2022-3-2

OCR を用いた崩れた表記における自動修正手法の有効性について

秋山 大五郎 † 松原 雅文 ‡

岩手県立大学大学院ソフトウェア情報学研究科 † 岩手県立大学ソフトウェア情報学部 ‡

1. はじめに

近年，SNS は多くのユーザが積極的に情報を発信する場となっている．多種多様なユーザが発信するデータは膨大であり，それらのデータを利用した研究も盛んに行われている．しかし，SNS 上のショートメッセージには「すっっっご」(すごい),「おっはー」(おはよう) などの口語的表現や,「ネ申」(神),「ナニヒ U」(たにし) などのネット特有の崩れた表記の単語が含まれており，それらの表現は一般的な形態素解析器の辞書に登録されていないため，未知語として判断されてしまい，単語が持つ本来の意味を機械が認識できないといった問題がある．

そこで我々は,「ネ申」(神) や「ナニヒ U」(たにし) といった文字の形容をもとにした崩れた表記に対して，これを OCR を利用して自動修正する手法を提案している [1]．また，崩れた表記に変換する辞書を用いて作成したデータにより性能評価を行った [2]．しかし，全体的な認識精度が低く，一部の崩れた表記に対し正規の表記と認識できないといった課題があった．

そこで，本稿では，OCR を用いて崩れた表記を認識する処理の精度を向上させるため，文字列を画像に変換する際の画像生成方法と画像生成設定の検討を行い，提案手法の有効性を検証する．

2. 関連研究

崩れた表記の単語を正しい単語に修正する手法として，崩れた表記が少ない文書から修正候補を取得する手法 [3] がある．文章中の崩れた表記の単語において左右の単語とマッチする文章を崩れた表現の少ない文書から検索し，修正候補を取得する．取得した修正候補に対して修正候補の出現頻度，修正前後の文字列間における編集距離，修正前の形態素解析コスト値に基づいたスコアリングを行い，総合スコアが一定以上のものを修正ルールとする手法である．

この手法では，本来修正する必要がない未知語に対しても修正処理を行ってしまう問題点がある．これに対し，フィルター処理を行うことで修正が不必要な未知語を修正対象から除外する手法 [4] がある．未知語として判断された単語に対し，TF-IDF(Term Frequency-Inverse Document Frequency) 処理を行い，TF-IDF 値をもとに検出された未知語が一般的かの判断を行う．一般的な未知語を処理対象から除外することによって崩れた表記の単語のみを処理する手法である．

これらの手法は文脈情報から修正候補を取得しているため，文章全体が崩れた表記の場合，修正候補を正しく取得することが難しい．そのため，我々は崩れた表記が含まれる文章に対し OCR を用いることで，文字の形容から修正候補を取得可能とする手法を提案している．しかし，崩れた表記に変換する辞書を用いて作成したデータにより性能評価を行った結果，一部の崩れた表記の文字列を正しく認識できていないことが分かった．

そこで，本稿では，文字列を画像に変換する際の画像生成方法と画像生成設定の組み合わせについて検討を行い，これを用いて行った評価実験の結果から，本提案手法の有効性を示す．

3. 崩れた表記

本稿における崩れた表記の定義は，池田らの研究 [3] と同様に正規の表現が形態素解析の辞書に登録されているが，表現上の揺らぎなどによって未知語として扱われるもの，とする．崩れた表記は正規の表記から派生したものが多く，一部の崩れた表記は崩れた表記から派生している．また，崩れた表記は発生と消滅を繰り返すため，すべての崩れた表記をあらかじめ辞書に登録することは難しい．そのため，提案手法のように崩れた表記から対応する正規の表記の特徴を利用して修正を行う方が，辞書を作成して修正を行うよりも適していると考えられる．

今回修正対象とする崩れた表記とその他一部の崩れた表記，またそれぞれの具体例を表 1 に示す．崩れた表記は大きく分けて (1) 大文字を小文字で代替する表記，(2) 正規の表記を複数の文字で代替する表記，(3) 発音の変化による文字の挿入と削除の表記，(4) 似た発音の単語で代替している表記の 4 種類に分けられる．今回は，この 4 種類の崩れた表記のうち，崩れた表記の形容をもとに修正が可能だと考えられる「大文字を小文字で代替している表記」と「正規の表記を複数の文字で代替する表記」を対象とする．その他の「発音の変化による文字の挿入と削除の表記」と「似た発音の単語で代替している表記」に関しては，今後，関連研究の手法や文字列の情報に音素の情報を加えるなどの手法を用いて修正したいと考えている．

Effectiveness of Automatic Correction Method for Informal Words Using OCR

Daigoro Akiyama†, Masafumi Matsuhara‡

†Graduate School of Software and Information Science, Iwate Prefectural University, ‡Faculty of Software and Information Science, Iwate Prefectural University

表 1: 対象とする崩れた表記と具体例

	分類	具体例
対象とする崩れた表記	大文字を小文字で代替	ゎたしは (わたしは)，かわぃぃ (かわいい)，うるさぃ (うるさい)，おおきい (ぉぉきい)
	併せ字による代替	カヽわレヽレヽ (かわいい)，⊇ωにちレよ (こんにちは)，米青ネ申禾斗 (精神科)，イ為牛勿 (偽物)
その他の崩れた表記	発音の変化による代替	すぁーん (さん)，でっかぁ (でかい)，よろしくぅ (よろしく)，やっべぇ (やばい)
	似た発音の文字による代替	53 ばこ (ゴミばこ)，4 まうま (しまうま)，4649(よ ろ しく)，99 しゃ (救急車)

4. 提案手法

4.1. 提案手法の概要

提案手法における OCR を用いた修正候補取得部分の処理を図 1 に示す．提案手法では，崩れた表記を多く含む SNS などの文書を入力とし，文章を 2 文字区切りで分割する．次に分割した文字列が描かれている画像を生成する．そして生成した画像から OCR を用いて修正候補を取得する．最後に取得した修正候補をスコアリングし，最適な修正候補を出力する．

4.2. 文章の分割処理

崩れた表記が多く含まれた文章を 2 文字区切りで分割する．これは，今回対象とする崩れた表記の中で「ネ申」(神) や「弓長」(張) などの「正規の表記を複数の文字で代替する表記」の多くが 2 文字以下で構成されているためである．

形態素解析で未知語と判定された文字列に対し修正を行う場合，崩れた表記の組み合わせによって未知語と判定されないものや，組み合わせて 1 つの文字を表している崩れた表記の構成要素が別々の未知語として判定されるという問題点がある．また，文章全体に対し OCR を用いて文字列を取得すると，文章全体の認識精度が高い場合には，崩れた表記の文字列はそのまま崩れた表記の文字列として取得される．逆に，文章全体の認識精度が低い場合には，一部の崩れた表記の文字列を正規表記の文字列と認識できるが，そもそもの認識精度が低いため，もともと正規表記である文字列を正しく認識することができないという問題点がある．これらの問題点を解決するため，一定の文字数で文章を区切り分割することとした．

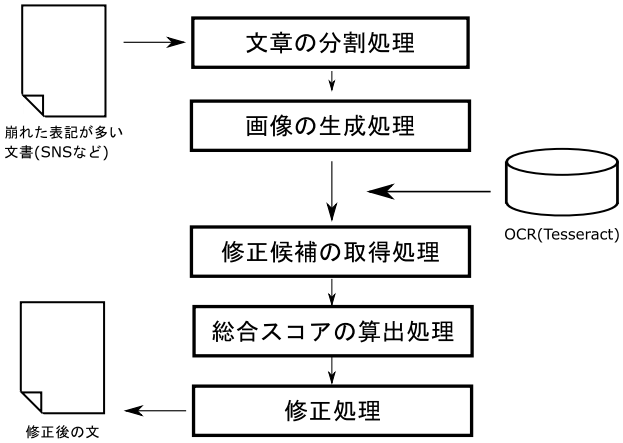

図 1: 提案手法の全体像

4.3. 画像の生成処理

4.3.1 概要

分割した文字列が描かれている画像を生成する．崩れた表記は 2 文字で正規表記 1 文字を表していることが多いため，崩れた表記の 2 文字が描かれている画像を生成する．本稿では，崩れた表記の文字列に対応する正規表記文字を OCR が正しく認識できるようにするため，2 種類の画像生成方法を提案する．

4.3.2 文字幅を調節可能な画像生成

文字幅を調節可能な画像生成の例を図 2 に示す．「正規の表記を複数の文字で代替する表記」に属する偏と旁に分けられる漢字の認識率を向上させるための画像生成方法である．

この生成方法においては，最初に左右の文字が別々に描かれた 2 枚の画像を生成する．次に各画像をそれぞれ異なる比率でリサイズする．最後に 2 枚の画像を結合し，全体的にリサイズを行うという流れで，正規表記 1 文字に対する画像を生成している．

図 2: 文字幅を調節可能な画像生成の例 (イ可：何)

4.3.3 文字の場所を調節可能な画像生成

文字の場所を調節可能な画像生成の例を図 3 に示す．文字幅を調節可能な画像生成と異なり文字を描く座標を自在に調整可能な画像生成方法である．

この生成方法においては，最初に画像サイズを決定し，画像を生成する．次に特定の座標に文字を描く．最後にリサイズを行うという流れで画像を生成している．

図 3: 文字の場所を調節可能な画像生成の例 (イ可：何)

4.4. 修正候補文字列の検索

生成された画像から OCR(Optical Character Recognition) を用いて修正候補文字列を取得する．

4.4.1 OCR

OCR は，テキストが含まれる画像をテキストに変換する技術であり，一般的に紙に記載されたデータを取得する際に使用される．本研究では，OCR における，画像からテキストに変換する技術を用いて，崩れた表記の文字列が持つ形容の特徴からもとの正規表記の文字列を取得している．今回は，公開されている OCR エンジンの Tesseract [1] を用いた．

4.5. 総合スコアの算出

検索結果における修正文字列の出現頻度，修正前後の文字列間における編集距離を用い，これらをもとに総合スコアを算出する．その後，修正候補文字列の総合スコアが最も高いものを修正文字列として出力する．

このようにして，OCR を利用して崩れた表記を自動で修正する手法の実現を目指している．

5. 実験

5.1. 実験条件

提案手法における OCR を用いた修正候補の取得精度の向上を目的として，画像生成設定の最適な組み合わせを探索する．

実験には我々が作成した崩れた表記辞書を使用した．崩れた表記辞書の一部を表 2 に示す．崩れた表記辞書には，2 ちゃんねる[2]という掲示板に投稿された正規表記のテキスト約 100 万個をギャル文字変換器を用いてギャル文字が含まれる文章に変換した際の各文字の組み合わせ 1,383 個が登録されている．登録されている組み合わせのうち正規表記 1 文字に対し，1 文字の崩れた表記の組み合わせが 90 個，2 文字の崩れた表記の組み合わせが 1,203 個，3 文字の崩れた表記の組み合わせが 90 個となっている．また，実験には崩れた表記の組み合わせは 2 文字の崩れた表記の組み合わせのみを使用した．

データに対し，最適な画像生成設定を探索するため Beam Search を用い，各組み合せのうち上位 3 つの組み合わせを保持するように設定した．文字幅を調節可能な画像生成の探索範囲を表 3 に，文字の場所を調節可能な画像生成の探索範囲を表 4 に示す．各組み合せの評価には，正解率を使用し，各画像生成方法で 10 回探索を繰り返した．2 回目以降の探索ではそれ以前の組み合わせで認識できなかった崩れた表記のみについて評価している．

[1] https://tesseract-ocr.github.io/
[2] https://www.2ch.sc/

表 2: 崩れた表記辞書の例

崩れた表記の文字数	具体例 (正規)	具体例 (崩れ)
1 文字	く	＜
	う	ぅ
	ア	了
2 文字	俺	イ奄
	初	ネ刀
	ほ	レま
3 文字	拠	才夂几
	側	イ貝リ
	蹴	足京尤

表 3: 文字幅を調節可能な画像生成の探索範囲

	min	max	step	start
リサイズ比率 (1 枚目)	0.2	0.9	0.1	0.5
リサイズ比率 (2 枚目)	0.2	0.9	0.1	0.5
リサイズ比率 (結合後)	0.5	1.5	0.1	1.0
フォントサイズ	5	50	5	20

5.2. 実験結果

文字幅を調節可能な画像生成における探索結果の認識結果を表 5 に，文字の場所を調節可能な画像生成における探索結果の認識結果を表 6 に示す．追加認識文字数は新しく認識可能になった文字数を表し，累計認識文字数は前回の探索までの累計認識文字数と追加認識文字数の和を表し，累計正解率は 2 文字の崩れた表記の総数 1,203 で累計認識文字数を除した値を表している．

また，各画像生成方法の OCR 認識結果の比較と認識例を表 7 に示す．方法 1 は文字幅を調節可能な画像生成を表し，方法 2 は文字の場所を調節可能な画像生成を表している．

5.3. 考察

実験結果から文字幅を調節可能な画像生成で 10 枚の画像を生成した場合の正解率は約 45%となり，文字の場所を調節可能な画像生成で 10 枚の画像を生成した場合の正解率は約 50%となった．また，2 種類の画像生成方法を組み合わせることによって正解率は約 59%まで向上した．これは，文字幅を調節可能な画像生成では偏と旁の幅に応じた画像生成が行えるため，偏と旁の幅に差がある崩れた表記 (抜：才友，減：シ咸など) の認識が可能となったことによるものであると考えられる．また，文字の場所を調節可能な画像生成では文字を中央以外に描くことができるため，間違った漢字を利用した崩れた表記 (難：英隹，殿：展殳など) の認識が可能となったことも要因だと考えられる．

なお，2 種類の画像生成方法では対応できず，目視でも判断が難しい一部の崩れた表記の存在も確認された．これに関しては，OCR のみでは対応が難しいため，OCR を用いて一度修正を行い，修正が難しい箇所は文

表 4: 文字の場所を調節可能な画像生成の探索範囲

	min	max	step	start
リサイズ比率 (縦)	0.8	2.0	0.2	1.0
リサイズ比率 (横)	0.8	2.0	0.2	1.0
画像サイズ比率 (縦)	1	4	1	2
画像サイズ比率 (横)	1	4	1	2
書き始め座標 (縦)	-15	30	5	5
書き始め座標 (横)	-15	30	5	5
フォントサイズ	10	40	5	20

表 5: 文字幅を調節可能な画像生成における各探索結果の認識結果

探索回数	追加認識文字数	累計認識文字数	累計正解率
1 回目	207	207	0.172
2 回目	98	305	0.252
3 回目	73	378	0.314
4 回目	43	421	0.349
5 回目	34	455	0.378
6 回目	26	481	0.400
7 回目	20	501	0.416
8 回目	14	515	0.428
9 回目	12	527	0.438
10 回目	15	542	0.451

脈情報などを利用した手法を用いて修正を行う必要がある．

6. おわりに

本稿では，OCR を用いた修正候補獲得の精度向上のため，崩れた表記の形容に沿った画像生成方法を 2 種類提案し，最適な画像生成設定の探索を行った．実験結果から，2 種類の画像生成方法を用いることで OCR の認識精度が向上し，より多くの正しい修正候補の獲得が可能になった．

今後は，得られた修正候補をもとに総合スコアの算出を行い，実際の崩れた表記に対して自動修正を行う予定である．また，2 つの画像生成方法は別々に探索を行ったため，2 つの探索結果の中で認識可能な崩れた表記の多くが重複する画像生成設定が存在する可能性がある．そのため，正解率を確保しつつ認識可能な崩れた表記の重複を多く持つ画像生成設定の組み合わせを省く必要がある．また，今回対象としていない「発音の変化による文字の挿入と削除の表記」と「似た発音の単語で代替している表記」に関しても文字列から得られる情報に加え対応した音素を入力に用いるなどの手法を用いて修正することを検討している．

表 6: 文字の場所を調節可能な画像生成における各探索結果の認識結果

探索回数	追加認識文字数	累計認識文字数	累計正解率
1 回目	226	226	0.188
2 回目	192	418	0.347
3 回目	33	451	0.375
4 回目	77	528	0.439
5 回目	13	541	0.450
6 回目	27	568	0.472
7 回目	16	584	0.485
8 回目	5	589	0.490
9 回目	7	596	0.495
10 回目	4	600	0.499

表 7: 各画像生成方法の OCR 認識結果

画像生成方法	認識数	正解率	認識例 正規	崩れ
方法 1	600	0.499	城 憶	土成 小意
方法 2	542	0.451	特 は	牛寺 レよ
方法 1∨ 方法 2	705	0.586	的 漁	白勺 シ魚
方法 1∧¬ 方法 2	105	-	抜 減	才友 シ咸
¬ 方法 1∧ 方法 2	163	-	難 殿	英隹 展殳

謝辞

本研究の一部は JSPS 科研費 21K12611 の助成を受けたものである．

参考文献

1) 秋山 大五郎, 松原 雅文: OCR を利用した崩れた表記の自動修正手法の提案, 情報処理学会第 84 回全国大会,5V-07, 愛媛大学城北キャンパス, ハイブリッド開催,March 2022.

2) 秋山 大五郎, 松原 雅文: OCR を利用した崩れた表記の自動修正手法の性能評価, 第 21 回情報科学技術フォーラム,6E-06, 慶應義塾大学矢上キャンパス, ハイブリッド開催,August 2022.

3) 池田 和史, 柳原 正, 松本 一則, 滝嶋 康弘：くだけた表現を高精度に解析するための正規化ルール自動生成手法, 情報処理学会論文誌データベース（TOD）,Vol.3,No.3,pp.68-77,2010.

4) 星野 恵以子, 寺田 篤史, 村上 久, 秋吉 政徳：ソーシャルメディアに現れるくだけた表現を含む口語的表現の自動修正方式, 第 79 回全国大会講演論文集,Vol.2017,No.1,pp.595-596,2017.