

## E-056

## 携帯電話における入力誤り自動訂正手法の日常的な文章に対する有効性について

Effectiveness of Automatic Correction of Erroneous Text Input on Mobile Phone for Daily Sentences.

菊地 直樹<sup>†</sup>      松原 雅文<sup>†</sup>      Goutam Chakraborty<sup>†</sup>      馬淵 浩司<sup>†</sup>  
 Naoki Kikuchi      Masafumi Matsuhara      Goutam Chakraborty      Hiroshi Mabuchi

## 1. はじめに

近年、携帯電話をはじめとした携帯端末の普及とメール機能、ブラウザ性能の向上、また、各種 SNS や twitter の普及によって、携帯端末での文字入力機会と必要性は増大している。携帯端末は小型であるため、常に持ち歩いている使用が可能であり、文字入力の状況も歩きながらやテレビを見ながらなど、他の動作と並行して行われることが想定される。そのような場合、ユーザは入力するために画面を終始注視しているわけではない。そのため誤入力が増え、訂正のために打鍵数が増加し、迅速な入力が困難になるという問題が発生する。

そこで、この問題を解決するため、携帯電話での文字入力における誤り自動訂正手法 [1] を提案している。本手法では、誤りを含む入力文字列に対して、かな漢字変換を行った際に、システムにより自動的に誤り訂正を含む変換候補が生成される。そして、出力された変換候補の中から、正しい変換結果をユーザが選択することにより処理が完了する。これにより、誤入力の訂正にかかる手間を軽減することができる。

本稿では、本手法の概要を示し、実際の文字入力データから得られた、日常的な文章に対して実験を行い、得られた結果から本手法の有効性を示す。

## 2. 対象とする誤入力

本手法で訂正対象とする誤入力は、同一キー打鍵数誤差  $\pm 1$  の誤りである。この入力誤りは、例えば「そうじ」と入力しようとして「せうじ」や「そえじ」というように、キーを打鍵する回数を 1 回多く、または少なく打鍵してしまった場合の誤りである。これは現在の一般的な携帯電話の入力方式が、1 つのキーに複数のかな文字を割り当てており、1 文字入力のために同じキーを複数回打鍵する必要があることによって発生する入力誤りである。なお、本手法では、入力文字列中の 1 文字を誤った場合の誤入力を対象としている。

## 3. 提案手法

## 3.1 概要

本手法における処理の流れを図 1 に示す。ユーザが入力した文字列へのかな漢字変換の際に、システムにより入力文字列に対しての訂正候補かな文字列が自動的に生成される。次に、生成された訂正候補に対し、入力傾向に基づいた数値を利用して重み付けが行われる。その後、訂正候補かな文字列に対して辞書照合を行い、候補を漢字に変換する。そして、変換が行われた候補を、先ほど与えられた重み付けを用いて、ユーザが意図していた可能性の高い順にソートし、変換結果として表示する。最

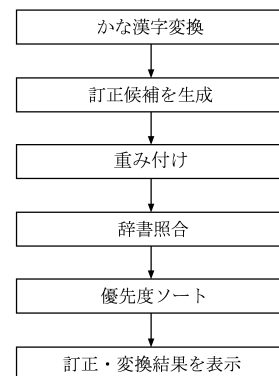


図 1: 提案手法の流れ

終的に、表示された変換候補一覧の中からユーザが候補を選択することで誤入力の訂正を完了する。

## 3.2 誤り訂正候補生成

入力された文字列から、まず、同一キー打鍵誤差  $\pm 1$  の誤りに基づき訂正候補の生成を行う。文字数が  $n$  文字であった場合、生成される候補数は基本的に  $2n + 1$  通りである。例えば「そうじ」という 3 文字の文字列が入力された場合、「しうじ」、「せうじ」、「すいじ」、「すえじ」、「そうざ」、「そうず」、「そうじ」という 7 個の候補文字列が生成される。

## 3.3 重み付け

生成された候補に対して、携帯電話での文字入力傾向に基づいて重み付けを行う。重み付けには誤入力傾向と正入力傾向を用いる。正入力傾向を  $c$ 、誤入力傾向を  $e$  とすると、重み  $w$  の値は  $w = \alpha c - \beta e$  の式で表される。ここで  $\alpha, \beta$  は定数である。

## 3.4 辞書照合

生成された候補に対して辞書照合を行い、候補を漢字へと変換する。照合の際には SKK 辞書を用いる [2]。この辞書は、あらかじめ言語的に現れやすい候補が上位にくるように語が収録されている。さらに、活用する語の送り仮名の開始位置の情報を持っている。そのため、生成された候補に対してそのまま辞書照合を行うことが可能である。今回は、SKK 辞書の中でも M サイズの辞書を使用する。

## 3.5 優先度ソート

辞書照合によって漢字へと変換された候補に対して、与えられた重み付けの値を用いて優先度を決定し、優先度の高い順にソートを行い、変換候補を表示する。

その後、ユーザによって候補から正解である単語が選択され、処理が完了となる。

<sup>†</sup> 岩手県立大学大学院 ソフトウェア情報学研究科

表 1: か行:誤入力の分類例

誤りの種類	1 → 2	2 → 1	2 → 3	3 → 2	3 → 4
重み	1	1	0	7	8
誤りの種類	4 → 3	4 → 5	5 → 4	5 → 6	-
重み	2	5	2	14	-

表 2: か行:正入力の分類例

打鍵数	1	2	3	4	5
重み	193	86	144	65	70

表 3: 評価実験結果

順位	誤入力		正入力		合計	
	件数	精度%	件数	精度%	件数	精度%
1 位	315	63.0	2595	86.5	2910	83.2
2 位以下	41	8.2	120	4.0	161	4.6
未変換	144	28.8	285	9.5	429	12.3

単語を見ると、読みが 2 文字の単語の場合に、正解の順位が低くなっていることが確認された。これは、読みが 2 文字の場合、生成された候補を辞書照合する際に、一致する候補が多く存在するためであると考えられる。この問題の解決のためには、現在重み付けを与える際に使用している  $\alpha$ ,  $\beta$  の数値を検討し、最適な値を使用することや、各利用者ごとの、単語の利用履歴の情報を重み付けの際に考慮する手法を導入することが有効であると考えられる。

また、未変換となった単語が全体で 429 件確認された。これは今回使用した辞書の中に、対象となる単語が登録されていなかったために未変換となっている。これにより、日常的な文章を対象とした場合、現在の辞書サイズでは不十分であるということが判明したため、辞書サイズの検討を行う。以上のように、合計で 83% の精度が得られ、本手法の有効性を示すことができた。

## 5. おわりに

本稿では、携帯電話での文字入力における誤り自動訂正手法を提案し、実験によって、本手法の日常的な文章に対しての有効性の評価を行った。本手法では、文字循環指定方式での入力傾向を利用して、誤り訂正を含む変換候補を生成し、ソートを行うことでユーザの意図していた順に出力する。これにより誤入力訂正にかかる手間を軽減することが可能である。また、重み付けの際に誤入力傾向と正入力傾向を利用することで、入力文字列が正しい場合、誤っている場合の両方に対して、区別することなく処理を行うことが可能となっている。

評価実験として twitter より得られた、日常的な文章に対し、本手法によるかな漢字変換処理を行った。その結果、全体で 83.2 % の精度が得られ、本手法の有効性が示された。

今後は、使用辞書のサイズを変更しての実験を行う。また、より多くの文字入力データを用いて実験を行い、詳細な入力傾向を得ることで、精度の向上を図る。

さらに、候補の出現順位による評価のみではなく、本手法を使用した際の打鍵数による評価も行っていく予定である。

## 参考文献

- [1] 菊地 直樹, 松原 雅文, Goutam Chakraborty, 馬淵 浩司, “携帯電話での文字入力における誤り自動訂正手法の性能評価” 情報科学技術フォーラム講演論文集 9(2), pp.319-320, August 2010.
- [2] 佐藤 雅彦, “Simple Kana to Kanji conversion program”, <http://openlab.jp/skk/index-j.html>.

## 4. 評価実験

### 4.1 実験方法

本実験では、twitter の投稿より得られた文章データに対し、かな漢字変換を行い、訂正変換精度を示す。実験に使用したデータは日本語約 50,000 文字からなる日常的な文章データで、その中の 3,500 単語に対して処理を行った。そのうち誤入力を含む単語は 500 件、正入力単語は 3,000 件である。

評価には、システムにより生成された変換候補の中で、正解の文字列が何位に出現しているかという情報を使用する。正解文字列が候補の先頭に出現したものを 1 位と分類し、正解文字列が先頭以外に出現したものをまとめて 2 位以下と分類する。また、候補の中に正解文字列が出現しなかったものを未変換として分類する。

### 4.2 重み付け

誤り訂正候補の重み付けには、実際の携帯電話での文字入力データから得られた入力傾向を用いる。誤入力傾向は、同一キー打鍵の際に何打鍵から何打鍵へ誤ったかという情報に基づき、誤りの種類を 10 通りに分類し、さらにその分類を「あ行」から「わ行」までの各行ごとに分けた件数である。

表 1 に、か行の誤り分類の例を示す。例えば、「こ」を「け」と誤って入力した場合、「か行:5 → 4」という誤りに分類される。

正入力傾向は、入力データ中で正しく入力された文字を分類した件数である。表 2 に、か行の正入力の分類例を示す。上記の例の「こ」に対して判定を行う場合、70 という数値が与えられる。

上記の誤入力傾向と正入力傾向を基に重みの値を決定する。重み  $w$  の値は  $w = \alpha c - \beta e$  で示される。今回の実験では  $\alpha$ ,  $\beta$  の値を  $\alpha = 1.8$ ,  $\beta = 0.8$  と設定した。

### 4.3 実験結果および考察

実験結果を表 3 に示す。処理を行った結果、誤入力 314 件、正入力 2595 件の単語で、正解単語が訂正候補の 1 位に出力された。訂正結果をみると、「お願い」という送り仮名を含む単語を「おのがい」と誤入力した場合のもの、「渋谷」「盛岡」といった地名など、日常的に使用するであろう単語について正しく処理が行われている。また、訂正後の正解単語が候補の 2 位以下に出力された単語が 41 個確認され、正入力でも、順位が 2 位以下で出力された例が 120 件確認された。これら 2 位以下の候補の平均順位は約 3.7 位であった。候補の出現順位が低い