

# スマートフォンを用いた行動記録を支援する テキスト入力システム

鈴木 孝宏<sup>1</sup> 角 康之<sup>2</sup> 美馬 義亮<sup>2</sup>

**概要:** スマートフォンを使用して Twitter につぶやきを投稿するために掛かるステップを簡略化するために, 予め登録した文章やフレーズを手軽な操作で投稿出来るアプリケーションを開発した. アプリケーションでは, 現在のファイルシステムのような階層構造, 階層に含まれるコンテンツを示すアイコンを用いることで, 目的となるテキストにたどり着きやすいよう工夫した. アイコンなしのリスト, アイコンのみのリストを用いたアプリケーションを開発し, ユーザテストを実施した. これらの開発とユーザテストから, アプリケーションを改善するフィードバックを得た.

## A Text Input System for Activity Recording using Smartphone

TAKAHIRO SUZUKI<sup>1</sup> YASUYUKI SUMI<sup>2</sup> YOSHIAKI MIMA<sup>2</sup>

### 1. はじめに

スマートフォンなど携帯情報端末の普及により, Twitter などの SNS を通して行動を記録, 共有しようとするユーザも見られるようになった. そのようなユーザは, 記録時に行っているタスクを中断せざるを得ないため, なるべく短時間でテキストを入力したいというニーズを持っている. 現状のテキスト入力システムでは, ソフトウェアキーボードを用いてすべての文字を入力する必要があり, POBox[1] のような予測入力を利用して入力する場合も, 入力したい単語やフレーズの最初の数文字を入力し, 変換候補の中からユーザが選択して投稿文を作成する必要がある. 一方, スマートフォンでは手書き入力や音声入力を利用することもできるが, 投稿する文章をすべて入力することには変わらない. また, 手書き入力は入力文字の画数に比例して入力に時間が掛かり, 音声入力は電車内など雑音の多い場所や私語を慎む状況下での入力には不向きである.

本稿では, 行動を記録, 共有する 140 字以内の短い文章を Twitter に投稿するステップを簡略化するために開発したシステムと, ユーザテストから得られた知見について述べる. 本システムは, 投稿するテキストを予めテンプレ

ートとして登録しておき, 投稿時には登録したテキストを呼び出す簡単なタップ操作のみで投稿が完了するというアプローチによりステップを簡略化する.

以下の節では, 本稿で対象としている行動を記録するテキストについて述べ, 開発したシステムの詳細と使用方法などを紹介する.

### 2. 行動を記録する文章

Twitter に投稿されるツイートのうち行動記録となるつぶやきは, 図 1 のように何かタスクを終えたことを共有する内容や, 図 2 のようにユーザが現在遂行中のタスクを共有するものが存在する. このようなテキストは,

- タスク
- タスクの遂行状態 (完了・未完了・遂行予定)
- 場所

といった複数の情報を持っており, これらの情報をテキストに含めることで行動記録として共有可能になる. しかしながら, 短い時間の中で必要な情報を満たしたテキストを作成することは, 常に可能とは限らない. テキストが完成しきらず後回しになり, 再び書き始めたときには機会を逃していることも少なくはない. このことから, 予め行動を記録するテキストをテンプレートとして用意しておき, 必要になったときに少ないステップでテキストを呼び出すと

<sup>1</sup> 公立はこだて未来大学大学院システム情報科学研究科

<sup>2</sup> 公立はこだて未来大学システム情報科学部



図 1 タスク完了のツイートの例



図 2 タスク完了・遂行予定のツイートの例

いうアプローチが、作業中のタスクを中断しすぎず、短時間でテキストを投稿したいユーザにとって有効であると考えられる。

### 3. 手数をかけずにテンプレートを選択する方法の検討

本稿で開発するシステムは、完成したテキストを短時間で Twitter に投稿し、行動を記録、共有する手法として、予め投稿するテキストをテンプレートとして登録しておくという方法を用いる。ここでは、ユーザが登録したテキストを短時間に探索する方法を検討した過程を示し、本システムの持つ要件について考える。まず、ユーザが使用するテンプレートを見つけ出すステップを簡略化する手法として、次の 2 つの方法が考えられる。画面のデザインからのアプローチと、荒川らの分析 [2] にもあるように、スマートフォンで取得出来るコンテキストを利用した推薦である。具体的には、

- (1) 一覧性を持ったレイアウトでテンプレートを画面に表示する
- (2) 使用履歴や時刻・位置情報といったコンテキストを利用してテンプレートをユーザに推薦する

の 2 つである。本稿では、まず (1) の方法でシステムを実装し、検討を行う。なお、システムは Android OS 用に実装し、Twitter へのテキスト投稿機能は Twitter API の Java ラッパである Twitter4J [3] を利用する。

#### 3.1 リストによるテンプレート管理

一覧性を持ったレイアウトの中で最もシンプルなもの、すべてのテンプレートをリスト状に管理することである。ここでは、図 3 のデザインで実装し、登録してあるテキストをリストで表示する仕様にしてある。長い文章の場合は、複数行で表示することで文末まで見通すことが可能である。画面に表示しきれなかったテンプレートは、スク



図 3 リストによるテンプレート管理手法の検討

ロール操作により表示される。Twitter への投稿方法は、リストの中から使用したいテンプレートを選択する、タップ操作の 1 ステップで完了する。また、ユーザが新たにテンプレートを登録したい場合は、画面右上にあるテンプレート追加ボタンをタップすることで新たなテキストを登録することができる。

一方、この手法ではすべてのテンプレートがひと続きのリスト構造で管理されており、テンプレートの数が多くなるとつれて目的のテンプレートを見つけ出すことが困難になる。加えて、スクロール量も大きくなるため、画面をタップする回数は必然的に増えてしまうといった欠点がある。この点に対して、野田らの例文分類辞書 [4] の考え方を参考に、テンプレートを利用する状況や意味ごとにカテゴリ分類ができるよう、階層構造を導入する必要がある。

#### 3.2 2 階層によるテンプレート管理

テンプレート数が多くなった場合に対応出来るよう、カテゴリ層とテンプレート層の 2 階層によるテンプレート管理を検討する。この場合、前述の仕様と同様に一覧性を維持しつつ、カテゴリ層があることで検索性がさらに向上する。実装したシステムのカテゴリ層が表示されている様子を図 4 に示す。カテゴリ層は、テンプレートの使用される状況やその意味に応じてテンプレートを分類する機能を提供する。カテゴリ名をタップすることで、その下層であるテンプレート層に登録されているテンプレートを展開表示することができる仕様である。カテゴリ名をタップした状態が、図 5 である。展開されたテンプレート層には、仕様履歴に応じてテンプレートの順番がソートされる機能を追加してあり、操作をより快適にしている。これは、3 章の冒頭で述べた簡略化手法の 2 を一部実装したものである。

一方で、展開されたテンプレート層が長くなれば、3.1 同様スクロールしなければ見つけられないテンプレートが発生する。このことから、登録出来るカテゴリ数・テンプレ

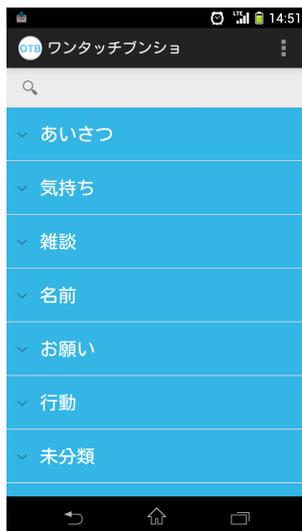


図 4 2 階層によるテンプレート管理 (カテゴリ層)



図 6 アイコン画像を適用した本システム



図 5 2 階層によるテンプレート管理 (展開されたテンプレート層)

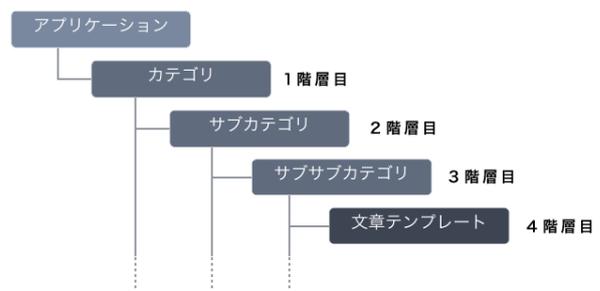


図 7 アプリケーションの階層構造の概念

レート数に制限を設ける必要がある。加えて、2階層によるテンプレート管理は、カテゴリ選択時に一通りテキストを読む必要があるため、カテゴリ層をアイコン化することが選択時の負担軽減に繋がると考えられる。

以上の検討から、本稿で開発するシステムの要件として、

- 階層構造によりテンプレートを分類することができる
  - テンプレート数・カテゴリ数に上限を設ける
  - 分類したカテゴリには画像を設定出来るようにする
- といった要件を設定する。

#### 4. 提案システム

本稿では、行動記録するテキストを予めテンプレートとして登録しておき、テンプレートを選択するタップ操作のみで Twitter に共有することのできるシステムを開発する。システムのスクリーンショットを図 6 に示す。前節で設定した要件にも触れながら、システムの全体像について以下

で述べていく。

このシステムでは、カテゴリ層を 3 階層、テンプレート層を 1 階層の 4 つの階層構造でテンプレートを分類することができる仕様になっている。図 6 ではアプリケーションを起動してすぐに表示される 1 階層目のカテゴリ層の画面である。1 画面で最大 9 のカテゴリ・テンプレートを管理することができ、9 つのアイコンは 1~3 階層目ではカテゴリ、4 階層目ではテンプレートを示している。カテゴリやテンプレートの準備は、ユーザ各自が行う仕様になっている。アイコンには、端末に搭載されているカメラ撮影した画像を設定出来るようになっている。

つぶやきを投稿するステップは、タップのみで完了する。3.2 で検討したシステムと異なり、画像が設定されていることにより、カテゴリやテンプレートのテキストを読まずにカテゴリの下層へと進むことができるようになっているため、選択時の負担が軽減されている。

3.1 から引き続き、使用履歴によるソート機能は継続している。この機能は、ユーザの好みに応じて無効にもできるよう、設定画面でチェックを外すことによりソートしない設定にもできる (図 8)。また、カテゴリ・テンプレートの編集機能も実装し、登録したテキストや画像を後から変更できるようになっている。変更時に誤って Twitter へつ



図 8 設定画面のスクリーンショット

ぶやくことのないよう、Twitter 投稿機能の無効化もできるように設定画面を実装してある。

このように、3 章で検討してきたシステムの要素を残しつつ、検討から見いだされた要件も満たしたことで、テキスト投稿の効率化を実現している。

## 5. システムのユーザテスト

本稿で開発したアプリケーションを、Twitter を比較的頻繁に利用するユーザと、アカウントはあるものの頻繁には利用しないユーザの 2 グループでユーザテストを行う。ユーザの端末にアプリケーションをインストールしてもらい、数日間の試用の後、アンケートに回答してもらうことでアプリケーションに対するフィードバックを得た。

いずれのグループからも、使えそうというフィードバックが得られた一方、アプリケーションの未実装部分に対する改善のアイデアを多く得ることができた。アプリケーションそのものに対するフィードバックは、初回起動時に簡単なインストラクションがなければ難しい、ヘルプ機能が欲しいというものがあった。今回のテストでは、インストール時に操作方法を口頭で解説し、使用方法をまとめた説明書も配布したため、カテゴリ・テンプレートの追加、投稿までの操作はできていた。ユーザらはスマートフォンの使用には慣れているものの、アプリケーションの使用に関しては、システムからの積極的なアシストがなければ使用が難しいということがわかる。このようなユーザの操作や設定をサポートする機能の実装は、今後追加すべき課題である。

Twitter を頻繁に使用するグループからは、アイコン画像の設定に関して、カメラから直接画像を設定したいという意見があった。現状では、端末に保存された画像しか用いることができないことから、このような意見が出たと考えられる。画像の設定時にカメラから直接設定できること

は、ユーザビリティの向上が期待できる。

本稿で実施したユーザテストは、ゼミの学生など数名による試用しか行っていないため、本格的な検証・試用は課題として残る。一方で、使えそうというポジティブなフィードバックが得られたことから、今後は、ヘルプ機能など未実装となっている部分を実装し、より多くのユーザに使用してもらうことでアプリケーションの改善を進めていく。

## 6. まとめと今後の発展

本稿では、スマートフォンを使用して Twitter につぶやきを投稿するために掛かるステップを簡略化することを目的に、行動を記録、共有するテキストを少ないステップで投稿可能にするシステムについて述べた。投稿するテキストを予めテンプレートとして登録しておくというアプローチを用い、ユーザがテンプレートを探す 2 つの手法を検討した。テンプレートをリスト構造で管理する方法、テンプレートを 2 階層で管理する方法の 2 手法を検討し、テンプレートを階層構造で管理すること、アイコンとしての画像を設定すること、登録出来るカテゴリ・テンプレート数に上限を設けるといった要件を設定し、本稿で提案するアプリケーションを開発した。操作性の向上やテンプレート選択時の負担を軽減できる仕様を目指した。

ユーザテストからは、アプリケーションを改善する有意義なフィードバックが得られ、ユーザをサポートするヘルプ機能の実装など今後の課題を見いだすことができた。テンプレートの扱いにも課題はまだあり、たとえば綾塚らの GhostTweet[5] のようにテンプレートを柔軟に変化できるようにするなど、さらなる発展があると考えられる。

**謝辞** 本研究で開発しているシステムは、IPA2013 年度未踏 IT 人材発掘・育成事業の支援を受けて行っている。開発の指導をして下さっている大阪大学の石黒浩教授に感謝の意を表す。

## 参考文献

- [1] T. Masui, "An efficient text input method for pen-based computers," in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, 1998, pp. 328-335.
- [2] 荒川豊, 田頭茂明, and 福田晃, "Twitter を用いたテキストと入力文字列の相関関係分析," 情報処理学会論文誌, vol. 52, no. 7, pp. 2268-2276, Jul. 2011.
- [3] Twitter4J - A Java library for the Twitter API, 入手先 (<http://twitter4j.org/ja/>) (2014.04.30).
- [4] 野田隆裕, 白井治彦, 黒岩丈介, 小高知宏, and 小倉久和, "携帯情報端末における例文に基づく日本語入力方式," 福井大学大学院工学研究科研究報告, vol. 56, pp. 69-76, Mar. 2008.
- [5] 綾塚祐二 and 那和一成, "GhostTweet: 短い文の入力のための新たなパラダイム," 情報処理学会研究報告. HCI, ヒューマンコンピュータインタラクション研究会報告, vol. 2012, no. 6, pp. 1-5, Jul. 2012.