

コンテキスト情報を利用したテキスト入力システム

鈴木 孝宏¹ 美馬 義亮² 角 康之²

概要: Twitter のようなリアルタイム性のあるテキストを効率的に入力可能にするために、状況や文脈(コンテキスト情報)を利用して文を作成するテキスト入力システムを提案する。さらに、そのシステムのユーザと対話しながら文を形成するための構造(アーキテクチャ), および文の生成手法・表現の調整方法を記述する。学生を対象に行ったユーザビリティテストとコミュニティ FM におけるシステムの試用から、システムを利用して入力した文と、過去に Twitter に投稿された文を比較し、さらに文の作成支援の側面から有用性を示す。

Text Input System Using Contextual Information

SUZUKI TAKAHIRO¹ MIMA YOSHIKI² SUMI YASUYUKI²

Abstract: In this paper, a text input system, which uses contextual information, is proposed. The research issue is reduction of operation steps on text input. The way of generating sentences in the system is described. In addition, methods for adjusting phrases through the sentences generation is explained. Finally, usability of this system revealed by comparing generated sentences and actual posted sentences.

1. はじめに

携帯情報端末の普及により、ショートメッセージや LINE のようにリアルタイム性の高いテキストによるコミュニケーションが活発化してきている。加えて、Twitter や Facebook といった SNS も普及してきており、パーソナルコンピュータでもリアルタイム性のあるテキストを扱う機会が増えた。こうした中、即時性の高いテキストを短い時間で入力したいというニーズを満たすため、効率的なテキスト入力の需要が高まっている。

図 1 は、キーボードによるテキスト入力の目的を、「入力にかかる時間」と「入力するテキスト長」を軸にして整理したものである。入力する文が長くなるほど短時間で入力することは、キーボードでは困難であるため、細字となっている領域が存在する。これらは、本来、ニーズが存在しているが、実際にそのニーズを満たす技術が存在しないため、対応がなされていないものである。このことから、我々は、リアルタイム性の高い文を短時間で入力できるよ

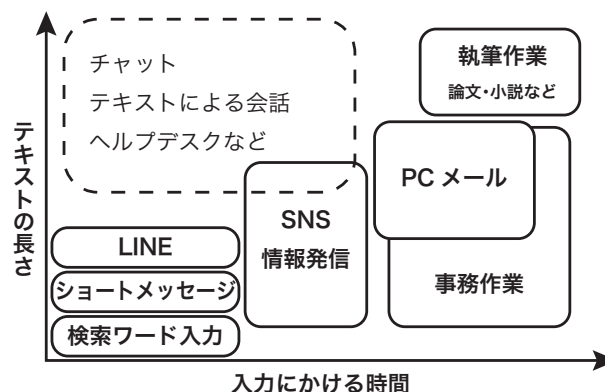


図 1 入力目的と文の長さ・入力時間の関係

うに効率的なテキスト入力システムの実現を目指す。

まずは、この空間に最も近く、かつリアルタイム性が求められる SNS への情報発信をターゲットとする。本稿では、Twitter へ投稿するテキストを効率的に入力可能にする新しいテキスト入力システムのコンセプトについて述べ、さらに具体的な実装例としてラジオ番組でを使用することを想定し実装したシステムを紹介する。実装したシステムが、効率的な入力を実現できているかを確認した検証実

¹ 公立はこだて未来大学大学院システム情報科学研究科

² 公立はこだて未来大学システム情報科学部

験の結果を報告する。

2. 関連研究

これまで日本語のテキスト入力を効率化する研究のひとつに、予測入力・予測変換がある。POBox[1]に代表される携帯電話の予測変換は、ユーザが入力した数文字のひらがなに基づいてユーザが入力しようとしている単語やフレーズを予測・提示する技術である。こうした入力予測・予測変換は、テキストを入力する際のタップ数を減らす手助けになっており、携帯端末には欠かせない機能である。さらに、近年ではパーソナルコンピュータ向けのIMEでも変換の推測機能が搭載されるようになってきている[2]。このように、端末の種類に関係なく、予測や推測による変換が使用されている。しかしながら、適切な予測ができず結果的に全ての文字を入力しなければ意図したテキストが入力できない場合もある。これは、システムがユーザの意図にまでアクセスできないことから発生する。

綾塚らは、GhostTweet というシステムを提案した[3]。このシステムは、スマートフォンで取得できる位置情報をもとに、場所に依存するツイートを投稿できるようにしたものである。入力方法は、完成文変更方式という従来の入力とは異なる手法を提案している。完成文変更方式は、入力システム側から完成した状態の文をユーザに提示し、ユーザがそれに対して修正を行い、入力したい文へと完成度を高めていく手法である。本研究は、完成文変更方式の考え方を参考にしている。しかし、本研究では利用するコンテキスト情報が主に時刻であり、文の生成方法はあらかじめ用意した文モデルを利用している点で異なっている。

3. コンテキスト情報を利用したテキスト入力

Twitter に投稿されるテキストの中には、「未来館なう」というような場所や時間などの情報が表れているものが存在する。このようなリアルタイム性の高いテキストには、文の作り手である投稿者の周囲の状況が表れている。ところが一般的なIMEは、ユーザの状況を考慮した変換にはなっていない。そのため、状況とは関係の無い単語が候補として示されることもあり効率的ではない。つまり、効率的なテキスト入力を実現するには、ユーザの置かれている状況(コンテキスト)を考慮する必要がある。

では、ユーザの置かれている状況には、どのような種類のものがあるだろうか。角は、人の周辺の状況として、以下の5つのW(Who, What, Where, When, Why)とそれに対応する項目をあげている[4]。これら5つのWは、状況を定義するパラメータとして見る事ができる。我々は、このようなパラメータをシステムが認識できるようにし、テキスト入力へ応用する(図2)。

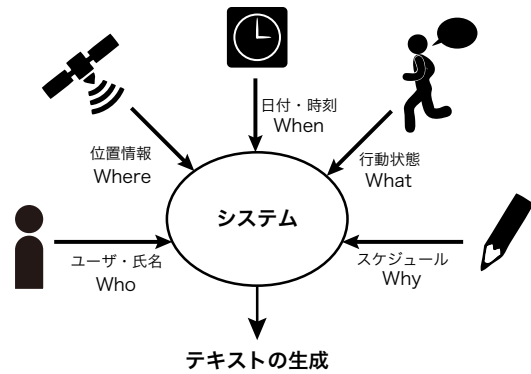


図2 提案するシステムのコンセプト

3.1 システムのアーキテクチャ

従来の入力方式は、文頭から文末へ向けて人間が想起した文の全ての文字を指定するという考え方に基づき入力していた。これは、人間からシステムへの一方通行のアクションである。本稿で提案するテキスト入力は、システムと人間がインタラクションしながら文を入力する。図3はシステムの文生成までのフローである。システムと人間がインタラクションして文を入力するとは、システムと人間が一緒になって文を作成、入力するということである。たとえば、システムが認識したコンテキスト情報をもとに文を作り人間に提案したり、それを人間が調整したり、システムが別の文の候補を提示したりする。システムと人間が影響し合いながらテキストを入力することで、効率的な入力、文の作成支援が可能になる。

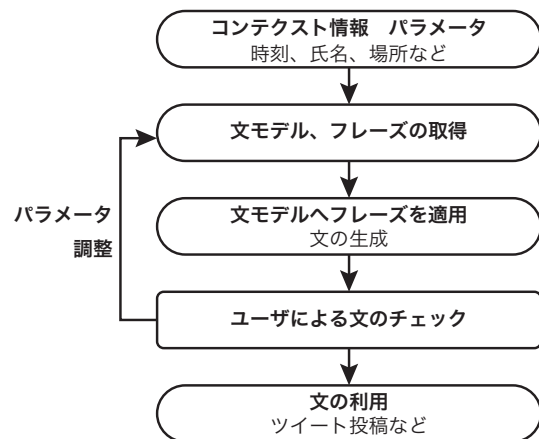


図3 文生成の流れ

3.2 文の生成方法

文の生成は、もとなる文のモデルを用意することで行う。文モデルは、文の意味構造を記述した抽象化された構造を持つ文であり、変数を含んでいる。たとえば、主語の部分を変数で記述し、状況に応じて具体的な人物名や指示語で置き換えることを行う。こうすることで、人間が読むことのできる文が生成できる。

生成する文のバリエーションは、文モデルを複数用意したり、文モデルが持つ変数の順番を入れ替えるなどすることによって、多くの組み合わせを作ることができる。また、ツイート時のコンテキストに近い特徴量を持つ過去のツイートを分析し、いくつかの基本文型を組み合わせることによって文のバリエーションを増やすことが可能である。

3.3 コンテキスト情報

具体的なコンテキストはパネルからユーザが入力するが、あわせて、利用している機器に登録された情報やセンサからの入力を利用する。さらに、コンテキストを広く認識するために、インターネット上で公開されている気象情報の API や、ニュースの RSS フィードなどを用いるのも良い。コンテキスト情報へのタッチポイントを増やし、よりコンテキストアウェアなテキスト入力が可能になる。

3.4 表現の調整方法

システムがコンテキスト情報をもとに文を生成しても、必ずしも人間の意図に近い文が作られるとは限らない。すなわち、文の表現を人間の意図に近づける調整をするためには、システムがユーザからのパラメータを受け取る仕組みが必要である。たとえば、人間が文を作っている際、入力した文が長すぎると感じたら、文の意味を変えないように文を短くする必要がある。このとき、人間は、文中に含まれるフレーズや語を別の表現のものに交換したり、文の要旨には大きく作用しない修飾語を削除したりする。この作業をシステムが行う場合、システムは自ら生成した文の意味を理解しなければならない。考えられる手法としては、生成した文を MeCab[5]などで形態素解析して品詞分解を行い、修飾語を削っていくという作業になるだろう。あるいは、CaboCha[6]などの係り受け解析を用いると、より正確に文の長さ調整が可能かもしれない。

このように、コンテキスト情報を用いることによって、Twitter でつぶやかれるツイートのようなリアルタイム性のあるテキストを効率的に入力可能にするシステムを構築することが可能になる。

4. 実装—ラジオ番組向けツイート入力支援

本稿では、提案するコンセプトの効果を確認するため、コミュニティ FM で使用することを想定してシステムを開発した。今回は、放送中の番組を紹介する文を生成するシステムとして実装してある。

コミュニティ FM のアカウントから投稿されるツイートは、投稿時に放送している番組内容の告知が多く、リアルタイム性の高い情報を発信している。また、番組は基本的に 1 週間単位でスケジュールされているため周期性がある。このことから、時刻情報をコンテキストに文生成が可能と考えた。

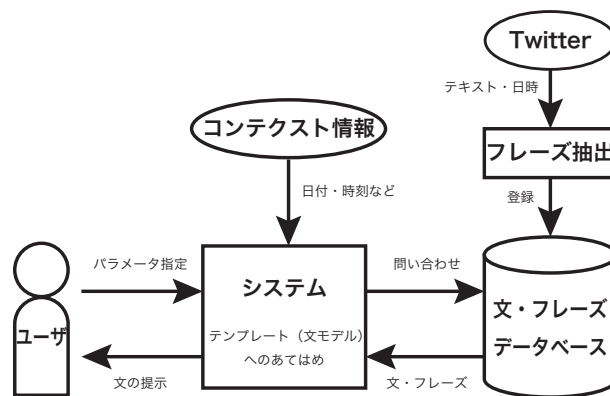


図 4 システムとユーザの対話

今回の実装では、このシステムを使用すると、ユーザは簡単なパラメータ調整のみで文を生成し、入力することができる。使用するコンテキストは、日付・曜日・時刻であり、図 2 における、Where はコミュニティ FM という状況で固定である。また、What, Who, Why に当たる部分は、画面に表示するインタフェースでパラメータを調整する。

4.1 文の生成

文の生成は、月日、曜日、時刻のコンテキスト情報と、ユーザが指定した各パラメータの値を使用して行う。図 4 は、文が生成されるまでのシステムとユーザの対話の関係である。システムは、コンテキスト情報とパラメータをデータベースに渡し、文のリストを要求する。データベースは受け取った情報をもとに文を生成し、システムへ返している。ユーザはシステムを通してデータベースが生成した文を確認し、文を変化させる場合はパラメータを調整すれば良い。

文の生成は、フレーズ変数を配置して文モデルを作成し、そのモデルに具体的なテキストを当てはめることで実現している。図 5 は、文モデルと変数の例である。文モ

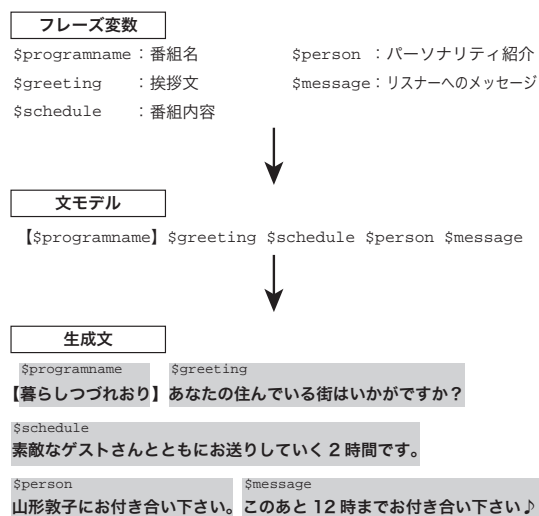


図 5 文モデルと変数の例

デルは5つのフレーズ変数を用いて記述している。これは、対象としたコミュニティFMのTwitterアカウントで投稿されたテキストを分析したところ、多くが、番組名(\$programname)・挨拶文(\$greeting)・パーソナリティ紹介(\$person)・番組内容の紹介(\$schedule)・リスナーへのメッセージ(\$message)という要素で成り立っていたためである。

4.2 コンテキスト情報によるフレーズ提示

本システムは、設定しているコンテキスト情報(月日、曜日、時刻)を変化させると、生成された文中のフレーズが変化する。図6は、上2つが月日・曜日は変えず時間のみを変化させたときの例である。下2つは日付・時間は変えず曜日のみを変化させた例である。日時等のコンテキストを変化させたときに番組名リストが変化する。この仕組みは、コミュニティFMのTwitterアカウントで過去につぶやかれたツイートの解析によって求めた、番組名の出現頻度に基づいている。あらかじめ、各番組名の出現回数を各曜日・1時間ごとに集計しデータベース化してある。同様に、1年間の各日付ごとでも出現回数を集計しデータベース化した。

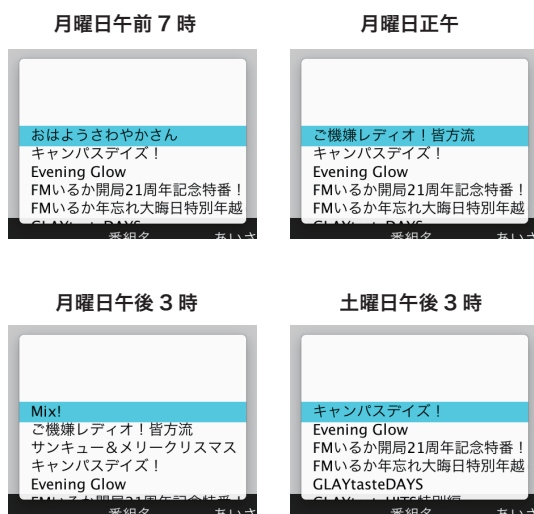


図6 コンテキストを変化させた例

さらに、番組名を定めると、それと高い頻度で共起するフレーズを画面に表示する。これは、番組名とフレーズを関連づけたデータベースに問い合わせることで表示している。このデータベースは、指定した番組名を含む過去のツイートをリストアップし、手作業でフレーズを抜き出し、データベースに登録したものである。

4.3 システムの操作

図7はシステムの実行画面である。画面中央には、システムが生成した文の候補を表示する。ユーザは、ツイートに使用する文をひとつ選ぶことができる。文の選択は、画

面下の最も右側にあるホイール状のインターフェースで行う。他の5つのホイールは、文中のフレーズを変化させるためのインターフェースである。生成する文の種類を変えるには、画面下の左側にある6つのボタンから選択する。画面上部には、システムを実行している現在の月日、曜日、時刻(時)が表示されている。各項目がコンボボックスの形式で選択、設定できるようになっている。これは、現在とは異なる特定の日時における文生成をシミュレートしたい場合に使用する。

5. 評価実験

コミュニティFM向けに実装したシステムのユーザビリティテストを行った。この実験では、提案システムで入力してもらった文の印象評価を行う。

5.1 手順

テキスト入力の実行

提案システムとキーボードで、テキストを入力する実験を行う。被験者は21歳から24歳の男女の12人だった。実験は、コミュニティFMで放送中にシステムを使用することを想定して行った。CM中や音楽を流している間の短い時間にツイート文を入力する状況をシミュレートするため、1回の入力タスクに15秒・30秒・45秒の制限時間を設け、4人ずつ割り当てた。

まず被験者には、前提となる状況を説明した。

- 自身はラジオパーソナリティである(名前は指定する)
- 制限時間内に番組を告知するツイートを入力する
- 担当している番組は、朝の情報番組である

割り当てた4人のうち、2人には提案システムで、残る2人にはキーボードで入力してもらった。提案システムで入力するグループには、数分間の操作練習を行ってもらった。キーボードで入力するグループには、紙に印刷したツイート例(実際にコミュニティFMのアカウントで投稿されたもの)を見てもらい、ツイートのイメージを考えてもらった。

それを終えたところで、入力の実行を行った。1人あたり3回の試行を行った。制限時間内でテキストを入力してもらった。キーボードによる入力では、制限時間内で入力しきれない場合がある。もし文が完成せず時間切れになってしまった場合は、未完成の文とは別に、あとから完成するまで入力してもらった。この試行で、制限時間15秒・30秒・45秒、それぞれ18パターンの文が得られた。

文の印象評価

次に、入力してもらったテキストの印象評価を行った。被験者は20歳から24歳の男女、12人だった。

まず、先の試行で作成してもらった18パターンの文と、コミュニティFMのTwitterアカウントで過去にツイートされたものから抽出した6つの文を、ランダムに並べた評

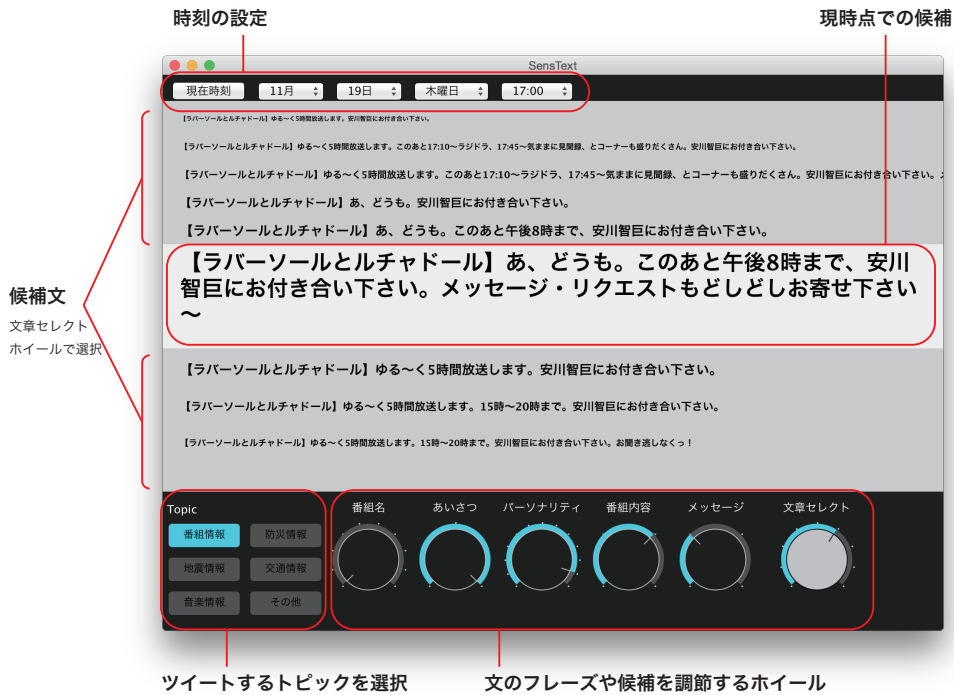


図 7 システムのスクリーンショット

備用シートを作成した。このシートには、24 パターンの各文と、それぞれの文を評価を記入する枠だけが用意し、どのシステムで入力したものであるかは分からないようにした。評価は、0~5 の 6 段階評価とした。このシートは、15 秒・30 秒・45 秒の文を分けて作成した。被験者を 3 つのグループに分け、それぞれに 15 秒・30 秒・45 秒の文を評価シートを渡し、評価を記入してもらった。

5.2 結果

テキスト入力の試行

図 8 は提案システムで入力されたテキストの一部、図 9 はキーボードで入力されたテキストの一部である。提案システムで入力したテキストは、全て文として成立したものになっていたが、中には 1 つの文に同じ言い回しが 2 回現れてしまったものもあった。一方、キーボードで入力したテキストは、あらかじめサンプルとして提示した過去のツイートを活用しながら入力できていた(天気に関する記述や、 を用いた表現)。しかし、制限時間では入力しきれなかったものがあった。

【おはようさわやかさん】きれいな朝焼けが見えます、おはようございます。安川智巨にお付き合い下さい。

【暮らしつづれおり】おはようございます。素敵なお客様との対話の時間をゆっくりお楽しみ下さいね！山形敦子にお付き合い下さい。このあと 12 時までお付き合い下さい

図 8 提案システムで入力した文(一部)

【おはようさわやかさん】おはようございます！今日はとても冷え込みましたね。今朝の函館はあおぞらがひろがりました！この後 10 時までおとどk

【暮らしつづれおり】皆さんからのメッセージもまだまだ募集しています！山形敦子と皆さんで楽しい時間を作りましょう

図 9 キーボードで入力した文(一部)

文の印象評価

	15 秒	30 秒	45 秒
過去のツイート	3.75	3.88	3.29
提案システム	3.67	3.56	2.33
キーボード(制限時間内)	2.96	3.25	2.42
キーボード(完成)	3.88	4.04	2.88
	n=4	n=4	n=4

表 1 印象評価の結果

表 1 は、印象評価の結果である。表の「キーボード(制限時間内)」は、キーボードで入力した各試行の、制限時間までに入力したテキストを評価対象とした。すなわち、制限時間いっぱい時点で文が完成しているもの、完成していないものが混在している。「キーボード(完成)」は、キーボードで入力した各試行の、文として完成した状態のものを評価対象とした。

5.3 考察

印象評価

過去のツイートと提案システムの平均スコアを比較する

と、過去のツイートの方が高くなった。これは、過去のツイートで比較対象としたものが、番組開始時にツイートされたものであり、ツイート文の作成にかけられる時間が十分にあったためであると考えられる。他のデータで見ると、キーボード（制限時間内）とキーボード（完成）を比較したとき、時間をかけた方である「完成」の方がスコアが高く、評価が過去のツイートのそれに近くなっている。このことから、作成に時間をかけた文は評価が高いことが分かる。

提案システムの平均スコアは、15秒、30秒においては、キーボード（制限時間内）よりも高い平均スコアだった。その一方、45秒においては、ほぼ近い値となったが逆転している。これは、45秒になったことによって、多様な文を発想する時間が合ったためと考えられる。また、提案システムが生成できる文のバリエーションにはまだ限界があったためと考えられる。

さらに、提案システムの評価が過去のツイートと比較してスコアが低くなった点について考える。提案システムで入力した各テキストの平均値をみると、スコアの低い文には、図8であげたように、フレーズが繰り返されてしまったものが目立っていた。通常の入力時には避ける「同じフレーズの繰り返し」が文に現れると、読み手の印象を下げるといえることになると分かった。

テキストの入力

入力時には、被験者の発話内容にも着目して行った。「入力できる文のパターンが少ない」や「挨拶文は必ず入れなければならないの?」というような発話があった。これには、フレーズデータベースがまだ充実されていなかったり、文モデルのバリエーションがまだ多くないためであると考えている。一方で、「イメージーションが膨らむ!」というポジティブな発話もあり、文の作成支援としての効果がある可能性を示している。

6. コミュニティ FM での試用

開発したシステムを、ラジオパーソナリティの方にも体験してもらい、フィードバックをもらった。システムを10分間使用してもらい、操作感や、放送中のブース内で使用した場合にどのようなことが起こるかの感想をもらった。

試用から得られたフィードバックは以下であった。

- 自分がこれから放送しようとしている番組の紹介を作ってもらって便利.....(1)
- ツイートの内容に困ったときは、これを使えばツイートが書ける...(2)
- 初心者ADでも、工夫を凝らしたツイートが書ける.....(3)
- マウスではない操作もできる方が良い...(4)

(2)、(3)のフィードバックにあるように、文の作成支援としてシステムが機能していることを確認することができた。

また、(1)にあるように、コンテキスト情報をもとに文を生成する機能は好評であった。しかしながら、(4)にあるように、現場での使用を考えると、マウスはクリック音があるためマイクが機能している間は使いづらいというフィードバックがあった。マウスやキーボードとは異なる、システム専用のコントローラがあると良いだろう。

7. まとめ

本稿では、Twitterのようなリアルタイム性のあるテキストをすばやく効率的に入力可能にすることを目的に、状況や文脈（コンテキスト情報）を利用して文を作成するテキスト入力システムを提案した。このシステムは、コンテキスト情報からその状況にふさわしい文を生成、ユーザに提示し、ユーザと対話しながら文を意図に近づけていくという手法で入力する。今回は、リアルタイム性の高いテキストの入力が求められるコミュニティ FM で使用する事を想定したアプリケーションを開発した。日付、曜日、時刻のコンテキストを用い、文の生成を行った。評価実験とコミュニティ FM での試用からは、文の作成支援としてシステムが機能していることを確認することができた。

今後は、マウスを使用しなくてもアプリケーションを操作できる消音コントローラの製作を行い、フレーズのバリエーションや文モデルのバリエーションの追加を行う。

謝辞 本稿で提案したシステムは、IPA2013年度未踏IT人材発掘・育成事業の支援を受けて開発したテキスト入力システムを進展させ、新しい機能を実装したものです。開発プロジェクトの指導をして下さった大阪大学の石黒浩教授に感謝申し上げます。

参考文献

- [1] Masui, T.: An efficient text input method for pen-based computers, *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '98, New York, NY, USA, ACM Press/Addison-Wesley Publishing Co., pp. 328-335 (1998).
- [2] JustSystems Corporation: ATOK.com (オンライン), 入手先 (<http://www.atok.com>) (2012.12.12).
- [3] 綾塚祐二, 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 (2012).
- [4] 角康之: 実世界インタラクションにおける状況認識の役割, システム/制御/情報: システム制御情報学会誌, Vol. 47, No. 4, pp. 179-184 (2003).
- [5] 工藤拓, 山本薫, 松本裕治: Conditional Random Fieldsを用いた日本語形態素解析(解析), 情報処理学会研究報告. 自然言語処理研究会報告, Vol. 2004, No. 47, pp. 89-96 (2004).
- [6] 工藤拓, 松本裕治: 相対的な係りやすさを考慮した日本語係り受け解析モデル(自然言語), 情報処理学会論文誌, Vol. 46, No. 4, pp. 1082-1092 (2005).