

修士論文

通知情報調査システムの構築と 端末側での適応的通知タイミング制御手法の提案

高城 賢大

2019 年 1 月 31 日

奈良先端科学技術大学院大学
情報科学研究科

本論文は奈良先端科学技術大学院大学情報科学研究科に
修士(工学) 授与の要件として提出した修士論文である。

高城 賢大

審査委員：

安本 慶一教授 (主指導教員)

清川 清 教授 (副指導教員)

荒川 豊 准教授 (副指導教員)

諏訪 博彦 助教 (副指導教員)

通知情報調査システムの構築と 端末側での適応的通知タイミング制御手法の提案*

高城 賢大

内容梗概

自由にアプリケーションを追加することができるスマートフォンが全世界に広がっている。インストール可能なアプリケーション数は、2017年時点で、iPhone向けのものだけでも210万本も公開されており、ユーザはそれぞれの目的に応じて多数かつ多様なアプリケーションをインストールしている。これらのアプリケーションには情報の到着をユーザに伝える通知機能が備わっている。通知機能は、定期的に発動するものもあれば、情報提供者から送られるPush通知というものもある。近年、この通知が多くなりすぎて、重要な情報に気づかない、そもそも通知を見なくなった、という問題が起きている。そこで、ユーザの状態（コンテキスト）を認識して割り込み通知のタイミングを最適化するという研究が広がっている。例えば、ユーザに動きを検知して、立ち止まっているときにまとめて通知をすることでクリック率(CTR)が改善されるといった研究が有名である。しかしながら、このような既存研究の多くは、対象となるアプリ以外は通知タイミングを制御しないアプリと想定している。将来的に、すべてのアプリが同じような通知タイミング制御を行う用になった場合、結局、通知の一極集中が発生し、通知タイミング制御の効果が発揮されないことが考えられる。また、インストールされているアプリがユーザ毎に異なるため、コンテキスト認識だけでなく、ユーザの端末に導入されている全てのアプリの挙動を考慮した通知タイミングの制御が必要であると考えられる。そこで、本研究では、まず、コンテキストに加え、全てのアプリの挙動を考慮した通知タイミング制御を適応的通知タイミング制御

*奈良先端科学技術大学院大学 情報科学研究科 修士論文, 2019年1月31日.

と定義し、この実現に向け、ユーザが受け取る通知の多様性調査及び制御モデルの構築とその評価を行なった。本稿ではまず、プライバシーを排除しつつ通知情報を取得し続けるシステムを作成し、クラウドソーシングを用いて実際に収集した実験結果を報告する。次に、受け取るそれぞれの通知に対し、ユーザがクリックするか否かを予測するモデルの構築を行った。実験期間内の通知クリック回数が50回以下のユーザを除外し、14人から得た通知データを使用し、実際にクリックされる通知の予測モデルの精度評価を行なった。クリックやデリートなど、ユーザによる直前の通知アクションからの通知の蓄積件数をパラメータに加えることで、推定精度が大きく向上することがわかった。この結果から、端末側で全てのアプリの通知挙動を使用しタイミング制御を行うことは、通知のCTRを向上させる上で有効な手法であることを示した。

キーワード

通知制御, モバイルアプリケーション, アダプティブコンピューティング, コンテキストウェアネス, 割り込み通知, モバイル調査システム

Design and Implementation of Notification Information Survey System and proposal of User-side Adaptive Notification Management Method*

Kenta Taki

Abstract

Smartphones that can add applications freely are spreading all over the world. By 2017, the number of applications that can be installed is 2.1 million published only on the App store, and users install many and diverse applications according to their purpose. These applications have a notification function to notify the arrival of information. Some notification functions are invoked periodically, while others are push notifications sent from information providers. In recent years, the increase of notification causes the problem that the user do not noticing important information, or not seeing any notification at all. Therefore, many research on notification optimization by taking the user's state (context) into account have conducted. One of the famous studies reported that the click rate (CTR) can be improved by detecting the motion to the user and notifying them collectively when stopped. However, if all applications are controlled by the same notification timing, interrupt timing will become concentrated and the effects of notification timing control may not be exerted in the future. In addition, since the installed applications are different by each user, it is necessary

*Master's Thesis, Graduate School of Information Science, Nara Institute of Science and Technology, January 31, 2019.

to control notification timings with taking the behaviors of all the applications installed on a user's smartphone into consideration. In this thesis, we define notification timing control that considers the behaviors of all installed applications as "Adaptive Notification Management", and conduct the survey to clarify the diversity of notifications received by developing the system that can acquire all notification information while excluding privacy data. We also report on experimental results using actual data collected using crowdsourcing. Next, we built a model to estimate whether or not the user clicks on all the received notifications from multiple applications. We excluded users with a click number of notifications of 50 times or less during the experiment period of 2 weeks from the data set and evaluated the estimation accuracy of the notifications actually clicked. It is found that the estimation accuracy is greatly improved by adding the number of unopened notification counted from the last notification when the user performed the reaction to the parameter. From this result, we can say that a timing control using the notification behavior of all applications on the use-side is an effective method to improve the clickthrough rate (CTR) of the notification.

Keywords:

Notification management, Mobile application, Adaptive computing, Context awareness, Interrupt notification, Mobile survey system

目次

1. はじめに	1
2. 関連サービスと既存研究	3
2.1 スマホ OS 側の通知施策	3
2.2 通知割り込みに関する研究	4
2.3 通知の個人最適化に関する研究	5
2.4 Notification Listener service API を利用した研究	5
2.5 通知カテゴリから見た応答性の研究	6
2.6 本研究の位置づけ	6
3. 通知情報調査システムによる調査	7
3.1 調査システム	8
3.1.1 システムアーキテクチャ	8
3.1.2 プライバシ保護	8
3.2 調査実験	11
3.2.1 調査手法	11
3.2.2 調査条件	11
3.3 実験結果	12
3.3.1 アプリのカテゴリごとの解析結果と考察	12
3.3.2 スポットごとの結果と考察	17
3.3.3 時間帯ごとの結果と考察	19
3.3.4 この章のまとめ	19
4. 制御モデルの構築とその評価	22
4.1 通知クリックアクションの予測方法	22
4.2 全ユーザのデータセットを用いた予測結果	23
4.3 各ユーザ毎のクリックアクションの予測結果	25
4.3.1 4つの正解ラベルの予測予測	25
4.3.2 ポジティブな通知又は、ネガティブな通知であることの予測	26

4.3.3	ポジティブな通知の学習データ数と予測精度	28
4.4	考察	29
5.	結論	31
5.1	本研究のまとめ	31
5.2	本研究の課題と今後の展望	32
	謝辞	34
	参考文献	35

図 目 次

1	iOS12 からの新機能	3
2	Android 9 pie からの新機能	4
3	Notification Logger のスクリーンショット	7
4	システム構成図	9
5	通知クリックまでの経過時間ヒストグラム	15
6	直前の通知からの経過時間	17
7	通知受信上位 5 スポットでのクリック率	18
8	上位 5 スポットでの通知クリックまでの経過時間 CDF	19
9	時間に対するスポットの分布	20
10	通知受信時間帯ごとの通知数とクリック数	21
11	ポジティブな通知の正解値ラベル定義	23
12	ネガティブな通知の正解値ラベル定義	24
13	全ユーザのデータセットを用いたクリック予測の Confusion matrix	26
14	個人での予測に対する Confusion matrix	27
15	経過日数に対するポジティブな通知予測結果	28
16	データ数を元にした学習回数に対するポジティブな通知予測結果	29

表 目 次

1	収集データ	10
2	被験者属性	12
3	実験結果概要	13
4	カテゴリ毎のユーザアクションの数	14
5	3600s 以内にクリックされた通知の直前の通知	16
6	各分類器による機械学習の結果	25

1. はじめに

全世界でのスマートフォンアプリのダウンロード数は500億を突破し [1], 日々, 新しくて便利なアプリがストア上に公開されている. ユーザはそれぞれの目的に応じて多様なアプリをインストールしており, それらのアプリの多くには通知機能が備わっている. 通知機能には, 現在, 定期的に発動する通知と, 配信者から送られる Push 通知の2種類が存在するが, そのいずれもアプリケーションがユーザに対して能動的に情報通知を行う重要な手段であり, 通知を効果的に活用することで, アプリのリテンション率 (顧客維持率) が大幅に向上することも明らかになっている [2]. その効果の高さ故に, 様々な企業やアプリ開発者が通知機能を使用しており, ユーザが受け取る情報通知の数は年々増大している. しかし, 人間が認知できる情報の量は変わらないため, 不適切なタイミングでの通知は, ユーザのストレスの増加や生産性の低下を引き起こしてしまう [3][4]. そこで, iOS や Android の OS 開発者側は近年のアップデートの中で, スマホのユーザ体験を高く保つために, 通知の受信時間の詳細な設定や不必要な通知をユーザがオフにしやすいようなユーザインターフェイスを導入しており [5][6], ユーザ体験を阻害するアプリからの通知はなるべく除外する方針を暗黙的に示している.

一方で, 一度通知をオフにされてしまうと, アプリのリテンション率が低下してしまうことから, アプリ開発者側もユーザ体験を阻害することなく通知をタップさせるために, コンテキスト認識を活用した最適なタイミングでの割り込みに関する研究等を多く行っている. 大越らは機械学習に基づく解析により, プッシュ通知がクリックされやすいブレイクポイントをユーザの行動の節目に検知する手法を提案している [7]. また, 赤池らは応答時間帯と受信者の忙しさに基づいた通知によりユーザのクリック率を向上させる手法を提案している [8].

これらの手法に基づきタイミングの制御を行うことで, 通知に対するユーザの応答率を向上させることが可能になることが明らかになった. しかし, 制御対象となる単一のアプリだけでなく, スマホ内の複数のアプリケーションが全て同様の制御を行った時, 有限な資源である人間一人の注意 (アテンション) に対し一極集中化が起こってしまうことが考えられる. この問題を解決するためには, ユーザ一人一人の全ての通知に対する挙動を考慮した制御を行う必要があると考えら

れる．しかし，ユーザ端末にインストールされているアプリの種類や，そのアプリに対する通知の許可状況はユーザごとに異なるため，端末内に存在するアプリを知り得ないサービス提供側がこれを考慮して制御を行うのは困難である．実際に Yahoo と共同で行われている大越らの研究 [7] の実験系においても，制御対象である Yahoo のアプリ以外は未制御の通常アプリとして想定している．そのため，もし他の複数アプリのすべてが検知されたブレイクポイントに対して通知を行った時，その制御パフォーマンスに悪影響を与える可能性がある．そこで，本研究では，ユーザの端末に存在する全てのアプリの通知に対するユーザの挙動を調査した上で，その挙動を考慮しながら端末側で通知の発火をタップされやすいタイミングに制御する手法を検討する．これを適応的通知タイミング制御と呼ぶ．

本研究では，適応的通知タイミング制御の実現のために，ユーザが受け取る通知の多様性とそれぞれに対する挙動を調査するため，プライバシーを排除しつつ全種類の通知情報を取得し続けるシステムの構築を行った．さらにそれを用いて，クラウドソーシングで募集した一般ユーザを対象に 4 週間の調査を行ない，実験参加者 20 人から 28 日間で 95,910 件の通知データを収集した．収集したデータセットに対し，統計解析を行ったところ，通知応答率はアプリの種類に依存し，反応時間にはアプリの種類と通知受信時の場所が影響を及ぼすことがわかった．また，アプリの通知開封はそのアプリの通知タイミングだけではなく，他のアプリの通知との組み合わせにも関係があり，他のアプリの通知によってアプリの開封が誘起されている場合も多いことが新たに分かり，直前に受け取った通知が応答性と相関があることが示唆できた．

次に通知のクリック予測モデルの構築を行ない，調査実験で収集したデータを用いて評価を行なった．アプリのカテゴリ，通知の受信時間，受信スポット，ユーザの行動に加え，直前の応答からの通知蓄積件を属性値，通知に対するユーザのクリックやデリートのアクションを正解値を入力値として，機械学習によるクリックされる通知の予測を行った．実測値の予測精度 7.7% に対して，提案モデルでの予測精度は 26% となり，通知の蓄積数をパラメータに加えることで予測精度が約 15% 向上することがわかった．

2. 関連サービスと既存研究

2.1 スマホ OS 側の通知施策

スマートフォン OS の昨今のアップデートにおいて、UX(ユーザエクスペリエンス)を下げないための工夫として、通知施策にチカラを入れている傾向が見られる。例えば、iPhone の最新 OS である iOS12 からは、図1のように、アプリごとに通知がグルーピングされる機能が追加され、スマホのロック画面からでもワンタップで通知をオフにできるようになった。また、Android の最新 OS、Android 9 pie ではスマートフォンを裏返すだけでマナーモードにすることができる機能(図2)など、ユニークな機能が追加されている。このようにスマホ OS 側もユーザ体験を阻害するアプリからの通知はなるべく除外する方針を暗黙的に示している。

iOS12



図 1 iOS12 からの新機能

Android 9 pie



スマートフォンを裏返すとマナーモードになる「Shush」機能

図 2 Android 9 pie からの新機能

2.2 通知割り込みに関する研究

モバイル端末からの情報通知は、ユーザに対する「割り込み」と見なすことができる。この「割り込み」がユーザにとって不適切なタイミングで行われた時、ユーザのストレスの増加や生産性の低下を引き起こしてしまうことが報告されている [9]。

この問題を解決するために、大越らはプッシュ通知が開封されやすい「ブレイクポイント」を機械学習に基づき検知し、スマートフォンが受信した通知をその瞬間にずらして発火させる事により、通知への反応時間を短縮する手法を提案している [7]。赤池らは応答時間帯と機械学習に基づいて検知した受信者の「忙しさ」を考慮した通知によりユーザの応答性を向上させる手法を提案している [8]。

2.3 通知の個人最適化に関する研究

上記の研究はサービスのプロバイダ側が対象のアプリをユーザのコンテキストを元に制御する事で応答率をあげようとする手法を提案しているが、端末側で受け取った通知情報を元にデバイスのコンテキストを分析する研究も行われている。コンテキストウェア・コンピューティングの普及とモバイル端末の飛躍的な性能の向上により、旧来の様にモバイルユーザ全員に対して同じサービスを提供するのではなく、端末側でユーザー一人一人の行動や環境に適合したチューニングを行いながら、サービスを提供することができる時代になった。まず、Rosenthalらはユーザのコンテキスト情報に基づいてモバイル端末を自動でミュートに切り替える手法を提案した [10]。Goyalらはウェアラブルデバイスを用いて皮膚電気活動(EDA)からユーザが通知に注意を払いやすいタイミングの検知を試みている [11]。また、Aminikhanghahiらは行動認識と教師あり学習によるフレームワークを組み合わせることで、反応率を 12.8%から 93.2%に向上させる手法 [12]、Hoらはユーザコンテキストを使用した、強化学習ベースでの通知タイミングのパーソナライゼーション手法を提案している [13]。

2.4 Notification Listener service API を利用した研究

Android 端末では「Notification Listener Service」 [14] と呼ばれる API を使用することができる。これによって、端末でユーザが通知を受信または削除した時の通知アプリ名、テキストメッセージ、タイムスタンプなど情報を受け取ることができる。Weberらはモバイルデバイスに関する通知研究のために 6 万人を超えるユーザが使用するオープンソースフレームワークを開発した [15]。また、Sahamiらはユーザやイベントに関する情報を含むメッセージアプリの通知を評価し、ユーザの主観に焦点を当てた通知分析を行った [16]。

2.5 通知カテゴリから見た応答性の研究

Sahami らは 40,000 人以上のユーザから約 2 億件の通知を収集した。それらをアプリの種類ごとにカテゴリライズし、デスクトップアプリと連携することによりすることで、通知への応答性を客観的、主観的に評価している。ユーザはメッセージャー、他のコミュニケーションアプリ、およびカレンダーからの通知を重視し、また、通知は正午付近と夜に多く受信されていることが報告されている。[16][17].

2.6 本研究の位置づけ

大越らや Aminikhanghahi らの研究の貢献で、コンテキスト認識によりユーザが反応しやすいタイミングを見つけ出し、通知を発火させることで反応率を向上させることが可能になった。しかし、1 章で述べたように、将来的に、複数のアプリが同様のタイミング制御を行うようになった時に、割り込みの一極集中化が起こってしまい、パフォーマンスに影響を与えてしまう。

そこで、本稿の位置づけは、次章以降で述べる、スマホ内の全通知をログするアプリを用い、ユーザが受け取る通知の多様性とそれぞれに対する挙動を調査と収集したデータを用いて通知タイミングの制御モデルを構築し、評価を行うことで、端末が受信する複数アプリからの全ての通知状況にも適応した通知制御手法を検討することである。

3. 通知情報調査システムによる調査

本研究において、ユーザの端末に届く通知にどのような多様性があるかを調査するためにユーザの通知情報を端末内に記録できる Android アプリ「Notification Logger」(図 3)を開発した。以降では、3.1 節にてシステムのアーキテクチャについて述べ、3.2 節にてデータ収集時におけるユーザプライバシーの取り扱いについて述べる。



図 3 Notification Logger のスクリーンショット

3.1 調査システム

3.1.1 システムアーキテクチャ

図 4 にスマホアプリ Notification Logger のシステム構成を示す。本実験ではまず、ユーザが通知を許可している全アプリの通知に対する受信または削除の挙動をバックグラウンドで動作する NotificationListenerService [14] を通して取得する。そしてその時のユーザの行動と位置情報をそれぞれ、Google Activity Recognition API, Google Fused Location Provider API から取得し端末内に保存する。その後、ユーザ手動で保存したログデータをエクスポートできる仕組みをとっている。2.3 節で述べた関連研究との大きな違いとしては、Android 8.0 以降の最新 OS から取得可能になった通知アクション（通知クリック、通知削除、同一アプリの通知削除など）のロギングに対応している点である。これにより、これまでは通知が消された事しか確認できなかったのに対し、ユーザが通知を開封し中身を確認したかどうか、という通知に対するアクションをロギング可能になっている。本調査では主に、このユーザの通知開封動作を中心とした解析を行う。

3.1.2 プライバシ保護

表 1 にシステムで収集するデータを示す。NotificationListenerService を用いて通知の受信時と削除時のタイムスタンプとその通知に含まれるアプリ名を取得し、その通知の受信又は削除を通知タイプ、通知に対してユーザが行った操作を通知アクションとして取得した。また、その時のユーザの位置情報を Google Fused Location Provider API から、ユーザの行動を Google Activity Recognition API から取得した。通知に含まれるテキストメッセージも本来は取得することができるが、今回の実験では被験者の参加ハードルを下げるためにメッセージ内容の取得は敢えて行っていない。それでもまだ、収集するデータの中にはユーザが受け取った通知のアプリ名やユーザの位置情報、行動情報など個人情報データが含まれている可能性があるため、今回の調査システムでは、最大限ユーザのプライバシーを考慮した設計としている。具体的には、ロギングしたデータは、自動的にサーバにアップロードされず、端末内に保管する仕組みとした。そして、データ

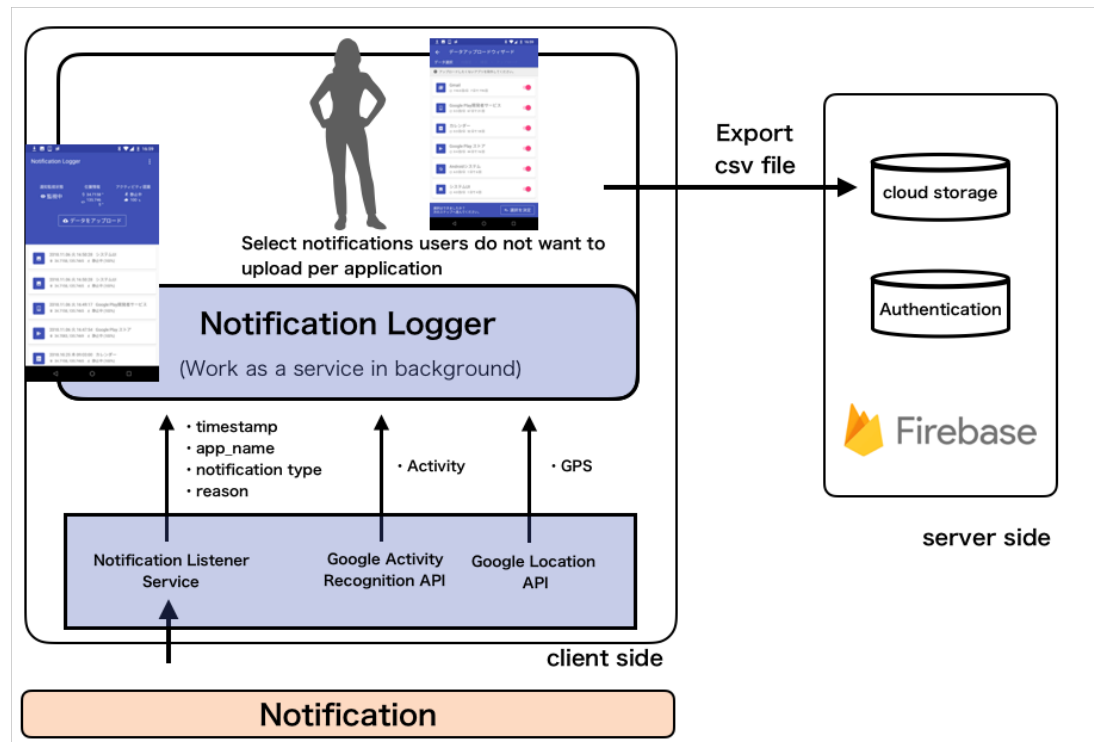


図 4 システム構成図

提供の際には、ユーザ自身がロギングされた通知情報を確認し、アプリ単位で提供データの取捨選択できるようにしている。

表 1 収集データ

データ	概要
Timestamp	通知アクション発生時間
位置情報 ¹	通知アクション発生時の GPS 情報 (Latitude, Longitude)
App Name ²	通知アクション対象のアプリ
通知タイプ ²	Posted Removed
通知アクション ² (Reason code)	REASON_CLICK(1) ³ REASON_CANCEL(2, 8) ⁴ REASON_APP_CANCEL_ALL(9) ⁵ REASON_GROUP_SUMMARY_CANCELED(12) ⁶
行動認識 ⁷	IN_VEHICLE ON_BICYCLE ON_FOOT STILL TILTING UNKNOWN

¹ Google Fused Location Provider API を使用

² NotificationListenerService を使用

³ 通知クリックアクション

⁴ 通知単一削除アクション。本研究では REASON_CANCEL (2) と REASON_APP_CANCEL (8) は同一のアクションとみなす

⁵ 通知グループ削除アクション

⁶ 通知全削除アクション

⁷ Google Activity Recognition API を使用

3.2 調査実験

本調査実験ではユーザが受け取るアプリの種類，ユーザのコンテキスト (通知を受け取った時間帯や場所，その時の行動) の違いによる，通知の応答性を調査することを目的とする．3.2.1 項で調査手法について述べ，3.2.2 項で調査条件，被験者の属性について述べる．

3.2.1 調査手法

本実験では，国内大手のクラウドソーシングサービスである「クラウドワークス」を使用して被験者の募集を行った．応募者の内 20 人を対象に 4 週間のデータ収集実験を行った．被験者には普段から使用している個人の Android スマートフォンに 3 章で述べた「Notification Logger」をインストールしてもらい，位置情報とセンサ情報の取得許可を行ない，通知が監視中になった状態で普段通りの生活を送ってもらった．被験者には 1 週間毎にリマインドを送り，データのアップロードを 4 回に分けて行なってもらった．

3.2.2 調査条件

Notification Logger での通知取得で使用している「Notification Lister Service」では AndroidOS のバージョンが Android8.0 以上の端末でしか通知タップ (CLICK, 開封) の挙動を取得することができないため，Android8.0 以上の OS が搭載されている端末を使用している応募者のみを採用した．表 2 に被験者の属性を示す．被験者は 20~49 歳までの男女 20 人を対象とした．

表 2 被験者属性

表 2 被験者属性			
			人数
被験者の属性内訳	性別	男性	7
		女性	13
	年齢	20-29 歳	8
		30-39 歳	9
		40-49 歳	3
	職業	会社員	9
		主婦	6
		フリーター	3
		学生	2
被験者数計			20

3.3 実験結果

表 3 に実験結果の概要を示す。本実験では 4 週間で 95,910 件の通知データを取得することができた。通知データの内訳は、通知の受信件数が 68,261 件、通知の削除件数が 20,345 件であった。削除の内 3,506 件が通知タップ（開封）による削除であった。

3.3.1 アプリのカテゴリごとの解析結果と考察

カテゴリ分けしたアプリの種類ごとのクリック（開封）率と通知クリックまでの経過時間を考察する。

■ アプリのカテゴリ毎のユーザアクション まず、アプリ名から通知のカテゴリ分けを行なった。LINE や facebook messenger のようなコミュニケーションアプリを *messenger*，メールアプリを *mail*，Instagram や Twitter のような SNS アプリを *social*，ソーシャルゲームアプリを *game*，Yahoo ニュースやグノシーのようなニュースアプリを *news*，地図アプリを *map*，お店からのクーポンアプリやポイントアプリを *shop*，Android OS 由来のアプリを *system*，Google chrome

表 3 実験結果概要

	項目	結果
概要	総データ数	95,910
	受信件数	68,261
	削除件数	27,649
通知アクション別内訳	通知クリック (開封)	3,506
	単一の通知削除	12,490
	グループごとの通知削除	3,643
	全ての通知削除	92
	ステータスバーからの削除	5,144
	ステータスバーからの全て削除	2,677
	通知パッケージの変更	83
	通知チャンネルの Ban	6
	通知のタイムアウト	8

のようなブラウザアプリをを *browser*, キャッシュレス決済アプリをを *payment*, Google Play Music や Spotify のような音楽アプリをを *music*, 電話アプリをを *phone call*, Strava などのダイエットアプリや健康アプリをを *lifelog*, その他のアプリをを *other* と定義し分類分けを行なった。

表 4 にカテゴリ毎のユーザの詳細なアクション数を示す。メッセージャーがもっともクリック数が多く、グループごとの削除も多かった。メールはメッセージャーよりもポスト数が多いが、クリック数が少なくシングルデリートが多かった。このことからメールはユーザにとって開きたくない通知が多いということが考察できる。マネーアプリのポストに対するクリック数が多く、レスポンスレートがもっとも高い結果となった。クラウドソーシング登録者は、お金に関する関心が高く、このカテゴリーは通知を開きやすいコンテンツであることが考察できる。

データの中には OS システムに関する通知はも多く含まれていたが、スマホ上には通知として表示されないものや、アプリのダウンロード動作中等に NotificationListenerService があたかも複数回の通知を受信したかの様に連続してログをキャッチしてしまう現象が多く見られた。

また、ライフログはポスト数が多いがリムーブアクションがほとんど

表 4 カテゴリ毎のユーザアクションの数

	post	click	singe delete	group delete	all delete	status bar delete	status bar all delete	others
messenger	17, 854	1, 559	659	2, 931	0	1, 236	211	11
mail	18,902	712	2,748	1,651	1	1, 412	973	14
social	6,001	263	58	1, 266	0	636	219	9
money	3,946	746	81	251	1	462	270	17
game	1,132	82	4	17	78	279	77	7
news	770	4	0	4	0	15	111	4
map	3,964	22	0	331	0	44	102	7
shop	664	23	2	17	1	147	77	5
system	13,209	72	77	4, 132	7	486	172	4
browser	1, 552	5	0	196	0	146	364	6
payment	45	0	0	0	0	14	8	0
music	100	3	0	1	0	2	39	1
phone call	284	3	0	103	0	5	11	0
life log	21, 068	2	0	27	1	4	2	0
others	6, 419	10	14	1, 563	3	256	41	12

見られないことから、ロギングのためにバックグラウンドでのサイレント通知をポストすることでアプリを起動し続ける仕組みが実装されている可能性が示唆できた。

■ 通知クリックまでの経過時間 通知を受信してからそのアプリがクリックされるまでの経過時間を元に考察を行う。図 5 に通知の受信からそのアプリがクリックされるまでの経過時間のヒストグラムを示す。図からクリックが観測された通知の内、そのほとんどが受信してから約 1 時間 (3600s) 以内にクリックされていることがわかる。よって以降は経過時間が 1 時間以内の通知クリックを対象とした解析を行う。

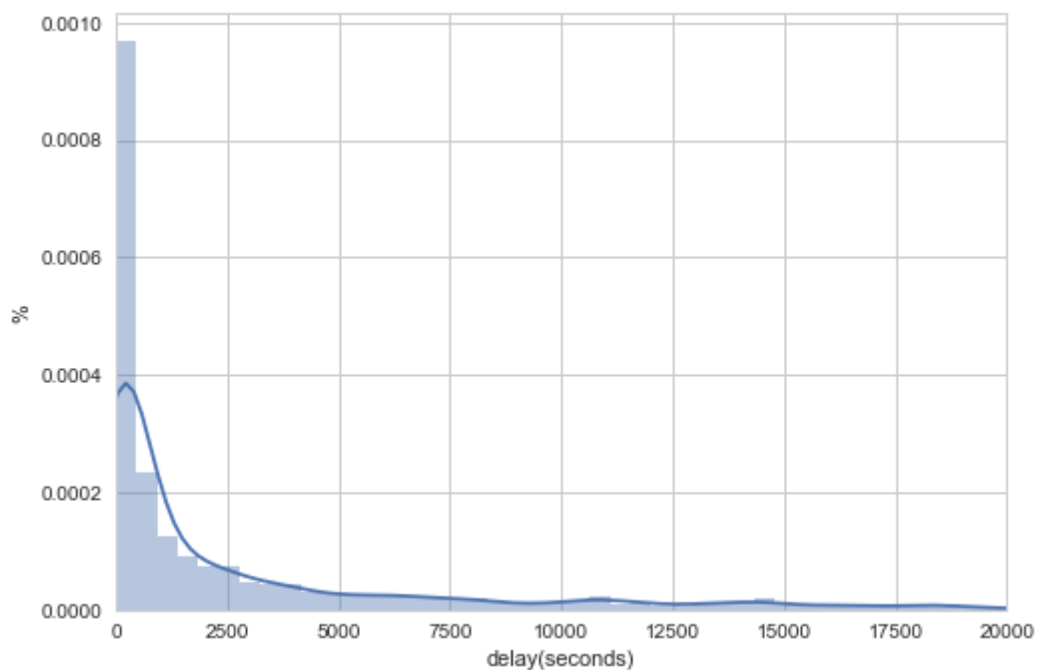


図 5 通知クリックまでの経過時間ヒストグラム

■ 通知クリックの誘起要因に関する考察 図6のように，ユーザによるクリックイベントはそのアプリ自身の通知によって必ずしも引き起こされているとは限らない．例えば，a3のアプリの通知を受信した時タイミングで，以前に受信していたb1の通知を思い出してクリックする可能性も検討しなければならない．表5にユーザが開封したある通知に対して，直前に受信した通知が同一のアプリである場合と異なるアプリであった場合の通知数を集計した結果を示す．これより，多くの通知がクリックの直前に他のアプリからの通知を受け取っており，クリック率を予測する上で，実際にクリックを誘起した直前の通知が何であったかを考慮する必要があるということが示唆された．

表 5 3600s 以内にクリックされた通知の直前の通知

アプリの種類	直前の通知が同一のアプリ	直前の通知が他のアプリ
messenger	1075	384
mail	181	456
social	68	139
money	157	516
game	4	66
news	0	46
map	6	15
shop	0	21
system	36	28
browser	1	4
payment	0	0
music	2	1
phone call	1	1
lifelog	0	2
Others	3	5
Total	1,534	1,684

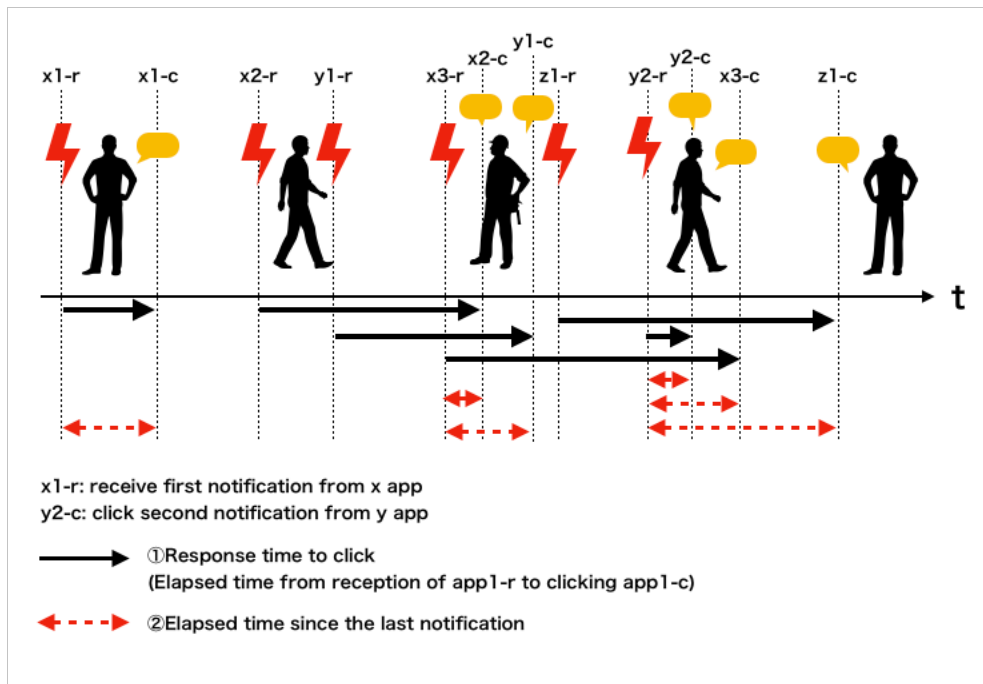


図 6 直前の通知からの経過時間

3.3.2 スポットごとの結果と考察

■ スポットごとのクリック率 ユーザの位置情報を一定の範囲ごとに解析するために、取得した緯度経度の位置情報から GeoHex と呼ばれる尺度への変換を行った。これは世界中を Hex(正六角形) で隙間のなく埋め尽くし、世界中の全ての地点を表現するための尺度である [18]。本研究では、GeoHex Python ライブラリ¹を用い、GeoHex のレベルを中心間距離が約 2.7km となるレベル 6 として変換を行った。変換後のスポットに対しデータ数が多い順に、上からユーザが頻繁に滞在しているスポットとして ID の振り分けを行った。id1 は、それぞれのユーザがもっとも頻繁に滞在していたスポット、id2 はその次に頻繁に滞在していたスポットを表している。変換したスポットの滞在頻度の上位 5 地点におけるクリック率を図 7 に示す。グラフより、id2 は全スポットの中でも 2 番目に高頻度で滞在している場所にも関わらず、クリック率が最も低くなっていることがわかる。

¹<https://pypi.org/project/py-geohex3/>

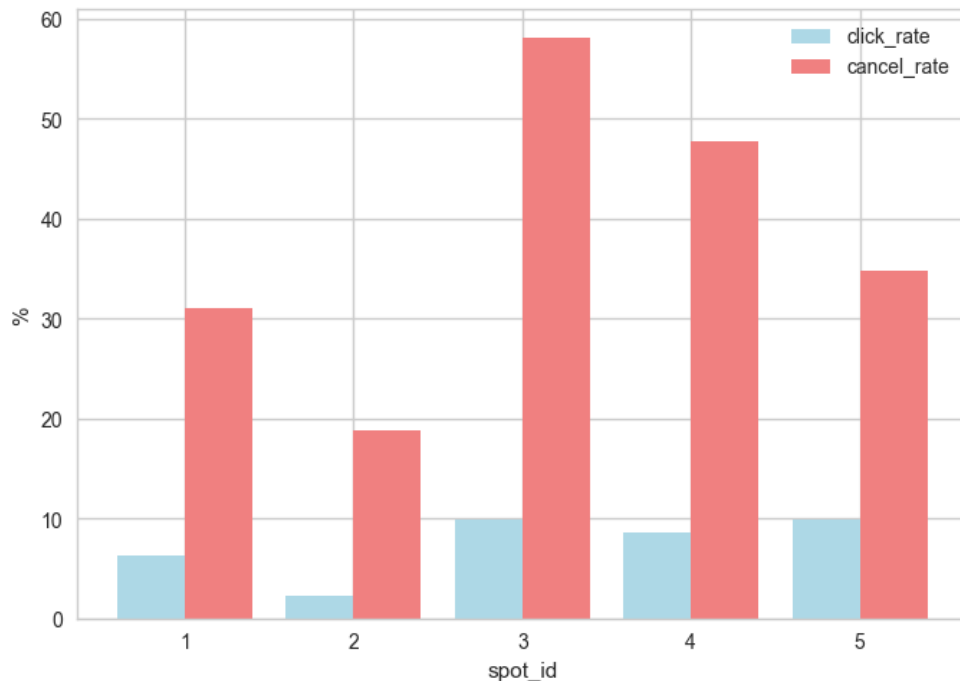


図 7 通知受信上位 5 スポットでのクリック率

■ スポットごとの経過時間 CDF スポットごとの通知クリックまでの経過時間 CDF を図 8 に示す。id2 はスポットにおけるクリックまでの経過時間が最も長かったことから、図 7 の id2 における応答率と合わせて、ユーザにとって二番目に高頻度で滞在するスポットは最も通知に応答するのが困難なスポットになる可能性の高いことがわかった。

■ ユーザとスポットの関係性に関する考察 滞在頻度上位 5 位のスポットと時間の関係を 図 9 に示した、スポットの時間ごとの分布から、id1 は深夜によく滞在しているスポット、id2 は統計的にユーザが日中に高頻度で滞在しているスポットになっていることが見てとれる。このことから id1 はユーザの自宅、id2 はユーザにとっての職場や学校などである可能性が高いということが考察できる。

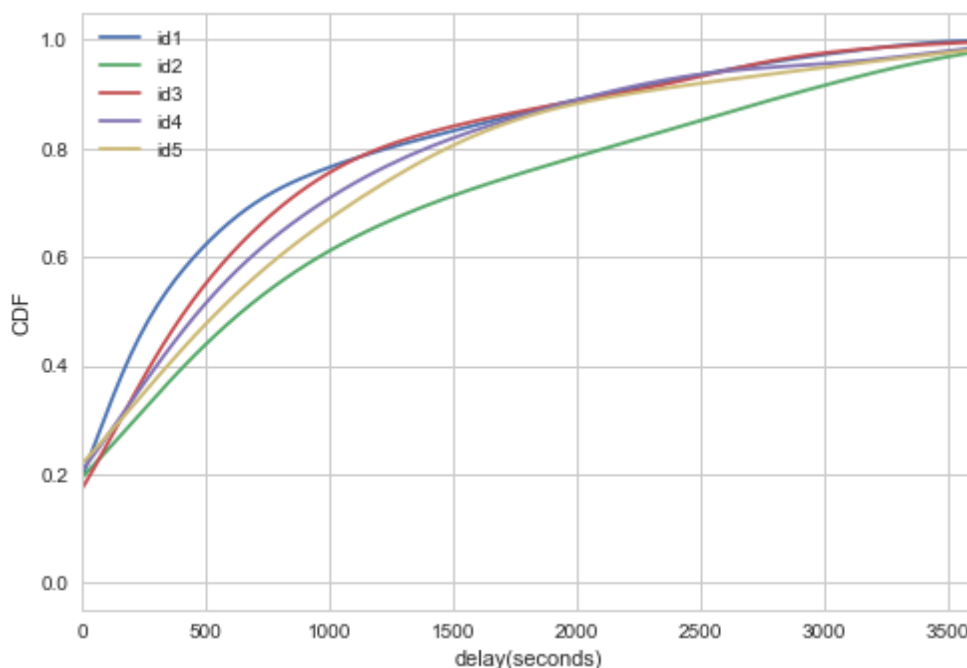


図 8 上位 5 スポットでの通知クリックまでの経過時間 CDF

3.3.3 時間帯ごとの結果と考察

■ 時間帯ごとのクリック率 時間帯ごとのクリック率を図 10 に示す。グラフから通知の受信数は日中の 12 時と 18 時から 20 時にかけてピークになっていることがわかる，しかしながら，通知に対するクリック数に関しては一般的な人の睡眠時間である 1 時から 7 時ごろにかけては減少しているものの，それ以外の時間はクリック数にあまり変化が見られないことがわかる。このことから，ユーザが各時間においてクリックできる通知数に関しては上限があり，それ以上通知を送ってもクリック率は下がってしまう可能性が高いことが考察できる。

3.3.4 この章のまとめ

本調査では，アプリを 7 つのカテゴリに分けて調査を行なったが，その中でもマネーアプリにおいては応答率が高いという傾向が，メッセージアプリにおいて

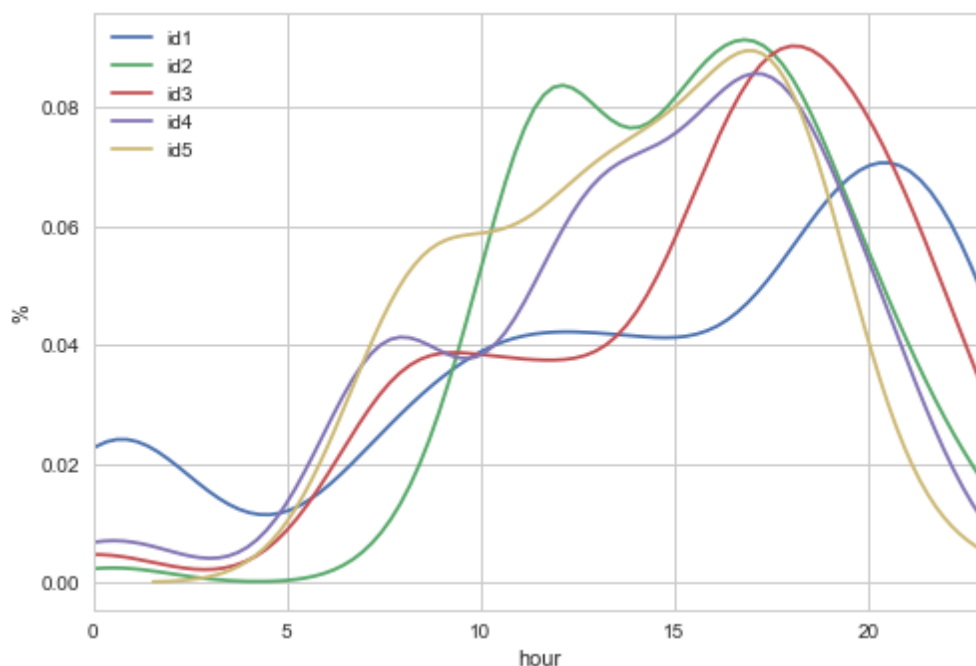


図 9 時間に対するスポットの分布

は通知クリックまでの経過時間が短いという傾向が顕著に見られた。このことから、アプリのカテゴリ分けは適応的タイミング制御において通知の応答性を予測する上で有効な指標になり得ることが分かる。

表 5 から、アプリの通知クリックアクションは直前の他の通知に誘起されている場合も多いと考察できる。そのため、単一のアプリだけでなく、直前に通知された全てのアプリと通知開封との相関関係を利用し、通知応答をより効果的に誘起するための制御方法を提案できる可能性がある。

また、ユーザにとって二番目に高頻度で滞在するスポットは最も通知に応答するのが困難なスポットになる可能性が高いことが今回の調査で明らかになった。そのスポットを動的に避けるモデルを制御システムに組み込むことで応答率の向上に繋がると考える。

最後に、ユーザが起きている間の単位時間あたりに通知に応答できる数はおおよそ上限が決まっていることが本調査で新たに明らかになった。このことから、

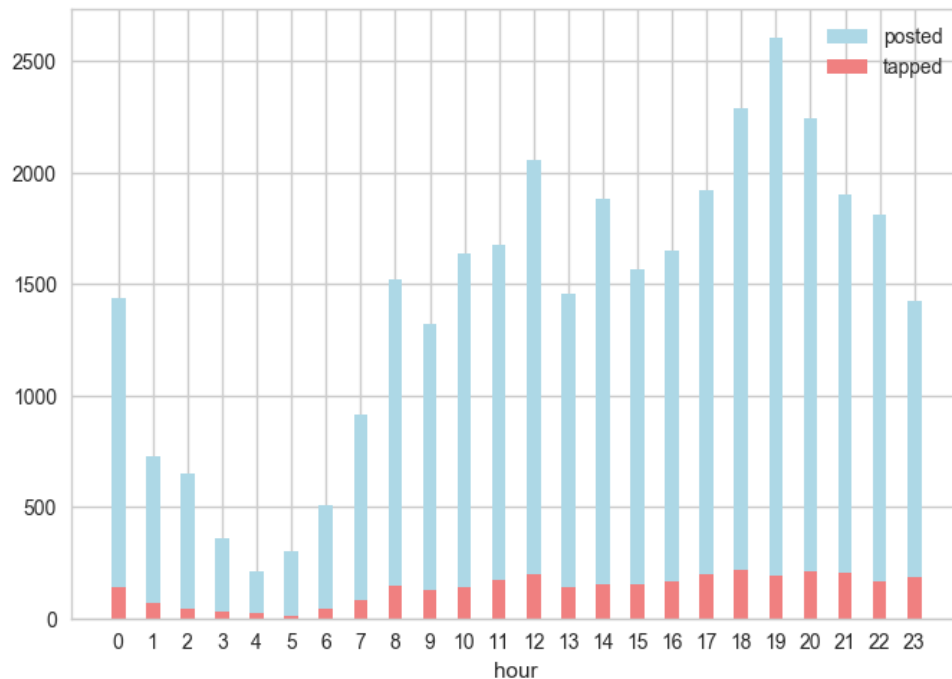


図 10 通知受信時間帯ごとの通知数とクリック数

「夕方の帰宅タイミングに通知する」といったある時間を狙い撃ちする通知手法はそれほど効果的ではなく，ユーザ個人ごとに単位時間あたりの通知開封数の上限予測が可能になれば，他のアプリからの通知も合わせたリアルタイムな通知の受信状況を考慮し，上限数に近い時間帯は避けて後ろの時間に通知の発火タイミングをずらすなど，適応的な制御によってクリック率をあげられる可能性があることがわかった．

4. 制御モデルの構築とその評価

3章では、ユーザの端末が受診する全ての通知データを用いて統計的な解析を行い、全ユーザで共通する通知反応の傾向を確認した。本章では、クリックされやすいタイミングを狙って、通知発火タイミングを制御するために、3章の実験データに対して、受信した通知に対してユーザがクリック挙動を行うかどうかを予測するモデルの構築と評価を行うことで、適応的通知タイミング制御の実現に向けた考察を行う。まず3.1節において、機械学習による通知クリックの予測方法について説明する。次に3.2節で、全ユーザデータを統合したデータセットを用いて機械学習を行なった予測結果を示し、3.3節において、ユーザ個人ごとの予測結果を示す。最後に3.4節で予測モデルの性能評価とディスカッションを行う。

4.1 通知クリックアクションの予測方法

被験者から得られたデータより、以下の7つの属性値と、正解値として通知に対する被験者のアクションを分類分けしたデータを入力値として、教師あり学習を用いて通知クリックの予測を行う。

- アプリのカテゴリ (表4の15種類)
- 通知受信時のユーザの行動 (Google Activity Recognition APIによって認識された、表1の6種類の行動)
- ユーザが通知を受信した時間 (24時間)
- ユーザが通知を受信した分 (60分)
- ユーザが通知を受信した曜日 (月火水木金土日)
- ユーザが通知をスポット (3.3.2節で定義した、ユーザの滞在頻度が高い上位4スポット)

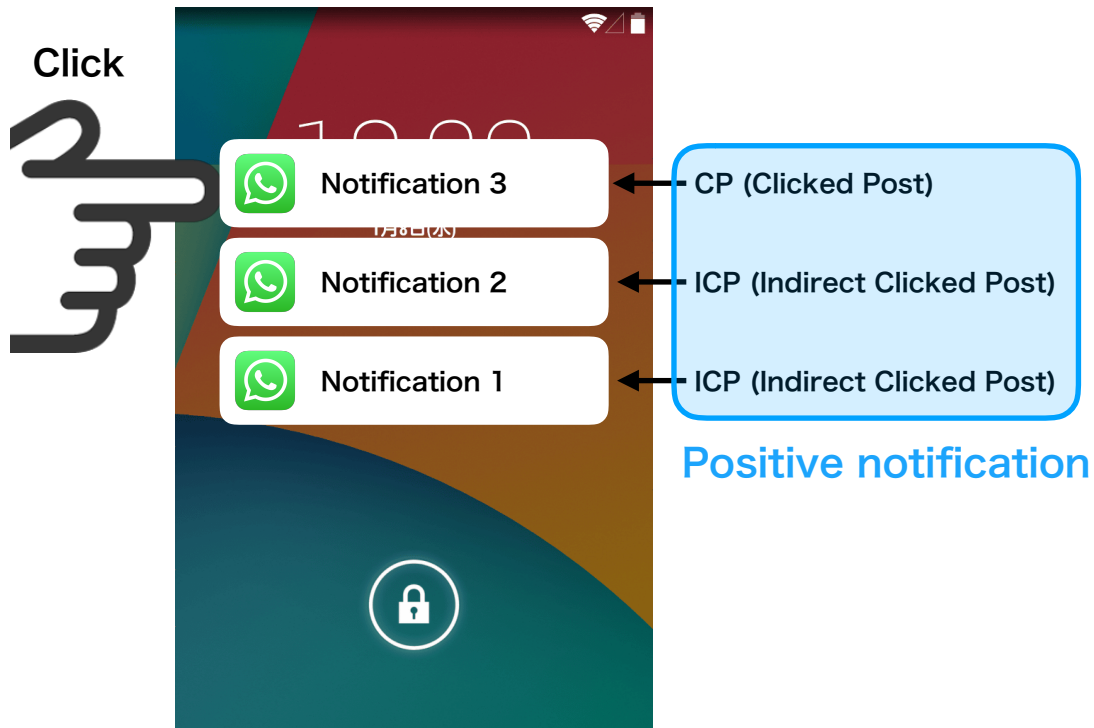


図 11 ポジティブな通知の正解値ラベル定義

- 前回のユーザアクションからの通知ストック数 (前回ユーザが通知をクリックまたは削除した時点からの通知の受信件数)

ユーザの中には実験期間中、極端に通知のクリック数が少ないユーザがいたため、そのユーザ達をデータから除外し、14名のデータに対し機械学習を適用させた。

4.2 全ユーザのデータセットを用いた予測結果

図 11, 図 12 のように取得通知データを CP (Clicked Post), ICP (Indirect Clicked Post), DP (Deleted Post), IDP (Indirect Deleted Post) の 4 段階に分類し、そのうち受信通知どのパターンに属するかを予測した。

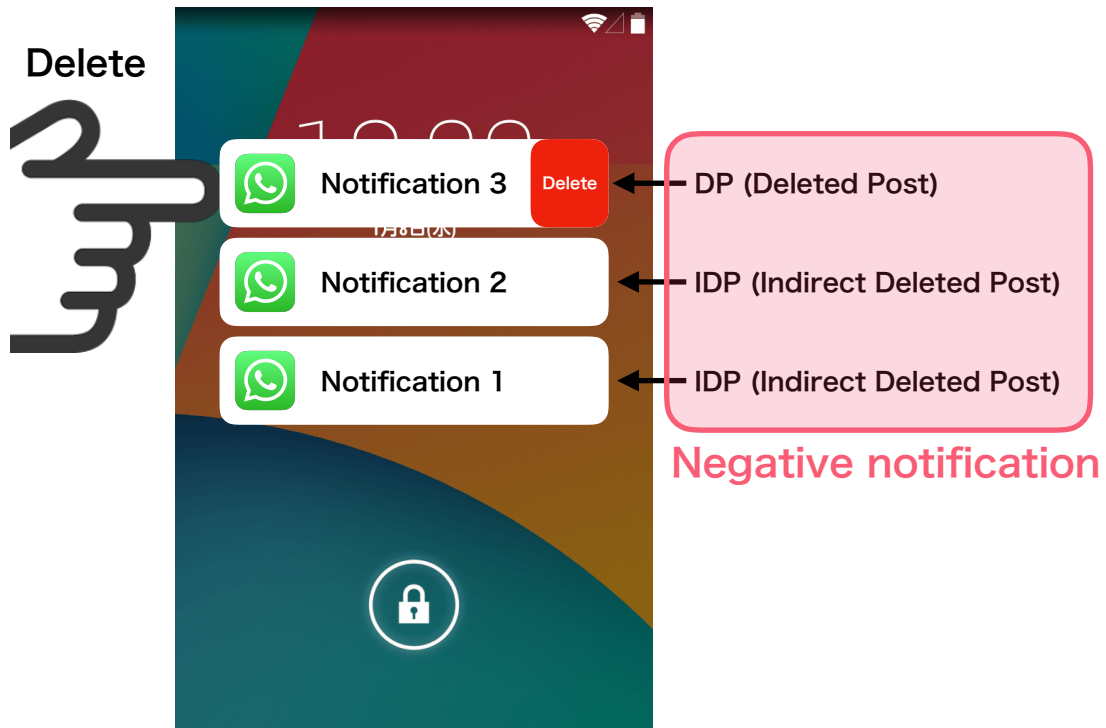


図 12 ネガティブな通知の正解値ラベル定義

まず，全対象ユーザ 14 人のデータに対し，SVM，Random Forest，GBDT アルゴリズムを使用し予測を行なった．尚，全体の通知数 38,779 件に対し，通知クリック数は 2,984 件と数が少なく，本研究においてはクリックにおける 0 番ラベルの Recall(再現率)をもっとも重要な値として考察を行う．各分類器による機械学習の結果結果を表 6 に示す．Random Forest アルゴリズムの 0 番ラベルの Recall 精度が圧倒的に高く，約 27% で通知クリックを予測することができた．

図 13 に Random Forest の Confusion matrix を示す．左の図は最後のユーザアクションからの通知の蓄積数をデータセットに加えた場合の結果で，右図は加えなかった結果である．この結果から，通知の蓄積数をパラメータに加えることで，予測精度が 15% 近く向上することがわかった．これは，ユーザの端末側で受信している全通知データを利用しなければ実現ができない結果であり，端末側で蓄積

表 6 各分類器による機械学習の結果

		CP	ICP	DP	IDP
Random Forest	recall	0.268	0.155	0.714	0.643
	precision	0.297	0.195	0.640	0.699
	F1-score	0.282	0.173	0.675	0.670
Gradient Boosting Decision Tree	recall	0.062	0.022	0.859	0.536
	precision	0.481	0.391	0.572	0.743
	F1-score	0.111	0.042	0.687	0.623
SVM	recall	0.020	0.037	0.801	0.649
	precision	0.333	0.432	0.612	0.684
	F1-score	0.039	0.068	0.694	0.666
Chance Rate		0.077	0.053	0.459	0.406

されている通知データを使用することで、通知のクリックされやすいタイミングの予測精度が向上するという発見は本研究の大きな貢献であると言える。

通知されるクリックの得られた予測精度は理想的な値であるとはいえないが、図??からも、ポストされた通知からクリックされる可能性のある通知をランダムに選んだ場合の実測値は7.7%であることから、提案モデルを用いることで、ランダムな実測値よりも約3倍の精度で予測できることが確認できた。

4.3 各ユーザ毎のクリックアクションの予測結果

4.3.1 4つの正解ラベルの予測予測

次に、個別のユーザデータでのクリックアクション予測を行う。14名の被験者それぞれのデータに対し、Random Forest アルゴリズムによる通知クリックの予測を行なった。もっとも高い精度が得られたユーザの予測結果を図14の Confusion matrix で示す。もっとも精度が高いユーザにおいて63%の精度でクリックされる通知を予測できることが確認できた。

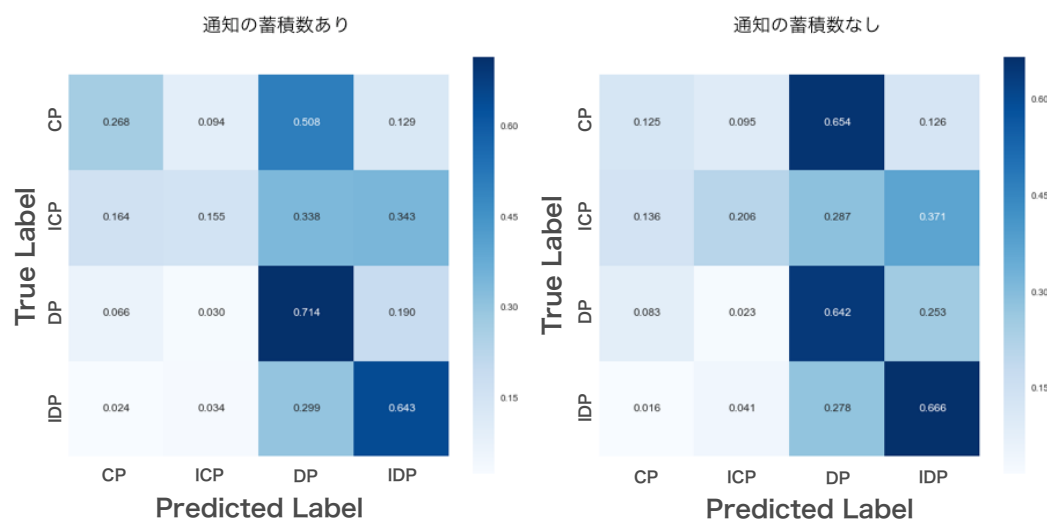


図 13 全ユーザのデータセットを用いたクリック予測の Confusion matrix

4.3.2 ポジティブな通知又は、ネガティブな通知であることの予測

4.2 節の結果からも分かるように、蓄積されている通知と実際に通知がクリックされるかどうかは相関がある。そのため、プッシュされた通知を発火させるべきか否かを判定するアルゴリズムの実環境での実装を考える際、その通知が直接すぐにクリックされることがなくてもユーザに後々同アプリからの通知をクリックさせるために、一度発火させ、蓄積すべきかどうかの判定も重要になる。そこで、これまで使用していた4つのラベルの内、図 11 より、直接クリックされた通知 CP とそのクリックにより開かれた通知 ICP のラベルを**ポジティブな通知**、図 12 より、直接デリートされた通知 DP とそのデリートによって一緒にデリートされた通知 IDP のラベルを**ネガティブな通知**とし、機械学習により、post された通

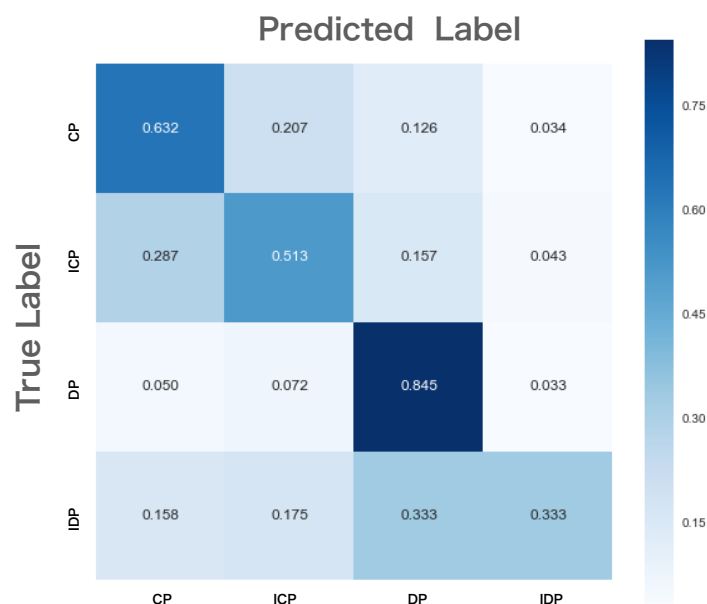


図 14 個人での予測に対する Confusion matrix

知がどちらに分類されるかの予測を行なった．ここでは，より実環境での使用を想定し，毎日 24 時にその日までに蓄積されたデータを学習し，翌日以降の通知を予測するという様に，段階的な学習におけるモデルの評価を行う．例えば，先ほどのユーザの 30 日間のデータにおいて，1 日目をトレーニングデータとし，2 日目から 8 日目をテストデータ，1 日目と 2 日目をトレーニングデータとし，3 日目から 9 日目までをテストデータ，1 日目，2 日目，3 日目をトレーニングデータとし，4 日目から 10 日目までをテストデータとし学習を 23 日目まで繰り返し，その経過日数に対する学習精度の評価を行う．

対象の全被験者に対し，ポジティブな通知の経過日数に対する予測結果を図 15 に示す．user1 だけ，予測精度が高く，日数が経過するとともに上下にブレながらも右肩上がりに予測精度が向上していることが確認できた．若干右肩上がりになっているように見えるユーザもいるが，その他のほとんどのユーザは精度が横這いになっており，日数が経過してもうまく学習ができていない様に見えた．

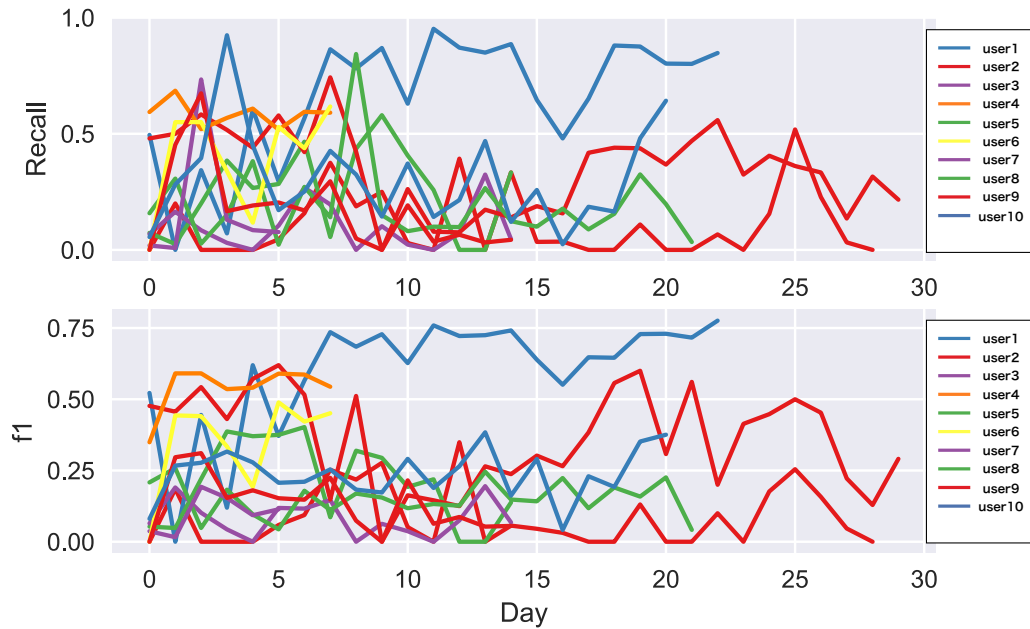


図 15 経過日数に対するポジティブな通知予測結果

4.3.3 ポジティブな通知の学習データ数と予測精度

4.3.2 項でうまく学習を行えなかった原因として、ポジティブな通知データが十分に蓄積されていなかったために、十分な学習を行えていない可能性があると考え、蓄積クリックデータ数ごとの予測精度の評価を行った。全ての対象被験者に対し、ポジティブな通知数が10件増えるごとにその時点までのデータをトレーニングデータ、その後50件ポジティブな通知が蓄積されるまでをテストデータとし予測を行なった。

結果を図16に示す。これから、4.3.3 項で日数に対して予測精度が良かったユーザのデータはポジティブな通知件数が十分にあり、学習する度、徐々に予測精度が向上していることが分かる。一方予測精度が悪かったユーザに関しては上記のユーザと比較して圧倒的に学習回数が少なく、データ不足により学習回数が足り

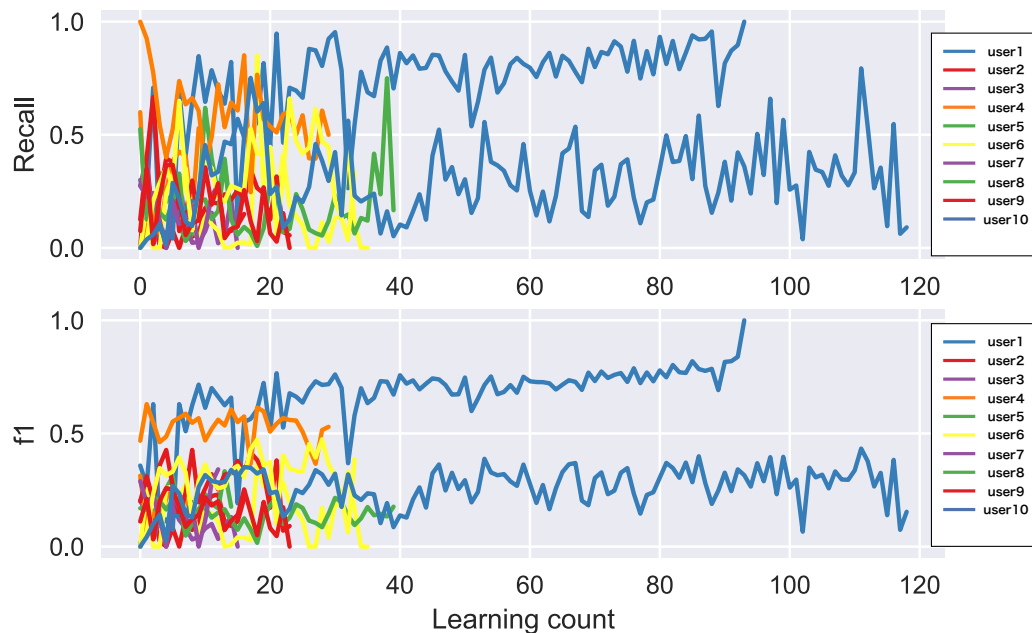


図 16 データ数を元にした学習回数に対するポジティブな通知予測結果

ていなかったために予測精度が十分でなかったことが確認できた。

4.4 考察

本章では、被験者から収集した通知データを用いて、機械学習による通知のクリック予測および、通知クリックに繋がるポジティブな通知の予測モデルを構築した。初めに、全ユーザのデータに対して SVM, Random Forest, GBDT アルゴリズムを使用しクリックされる通知の予測を行なった。この結果より、最後のユーザアクションからの通知の蓄積数をデータセットに加えることで、予測精度が大幅に向上することが確認できた。ここから、通知の蓄積数が増えることで、ユーザがスマホのロック画面に溜まっている通知のことを気にすることにより、

クリック動作やまとめた削除動作を行なったりするシチュエーションが考えられ、それらユーザの主観的な感情が予測精度に直結している可能性が本研究で示唆できた。本研究ではクリックに対するユーザの主観的な感情等を調査することはできなかったため、ユーザに対して都合の悪い割り込みを減らすという価値を提供することができるかは定かではないが、少なくともアプリの配信者側としては、現在の通知蓄積数を考慮し、もっともクリックされやすいタイミングに通知の発火をずらすといった端末側での制御手法を提案できた。そして、今後ユーザの通知に対する主観的な感情を考慮することができるようになれば、よりユーザエクスペリエンスの高い通知制御を実現できる可能性があることがわかった。

次に、各ユーザ毎にクリック予測を行なった。正解ラベルをクリックに向けて通知すべきポジティブな通知とクリックに繋がらないネガティブな通知とし、提案モデルで毎日学習を繰り返すことで、ポジティブな通知の予測精度がどれだけ向上していくかの調査を行なった。結果、通知のクリック件数が十分に収集できているユーザに関しては経過日数ごとに少しずつ予測精度が向上していく現象が観測できたが、同時に、本研究の被験者のほとんどが、実験期間内に学習を行うための十分なデータを得られていなかったことが明らかになった。頻繁に通知をクリックするユーザに関しては、1週間程度で本提案モデルによる制御が適用可能であるが、頻繁にクリックを行わないユーザは2週間から3週間程度でも十分に学習できないことがわかった。

5. 結論

5.1 本研究のまとめ

本研究ではまず、適応的通知タイミング制御の実現に向け、ユーザが受け取る通知をプライバシーを考慮し安全に調査できるシステムを開発し、ユーザが受信する全通知の多様性とそれぞれに対するユーザの挙動を調査した。クラウドソーシングで募った20人の被験者を対象に、4週間の調査実験を行い95,910件の通知に関するデータを収集し、統計解析を行なった。通知をアプリの種類ごとに15のカテゴリに分類し、各カテゴリ毎にユーザが行なった通知に対するアクションを解析した結果、メッセージングアプリのクリック数がもっとも多いことが確認できた。メールは通知件数に対してクリック数が少なく、通知単体でのデリートアクションが多かった。また、お金に関係するアプリの通知応答率が最も高いことがわかった。通知のクリック挙動の直前の通知に着目したところ、多くの通知クリック挙動を引き起こしている直前の通知は同一アプリからの通知ではなく、別のアプリからの通知であることがわかった。この時点で、通知制御において、直前の通知挙動を考慮することの重要性を示唆できた。また、GeoHexと呼ばれる位置情報の尺度変換を用いて、ユーザにとって滞在率が高いスポットに対する通知応答性の解析を行なった。そして、ユーザにとって2番めに高頻度で滞在しているスポットでは通知のクリック率および反応速度が大きく低下することが確認できた。スポットの重要度と滞在時間帯を比較した結果、2番目のスポットはユーザにとって職場や学校に当たる場所である可能性を確認できた。また、時間帯に対する応答性調査を解析した結果、通知の受信数には大きな偏りがあるものの、通知クリック数はほぼ横並びで変わらないことが明らかになった。

次に、収集したデータを用い、7つの属性値(アプリのカテゴリ、通知受信時のユーザ行動、受信時間(時間、分)、受信した曜日、受信時の滞在スポット、前回のアクションからの通知蓄積件数)と正解値(通知に対するユーザの4つのアクション)を入力値として、3つの機械学習アルゴリズムによる通知のクリック予測を行なった。結果、全ユーザデータを用いた場合、クリックの予測精度が約26.8%となり、実測値の7.7%に対して高い精度が得られた。また、7つの属性値

のうち、前回のアクションからの通知蓄積件数をデータセットに加えることで、予測精度が15%も向上することが確認できた。最後に各ユーザ毎の通知クリック予測を行なった。ここでは、実践的な環境を想定して、発火させるべきであった通知を「ポジティブな通知」、発火させるべきでなかった通知を「ネガティブな通知」とし、通知データを毎日学習させ、ポジティブな通知の予測結果を経過日数で評価した。結果、一部のユーザは約1週間を超えたあたりから予測精度が向上し始め3週間目には約80%から90%の精度でポジティブな通知を予測することが可能となった。一方他の多くの被験者は精度が経過日数に対して横ばいの結果となった。ポジティブな通知を一定数得る度に学習を行うというモデルを構築して予測を行なった結果、上記で高い予測結果を得られなかったユーザのほとんどが、データ数不足により上手く学習できていなかったことが確認できたと同時に、もう少し長期で実験を行い、十分なデータ数を確保できれば、予測精度を向上させることができる可能性も示唆できた。

5.2 本研究の課題と今後の展望

5.1 節であげたように、もう少し長期間でのデータ収集によりパフォーマンスを向上させられる可能性はある。また、図16では学習の度に予測精度のぶれが多く見られたが、Random Forestのような分類器は少数派クラスを誤って分類しても、データセット全体の予測精度を最適化するアルゴリズムであるため、発生頻度の低いクラスに対して重み付けを行ったり、強化学習等を用いて、間違った予測を行った際に負の報酬を与える等の工夫をすることで、予測精度が向上する可能性があり、まだまだモデルに改善の余地があると考えられる [19] [20]。

本研究では主に、通知がクリックされる可能性が高いタイミングを探る端末側での制御手法を提案したが、理想の制御モデルとしては、中立な立場でユーザにとって作業を中断させないタイミング且つなるべくアプリの通知がクリックされるタイミング制御を行うことであり、ユーザ視点での適切な割り込みタイミング制御とアプリ配信者視点でのタイミング制御のトレードオフ問題は未だ解決できていない。今後は、本研究で提案したアプリのカテゴリ分け手法を用いて、競合した場合にユーザ体験を損なうような相性の悪いアプリ通知の組み合わせを避

けるといった制御手法の検討，また，ユーザのコンテキスト，特に今回提案した制御モデルを用いて発火させた通知タイミングに対するユーザの感情のように，心理面でのコンテキストデータも考慮した中立的な制御モデルの構築を行う必要がある。「ユーザが不快に思い，通知許可のオフに繋がりそうな通知を発火させない」といった機能を持ったオープンな通知制御アプリを開発することで，上記のようなトレードオフ問題の解決に繋がる可能性があると考える．

謝辞

本研究を進めるにあたり、安本慶一教授には、お忙しい中、様々なご指導及びご指摘をいただきました。また、研究に関してたくさんの助言を頂いただけでなく、個人的に行なっていた、課外活動へも理解を示していただき、温かく見守って頂いたおかげで、充実した学生生活を送ることができました。心から感謝申し上げます。

清川清教授には、ご多用にも関わらず、論文審査委員を務めてくださり、心より感謝申し上げます。また、中間発表にて、有用なコメントをくださりましてありがとうございました。

荒川豊准教授には、本研究を進めるにあたり、研究アイデアや数え切れないほどの助言とサポートを頂きました。研究の進め方や発表練習、論文添削においても多くの知見をいただき、本当に多くの学びを得ることができました。また、学生起業を行った私のことをよく理解して頂き、事業に対しても多くのアイデアやサポートを頂きました。おかげさまで、なんとか2年間の学生生活を終えることができました。心より感謝を申し上げます。

諏訪博彦助教には、本研究を進めるにあたり、解析方法の助言や研究の進め方に関するアドバイスを多く頂きました。また、研究活動以外でも様々な相談をさせて頂きました。心より感謝申し上げます。

藤本まなと助教には、研究室でもフランクに声を掛けてくださり、研究からプライベートなことまでたくさんの相談に乗って頂きました。途中で終わってしまい少し心残りではありますが、介護施設の研究に関してもたくさんのサポートや助言を頂きました。心より感謝申し上げます。

金岡恵事務補佐員、山内奈緒事務補佐員、尾川恵理事務補佐員、南あずさ事務補佐員には、諸々の書類の手続きを進めてくださり、研究生活を支えていただきました。謹んでお礼申し上げます。

私生活においてもさまざまな場面で大変お世話になった本学ユビキタスコンピューティングシステム研究室の同期・先輩・後輩の皆様へ、心から感謝の気持ちとお礼を申し上げたく、謝辞にかえさせていただきます。

参考文献

- [1] engadget. App store shrank for first time in 2017 thanks to crackdowns on spam, clones and more. <https://japanese.engadget.com/2018/07/17/app-store-google-play-2/>.
- [2] Urban Airship. How Push Notifications Impact Mobile App Retention Rates. http://grow.urbanairship.com/rs/313-QPJ-195/images/WP_App_Retention_Rates_Benchmarks.pdf.
- [3] E. Cutrell, M. Czerwinski, and E. Horvitz. Notification, disruption, and memory: Effects of messaging interruptions on memory and performance. *Proceedings of the INTERACT 2001*, No. 1999, pp. 263–269, 2001. doi: 10.1002/j.2168-0159.2014.tb00200.x.
- [4] Brian P Bailey and Joseph A Konstan. On the need for attention-aware systems : Measuring effects of interruption on task performance , error rate , and affective state, Vol. 22. pp. 685–708, 2006. doi: 10.1016/j.chb.2005.12.009.
- [5] Apple. iOS12. <https://www.apple.com/jp/ios/ios-12/>.
- [6] Google. Android 9 Pie. <https://www.android.com/versions/pie-9-0/>.
- [7] Tadashi Okoshi, Kota Tsubouchi, Masaya Taji, Takanori Ichikawa, and Hideyuki Tokuda. Attention and engagement-awareness in the wild: A large-scale study with adaptive notifications. *2017 IEEE International Conference on Pervasive Computing and Communications, PerCom 2017*, pp. 100–110, 2017. doi: 10.1109/PERCOM.2017.7917856.
- [8] 赤池 勇磨, 荒川 豊, 諏訪 博彦, and 安本 慶一. 参加型センシングの効率化に向けたコンテキストに基づく応答の推定. *情報処理学会論文誌*, Vol. 57, No. 2, pp. 543–552, 2016.

- [9] Gloria Mark, Shamsi T. Iqbal, Mary Czerwinski, Paul Johns, Akane Sano, and Yuliya Lutchyn. Email Duration, Batching and Self-interruption. *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*, pp. 1717–1728, 2016. doi: 10.1145/2858036.2858262.
- [10] Stephanie Rosenthal, Anind K Dey, and Manuela Veloso. Using Decision-Theoretic Experience Sampling to Build Personalized Mobile Phone Interruption Models 1 Introduction.
- [11] Nitesh Goyal and Susan R Fussell. Intelligent Interruption Management using Electro Dermal Activity based Physiological Sensor for Collaborative Sensemaking, Vol. 1, No. 3. pp. 1–21, 2017.
- [12] Samaneh Aminikhanghahi, Ramin Fallahzadeh, Diane J Cook, and Lawrence Holder. Thyme : Improving Smartphone Prompt Timing through Activity Awareness, No. November, 2017. doi: 10.1109/ICMLA.2017.0-141.
- [13] Bo-jhang Ho, Los Angeles, Mehmet Koseoglu, and Los Angeles. Nurture : Notifying Users at the Right Time Using Reinforcement Learning, 2018.
- [14] Android Developers. 2018. NotificationListenerService. <https://developer.android.com/reference/android/service/notification/NotificationListenerService.html>.
- [15] Dominik Weber and Alexandra Voit. Notification Log : An Open-Source Framework for Notification Research on Mobile Devices. pp. 1271–1278.
- [16] Alireza Sahami Shirazi, Niels Henze, Tilman Dingler, Martin Pielot, Dominik Weber, and Albrecht Schmidt. Large-scale assessment of mobile notifications. *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*, pp. 3055–3064, 2014. doi: 10.1145/2556288.2557189.

- [17] Martin Pielot, Karen Church, and Rodrigo De Oliveira. An In-Situ Study of Mobile Phone Notifications. pp. 233–242, 2014.
- [18] @sa2da. GeoHexx- Hexagonal Geo-coding System. <https://sites.google.com/site/geohexdocs/>.
- [19] Marcus A Maloof. Learning When Data Sets are Imbalanced and When Costs are Unequal and Unknown, 2003.
- [20] Xu-ying Liu. The Influence of Class Imbalance on Cost-Sensitive Learning : An Empirical Study. pp. 1–5, 2006.