the same suggestion can change over time. e.g., some suggestions are more preferred in winter or some suggestions are not actionable during less stressed episodes. Therefore, the value of a, i.e., the importance of user preference or motivation in "$u = ap + bv$", can change, which makes a

hard to determine. Furthermore, the value of *a* and *b* can vary for different people. All of these reasons combined, we decided to not use MAUT in MyBehavior.

2. **Multiple objective optimization methods**: In this category of techniques, there is no assumption of a global ranking which is a function of the different objective functions. These techniques, often referred as multiple-objective mathematical programming [164], use pareto-optimality to find the final global ranking. In MyBehavior, we use a pareto-frontier method which is a multiple objective optimization method.
3. **Outranking relations models**: Outranking relations models are flexible models that allow for "incomparability" in addition "equally preferred" and "strictly preferred". MAUT and pareto-froniter, on the other hand, do not allow for incomparability. However, the flexibility makes the outranking algorithms more complicated [133], since it introduces a measure of the extent of incomparability. In addition, the user-interactions for incomparability need more effort from the programmer/researcher to code, and also requires more input from the end-user. As a result, we opted for the up-down-remove user interactions, which always rank some suggestions more or less preferred than the other, without introducing a notion of incomparability.
4. **Preference learning models**: Preference learning methods learn preference as a machine learning model. The data from past decisions, often from same or other users, are utilized to learn patterns of preferences over multiple objectives, which are later used to infer future preferences [9][7][134][140]. There are both collaborative filtering methods and

content based methods to learning preference with multiple objectives or ranking [9]. Collaborative filtering techniques assume similarity between users and are often black boxes which are hard to interpret [9]. On the other hand, content based methods require data before the model is deployed [7][134][140]. In MyBehavior, such data are not available ahead of time. Therefore we predefine a strategy with pareto-frontier for final ranking that does not require data ahead of time.

CHAPTER 6

EVALUATION OF MYBEHAVIOR

In earlier chapters, we described several iterations of design and development of MyBehavior. These iterations increased MyBehavior's user experience and ensured that machine generated suggestions are effectively communicated to its users. After the improvements, we deployed MyBehavior to show its efficacy quantitatively. In that vein, we conducted a 14-week deployment and evaluation study. The purpose of the evaluation is two fold: (1) to test whether MyBehavior has better efficacy in promoting better exercise and dietary behavior compared to a control condition, and (2) to assess if MyBehavior can enable positive change beyond the initial novelty period. However, quantitively demonstrating behavior change with traditional randomized experiments is often infeasible for early behavior change application like MyBehavior. This is because a large number of people need to be randomized in groups and use the app for a long time [80]. Therefore we used an alternative experiment design that is appropriate to quantitively demonstrate early efficacy of MyBehavior. In the rest of this section, we detail the study design, statistical analysis methods and report results.

6.1 Study design considerations

Randomized Controlled Trials (RCTs) are the standard way to demonstrate the efficacy of interventions. In RCTs, between-subject designs are followed comparing control and experimental groups. However, for meaningful comparisons between the groups, potential confounds need to be minimized, for example in our case Body Mass Index or proficiency with technology. RCTs typically achieve this by using large numbers of participants in each group. Moreover, RCTs also can span many years to show long term change [80].

RCTs would be ideal, such large-scale studies are hard to achieve for novel systems, partly due to insufficient resources [138]. Furthermore, there is a need to validate the adoption and potential for change before investing the time and resources required for an RCT. As a remedy, alternative methods of evaluation have been proposed in HCI [80] and mobile health [42][114] literature. For instance, Klasanja et al. [80] argue to evaluate new behavior change technology with small-scale pilot studies. These pilot studies can show early effects and give guidelines for future improvements. As we have described, several pilot studies have been done to make design improvements to MyBehavior. Dallery et al. [42] and Onghena et al. [114] argued that small scale within subject trial, sometimes referred as "single case experiment design", are sufficient to evaluate early behavior change technology. Since subjects are compared with themselves, there is no issue of the aforementioned confounds caused by differences among people. Further, sufficient statistical power can be achieve of a large number of repeated samples are available for a single individual. Repeated samples can be collected easily with automated sensing or daily manual logging [42].

In our study, we follow a single case experiment paradigm called *multiple baseline* design [42]. In a multiple baseline design, subjects are initially exposed to the control condition, which is followed by the experiment condition. However, the duration of the control condition before the experiment condition varies for different users. Such a variation is made as a *replication* strategy to show that the desired dependent variable consistently changes in the desired direction after the experiment condition starts. Multiple baseline also suits well to problems where there are learning effects where it is hard to remove In our study, participants are exposed to either 2, 3 or 4 weeks of control condition before using MyBehavior as part of multiple baseline design. We also run the experiment condition for longer (7-9 weeks) than control condition. We do so to investigate MyBehavior's influence beyond initial novelty periods.

6.2 Study procedure and participants

We sent an invitation for participating in MyBehavior's user study through Cornell University's Wellness Center's email list. Interested individuals emailed back an investigator and were requested to fill out a prescreening survey. The survey asked for age, gender, experience in using smartphones etc. We also asked *readiness* to act on healthier behavior as defined by the Transtheoretical model (TTM) [122][1]. We only included participants with (i) sound proficiency in using smartphones (ii) are either in ready or acting stages of TTM since in these stages people are willing or acting towards changing their behaviors [127]. The study investigators met with eligible participants and installed MyBehavior on their phones. In these meetings, we provided basic instructions to use MyBehavior. Participants also entered their gender, weight, height and weekly weight loss goals. Then the Harris-Benedict equation [66] is used to translate weight loss goals to daily calorie intake and expenditure goals.

The day after the face-to-face meeting, participants started the *baseline phase*

[1] TTM defines several stages to readiness: "Precontemplation" represents a stage of not feeling the need to change while in "Ready" stage there is intention to start eating well or doing exercise in near future but not taking actions. "Acting" stage on the other hand represents already taking actions but still need to strengthen commitment, or fight urges to slip. Finally, "Maintaining" stage means a lifestyle with regular health eating and exercise.

Variable	n(%)
Gender	
Male	7(43.7)
Female	9(56.3)
Age	
18 - 29	4(25.0)
30 - 39	6(37.5)
40 - 49	3(18.7)
> 50	3(18.7)
Stage of behavior change before the study	
Ready	7(43.7)
Acting	9(56.3)
Previous experience with self-tracking	
Maintained food diary	13(81.3)
Maintained exercise diary	11(68.7)

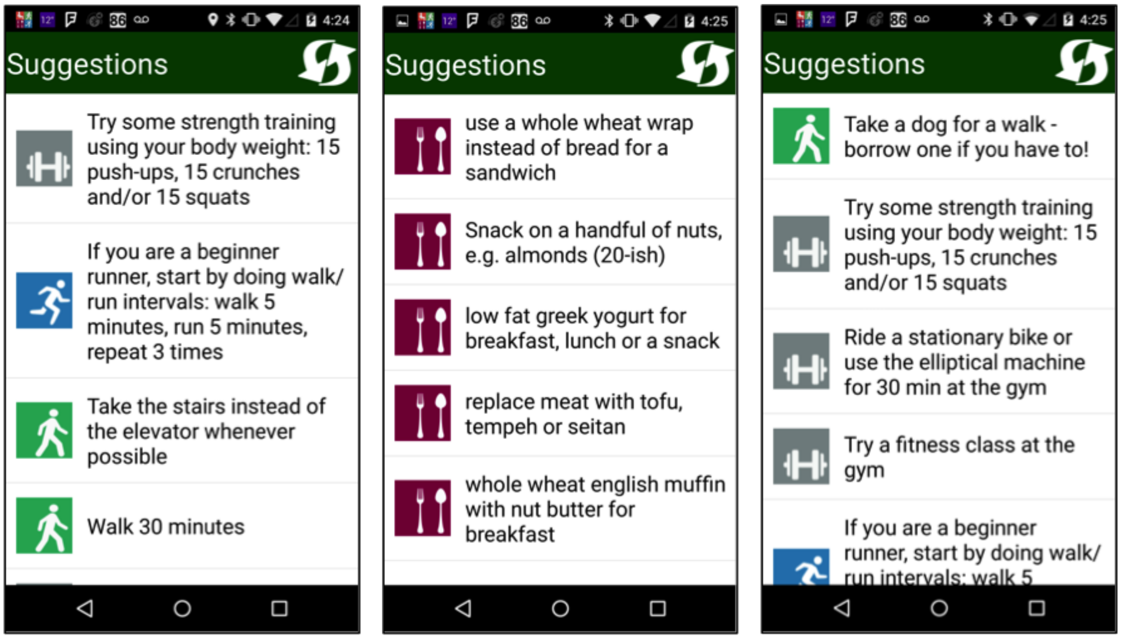
Table 6.1: User demographics in the long term study

of the study. In this phase, calorie goals were displayed in an on-screen widget in the phone's home screen. This widget also incorporated realtime updates of user's daily calorie intake and expenditure. We also added a daily chronological summary of physical activities and food intake. No suggestions were provided in this baseline phase. Note here that such widgets and daily logs are common for many modern health and fitness applications [51][106]. We ran the baseline phase for 3 weeks, since starting to use a health application often makes users more active temporarily even though no intervention is used. Such an effect is often referred to as "novelty effect" [132]. After the baseline phase, participants were exposed to the control condition of the study. Participants received generic prescriptive recommendations generated from a pool of 42 suggestions for healthy living, such as "walk for 30 minutes" and "eat fish for dinner". A certified fitness professional created these generic suggestions after following National Institute of Health resources [109][108]. An external nutrition counselor also reviewed the suggestions to ensure that they were both healthy and achievable. We followed the multiple baseline design as described before and continue the control condition for different durations for different participants. The control condition ranged between 2-4 weeks depending on participants. Each day of the control phase, 8 physical activity and 8 food suggestions were randomly selected from the 42 prescriptive suggestions. These suggestions are shown in a list similar to MyBehavior suggestions and a few screenshots of suggestions during control phase are shown in Figure 6.1. The entire list of 42 suggestions are added as supporting material of this paper. After the control phase, participants received MyBehavior suggestions for 7-9 weeks. Total participation period did not exceed 14 weeks for any participant. Participants were compensated $120 for their regular participation in the study.

We recruited 16 participants. Table 6.1 shows the participant demographics. Our sample size was determined by following the literature of single case experiment design [42]. The literature argues that $n \geq 4$ is sufficient for statistical power if enough repeated samples are collected per participant.

6.3 Outcome measures of the study

We utilize the food and exercise log data to measure changes in food calorie intake and calorie loss in exercise. During the study, we also used an in-phone survey that users filled out daily. The survey asks 5 questions as listed in Table 6.2. For the number of suggestions followed, we use self-report since it is hard to objectively judge whether an activity is done as part of regular actions



Suggestions

Try some strength training using your body weight: 15 push-ups, 15 crunches and/or 15 squats

If you are a beginner runner, start by doing walk/ run intervals: walk 5 minutes, run 5 minutes, repeat 3 times

Take the stairs instead of the elevator whenever possible

Walk 30 minutes

(a)

Suggestions

use a whole wheat wrap instead of bread for a sandwich

Snack on a handful of nuts, e.g. almonds (20-ish)

low fat greek yogurt for breakfast, lunch or a snack

replace meat with tofu, tempeh or seitan

whole wheat english muffin with nut butter for breakfast

(b)

Suggestions

Take a dog for a walk - borrow one if you have to!

Try some strength training using your body weight: 15 push-ups, 15 crunches and/or 15 squats

Ride a stationary bike or use the elliptical machine for 30 min at the gym

Try a fitness class at the gym

If you are a beginner runner, start by doing walk/ run intervals: walk 5

(c)

Figure 6.1: Screenshots of suggestions from the control phase. These suggestions are randomly selected everyday from the pool of 42 suggestions described below. (a) A list of physical activity suggestions on a specific day (b) A list of food suggestions on a specific day (c) A list of physical activity suggestions on another day.

or as a result of the suggestion. We ask how many suggestions users *wanted* to follow to measure user intentions or attitude [13]. Past literature shows that attitudes or intentions often indicate 19%-39% of future behavior [19]. A higher score in the 3rd question means the suggestions relate to a user's life and are potentially easy to implement. We ask the 4th and 5th questions because we want to investigate how MyBehavior suggestions perform against negative life circumstances as barriers and negative emotions have been shown to reduce chances of change [68].

Although weight loss is MyBehavior's main long term goal, calorie loss or user intentions to follow suggestions are important mediators to achieve weight loss. Recent work on adaptive interventions in clinical psychology (e.g., Behavior Intervention Technology [101]) and just-in-time adaptive interventions [107]

Daily phone survey

1. How many suggestions were you able to follow today?

2. How many suggestions did you want to follow?

3. How well did the suggestions relate to your life.
   - likert scale 1-7
   - 1- doesn't relate to your life
   - 7- relates to your life perfectly

4. Did you encounter any barrier to follow the suggestions today (e.g., weather or deadline)?
   - Yes/No

5. Rate your emotional state today
   - photographic affect meter (PAM) scale [121]

Table 6.2: Users answered the above 5 questions in a daily phone survey

argue that calorie loss or positive activities are essential subaims and are valid outcome measures for weight reduction applications.

6.4 Analysis plan

We analyze the efficacy of MyBehavior against control condition by modeling our outcome measures (e.g., caloric loss or number of suggestions followed) as continuous variables using mixed effect models against time. We use mixed effect models [115] since they can handle imbalanced control vs. experiment conditions [36][120] and correlated data points from the same user [44].

$$y_{it} = \beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3 + \beta_{int} x_{it} + b_0 + b_t t + \epsilon_{it} \tag{6.1}$$

In the models, we use the equation 6.1 for mixed effect analysis. y_{it} denotes the outcome variable for subject i at time instance t.We use intercept and time as random effects as b_0 and b_t respectively to allow for inter-subject variations in initial starting points and growths over time [31][103]. Including these random effects significantly increased likelihood over fixed-effect-only models in likelihood ratio tests [146]. Such an increase in model fit (i.e., likelihood) means inter-subject variability exists in our dataset and including random effect is necessary to properly isolate inter-subject variability from actual trends in fixed effects. As fixed effects, we use time and intervention type (i.e., control vs experiment). The co-fixed-effect efficient for intervention is denoted as β_{int} and intervention types are coded 0 for control and 1 for experiment phase. For time, we count time in weeks and denote it as t. The first week of the control phase is coded as 0 and incremented by 1 after each subsequent week. We observed non-linear changes in outcome measures over time, so we use non-linear time effect up to cubic polynomials [146]. β_1,β_2,β_3 are used for the fixed effect co-efficient of t, t^2, t^3 respectively. In general, considering such polynomial time effects shows significant improvements in likelihood ratio tests compared to models without such polynomial time effects. On exception is for number of minutes walked where time or its polynomial forms as fixed effects did not improve the likelihood significantly. This approach of centering [146] time and intervention adjusts for time related effects (e.g., weather effect, or changes due to logging for longer periods) and isolates the change with MyBehavior over control as the co-efficient of intervention fixed effect (β_{int}). In other words, β_{int}s reflect changes (e.g., number of minutes walked more) at the points of introducing MyBehavior. Finally, for the survey response of number of suggestions followed, we additionally include emotional state, barrier and their interaction with intervention types as fixed effects. We add these extra terms to explore interplay between MyBehavior and emotional states/barriers. Emotional states are coded as 0,1,2,3 respectively for negative high, negative low, positive low and positive high. Barriers are coded as 1 for presence of barrier and 0 for absence. Both barriers and emotional states are considered as categorical in the mixed model. The analyses are run using Matlab's statistical analysis toolbox with maximum likelihood.

Given significant intervention effects are achieved with mixed effect models, we explore the real-world end effect of MyBehavior in post-hoc analysis. We compare 2-4 weeks of using control condition to last 3 weeks of using MyBehavior. We consider the last 3 weeks of MyBehavior to measure change beyond initial novelty periods. Specifically, we describe the mean and standard deviations for these two conditions. We then use student t-tests and Cohen-d to measure the statistical significance and effect size. Similar pre-post analysis to measure real world end effect has been done in [31].

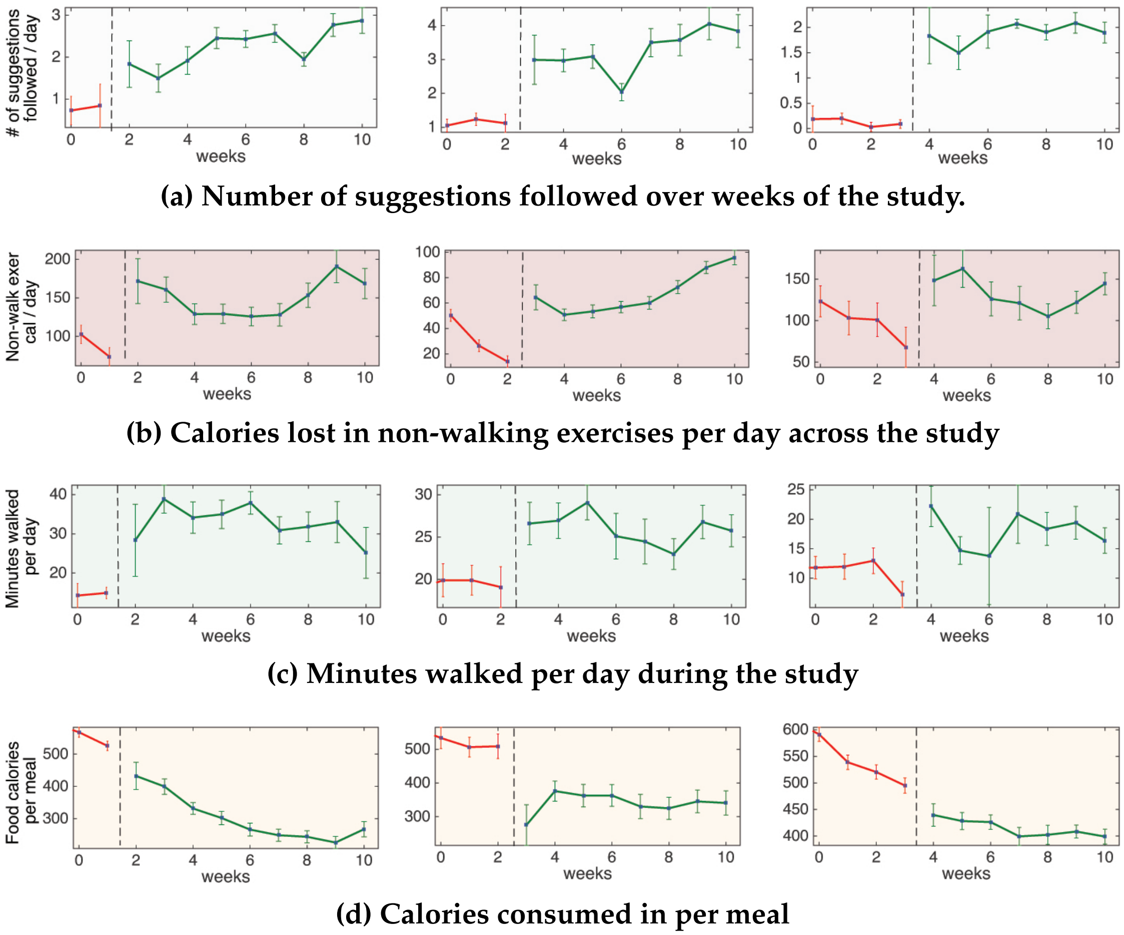
6.5 Results

6.5.1 Comparison with the control condition

Table 6.3 shows the results from the mixed model analyses for different outcome measures. Due to space limitations, we only include the relevant statistics. In 2nd column, we report the coefficient of intervention fixed effect (β_i) and its significance. In third column, we also report the standard model fit statistics that underpin the values of β_is. We include standard model fit statistics namely deviance, AIC and BIC scores [146]. We add significance of the fitted models (LR) against unconditional mean models (i.e., a baseline mixed model with only intercept as both fixed and random effects) [146] using a likelihood ratio test. From table 6.3, we observe that all the fitted mixed models for different outcome measures are significant improvements over the unconditional mean model. Furthermore, use of MyBehavior compared to control condition results in increased number of suggestions followed ($\beta_{int} = 1.2, p < 0.0005$), walking minutes ($\beta_{int} = 10.1, p < 0.005$) and calories burnt in non-walking exercises ($\beta_{int} = 42.1, p < 0.05$) per day. Calorie consumption also decreased per meal ($\beta_{int} = -56.1, p < 0.05$).

Figure 6.2 shows different outcome measures (i.e., number of suggestions followed, minutes walked, calories burnt in exercise, calorie intake in meals) over time as commonly reported in multiple baseline designs [42][24]. All these values are predicted from the mixed models. For each outcome measure, we create three groups representing 2, 3, 4 weeks of using control conditions before exposing to MyBehavior. A dotted line shows the start time of using MyBehavior. Improvements in all outcome measures can be seen to occur in Figure 5 after the introduction of the MyBehavior phase irrespective of the start times. However, patterns over time differ for different outcome measures. Minutes walked did not change much over time. On the other hand, food calories consumption generally decreased over time although introduction of MyBehavior had some effect. Non-walking exercises generally decreased in control over time, but were sustained during MyBehavior usage.

Subjective responses namely number of suggestions participants wanted to follow ($\beta_{int} = 2.9, p < 0.0005$) and relatedness of suggestions to life ($\beta_{int} = 0.5, p < 0.0005$) were also higher for MyBehavior compared to control (Table 6.3). Including emotional state, barriers and their interactions with interventions sig-



(a) Number of suggestions followed over weeks of the study.

(b) Calories lost in non-walking exercises per day across the study

(c) Minutes walked per day during the study

(d) Calories consumed in per meal

Figure 6.2: Changes in user behavior as predicted by the mixed model for multiple baseline design. The dotted lines represent the start of the intervention of MyBehavior. Left, middle, and right figures respectively show results from participants where intervention were started after 2, 3 and 4 weeks of using the control. Red color represents control phase where as green represents periods of using MyBehavior.

nificantly improved likelihood of predicting number of suggestions followed compared to excluding them in likelihood ratio tests ($p = 0.05$). This means that there are significant interactions of MyBehavior vs. control with emotional state and barriers. Figure 6.3 visualizes these interactions as distributions of number of suggestions followed for different emotional states and barrier conditions.

Outcome measure	β_{int}	$-2logL$	AIC	BIC	LR
# of sug. followed	1.2^{***}	2491	2517	2576	***
# of sug. wanted	2.9^{***}	2496	2518	2568	***
relatedness	0.5^{***}	1551	1573	1623	**
walking/day (min)[‡]	10.1^{**}	4795	4809	4839	***
exercise/day (cal)[a]	42.1^{*}	10959	10973	11006	**
each meal (cal)	-56.1^{*}	16151	16165	16200	***

$^{***}p < 0.0005$; $^{**}p < 0.005$; $^{*}p < 0.05$; $\sim p > 0.1$

[a] non-walking exercises combined

[‡] without time as fixed-effect

Table 6.3: Summary of statistical differences between control and MyBehavior as collected from survey, physical activity and dietary logs

6.5.2 Pre-post real-world effect analysis

Pre-post analysis is summarized in Table 6.4. For all the outcome measures, values of Cohen-d indicate medium to large effects of MyBehavior. Although not shown in the table, all these changes are also statistically significant ($p < 0.05$) in student t-tests. An additional result we point to is the changes in number of suggestions followed for barriers and emotional states. Users followed more MyBehavior suggestions where there was no barrier ($p < 0.001, d = 0.84$) such as bad weather. Similar significant increase is also found for positive emotion ($p < 0.001, d = 0.82$).

Furthermore, MyBehavior suggestions were still followed more than control suggestions even when there were barriers ($p < 0.001, d = 0.44$) or when the user experienced negative emotion ($p < 0.001, d = 0.55$). However, effect sizes are smaller for barrier and negative emotions.

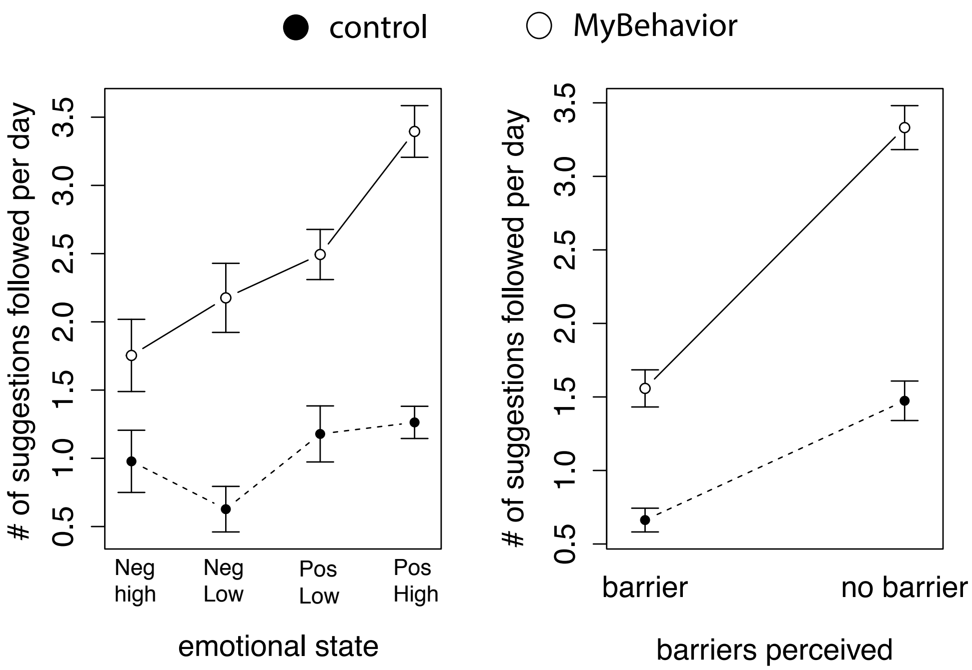


Figure 6.3: Number of suggestions followed for control and experiment conditions with respect to barriers and emotional states

6.6 Discussion of results

In a 14-week study, participants subjectively reported MyBehavior suggestions to be more related to their life and they wanted to follow the suggestions in higher numbers. We believe such higher actionability and relatedness result from MyBehavior's prioritization of low effort suggestions. The higher actionability and relatedness also translated to actual behavior with increased walking, exercise and decreased food calorie intake. These favorable results are replicated as part of multiple baseline design as shown in Figure 6.2. This adoption may result from low-effort suggestions that should enable actual adoption according several behavior change theories [52][76][17][68][13]. Finally, in the pre-post real-world effect analysis, MyBehavior suggestions were followed more during no-barrier or positive emotions states compared to barriers or negative

Outcome measure	Control	MyBehavior	Cohen-*d*
# of sug. followed	1.1 (1.1)	3.1 (2.7)	0.76
# of sug. wanted	2.1 (1.2)	4.4 (2.4)	1.07
relatedness	3.8 (1.1)	4.5 (1.2)	0.54
walking/day (min)	14.5 (5.9)	24.9 (7.4)	1.41
exercise/day (cal)[a]	83.5 (33.1)	126.7 (35.3)	1.23
each meal (cal)	540 (137.2)	362 (134.1)	1.30
# of sug. followed[b]	1.3 (2.2)	3.4 (2.8)	0.84
# of sug. followed[c]	0.6 (2.1)	1.6 (2.5)	0.44
# of sug. followed[d]	1.2 (1.9)	3.2 (2.6)	0.82
# of sug. followed[e]	0.7 (1.5)	1.9 (2.1)	0.55

[a]non-walking exercises combined

[b] for no barrier, [c] with barrier

[d] for positive emotion, [e] for negative emotion

Table 6.4: Pre-post analysis for the control condition and last 3 weeks of experiment condition. Means and standard deviations (within bracket) are shown along with effect size measures.

emotional states. We believe this happens because low effort suggestions similar to MyBehavior are adopted in higher numbers during high motivation states like no-barrier or positive emotions [52]. Nonetheless, some MyBehavior suggestions were followed during barrier or negative emotional states. According to Fogg [52], low-effort suggestions similar to MyBehavior may still stay actionable in low motivation states like with barriers and negative emotions.

CHAPTER 7
DISCUSSION AND CONCLUSION

The last chapter is concluded with a reflection on the MyBehavior work so far. A few guidelines are also discussed for future MyBehavior-alike systems[1]. Specifically, I will divide the description into the following two parts.

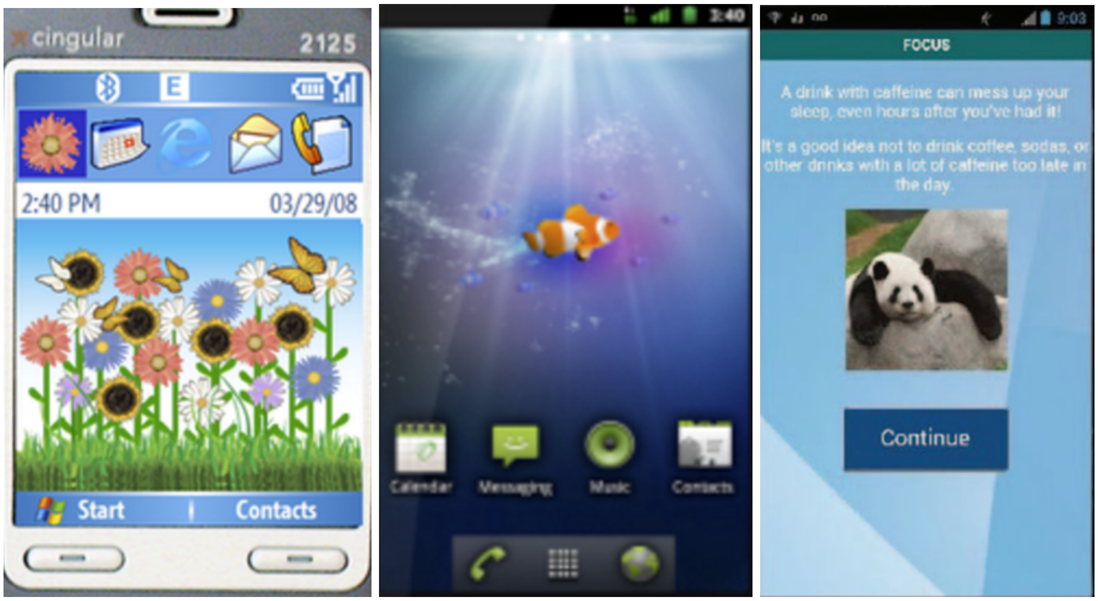
1. I will examine what MyBehavior research implies for the existing research on health interventions. I will argue that there is a historical trend towards tailoring interventions, and MyBehavior follows this trend with an entirely new way to tailor suggestions at a personal and contextual level. Following that I will argue why such personalization is important, and why researchers need to investigate MyBehavior-alike technologies for other health domains.
2. In the second part, I will describe a few common patterns in MyBehavior-alike technologies, and how ideas of MyBehavior can be extended for other health domains.

7.1 MyBehavior, a new way to tailor suggestions

In behavioral intervention design, there is a growing trend to tailor interventions. Considering the fact that patients have differences and their needs change over time, interventions with personalization or adaption functionality can arguably achieve superior results. In the following, I will discuss a few categories of tailoring from past literature, and describe where MyBehavior falls in these

[1] For the rest of this chapter, "MyBehavior-alike systems" would be synonymous to mobile data driven health recommender systems.

categories. Furthermore, I will discuss why the different categories of tailoring may increase intervention efficacy, and where they can have shortcomings. At the end, I will argue that MyBehavior starts a new way of tailoring suggestions, which follows the trend of increasing tailoring. Also, I will argue that there are other domains where similar MyBehavior-alike personalization can be applied. In the next section, I will follow-up this section's discussion and describe the roadmap of how MyBehavior alike personalized suggestions can be created for other domains.



A drink with caffeine can mess up your sleep, even hours after you've had it!

It's a good idea not to drink coffee, sodas, or other drinks with a lot of caffeine too late in the day.

(a) (b) (c)

Figure 7.1: Three examples of single component interventions. (a) and (b) respectively shows ubifit [38] and bewell [89], both of which uses priming. (c) shows a schizophrenia intervention app, called FOCUS, that uses cognitive behavioral therapy [20]

7.1.1 Single component interventions

The first category of interventions only include one fixed component (Figure 7.1). For instance, Ubifit [38] and BeWell [87] used priming to influence a user's physical, activity and social interactions[2]. Many modern mHealth apps deploy other forms of single component interventions, e.g., gamification [99], social support [18] etc. Another common technique for intervention is the manipulation of motivations with positive and negative reinforcements (e.g., operant conditioning [62]). Cognitive behavioral therapy is a common method to break beliefs and barriers for mental illness [61][144][20].

These single-component interventions have two major limitations (i) One single type of intervention almost never works for 100% of the population. Therefore, there is always a fraction of the population that is not reacting positively to the intervention [37] (ii) Changing towards a healthier lifestyle is a dynamic process, and people move through different stages as they begin-to-change, maintain-the-change or relapse-from-the-change [159]. The requirements for different stages are unique. With these two reasons combined, single component interventions can be less effective compared to methods that tailor and adapt.

7.1.2 Multiple-fixed-phase multi-component interventions

Acknowledging the need for adaptation, one way to adapt the treatment can be to break down the process of treatment or behavior change into several phases, and subsequently tailor treatment according to the stages. We refer to such techniques as multiple-fixed-phase interventions. Transtheoritical model (TTM) [72] is a popular and well-used multi-phase intervention (Figure 7.2). TTM breaks down the behavior change process into 4 fixed stages: *pre-*

[2] A **component** is the type of intervention used. e.g., for depression, one component can be different types of antidepressant drugs. Another component can be dosage level of the antidepressant. Components are also often referred as factors in factorial experiment design [115].

contemplation stage, where a patient is not intending to make any change; *contemplation* stage, where a patient is intending to change, but is not making any changes; *action* stage, where a patient is taking actions but needs to strengthen commitment; *maintenance* stage, where a patient has formed habits and regularly maintaining the changes. TTM based interventions showed greater efficacy across different domains of health; e.g., smoking cessation, weight loss etc.; over the non-tailored counterparts.

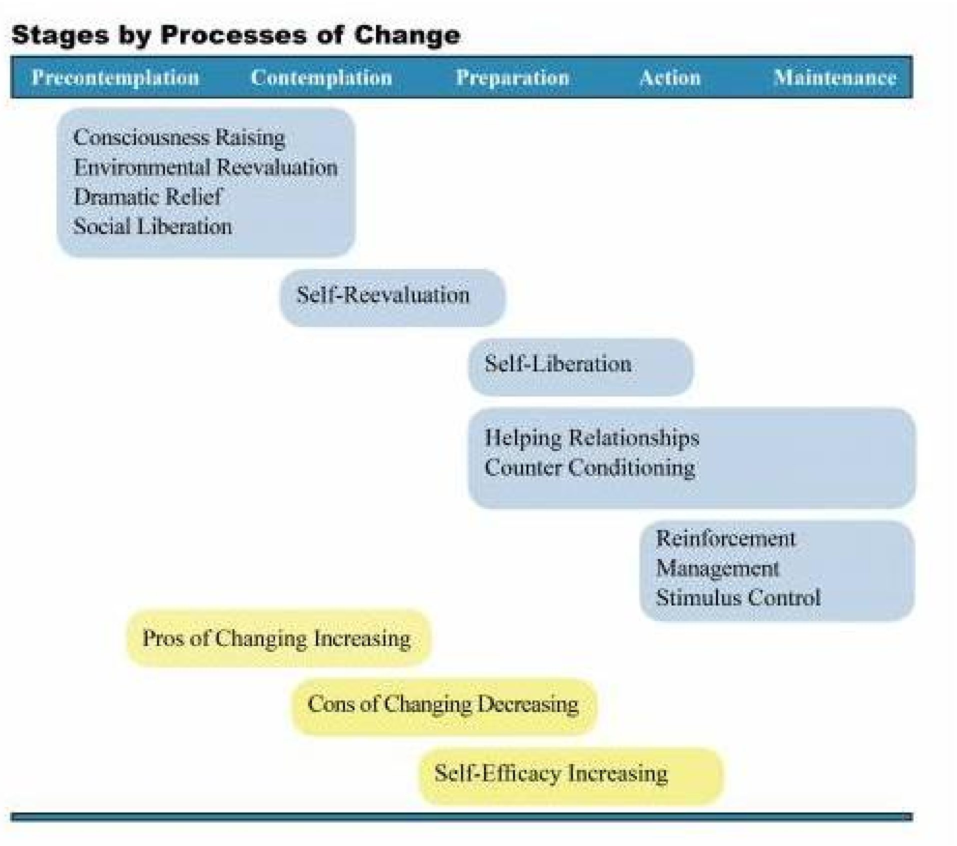
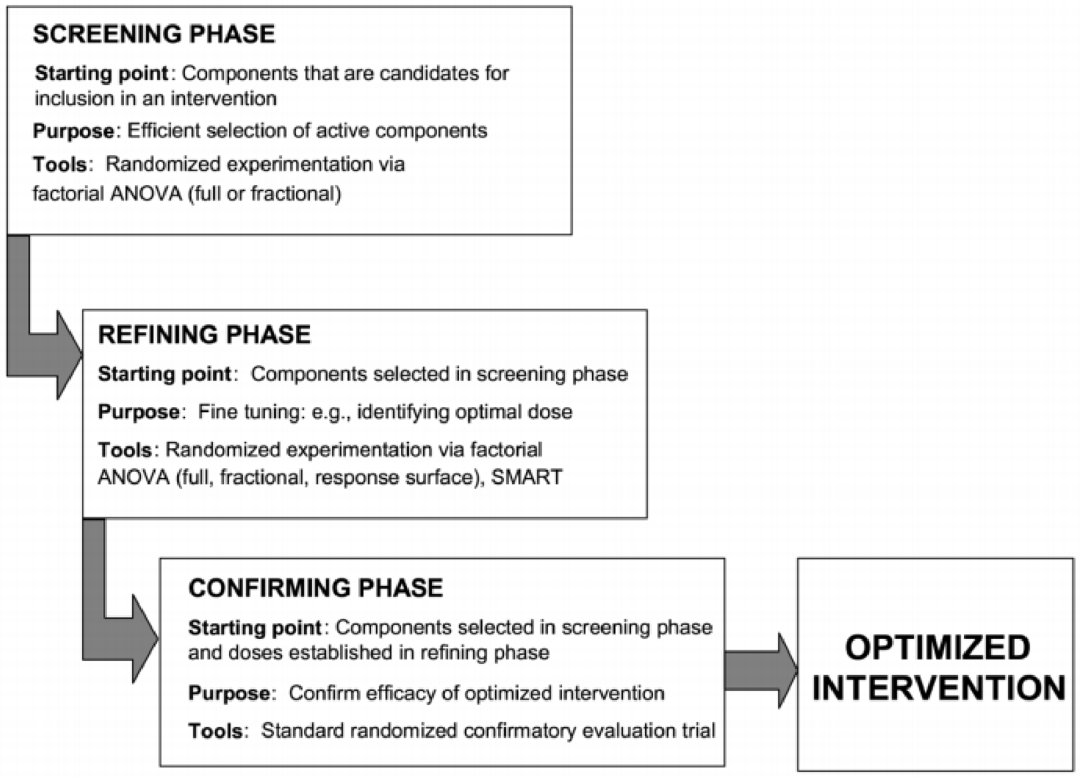


Figure 7.2: Stages of change in trans-theoritical model [159]

In subsequent research, TTM is combined with other components for specific domain, e.g., for smoking cessation, a negatively framed message (e..g, Continuing to smoke will increase your risk of serious health problems) can have a greater effect than a positively framed message (e.g., Quitting smoking will make you feel healthier) when in contemplation stage [37]. A common problem of TTM, however, is the ambiguous boundaries between different stages of change as discussed in section 2.4.



SCREENING PHASE

Starting point: Components that are candidates for inclusion in an intervention

Purpose: Efficient selection of active components

Tools: Randomized experimentation via factorial ANOVA (full or fractional)

REFINING PHASE

Starting point: Components selected in screening phase

Purpose: Fine tuning: e.g., identifying optimal dose

Tools: Randomized experimentation via factorial ANOVA (full, fractional, response surface), SMART

CONFIRMING PHASE

Starting point: Components selected in screening phase and doses established in refining phase

Purpose: Confirm efficacy of optimized intervention

Tools: Standard randomized confirmatory evaluation trial

OPTIMIZED INTERVENTION

Figure 7.3: Phases of MOST

7.1.3 Data driven multi-component tailoring

Several limitations of TTM are overcome with data-driven tailoring approaches, namely Multi-phase Optimization Strategy (MOST) and Sequential Multiple Assignment Randomized Trial (SMART) [37]. MOST considers several treatment components, finds the combinations of components that are known to work better together, and uses the combination for future treatments. e.g., for depression treatment, intervention components may include (1) age and (2) type of therapy: anti-depression medication or cognitive behavioral therapy.

MOST first conducts a screening phase, where it determines what type of component combinations, i.e., antidepressant drug or cognitive behavioral therapy, work best at what age (Figure 7.4). Typically a fully-crossed factorial design is used to find best intervention combinations. When the screening phase is conducted to create initial guesses, a confirmatory phase follows the screening phase, where a randomized control trial is conducted to confirm the efficacy of the best known treatment component combinations established in the screening phase. An outcome of a MOST experiment is a strategy for adapting or tailoring future treatments. For instance, a treatment might suggest to use cognitive behavioral therapy for patients under the age of 30 years.

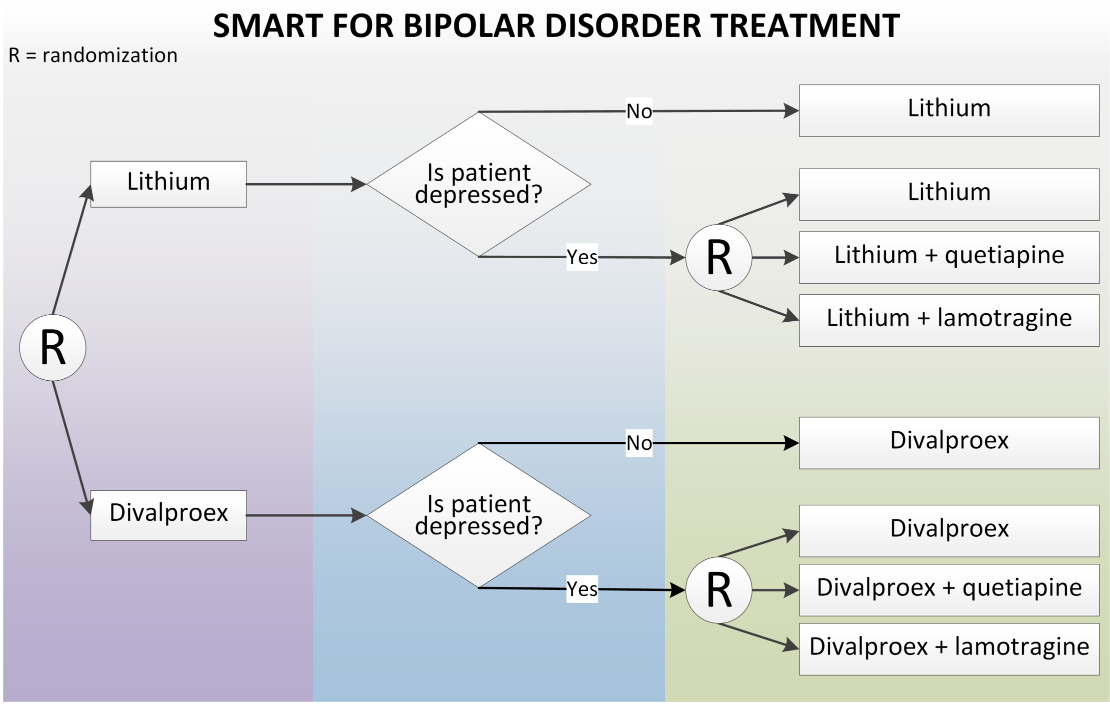


Figure 7.4: SMART for Bi-polar intervention

On the other hand, Sequential Multiple Assignment Randomized Trial (SMART) applies data-driven tailoring similar to MOST, but SMART makes assessment of patient health status at multiple points over time and adapts interventions accordingly[3]. In many cases, behavior change intervention is a process of overtime, and if one intervention component is not effective then a different component is tried. SMART uses reinforcement learning techniques to learn actions or treatments for different assessments made at different times. For example, a negatively framed message can be issued when a patient starts to relapse to smoking. More complicated examples can depend on certain values of assessments/observations, and subsequently tailor the interventions (Figure 7.4).

7.1.4 Personalized components and continuous tailoring

However, there are a few limitations of prior data driven assessment techniques, especially considering the modern technical advances of mobile computing and sensing. First, SMART does data driven adaptation/tailoring, but the adaptation/tailoring happens after fixed intervals, e.g., often after days or weeks[4]. However, tailoring can go beyond the fixed interval measurements, and can be continuously conducted. e.g., a walking suggestion can be given when an individual comes to his/her office, no matter when they arrive at the office. Such continuous measurement and subsequent adaptation of interventions is now feasible, because our phones can understand the contextual changes, and can also deliver the intervention at the right place and time. Secondly, the treatment messages/medications in SMART are a predetermined set, the appropriate messages/medications are adapted to the response of the user. e.g., SMART may have a generic pool of messages namely walk 30 minutes or take small walking breaks. Users see messages, they respond/follow the suggestions, and SMART adapts future suggestions. However, the interventions could be per-

[3] MOST only intervention at the start

[4] There is a modern version of SMART, called Just-in-time-adaptive-intervention (JITAI), where the assessments are made 4 times a day [107][93].

sonalized itself, and can be derived from the life or routine a patient lives in (e.g., MyBehavior). And, if interventions are derived from an individuals behaviors, the treatments become easy to follow, which alleviates the important problem of adherence.

7.2 Ways to generalize and extend MyBehavior

As discussed earlier, MyBehavior shows an entirely new way to create personalized and contextualized intervention, where prior work only used a generic set of suggestions that are same across population. Furthermore, it is straightforward to extend MyBehavior, such that a user is prompted with the personalized suggestions at the right context or moment. e.g., every time a user reaches her office, a reminder can be given to take small walking breaks. We strongly believe such contextualization and personalization can go beyond food and exercise suggestions, and can be extended for other health domains. In addition, such data-driven personalization can be low-effort and meet the needs of users better, as mentioned before.

In the following, I will discuss several recipes and future challenges for developing MyBehavior-alike systems. I will go beyond food and physical activity suggestions, and discuss other health domains where MyBehavior-alike technologies can be useful. In addition, I will use data from MyBehavior studies, and several other data sources to solidify my points.

Recipe #1: Focus on personalizing treatment first

Our first recipe is to personalize recommendations first, and then try to adapt/appropriate the recommendations for different contexts. In other words, the first step should only contain suggestions, and the users should themselves figure out when or where to perform what suggestion[5]. We recommend this recipe for the following reasons:

Adapting to context is hard, especially with less data

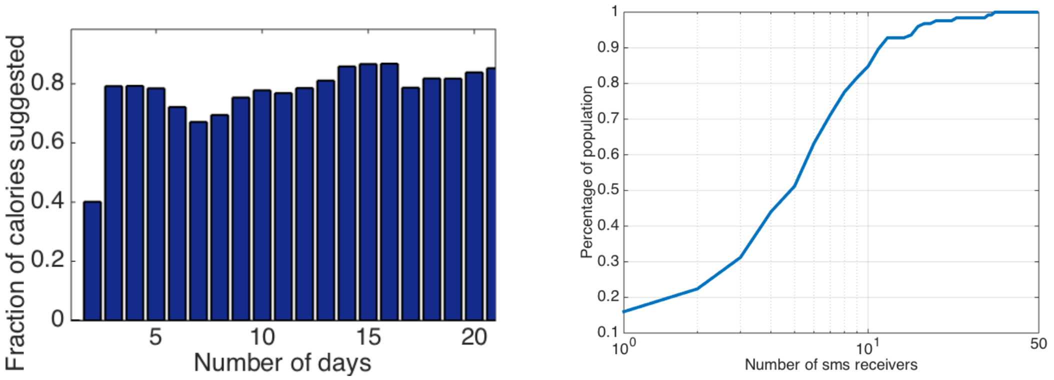
We argue for this proposition with an example. Let us consider the system suggests to go to gym, which is a past behavior from the user. Now people normally want to go to the gym 3-4 times per week, but they may fail to do so regularly because of high stress, negative emotional state or environmental barrier like snow storm. It is quite hard to quantify or adapt gym suggestions for all these contexts, i.e., emotion, stress, or barriers. This is because of two reasons: (i) we need to interrupt the user to get input about contexts, (ii) the amount of different contexts can be many, and they can sometimes differ for different individuals. Therefore, it is hard to adapt suggestions for all these contexts, especially when there is less data at the beginning.

Human behavior is highly predictable, even with less data

Personalized health suggestions are created by relating the suggestions to user behaviors. Fortunately many of the user behaviors are highly predictable. For instance, the places we stay sitting or normally wake are quite repetitive. Furthermore, most of our daily social interactions are also limited to a number of people. Formally, most of our behaviors follow a heavy tail nature. Figure 7.5 shows three cases where we plot the distribution of exercise, and social interaction behaviors. Figure 7.5a shows nearly 80% of calorie burning physical activity behaviors can be predicted in less than 10 days. This means people are highly repetitive, and their behaviors can be predicted within a small amount

[5] This is exactly what MyBehavior does in its current iteration: it just provides some personalized suggestions and user figure out to them in the right contexts

of time. Figure 7.5b shows the case for social interactions, where nearly 75% of the users sent 80% of their SMSs to less than 7 people. Therefore peoples' social behaviors are highly predictable, and a feedback system can promote socialization or social support by taking advantage of this regular nature of our social life.



(a) **(b)**

Figure 7.5: High repetitiveness in human behavior (a) shows the percentage of physical activity behaviors discovered over days and (b) shows how many receivers it took too send 80% SMSs (N = 125) [11]. The fraction of users are on the Y-axis and the number of receivers in the X-axis.

A bandit can easily pick these frequent or repetitive behaviors early as part of exploit suggestions. Furthermore, since human behaviors can change, Bandit in an adversarial setting can detect these changes and adapt to the most recent behaviors.

Recipe #2: Concentrate on contextualized adaptation second

In the earlier recipe, we argued that it is hard to contextually adapt the suggestions. However as more data becomes available, a few suggestions can be contextually adapted without making errors. Specifically, we hypothesize that there are three different levels of contexts that can be relevant for contextualizing health suggestions:

Globally fixed contexts

These types of contexts are not individualized, and are generally true across populations. For instance, during inclement weather, the system can refrain from suggesting outdoor activities. Similarly, snowboarding is an exercise that can be done only in winter. Other less obvious circumstances can be when users have barriers; e.g., negative emotions, depressive symptoms or injuries; users need to be suggested only activities that are not hard. For these types of cases, the system can have predefined models that do not need adaptation across users.

Individually fixed contexts

Some contexts are fixed at an individual level. For instance, some activities are carried out only in the weekend or specific times of week/day. Furthermore, user locations are big determinants of what actions they can perform. For instance, when I moved to SF for a summer internship from Ithaca in 2014, a lot of my activities changed. I could not eat some foods I used to eat, I could not do some exercises I used to do, and I did not talk to some people I used to talk to. Most importantly, some of the location changes at a smaller time scale also affect which suggestions can be carried out. e.g., every time we reach the office, no activity suggestion near our home can be performed. Many of these contexts can be mined/learned with simple heuristics, e.g., suggesting actions that are only near by.

Individually dynamic/changing contexts

These are contexts that are distinct and dynamic to the person/user. For instance, different users have varying needs and they change over time. Furthermore, similar suggestions may work for one user and it may not work for others. Part of these person-specific differences can be learned using an interactive learning technique like reinforcement learning. However, sometimes learning may not be sufficient, because all human behaviors can not be predicted, as humans have free-will on what they want to do. Fortunately, a large amount of human behaviors are repetitive, and therefore predictable. As a result, the hope is most of the context-aware-suggestions would be accurately delivered.

Recipe #3: Ground personalized treatments in theory

It is important that the suggestions or treatments are grounded in theory of behavior change and medical literature. There are at least two reasons behind this proposition: Firstly, the use of theories ensures that we are not reinventing the wheel, and making use of the existing knowledge of health behavior change that are known to work. We discussed this issue at length in Chapter 2.

The second reason to include theory is to rank the suggestions, so that the top ranked suggestions are the most effective to make a desired change. Note this is very important, when there is no data from which some labeled rank can be learned [43][71]. For instance, MyBehavior starts without labeled ranked data, and MyBehavior has to somehow find a way to transform the data to a ranked suggestion list. We got out of this predicament by imposing the "low-effort" proposition from Fogg's behavior model. More precisely, the embedding of theory into the ranking suggestions helped us to crack open the sensor data driven health behavior change space. For similar mobile-data driven recommender systems, theory can guide what criteria should be optimized. e.g., for socialization, the system can optimize for increased social support from strong ties [139] and building a bigger network of people including weak ties [137].

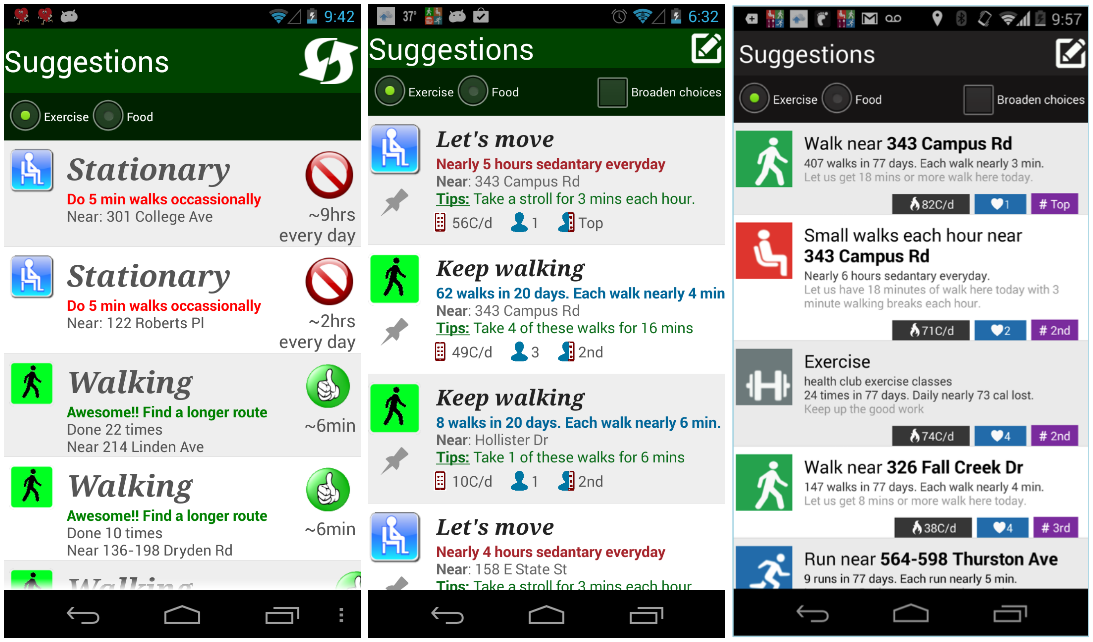
In addition, if we look closely in other domains of ranking, e.g., information retrieval or search engine, similar outside knowledge is imposed when no ranking data is available. For instance, information retrieval used tf-idf [135], and search engines used hubs-authority [81] or page rank [116] to impose importance of information in web-corpus or collection of documents. However, once user data becomes available, more and more subtleties can be studied and learned from data. e.g., many of web-search engines personalize content based on users' clicks or browsing behaviors [33].

Recipe #4: Iterate on design and ideas

It is hard, if not impossible, to make a first version of MyBehavior-alike technology and expect the first version to work perfectly in all situations. There are always scopes for further improvement. There are two types of these improvements: (1) improving the usability of app with iterative design (2) improving on the core ideas of the feedback mechanism itself, and generate knowledge for subsequent research. I explain them in more detail below:

Usability research: Machine learning and behavior change theories alone can not immediately transfer into suggestions that users can understand and follow[6]. To enhance understanding, usability testing and improvements are necessary. These improvements can be done with small scale pilot studies [80]. We have discussed such usability concerns earlier, where there were problems with manual journaling and human-control. In addition, there were multiple iterations of MyBehavior design to effectively communicate the suggestions. Figure 7.6 shows several of such user interface design changes.

[6] Mohr et al. [102] argues that mobile health app should have two aims. (1) clinical aim to improve life (2) usability aim of making the app usable.



9:42

Suggestions

● Exercise ○ Food

Stationary
Do 5 min walks occassionally
Near: 301 College Ave
~9hrs every day

Stationary
Do 5 min walks occassionally
Near: 122 Roberts Pl
~2hrs every day

Walking
Awesome!! Find a longer route
Done 22 times
Near 214 Linden Ave
~6min

Walking
Awesome!! Find a longer route
Done 10 times
Near 136-198 Dryden Rd
~6min

6:32

Suggestions

● Exercise ○ Food □ Broaden choices

Let's move
Nearly 5 hours sedantary everyday
Near: 343 Campus Rd
Tips: Take a stroll for 3 mins each hour.
56C/d 1 Top

Keep walking
62 walks in 20 days. Each walk nearly 4 min
Near: 343 Campus Rd
Tips: Take 4 of these walks for 16 mins
49C/d 3 2nd

Keep walking
8 walks in 20 days. Each walk nearly 6 min.
Near: Hollister Dr
Tips: Take 1 of these walks for 6 mins
10C/d 1 2nd

Let's move
Nearly 4 hours sedantary everyday
Near: 158 E State St
Tips: Take a stroll for 3 mins each hour

9:57

Suggestions

● Exercise ○ Food □ Broaden choices

Walk near **343 Campus Rd**
407 walks in 77 days. Each walk nearly 3 min.
Let us get 18 mins or more walk here today.
82C/d ♥1 # Top

Small walks each hour near **343 Campus Rd**
Nearly 6 hours sedantary everyday.
Let us have 18 minutes of walk here today with 3 minute walking breaks each hour.
71C/d ♥2 # 2nd

Exercise
health club exercise classes
24 times in 77 days. Daily nearly 73 cal lost.
Keep up the good work
74C/d ♥4 # 2nd

Walk near **326 Fall Creek Dr**
147 walks in 77 days. Each walk nearly 4 min.
Let us get 8 mins or more walk here today.
38C/d ♥4 # 3rd

Run near **564-598 Thurston Ave**
9 runs in 77 days. Each run nearly 5 min.

Figure 7.6: Evolution of messaging used in MyBehavior suggestions. Left to right is older to newer.

Iterative improvement of ideas: Mixed methodologists argue that knowledge generation in science is an iterative process that loops between inductive and deductive phases [151]. In the inductive phase, initial ideas are gathered from existing theories and past domain knowledge. Subsequently a few hypotheses are generated. Then an app is made, and the hypotheses are tested. The result of the hypotheses are analyzed in the deductive phase. In addition to results, other observations can be acquired with interviews and log. These data can be analyzed with thematic analysis [28] and secondary analysis [41][73]. Such analysis can reveal what the initial version what was missing and what can be added for the future version. With that we again enter the inductive phase.

For instance, the first version of MyBehavior was created following the low effort theory from the Fogg Behavior Model (FBM) [52]. This first version showed some efficacy, but we found limitations in user control. We combined the finding with the motivational aspect of FBM, and the second version is created. However, users wanted to be notified to perform the suggestions. FBM also mentions that the user needs to be prompted at the right moment of when s/he can act on the suggestion. A future version of MyBehavior is already in works which can prompt the users at the right location and time that is appropriate to follow a suggestion.

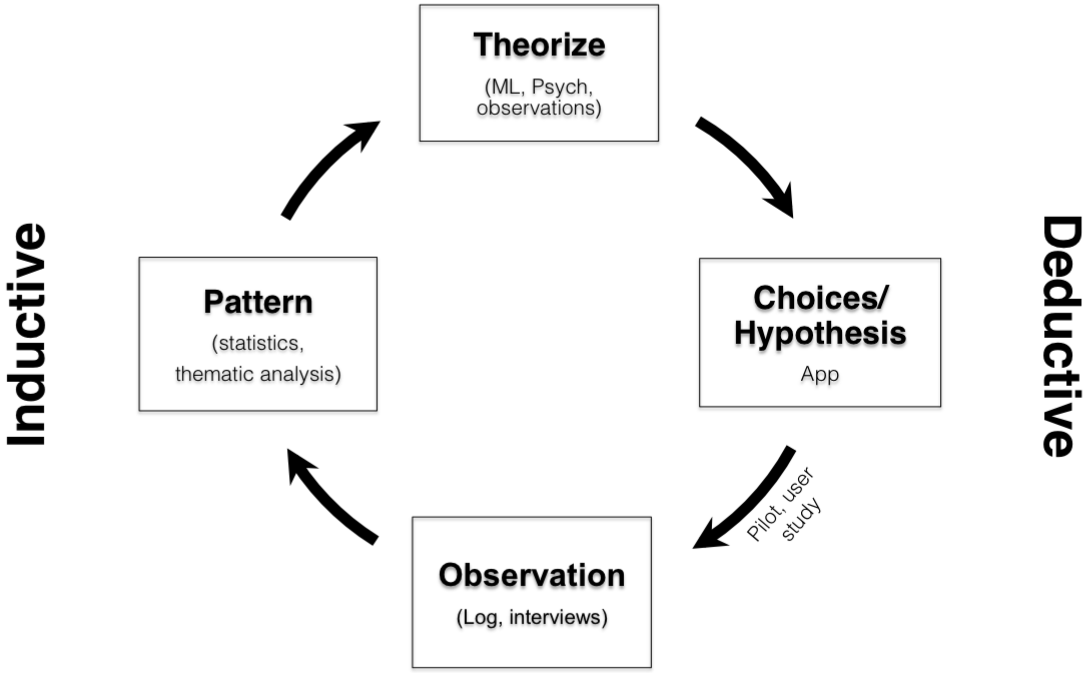


Figure 7.7: An iterative model of ideas and knowledge generation

Recipe #5: Make all the information reusable for later

The system operations in MyBehavior or similar systems can get complicated because of the large amount of codes that get executed in a stage-by-stage manner. e.g., in MyBehavior, the millisecond accelerometer data needs to flow through several stages of processing pipelines to become human-understandable health suggestions. These suggestions can be extended with more functionalities (e.g., on a bad weather day, only indoor suggestions will be given). In that vein, if we want to create a MyBehavior for stress interventions then the system needs to detect stressful episodes either from speech [95] or other sensors [6]. It can then inform the user's friends to talk or SMS to the user for social support.

In order to support such complex interactions, a modular and extensible design with a client-server architecture is best suited. The modular design would help better source code management and re-use of existing codes. The extensible design can help to build further new functionality. Finally, the client-server architecture will allow for maximum decoupling of code [136]; i.e., the clients and servers will run in their own processes and will only exchange necessary data with nearly no shared code. Due to this decoupling, if the client or server malfunction then the errors will not propagate from one code module to the other.

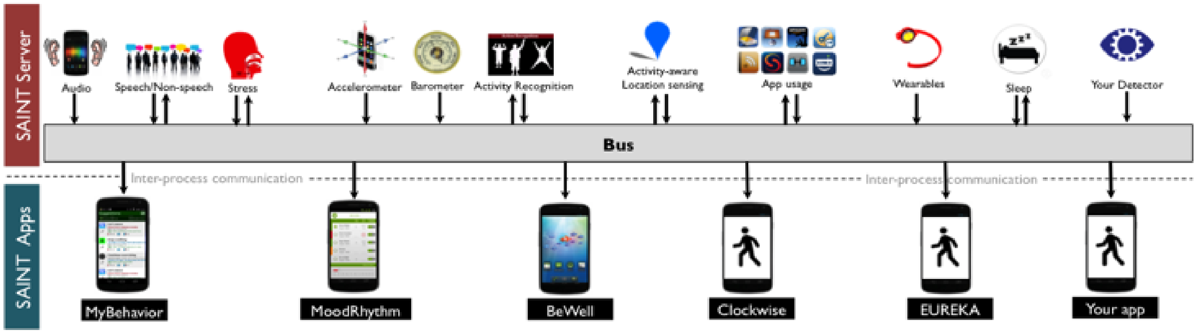


Figure 7.8: The architecture of SAINT sensing and inference framework. SAINT provides a unified bus interface to share data across sensing and inference modules. Also client applications can connect with SAINT to receive SAINTs sensing and inferred data.

In order to tackle such a modular, extensible and client-server architecture, we built a sensing and inference toolkit, called SAINT [125]. Inside SAINT, the sensing and inference codes are separated into distinct modules. These modules can share their data over a unified bus to create newer modules (Figure 7.8). Architecturally the bus system is similar to publisher-subscriber architecture [49]. The bus is implemented in an energy efficient manner with fast circular buffers and inters process communication.

Challenge #1: Hard to quantify rewards

An important question is how we can extend MyBehavior's personalized suggestion for other domains, namely sleep, stress, etc. Technically, what needs to be done is to understand some user behaviors that can be targeted for suggestions in order to improve certain well-being. The behavior to target are typically identified in terms of the reward in the overall payoff, which is the same as overall well-being improvement. However, a favorable thing for physical activity or food was that every time an action is taken (or suggestion is followed), then the reward, i.e., calorie loss or gain, was observed immediately, and these calorie gains/lost accumulated for overall fat loss.

This is not the case for other problems. For instance, a sleep recommender system can suggest to not drink coffee, not to workout late in the night, or have sunshine early in the morning etc [34][3]. There may be other behavioral factors. e.g., for some individuals, if they go to play indoor soccer or some other sports in the afternoon, then they tend to have a better sleep later in the day. However, for some people, some of these suggestions are effective, and for others, the suggestion may not work; therefore, adaptation and personalization are necessary. But, every time coffees are drunk, the immediate effect of drinking coffee can vary for different people, and the effect is not evident unless we wait at the end of the day to see how much they have actually slept. Furthermore, the user may have done other things that may or may not affect sleep.

Fortunately, such an adaptive system is still an online decision-making problem, similar to bandit algorithms. Bandit algorithms are well studied and are already extended for a lot of different circumstances than the standard setting described earlier [29]. One of such bandit is called the “complex action bandits”, where an action consists of a collection of arms, and an aggregate reward is received at the end of day for the action [60]. Complex action bandits use a Bayesian bandit approach to update joint probability distribution of rewards for arms, and subsequently suggest arms that have high marginal rewards [74]. For sleep recommendation, this is similar to giving a few suggestions for sleeping better; e.g., drink less coffee, expose to sunlight earlier in the day; together at the start of the day. At the end of the day, a feedback is received on how long the users slept. From the reward, we can figure out whether the sleep was good or bad, and which behaviors commonly appear for good or bad sleep. Subsequently personalized suggestions would be issued to continue/avoid the behaviors that cause good/bad sleep.

Challenge #2: Going social

MyBehavior is currently based on leveraging a users own behavior. However, people by nature are not solitary and they are influenced by social contexts [45]. Therefore behavioral change can be influenced by the interactions of other people. For instance, the food recommendation could be augmented by foods that nearby friends are consuming. Such local-social information can increase likelihood of adherence. Systems could leverage which food items are more easily accessible if local information is known [52] or a user may be more influenced by food items present in a current social group [35]. Computationally, the inclusion of local-social factors mean that we are moving from a single agent to a multi-agent problem. There is already extensive literature on multi-agent systems and game theory, which could be utilized to build intelligent local-social recommender systems [150].

7.3 Conclusion

Health data acquisition using smartphones is becoming more commonplace with a myriad of health apps. In addition, hardware manufacturers, such as Apple and Google, now support efficient sensing and processing at the hardware level, which is making data collection even more achievable. However, these improvements in measurements or acquisition did not match with health feedback application that utilizes the finer details in the data. In this chapter, we presented an in-depth case study of MyBehavior, which is the first attempt to bridge the gap between mobile data and health recommendation with a deeper analysis of data. MyBehavior provides specific personalized health recommendations from physical activity data, using off-the-shelf reinforcement learning techniques. MyBehavior has also been shown to promote higher levels of physical activity than generic suggestions from health coaches. We have also presented several extensions to the MyBehavior idea in domains of food and chronic back pain. Several takeaways for future MyBehavior alike systems, along with open questions for future explorations, are also discussed. We believe this is just the start, and we envision MyBehavior like systems would be more common as we move into a future, where large amount of personal data are available through mobile sensors, health apps, phone usage traces, and wearables. Similar automated technologies for personalized recommendations, namely Netflix for movies or Google for web search, have already revolutionized the way we consume entertainment and information. MyBehavior and similar technologies can do the same, and can provide personalized health recommendations automatically at scale.

CHAPTER 8

APPENDIX

8.1 List of all generic suggestions

Summary of all the generic suggestions:

Food suggestions

1. Snack on a handful of nuts, e.g. almonds (20ish)
2. eat eggs (hardboiled, poached, scrambled) for any meal or snack
3. oatmeal for breakfast
4. put mustard instead of mayonnaise on sandwich
5. use a whole wheat wrap instead of bread for a sandwich
6. snack on baby carrots or celery sticks with hummus
7. apple with nut butter for a snack
8. eat soup (clear broth not creamy)
9. small piece of lowfat cheese or a string cheese for a snack
10. try sweet potatoes instead of white potatoes
11. snack on salsa and baked tortilla chips
12. put lowfat turkey and lots of veggies on a sandwich
13. wholegrain pretzels for a snack
14. snack on a rice cake with hummus or nut butter
15. whole wheat english muffin with nut butter for breakfast

16. eat fresh fruit (apple, banana, orange, pear, plum, red grapes, kiwi, berries)
17. eat whole fruit instead of drinking fruit juice
18. use skim milk instead of halfandhalf or whole milk in coffee
19. low fat greek yogurt for breakfast, lunch or a snack
20. cottage cheese for a snack or lunch
21. snack on airpopped or light microwave popcorn
22. drink green tea or water instead of soda
23. salad with light dressing for lunch
24. steam or roast your vegetables (broccoli, squash, asparagus, etc)
25. shrimp or other shellfish for dinner
26. skinless chicken breast for dinner
27. fish for dinner
28. lean meat (pork tenderloin, lean beef, etc) for dinner
29. replace meat with tofu, tempeh or seitan
30. eat beans, lentils, or chickpeas instead of meat
31. wholewheat pasta with red sauce and veggies for dinner
32. eat brown rice instead of white

Exercise suggestions

1. Walk 30 minutes
2. Add intervals: walk 5 minutes, run 5 minutes, repeat 3 times

3. Take the stairs instead of the elevator whenever possible
4. Take a dog for a walk
5. Swim a lap, rest for 1 minute, repeat 510 times
6. Try a fitness class at the gym
7. Strength training bodyweight exercises like pushups, tricep dips, squats, lunges, planks
8. Yoga
9. Park at the far end of the parking lot to walk further
10. Yardwork

BIBLIOGRAPHY

[1] Basis b1, nov 2015.

[2] Emotiv epoc, nov 2015.

[3] Saeed Abdullah, Mark Matthews, Elizabeth L Murnane, Geri Gay, and Tanzeem Choudhury. Towards circadian computing: early to bed and early to rise makes some of us unhealthy and sleep deprived. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 673–684. ACM, 2014.

[4] Edward Abrahams and Mike Silver. The history of personalized medicine. *Integrative Neuroscience and Personalized Medicine*, pages 3–16, 2010.

[5] Anonymus AC04472789. *Thermal environmental conditions for human occupancy*. Ashrae, 2004.

[6] Phil Adams, Mashfiqui Rabbi, Tauhidur Rahman, Mark Matthews, Amy Voida, Geri Gay, Tanzeem Choudhury, and Stephen Voida. Towards personal stress informatics: Comparing minimally invasive techniques for measuring daily stress in the wild. In *Proceedings of the 8th International Conference on Pervasive Computing Technologies for Healthcare*, pages 72–79. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2014.

[7] Gediminas Adomavicius and YoungOk Kwon. New recommendation techniques for multicriteria rating systems. *Intelligent Systems, IEEE*, 22(3):48–55, 2007.

[8] Gediminas Adomavicius, Nikos Manouselis, and YoungOk Kwon. Multi-criteria recommender systems. In *Recommender systems handbook*, pages 769–803. Springer, 2011.

[9] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6):734–749, 2005.

[10] Gediminas Adomavicius and Alexander Tuzhilin. Context-aware rec-

ommender systems. In *Recommender systems handbook*, pages 217–253. Springer, 2011.

[11] Nadav Aharony, Wei Pan, Cory Ip, Inas Khayal, and Alex Pentland. Social fmri: Investigating and shaping social mechanisms in the real world. *Pervasive and Mobile Computing*, 7(6):643–659, 2011.

[12] Barbara E Ainsworth, William L Haskell, Stephen D Herrmann, Nathanael Meckes, David R Bassett, Catrine Tudor-Locke, Jennifer L Greer, Jesse Vezina, Melicia C Whitt-Glover, and Arthur S Leon. 2011 compendium of physical activities: a second update of codes and met values. *Medicine and science in sports and exercise*, 43(8):1575–1581, 2011.

[13] Icek Ajzen. Theory of planned behavior. *Handb Theor Soc Psychol Vol One*, 1:438, 2011.

[14] Amazon Mechanical Turk. http://www.mturk.com/, 2013. [Online; accessed 19 March 2013].

[15] Boris Aronov, Sariel Har-peled, Christian Knauer, Yusu Wang, and Carola Wenk. Frchet distance for curves, revisited. In *In ESA*, 2006.

[16] Daniel Ashbrook and Thad Starner. Using gps to learn significant locations and predict movement across multiple users. *Personal and Ubiquitous Computing*, 7(5):275–286, 2003.

[17] Albert Bandura and David C McClelland. Social learning theory. 1977.

[18] Eric PS Baumer, Vera Khovanskaya, Phil Adams, John P Pollak, Stephen Voida, and Geri Gay. Designing for engaging experiences in mobile social-health support systems. *IEEE Pervasive Computing*, (3):32–39, 2013.

[19] Behavioral Intentions. http://chirr.nlm.nih.gov/behavioral-intention.php, 2013. [Online; accessed 26 February 2015].

[20] Dror Ben-Zeev, Christopher J Brenner, Mark Begale, Jennifer Duffecy, David C Mohr, and Kim T Mueser. Feasibility, acceptability, and preliminary efficacy of a smartphone intervention for schizophrenia. *Schizophrenia bulletin*, page sbu033, 2014.

[21] Yoshua Bengio, Ian J. Goodfellow, and Aaron Courville. Deep learning. Book in preparation for MIT Press, 2015.

[22] Alina Beygelzimer, John Langford, Lihong Li, Lev Reyzin, and Robert E Schapire. Contextual bandit algorithms with supervised learning guarantees. *arXiv preprint arXiv:1002.4058*, 2010.

[23] Jeffrey P Bigham, Michael S Bernstein, and Eytan Adar. Human-computer interaction and collective intelligence. 2014.

[24] Anthony Biglan, Dennis Ary, and Alexander C Wagenaar. The value of interrupted time-series experiments for community intervention research. *Prevention Science*, 1(1):31–49, 2000.

[25] Steven Bird. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics, 2006.

[26] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[27] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[28] Virginia Braun and Victoria Clarke. Using thematic analysis in psychology. *Qualitative research in psychology*, 3(2):77–101, 2006.

[29] Sébastien Bubeck and Nicolo Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *arXiv preprint arXiv:1204.5721*, 2012.

[30] Robin Burke. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4):331–370, 2002.

[31] Michelle Nicole Burns, Mark Begale, Jennifer Duffecy, Darren Gergle, Chris J Karr, Emily Giangrande, and David C Mohr. Harnessing context sensing to develop a mobile intervention for depression. *Journal of medical Internet research*, 13(3), 2011.

[32] Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge University Press, 2006.

[33] Olivier Chapelle, Thorsten Joachims, Filip Radlinski, and Yisong Yue. Large-scale validation and analysis of interleaved search evaluation. *ACM Transactions on Information Systems (TOIS)*, 30(1):6, 2012.

[34] Eun Kyoung Choe, Bongshin Lee, Matthew Kay, Wanda Pratt, and Julie A Kientz. Sleeptight: low-burden, self-monitoring technology for capturing and reflecting on sleep behaviors. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 121–132. ACM, 2015.

[35] Robert B Cialdini and Nathalie Garde. *Influence*. A. Michel, 1987.

[36] Avital Cnaan, NM Laird, and Peter Slasor. Tutorial in biostatistics: Using the general linear mixed model to analyse unbalanced repeated measures and longitudinal data. *Stat Med*, 16:2349–2380, 1997.

[37] Linda M Collins, Susan A Murphy, and Victor Strecher. The multiphase optimization strategy (most) and the sequential multiple assignment randomized trial (smart): new methods for more potent ehealth interventions. *American journal of preventive medicine*, 32(5):S112–S118, 2007.

[38] Sunny Consolvo, David W McDonald, Tammy Toscos, Mike Y Chen, Jon Froehlich, Beverly Harrison, Predrag Klasnja, Anthony LaMarca, Louis LeGrand, Ryan Libby, et al. Activity sensing in the wild: a field trial of ubifit garden. In *Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 1797–1806. ACM, 2008.

[39] Felicia Cordeiro, Elizabeth Bales, Erin Cherry, and James Fogarty. Rethinking the mobile food journal: Exploring opportunities for lightweight photo-based capture. *CHI 2015*, 2015.

[40] Mary Ruth Craig, Alan R Kristal, Carrie L Cheney, and Ann L Shattuck. The prevalence and impact of atypicaldays in 4-day food records. *Journal of the American Dietetic Association*, 100(4):421–427, 2000.

[41] Ashley Crossman. Data Sources For Sociological Research.

[42] Jesse Dallery, Rachel N Cassidy, and Bethany R Raiff. Single-case experimental designs to evaluate novel technology-based health interventions. *Journal of medical Internet research*, 15(2), 2013.

[43] Marco De Gemmis, Leo Iaquinta, Pasquale Lops, Cataldo Musto, Fedelucio Narducci, and Giovanni Semeraro. Preference learning in recommender systems. *Preference Learning*, 41, 2009.

[44] Peter Diggle, Patrick Heagerty, Kung-Yee Liang, and Scott Zeger. *Analysis of longitudinal data*. Oxford University Press, 2002.

[45] Paul Dourish. *Where the action is: the foundations of embodied interaction*. MIT press, 2004.

[46] Charles Duhigg. *The power of habit: Why we do what we do in life and business*, volume 34. Random House, 2012.

[47] Thomas Eiter and Heikki Mannila. Computing discrete fréchet distance. Technical Report CD-TR 94/64, TU Vienna, 1994.

[48] Deborah Estrin. Small data, where n = me. *Commun. ACM*, 57(4):32–34, April 2014.

[49] Patrick Th Eugster, Pascal A Felber, Rachid Guerraoui, and Anne-Marie Kermarrec. The many faces of publish/subscribe. *ACM Computing Surveys (CSUR)*, 35(2):114–131, 2003.

[50] Kerry E Evers, James O Prochaska, Janet L Johnson, Leanne M Mauriello, Julie A Padula, and Janice M Prochaska. A randomized clinical trial of a population-and transtheoretical model-based stress-management intervention. *Health Psychology*, 25(4):521, 2006.

[51] Fitbit, Inc. http://www.fitbit.com/, 2013. [Online; accessed 19 March 2013].

[52] BJ Fogg. A behavior model for persuasive design. In *Proceedings of the 4th international Conference on Persuasive Technology*, page 40. ACM, 2009.

[53] BJ Fogg. Baby steps for behavior change, 2015. [Online; accessed 11-November-2015].

[54] US Food, Drug Administration, et al. Paving the way for personalized medicine: Fdas role in a new era of medical product development. *Silver Spring, MD: US Food and Drug Administration*, 2013.

[55] Julien Freudiger, Reza Shokri, and Jean-Pierre Hubaux. Evaluating the privacy risk of location-based services. In *Financial Cryptography and Data Security*, pages 31–46. Springer, 2012.

[56] Benjamin Gardner. Habit as automaticity, not frequency. *European Health Psychologist*, 14(2):32–36, 2012.

[57] Benjamin Gardner, Gert-Jan de Bruijn, and Phillippa Lally. A systematic review and meta-analysis of applications of the self-report habit index to nutrition and physical activity behaviours. *Annals of Behavioral Medicine*, 42(2):174–187, 2011.

[58] Marta C Gonzalez, Cesar A Hidalgo, and Albert-Laszlo Barabasi. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, 2008.

[59] Google. Android sensor hub, nov 2015.

[60] Aditya Gopalan, Shie Mannor, and Yishay Mansour. Thompson sampling for complex online problems. In *Proceedings of The 31st International Conference on Machine Learning*, pages 100–108, 2014.

[61] Eric Granholm, Dror Ben-Zeev, Peter C Link, Kristen R Bradshaw, and Jason L Holden. Mobile assessment and treatment for schizophrenia (mats): a pilot trial of an interactive text-messaging intervention for medication adherence, socialization, and auditory hallucinations. *Schizophrenia bulletin*, page sbr155, 2011.

[62] Peter Gray. *Psychology*. Worth Publishers, 1999.

[63] Andrea Grimes, Vasudhara Kantroo, and Rebecca E Grinter. Let's play!: mobile health games for adults. In *Proceedings of the 12th ACM international conference on Ubiquitous computing*, pages 241–250. ACM, 2010.

[64] William M Grove. Thinking clearly about psychology.

[65] Gordon H Guyatt, Jana L Keller, Roman Jaeschke, David Rosenbloom, Jonathan D Adachi, and Michael T Newhouse. The n-of-1 randomized controlled trial: clinical usefulness: our three-year experience. *Annals of Internal Medicine*, 112(4):293–299, 1990.

[66] JA Harris and FG Benedict. Biometric studies of basal metabolism. *Washington, DC: Carnegie Institution*, 1919.

[67] David Haytowitz, Linda Lemar, Pamela Pehrsson, Jacob Exler, Kristine Patterson, Robin Thomas, Melissa Nickle, Juhi Williams, Bethany Show-

ell, Mona Khan, et al. Usda national nutrient database for standard reference, release 24, 2011.

[68] Godfrey Hochbaum, Irwin Rosenstock, and Stephen Kegels. Health belief model. *United States Public Health Service*, 1952.

[69] Cheng-Kang Hsieh, Hongsuda Tangmunarunkit, Faisal Alquaddoomi, John Jenkins, Jinha Kang, Cameron Ketcham, Brent Longstaff, Joshua Selsky, Betta Dawson, Dallas Swendeman, et al. Lifestreams: A modular sense-making toolset for identifying important patterns from everyday life. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, page 5. ACM, 2013.

[70] Eric Jacquet-Lagreze and Yannis Siskos. Preference disaggregation: 20 years of mcda experience. *European Journal of Operational Research*, 130(2):233–245, 2001.

[71] Thorsten Joachims. Advances in kernel methods. chapter Making Large-scale Support Vector Machine Learning Practical, pages 169–184. MIT Press, Cambridge, MA, USA, 1999.

[72] Sara S Johnson, Andrea L Paiva, Carol O Cummins, Janet L Johnson, Sharon J Dyment, Julie A Wright, James O Prochaska, Janice M Prochaska, and Karen Sherman. Transtheoretical model-based multiple behavior intervention for weight management: effectiveness on a population basis. *Preventive medicine*, 46(3):238–246, 2008.

[73] Melissa P Johnston. Secondary data analysis: a method of which the time has come. *Qualitative and Quantitative Methods in Libraries (QQML)*, 3:619–626, 2014.

[74] Michael Irwin Jordan. *Learning in graphical models*, volume 89. Springer Science & Business Media, 1998.

[75] Dan Jurafsky and James H Martin. *Speech & language processing*. Pearson Education India, 2000.

[76] Daniel Kahneman. *Thinking, fast and slow*. Macmillan, 2011.

[77] Paul B Kantor, Lior Rokach, Francesco Ricci, and Bracha Shapira. *Recommender systems handbook*. Springer, 2011.

[78] Michael Kenteris, Damianos Gavalas, and Daphne Economou. Electronic mobile guides: a survey. *Personal and ubiquitous computing*, 15(1):97–111, 2011.

[79] Sung Chan Kim, Jae Hwan Kim, and Jang Hyun Yoon. Method and system for providing location-based advertisement contents, March 6 2012. US Patent App. 13/413,128.

[80] Predrag Klasnja, Sunny Consolvo, and Wanda Pratt. How to evaluate technologies for health behavior change in hci research. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3063–3072. ACM, 2011.

[81] Jon M Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[82] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[83] John Krumm. Processing sequential sensor data. *Ubiquitous computing fundamentals*, pages 286–319, 2010.

[84] Rita Kukafka. Tailored health communication. *Consumer Health Informatics: Informing Consumers and Improving Health Care*, pages 22–33, 2005.

[85] Kleanthi Lakiotaki, Stelios Tsafarakis, and Nikolaos Matsatsinis. Uta-rec: a recommender system based on multiple criteria analysis. In *Proceedings of the 2008 ACM conference on Recommender systems*, pages 219–226. ACM, 2008.

[86] Phillippa Lally, Cornelia HM Van Jaarsveld, Henry WW Potts, and Jane Wardle. How are habits formed: Modelling habit formation in the real world. *European Journal of Social Psychology*, 40(6):998–1009, 2010.

[87] N.D. Lane, M. Mohammod, M. Lin, X. Yang, H. Lu, S. Ali, A. Doryab, E. Berke, T. Choudhury, and A.T. Campbell. Bewell: A smartphone application to monitor, model and promote wellbeing.

[88] Nicholas D Lane, Emiliano Miluzzo, Hong Lu, Daniel Peebles, Tanzeem Choudhury, and Andrew T Campbell. A survey of mobile phone sensing. *Communications Magazine, IEEE*, 48(9):140–150, 2010.

[89] Nicholas D Lane, Mashfiqui Mohammod, Mu Lin, Xiaochao Yang, Hong Lu, Shahid Ali, Afsaneh Doryab, Ethan Berke, Tanzeem Choudhury, and Andrew T Campbell. Bewell: A smartphone application to monitor, model and promote wellbeing. In *5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth2011)*, 2011.

[90] Richard S Lazarus. Psychological stress and the coping process. 1966.

[91] Thomas C Leonard. Richard h. thaler, cass r. sunstein, nudge: Improving decisions about health, wealth, and happiness. *Constitutional Political Economy*, 19(4):356–360, 2008.

[92] Deborah A Levesque, Deborah F Van Marter, Robert J Schneider, Mark R Bauer, David N Goldberg, James O Prochaska, and Janice M Prochaska. Randomized trial of a computer-tailored intervention for patients with depression. *American Journal of Health Promotion*, 26(2):77–89, 2011.

[93] Peng Liao, Predrag Klasnja, Ambuj Tewari, and Susan A Murphy. Micro-randomized trials in mhealth. *arXiv preprint arXiv:1504.00238*, 2015.

[94] Julia H Littell and Heather Girvin. Stages of change a critique. *Behavior Modification*, 26(2):223–273, 2002.

[95] Hong Lu, Denise Frauendorfer, Mashfiqui Rabbi, Marianne Schmid Mast, Gokul T Chittaranjan, Andrew T Campbell, Daniel Gatica-Perez, and Tanzeem Choudhury. Stresssense: Detecting stress in unconstrained acoustic environments using smartphones. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 351–360. ACM, 2012.

[96] Hong Lu, Jun Yang, Zhigang Liu, Nicholas D Lane, Tanzeem Choudhury, and Andrew T Campbell. The jigsaw continuous sensing engine for mobile phone applications. In *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pages 71–84. ACM, 2010.

[97] editor. Madden GJ. APA Handbook of Behavior Analysis., Washington, DC, Sep 16, 2013.

[98] Melissa Mazmanian, Joanne Yates, and Wanda Orlikowski. Ubiquitous email: Individual experiences and organizational consequences of blackberry use. In *Academy of Management Proceedings*, volume 2006, pages D1–D6. Academy of Management, 2006.

[99] Simon McCallum. Gamification and serious games for personalized health. *Stud Health Technol Inform*, 177:85–96, 2012.

[100] Austin Meyers, Nick Johnston, Vivek Rathod, Anoop Korattikara, Alex Gorban, Nathan Silberman, Sergio Guadarrama, George Papandreou, Jonathan Huang, and Kevin P Murphy. Im2calories: towards an automated mobile vision food diary. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1233–1241, 2015.

[101] C. David Mohr, M. Stephen Schueller, Enid Montague, Nicole Michelle Burns, and Parisa Rashidi. The behavioral intervention technology model: An integrated conceptual and technological framework for ehealth and mhealth interventions. *J Med Internet Res*, 16(6):e146, Jun 2014.

[102] David C Mohr, Michelle Nicole Burns, Stephen M Schueller, Gregory Clarke, and Michael Klinkman. Behavioral intervention technologies: evidence review and recommendations for future research in mental health. *General hospital psychiatry*, 35(4):332–338, 2013.

[103] Leanne G Morrison, Charlie Hargood, Sharon Xiaowen Lin, Laura Dennison, Judith Joseph, Stephanie Hughes, Danius T Michaelides, Derek Johnston, Marie Johnston, Susan Michie, et al. Understanding usage of a hybrid website and smartphone app for weight management: A mixed-methods study. *Journal of medical Internet research*, 16(10), 2014.

[104] Glenn R Morrow. Plato's conception of persuasion. *The Philosophical Review*, pages 234–250, 1953.

[105] Moves App. http://www.moves-app.com/, 2013. [Online; accessed 12 Nov 2015].

[106] MyFitnessPal, LLC. http://www.myfitnesspal.com/, 2013. [Online; accessed 19 March 2014].

[107] Inbal Nahum-Shani, Shawna N Smith, Ambuj Tewari, Katie Witkiewitz, Linda M Collins, Bonnie Spring, and S Murphy. Just in time adaptive interventions (jitais): An organizing framework for ongoing health behavior support. *Methodology Center technical report*, (14-126), 2014.

[108] National Heart, Lung and Blood Institute, National Institutes of Health. Getting started and staying active. http://www.nhlbi.nih.gov/health/public/heart/obesity/lose_wt/calories.htm, 2011. [Online; accessed 19 March 2013.

[109] National Heart, Lung and Blood Institute, National Institutes of Health. Healthy eating plan. http://www.nhlbi.nih.gov/health/public/heart/obesity/lose_wt/calories.htm, 2013. [Online; accessed 19 March 2013].

[110] Yvonnick Noël. Recovering unimodal latent patterns of change by unfolding analysis: Application to smoking cessation. *Psychological Methods*, 4(2):173, 1999.

[111] Jon Noronha, Eric Hysen, Haoqi Zhang, and Krzysztof Z Gajos. Platemate: crowdsourcing nutritional analysis from food photographs. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pages 1–12. ACM, 2011.

[112] Google Now. Google now, nov 2015.

[113] Richard L Oliver. Expectancy theory predictions of salesmen's performance. *Journal of Marketing Research*, pages 243–253, 1974.

[114] Patrick Onghena and Eugene S Edgington. Customization of pain treatments: Single-case design and analysis. *The Clinical journal of pain*, 21(1):56–68, 2005.

[115] R Lyman Ott and Micheal T Longnecker. *An Introduction to Statistical Methods and Data analysis, 4th*. New York: Duxbury Press, 1993.

[116] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: bringing order to the web. 1999.

[117] Panos M Pardalos, Yannis Siskos, and Constantin Zopounidis. *Advances in multicriteria analysis*, volume 5. Springer Science & Business Media, 2013.

[118] Shwetak N Patel, Julie A Kientz, Gillian R Hayes, Sooraj Bhat, and Gregory D Abowd. Farther than you may think: An empirical investigation of the proximity of users to their mobile phones. In *UbiComp 2006: Ubiquitous Computing*, pages 123–140. Springer, 2006.

[119] Christine A Pellegrini, Sara A Hoffman, Linda M Collins, and Bonnie Spring. Optimization of remotely delivered intensive lifestyle treatment for obesity using the multiphase optimization strategy: Opt-in study protocol. *Contemporary clinical trials*, 38(2):251–259, 2014.

[120] J Pinheiro and D Bates. Nlme: Software for mixed-effects models, 2000.

[121] John P Pollak, Phil Adams, and Geri Gay. Pam: a photographic affect meter for frequent, in situ measurement of affect. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 725–734. ACM, 2011.

[122] James O Prochaska and Wayne F Velicer. The transtheoretical model of health behavior change. *American journal of health promotion*, 12(1):38–48, 1997.

[123] Mashfiqui Rabbi and Syed Ishtiaque Ahmed. Sensing stress network for social coping. In *Proceedings of the companion publication of the 17th ACM conference on Computer supported cooperative work & social computing*, pages 225–228. ACM, 2014.

[124] Mashfiqui Rabbi, Shahid Ali, Tanzeem Choudhury, and Ethan Berke. Passive and in-situ assessment of mental and physical well-being using mobile sensors. In *Proc. 13th ACM Intl Conf. Ubiquitous Computing*, pages 385–394, 2011.

[125] Mashfiqui Rabbi, Thiago Caetano, Jean Costa, Saeed Abdullah, Mi Zhang, and Tanzeem Choudhury. Saint: A scalable sensing and inference toolkit. 2015.

[126] Mashfiqui Rabbi, Jean Costa, Fabian Okeke, Max Schachere, Mi Zhang, and Tanzeem Choudhury. An intelligent crowd-worker selection approach for reliable content labeling of food images. In *Proceedings of the Conference on Wireless Health*, WH '15, pages 9:1–9:8, New York, NY, USA, 2015. ACM.

[127] Mashfiqui Rabbi, Angela Pfammatter, Mi Zhang, Bonnie Spring, and Tanzeem Choudhury. Automated personalized feedback for physical activity and dietary behavior change with mobile phones: A randomized controlled trial on adults. *JMIR mHealth uHealth*, 3(2):e42, May 2015.

[128] Francesco Ricci. Mobile recommender systems. *Information Technology & Tourism*, 12(3):205–231, 2010.

[129] George Ritzer. *Sociological theory*. Tata McGraw-Hill Education, 2008.

[130] Herbert Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

[131] MARY CATHERINE Roberts, AVERY ST Dizier, and JOSHUA Vaughan. Multiobjective optimization: Portfolio optimization based on goal programming methods.

[132] Everett M Rogers. *Diffusion of innovations*. Simon and Schuster, 2010.

[133] Bernard Roy. The outranking approach and the foundations of electre methods. *Theory and decision*, 31(1):49–73, 1991.

[134] Nachiketa Sahoo, Ramayya Krishnan, George Duncan, and James P Callan. Collaborative filtering with multi-component rating for recommender systems. In *Proceedings of the sixteenth workshop on information technologies and systems*, 2006.

[135] Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[136] Jerome H Saltzer and M Frans Kaashoek. *Principles of computer system design: an introduction*. Morgan Kaufmann, 2009.

[137] Gillian M Sandstrom and Elizabeth W Dunn. Social interactions and well-being the surprising power of weak ties. *Personality and Social Psychology Bulletin*, page 0146167214529799, 2014.

[138] Robert William Sanson-Fisher, Billie Bonevski, Lawrence W Green, and Cate DEste. Limitations of the randomized controlled trial in evaluating population-based health interventions. *American journal of preventive medicine*, 33(2):155–161, 2007.

[139] Robert M Sapolsky. *Why zebras don't get ulcers: The acclaimed guide to stress, stress-related diseases, and coping-now revised and updated*. Macmillan, 2004.

[140] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Application of dimensionality reduction in recommender system-a case study. Technical report, DTIC Document, 2000.

[141] Paul A Scuffham, Jane Nikles, Geoffrey K Mitchell, Michael J Yelland, Norma Vine, Christopher J Poulos, Peter I Pillans, Guy Bashford, Chris Del Mar, Philip J Schluter, et al. Using n-of-1 trials to improve patient

management and save costs. *Journal of general internal medicine*, 25(9):906–913, 2010.

[142] Rahul C. Shah, Chieh-yih Wan, Hong Lu, and Lama Nachman. Classifying the mode of transportation on mobile phones using gis information. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '14, pages 225–229, New York, NY, USA, 2014. ACM.

[143] Haichen Shen, Aruna Balasubramanian, Anthony LaMarca, and David Wetherall. Enhancing mobile apps to use sensor hubs without programmer effort. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '15, pages 227–238, New York, NY, USA, 2015. ACM.

[144] Sally A Shumaker, Judith K Ockene, and Kristin A Riekert. *The handbook of health behavior change*. Springer Publishing Company, 2009.

[145] Katie Siek, Kay H Connelly, Yvonne Rogers, Paul Rohwer, Desiree Lambert, Janet L Welch, et al. When do we eat? an evaluation of food items input into an electronic food monitoring application. In *Pervasive Health Conference and Workshops, 2006*, pages 1–10. IEEE, 2006.

[146] Judith D Singer and John B Willett. *Applied longitudinal data analysis: Modeling change and event occurrence*. Oxford university press, 2003.

[147] R Sriraghavendra, K Karthik, and Chiranjib Bhattacharyya. Fréchet distance based approach for searching online handwritten documents. In *Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on*, volume 1, pages 461–465. IEEE, 2007.

[148] Lucy A Suchman. *Plans and situated actions: the problem of human-machine communication*. Cambridge university press, 1987.

[149] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 1998.

[150] Ming Tan. Multi-agent reinforcement learning: Independent vs. cooperative agents. In *Proceedings of the tenth international conference on machine learning*, pages 330–337, 1993.

[151] Abbas Tashakkori and Charles Teddlie. *Mixed methodology: Combining*

qualitative and quantitative approaches, volume 46. SAGE Publications, Incorporated, 1998.

[152] Edison Thomaz, Aman Parnami, Irfan Essa, and Gregory D Abowd. Feasibility of identifying eating moments from first-person images leveraging human computation. In *Proceedings of the 4th International SenseCam & Pervasive Imaging Conference*, pages 26–33. ACM, 2013.

[153] Edison Thomaz, Cheng Zhang, Irfan Essa, and Gregory D Abowd. Inferring meal eating activities in real world settings from ambient sounds: A feasibility study. In *Proceedings of the 20th International Conference on Intelligent User Interfaces*, pages 427–431. ACM, 2015.

[154] Wayne F Velicer, Colleen A Redding, Xiaowu Sun, and James O Prochaska. Demographic variables, smoking variables, and outcome across five studies. *Health Psychology*, 26(3):278, 2007.

[155] Rui Wang, Fanglin Chen, Zhenyu Chen, Tianxing Li, Gabriella Harari, Stefanie Tignor, Xia Zhou, Dror Ben-Zeev, and Andrew T Campbell. Studentlife: assessing mental health, academic performance and behavioral trends of college students using smartphones. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 3–14. ACM, 2014.

[156] Robert West. Time for a change: putting the transtheoretical (stages of change) model to rest. *Addiction*, 100(8):1036–1039, 2005.

[157] Wikipedia. Apple motion coprocessors — wikipedia, the free encyclopedia, 2015. [Online; accessed 11-January-2016].

[158] Wikipedia. Persuasion — wikipedia, the free encyclopedia, 2015. [Online; accessed 11-November-2015].

[159] Wikipedia. Transtheoretical model — wikipedia, the free encyclopedia, 2016. [Online; accessed 10-January-2016].

[160] Danny Wyatt, Tanzeem Choudhury, and Jeff Bilmes. Conversation detection and speaker segmentation in privacy-sensitive situated speech data. In *Proc. of Interspeech*, 2007.

[161] CH Yang, JP Maher, and DE Conroy. Implementation of behavior change

techniques in mobile applications for physical activity. *American journal of preventive medicine*, 48(4):452, 2015.

[162] Longqi Yang, Yin Cui, Fan Zhang, John P Pollak, Serge Belongie, and Deborah Estrin. Plateclick: Bootstrapping food preferences through an adaptive visual interface. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 183–192. ACM, 2015.

[163] Tae-Jung Yun, Hee Young Jeong, Tanisha D Hill, Burt Lesnick, Randall Brown, Gregory D Abowd, and Rosa I Arriaga. Using sms to provide continuous assessment and improve health outcomes for children with asthma. In *Proceedings of the 2nd ACM SIGHIT International Health Informatics Symposium*, pages 621–630. ACM, 2012.

[164] Milan Zeleny. *Linear multiobjective programming*, volume 95. Springer Science & Business Media, 2012.

[165] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. Birch: an efficient data clustering method for very large databases. In *ACM SIGMOD Record*, volume 25, pages 103–114. ACM, 1996.

[166] Changqing Zhou, Dan Frankowski, Pamela Ludford, Shashi Shekhar, and Loren Terveen. Discovering personal gazetteers: An interactive clustering approach. In *Proceedings of the 12th Annual ACM International Workshop on Geographic Information Systems*, GIS '04, pages 266–273, New York, NY, USA, 2004. ACM.