





図 1. 開発したアプリのインターフェース

図 1 に示した画面には 2.2 で定義したユーザの状況名がそれぞれボタンで表示されており、測定する状況のボタンをタップすることでセンサの測定を開始する。本アプリでは、0.1 秒ごとにセンサデータを取得してサーバに送信する。また、複数のセンサを並列に測定することができる。さらに、測定時間や状況の定義をサーバ側で設定することができるため、アプリを再インストールせずに測定条件を変更させることができる。アプリの開発には React Native<sup>[3]</sup>を使用しており、サーバサイドの開発には Django<sup>[4]</sup>を使用した。

## 4. 機械学習によるユーザの状況判別

### 4.1. センサデータの収集

開発したセンサデータ収集用アプリを用いて、実際に 8 名の協力者にセンサデータの測定を行ってもらい、複数のデバイスによるセンサデータの収集を行った。今回、各状況について 150 秒間のデータを測定し、1 回あたりの測定時間を 30 秒とした。各状況を示すボタンをタップすることで加速度センサとジャイロスコープの測定が並列に実行される。収集を行った結果、加速度とジャイロのそれぞれで 3030 秒間のデータを取得することができた。

### 4.2. 状況予測結果

今回、状況予測のために用いるセンサデータには、重力成分を除いた加速度値を用いた。また、使用する機械学習アルゴリズムには Random Forest を採用した。収集した 3030 個のデータを教師データとテストデータにランダムで 3:1 に分割し、教師データを用いて生成される学習モデルを使用してテストデータの予測を行い、その結果から学習モデルの評価を行った。その結果、適合率と再現率の調和平均である F 値が約 96% となり、高い精度で正しい状況に分類することができた。ただし、今回使用したデータはある程度限られた状況で測定されたものであり、実際にユーザの状況を判別するためには、さらに多くの状況のデータを収集する必要がある。

## 5. おわりに

本稿では、コンテキスト適応型通知を実現するためのユーザ状況判別手法について提案した。今後は、加速度センサ以外のセンサも用いることで、より細かいユーザのコンテキストに対応し、ユーザごとに最適なコンテキストで通知を行う仕組みを開発したい。

## 参考文献

- [1] 三好康夫, 富田大樹: “Web 学習習慣化・促進支援のための学習コミュニティサイトとブラウザ拡張機能の開発”, 教育システム情報学会研究報告, Vol.30, No7, pp.127-134, 2016.
- [2] “スマホユーザー5 割弱、大量プッシュ通知は「アプリ削除」 - ITmedia Mobile”, <http://www.itmedia.co.jp/mobile/articles/1603/17/news086.html>, 2017 年 1 月 30 日現在.
- [3] “React Native | A framework for building native apps using React”, <https://facebook.github.io/react-native/>, 2017 年 1 月 30 日現在.
- [4] “The Web framework for perfectionists with deadlines | Django”, <https://www.djangoproject.com>, 2017 年 1 月 30 日現在.