Reachable but not Receptive: Enhancing Smartphone Interruptibility Prediction by Modelling the Extent of User Engagement with Notifications

Liam D. Turner*, Stuart M. Allen, Roger M. Whitaker

School of Computer Science & Informatics, Cardiff University, Cardiff

Abstract

Smartphone notifications frequently interrupt our daily lives, often at inopportune moments. We propose the decision-on-information-gain model, which extends the existing data collection convention to capture a range of interruptibility behaviour implicitly. Through a six-month in-the-wild study of 11,346 notifications, we find that this approach captures up to 125% more interruptibility cases. Secondly, we find different correlating contextual features for different behaviour using the approach and find that predictive models can be built with >80% precision for most users. However we note discrepancies in performance across labelling, training, and evaluation methods, creating design considerations for future systems.

Keywords: Interruptibility, notifications, human behaviour, smartphone, mobile

1. Introduction

The relationship between a human and their smartphone has reached the point where it forms a cognitive extension [1] and offers deep insight into human behaviour (e.g., [2]). Smartphone notifications interrupt us through audio and visual cues for a wide variety of reasons; to inform, persuade, or prompt for a reaction. As more and more applications compete for our attention, the cognitive burden on the user to manage their interruptibility for different notifications clearly increases. While mechanisms exist for controlling notifications (e.g. silent mode), managing these can be complex to set up and maintain, as the user needs to consciously reflect on their own behaviour and preferences. Consequently, intelligent systems to pro-actively assess the nature and extent of our interruptibility are therefore highly desirable for both the user and interrupting applications [3].

*Corresponding author

Email addresses: TurnerL9@cardiff.ac.uk (Liam D. Turner), AllenSM@cardiff.ac.uk (Stuart M. Allen), WhitakerRM@cardiff.ac.uk (Roger M. Whitaker)

Diverse approaches have been proposed to facilitate intelligent interruption for the smartphone [4], across phone calls [5, 6] and notifications [7]. This has included determining the influence of contextual factors [8, 9], exploring methods of labelling interruptibility [10, 11, 12] and training predictive models [13, 7]. However, interruption is challenging to observe in isolation because interrupting the user to ask how interruptible they are is itself an interruption. In doing so, this simplification does not consider the per-application variability in: notification content and purpose [9]; the extent a response can be made (no response, partial, or complete) [12]; or the subjectiveness in what response behaviour signifies a successful interruption [4]. This motivates modelling interruptibility with enhanced granularity.

To achieve this we explore prediction based on the human decision making process that occurs in response to a notification, specifically modelling the extent to which they respond after they are initially interrupted and then provided with more information. Referred to as the *decision-on-information-gain* (DOIG) model, this approach extends the existing convention to label response behaviour from how a response is made (as well as if) and without the reliance on surveys (e.g., [14, 7, 9]). As noted in previous work [12], this approach synthesises multiple definitions of interruptibility to enable flexibility in how a user's interruptibility is defined, removing the rigidity of the single definitions used in previous studies (e.g., [15, 16, 14]). When combined with smartphone sensor data, the DOIG model provides a basis for prediction that is flexible to the priorities of different applications. We explore this through:

- an in-the-wild study that captured user responses to notifications implicitly, using the DOIG model;
- examining the extent of interruptibility misclassification using the existing convention for the 11,346 notifications collected, by considering the impact of partial user responses that are captured;
- determining the extent to which prediction of different user responses can be made using the DOIG model, along with different training and evaluation methods.

Collectively, these results justify the use of the DOIG model as a useful framework for collecting and labelling interruption behaviour that can influence the design of intelligent notification components within individual applications.

2. Background

Interruptibility studies have historically focused on communication prompts, such as phone calls (e.g., [13]), email (e.g., [17]), or instant messages (e.g., [18]) and often focus on particular locations (e.g., the workplace [19]). A recent systemic survey [4] has revealed a diverse range of methodologies for representing influential factors (e.g., [20, 8]), collecting relevant contextual features (e.g., [21, 7]), and enabling prediction through machine learning (e.g., [22, 19, 13]). However a key finding was that interruption management for technology remains at a formative stage, with widespread alternative approaches, scenarios and assumptions [4].

Studies have also used a broad spectrum of different interruptions and environments. Some studies create particular definitions of interruptibility relevant to a scenario [23], such as finding breakpoints in PC work tasks (e.g., [24, 25, 26]), while others focus on specific response behaviour, such as whether the user is attentive [7, 18], or receptive to the content [27]. This indicates the challenge of achieving generality in this field; consequently we believe that approaches supporting the variability in studying interruptibility to be timely and valuable - which motivates our investigation.

2.1. Representing interruptions and response behaviour

Representing interruption behaviour is commonly achieved through labelling contextual data, with the label derived from a user's response behaviour to an interruption. The majority of empirical studies rely on the user completing an explicit survey (e.g. [7]) or undertaking an implicit labelling task (e.g. answering a phone call [13]). This approach assumes that if the user is interruptible, then they will complete the labelling task, and often results in the counter intuitive approach of interrupting the user to ask how interruptible they are. As a result, these approaches can be grouped together [4] as a *black-box* model [12], where the focus is on completing a specific end-goal behaviour that denotes interruptibility. While this approach is useful in that it can be wrapped around any interruption, it under-represents scenarios where the user has choice and degrees of freedom in how they respond.

A key early contribution that sought to understand the different ways in which users might handle interruptions arose from the work of McFarlane and Latorella [28]. This involved proposing an abstract representation of the interruption process for machine-to-human interactions (Interruption Management Stage Model), which adopts a series of decision-making steps. This general approach is highly relevant to smartphone notifications because it allows us to

model the choices that a user makes in response to an interruption. However, to the best of our knowledge it has not been previously explored in the context of smartphones [4, 12].

In parallel, the improved sensing and computational capabilities of the smartphone have enabled variety and richness in observable data [29, 30] in comparison to experimentation within controlled environments (e.g., [31]). Despite this, we note that human annotation of smartphone interruptions is widely used (e.g. [7]). However there is an increasing opportunity to implicitly sense and report context that is aligned with an interruption [4] and the environment it occurs in. This is not only in terms of potentially relevant features on which to base predictions, but also in implicitly capturing response behaviour for labelling interruptibility.

2.2. Predicting notification interruptibility

Studies predicting interruptibility using machine learning typically use a single definition of interruptibility, a single point in the response deemed the *measure of success*, and a priority of minimising either false-positives or false negatives [4]. Additionally, degrees of freedom exist concerning pre-processing, training, and evaluation, with limited direct comparison of different choices.

Training and testing methods have historically involved an offline learning environment using the aggregated data of all users (e.g. [14, 12]). It is common for multiple classifiers to be explored through a bottom-up approach (e.g., [13, 12]), although some studies use a single classifier (e.g., [14]). This has resulted in a range of classifiers being identified as the most suitable [4], including: naïve Bayes [19], J48/C4.5 trees [14, 32], and Association Rules [13]. Some approaches extend this to reduce model complexity in the quantity of training data (e.g., [11]) or feature vector size (e.g., [31]). While results of previous studies have been promising in achieving high accuracies, the tight coupling between the specific study environment and the conclusions made create challenges for benchmarking between studies.

Recent works have also included analysis of other training methods, such as online learning, where models are re-trained periodically (e.g., [13]). Additionally, building personalised models rather than from aggregated data has also been a recent focus, from the hypothesis that individual interruption habits are non-uniform (e.g., [7, 9]). Typically, previous works have focused on exploring one or the other (e.g., [13]) and only a few recent studies compare variations in these components together (e.g., [9]). However, we note that generally these analyses do not extend to different definitions of interruptibility or evaluation metrics.

Evaluation is typically performed using standardised metrics, including: precision and recall (e.g., [12]), specificity and sensitivity (e.g., [9]), F-measure (e.g., [33]), Kappa statistics (e.g., [14]), or Area Under Curve values (e.g., [34, 14]). Whilst the suitability of these criteria has been debated in the wider area of machine learning (e.g., [35]), their suitability for interruptibility arguably has an additional layer of complexity [4]. For example, a hypothetical application may class ineffective interruptions as the most important to minimise (i.e., false positives), whereas another may class missed opportunities as the most important (i.e., false negatives). We observe that, generally, both cases are not considered together within studies, contributing to the challenges of determining the most widely applicable techniques.

3. Facilitating multiple interruptibility definitions through observing user response behaviour

We introduce a framework to capture response behaviour to interruptions called the *decision-on-information-gain* model. It extends the existing black-box [12] convention by decomposing how a response is made using the interactions performed on the device after an interruption is issued, rather than only if the full notification content is consumed. Whilst the focus of this paper is on Android smartphone notifications, the concept can be generalised, consistent with the previous work by McFarlane and Latorella [28]. Their proposed model suggests that an interruption triggers a linear task rescheduling process made up of a series of micro-decisions underneath the larger decision to begin responding or not. Building on this, we suggest that these micro-decisions extend into the response to the interruption itself as the user gains more information. This differs from the typical convention of empirical studies, which assume that a user will either respond fully after deciding to start, or not start at all. [4].

3.1. Decision-on-information-gain (DOIG) model

The DOIG model follows the decisions a user *must* engage with (either consciously or subconsciously) in response to a notification. The initial decision is whether to switch focus after being prompted. Subsequently there are k points where extra information is provided (such as the identity of the interrupter or the subject topic). This produces a set of $k+1$ sequential decisions that are required for a complete response to the interruption, D=$\{d_1, d_2, ...d_{k+1}\}$ - where decision d_i precedes d_{i+1}. These represent possible conscious or subconscious decisions made by the user when interacting with the device in order to retrieve more information about the interruption. While the exact number of decisions may vary based on the interruption characteristics, we propose that a decision will occur each time the user

is given new information as they respond. It is important to note that this approach intends to observe the natural decisions that are already being made and that this does not change the response process in any way.

A sub-sequence $\{d_1, d_2, ...d_i\}$, where $i \leq (k+1)$, captures the extent of the users response, with d_i indicating the exit decision. In comparison, a black-box approach [4, 12] assumes that for an interruption to be successful, a complete response must be performed, that is while all decision steps $d_1, \ldots, d_{k+1}$ are assumed to be carried out, only the final decision d_{k+1} is assessed. Consequently the black-box approach is inherently susceptible to under-representing the choices that a user makes during the response as they are presented with more information about the interruption. This is particularly useful for applications that can consider an interruption to be successful at an earlier decision than d_{k+1}, i.e. a partial response where the notification is noticed but not consumed.

3.2. Applying the DOIG model for Android notifications

The focus of this study is on notifications that are provided through the Android operation system. The nature of Android notifications requires the user to discover information about a notification in stages. This enables the user to make decisions on whether to continue on towards consuming a notification, or abandon the response part way through. Rather than making an assumption on what point in the response behaviour correctly signifies being interruptible (i.e. the measure of success), which will likely change on a per-application basis, we map the DOIG model to a range of possible responses that can be expected (Figure 1), these are:

- **Null Responses** - Cases where the user does not show any observable response behaviour, either because the user was not physically interrupted or did not want to switch tasks for any notification, from any application.
- **Partial Responses** - Cases where the user begins to respond, but abandons after further information. For example, they interact with the smartphone, discover the notification relates to an email but exit at that point (or after reading the sender or subject).
- **Complete Responses** - Cases where the user consumes the notification and completes a response. For example, tapping on the notification and reading an email or filling in a survey.

Given that a response can be null, partial or complete, the potential measures for success can be defined as whether the user is *reachable* [12], willing to *engage* [12] to some extent, or is *receptive* [27, 12, 9] to what they are interrupted

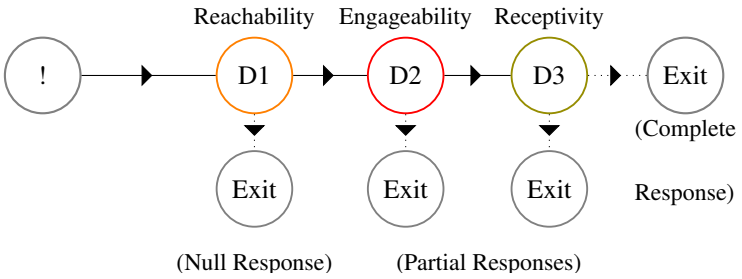


Figure 1: A visualisation of the linear sequence of decisions made during notification response. After the interruption occurs (!), at each point new information is given (e.g. the application icon) the user must decide (e.g., D1) whether to continue on to the next decision (e.g. D2), (up until either the notification is consumed) or exit at a particular decision.

with, from this we define:

- **Reachability** indicates whether a response will at least be started, or not (i.e. not null)
- **Engage-ability** indicates whether a response will be started but abandoned without consuming the notification (i.e. partial), either because the notification summary is sufficient, or it is undesirable to pursue it further.
- **Receptivity** indicates whether the user is receptive to the notification content and consumes it (i.e. complete).

Modelling a range of response behaviour (as shown in Figure 1) means that diverse interruption characteristics of smartphone applications can be accounted for. It may be that an application considers a notification to be a success if the user was reachable (i.e. the response is not null), such as reminders. Whereas others may require the user to reach a specific later stage in the response (i.e. at least engage-able), or consume it completely and open the application (i.e. receptive). These three independent measures fit together under the wider umbrella of notification interruptibility, providing flexibility for labelling interruption behaviour on a per-application basis. This is contrary to the wider research space, which typically predicts using a single measure of success (e.g., just receptivity [27]) and relies on the user to open the interrupting application (e.g., [18]) or fill in a survey (e.g., [7, 14]) to label their interruptibility.

3.2.1. Limitations and flexibility in applying the model for Android

Due to technical restrictions imposed by the Android operating system, some relevant UI events (e.g. accessing the Notification Drawer) are not observable by third party applications without privacy-sensitive Accessibility permissions. This limits which decisions are observable, particularly when the device is in-use. If the device is not-in-use when the

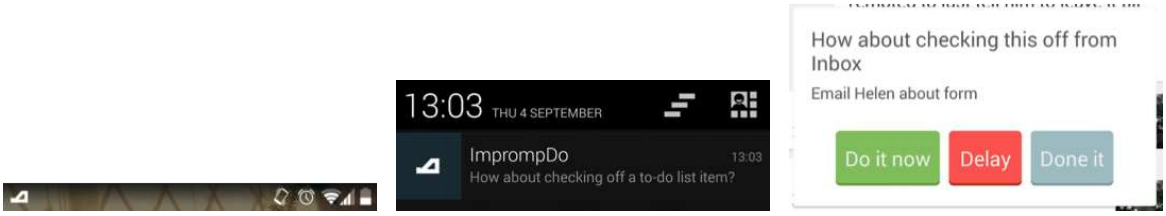
notification is delivered, we can observe decisions being made through the process of the user turning the screen on and unlocking the device [12]. If the device is already in-use, the same sequential decision process occurs; however this results in no observable system events for D1 and D2 currently (for most applications). Nevertheless, the DOIG model is robust to future changes to the Android operating system where further decision behaviour (i.e., the number of decisions in D) could be explored through new or adapted APIs.

It is important to note that the example provided and visualised in Figure 1 represents a typical Android notification. Whilst the notification convention is a standardised and imposes design constraints, some variability remains in what information can be presented, when, and how by individual applications. As a result notifications that deviate from the default configuration may change what decisions are individually observable. For example, D1 and D2 may be merged if the tone used for interruption is distinguishable for a given application. Additionally, other smartphone operating systems (such as iOS) have slightly different implementations of notifications. However, as we wish to observe and not change how a notification is presented and responded to, these additional constraints require a flexible model, which the DOIG model allows through a variable number of abstract decisions.

4. In-the-wild study: ImprompDo

To observe whether the DOIG model brings a useful utility in capturing and representing response behaviour towards Android notifications, we developed a bespoke Android application, called *ImprompDo*, that captures context data and response behaviour to notifications in-the-wild. The application was distributed publicly through the Google Play Store, for devices running Android 4.0 to 4.4 (inclusive), which covered 85%-94% of the market distribution at the time of the study. After a process of informed consent, on-going anonymous participation is incentivised through the facilitating a useful role as a productivity tool. This aimed to promote natural behaviour rather than relying on volunteers that were willing to be interrupted, as seen in previous studies (e.g., [9]).

The case study represents a real world application where an intelligent interruption system would be suitable, which has been a common design choice of similar empirical studies in the area, including mood diaries (e.g., [14]), instant message communications (e.g., [15]), or news stories and weather updates (e.g., [9]). Ideally, a dataset should contain response behaviour that represents all possible notifications. However, in reality notifications are diverse in design and



(a) The application icon shown for an example ImprompDo notification

(b) The Notification Drawer showing an ImprompDo notification

(c) The application content shown if the ImprompDo notification is consumed

Figure 2: The Android notification response process used

purpose, and experimenting a one-size-fits-all notification would not be possible beyond a controlled research study. Interrupting the user without a purpose in an effort to be more generic would make the notification unrepresentative of all practical scenarios. To circumvent this, we perform our analyses using different independent measures of success in the response, to represent the spectrum of requirements other applications may have as far as possible.

4.1. Interrupting notifications

Each user is interrupted several times based on one of four randomly selected triggers, inspired from conclusions of related works (e.g. [23, 36]). These are: at a random time; at the end of a period of acceleration; an X in 10 chance to occur at a random time, where X increments or decrements each time a notification in that hour on previous days is consumed or not; and a binary Logistic Regression model trained from whether notifications were fully consumed in similar contexts in the previous seven days. Notifications used the device's default tone, vibration pattern and visual cues, while adhering to the device's global volume settings at the time. If the user is interrupted, they respond in the same way as any other Android notification (Figures 2a through 2c). That is, assuming the user decides to continue at every micro-decision, they turn on the screen, unlock the device (unless it is already in-use), access the notification drawer and tap on or dismiss the notification. The user is then presented with a random to-do item and buttons to manage it.

As we are focused on near-real time interruptibility, we remove the notification after 30 seconds if the user did not consume the notification. This allows us to assess the immediate interruptibility of the user in various contexts and minimise a response being the result of a coincidental interaction with the device at a later time.

4.2. Implicit data collection alongside notifications

As notifications are delivered we adopt the use of sensors and software APIs on the device that are sampled using an intermittent background service. This *implicit sampling* provides an in situ representation of the smartphone and environment and a trace of how the user naturally interacted with the device in response to the notification. These data traces are used to create the feature vectors and different interruptibility labels (representing the different interruptibility definitions). In comparison to previous studies this removes limitations such as: relying on the user to provide information and labelling through surveys (e.g., [7]); permissions that are privacy invasive and out-of-place for most applications (e.g. [37]) and needing persistent monitoring of device state changes (e.g., [18]).

To gain contextual data for prediction, we sample a wide variety of different data sources available on the device. However to maintain wider applicability beyond the scope of this study application, we chose data sources that: are present on the majority of devices; do not require additional privacy invasive permissions that would not be suitable for most applications (e.g. microphone, location, calendar), which may also introduce a behavioural bias even if the user accepts them [38]; require persistent monitoring of the device (e.g. device usage data and detailed activity recognition [18]); or require a fundamental change to how a user interacts with an application (e.g. in needing to answer surveys [7, 9]). As a result, we collected data from the: accelerometer (linear acceleration and gravity pseudo-sensors), light sensor, proximity sensor, battery charging state, screen on/off state, lock state, global volume state and the current timestamp. These data sources have also been used successfully in previous interruptibility studies [4].

All data sources are sampled starting 5 seconds before the interruption is scheduled until either 30 seconds have elapsed or the notification is consumed/dismissed (D3, as shown in Figure 1). The sampling consists of taking sets of raw data vectors, each containing a reading from all data sources. As readings are delivered by Android asynchronously, a window is opened to listen for data. It is closed when either at least a single reading is collected from all data sources or a timeout of 2 seconds has elapsed. Only the most recent readings are then retained so that the variance in reading times is minimised. If no data readings were available after 2 seconds, the reading for that sensor/API is set to null. A new sampling window is then opened immediately, subject to device speed and system stability. To create features from the raw data, mean values were used across all readings that occurred within the period prior to the interruption.

For the response behaviour, the indicators of the micro-decisions being made are determined by changes in the

screen state, lock state, and notification interaction events; which occur as a by-product of the user conducting the response to each notification (as described in Section 3.2). The decisions that are able to be captured through interaction with the smartphone vary depending on whether the device is in-use or not at the time the interruption occurred. To determine this, readings from the screen state API are used; if the screen was off we deem the device to be not-in-use, otherwise it is considered in-use. As data readings naturally occurred at irregular intervals, we determined this from the reading taken closest to the time of the interruption, within $\pm.5$ seconds.

4.3. Dataset

The dataset contains 11,346 Android notifications, each with an associated set of raw data vectors containing sensor and API data. This was collected over 178 days between July 2014 and January 2015, with 224 participants installing the application over the period and 93 (41.5%) providing data for at least 1 notification - producing a relatively large population in comparison to similar studies [4]. Participants used the application for an average of 26.457 days ($\text{Min} = 1, \text{Max} = 129.624, \text{SD} = 35.633$), received an average of 122 notifications ($\text{Min} = 1, \text{Max} = 781, \text{SD} = 175.325$), with each notification having an average of 65.269 data vectors ($\text{Min} = 0, \text{Max} = 840, \text{SD} = 7.564$). As fewer decisions can be observed if the device was in-use than not, we split the data into two groups. However, we could not determine the in-use state for 1267 notifications (11.2%), which were then excluded from the analysis.

4.4. Examining the benefit of the DOIG model against the black-box convention

We hypothesise that extending the black-box approach captures additional useful information. To measure this we compare the number of cases where the user at least partially responds to the notification (i.e. the user is reachable, engage-able, or receptive), against those that would be captured in a black-box approach (i.e. receptive only). A response is considered partial if the user progresses past D1 (reachable) but does not consume the notification (not receptive), i.e. it either expires or it is dismissed. A black-box approach would typically only capture complete receptive responses (i.e. they pass D3), where a notification is tapped upon (and potentially a survey is completed [7]). However, it should be noted this could also include other notification interactions, such as if it is dismissed rather than being tapped on. Therefore we conduct our analysis for cases that include dismissals and those which do not.

The results show that 1317/10059 (13.1%) of all cases were partial responses if dismissals are included, or 802/10059 (8%) if not. These cases would be missed by a commonly used black-box approach, which would misclassify these cases as the same as a null-response (i.e., not reachable). By combining partial responses and complete responses, the total number of cases where at least some degree of interruptibility was shown increases from 1056 with a black-box approach to 2373 with the DOIG model if dismissals are considered as partial responses - a substantial 124.7% increase. Alternatively, if dismissals are captured by a black-box approach, this increases the total from 1571 to 2373, a 51.1% increase. These results show that using the DOIG model to capture user interactions with the device, and subsequently observe the micro-decisions being made, is suitable at isolating responses that are: not started, i.e. null (unreachable) responses; those that are started but abandoned, i.e. partial (engaged) responses; and those which consume the notification, i.e. complete (receptive) responses.

From a usability standpoint, this suggests that observing the response process using the DOIG model is more worthwhile for applications rather than solely relying on whether notifications are consumed. For example, the ImprompDo application represents a use case where knowing that the user was at least reached is useful, as this is indicative that the user made a decision regarding their productivity. This is not exclusive to to-do list applications and applies to other applications which issue single purpose notifications (e.g., in hydration or exercise reminders) where merely seeing that a notification has arrived may have the desired affect, even if the notification is then not consumed.

For other applications which require the user to completely consume the notification to be considered successful, the DOIG model still provides a useful utility in being able to distinguish between cases where the user did not respond at all and those where they partially responded (i.e., they were at least reachable). From a practical standpoint, the data collection application itself serves as evidence that implicit observation using the DOIG model is feasible, and does not require privacy sensitive permissions or a persistent background service, which has commonly been used previously (e.g., [18]). We move forward with investigations into whether different responses can be isolated through different contextual data, and are subsequently predictable.

4.5. Correlations between contextual data and reachability, engage-ability, and receptivity

A hypothetical application will choose an interruptibility label from each use-state to measure the success of their notifications with (e.g. reachability). Table 1 shows which contextual variables are correlated with which labels,

Table 1: P-values indicating significance of each feature before the interruption and the outcome of each decision. Bold values show significance using $p < .05$. * Mann-Whitney U Test ** Kruskal-Wallis 1-way ANOVA.

	Not in-use			**In use**
Feature variables	**Reachability**	**Engage-ability**	**Receptivity**	**Receptivity**
Accelerating* (False, True)	.186	.458	.072	**.000**
Ambient Light** (Dark, Dim, Light, Bright)	**.000**	**.039**	**.000**	**.000**
Screen Covered* (False, True)	**.000**	.187	**.000**	**.005**
Volume State** (Silent, Vibrate, Audible)	**.000**	**.009**	**.011**	**.000**
Orientation** (Flat, Upright, Other)	**.000**	.098	**.000**	**.000**
Charging State* (False, True)	**.000**	**.001**	.145	.177
Time of Day**(Morning, Afternoon, Evening, Night)	**.002**	.125	.936	**.000**
Day of the Week**	.509	.794	.100	**.000**
Number of cases (n)	7737	1798	1469	2322

determined from whether the differences in the underlying distributions are statistically significant. Initial inspection reveals that some features are only correlated for some labels and these differences also extend between whether the device is in-use. From this we can suggest that different contextual data may be (consciously or subconsciously) relevant to the user's decision behaviour in their response, indicating the potential for predictability of different definitions of interruptibility using implicitly sampled data.

While correlation does not imply causation, closer inspection of individual variables reveals logically possible effects. For example, the "Volume State" is significant for reachability when not-in-use ($\chi^2(2, 7737) = 202.209, p < .001$). This is expected, as this is a common mechanism to control physical interruptions from the device. Pairwise post-hoc tests reflect this, with statistical significance shown for silent and audible ($p < .001, r = -.170$), silent and vibrate ($p < .001, r = -.242$) pairs. Analysis of the affect size supports this further with a medium strength for both. Furthermore, the difference between vibrate and audible is also significant, but with a much smaller affect size ($p < .003, r = .040$). Interestingly, despite the design of the vibration setting intending to lessen the impact of an

interruption, which is arguably closer to silent mode, in practice the affect size shows that user behaviour towards interruptions through vibrations patterns is more similar to audible tones.

A further example is "Orientation" being significant when the device is in-use for receptivity ($\chi^2(2,2141) = 20.924, p < .001$). Pairwise post-hoc tests revealed the significance pairs to be between groups where the device was flat and those when upright ($p < .001, r = -.087$), and between other orientations and upright ($p < .001, r = .145$). It could be assumed that when a device is being used for active interaction, it will likely be relatively upright in the user's hand, whereas other positions (such as when unlocked flat on a table) may produce false positives. This is reflected in the p-values and affect sizes of these pairwise comparisons, and further supported by the difference between flat and other orientation groups not being significant. This suggests that a multi-modal approach, using measures in addition to the screen state, could be used to determine whether the device is in-use in the future.

Other variables have more unexpected outcomes, for example, whether the device is "Accelerating" is significant when the device is in-use ($U = 482,548, p < .001, z = 3.788, r = .082$) but not when not-in-use. This is unexpected as if the device is already in-use, it could be assumed that the user would be more attentive to notifications, regardless of whether they were accelerating. However, this could be explained by the level of focus the user has on an important task when the device is in-use. The same argument concerning the current task being performed could also apply to other variables when the device is in-use. For example "Screen Covered" ($U = 147,285, p < .005, z = -2.815, r = -0.063$) "Ambient Light" ($\chi^2(2,2138) = 20.463, p < .001$), and "Volume State" ($\chi^2(2,2322) = 25.316, p < .001$) are all statistically significant, however for only a subset of pairs within these (e.g. Dark and Dim ($p < .001, r = -.1$), and Dark and Light ($p < .004, r = -.092$)). Across these the affect size was low, suggesting that the significance may due to cases where the device was not in active use, but the screen remained on.

The significance of temporal variables also differs across the use-states. Firstly, the "Time of Day" was significant for receptivity when the device is in-use ($\chi^2(3,2322) = 27.008, p < .001$), with pairwise-tests revealing the difference between Morning and the other groups having the highest affect size (Afternoon ($p < .004, r = -.083$), Evening ($p < .028, r = -.085$), Night ($p < .001, r = -.154$)). This suggests that when the device is in-use in the morning, users are typically focused on their current task and are less susceptible to interruption from notifications. Finally, the "Day of the Week" is also significant for D3 when the device is in-use ($\chi^2(6,2322) = 24.191, p < .001$), but with only a few

Table 2: P-values for whether a relationship exists between the mean value each feature has in the readings taken between decisions and subsequent later decisions. Bold values show significant differences ($p < .05$). * Mann-Whitney U Test ** Kruskal-Wallis 1-way ANOVA. Rc=Reachability, Eg=Engage-ability, Rv = Receptivity.

	Not in-use						**In use**
Context between:	**(Interruption-D1)**			**(D1-D2)**		**(D2-D3)**	**(Interruption-D3)**
Correlates with the outcome of:	**Rc**	**Eg**	**Rv**	**Eg**	**Rv**	**Rv**	**Rv**
Accelerating*	**.000**	.890	.676	**.000**	.064	**.000**	**.000**
Ambient Light**	**.000**	.157	**.000**	**.000**	**.009**	**.013**	**.000**
Screen Covered*	**.000**	.079	**.000**	**.000**	.287	**.001**	**.000**
Volume State**	**.000**	**.007**	**.008**	**.002**	.112	**.003**	**.000**
Orientation**	**.000**	.247	**.000**	**.017**	**.000**	**.001**	**.000**
Charging State*	**.000**	**.005**	.314	**.001**	.055	**.046**	.231

significant pairs and low affect sizes.

In summary, these results suggest that different contextual variables before the interruption may be influential on the decision-making process in response to notifications, even when the user's focus is already on the device. This can be further supported by similar findings in the contexts after the interruption (Table 2), suggesting that different sets of contexts may influence the expected response behaviour. Going forward, while the different significant features and various affect sizes suggest predictability, we explore the expected performances of reachability, engage-ability, and receptivity models with various machine learning methods.

5. Predicting reachability, engage-ability, and receptivity

In this section we explore the extent to which reachability, engage-ability, and receptivity are predictable. Our analysis is structured as follows. Firstly, we explore the performance of a typical user from the entire dataset in an offline setting and compare this against personalised models for each individual's data. We then compare the performance against existing Android conventions, in order to determine whether this personalisation is worthwhile. Finally, we examine an online learning setting where models are retrained at the end of each day with new experiences.

5.1. Machine learning approach

We have used machine learning [39] to investigate the prediction performance of reachability, engage-ability, and receptivity models, using the following strategy:

Pre-processing - Analysis of the dataset reveals that the class (label) distribution is imbalanced since the majority of notifications are null-responses (i.e., users were unreachable). Without pre-processing, this could lead to false reporting in model performance, for example, if a model always predicts a single class and 80% of the data is labelled with that class, then the model is trivially correct 80% of the time, but practically useless. To prevent this, random-under-sampling (RUS) [35] is used to produce 100 evenly distributed datasets for each model. As a result of this, some users may have too few resulting data-points to build personalised models, to avoid misrepresenting performance we remove these users where relevant.

Classification - Numerous classification algorithms have been used across similar studies, with little agreement on the most suitable [4]. Previous work using the ImprompDo dataset and the DOIG model [12] revealed minimal performance differences across various Bayesian, tree, and function based classifiers. We used a J48 tree (C4.5) as it offers several advantages beyond performance. Firstly, it is easily interpretable and has been used successfully in similar studies (e.g., [14, 32]). Secondly, models created for when the device is in-use and not can be merged together by adding a top-level node (i.e. in-use? $\{true, false\}$), rather than managing multiple models. Finally, storage and traversal of the tree is computationally inexpensive, an important factor for smartphones with limited resources.

Training and testing - For each observable measure of interruption success, we adopted three approaches to splitting the data: Aggregate Trained and Aggregate Tested (AT-AT) where training and testing data is split from the same aggregated dataset from all users; Aggregate Trained and Personally Tested (AT-PT) where for each user, the models are trained from the data of all other users, and tested only against that selected users data; and Personally Trained and Personally Tested (PT-PT) where training and testing data are both from the data of each individual user. However, as participation levels of individual users varied, some users may not have data for all classes, such as if no notifications occurred when the device was in-use; these users are excluded where relevant. For testing our models we used 10-fold cross-validation on the AT-AT and PT-PT models. As AT-PT models use separate training and testing datasets, cross-validation would not be suitable. However, as the above analysis is performed on 100 RUS datasets (as defined in

pre-processing), this mitigates this issue.

Performance evaluation - Different applications may have different perspectives on the overall suitability of a predictive model. For example, a decision-making system mediating interruptions on behalf of the user may consider interruption cost (false positives) to be the most important to minimise, whereas a decision-making component in an interrupting application may consider missed opportunity cost to be as important (false negatives). To consider this, we evaluate each of our independent predictive models using two groups of standardised metrics, which are derived from the confusion matrix produced in the evaluation:

- **PPV and Sensitivity**: The positive predictive value (PPV) and sensitivity values refer to the precision and recall metrics for binary classification, where we are interested in performance for cases that should be predicted as "true", i.e. the proportion of cases that were correctly classified as reachable, engage-able, or receptive, and the proportion of cases that were correctly identified against the total number of cases that exist respectively.
- **NPV and Specificity**: The negative predictive value (NPV) and specificity refer to the precision and recall metrics where we are interested in performance for cases that should be predicted as "false", i.e. the proportion of cases that were correctly classified as not reachable, not engage-able, or not receptive, and the proportion of these cases that were correctly identified against the total number of cases that exist respectively.

Applications that wish to avoid missed opportunities to interrupt will likely focus on PPV and sensitivity. Applications wishing to avoid interrupting during ineffective moments (i.e. the user wont likely produce the desired response behaviour) will focus on NPV and specificity.

6. Results: How well can response behaviour be predicted?

We first investigate how well reachability, engage-ability and receptivity can be predicted for a typical user, using the aggregated dataset from all users. The results, shown in Table 3, extend our analysis of different correlating features (Section 4.5), by showing that despite these differences, each measure of success is reasonably predictable across all metrics. While the mean performance is not very high, as the participation of users varied (and likely their individual

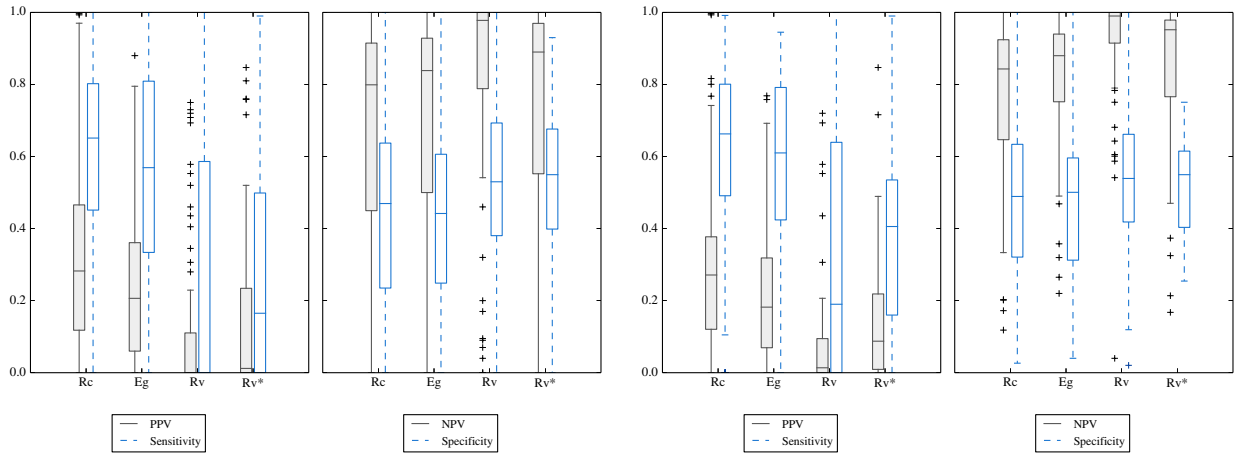
	Device Not In use			Device In Use
Metric	**Reachability (Rc)**	**Engage-ability (Eg)**	**Receptivity (Rv)**	**Receptivity (Rv)**
PPV	0.586	0.582	0.617	0.594
Sensitivity	0.699	0.677	0.684	0.610
NPV	0.627	0.614	0.646	0.600
Specificity	0.505	0.514	0.576	0.582

Table 3: Classifier performance (J48) of the aggregated dataset (AT-AT), using models with different measures of interruption success.

interruption habits) this is not unexpected. The performance is also similar to other recent studies (e.g., [7, 16, 40]), including those inferring interruptibility from content data over context (e.g., [9]) and other attentive states (e.g., [41]).

Closer inspection of the metrics reveals further patterns. Firstly, the predictive models offer higher precision in avoiding untimely interruptions (NPV) than finding opportunities (PPV), suggesting that correctly identifying interruptible moments is more challenging, at least for one-size-fits-all models from aggregated data; however the reverse is true for identifying all of these cases (specificity and sensitivity). Secondly, for cases where the device is not-in-use, performance typically increases for the measures of success that correspond to later points in the response. This suggests that context, as well as content [9], is a factor that affects the receptivity towards the interruption. Another unexpected result is the worse performance for receptivity when the device is in-use. This could be explained by the unknown level of engagement that the user had with their device at that time, with task engagement previously been shown to be an additional influential factor [32, 42, 26].

The results provide an indication of the expected performance of a one-size-fits-all model built from the aggregated data of all users. However, as individual users in the ImprompDo dataset participated for different periods of time, experienced different contexts, and likely have their own interruption habits, this model may not be representative of every user. We move forward by determining whether this one-size-fits-all model performance is actually reflective of the performance that would be experienced for individual users in the dataset, and how this compares against building personalised models for each user.



(a) All users. Reachability (N=92), Engage-ability(N=92), Receptivity (not-in-use: N=92, in-use: N=83).

(b) Users with >10 notifications. Reachability (N=63), Engage-ability(N=63), Receptivity (not-in-use: N=63, in-use: N=41).

Figure 3: Distribution of user performance for models trained from aggregate data (AT-PT)

6.1. Aggregate vs personalised predictive models

We extend the analysis to explore whether the performance of our typical user model is representative of the real world; where user participation would be self-selecting and level of engagement would vary. To investigate the potential effects of this, we build separate models for testing each user's data individually. As well as testing at an individual level, a hypothetical application will have to decide what data to train from. While personalised models of interruptibility have previously been successful [43, 7], the associated computational, temporal, and storage overheads in collecting data and training models may outweigh performance benefits, on a per-application basis. We therefore conduct our analysis with both aggregated and personalised models.

6.1.1. Training from aggregate data (AT-PT)

The first set of models were built where, for each user, the training data consists of the aggregated data of all other users, with the selected user's data used as testing data. This enables us to simulate the performance of new users installing the application where a set of training data from other users already exists. Figure 3a shows the distribution in performance across all individual users and Figure 3b shows results only for more active users (i.e., with >10 notifications). We focus our analysis on the pruned dataset, as while the effect on the overall distribution and medians

is low, this removes outlier performances at the lower and higher quartiles.

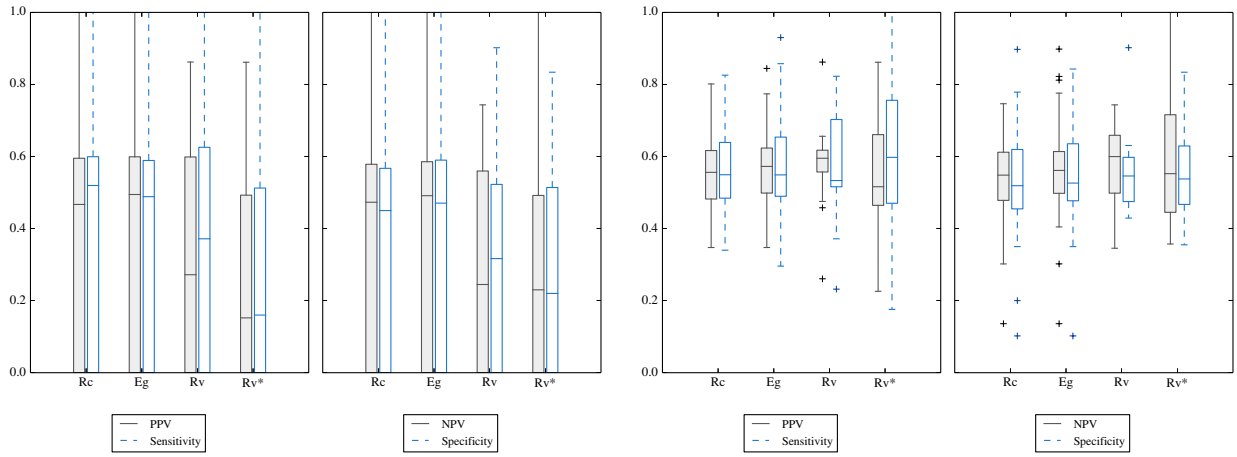
The results show that models trained from aggregated data perform very well at correctly predicting that the user is not reachable, willing to engage, or receptive (NPV) for most users (>.80), with receptivity also having much smaller variance. However, these models perform worse at correctly predicting opportune moments (PPV) for most users, across all measures of success. This suggests that individual users are likely to be interruptible in very different contexts, whereas users are not interruptible in similar contexts; which is logical such as during driving. For the recall metrics (sensitivity and specificity), the median performances are close to our typical user model (Table 3) for reachability and engage-ability (and similar studies, e.g. [7]), with the exception of sensitivity for receptivity; however the variance across users is generally high.

In comparison with our one-size-fits-all typical user (Section 6), the results highlight the diversity in interruption habits across users, suggesting that the typical user model predominantly either underestimates or overestimates per-user performance. The suitability of training from an aggregated dataset is therefore largely dependent on whether an application's desired error priority is to avoid missed opportunities or ineffective interruptions. From the perspective of an application like the to-do list data collection application, where reachability is the likely label and the priority is finding opportunities to prompt for productivity, this suggests that an aggregated model may not be suitable. Nevertheless being able to correctly predict the inverse, that the user is not reachable, could still be useful.

6.1.2. Training from personal data (PT-PT)

The second set of models were trained and tested only using each user's individual data. Figure 4a shows the performance of all users and Figure 4b shows only those with >10 notifications. In this case, the pruning operation reduces the variance across users considerably. As users experienced various contexts naturally, this could be explained by some contexts not being experienced frequently, which could lead to model defects such as *concept drift* [13]. To avoid under or over representing performance, we removed users that produced models for only a single class (i.e. they were always receptive or not) after the pre-processing and pruning operations were performed.

For the pruned dataset, the results show that the use of personalised models typically outperforms the aggregately trained models (AT-PT, Figure 3) if the end-goal objective is to predict opportune moments to interrupt. However, the models perform worse than the aggregate trained models in avoiding ineffective interruptions, yet not worse than



(a) All users. Reachability (N=75), Engage-ability(N=73), Receptivity (not-in-use: N=43, in-use: N=45).

(b) Users with >10 notifications. Reachability (N=43), Engage-ability(N=44), Receptivity (not-in-use: N=17, in-use: N=16).

Figure 4: Distribution of user performance for personalised models (PT-PT)

the typical user model. This suggests that for applications with a greater priority in avoiding missed opportunities to interrupt (such as the ImprompDo application), or for those wishing to perform reasonably well at both, personalised models are better suited than those aggregately trained. This reflects previous conclusions [7, 9], but also shows that this extends beyond a single measure of success and evaluation metrics.

Closer inspection of the performances reveals differences in the distributions of reachability and engage-ability as compared to receptivity, similarly to AT-PT. When the device is not-in-use, the variance in the predictive performance is the lowest across all metrics, yet when the device is in-use the variance is the largest across all metrics. Despite this, the low variance across users suggests that personalised models may be more suitable for applications where performance across users needs to be somewhat consistent. However these differences may be due to the fewer number of users for these models.

6.2. Comparing performance against common smartphone conventions

Analysis of training from aggregate and personalised data has revealed differences in expected prediction performance across different measures of success and evaluation criteria. Previous studies on inferring other attentive states (e.g., [41]) have found that despite classifier accuracy not being considerably high, the models still bring notable

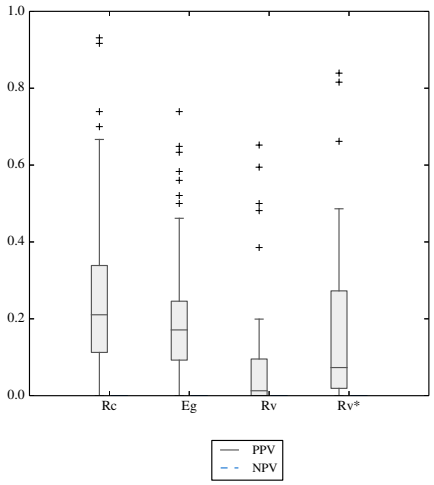


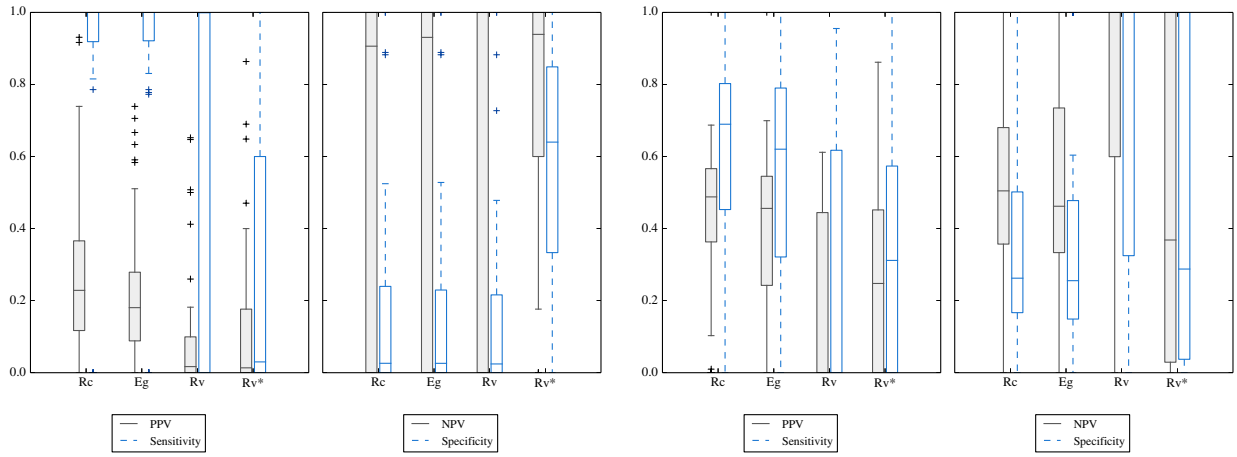
Figure 5: Baseline 1 precision performance across users - The user is always interruptible (default application assumption). Sensitivity is 1.0 and 0 for NPV and specificity, across all models.

improvement. We explore how the performance of our multi-modal models using the DOIG model compares against the existing typical conventions available on Android devices, through two baselines.

Firstly, we simulate the default setting of applications, where a notification can interrupt at all times, by classifying every instance as the user being reachable, willing to engage, or receptive; similarly to previous studies (e.g., [14, 19]). Secondly, we build models based only on the ringer state just before the notification, which enables interruptibility to be declared as a blanket rule. For labelling, we build upon the correlation analysis (Section 4.5), which found significant differences and moderate affect sizes between a silent ringer state and both vibrate and audible, by labelling that the user is not interruptible if the device is silent. From these baselines we aim to achieve the following: 1) determine whether having an interruptibility model is worthwhile at all, and 2) whether a multi-modal model from implicitly observable sensor and API data is worthwhile over only using the user-declared ringer state. However this is only indicative of the data sources chosen and not the suitability of the DOIG model for labelling behaviour (Section 4.4).

6.2.1. Baseline performance

The predictive performance of our first baseline, that the user is always interruptible, is shown in Figure 5. In finding opportunities to interrupt (PPV), the distribution across the measures of success indicates that user's are often more reachable and engage-able to being interrupted by a notification, than receptive to a specific notification. This further



(a) Aggregately trained models (AT-PT)

(b) Personalised models (PT-PT)

Figure 6: Baseline 2 performance across users - The user is interruptible if the device is not silent.

supports the frequency statistics (Section 4.4), in that different types of response behaviour is important to consider [4, 12] and favours the DOIG model over the existing convention. As smartphones do allow a degree of manual-rule based interruption management, Figure 6 shows the performance of models trained from whether the device is silent (baseline 2). For both aggregately trained (AT-PT) and personalised models (PT-PT), the relative differences across the different measures of success are similar to the always interruptible baseline - further supporting that the measure of success chosen should be an important consideration.

Comparing the AT-PT ringer baseline (Figure 6a) against our AT-PT multi-modal model, the baseline performs worse at correctly classifying interruptible moments (PPV) against all of the multi-modal models. Additionally, the median performance for correctly classifying ineffective interruptions (NPV) is better for a large proportion of users. This suggests that just using the ringer state may be a better choice than a multi-modal approach, if this is the sole priority. However this is not reflected in the recall values, which show the opposite distribution to PPV and NPV, when the device is not-in-use. Overall, this suggests that users do not always base their decisions in response to a notification purely on the ringer state rule they have set. While the ringer state is clearly influential, comparisons with the multi-model model suggest that other contextual features are also useful features.

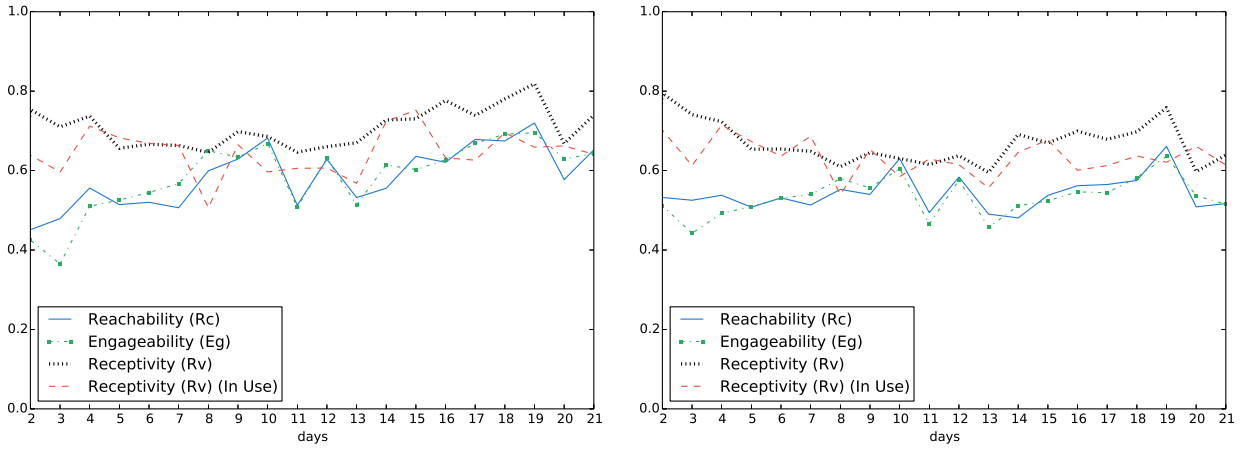
Against our multi-modal PT-PT models, the PT-PT baseline's median reachability and engage-ability performance

is slightly worse across all precision metrics (PPV and NPV), with better sensitivity and worse specificity (Figure 6b) . For receptivity, the baseline is worse at correctly classifying opportunities (PPV), but better at avoiding inappropriate moments (if the device is not in use), for most users. Overall, when considering the entire distributions our multi-modal models have notably less variance across users. This suggests that user's likely use their manual ringer rules differently, and that there may be cases where user's unintentionally forget to change the ringer state at the exact moment their interruptibility changes. Coupling these results with the sole reliance on the human effort required to manage the ringer state, the results suggest that the use of a multi-modal trained interruptibility system is more worthwhile if the objective is to find opportune moments to interrupt; regardless of the measure of success used.

6.3. Performance in an online learning setting

The evaluation of predictive models in an offline environment can provide a useful indication of the overall predictability. However, a hypothetical application won't have this data when a user first installs it. As the baseline analysis suggested in favour of a personalised multi-modal models for applications wishing to perform well at avoiding both false-positive and false-negative predictions, or just false-negatives, we extend our analysis into an online learning environment. To investigate this, we took users with at least 21 days worth of data and, starting from the second day, retrained the predictive models daily, using all data from the previous day(s) as the training data and all the data for that day as the test data. This approach has been used in similar studies (e.g., [9]) and allows us to examine how many days of participation a predictive model would likely need to reach an peak daily performance.

Figure 7 shows the mean performance across users when considering all metrics: weighted precision (PPV and NPV) and recall (sensitivity and specificity). The results indicate that for receptivity, the models perform reasonably well initially, with minor fluctuation between days. For reachability and engage-ability models this is much longer (6-7 days). This suggests that these response behaviours may be more sensitive to differences within similar contexts, where several days worth of behaviour is needed to better distinguish between reachable and unreachable, and engage-able and non-engage-able contexts. This is surprising given that reachability and engage-ability consistently performed better than receptivity in an offline setting. In comparison, previous work in the area relying on labelling notification content has shown to require up to 9 days of training [9]. However we found that for individual metrics, the performance of PPV and sensitivity performed much worse than the weighted values. While this may be influenced by the random-



(a) Weighted Precision (PPV/NPV)

(b) Weighted Recall (Sensitivity/Specificity)

Figure 7: Online learning visualisation for the first 21 days, using the mean value of users with >21 days participation. Reachability (N=27), Engage-ability(N=27), Receptivity (not-in-use: N=27, in-use: N=18).

under-sampling pre-processing step, this suggests that training from more data (e.g. in an offline environment) should be preferred where possible, if an application's priority is to avoid missed opportunities to interrupt.

Overall, the results support the use of the DOIG model in online learning environments and that predictive models built using only implicitly sample contextual data can perform initially where the number of data points will be small. However, as with offline learning, the priorities in the evaluation metrics produce wide variance in expected prediction performance. In the case of the ImprompDo data collection application, this may not be an issue as changes to productivity habits are likely to take time. However, this may not be true for all applications, where the use of an offline trained aggregate model may be more suitable, if only temporarily until personalised data has been collected.

7. Discussion

The analysis conducted raises several design considerations that support the development of future notification mechanisms, addressing limitations exposed in the existing literature [4]. Firstly, we present the DOIG model that extends the existing black-box convention [12] for implicitly capturing and representing interruption response behaviour, enabling per-application flexibility through considering different interpretations of interruptibility. Despite being limited to observable decisions, we find support for the model through an in-the-wild case study; with evidence of isolating

different response behaviour and subsequently reducing the potential for false-negative classifications, in comparison to the existing convention (Section 4.4). This is further supported through finding different correlating contextual features for different interruptibility labels (Section 4.5) and that each is typically predictable in line with existing studies (Section 6), but with notable differences across training and evaluation methods (Sections 6.1.1 - 6.3).

In exploring the predictability of the DOIG model we can also create further considerations depending on the priorities of individual applications. If a hypothetical application is seeking to predict opportune moments to interrupt, by prioritising true-positive classifications, we find that personalised models typically outperformed a model trained from aggregated data. We also find that a multi-modal approach brings greater stability over common mechanisms on the device through reduced variation across users and that these models perform best in an offline environment. Crucially, we found minimal differences across reachability, engage-ability, or receptivity labelled models for personalised training, suggesting that an application choosing one over another is unlikely to be disadvantaged.

If an application is seeking to avoid issuing notifications that will not likely produce their desired response behaviour (e.g., being at least reachable) by prioritising true-negative classifications, we find that a model trained from aggregate data typically outperformed personalised models. This suggests that un-interruptible moments are more common across individuals, than interruptible moments, where the absence of personal data could be supplied by the aggregated data of other users. Interestingly, we find that receptivity typically outperforms reachability and engage-ability for these models, suggesting that curiosity to investigate notifications may have an impact for proportion of users, supporting the need to consider how a response is made when labelling their interruptibility (i.e. using the DOIG model). We also find that just considering whether the device is silent or not can provide similar benefits to a multi-modal approach; however with much larger variance across users (Figure 6a). This is likely due to different individual habits in changing the ringer state and the potential for other cues (e.g. flashing LEDs). However, it is interesting that the inclusion of additional context data (Figure 3b) improves upon this unreliability through reduced variance across users.

7.1. Limitations

Our in-the-wild study aimed to represent as many different real-world application use cases as possible. However in doing so, the contextual features used for prediction were limited to those that any Android application could adopt without a fundamental change to their permissions or design (as discussed in Section 4.2). On a per-application basis,

additional features could be feasible to sample, such as the current location, activity, or social situation. While their exclusion enables wide applicability of the results in this study, in enabling any Android application to implement the case study design, the results may underestimate the predictive performance that could be achieved for applications that can consider additional data sources.

Previous work has shown that additional data sources may have predictive power for at least some definitions of interruptibility. For example, the time since the last device activity (e.g., [18]), calendar data (e.g., [44]), current task data (e.g., [32, 45, 17]), location [14], microphone data (e.g., [19]), or activity recognition (e.g., [9]). Additionally for applications with highly variable notification content (e.g., instant messages), including factors relating to the content (e.g. the sender) could provide a benefit to predicting receptivity. Future work could explore, on a per-application basis, the trade-off of adding more data sources with any increased predictive performance [4] and whether their predictive power could enable further feature reduction [19].

8. Conclusions

Systems capable of predicting interruptibility are of increasing interest as applications demanding our attention become more ubiquitous. Whilst considerable progress has been made concerning interruptibility over the last decade, the area is now diverse with specific solutions for specific interruptions and environments, where the boundaries of wider applicability is unclear. Key areas in need of greater attention [4] include: accommodating various definitions in what behaviour makes a notification successful; decreasing the additional cognitive burden placed on the user to collect and label data; and greater exploration into machine learning methods, in-line with per-application priorities.

In this work we firstly propose the decision-on-information-gain model, which involves identifying the conscious or subconscious micro-decision steps that a user must make in responding to an interruption. The approach lends itself well to typical Android notifications, as there is a well-defined and implicitly observable process of interactions that are naturally performed. The current "black-box" convention [4, 12] under-represents this process by primarily relying on whether the notification is fully consumed, or requiring a survey to be completed [7]. In reality, a user may respond partially and could therefore be interruptible, but not for a particular application, summary topic, or specific content; the distinction of which is valuable for different applications. Through an in-the-wild field study of 11,346

Android notifications we implicitly observed that a significant number of responses fell into this category, with our approach increasing the potential number of responses to consider by up to 124.7%. This reduces the potential for misclassifications that the user is not at all interruptible in comparison to the existing convention. Future work could explore the extent to which this also applies to other types of notifications, through further empirical case studies.

Secondly, we examine the differences in adopting different definitions of interruptibility using our approach, (i.e., reachability, engage-ability, and receptivity). We find differences in the correlating contexts just before the notification, as well as in the prediction performance across various machine learning conventions; including training data selection, training environments, and evaluation metrics. Some combinations of these variations produced >80% precision performance for the majority of users, however we note variability across the models created. From this we propose several design considerations based on whether a hypothetical application's priority is to find opportunities to interrupt, or avoid ineffective notifications. Overall, for future research and the design of intelligent interruption systems using Android notifications, these results further support the use of the DOIG model, but also highlights the dangers of assuming wider applicability beyond the confines of a single set of labelling, training, and evaluation choices.

References

[1] R. M. Whitaker, M. Chorley, S. M. Allen, New frontiers for crowdsourcing: The extended mind, in: System Sciences (HICSS), 2015 48th Hawaii International Conference on, IEEE, 2015, pp. 1635–1644.

[2] M. J. Chorley, R. M. Whitaker, S. M. Allen, Personality and location-based social networks, Computers in Human Behavior 46 (2015) 45–56.

[3] D. McFarlane, Comparison of four primary methods for coordinating the interruption of people in human-computer interaction, Human-Computer Interaction 17 (1) (2002) 63–139.

[4] L. D. Turner, S. M. Allen, R. M. Whitaker, Interruptibility prediction for ubiquitous systems: Conventions and new directions from a growing field, in: Proc. UbiComp'15, ACM, 2015, pp. 801–812.

[5] S. Grandhi, Q. Jones, Technology-mediated interruption management, International Journal of Human-Computer Studies 68 (5) (2010) 288–306.

[6] S. Grandhi, Q. Jones, Knock, knock! who's there? putting the user in control of managing interruptions, International Journal of Human-Computer Studies 79 (2015) 35–50.

[7] V. Pejovic, M. Musolesi, Interruptme: designing intelligent prompting mechanisms for pervasive applications, in: Proc. UbiComp'14, ACM, 2014, pp. 897–908.

[8] S. Moran, J. E. Fischer, Designing notifications for ubiquitous monitoring systems, in: Proc. PerCom'13 (PERCOM Workshops), IEEE, 2013, pp. 115–120.

[9] A. Mehrotra, M. Musolesi, R. Hendley, V. Pejovic, Designing content-driven intelligent notification mechanisms for mobile applications, in: Proc. UbiComp'15, ACM, 2015, pp. 813–824.

[10] R. Fisher, R. Simmons, Smartphone interruptibility using density-weighted uncertainty sampling with reinforcement learning, in: ICMLA'11, Vol. 1, IEEE, 2011, pp. 436–441.

[11] S. Rosenthal, A. K. Dey, M. Veloso, Using decision-theoretic experience sampling to build personalized mobile phone interruption models, in: Pervasive Computing, Springer, 2011, pp. 170–187.

[12] L. D. Turner, S. M. Allen, R. M. Whitaker, Push or delay? decomposing smartphone notification response behaviour, in: Human Behavior Understanding, Vol. 9277 of Lecture Notes in Computer Science, Springer International Publishing, 2015, pp. 69–83.

[13] J. Smith, A. Lavygina, J. Ma, A. Russo, N. Dulay, Learning to recognise disruptive smartphone notifications, in: Proc. MobileHCI'14, ACM, 2014, pp. 121–124.

[14] B. Poppinga, W. Heuten, S. Boll, Sensor-based identification of opportune moments for triggering notifications, Pervasive Computing, IEEE 13 (1) (2014) 22–29.

[15] D. Avrahami, S. E. Hudson, Responsiveness in instant messaging: predictive models supporting inter-personal communication, in: Proc. CHI'06, ACM, 2006, pp. 731–740.

[16] T. Tanaka, K. Fujita, Study of user interruptibility estimation based on focused application switching, in: Proc. CSCW'11, ACM, 2011, pp. 721–724.

[17] S. T. Iqbal, E. Horvitz, Notifications and awareness: a field study of alert usage and preferences, in: Proc. CSCW'10, ACM, 2010, pp. 27–30.

[18] M. Pielot, R. de Oliveira, H. Kwak, N. Oliver, Didn't you see my message?: predicting attentiveness to mobile instant messages, in: Proc. CHI'14, ACM, 2014, pp. 3319–3328.

[19] J. Fogarty, S. E. Hudson, C. G. Atkeson, D. Avrahami, J. Forlizzi, S. Kiesler, J. C. Lee, J. Yang, Predicting human interruptibility with sensors, ACM Transactions on Computer-Human Interaction (TOCHI) 12 (1) (2005) 119–146.

[20] N. Kern, S. Antifakos, B. Schiele, A. Schwaninger, A model for human interruptability: experimental evaluation and automatic estimation from wearable sensors, in: Proc. ISWC'04, Vol. 1, IEEE, 2004, pp. 158–165.

[21] D. Avrahami, J. Fogarty, S. E. Hudson, Biases in human estimation of interruptibility: effects and implications for practice, in: Proc. CHI'07, ACM, 2007, pp. 50–60.

[22] S. Hudson, J. Fogarty, C. Atkeson, D. Avrahami, J. Forlizzi, S. Kiesler, J. Lee, J. Yang, Predicting human interruptibility with sensors: a wizard of oz feasibility study, in: Proc. CHI'03, ACM, 2003, pp. 257–264.

[23] J. Ho, S. S. Intille, Using context-aware computing to reduce the perceived burden of interruptions from mobile devices, in: Proc. CHI'05, ACM, 2005, pp. 909–918.

[24] B. P. Bailey, S. T. Iqbal, Understanding changes in mental workload during execution of goal-directed tasks and its application for interruption management, ACM Transactions on Computer-Human Interaction (TOCHI) 14 (4) (2008) 21.

[25] S. T. Iqbal, B. P. Bailey, Effects of intelligent notification management on users and their tasks, in: Proc. CHI'08, ACM, 2008, pp. 93–102.

[26] S. T. Iqbal, B. P. Bailey, Oasis: A framework for linking notification delivery to the perceptual structure of goal-directed tasks, ACM Transactions on Computer-Human Interaction (TOCHI) 17 (4) (2010) 15.

[27] J. E. Fischer, N. Yee, V. Bellotti, N. Good, S. Benford, C. Greenhalgh, Effects of content and time of delivery on receptivity to mobile interruptions, in: Proc. MobileHCI'10, ACM, 2010, pp. 103–112.

[28] D. C. McFarlane, K. A. Latorella, The scope and importance of human interruption in human-computer interaction design, Human-Computer Interaction 17 (1) (2002) 1–61.

[29] G. Miller, The smartphone psychology manifesto, Perspectives on Psychological Science 7 (3) (2012) 221–237.

[30] S. Giordano, D. Puccinelli, When sensing goes pervasive, Pervasive and Mobile Computing 17 (2015) 175–183.

[31] J. Fogarty, S. E. Hudson, J. Lai, Examining the robustness of sensor-based statistical models of human interruptibility, in: Proc. CHI'04, ACM, 2004, pp. 207–214.

[32] T. Okoshi, J. Ramos, H. Nozaki, J. Nakazawa, A. K. Dey, H. Tokuda, Attelia: Reducing user's cognitive load due to interruptive notifications on smart phones, in: Proc. PerCom'15, IEEE, 2015, pp. 96–104.

[33] H. Sarker, M. Sharmin, A. A. Ali, M. M. Rahman, R. Bari, S. M. Hossain, S. Kumar, Assessing the availability of users to engage in just-in-time intervention in the natural environment, in: Proc. UbiComp'14, ACM, 2014, pp. 909–920.

[34] S. Mathan, S. Whitlow, M. Dorneich, P. Ververs, G. Davis, Neurophysiological estimation of interruptibility: Demonstrating feasibility in a field context, in: In Proceedings of the 4th International Conference of the Augmented Cognition Society, 2007, pp. 51–58.

[35] H. He, E. A. Garcia, Learning from imbalanced data, Knowledge and Data Engineering, IEEE Transactions on 21 (9) (2009) 1263–1284.

[36] S. Kim, J. Chun, A. K. Dey, Sensors know when to interrupt you in the car: Detecting driver interruptibility through monitoring of peripheral interactions, in: Proc. CHI'15, ACM, 2015, pp. 487–496.

[37] A. Mehrotra, V. Pejovic, J. Vermeulen, R. Hendley, M. Musolesi, My phone and me: Understanding people's receptivity to mobile notifications, in: Proc. CHI'16, ACM, 2016, pp. 1021–1032.

[38] N. D. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, A. T. Campbell, A survey of mobile phone sensing, Communications Magazine, IEEE 48 (9) (2010) 140–150.

[39] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten, The weka data mining software: an update, ACM SIGKDD explorations newsletter 11 (1) (2009) 10–18.

[40] T. Tanaka, R. Abe, K. Aoki, K. Fujita, Interruptibility estimation based on head motion and pc operation, International Journal of Human-Computer Interaction 31 (3) (2015) 167–179.

[41] M. Pielot, T. Dingler, J. S. Pedro, N. Oliver, When attention is not scarce - detecting boredom from mobile phone usage, in: Proc. UbiComp'15, ACM, 2015, pp. 825–836.

[42] S. T. Iqbal, B. P. Bailey, Leveraging characteristics of task structure to predict the cost of interruption, in: Proc. CHI'06, ACM, 2006, pp. 741–750.

[43] G. H. Ter Hofte, Xensible interruptions from your mobile phone, in: Proc. MobileHCI'07, ACM, 2007, pp. 178–181.

[44] N. Kern, B. Schiele, Towards personalized mobile interruptibility estimation, in: Location-and Context-Awareness, Springer, 2006, pp. 134–150.

[45] T. Okoshi, H. Nozaki, J. Nakazawa, H. Tokuda, J. Ramos, A. K. Dey, Towards attention-aware adaptive notification on smart phones, Pervasive and Mobile Computing 26 (2016) 17–34.