

# スマートフォンの通知の取られ方にもとづく ユーザおよび端末の状態識別手法

澤野 亮太<sup>1</sup> 村尾 和哉<sup>1</sup>

**概要:** さまざまなセンサを搭載したスマートフォンやウェアラブルデバイスの普及により, 時間や場所を問わず人間の行動や生体情報, 周囲の状況を認識できるようになった. センサデータから人間の行動や状況を推定する処理は, 正解情報 (アノテーション) が付与されたデータセットを用いてユーザの行動や状況を解釈するモデルを事前に構築する必要がある. そのため, モデルの高性能化や性能評価のために多量かつ多様なアノテーション付きデータセットが必要となる. 加速度センサの値は後から見てユーザの状況を判断することは困難なため, 加速度データにアノテーションを付与する手法が必要である. 本研究では, スマートフォンなどの端末が生成する通知に対するユーザの対応からユーザや端末の状況を推定する手法を提案する. 通知発生から削除までの応答時間による状態推定と端末内の加速度センサの値を利用した状態推定を併用して, ユーザや端末の状態推定を行い, 推定結果の確信度がきわめて高い場合にセンサデータへのアノテーション付与を行う. 評価実験より, アノテーションを付与したい7つの状態に対して, ユーザ非依存とユーザ依存の実験を行い, 平均適合率 0.769 と 0.963 を得た. また, 自然環境における実験も行い, ユーザが反応した 45 個の通知に対して, 25 個に正しいアノテーションが付与され, 19 個にアノテーション付与を行わず, 誤って付与したアノテーションは 1 個であった.

## 1. はじめに

さまざまなセンサを搭載したスマートフォンやウェアラブルデバイスの普及により, 時間や場所を問わず人間の行動や生体情報, 周囲の状況を加速度 [8] や角速度, 光, 脈拍, 位置, 電波状況, 筋電計 (EMG) [16], 心電計 (ECG) [6], 皮膚電極反射 (GSR) [12], 自作のデバイス [11] を用いて認識できるようになった. 人間の行動をセンシングして得られる情報は, 生活パターンを自動抽出し, 過労や運動不足を警告する健康管理システム [12], 組立作業において現在の動作から次の動作を予測して手順書や必要機材を提示する作業支援システム [23], 病院外における患者の服薬や血糖値計測の時刻を医師が正確に把握する医療支援システム, タックルやスプリントの回数及び強度を取得するスポーツ支援システム<sup>\*1</sup>, 歩容や機器の操作にもとづく本人認証, 動作で機器を操作する入力インタフェースやゲーム [18], などの分野においてこれまでは手作業や人間の勘, 人海戦術で行われてきた作業, 実現困難であった高度な作業の支援への応用が期待されている.

センサデータから人間の行動や状況を推定する処理は, 正解情報 (アノテーション) が付与されたデータセットを

用いてユーザの行動や状況を解釈するモデルを事前に構築し, 未知のセンサデータを識別する. そのため, モデルの高性能化や性能評価のために多量かつ多様なアノテーション付きデータセットが必要となる. センサデータのみならずサーバやクラウドストレージに Wi-Fi や携帯網経由で低コストで収集できるが, アノテーションはセンサデータ収集とは別のタスクで収集する必要がある. アノテーションは正確性が重要であり, 実験主催者が被験者を監視できる場合や被験者が小型カメラを装着できる場合は正確なアノテーションを収集できるが, アプリを配布して一般ユーザから幅広くデータを収集する環境では正確でないアノテーションが付与される可能性がある. また, 画像や音声, テキストは人間が理解できるため, あとから人間がデータを見聞きしてアノテーションを付与できるが, 加速度データを見てユーザの状況を判断することは困難であるため, センサデータを採取しながらアノテーションを記録する必要がある. しかしながら, ユーザ本人が記録したアノテーションは正確性が低く, 研究者が帯同して記録したアノテーションは正確であるがスケールできないという問題がある.

一方で, 端末が扱うデータの多様化, 高品質化, 大容量化によって, スケジュールのリマインダ, ソーシャルメディアの友人からのメッセージ, 天気予報, 最新ニュースなど

<sup>1</sup> 立命館大学 情報理工学研究科

<sup>\*1</sup> CATAPULT, <https://www.catapultsports.com/>

の通知が絶え間なく発生する。いくつかの通知はユーザに即座に取られるが、多くの通知は眠っていてスマートフォンから離れているなどの理由でユーザが気づかない、またはプレゼンテーション途中や満員電車の中などの理由で取ることができない。オンラインショッピングやニュースのアプリの通知に対する反応率は売上高や広告費に影響するため、通知の開封率を上げるための研究も注目されている。

本研究ではスマートフォンなどの端末が生成する通知に対するユーザの対応からユーザや端末の状況を推定する手法を提案する。通知に対するユーザの応答時間と端末内の加速度センサの値からユーザや端末の状況を推定し、きわめて確信度が高い推定結果をアノテーションとしてセンサデータに付与する。採取したデータすべてにアノテーションを付与する必要はなく、採取したデータのうち僅かでも自動的にアノテーションが付与できれば大規模データセットの自動構築を実現できる。通知が取られるかどうかはユーザやデバイスの状況に関係していると考えられる。通知の頻度が増加したことで、通知への応答をセンサとして利用できる。一般的な人間の行動認識では継続的またはアプリケーションの任意のタイミングでの認識結果が必要となる。通知の発生は散発的であるが、ラベル付けされていないデータは学習に使用しなければよいだけであって、アノテーションが散発的に付与されても構わない。これらの理由から、アノテーションと通知は相性が良いと考えている。

## 2. 関連研究

### 2.1 加速度データへのアノテーション付与手法

センシングデータにアノテーション（ラベル）を付与する方法として、センシングデータ採取中の行動をビデオカメラなどで撮影し、あとから動画とセンシングデータの波形を比較して照合する方法や、紙とペンあるいはスマートフォンで時刻と行動を記録する方法などがある。動画を用いる手法では動画とセンシングデータを同時に見ながら手作業でラベリングを行う必要があり、収集したデータの何倍もの時間を要する。メモを用いる手法では、センシングデータ収集中に記録作業を行うため、データに不要なノイズが含まれる。また、行動するたびに記録しなければならないという煩わしさもある。ラベルを付与する作業は多くの時間と人手を要するため、行動認識システム構築の障壁となっている。

また、これまでに提案されている行動認識システムの多くは、すべてのデータにラベルが付与されたデータを用いてシステムの学習を行っている。すべてにラベルが付いたデータで行う学習は教師あり学習（(Full-)Supervised learning）と呼ばれる。これに対して、部分的にしかラベルが付与されていないデータを用いて行う学習は半教師あり

学習（Semi-supervised learning）と呼ばれる。教師あり学習も半教師あり学習も最終的な行動の認識結果を得るために分類器を使用する。このような分類器は、すべてのデータにラベルが付与されている必要があり、付与されていないデータは一般的に破棄される。そのため、半教師あり学習では、一部のラベル付きデータを用いて残りのラベルがっていないデータのラベリングを行い、すべてのデータにラベルを付与したのち、教師あり学習と同じプロセスをとる。ラベルは推測であるため、必ずしも正しいとは限らない。半教師あり学習の研究として、文献 [15] は複数の半教師あり学習の手法を行動認識に適用して性能を比較しており、その中でシンプルな手法である Self-training [17] が紹介されている。Self-training は少量のラベル付きデータと大量のラベルなしデータがある場合に、少量のラベル付きデータで認識モデルの構築を行い、そのモデルを用いてラベルなしデータを分類し、分類結果をラベルとして付与する。これによってすべてのデータにラベルが付与され、全てのデータを用いて認識モデルの再構築を行う。

文献 [14] では、特徴量空間上で類似しているデータや時間空間上で類似しているデータは同じラベルである確率が高いことに着目し、すべてのデータをノード、類似度をエッジとしたグラフを作成し、特徴量および時間の距離からデータの類似度を計算する。ラベル付きデータと類似度が高いラベルなしデータには同じラベルを与えることでラベルを拡散している。その際には、データを構成する行動の種類の前分布の情報を利用し、拡散するラベルの数が前分布に従うようにラベリングしている。評価では事前に与えるラベルの間隔を 10 分から 180 分まで変化させてラベリング精度を計測しており、10 分間隔では 90%、180 分間隔では 55% の精度が得られている。10 分間隔は全データ量の 2.5%、180 分間隔は全データ量の 0.1% がラベル付けされていることに相当する。

また、固有空間（Eigenspace）を利用した研究が提案されている [22]。多次元データの次元削減手法などに用いられる主成分分析（PCA）で得られる固有空間は 1 つであるが、Multiple eigenspace algorithm を用いると複数の固有空間を発見でき、加速度などのデータに適用すると各サンプルがどこか 1 つの固有空間に割り当てられる。この固有空間と行動の種類に相関があることに着目し、固有空間のインデックスをデータに付与して SVM を学習し、エラー率から固有空間の統廃合を行う。これに外部から少量のラベル付きデータを与えることで、固有空間のインデックスと行動を関連付けてラベル付けを行う評価では、ラベル付きデータの割合が 80% の場合に 8 種類の行動の認識率が 88.3% であり、20% で 80.3% となっている。また、ラベル付きデータのみを用いて学習した場合と比較してもこの手法の方が高い性能を示すことが確認されている。しかし、これらの評価はラベルの量を均等に減らされた環境

で評価しており、ラベリングを行う頻度が減っただけで、センシングデータ採取中の行動を記録しなければならず、根本的な解決にはなっていない。

それに対して村尾ら [13] は、行動が行われた順序の情報のみを用いてラベリング作業を自動化する手法を提案した。提案された方法は、ラベル付けされていないデータをセグメントに分割およびクラスタに分類し、各セグメントにクラスタを割り当て、記録された行動の順序にもっとも一致するようにラベルを割り当てる。この手法は7種類行動を含むデータに対して0.812の適合率を得た。また、提案手法でラベル付けされたデータを学習に用いた際の認識精度は再現率0.871となり、これは正しくラベル付けされたデータを用いた際と比べて同等である。

## 2.2 通知の影響、制御および管理

スマートフォンなどのモバイル端末で発生する通知によるユーザのパフォーマンスへの影響が報告されている。通知はユーザに対する「割り込み」とみなすことができ、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とき、ユーザのストレスの増加や生産性の低下を引き起こすことが報告されている [7], [9]。大越ら [19], [20] は、プッシュ通知を開封しやすいタイミングとしてユーザの行動の境界である Physical Activity Breakpoint に注目し、スマートフォンに内蔵された加速度センサを用いた行動認識の結果からプッシュ通知が開封されやすい Breakpoint を検出し、スマートフォンがプッシュ通知を受信した際に、その通知をブレークポイントまで遅延させてユーザに提示するシステムを構築し、通知の開封率を向上させている。また、大越ら [21] は実サービスでの運用実験も行った。Yahoo! JAPAN の Android アプリにモバイルセンシングと機械学習を用いた推定ロジックを組み込み68万人のユーザに対して実験を行った。結果としてユーザの通知への応答時間の短縮とユーザが週に何回アプリを利用するかを調査しユーザエンゲージメントの改善を確認することに成功した。また、Hoら [4] も通知タイミングをユーザごとに最適化する手法として、ユーザコンテキストを使用したパーソナライゼーション手法を提案している。これらの研究は通知のタイミングを制御することで通知への応答性や通知の開封率を向上させてきた。一方でユーザの趣味や嗜好を読み取り、ユーザ一人ひとりにとって有意義なコンテンツやサービスを提供するための研究も行われている。宣伝用の電子メールやソーシャルネットワーク上のゲームへの招待などのユーザの興味がない通知は開封せずに削除する傾向があることが報告されている [3], [10]。また、ユーザとの関連の低いアプリケーションやアンインストールされるアプリケーションからの通知はユーザを苛立たせることも報告されている [2], [3]。Mehrotraら [1] は、通知タイトルと行動、時間、場所などの組合せからユーザの好みを

検出し、有用でないと判断した通知をフィルタリングし、ユーザにとって最適な通知のみをプッシュするための割り込み管理システムを提案した。

本研究のアプローチは、通知を制御するのではなく、通知をセンサとして使用することである。加速度センサの値は通知によって区切られ、加速度センサの値と応答時間を併用してユーザおよび端末の状況を推定する。提案手法の実現によって人間の行動認識のための大規模ラベル付きデータセットの構築を可能にする。

## 3. 提案手法

本節ではスマートフォンなどの端末が生成する通知に対するユーザの対応からユーザおよび端末の状況を推定する提案手法の詳細を述べる。

### 3.1 想定環境

提案手法の利用者としては、新たに提案した認識アルゴリズムの性能を評価したい研究者や行動認識技術を使用して新しいアプリケーションを開発するアプリケーションエンジニアを想定している。図1に示すように、研究者やエンジニアは Amazon Mechanical Turk などのクラウドソーシングサービスでアルバイトを探したり、知人からボランティアを探し、アルバイトやボランティアはセンサデータと通知への応答を記録できる筆者らの提案アプリをインストールする。そして、スマートフォンの使用中に、ユーザがインストールしている一般のアプリが通知を生成し、その通知がユーザによって削除されると、提案アプリがアプリ名、通知生成時刻、通知削除時刻などの通知に関する情報を記録する。提案アプリの詳細は4章で詳しく説明する。提案アプリはセンサデータと通知への応答を一定時間間隔でサーバまたはクラウドストレージにアップロードする。サーバ上では提案手法が通知への応答時間と加速度センサの値を使用して加速度データにアノテーションを付与する。最後に、研究者はアノテーション付きデータをダウンロードして入手できる。提案アプリをインストールしたスマートフォンに対して、性別や年齢層などのユーザ属性でグループ化された任意のユーザに研究者からダミー通知を生成することもできるため、同一条件下でのユーザの対応を調査することもできる。これにより、ユーザがアプリをインストールするだけで自動的にアノテーション付きデータが収集できる。

### 3.2 提案手法の概要

提案手法の概要を図2に示す。事前に学習フェーズとして通知発生から通知削除までの応答時間と通知発生から通知削除までの区間の端末の加速度データを採取しておき、付与したいアノテーションの種類でラベル付けし、分類器のモデルを構築しておく。次に、推定フェーズとして通知

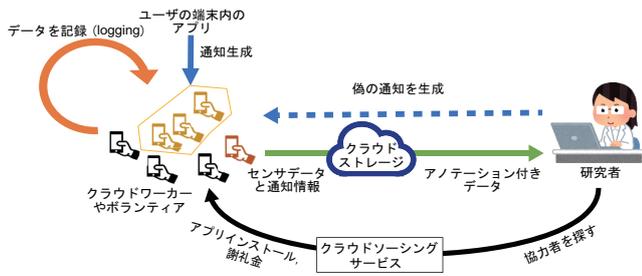


図 1 想定環境

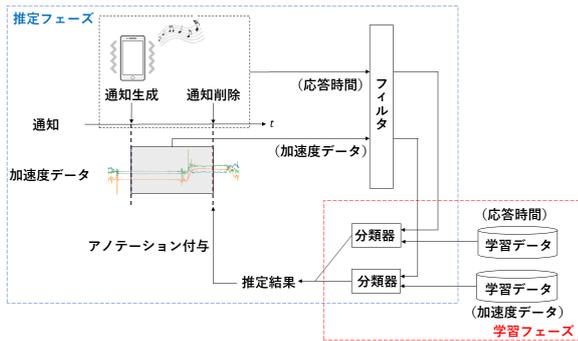


図 2 提案手法の概要

発生から通知削除までの応答時間と応通知発生から通知削除までの端末の加速度データを取得し、応答時間が長いデータは破棄するフィルタを適用する。フィルタリングを通過したデータについて、応答時間と加速度データを並列して、事前に作成した分類器にかけて分類結果を得る。応答時間による分類結果と加速度データによる分類結果が一致し、ともに確信度が極めて高い場合のみ分類結果をアノテーションとしてセンサデータに付与する。

### 3.3 応答時間によるフィルタリング

通知が生成されたとき、スマートフォンはLED、音、振動、および画面上の視覚効果を通して通知をユーザーに知らせ、提案手法はタイムスタンプ  $T_{gen}$  を記録する。ユーザーが通知バーに表示されている通知をスワイプまたはタップして通知を消すと、システムはタイムスタンプ  $T_{rm}$  を記録する。そして、通知生成と通知削除のタイムスタンプの時間差  $T_{diff}$  は次式で計算し、本研究ではこれを応答時間と呼ぶ。

$$T_{diff} = T_{rm} - T_{gen} \quad (1)$$

ユーザーが通知にすぐに気付かずその後で削除した場合は、応答時間が長くなる。例えば会議中などユーザーが通知に気づいたが意図的に通知に回答しなかった場合、通知削除のタイムスタンプ  $T_{rm}$  が記録されないため、応答時間を計算できない。この場合、システムは通知の削除に時間がかかった理由を特定することができないため、提案手法は応答時間が  $T_{filter}$  より長い通知は破棄し、アノテーション付与をあきらめる。 $T_{filter}$  は事前に採取したすべてのアノ

テーションの応答時間の平均であり、本研究では10秒とする。

次に、フィルタを通過した通知について、通知生成から通知削除までの加速度データ

$$\begin{aligned} & [[x(T_{gen}), \dots, x(T_{rm})], \\ & y(T_{gen}), \dots, y(T_{rm}), \\ & z(T_{gen}), \dots, z(T_{rm})] \end{aligned}$$

を抽出する。 $x(t)$ ,  $y(t)$ ,  $z(t)$  はそれぞれ、時間  $t$  における加速度の3軸  $x$ ,  $y$ ,  $z$  の値である。

### 3.4 応答時間を用いた状態推定

事前に収集した応答時間からアノテーション付与を行いたい状態ごとに0から10秒の範囲でヒストグラムを作成する。ヒストグラムのビンの幅は0.5秒でビンの数は20とした。各ビンの頻度を  $h(k)$  ( $k = 1, \dots, 20$ ) とすると、応答時間による尤度は以下の式で求められる。

$$L(T_{diff}) = \frac{h(\text{ceil}(T_{diff}/0.5))}{\sum_{k=1}^{20} h(k)} \quad (2)$$

ただし、 $\text{ceil}(x)$  は天井関数であり、 $x$  を  $x$  以上の最小の整数を出力する。例えば、 $\text{ceil}(1.25) = 2$  となる。具体的には、100サンプルのデータセットで、応答時間が  $1 < t \leq 1.5$  の範囲にあるサンプル数が5の場合、 $L(1.2) = 5/100 = 0.05$  となる。

提案手法は付与したいすべてのアノテーションクラスの応答時間と尤度を計算する。もっとも高い尤度が閾値を下回る場合、応答時間を用いた状態推定結果は空集合  $\Phi$  とする。複数のアノテーションクラスの尤度が閾値を超えると、それらすべてのアノテーションクラスが応答時間を用いた状態推定の結果となる。各アノテーションクラスの応答時間の分布が重ったり、分布が広がる可能性があるため、少なくとも誤った推定結果は出力しないように閾値は低く設定する。本研究では正しい結果を破棄しないように閾値は0.05とする。

### 3.5 加速度データを用いた状態推定

データマイニングを行うためには、時系列データ間の類似度計算が必要である。類似性を計算する簡単な方法として、ユークリッド距離があるが、ユークリッド距離は時間的歪みを受けやすく、2つのデータのサンプル数が等しくなければならないといった問題点がある。本研究では加速度データを用いた状態推定を行うために Dynamic time-warping (DTW) [5] を用いる。DTWとは、動的時間伸縮法とも呼ばれ異なる2つの時系列波形の類似度を算出するアルゴリズムである。DTWの特徴として、時間的非線形圧縮距離を計算することで時系列のスケールや部分

的な速度に違いがあるデータ間やサンプル数の異なるデータ間の類似度を求めることができる。提案手法では入力加速度データと学習加速度データとの DTW 距離を計算する。DTW のアルゴリズムの詳細は [5] を参照していただきたい。

提案手法は、事前に収集したアノテーション付与を行いたい状態と入力された加速度データとの DTW 距離を計算する。計算された最短の DTW 距離が閾値よりも大きい場合、加速度データを用いた状態推定結果は空集合  $\Phi$  となる。それ以外の場合は、DTW 距離が最短の状態が推定結果となる。

実際の生活では、想定している状態以外の状態で通知が取られることが多いと考えられる。学習していない加速度データが提案手法に入力されると、すべてのアノテーションクラスとの DTW 距離が大きくなり、提案手法はそのなかで最短の DTW 距離のクラスを認識結果とするため、誤った結果が出力される。想定外の状態が通知が取られても間違った結果を出力しないために閾値を設定する。閾値は事前に収集したデータにおいてすべてのデータの組合せの DTW 距離を計算する。そして、あるデータに対して最小の DTW 距離となるもう一方のデータのアノテーションクラスが同一となる組合せにおける DTW 距離の  $\mu + 2\sigma$  と設定した。  $\mu$  と  $\sigma$  はそれぞれ対象の DTW 距離の集合の平均と標準偏差である。例えば、アノテーションの種類が 3 つで、各アノテーションごとに 5 サンプルの学習データがあるとすると、1 つのサンプルと自分を除く 14 サンプルの DTW 距離が計算できる。このうち最小となる DTW 距離を示す両サンプルが同一のアノテーションであれば前述の平均と標準偏差を計算する対象とする。これを 15 サンプルすべてに対して行う。

### 3.6 アノテーション付与

提案手法では応答時間を用いた状態推定の結果と加速度データを用いた状態推定の結果から最終的な状態推定の結果およびその結果をアノテーションとして出力するか否かを判断する。アノテーション付与を行うかどうかの判断の例を表 1 に示す。応答時間を用いた状態推定の結果と加速度データを用いた状態推定の結果が共通の状態の場合、その状態推定結果をアノテーションとして加速度データに付与する。これは表 1 のケース #1 と #2 に相当する。応答時間を用いた状態推定の結果と加速度データを用いた状態推定の結果に共通する状態がない場合、アノテーション付与は行われない。これは表 1 のケース #3 から #7 に相当する。

表 1 状態推定結果によるアノテーション付与の例 (C1, C2, C3 はアノテーション付与を行いたい状態,  $\Phi$  は空集合, - はアノテーションなし)

ケース	応答時間	加速度	アノテーション
#1	C1	C1	C1
#2	C1, C2	C1	C1
#3	C2	C1	-
#4	C1, C2	C3	-
#5	$\Phi$	C1	-
#6	C1	$\Phi$	-
#7	$\Phi$	$\Phi$	-

## 4. 実装

Android OS には Notification Listener Service<sup>\*2</sup> という API が存在し、通知に関する情報を取得できる。提案アプリは API を通して、通知の種類、内容、通知生成時刻、通知削除時刻、通知発生時の音、通知発生時のバイブレーション設定のオン/オフなど、本研究で使用する情報以外にも取得できる。提案アプリをインストールするとき、ユーザはアプリへの通知アクセスをオンすることを明示的に要求される。ストレージやカメラなどへの一般的なアクセス許可の確認とは異なる。アプリはバックグラウンドで実行され、ユーザインタフェースとデータ取得に分離されている。タスクリストからアプリを終了すると、ユーザインタフェースのみが終了し、データ取得はバックグラウンドで実行され続ける。その後アプリを再起動すると、ユーザインタフェースが起動し、データ取得は影響を受けない。ただし、アプリ情報画面で「無効にする」、「強制停止」、「アンインストール」を選択すると、データ取得も終了する。提案アプリは、加速度データと通知データを収集する。また、提案アプリは Google の Firebase Cloud Messaging (FCM)<sup>\*3</sup> の受信を可能としている。FCM は、メッセージを無料で配信するクロスプラットフォームメッセージングソリューションである。FCM を使用すると、IOS, Android, および Web でメッセージや通知を送受信できる。取得した通知情報のデータと加速度データは Firebase Cloud Storage に保存される。

## 5. 評価

提案手法の有効性を検証するために提案手法を用いたアノテーション付与精度を評価する。評価実験は 2 種類行い、1 つ目はアノテーションを付与したい複数の状態それぞれでデータを採取してアノテーション付与の精度を評価する実験室環境の実験である。2 つ目はアノテーションを付与したい状態を考慮せずに日常生活で 2 日間採取したデータ

\*2 <https://developer.android.com/reference/android/service/notification/NotificationListenerService>

\*3 <https://firebase.google.com/>

に対して提案手法を適用したときの結果をみる自然環境の実験である。自然環境の実験では被験者には普段の生活を行ってもらい、アノテーション付与を行いたい状態以外でのアノテーション付与の精度を検証するために行う。

## 5.1 実験室環境の実験

### 5.1.1 実験環境

被験者は20代男性5名である。アノテーション付与を行いたい状態として(1)立ってスマートフォンを手に持っている、(2)机の上にスマートフォンを置いて使用している、(3)机の上にスマートフォンを置いて使用していない、(4)スマートフォンを手に持って使用している、(5)立ってポケットの中にスマートフォンを入れている、(6)歩きながらスマートフォンを手に持っている、(7)歩きながらスマートフォンを使用している、の7種類を想定する。また、想定外の状態が通知が取られた場合のアノテーション付与の精度を調査するために、想定外の状態として(8)胸ポケットの中にスマートフォンを入れている、(9)ベッドの上にスマートフォンを置いて使用していない、(10)バッグの中にスマートフォンを入れている、の3種類を考える。

被験者にはスマートフォン(ASUS ZenFone3, ZS570KL, Android 8.0.0, 加速度のサンプリングレート 400Hz)を用いて(1)~(10)の状態において計算機(Lenovo ThinkPad X1 Carbon, OS Windows 10)からFCMを用いて通知を意図的に各状態約100回ずつ発生させた。収集したデータの数は100回×10状態×5被験者=5,000個であり、そのうち10秒以内に削除された4,868個を評価に使用した。被験者には通知に気づくと可能な限りすぐに通知を削除するように指示した。通知は音と振動で知らせるようにしてすべての通知はスワイプにより消された。通知が発生した時刻から通知をスワイプで消すまでの時間を応答時間として取得した。同時に通知発生から削除までの加速度データも取得した。

評価では(1)~(7)のデータを学習データ、(1)~(10)のデータをテストデータとして使用し、被験者同士で行う交差検証(ユーザ非依存)と被験者ごとにデータを分けて、その中で交差検証(ユーザ依存)の2パターンで検証した。比較手法として、応答時間だけを用いる場合と加速度だけを用いる場合の2種類を検証した。応答時間だけを用いる比較手法は、応答時間にもとづいて最大尤度の状態を見つけ、その尤度が閾値(提案手法と同じ0.05)以上の場合に出力する。加速度だけを用いる比較手法は、加速度データにもとづいて最短のDTW距離の状態を見つけ、DTW距離が閾値(提案手法と同じ学習データの $\mu + 2\sigma$ )以下の場合に出力する。

### 5.1.2 ユーザ非依存の結果

ユーザ非依存評価における、2つの比較手法と提案手法の7種類のアノテーションクラスに対するアノテーション

付与の適合率(Precision)、再現率(Recall)、およびF値(F-measure)を表2に示す。加えて詳細な考察を行うために、3つの手法の混同行列を図3、図4および図5に示す。「Unk.」はシステムがアノテーション付与を行わない数を示している。本研究の目的はアノテーション付与であるため、適合率が最も重要な指標となる。アノテーションをすべてのデータに付与する必要はないため再現率はそれほど重要ではないが、センサデータに付与されたアノテーションは正確でなければならないため、高い適合率が求められる。

結果から、応答時間を用いた比較手法では適合率は最大0.309、平均0.222で、加速度を用いた比較手法では適合率は最大0.936、平均0.743であった。提案手法では適合率は最大0.990、平均0.769でありもっとも良い結果となった。しかし、平均0.769の適合率では付与したアノテーションの2.3割は誤っていることを意味し、アノテーション付与の性能としては高いとはいえない。ユーザ非依存評価の結果から、応答時間の分布および通知を削除する際の行動はユーザによって異なることが分かる。

### 5.1.3 ユーザ依存の結果

ユーザ依存評価における、2つの比較手法と提案手法の7種類のアノテーションクラスに対するアノテーション付与の適合率(Precision)、再現率(Recall)、およびF値(F-measure)を表3に示す。加えて詳細な考察を行うために、3つの手法の混同行列を図6、図7および図8に示す。

結果から、応答時間を用いた比較手法では適合率は最大0.473、平均0.400で、加速度を用いた比較手法では適合率は最大0.998、平均0.930であった。提案手法では適合率は最大1.00、平均0.963であり、3つの手法で最も高い適合率を得ることができた。

### 5.1.4 考察

図3と図6より、応答時間を用いた比較手法は出力された結果が対角線上のセル以外に分布しており、多くのアノテーション付与が誤っていることが分かる。想定外の状態(8)、(9)、および(10)では約500の入力データのうち50から183程度しか「Unknown」に分類されていない。これは、いくつかのアノテーションクラスの応答時間の分布が重なり合っており、応答時間だけで状態を特定することは困難であると分かる。

ユーザ依存評価における加速度を用いた比較手法の結果、図7より、入力(1)から(7)に対する出力の多くが正しいことが分かる。しかし、想定外の状態(9)および(10)に対しては90.7%および75.4%で「Unknown」と推定された。これは想定外の動作が行われた際に正しくアノテーション付与を行わないことを意味する。一方で、ユーザ非依存評価における加速度を用いた比較手法の結果、図4より、状態(6)と(7)の歩行中の状況で多くの誤出力が生じた。これは歩行動作が人によって異なるためであると考

表 2 ユーザ非依存評価での比較手法と提案手法のアノテーション付与性能

アノテーションクラス	応答時間			加速度			提案手法 (応答時間+加速度)		
	Precision	Recall	F-measure	Precision	Recall	F-measure	Precision	Recall	F-measure
(1) 立って手に持っている	0.108	0.139	0.121	0.497	0.831	0.622	0.509	0.632	0.564
(2) 机の上で使用している	0.246	0.088	0.130	0.883	1.00	0.938	0.891	0.919	0.905
(3) 机の上で使用していない	0.136	0.186	0.157	0.587	0.882	0.705	0.534	0.524	0.529
(4) 手に持って使用している	0.308	0.272	0.289	0.567	0.909	0.698	0.618	0.780	0.690
(5) 立ってポケットの中	0.231	0.524	0.321	0.905	0.916	0.911	0.990	0.628	0.769
(6) 歩きながら手に持っている	0.309	0.543	0.394	0.823	0.208	0.332	0.879	0.178	0.295
(7) 歩きながら使用している	0.214	0.285	0.244	0.936	0.451	0.609	0.960	0.398	0.562
平均	0.222	0.291	0.237	0.743	0.742	0.688	0.769	0.580	0.616

えられる。

最後に、図 8 のユーザ依存評価における提案手法の「Unknown」と分類された結果より、提案手法によって間違っていたアノテーションが付与されていたものが少なくなっていることが分かる。例として、図 7 では状態 (6) は (1)、状態 (10) は (3) と推定されている。状態 (1) と (3) の応答時間の分布は、ユーザが静止して手にスマートフォンをもっているため早くなる。入力データの加速度値が状態 (1) または (4) に近い場合でも、応答時間の分布が異なるため、図 8 に示すように提案手法では「Unknown」と分類できている。つまり、加速度による状態推定だけでは省くことができなかったケースも応答時間を用いた状態推定と併用することで省くことができ、提案手法と比較して適合率を 0.033 向上させることができた。

## 5.2 自然環境の実験

### 5.2.1 実験環境

被験者は実験室環境の実験に参加した男性 5 人のうち 3 人 (A,B, および C) である。被験者には 2 日間スマートフォン (ASUS ZenFone3, ZS570KL, Android 8.0.0, 加速度のサンプリングレート 400Hz) を使用して、普段通りの生活を行ってもらった。通知は FCM と Python 言語を用いたプログラムにより計算機 (Lenovo ThinkPad X1 Carbon, OS Windows 10) から 30 分ごとに発生させた。被験者には通知に対応した状態が (1)~(7) の状態ならば当てはまる状態を記録してもらい、(1)~(7) 以外の状態ならばその他とスマートフォンに記録してもらった。実験室環境の実験で収集したデータを学習データとして使用し、日常生活で採取したデータをテストデータとした。他の被験者のデータのみを学習に使用する (ユーザ非依存) の場合とテストデータの被験者を含む全員のデータを学習に使用する (ユーザ依存) の 2 パターンの実験を行う。

### 5.2.2 結果

通知の生成、削除、10 秒以内に削除された数、およびユーザに対するユーザ依存評価とユーザ非依存評価における提案手法の出力の結果を表 4 に示す。被験者 C につい

↓入力	出カ-	(1)	(2)	(3)	(4)	(5)	(6)	(7)	Unk.
(1)立って手に持っている		65	0	99	0	87	202	13	2
(2)机の上で使用している		61	47	0	154	1	81	186	4
(3)机の上で使用していない		96	0	90	0	132	88	0	77
(4)手に持って使用している		37	39	2	131	0	39	230	4
(5)立ってポケットの中		54	0	79	0	251	2	0	93
(6)歩きながら手に持っている		91	11	18	8	15	266	81	0
(7)歩きながら使用している		16	94	5	130	5	93	139	6
(8)胸ポケットの中		57	0	139	1	180	56	0	51
(9)ベッドの上で使用していない		124	0	134	0	152	33	0	50
(10)カバンの中		1	0	97	1	262	0	1	105

図 3 ユーザ非依存評価での応答時間を用いた比較手法によるアノテーション付与の混同行列

↓入力	出カ-	(1)	(2)	(3)	(4)	(5)	(6)	(7)	Unk.
(1)立って手に持っている		389	0	0	3	32	21	1	22
(2)机の上で使用している		0	534	0	0	0	0	0	0
(3)机の上で使用していない		0	57	426	0	0	0	0	0
(4)手に持って使用している		28	5	2	438	0	0	9	0
(5)立ってポケットの中		13	0	0	2	439	0	0	25
(6)歩きながら手に持っている		323	0	0	1	0	102	1	63
(7)歩きながら使用している		20	0	2	245	0	0	220	1
(8)胸ポケットの中		1	0	0	83	0	0	0	400
(9)ベッドの上で使用していない		0	8	73	0	0	0	0	412
(10)カバンの中		8	1	223	1	14	1	4	215

図 4 ユーザ非依存評価での加速度を用いた比較手法によるアノテーション付与の混同行列

↓入力	出カ-	(1)	(2)	(3)	(4)	(5)	(6)	(7)	Unk.
(1)立って手に持っている		296	0	0	1	3	12	0	156
(2)机の上で使用している		0	491	0	0	0	0	0	43
(3)机の上で使用していない		0	52	253	0	0	0	0	178
(4)手に持って使用している		2	5	0	376	0	0	8	91
(5)立ってポケットの中		9	0	0	0	301	0	0	169
(6)歩きながら手に持っている		272	0	0	1	0	87	0	130
(7)歩きながら使用している		3	0	2	229	0	0	194	60
(8)胸ポケットの中		0	0	0	0	0	0	0	484
(9)ベッドの上で使用していない		0	2	45	0	0	0	0	446
(10)カバンの中		0	1	174	1	0	0	0	291

図 5 ユーザ非依存評価での提案手法によるアノテーション付与の混同行列

ては、被験者周囲のネットワーク状態が悪かったため、5 時間分のデータしか記録されなかったため、5 時間分で評価した。

ユーザ依存評価では、被験者 A では 10 秒以内に削除された 17 回に対して 12 回に正しいアノテーションが付与され、残りの 5 回にアノテーションは付与されなかった。アノテーション付与の適合率は 1.00(12/12)、再現率は 0.706(12/17) であった。被験者 B では 23 回のデータに対して 11 回に正しいアノテーションが付与され、1 回に

表 3 ユーザ依存評価での 2 つの比較手法と提案手法のアノテーション付与性能

アノテーションクラス	応答時間			加速度			提案手法 (応答時間+加速度)		
	Precision	Recall	F-measure	Precision	Recall	F-measure	Precision	Recall	F-measure
(1) 立って手に持っている	0.338	0.679	0.451	0.870	0.987	0.925	0.953	0.908	0.930
(2) 机の上で使用している	0.417	0.360	0.387	0.998	1.00	0.999	1.00	0.955	0.977
(3) 机の上で使用していない	0.324	0.439	0.373	0.755	1.00	0.860	0.865	0.938	0.900
(4) 手に持って使用している	0.452	0.438	0.445	0.897	0.998	0.945	0.927	0.952	0.940
(5) 立ってポケットの中	0.334	0.712	0.455	0.994	0.971	0.982	0.993	0.881	0.934
(6) 歩きながら手に持っている	0.465	0.482	0.473	0.997	0.720	0.836	1.00	0.702	0.825
(7) 歩きながら使用している	0.473	0.424	0.447	0.998	0.924	0.960	1.00	0.879	0.936
平均	0.400	0.505	0.433	0.930	0.943	0.930	0.963	0.888	0.920

↓入力	出力↑	(1)	(2)	(3)	(4)	(5)	(6)	(7)	Unk.
(1)立って手に持っている		318	0	40	2	24	70	8	6
(2)机の上で使用している		20	192	0	140	0	64	117	1
(3)机の上で使用していない		106	0	212	0	124	36	0	5
(4)手に持って使用している		20	139	0	211	0	27	80	5
(5)立ってポケットの中		32	0	80	0	341	1	0	25
(6)歩きながら手に持っている		133	34	32	14	7	236	25	9
(7)歩きながら使用している		31	94	7	100	3	43	207	3
(8)胸ポケットの中		136	0	103	0	157	10	1	77
(9)ベッドの上で使用していない		130	0	147	0	131	21	0	64
(10)カバンの中		16	1	33	0	234	0	0	183

図 6 ユーザ依存評価での応答時間を用いた比較手法によるアノテーション付与の混同行列

↓入力	出力↑	(1)	(2)	(3)	(4)	(5)	(6)	(7)	Unk.
(1)立って手に持っている		462	0	0	3	1	0	0	2
(2)机の上で使用している		0	534	0	0	0	0	0	0
(3)机の上で使用していない		0	0	483	0	0	0	0	0
(4)手に持って使用している		0	0	1	481	0	0	0	0
(5)立ってポケットの中		0	0	0	1	465	1	0	12
(6)歩きながら手に持っている		68	0	0	3	0	353	0	66
(7)歩きながら使用している		1	0	0	33	0	0	451	3
(8)胸ポケットの中		0	0	0	13	0	0	1	470
(9)ベッドの上で使用していない		0	1	45	0	0	0	0	447
(10)カバンの中		0	0	111	2	2	0	0	352

図 7 ユーザ依存評価での加速度を用いた比較手法によるアノテーション付与の混同行列

↓入力	出力↑	(1)	(2)	(3)	(4)	(5)	(6)	(7)	Unk.
(1)立って手に持っている		425	0	0	1	1	0	0	41
(2)机の上で使用している		0	510	0	0	0	0	0	24
(3)机の上で使用していない		0	0	453	0	0	0	0	30
(4)手に持って使用している		0	0	0	459	0	0	0	23
(5)立ってポケットの中		0	0	0	0	422	0	0	57
(6)歩きながら手に持っている		21	0	0	2	0	344	0	123
(7)歩きながら使用している		0	0	0	31	0	0	429	28
(8)胸ポケットの中		0	0	0	1	0	0	0	483
(9)ベッドの上で使用していない		0	0	31	0	0	0	0	462
(10)カバンの中		0	0	40	1	2	0	0	424

図 8 ユーザ依存評価での提案手法によるアノテーション付与の混同行列

表 4 自然環境実験における通知の生成, 削除, 10 秒以内に削除された数, および提案手法の出力

被験者	ユーザの通知への反応			システムの結果					
	生成	削除	削除 (<10s)	ユーザ依存			ユーザ非依存		
				TP	TN	FP	TP	TN	FP
A	67	24	17	12	5	0	1	16	0
B	84	32	23	11	11	1	15	7	1
C	11	8	5	2	3	0	2	3	0

間違ったアノテーションが付与された. 残りの 11 回にアノテーションは付与されなかった. アノテーション付与の適合率は 0.917(11/12), 再現率は 0.478(11/23) であった. 被験者 C では 5 回のデータに対して 2 回に正しいアノテーションが付与され, 残りの 3 回にアノテーションは付与されなかった. アノテーション付与の適合率は 1.00(2/2), 再現率は 0.40(2/5) であった.

ユーザ非依存評価では, 被験者 A では 17 回のデータに対して 1 回に正しいアノテーションが付与され, 残りの 16 回にアノテーションは付与されなかった. アノテーション付与の適合率は 1.00(1/1), 再現率は 0.06(1/17) であった. 被験者 B では 23 回のデータに対して 15 回に正しいアノテーションが付与され, 1 回に間違ったアノテーションが付与された. 残りの 7 回にアノテーションは付与されなかった. アノテーション付与の適合率は 0.938(15/16), 再現率は 0.652(15/23) であった. 被験者 C では 5 回のデータに対して 2 回に正しいアノテーションが付与され, 残りの 3 回にアノテーションは付与されなかった. アノテーション付与の適合率は 1.00(2/2), 再現率は 0.40(2/5) であった.

### 5.2.3 考察

自然環境実験の結果より, 被験者 A はユーザ非依存評価よりユーザ依存評価で多くのアノテーションを付与することができた. これは被験者 B の実験室環境での通知への反応が早かったため, 自然環境における被験者 A の応答時間の分布に適合しなかったためである. ユーザ非依存評価において多数の人からのデータを学習する更なる調査が必要である.

## 6. リミテーション

実生活では通知をとる状況は無数に存在する. 本研究ではその中の一部の状況をアノテーションとしてを採用した. しかし, (5) スマートフォンをポケットに入れている状態, は立って止まっている状態を想定しているため, 座っている状態や歩いている状態では間違ったアノテーションを付与してしまう. 研究者が新しいアノテーションを付与したい場合は, その学習データが必要となり, 状態を増やすこ

とで他の状態の精度に干渉する可能性もある。

## 7. おわりに

本研究では、スマートフォンなどの端末が生成する通知に対するユーザの対応からユーザや端末の状況を推定する手法を提案した。通知発生から削除までの応答時間による状態推定と端末内の加速度センサの値を利用した状態推定を併用して、ユーザや端末の状態推定を行い、推定結果の確信度がきわめて高い場合にのみ、センサデータへのアノテーション付与を行う。評価実験より、アノテーションを付与したい7つの状態に対して、ユーザ非依存とユーザ依存の実験を行い、平均適合率0.769と0.963を得た。また、自然環境における実験も行い、ユーザが反応した45個の通知に対して、25個に正しいアノテーションが付与され、19個にアノテーション付与を行わず、誤って付与したアノテーションは1個であった。今後は長期の自然環境の実験を行い、提案手法がさまざまな想定外の状況で有効かを検証する。また、ユーザ非依存でもアノテーション付与が正しく行われるか更なる検証を行う。

**謝辞** 本研究の一部は、国立研究開発法人科学技術振興機構 (JST) 戦略的創造研究推進事業 ACT-I (17938567) の支援によるものである。ここに記して謝意を表す。

## 参考文献

- [1] Abhinav Mehrotra, Robert Hendley, and Mirco Musolesi. 2016. PrefMiner: Mining User's Preferences for Intelligent Mobile Notification Management. In *UbiComp 2016*. 1223-1234.
- [2] Adrienne Porter Felt, Serge Egelman, and David Wagner. 2012. I've got 99 problems, but vibration ain't one: a survey of smartphone users' concerns. In *SPSM 2012*. 33-44.
- [3] Alireza Sahami Shirazi, Niels Henze, Tilman Dingler, Martin Pielot, Dominik Weber, and Albrecht Schmidt. 2014. Large-scale assessment of mobile notifications. In *CHI 2014*. 3055-3064.
- [4] Bo-jhang Ho, Bharathan Balaji, Mehmet Koseoglu, and Los Mani Srivastave. 2018. Nurture: Notifying Users at the Right Time Using Reinforcement Learning. In *UbiComp 2018*. 1194-1201.
- [5] C. Myers, L. Rabiner, and A. Rosenberg. 1980. Performance tradeoffs in dynamic time warping algorithms for isolated word recognition. In *IEEE Transactions on Acoustics, Speech, and Signal Processing*. 623-635.
- [6] Chien-lung Shen, Tsair Kao, Ching-tang Huang, and Jun-huei Lee. 2006. Wearable band using a fabric-based sensor for exercise ecg monitoring. In *ISWC 2006*. 143-144.
- [7] Edward Cutell, Mary Czerwinski, and Eric Horvitz. 2001. Notification, Disruption, and Memory: Effects of messaging interruptions on memory and performance. In *INTERACT 2001*. 236-239.
- [8] Futoshi Naya, Ren Ohmura, Fusako Takayanagi, Haruo Noma, and Kiyoshi Kogure. 2006. Workers' Routine Activity Recognition using Body Movements and Location Information. In *ISWC 2006*. 105-108.
- [9] Gloria Mark, Shamsi T. Iqbal, Mary Czerwinski, Paul Johns, and Akane Sano. 2016. Email Duration, Batching and Self-interruption: Patterns of Email Use on Productivity and Stress. In *CHI 2016*. 1717-1728.
- [10] Joel E Fischer, Nick Yee, Victoria Bellotti, Nathan Good, Steve Benford, and Chris Greenhalgh. 2010. Effects of content and time of delivery on receptivity to mobile interruptions. In *MobileHCI 2010*. 103-112.
- [11] Kristof Van Laerhoven and Hans W Gellersen. 2004. Spine versus porcupine: a study in distributed wearable activity recognition. In *ISWC 2004*. 142-149.
- [12] Kazushige Ouchi, Takuji Suzuki, and Miwako Doi. 2002. Lifeminder: A wearable healthcare support system using user's context. In *IWSAWC 2002*. 791-792.
- [13] Kazuya Murao and Tsutomu Terada. 2013. Labeling Method for Acceleration Data using an Execution Sequence of Activities. In *HASCA 2013, UbiComp Adjunct*. 611-622.
- [14] Maja Stikic, Diane Larlus, and Bernt Schiele. 2009. Multi-graph based semi-supervised learning for activity recognition. In *ISWC 2009*. 85-92.
- [15] Maja Stikic, Kristof Van Laerhoven, and Bernt Schiele. 2008. Exploring semi-supervised and active learning for activity recognition. In *ISWC 2008*. 81-88.
- [16] Masashi Toda, Junichi Akita, Shigeru Sakurazawa, Keisuke Yanagihara, Mihoko Kunita, and Kunio Iwata. 2006. Wearable Biomedical Monitoring System Using TextileNet. In *ISWC 2006*. 119-120.
- [17] Olivier Chapelle, Bernhard Scholkopf, and Alexander Zien. 2006. *Adaptive computation and machine learning*. In MIT Press.
- [18] Ryo Izuta, Kazuya Murao, Tsutomu Terada, and Masahiko Tsukamoto. 2015. Early Gesture Recognition Method with an Accelerometer. In *International Journal of Pervasive Computing and Communications*, Vol. 11. 270-287. Issue 3.
- [19] Tadashi Okoshi, Julian Ramos, Hiroki Nozaki, Jin Nakazawa, Anind K. Dey, and Hideyuki Tokuda. 2015a. Attelia: Reducing User's Cognitive Load due to Interruptive Notifications on Smartphones. In *PerCom 2015*. 96-104.
- [20] Tadashi Okoshi, Julian Ramos, Hiroki Nozaki, Jin Nakazawa, Anind K. Dey, and Hideyuki Tokuda. 2015b. Reducing users' perceived mental effort due to interruptive notifications in multi-device mobile environments. In *UbiComp 2015*. 475-486.
- [21] Tadashi Okoshi, Kota Tsubouchi, Masaya Tajiri, Takamori Ichikawa, and Hideyuki Tokuda. 2017. Attention and Engagement-Awareness in the Wild: A Large-Scale Study with Adaptive Notifications. In *PerCom 2017*. 100-110.
- [22] Tam Huynh, and Bernt Schiele. 2006. Towards less supervision in activity recognition from wearable sensors. In *ISWC 2006*. 3-10.
- [23] Thomas Stiefmeier, Georg Ogris, Holger Junker, Paul Lukowicz, and Gerhard Tröster. 2006. Combining motion sensors and ultrasonic hands tracking for continuous activity recognition in a maintenance scenario. In *ISWC 2006*. 97-104.