Detecting Receptivity for mHealth Interventions in the Natural Environment

Varun Mishra[1,*], Florian Künzler[2,*], Jan-Niklas Kramer[3], Elgar Fleisch[2], Tobias Kowatsch[3] and David Kotz[1]

[1]Dartmouth College, [2]ETH Zürich, [3]University of St. Gallen

[*]Equal Contributions

Abstract

Just-In-Time Adaptive Intervention (JITAI) is an emerging technique with great potential to support health behavior by providing the right type and amount of support at the right time. A crucial aspect of JITAIs is properly timing the delivery of interventions, to ensure that a user is receptive and ready to process and use the support provided. Some prior works have explored the association of context and some user-specific traits on receptivity, and have built post-study machine-learning models to detect receptivity. For effective intervention delivery, however, a JITAI system needs to make in-the-moment decisions about a user's receptivity. To this end, we conducted a study in which we deployed machine-learning models to detect receptivity in the natural environment, i.e., in free-living conditions.

We leveraged prior work regarding receptivity to JITAIs and deployed a chatbot-based digital coach – Walkie – that provided physical-activity interventions and motivated participants to achieve their step goals. The Walkie app included two types of machine-learning model that used contextual information about a person to predict when a person is receptive: a *static model* that was built before the study started and remained constant for all participants and an *adaptive model* that continuously learned the receptivity of individual participants and updated itself as the study progressed. For comparison, we included a *control model* that sent intervention messages at random times. The app randomly selected a delivery model for each intervention message. We observed that the machine-learning models led up to a 40% improvement in receptivity as compared to the control model. Further, we evaluated the temporal dynamics of the different models and observed that receptivity to messages from the adaptive model increased over the course of the study.

1 Introduction

The ubiquituous presence of mobile technologies has enabled a wide array of research into mobile health (mHealth), from sensing health conditions to providing behavior-change interventions. In the past, ubiquitous technologies like smartphones and wearables have shown promise in detecting stress, anxiety, mood, depression, personality change, addictive behavior, physical activity and a host of other conditions [15, 28, 44, 43]. Furthermore, several studies have demonstrated the potential of smartphone-based digital interventions to affect positive behavior change for a range of conditions like smoking, alcohol disorder, eating disorders, and physical inactivity [39, 12, 16, 22]. The eventual goal in mHealth is to be able to combine the two components of accurate sensing and effective interventions to improve the quality of life amongst people suffering from various conditions.

JITAI is a novel intervention design that aims to deliver the right type and amount of support, at the right time, while adapting as-needed to the users' internal and external contextual change [31, 30]. Several studies have employed JITAI-like interventions to affect behavior change for physical inactivity [18, 8], alcohol use [13], mental illness [4], smoking [37], and obesity [3]. For JITAIs to be effective they must deliver the intervention at "the right time," notably, when a person enters a state of *vulnerability*, i.e., a period of heightened susceptibility for a negative health outcome [31]. For example, an intervention to alleviate stress levels should be delivered when a person is stressed or about to be stressed; delivering it when a person is not actually stressed would not be useful. Furthermore, it is also important to deliver interventions at a time when the participant is in a state of *receptivity*, i.e., a period when the participant is able to receive, process, and use the intervention provided [31]. For example, an intervention targeted at reducing sedentary behavior would be effective when a person is "available" to act on it; delivering it when a person is driving is not only sub-optimal but may also be dangerous.

Although prior mHealth research focuses on detecting states of vulnerability or on developing effective intervention mechanisms, little research has been done in identifying *states-of-receptivity*. Künzler et al. developed a smartphone app to deliver physical-activity interventions and explored how the passively collected contextual factors associated with receptivity in a study with 189 participants [21]. The authors also built machine-learning models to detect receptivity and claim to achieve a 77% improvement in F1-score over a biased random model. Choi et al. conducted a 3-week study with 31 participants in which the authors collected self-reports about the participants' context and cognitive/physical state to understand association with the response to relevant Just-In-Time (JIT) support targeted towards sedentary behavior [6]. Sarker et al. conducted a study with 30 participants to explore discriminative features and built machine-learning models to detect receptivity to Just-In-Time Intervention (JITI) with a reported 77.9% accuracy. The authors, however, did not deliver interventions; instead they used Ecological Momentary Assessment (EMA) and claimed that interactions with EMA prompts would be similar to interactions with intervention prompts.

All of those prior works, however, focused on data collection followed by post-study analysis and evaluation of post-study machine-learning models, with the expectation that the models would perform similarly when deployed in real-life conditions. In this paper, we go beyond post-study analysis: we deployed two different machine-learning models to predict in-the-moment receptivity, and used that prediction (in the moment!) to decide when to deliver the intervention. We deployed these models in a physical-activity app used by 83 participants in free-living conditions over a period of 3 weeks. Our goal was to evaluate whether such models actually helped increase receptivity to interventions.

Given the promising results reported by Künzler et al., and the fact that they were the only researchers to use passively collected context information to model receptivity towards actual mHealth interventions, we decided to build upon their work [21]. The authors graciously shared with us their iOS study app, intervention rules, and data. We used their data to build two different machine-learning models, which we later deployed in our field study: (a) a *static* model that remained constant for all participants through out the study, and (b) an *adaptive* model that continuously learnt the receptivity of individual participants from their enrollment in the study and updated the model as the study progressed; we delayed activation of this model until the participant had been in the study for 7 days, however, to ensure that enough data was collected for that participant before using the model's predictions. To compare the utility of these models, we also included (c) a *control* model that would send the intervention messages at a random time. We extended the app (Ally), previously built by Künzler et al., to incorporate the different models and enable *in-the-moment* detection of receptivity.[1] We also translated all of their intervention messages from German to English as our study was conducted in an English-speaking population.

We explore three core research questions (RQs):

- *RQ-1: On a population level, i.e., across intervention messages and across all users, does delivering interventions at a machine learning (ML)-detected time lead to higher receptivity than delivering interventions at a random time?*

  For all the messages sent in the study (across all participants), we evaluated how receptivity differed between messages delivered at 'opportune' moments, i.e., using the static and adaptive models, as compared to random times.

- *RQ-2: How does receptivity of individual users change when interventions are delivered through ML-based intervention timing vs. at random times?*

  For individual participants who received intervention messages at both (a) model-based times and (b) at random times, we evaluated how their individual receptivity changed with the different models.

- *RQ-3: How do the different models for predicting receptivity perform over time?*

  We evaluated how the participants' daily receptivity changed as the study progressed. Our goal was to understand whether the models performed consistently throughout the study or whether their effectiveness improved (or worsened) as the study progressed.

While exploring the three research questions, we make the following **contributions**:

- We built two separate models: (a) static and (b) adaptive, using previously collected data to model receptivity to chat-based JITAIs. We deployed both the models in a 3-week long field study with 83 participants, and

[1] During the first 7 days, the app randomly chose between the *control* and *static* models. After 7 days, the app randomly chose between the *control*, *static*, and *adaptive* models.

actually used the output of the model predictions to deliver JITAIs *in-the-moment*. Our work moves beyond post-study evaluations, and tests the effectiveness of deploying receptivity-detection models trained using data from a previous study. Further, this is the first work to *deploy* an adaptive model to detect receptivity to JITAI and observe how the model performance changes as the study progresses.

- We evaluated the performance of the two models with a *control* model (which sent intervention messages at random times), both on a population level and a participant level. We observed that the *static* model led to significantly higher receptivity than the *control* model in most instances, for both population level and participant level analyses, suggesting that it is possible to use machine-learning models to predict in-the-moment receptivity. We observed that a participant was more likely to respond to a message delivered at a predicted *opportune* moment than at a random moment.
- We evaluated the temporal dynamics of the different models to understand how they performed over the course of the study. We observed that while receptivity from the random model decreased over time, the static model was able to maintain consistent receptivity over time, and receptivity to messages delivered by the adaptive model improved as participants progressed in the study.

We begin with a review of related work in receptivity towards interventions, as well as closely related concepts of *interruptibility* and *engagement*. We then provide a background of the Ally app and operationalization of the different receptivity metrics, followed by our study design methodology and the results for our research questions. Finally, we conclude with a discussion of the implications, future work, and limitations of our work.

2 Related work

In the domain of ubiquitous computing, researchers have extensively explored a related concept of *interruptibility*. In the context of smartphone notifications, interruptibility is defined as a person's ability to be interrupted by an incoming notification by taking an action to open or view the notification content [25]. This stream of research has focused on the use of push notifications to attract users' attention, while making the recipients feel less 'interrupted' by the notifications. In this section, we discuss the prior research on interruptibility to smartphone notifications. We also discuss prior work in capturing participant *receptivity* in mHealth research.

Most prior 'interruptibility' research has focused on analyzing and understanding user interruptibility and on the association between various contextual factors and interruptibility. Researchers have studied factors such as time of day and day of week [34, 36, 2, 23], location [38, 35, 25], Bluetooth information (as a proxy for social context) [34], call and SMS logs (another proxy for social context) [36, 11], Wi-Fi connectivity [34, 36], and phone battery information [35]. While most of these factors were found to be a predictor of users' interruptibility, some studies have also contradicted those findings. For example, some have found time to be a significant predictor [34, 36], while others have found the opposite [23, 45]. The same applies for location, where Sarker et al. and others [36, 34, 38] found location was an important indicator of interruptibility and Mehrotra et al. found otherwise [25].

Other research investigated personality traits and mental state as potential predictors for receptivity. Happy and energetic participants showed a higher availability to interruptions, compared to stressed study participants [38]. Other researchers have shown the significance of personality traits to predict the response delay; in particular, neuroticism and extroversion were found to be significant [26].

Many studies have found a significant correlation between physical activity and interruptibility [33, 38, 14, 25]. The type of physical activity was also found to be significant; for example, people driving in a vehicle replied more slowly than people walking outside [38]. Further, 'breakpoints' in physical activity, e.g., from walking to standing, were found to be favorable times to trigger notifications [33]. Generalizing results, however, is difficult as small sample size and homogeneity of study participants is a common problem in this research area [20].

A few researchers have explored the concept of *engagement*. In the context of smartphone notifications, engagement usually follows after a person is interrupted and refers to the involvement of an user in a task or app that attracts and holds the user's attention [35]. Pielot et al. conducted a study ($n > 330$) in which they delivered eight different types of content and observed participants' engagement and responsiveness [35]. They then built predictive models to classify whether a participant would engage with the notification content, and found that their models led to an improvement of more than 66% over a baseline classifier. Related to engagement, Dingler et al. built an app aimed at improving users' foreign-language vocabulary through notifications and app usage through out the day [9]. The authors found that

several contextual factors relating to phone usage, e.g., number of phone unlocks in the last 5 minutes, time since last unlock, and number of notifications in the last 5 minutes, showed significant correlations to predict users' engagement with the content.

Some works have even deployed a machine-learning classifier to infer interruptibility. Okoshi et al. deployed a breakpoint-detection system to time notifications [32]. Based on a study with over 680,000 users, they found that response time to notifications was up to 49% lower if they were delivered during activity breakpoints. Pielot et al. deployed a model that allowed them to deliver entertaining content when the participant was likely bored; at such times, they found participants were more likely to engage [36].

In the domain of JITAIs and mHealth interventions, *receptivity* is defined as the person's ability to receive, process, and use the support (intervention) provided [31]. To compare with interruptibility and engagement, as highlighted by Künzler et al., "receptivity may be (loosely) conceptualized to encompass the combination of interruptibility (willingness to receive an intervention), engagement (receive the intervention), and the person's subjective perception of the intervention provided (process and use the intervention)" [21].

There have been a growing number of works exploring receptivity and interruptibility in the domain of mHealth. Sarker et al. conducted a study with 30 participants, and identified various contextual and physiological features for their machine-learning model to detect receptivity to EMA prompts [38]. The authors drew a parallel between EMA and interventions by claiming that interaction with self-report or EMA prompts and interaction with interventions would be similar. The authors also provided monetary incentives for EMA completion, however, which may have influenced the participants' receptivity. Further, Mishra et al. investigated contextual breakpoints and how they could be used to detect receptivity to EMA [27].

Choi et al. conducted a 3-week study with 31 participants in which the participants reported information about their context and cognitive/physical state; the paper explores the association of context and cognitive state with the relevant JIT support targeted towards sedentary behavior [6]. The authors identified several key factors relating to receptivity and showed that receptivity to JIT interventions is nuanced and context-dependant.

Künzler et al. developed a smartphone app to deliver physical-activity interventions and deployed it in a study with 189 participants [21]. They explored how passively collected contextual factors, like location, physical activity, time of day, type of day, phone interaction, and phone battery level, are associated with receptivity to interventions. The authors also explored the relationship between receptivity and participant-specific characteristics, like age, gender, personality and device type, and receptivity. They further built machine-learning models to detect receptivity and claim to achieve a 77% improvement in F1-score over a biased random model.

Morrison et al. focused on a mobile stress-management intervention [29]. Their system randomly assigned study participants to one of three groups. Each group was using a different method for receiving push notifications. One group received the notifications occasionally (not daily), one group received them daily, and one group used a machine-learning model to receive 'intelligent' notifications. The authors used time, location labels, and accelerometer features and trained personalized Naive Bayes models for each participant based on the data collected during an initial "learning period." The models, however, were trained based on whether a participant clicked on the notification, regardless if the participant started (engaged with) an intervention. As defined by Künzler et al., receptivity occurs when participants start interacting with the intervention [21].

In our work, we build upon the prior research by deploying machine-learning models to detect receptivity in real-life situations. We used the data collected by Künzler et al. in their study to build and deploy two machine-learning models to detect receptivity to physical-activity interventions. We deployed a static model that stayed unchanged through out the study, and an adaptive model that re-trained itself over the course of the study as participants interacted with the notifications. Our work moves beyond "post-study" model-evaluations, and evaluates the effectiveness of deploying receptivity-detection models trained using data from a previous study. Further, this is the first work to *deploy* an adaptive model to detect receptivity to JITAI and observe how the model performance changes as the study progresses.

3 Background

In this section, we discuss the app used and dataset collected from the study by Künzler et al. [21], followed by the operationalization of *receptivity*, and finally discuss how receptivity fits within the JITAI framework.

3.1 The Ally Study

In an effort to understand receptivity to JITAIs, Künzler et al. developed a mobile JITAI system to promote physical activity. Their "Ally" app – based on the open-source MobileCoach framework – was a chat-based digital coach (for Android and iOS phones) that delivered an actual behavior-change intervention aimed at increasing the participant's daily step count [17, 10]. The authors conducted the study with 189 participants in Switzerland, over a period of 6 weeks. Participants received notifications that encouraged them to engage in conversation with the digital coach, which was a German-speaking chatbot motivating participants to increase their physical activity as measured by daily step count. The study protocol was similar to that used by Kramer et al. [18].

The authors reported interesting findings about the association between receptivity and participant-specific traits (like age, gender, device type and personality) and contextual factors (like battery level, device interaction, physical activity, location, and time of day). The authors also built several machine-learning models to infer different aspects of receptivity, and reported a 77% increase in F1-score for detecting *just-in-time* receptivity, over a biased random classifier (with a combination of participant specific traits and contextual factors), and a 50% improvement in F1-score with contextual factors alone [21].

Given the promising results in their post-study analyses, we decided to build upon their work by deploying in-the-moment receptivity-detection models to evaluate how these models perform in real-world situations.

Another reason to build upon the work by Künzler et al. was the assumption the authors made about *receptivity*. In their intervention design, all conversations started with a generic greeting message like "Hello [participantName]" or "Good morning [participantName]". Only when the participant responded to the greeting message, did the coach start sending the actual intervention messages. Hence, when the authors refer to receptivity to a message, they refer to receptivity to the *initiating* message or "start-of-conversation" message. Given that the Ally participants responded to initiating messages without looking at the actual intervention, we believe it might be possible to build models using the data collected in the Ally study and deploy it in a different study following a similar "start-of-conversation" message strategy.

3.2 Receptivity within JITAI

As proposed by Nahum-Shani et al., JITAI has six key elements: a distal outcome, proximal outcomes, decision points, intervention options, tailoring variables, and decision rules [31]. *Distal outcome* is the ultimate goal the intervention is intended to achieve, *proximal outcomes* are short-term goals the intervention aims to achieve, *decision points* are points in time at which an intervention decision must be made, *tailoring variables* are information concerning the individual that is used to decide when (i.e., under what conditions) to provide an intervention and which intervention to provide, *intervention options* are an array of possible treatments/actions that might be employed at any given decision point, and *decision rules* link the tailoring variables and intervention options in a systematic way.

Within this model, *receptivity* can be considered as a *tailoring variable*, one that indicates whether the user is available to receive an intervention at any given time. At a *decision point*, the *decision rules* can check the receptivity tailoring variable to choose from the various *intervention options*, which could include postponing the intervention or delivering no interventions at that decision point. Hence, receptivity as a tailoring variable can help determine if, what, and when an intervention should be delivered.

3.3 Operationalizing Receptivity

Before discussing our methodology, it is important to establish precise metrics about what the models are trying to achieve. Since we build on the work by Künzler et al., we adopt their definition of receptivity:

- *Just-in-time response*: If a user views and responds to the initiating message within 10 minutes[2] of receiving the prompt, then the user is said to be in a receptive state and it counts as a 'just-in-time response'.
- *Response*: If the user responds to the initiating message at any time, even after the first 10 minutes, it counts as a 'response'.

[2]We chose a 10-minute window to remain consistent with the work by Künzler et al., who used a 10-minute window to define their receptivity metrics [21]. Further, prior work by Mehrotra et al. found that smartphone users accept (i.e., view the content of) over 60% of phone notifications within 10 minutes of delivery, after which the notifications are left unhandled for a long time. They concluded that the maximum time a user should take to handle a notification arriving in an interruptible moment is 10 minutes [25]. Since ours is one of the early works to deploy receptivity detection models, we decided to follow this evidence and also use a 10-minute window.

- *Response delay*: the time (in seconds) elapsed between receipt of the initiating message and the user's first reply to it.
- *Conversation engagement*: If the user replies to more than one message in a 10-minute window following the initiating message, it counts as 'conversation engagement'.

In some contexts we aggregate these metrics over a period of time, e.g., over one day or over the duration of the study. For a given period of time, the *just-in-time response rate* is the fraction of initiating messages for which there was a *just-in-time response*, the *overall response rate* is the fraction of initiating messages for which there was a *response*, the *conversation rate* is the fraction of initiating messages that counted as *conversation engagement*, and the *average response delay* is the mean *response delay* [21].

4 Approach

To enable us to explore our research questions, the authors of the original Ally app graciously shared their app, server backend, intervention messages, and data from their study [21]. In this section, we discuss how we modified their app to fit our research goals, the models deployed in the app, and the study methodology.

4.1 The Walkie app

We modified the iOS version of the Ally app to create a new app we call *Walkie*. Similar to Ally, Walkie is a chat-based digital coach aimed at increasing daily step count. We show a screenshot of the app in Figure 1. The app calculated a personalized step goal each day, based on the participants' 60^{th} percentile step count in the previous 9 days. The intervention components were chat-based conversational messages that were delivered by the digital coach and the participants had to choose from a set of pre-defined responses. The coach initiated the starting message of each conversation to each user at random times within certain time periods. In our study, we used three of the four intervention components used in the original Ally study [21, 18]: (1) The goal-setting prompt set the step goals for the day; it was delivered between 8 and 10 a.m daily. (2) The self-monitoring prompt informed participants about their progress and aimed to motivate them to complete their step goals; it was delivered at a random time between 10 a.m. and 6 p.m. to a randomly selected 50% of participants each day. (3) The goal-achievement prompt informed the participants whether they achieved their goal for the day and aimed to motivate them to complete future goals; it was delivered at 9 p.m. daily.

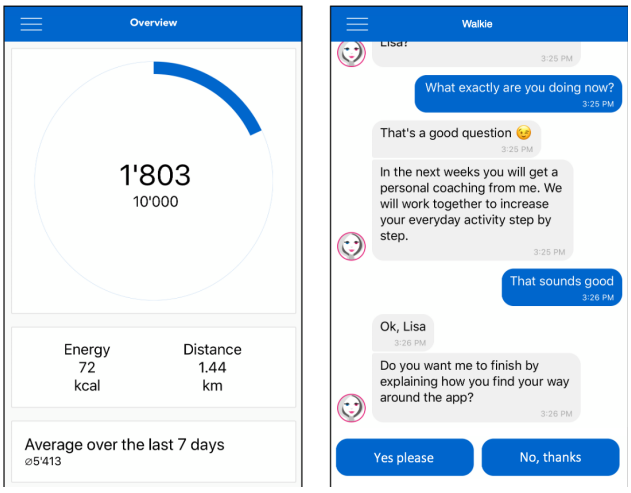


Figure 1: Two screenshots showing Walkie's dashboard (left) and the chat screen for the interventions (right).

Further, Walkie had a context-based receptivity module that continuously tracked several contextual features; Walkie used this module to time the delivery of notifications, as follows. For each day, for each participant, the server randomly chose three times (one in each of the three time blocks) to send a *silent* push notification to that participant's app. When

Walkie received the silent push from the server, it triggered the receptivity module to determine when to deliver that notification to the participant. During the first seven days, the receptivity module randomly selected either the control or static model, with equal weight. On the eighth day and after, the receptivity module randomly selected one of three models, with equal weight. (The seven-day 'warm-up' period allowed accumulation of participant-specific receptivity data before enabling the adaptive model.) For each initiating message received, the app recorded which model was used to time its delivery – control, static or adaptive. We detail the three models in Section 4.2.

Walkie then delivered the notification about the initiation prompt if and only if the selected model inferred the user would be receptive at the current time. The control module always agreed. The static and adaptive models used their classifier to determine whether the current moment is 'receptive'. If the models did not find the current moment to be *receptive*,[3] the app would try again by asking the same model every 5 minutes. If after 30 minutes the model never inferred an opportune moment, Walkie delivered the notification on the 31^{st} minute; in this case, it recorded the delivery mechanism as "control", since the notification was delivered at a random time, and not at an opportune moment.[4]

We used the Walkie app to conduct a within subjects study with three experimental conditions for delivering the interventions: *control*, *static*, and *adaptive*. It is important to note that the intervention delivery conditions did not affect the actual content of the interventions delivered by the app.

Regardless of the chosen delivery model, the participant's response to any initiating message provided new data for use by the adaptive model. There were three cases: (a) just-in-time response: the contextual state at the time of notification delivery was added with label 'receptive'; (b) later response: the contextual state at the time of notification delivery was added with label 'non-receptive', and the contextual state at the moment of response was added with label 'receptive' (since the participant was in a state-of-receptivity when they responded); (c) no response: the contextual state at the time of notification delivery was added with label 'non-receptive'. Whenever the adaptive model was selected as the delivery model, it first re-trained its model using any new data points added.

We diagram the system design in Figure 2.

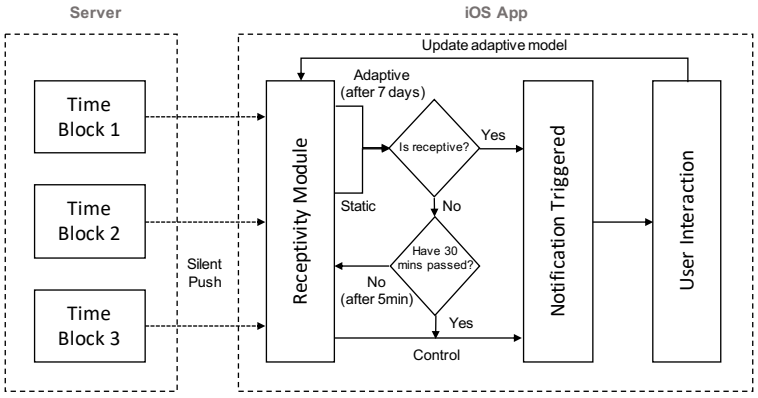


Figure 2: System design of the Walkie app.

4.2 Building the Detection Models

In their work, Künzler et al. built machine-learning models to estimate different aspects of receptivity: just-in-time response, conversation engagement, and response delay [21]. The authors shared with us the data they collected from iOS users in their study, enabling us to build new models that we deployed in Walkie. We first discuss the features we selected to use in our models, followed by a description of the models themselves.

[3] A receptive moment is one at which the model predicts that the participant will respond to the initiating message within the next 10 minutes.

[4] It can be argued that a notification delivered at the 31^{st} minute might not be exactly like 'control', since a machine-learning model has already passed it as non-receptive. We argue, however, that since the 'control' model delivers notifications at random times, and since 31 minutes later than a random time is still a random time, it can be assumed that the 'control' model delivered the notification. The goal of this work is to distinguish between moments identified as receptive by the machine-learning models and random moments. We argue all moments are either receptive or random, and the machine-learning models simply identify receptive moments from a series of random moments.

4.2.1 Choice of Features

In their work, Künzler et al. considered a variety of contextual features, like physical activity, device interaction, location type, type of day, time of day, and phone battery status [21]. Although most of those features can be calculated in real-time on an iOS device, one particular feature, *location type* (like home, work or transit), is more complicated. To accurately compute the location type, algorithms need several weeks of location data to meaningfully cluster and derive categories of location (such as 'home' or 'work'). In our 2-3 week study there was insufficient data to derive the location-type feature; furthermore, Künzler et al. found that (for Ally) iOS users' location type showed no significant associations with any receptivity metric [21]. We thus decided not to include location type as a feature in our models.

A list of features used in our models is shown in Table 1.

Table 1: List of features used in our models.

Category	*Features*	*Type*
Date/Time	Type of day	Categorical (weekday/weekend)
	Time of day	Categorical (morning, afternoon, evening)
Phone Battery	Battery Status	Categorical (charging, discharging, full)
	Battery Level	Numerical (1%-100%)
Device Interaction	Lock State	Categorical (locked, unlocked)
	Lock change time	Numerical (in seconds)
	Wi-Fi connection	Categorical (connected, disconnected)
Activity	Physical Activity	Categorical (still, on foot, on bike, running in vehicle)

4.2.2 The *Static* and *Adaptive* Models

We implemented two machine-learning models in Walkie, our iOS app. We trained the *static model* before deployment (using data from the study by Künzler et al.) and used it, unchanged, for all participants and all days throughout the study. The *adaptive model* used the receptivity data of individual participants as they progressed through the study; it was re-built (within the app) every time a new receptivity the system trigerred the adaptive model.

Both these models were trained to predict *just-in-time response*. While we use several metrics of receptivity in our work, the main emphasis is on the presence of a just-in-time response. For completeness, however, we report the effect of our models on the various receptivity metrics.

We next provide the details of each model.

Static Model: We used CoreML to build and integrate the static model with the iOS app [1]. We split the original Ally iOS data (with 141 users) into five equal non-overlapping groups. We used Leave-One-Group-Out (LOGO) cross-validation to evaluate two built-in models within CoreML – MLRandomForestClassifier and MLSupportVectorClassifier. These classifiers are CoreML's implementation of RandomForest and SVM, respectively.

We tuned the models to have higher recall, since we wanted Walkie to recognize most opportune moments, even if it was at the cost of precision. We compared the models with a random classifier as a baseline and chose the model that demonstrated a greater improvement in F1 score. The SVM classifier achieved a mean F1 score of 0.36, whereas the random baseline classifier achieved only F1 score of 0.25, which is an improvement of 40% over the baseline. The RandomForest classifier achieved a mean F1 score of 0.33, only 32% improvement over baseline. We thus chose the SVM classifier as the static model to be included in our app. We show the comparison between Precision, Recall and F1 score of these three models in Figure 3.

Adaptive Model: Implementing the adaptive model was more complicated because we needed to re-build the model every time a new receptivity data point was available. In iOS 13, Apple added functionality to update CoreML models on-the-fly. Because iOS 13 was released just a couple of months before our study, however, we chose not to depend on this new feature of CoreML. To do so may have unnecessarily narrowed our potential participant population by

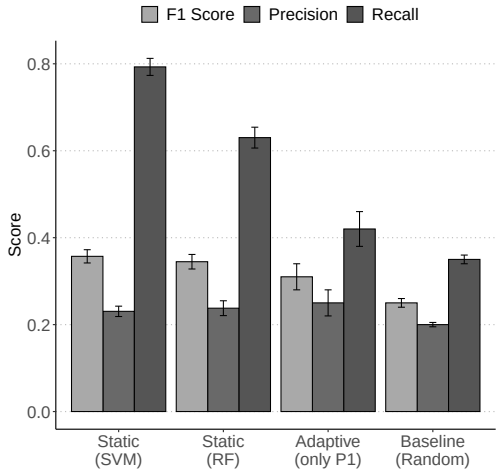


Figure 3: Comparison of the different models. All models were evaluated using 5-fold LOGO cross-validation

excluding those who had not yet updated to the latest iOS version. We instead used an open-source library called AIToolBox [7]. This library, however, had not been modified in a few years, so we had to make significant changes to make it compliant with the newer versions of iOS and Swift. Since the adaptive model would be re-trained for every notification, we wanted a model that would not be resource-intensive while being re-trained. Based on preliminary tests, Logistic Regression (LR) had the fastest training time on the device, without significantly sacrificing detection performance.

In the adaptive model, the participant's recent receptivity data was added to the model's training dataset to help with future detection. Given the structure of our study, however, each participant was prompted at most three times per day and there were thus few data points even after seven days. We thus followed a 'dual-model' approach: the adaptive model's output probability was the average of the output probability from 'P1', an LR model trained on data from the prior Ally study, and 'P2', an LR model trained on the participant's personalized data accumulated thus far.[5] If the output probability was greater than 0.50, the adaptive model classified that instance as 'receptive'. This dual-model approach enabled us to introduce a degree of personalization without being concerned about high variance of the personalized model developed from a limited set of data points.

We trained the P1 model using data from the previous Ally study. To ensure the model was light enough to run on the phone, we had to under-sample our training data.[6] We chose the Instance Hardening Threshold (IHT) method, which generates a balanced under-sampled dataset by eliminating instances that are frequently misclassified, i.e., have high instance hardness [40]. We evaluated the P1 model by LOGO cross-validation with the same non-overlapping groups we used for evaluating the static model. See Figure 3; P1 achieved an F1 score of 0.31, which is slightly lower than the F1 score for the static model, perhaps because P1 used LR whereas the static model used SVM, or perhaps because we had to train P1 on a trimmed dataset. It is important to note that, since P2 relies on continuous accumulation of data over the course of the study, we could not evaluate it before starting the study by only using past data from the Ally study. The current evaluation of P1 is just a validity check of our on-device light-weight model and how it performed when compared to the static model. The expectation was that in the live study, the adaptive model (i.e., P1 in conjunction with the continuously trained P2 model) would show improvements in performance over time.

Please note the purpose of deploying both static and adaptive models was not to compare these models with each other, but instead to observe (a) how did each model perform compared to the control model, and (b) how did the performance of the adaptive model change over time?

[5]We explain the rationale for a dual-model approach in the supplementary document.

[6]Unlike CoreML, which we used for the static model, the AIToolBox library does not export a pre-built model. The library trains and re-builds the model every time classification tasks need to be performed. Hence we had to include a trimmed version of the training data in the app so that the P1 model could train itself.

4.3 Study Logistics and Procedure

Unlike the Ally app, the Walkie app was released only for iOS users. Also, instead of releasing it through the App Store, we used Apple's in-house distribution program to distribute the apps to the participants, who could download the app by navigating to a specific webpage. Since the goal of the study was to evaluate participant receptivity, we did not want to bias the participant's interaction and usage of the app by providing monetary incentives for using the app or for engaging with the app. Instead, our study strategy used 'deception' to mask the actual goals of the study. During recruitment, we told the participants the goal of the study was to understand how different contexts affect the physical activity levels of a person throughout their day. We asked participants to interact naturally with the Walkie app and compensated them the equivalent of USD 25 if they installed the app for at least two-thirds of the study duration, i.e., 14 days.

The study protocol (including the use of deception) was approved by the Institutional Review Board (IRB) of the respective institutions. As required by the IRB, at the end of the 3-week period we emailed the participants informing them of the real goal of the study with an explanation of why deception was needed.

We used Facebook advertisements to reach potential participants, with the hope of reaching a diverse participant pool. Our search criteria was set to adults over 18 years, and belonging to a single timezone (due to technical limitations). Participants who clicked on the advertisement were taken to a landing page where we explained the study; and if interested, people could digitally sign the consent form. Once the consent form was signed we emailed the app download link and instructions to the prospective participants. The Facebook advertisement had a reach of over 30,000 people, out of which over 750 participants were redirected to our landing page; of those, 189 interested people filled out the consent form, of which 83 users downloaded the app and started the intervention. Of the 83 participants, 64 were female and 19 were male. The median age was 30 years ± 10.8 years. We had a staggered recruitment approach, and not all participants had the same start and end dates. While each participant was enrolled in the study for 3 weeks, our total data-collection period spanned a period of 6 weeks (between September 2019 and November 2019). We show a detailed demographic breakdown of participants along with their average response rates in Table 2.

Table 2: Participant demographics with average response rates.

	Male	Female	Undisclosed	Total
Showed interest in participating	47	142	1	190
Installed the app	19	64	0	83
Age	34 ± 11.7	30 ± 10.6		30 ± 10.8
Overall response rate	69.45%	70.67%		70.39%

Prior research papers on 'receptivity for interventions' and 'interruptibility to phone notifications' have defined a 'participant selection criteria' and included only data from those selected participants in their analysis [21, 6, 35, 26], like response to at least 14 questionnaires [26], or data collection for at least 10 days and 20 responses [35], or top 95 percentile of users who responded [21]. In our work, however, we avoid use of any such selection criteria and report results for data from all the 83 users for most of the analysis. Across the 83 users, we had 1091 messages delivered by the *control* model, 691 messages delivered by the *static* model, and 241 messages delivered by the *adaptive* model; resulting in a total of 2023 delivered messages. We present a distribution of the number of times each model was used to deliver a message over the course of the study in Figure 4.

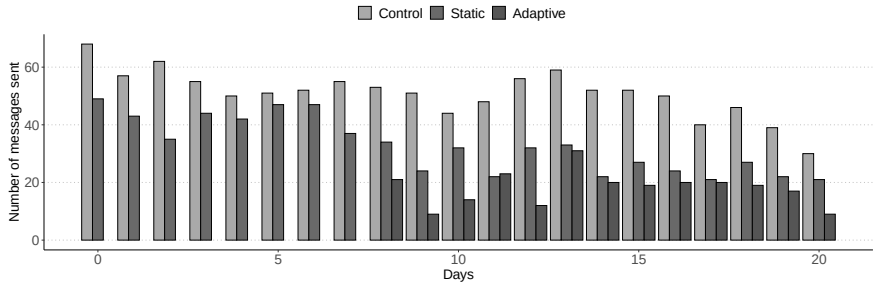


Figure 4: Daily distribution of the models used to trigger intervention alerts.

4.4 Problem with Deployment

Approximately two or three weeks from the start of data collection we realized the adaptive model was *never* getting triggered for participants. Even after the initial 7 days, the Walkie app was triggering only the control and static models, because a bug was preventing the app from triggering the adaptive model. We fixed the bug, and released an update and asked all users to update to the new version. Unfortunately some users had already completed their study duration, whereas others did not update the app. In the end, we eventually had only 61 users who received one or more initiating messages through the adaptive model. For these 61 users, we had 785 messages delivered by the control model, 541 messages delivered by the static model, and 241 messages delivered by the adaptive model, leading to a total of 1567 delivered messages.

5 Our Hypotheses

Before evaluating the results, we formed the following hypotheses based on our expectation of the outcomes if the models performed as intended.

For *RQ-1*,

- *H-1*: We hypothesize that on a population level, across all initiating messages, there would be a significant difference in receptivity across the three delivery models. The *static* and *adaptive* models should both have significantly higher receptivity than the *control* model.
- *H-2*: We hypothesize that on a population level, across all users, there would be a significant effect of model type on receptivity. The *static* and *adaptive* models should have significantly higher receptivity than the *control* model.[7]

For *RQ-2*,

- *H-3*: We hypothesize that individual participants would have higher receptivity when interventions were delivered through the *static* and *adaptive* models, as compared to the *control* model.

For *RQ-3*,

- *H-4*: We hypothesize that the receptivity to interventions delivered by the *static* model would remain constant over the course of the study.
- *H-5*: We hypothesize that the receptivity to interventions delivered by the *adaptive* model would increase as the study progressed.

Note that while the hypotheses mention *receptivity*, our primary focus is on *just-in-time response*. For completeness, however, we also report the other metrics of receptivity, which we term "secondary metrics". Hence, based on the type of metric, "better" receptivity would mean an increase in just-in-time response rate, increase in response rate, increase in conversation engagement rate, and a decrease in response delay.

6 Evaluation

In this section we analyze the receptivity data and evaluate our hypotheses. For clarity, we break the evaluation into three parts, to reflect the three research questions we seek to answer. Note, we adjusted all the p-values to account for multiple comparisons: we used the Benjamini-Hochberg (BH) procedure [5] to correct for the expected proportion of Type I errors (or False Discovery Rate) across all hypotheses.

[7] In *RQ-1* we aim to understand the differences on a population level. This includes two types of analyses: (a) when we consider all messages independently (*H-1*); and (b) when we consider average response across all the users (*H-2*). The *within* person differences are explored in *RQ-2*.

6.1 Exploring *RQ-1*

On a population level, i.e., across intervention messages and across all users, does delivering interventions at a ML-detected time lead to higher receptivity than delivering interventions at a random time?

We start by analyzing the receptivity metrics for each initiating message. We fit a binomial Generalized Linear Model (GLM) to evaluate the associations the ML models had on just-in-time response as compared to the control model.[8] We found that the different machine-learning models (control, static, and adaptive) had a significant effect on the just-in-time response ($\chi^2(2) = 23.189$, $p < 0.001$).[9] On post-hoc analysis with Dunnett's Test, we observed that the static model showed a significant improvement of approximately 40% in just-in-time receptivity when compared to the control model ($p < 0.001$). The adaptive model had an improvement of more than 15% over the control model, but the improvement was not significant ($p = 0.271$).

Next, we discuss the secondary metrics; we observed that the type of machine-learning model had a significant effect on the likelihood of "response", i.e., if the participant ever responded to the initiating message (irrespective of time), $\chi^2(2) = 15.001$, $p = 0.003$. Post-hoc analysis with Dunnett's Test revealed that, when compared with the control model, both the static and adaptive models led to a significant increase in likelihood of response, of approximately 12% each ($p = 0.003$ and $p = 0.048$, respectively). This is an interesting observation as it suggests that participants were more likely to respond to initiating messages that were generated through the static and adaptive models as compared to the control model, even though the adaptive model did not lead to a significantly higher likelihood of just-in-time response. Further, the type of machine-learning model showed a significant effect on conversation engagement, $\chi^2(2) = 18.741$, $p = 0.001$. Post-hoc analysis revealed that the static model led to a 36% increase in the likelihood of conversation engagement relative to the random model ($p < 0.001$). As in the case of just-in-time response, the adaptive model did not result in a significant improvement in conversation engagement, with an increase of 12% over the control model ($p = 0.439$). Finally, for response delay, one-way ANOVA did not reveal a significant effect of model type, $F(2, 1382) = 1.576, p = 0.310$; we nonetheless conducted Dunnett's test to observe the differences. Although the difference was not statistically significant, the static model – on average – led to a slightly shorter response delay of 16 minutes, about 17% faster than in the control model. The adaptive model did not show any differences as compared to the control model. We present the detailed results in Table 3.

In the above analysis, we looked at individual initiating messages. Next, we discuss the population-level differences across all the participants. To this end, we aggregate the receptivity metrics of each participant over the study period w.r.t. to the model used to deliver the initiating message. For example, over the course of the study, if a participant had just-in-time response to 2 out of 10 initiating messages delivered by the control model, 4 out of 8 messages by the static model, and 2 out of 5 messages delivered by the adaptive model, the participant's just-in-time response rate would be 0.2 for control, 0.5 for static, and 0.4 for adaptive. We computed these aggregates for all 83 users for each of the four metrics. Since we analyzed data from all participants and did not have an inclusion criteria, this analysis could be misleading if not done correctly. For instance, Alice received 2 messages and after responding to both the messages, just-in-time, she uninstalled the app within a day; whereas Bob completed all 21 days of the study, received 40 messages and gave just-in-time response to 10. In this scenario, Alice's just-in-time response rate would be 100% and Bob's would be 25%. If compared directly, Alice's response rate would lead to a skewed analysis. Most studies exclude such outliers based on some criteria. In our case, we felt there was no scientifically justifiable inclusion/exclusion criterion, so we chose to do a weighted analysis based on the total number of silent messages the server 'sent' to a participant, regardless of a response. The number of silent 'sent' messages is also a measure of how long a participant remained in the study. If the participant uninstalled the app, they were not sent any messages.

We used ANOVA to understand the effect of model type on participants' receptivity rates. We observed the model type had a significant effect on the just-in-time response rate ($F(2, 223) = 5.131, p = 0.018$). Post-hoc analysis with Dunnett's test revealed that the static model led to a just-in-time response rate approximately 36% higher than the control model ($p = 0.007$). The adaptive model resulted in just-in-time response rate 14% higher than the control model, but this was not statistically significant ($p = 0.505$). For the overall response rate, we observed that model type did not

[8] We choose the appropriate test based on the type of the dependent variable. We used a binomial GLM for the just-in-time response, response, and conversation-engagement metrics. We used a one-way Analysis of Variance (ANOVA) for response delay. From a typical statistical analysis perspective, it may initially seem that a population level analysis (across all intervention messages) violates the independence assumptions of GLM. However, it is important to consider the research question we aim to answer, i.e., how did the *static* and *adaptive* models perform when compared to the control model, across all intervention prompts. We argue that, from a machine-learning model's perspective, each new prediction is independent of prior predictions and hence can be considered independent. Thus, in this regard we are not violating the assumption for GLM. Such population level analyses is consistent with prior works in receptivity [21] and interruptibility [26, 24].

[9] The Chi Square test compares the GLM to a null model.

Table 3: Detailed analysis across all initiating messages.We report the absolute change of the static and dynamic models over the control model, along with the percentage improvement in brackets.

Comparison	Mean Difference (% change)	Std. Error	95% Confidence Interval		Adj. p-value	
			Lower Bound	UpperBound		
Just-in-time response (as likelihood; control = 0.284)						
static – control	+0.111 (+39.08%)	0.022	0.057	0.164	<0.001	***
adaptive – control	+0.044 (+15.49%)	0.033	–0.034	0.122	0.270	
Overall response (as likelihood; control = 0.656)						
static – control	+0.080 (+12.19%)	0.022	0.028	0.133	<0.003	**
adaptive – control	+0.077 (+11.73%)	0.032	0.006	0.154	0.048	*
Conversation engagement (as likelihood; control = 0.267)						
static – control	+0.097 (+36.32%)	0.022	0.045	0.150	<0.001	***
adaptive – control	+0.032 (+11.98%)	0.032	–0.044	0.109	0.439	
Response delay (in minutes; control = 90.530)						
static – control	–16.670 (–18.41%)	9.465	–38.870	5.532	0.144	
adaptive – control	–4.006 (–4.42%)	13.733	–36.227	28.213	0.819	
. $p < 0.1$, * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$						

have a significant effect ($F(2, 223) = 3.066, p = 0.097$). Although not statistically significant, both static and adaptive models had an improvement of approximately 12% over the control model ($p = 0.052$ and $p = 0.197$, respectively). Further, we observed that the model type had a significant effect on engagement rate ($F(2, 223) = 4.065, p = 0.042$), with the static model showing 33% improvement over the control model ($p = 0.016$). Finally, the models did not show any significant effect on response delay ($F(2, 214) = 0.053, p = 0.948$). We report the detailed Dunnett's test analysis in Table 4.

Table 4: Detailed analysis across all users. We report the absolute change of the static and dynamic models over the control model, along with the percentage improvement in brackets.

Comparison	Mean Difference (% change)	Std. Error	95% Confidence Interval		Adj. p-value	
			Lower Bound	Upper Bound		
Just-in-time response (as likelihood; control = 0.290)						
static – control	+0.106 (+36.55%)	0.033	0.008	0.203	0.007	**
adaptive – control	+0.041 (+14.13%)	0.048	–0.063	0.147	0.505	
Overall response (as likelihood; control = 0.682)						
static – control	+0.079 (+11.58%)	0.035	–0.022	0.181	0.052	
adaptive – control	+0.081 (+11.87%)	0.051	–0.029	0.191	0.197	
Conversation engagement (as likelihood; control = 0.229)						
static – control	+0.093 (+40.61%)	0.032	–0.003	0.190	0.016	**
adaptive – control	+0.030 (+13.10%)	0.047	–0.074	0.135	0.621	
Response delay (as minutes; control = 98.613)						
static – control	–4.443 (–4.50%)	15.409	–43.567	34.679	0.948	
adaptive – control	+1.647 (+1.67%)	22.769	–41.703	44.999	0.819	
. $p < 0.1$, * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$						

6.2 Exploring *RQ-2*

How does receptivity of individual users change when interventions are delivered through ML-based intervention timing vs. at random times?

While investigating the hypotheses for *RQ 1*, we found that overall, across the population, the static model led to a significant improvement in just-in-time response, overall response, and conversation engagement. While such an analysis provides a good representation of receptivity across all users, our second research question aimed to evaluate *within-participant* differences when they received a static or adaptive model as compared to the control model. The goal is to evaluate whether an individual participant's receptivity changed based on the model used to deliver the initiating message.

We used generalized linear mixed effects models for our analysis. Since the goal was to observe how individual participant's receptivity changed based on the model used to deliver the message, each user should receive at least one message through each model for a proper comparison. As noted in Section 4.4, however, a software problem meant only 61 participants received at least one initiating message through the adaptive model. Hence, for this analysis, we included data only from these 61 participants. This is the only analysis we had to limit to a subset of the initial 83 participants.

We observed that the model type had a significant effect on the just-in-time response rate ($\chi^2(2) = 13.433$, $p = 0.001$). On post-hoc analysis, we observed that the static model showed a significant improvement of over 36% in just-in-time receptivity when compared to the control model ($p = 0.002$). This result suggests that if a participant received a prompt from the static model, they were more likely to be receptive than if the same participant received the prompt through the control model. The adaptive model led to an increase of almost 10% over the control model, but the result was not significant ($p = 0.558$).

For the secondary metrics, we observed that the type of model had an effect on the likelihood of response ($\chi^2(2) = 8.364$, $p = 0.00$). Post-hoc analysis revealed that only the static model had a significant improvement over the control model, with an improvement of almost 10% ($p = 0.015$). Further, our analysis showed that the model type had a significant effect on the likelihood of conversation engagement ($\chi^2(2) = 10.407$, $p = 0.017$), with post-hoc analysis revealing that the static model led to an improvement of over 32% in the likelihood of conversation engagement over the control model ($p = 0.007$). Finally, using a repeated measures ANOVA, we did not find any significant effect of model type on the response delay. Although not statistically significant, the static and adaptive model led to a 20% and 13% reduction in time taken to respond to interventions, respectively. We present the detailed findings in Table 5.

Further, when we added a random slope for 'model type' to the mixed effects model, a comparison between the two did not reveal any difference ($p = 1.00$), thus suggesting that the within-participant differences were similar across all participants. These results are promising, and indicate that individual participants were more receptive when they received prompts by the static model as compared to the control model.

Given our inclusion criteria for this analysis, we have some users who received just one message from the adaptive model. It can be argued that including such users might introduce some outliers in our data, which might affect the analyses. However, given the lack of a better (and scientifically justifiable) inclusion criteria, we nonetheless included users who received only one message. To explore how a more stringent criterion would affect the results, we briefly analyzed the data from participants who received at least 5 messages in each category. Doing so ensured that we only included participants who were active (or stayed) in the study for a longer duration. We observed that the static model led to a significant increase in the just-in-time response rate, over 40% higher than the control model. This result suggests that the difference in models becomes more prominent as we evaluate the more "active" users. In a way, our results (presented in Table 5) can be thought of as the "worst-case" results.

6.3 Exploring *RQ-3*

How do the different models for predicting receptivity perform over time?

As we report in the preceding sections, the messages delivered by the static model led to receptivity metrics that were significantly higher than those the control model, for most situations. The adaptive model, however, did not seem to perform significantly better. Those results were based on an analysis across the full study period. A *day-by-day analysis*, however, may provide more insights regarding whether and how the adaptive model's performance changed over the days – in short, whether it adapted well to each participant. This analysis also helps to evaluate our third research

Table 5: Detailed analysis to understand *within-participant* differences. We report the absolute change of the static and dynamic models over the control model, along with the percentage improvement in brackets.

Comparison	Mean Difference (% change)	Std. Error	95% Confidence Interval		Adj. p-value	
			Lower Bound	Upper Bound		
Just-in-time response (as likelihood; control = 0.276)						
static – control	+0.101 (+36.60%)	0.033	0.035	0.170	0.002	**
adaptive – control	+0.027 (+9.58%)	0.041	–0.044	0.109	0.558	
Overall response (as likelihood; control = 0.738)						
static – control	+0.072 (+9.75%)	0.028	0.015	0.116	0.015	*
adaptive – control	+0.031 (+4.20%)	0.038	–0.046	0.092	0.493	
Conversation engagement (as likelihood; control = 0.261)						
static – control	+0.084 (+32.18%)	0.034	0.021	0.153	0.007	**
adaptive – control	+0.009 (+3.44%)	0.040	–0.057	0.089	0.819	
Response delay (as minutes; control = 99.500)						
static – control	–19.950 (–20.05%)	11.725	–39.500	3.500	0.124	
adaptive – control	–13.830 (–13.89%)	13.585	–41.000	13.500	0.439	

. $p < 0.1$, * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

question. Given the nature of the adaptive model, we expected it to improve over time as more individual-specific data was added to the model.

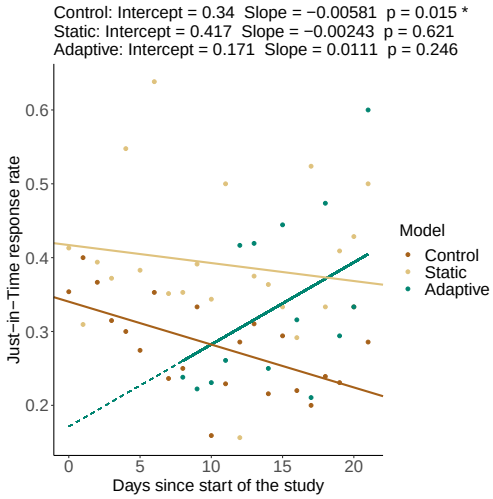
For this analysis, we arranged all the users according to their relative start day. So all users are arranged from day 1 to day 21, irrespective of their actual (calendar) start day. For each day, we measured the just-in-time response rate, response rate, conversation engagement, and average response delay across all users, and fit a weighted linear model to each model's response metric so we can observe the trend in receptivity across the days.[10] We show the trends in Figure 5.[11] As the study progressed, the just-in-time response rate dropped significantly for the control model ($p = 0.015$) (Figure 5a). For the static model, there was a slight downward trend, but it was not significant. For the adaptive model there was a steep upward trend with a slope of 0.011, suggesting that the just-in-time response to adaptive model increased by 1 percentage-point each day; this trend was not statistically significant ($p = 0.246$). The observation is quite encouraging, suggesting that the adaptive model was able to learn and personalize over time, and eventually improving the just-in-time response. In fact, after day 17, the adaptive model seems to have had higher just-in-time response rate than the static model. Further, on Day 21, the adaptive model had an increase of over 51% in just-in-time response rate as compared to Day 8.

Looking at the conversation-engagement rate (Figure 5b), the control model showed a significant downward trend ($p = 0.018$), suggesting that for messages delivered by the control model, the conversation-engagement rate declined over the course of the study. The static model had a slight, yet insignificant, downward trend, similar to the just-in-time response rate. The adaptive model had a significant positive trend ($p = 0.037$), with a slope of 0.017, which translates to a 1.7 percentage-point increase in conversation-engagement rate each day. This result further supports our expectation that the adaptive model would be able to learn from the personalized data and improve itself. Although conversation engagement was not the primary metric of focus, it still is an important part of the overall concept of receptivity, especially to chat-based interventions like Ally and Walkie.

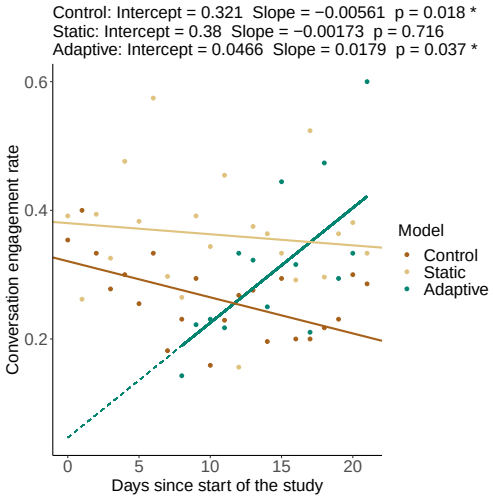
Next we evaluated the effect on the overall response rate (Figure 5c). Consistent with the two previous metrics, the control model had a significant downward trend, suggesting that overall response rate for messages delivered by the control model decreased over the course of the study ($p = 0.03$). The static model was more interesting: although

[10] Since the model selection was completely random and was happening in real time on the participants' phones, there could potentially be an imbalance on the total number of messages being triggered by the different models everyday. To account for this, we decided to fit weighted linear models. Each model is weighted on the number of messages *delivered* by that model. We include the results from an unweighted model in the Supplementary document, the results of which – because of bias – show stronger and significant trends; however, we argue weighted analysis is the proper approach, even though it leads to more conservative inferences.

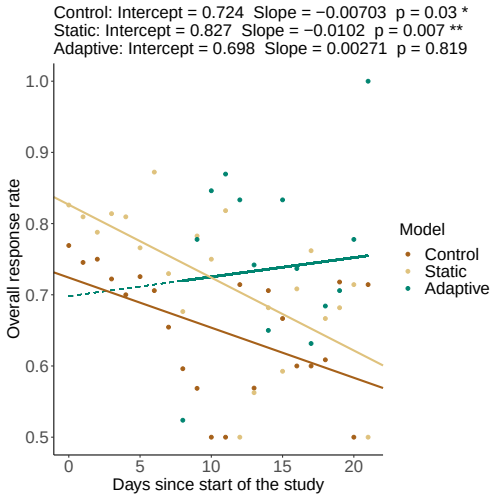
[11] As recommended by R4 in the previous submission cycle, we also tried an alternative approach to evaluate *RQ-3*. We added the interaction variable between day-of-study and the model type to the binomial generalized linear mixed effects model built for *RQ-2*. We obtained the same conclusions as our current analysis. We report the plots from the mixed effects model in the supplementary material.



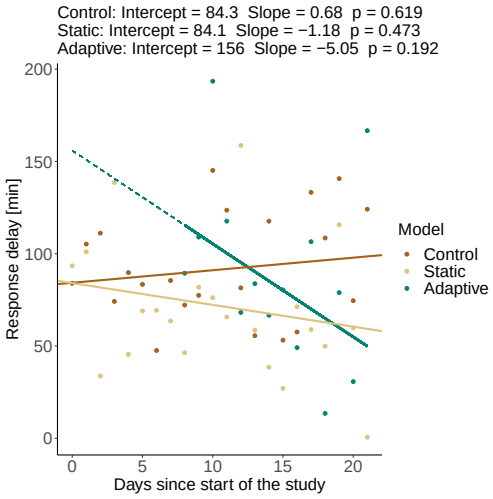
(a) Effect of the different models on the just-in-time response rate over time.



(b) Effect of the different models on the conversation-engagement rate over time.



(c) Effect of the different models on the overall response rate over time.



(d) Effect of the different models on the response delay over time.

Figure 5: The performance over time of the models on the receptivity metrics. The adaptive model was only activated starting day 8; the dotted lines represent the projection of the trend for the adaptive model from day 1 to day 7.

the just-in-time response rate to messages from the static model did not reduce over the days, there was a decline in the overall response rate ($p = 0.007$). In other words, the static model did what it was trained to do: it led to a higher just-in-time response rate than the control model, and kept it high as the just-in-time response rate for the control model fell. However, just like the control model, if the participant did not respond just-in-time, the overall response rate reduced over the days, following the trend of attrition common in mobile sensing and mHealth studies [44, 41]. The adaptive model was able to maintain a satisfying response rate throughout.

Finally, for response delay (Figure 5d), we did not discover a trend that was significant in any of the three models. We make several encouraging observations, however. At the beginning, the response delay for the control and static models was approximately the same, but they diverged as the study progressed; the control model had an increasing trend, whereas the static model had a decreasing trend, suggesting that the participants' response time to the static model improved as the study progressed. The adaptive model shows an even steeper improvement, with a slope of -5.05, suggesting an improvement of 5 minutes each day. It is important to note that the results for response delay were not significant, and more research is needed.

6.4 Summary of Results

Based on our analysis of RQ-1 and RQ-2, our first three hypotheses (H-1, H-2, and H-3) were partially true. The *static* model led to a significant improvement over the control model in just-in-time response rate, overall response rate, and conversation engagement. The *adaptive* model led to slight improvements over the control model, though the results were not significant. The results for RQ-3 provide some insights on the lack of significant improvement in the adaptive model. As shown in Figure 5, the receptivity to the adaptive model started similar to that of the control model, but it kept improving as the study progressed, although the average over the study was not significantly higher than the control model.

Furthermore, we observed that after Day 17, the just-in-time response rate from the adaptive model was higher than that from the static model. Similarly, after Day 16, the conversation engagement from adaptive model was higher than that from the static models. While these are encouraging results that indicate the adaptive model continued to improve over time, in our short study we could not observe the trend over a longer period of time. We hypothesize the adaptive model would continue to improve beyond Day 21, outperforming the static model, although it would eventually plateau. We anticipate future research will be able to test this hypothesis.

7 Discussion

In our work, we show that machine-learning models can be used to predict receptivity to interventions. We found that intervention alerts delivered by a *static* model performed significantly better than delivering interventions at random times. Further, we found that receptivity to an adaptive model – which learnt user specific features over the course of the study – improved as the study progressed. In this section, we discuss the implications of our results in three broad categories, along with the limitations, and future directions for each.

7.1 Domain-specific Models to Detect Receptivity

In prior work, Mehrotra et al. found that *notification category* was one of the top features to determine whether users would react to notifications [25], thus highlighting the importance of *who* or *what* is sending an alert. Along with *who* and *what* is sending an alert, Visuri et al. showed that the actual content of a prompt is also important [42]. Using a semantic analysis of a notification's content (along with contextual features) drastically improved the detection of opportune moments to smartphone notifications [42]. These findings show the importance of the content of a notification and could potentially affect the generalizability of ML models to different interventions.

In our work, we considered receptivity to the initial greeting message similar to the considerations made by Künzler et al. [21]. The initial message was a generic greeting message like "Hello [participantName]" or "Good morning [participantName]". Only after the participants replied back to the greeting did the actual intervention conversation start. Since the receptivity to interventions was not affected by the content of intervention, we argue that our results and models could be generalizable to other JITAIs with a similar level of intervention engagement, or to other JITAI that use chatbots for digital coaching.

It is important, however, to be mindful of the findings by Mehrotra et al. [25] and Visuri et al. [42]. We do not know whether and how our results might generalize to interventions that are more involved, or require the user to actively perform some task, e.g., to take 10 deep breaths or to take some medications. More research is needed, especially in the domain of cognitive availability, to explore how receptivity changes with intervention burden.

7.2 Dependence on Intervention Design and Effectiveness

Based on the definition of JITAI proposed by Nahum-Shani et al. [31], *receptivity* can be considered a *tailoring variable*, which indicates whether a user is available to receive an intervention at any given time. For receptivity to be an informative parameter, it is imperative that the actual intervention being delivered is effective. Intervention designs that are ineffective or cumbersome might not lead to engagement, regardless of whether a user is available to receive the information.

We used an intervention design similar to that proposed by Kramer et al. [18, 19]. Although evaluating the effectiveness of interventions is beyond the scope of this paper, Kramer et al. found that when compared to a baseline (no intervention) period, on average the participants in their study significantly improved their daily step counts by 438 steps [19].

Another aspect to consider is the determination of a time-window for receptivity. Based on prior work by Künzler et al. [21] and Mehrotra et al. [25], in our exploratory work we considered receptivity to interventions if a participant responded to an intervention message within 10-minutes of delivery. We believe that the choice of the time-window to determine receptivity would eventually depend on the intervention type and what is considered as an acceptable receptive duration for that intervention. Some time-critical intervention designs might require response within a 1-minute window to be considered "receptive", whereas others might consider a 1-hour response time acceptable. Depending on what is considered receptive, the models and their subsequent performance could significantly change.

In future work, we plan to delve deeper into the influence of intervention design and intervention effectiveness on receptivity-detection models.

7.3 Personalized Models to Detect Receptivity

In prior work, Morrison et al. deployed a mobile stress-management intervention system, where they built personalized Naive Bayes models to deliver interventions at *opportune* moments [29]. They found, however, that there was no difference in how the participants interacted with notifications at opportune times vs. those delivered at random times. One reason could be that they used a static personalized model trained after an initial "learning period." It is possible that the model did not have enough data to be adequately trained, i.e., the cold-start problem.

Our results show that the *adaptive* model (which followed a dual-model approach) started with poor performance, but the performance improved as more data was available. Further, even in our results, we observed that the adaptive model did not perform significantly better than the control model if we considered the entire study duration (RQ-1 and RQ-2). It was when we evaluated the day-by-day performance (RQ3), did we notice the increasing trend. These results suggest that personalized models can improve over time with more data. Further, given that the static model performed significantly better over the control model, a dual-model solution could be appropriate to deal with the cold-start problem. We discuss some potential approaches in the next section.

7.4 Building better Models

As an exploratory study, our work used several *in-the-moment* contextual features, like phone battery level, device interaction, date/time, and physical activity. Since our models led to significantly higher receptivity than a random model, it is natural to question whether incorporating more features could further improve receptivity. In our work, we did not consider the location of a user; although Künzler et al. did not find any significant association between a user's location and his/her receptivity, several other works have noted the effect of location type on notification engagement [38, 35]. Other features like demographic information [21, 36, 35] and personality traits [26, 21] have shown to correlate with receptivity and interruptibility.

Further, our results show that the performance of the *adaptive* model improved as the study progressed. This is one of the first works to show the improvement of an adaptive model to detect receptivity in a real-world deployment, and highlights the potential of deploying adaptive models that can be tuned to each participant for optimal performance in the long term. In our work, the adaptive model included two models (P1 & P2), and both had equal weights. Given

the promising results, we now hope to explore models with adaptive weights based on the relative number of data points to train each model; thus, initially P1 would have a higher weight, but over time as P2 accumulates more data points its weight would increase and eventually be higher than P1. In the future, we plan to understand how adding new features would affect model performance, and explore what type of adaptive model might be best, how to best change the weights of a personalized model, and also how to determine the optimal number of days to enable the personalized component.

Finally, although our results are promising, they are still preliminary. We had 83 users in our study who participated for only 3 weeks; most behavior change programs last longer than 3 weeks. Hence, more research is needed to evaluate model performance and how receptivity changes over a longer period of time.

8 Summary and Conclusion

We conducted a study in which we deployed machine-learning models that detect momentary receptivity in the natural environment. We leveraged prior work in receptivity to JITAIs and deployed a chatbot-based digital coach – Walkie – that provided physical-activity interventions and motivated participants to achieve their step goals. The Walkie app was available for iOS and included two machine-learning models that used information about the app user's context to predict whether that person was likely to be receptive at that moment. We used two types of machine-learning models: (a) a static model, that was built before the study started and remained constant for all participants and all time; and (b) an adaptive model, that continuously learnt the receptivity of individual participants and updated itself as the study progressed. For comparison, we deployed (c) a control model that sent intervention messages at random times. The choice of model to be used for delivery was randomized for each intervention message. We observed that messages delivered using the machine-learning models led up to a 40% improvement in receptivity as compared to the control model. Further, we evaluated the temporal dynamics of the different models over time, and observed that while the receptivity to messages from the control model declined, receptivity to messages from the adaptive model increased over the course of the study.

References

[1] Apple. Coreml. https://developer.apple.com/documentation/coreml, 2020. [Online; accessed 04-February-2020].

[2] Daniel Avrahami and Scott E Hudson. Responsiveness in instant messaging: predictive models supporting inter-personal communication. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 731–740. ACM, 2006.

[3] Stephanie Bauer, Judith de Niet, Reinier Timman, and Hans Kordy. Enhancement of care through self-monitoring and tailored feedback via text messaging and their use in the treatment of childhood overweight. *Patient education and counseling*, 79(3):315–319, 2010.

[4] 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*, 40(6):1244–1253, 2014.

[5] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, 57(1):289–300, 1995.

[6] Woohyeok Choi, Sangkeun Park, Duyeon Kim, Youn-kyung Lim, and Uichin Lee. Multi-stage receptivity model for mobile just-in-time health intervention. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 3(2), June 2019.

[7] Kevin Coble. Aitoolbox. https://github.com/KevinCoble/AIToolbox, 2020. [Online; accessed 04-February-2020; Git commit bada633].

[8] 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 SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 1797–1806. ACM, 2008.

[9] Tilman Dingler, Dominik Weber, Martin Pielot, Jennifer Cooper, Chung-Cheng Chang, and Niels Henze. Language learning on-the-go: Opportune moments and design of mobile microlearning sessions. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '17, New York, NY, USA, 2017. Association for Computing Machinery.

[10] Andreas Filler, Tobias Kowatsch, Severin Haug, Fabian Wahle, Thorsten Staake, and Elgar Fleisch. Mobilecoach: A novel open source platform for the design of evidence-based, scalable and low-cost behavioral health interventions: overview and preliminary evaluation in the public health context. In *Wireless Telecommunications Symposium (WTS), 2015*, pages 1–6. IEEE, 2015.

[11] Joel E Fischer, Chris Greenhalgh, and Steve Benford. Investigating episodes of mobile phone activity as indicators of opportune moments to deliver notifications. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, pages 181–190. ACM, 2011.

[12] Koel Ghorai, Shahriar Akter, Fatema Khatun, and Pradeep Ray. mhealth for smoking cessation programs: a systematic review. *Journal of personalized medicine*, 4(3):412–423, 2014.

[13] David H Gustafson, Fiona M McTavish, Ming-Yuan Chih, Amy K Atwood, Roberta A Johnson, Michael G Boyle, Michael S Levy, Hilary Driscoll, Steven M Chisholm, Lisa Dillenburg, et al. A smartphone application to support recovery from alcoholism: a randomized clinical trial. *JAMA Psychiatry*, 71(5):566–572, 2014.

[14] Joyce Ho and Stephen S Intille. Using context-aware computing to reduce the perceived burden of interruptions from mobile devices. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 909–918. ACM, 2005.

[15] Karen Hovsepian, Mustafa Al'Absi, Emre Ertin, Thomas Kamarck, Motohiro Nakajima, and Santosh Kumar. cStress: Towards a Gold Standard for Continuous Stress Assessment in the Mobile Environment. *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*, pages 493–504, 2015.

[16] Donna M Kazemi, Brian Borsari, Maureen J Levine, Shaoyu Li, Katie A Lamberson, and Laura A Matta. A systematic review of the mhealth interventions to prevent alcohol and substance abuse. *Journal of health communication*, 22(5):413–432, 2017.

[17] Tobias Kowatsch, Dirk Volland, Iris Shih, Dominik Rüegger, Florian Künzler, Filipe Barata, Andreas Filler, Dirk Büchter, Björn Brogle, Katrin Heldt, et al. Design and evaluation of a mobile chat app for the open source behavioral health intervention platform mobilecoach. In *International Conference on Design Science Research in Information Systems*, pages 485–489. Springer, 2017.

[18] Jan-Niklas Kramer, Florian Künzler, Varun Mishra, Bastien Presset, David Kotz, Shawna Smith, Urte Scholz, and Tobias Kowatsch. Investigating intervention components and exploring states of receptivity for a smartphone app to promote physical activity: protocol of a microrandomized trial. *JMIR research protocols*, 8(1):e11540, 2019.

[19] Jan-Niklas Kramer, Florian Künzler, Varun Mishra, Shawna N. Smith, David Kotz, Urte Scholz, Elgar Fleisch, and Tobias Kowatsch. Which Components of a Smartphone Walking App Help Users to Reach Personalized Step Goals? Results From an Optimization Trial. *Annals of Behavioral Medicine*, pages 1–11, March 2020.

[20] Florian Künzler, Jan-Niklas Kramer, and Tobias Kowatsch. Efficacy of mobile context-aware notification management systems: A systematic literature review and meta-analysis. In *Wireless and Mobile Computing, Networking and Communications (WiMob),*, pages 131–138. IEEE, 2017.

[21] Florian Künzler, Varun Mishra, Jan-Niklas Kramer, David Kotz, Elgar Fleisch, and Tobias Kowatsch. Exploring the state-of-receptivity for mhealth interventions. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. (IMWUT)*, 3(4), December 2019.

[22] Seth S Martin, David I Feldman, Roger S Blumenthal, Steven R Jones, Wendy S Post, Rebeccah A McKibben, Erin D Michos, Chiadi E Ndumele, Elizabeth V Ratchford, Josef Coresh, et al. mactive: a randomized clinical trial of an automated mhealth intervention for physical activity promotion. *Journal of the American Heart Association*, 4(11):e002239, 2015.

[23] Afra Mashhadi, Akhil Mathur, and Fahim Kawsar. The myth of subtle notifications. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, pages 111–114. ACM, 2014.

[24] Abhinav Mehrotra, Sandrine R Müller, Gabriella M Harari, Samuel D Gosling, Cecilia Mascolo, Mirco Musolesi, and Peter J Rentfrow. Understanding the role of places and activities on mobile phone interaction and usage patterns. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3):1–22, 2017.

[25] Abhinav Mehrotra, Mirco Musolesi, Robert Hendley, and Veljko Pejovic. Designing content-driven intelligent notification mechanisms for mobile applications. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 813–824. ACM, 2015.

[26] Abhinav Mehrotra, Veljko Pejovic, Jo Vermeulen, Robert Hendley, and Mirco Musolesi. My phone and me: understanding people's receptivity to mobile notifications. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI)*, pages 1021–1032. ACM, 2016.

[27] Varun Mishra, Byron Lowens, Sarah Lord, Kelly Caine, and David Kotz. Investigating contextual cues as indicators for EMA delivery. In *Proceedings of the International Workshop on Smart & Ambient Notification and Attention Management (UbiTtention)*, pages 935–940. ACM, September 2017.

[28] Varun Mishra, Gunnar Pope, Sarah Lord, Stephanie Lewia, Byron Lowens, Kelly Caine, Sougata Sen, Ryan Halter, and David Kotz. The case for a commodity hardware solution for stress detection. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct (UbiComp'18)*. ACM, October 2018.

[29] Leanne G Morrison, Charlie Hargood, Veljko Pejovic, Adam WA Geraghty, Scott Lloyd, Natalie Goodman, Danius T Michaelides, Anna Weston, Mirco Musolesi, Mark J Weal, et al. The effect of timing and frequency of push notifications on usage of a smartphone-based stress management intervention: An exploratory trial. *PloS one*, 12(1):e0169162, 2017.

[30] Inbal Nahum-Shani, Eric B Hekler, and Donna Spruijt-Metz. Building health behavior models to guide the development of just-in-time adaptive interventions: A pragmatic framework. *Health Psychology*, 34(S):1209, 2015.

[31] Inbal Nahum-Shani, Shawna N Smith, Bonnie J Spring, Linda M Collins, Katie Witkiewitz, Ambuj Tewari, and Susan A Murphy. Just-in-time adaptive interventions (jitais) in mobile health: key components and design principles for ongoing health behavior support. *Annals of Behavioral Medicine*, pages 1–17, 2016.

[32] T. Okoshi, K. Tsubouchi, M. Taji, T. Ichikawa, and H. Tokuda. Attention and engagement-awareness in the wild: A large-scale study with adaptive notifications. In *2017 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 100–110, March 2017.

[33] Tadashi Okoshi, Julian Ramos, Hiroki Nozaki, Jin Nakazawa, Anind K Dey, and Hideyuki Tokuda. Reducing users' perceived mental effort due to interruptive notifications in multi-device mobile environments. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 475–486. ACM, 2015.

[34] Veljko Pejovic and Mirco Musolesi. Interruptme: designing intelligent prompting mechanisms for pervasive applications. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 897–908. ACM, 2014.

[35] Martin Pielot, Bruno Cardoso, Kleomenis Katevas, Joan Serrà, Aleksandar Matic, and Nuria Oliver. Beyond interruptibility: Predicting opportune moments to engage mobile phone users. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3):91, 2017.

[36] Martin Pielot, Tilman Dingler, Jose San Pedro, and Nuria Oliver. When attention is not scarce-detecting boredom from mobile phone usage. In *Proceedings of the 2015 ACM international joint conference on pervasive and ubiquitous computing*, pages 825–836. ACM, 2015.

[37] William Riley, Jami Obermayer, and Jersino Jean-Mary. Internet and mobile phone text messaging intervention for college smokers. *Journal of American College Health*, 57(2):245–248, 2008.

[38] Hillol Sarker, Moushumi Sharmin, Amin Ahsan Ali, Md Mahbubur Rahman, Rummana Bari, Syed Monowar Hossain, and Santosh Kumar. Assessing the availability of users to engage in just-in-time intervention in the natural environment. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 909–920. ACM, 2014.

[39] Chen-Hsuan Shih, Naofumi Tomita, Yanick X Lukic, Álvaro Hernández Reguera, Elgar Fleisch, and Tobias Kowatsch. Breeze: Smartphone-based acoustic real-time detection of breathing phases for a gamified biofeedback breathing training. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(4):1–30, 2019.

[40] Michael R. Smith, Tony Martinez, and Christophe Giraud-Carrier. An instance level analysis of data complexity. *Machine Learning*, 95(2):225–256, May 2014.

[41] Vincent W. S. Tseng, Michael Merrill, Franziska Wittleder, Saeed Abdullah, Min Hane Aung, and Tanzeem Choudhury. Assessing mental health issues on college campuses: Preliminary findings from a pilot study. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, UbiComp '16, page 1200–1208, New York, NY, USA, 2016. ACM.

[42] Aku Visuri, Niels [van Berkel], Tadashi Okoshi, Jorge Goncalves, and Vassilis Kostakos. Understanding smartphone notifications' user interactions and content importance. *International Journal of Human-Computer Studies*, 128:72 – 85, 2019.

[43] Rui Wang, Min S. H. Aung, Saeed Abdullah, Rachel Brian, Andrew T. Campbell, Tanzeem Choudhury, Marta Hauser, John Kane, Michael Merrill, Emily A. Scherer, and et al. Crosscheck: Toward passive sensing and detection of mental health changes in people with schizophrenia. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '16, page 886–897, New York, NY, USA, 2016. ACM.

[44] 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*, UbiComp '14, page 3–14, New York, NY, USA, 2014. ACM.

[45] Tilo Westermann, Ina Wechsung, and Sebastian Möller. Smartphone notifications in context: A case study on receptivity by the example of an advertising service. In *Proceedings of the CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 2355–2361. ACM, 2016.