ブログにおける表記の揺れを修正するためのルール自動生成システムの提案

池田 和史† 柳原 正† 松本 一則† 滝嶋 康弘†
†KDDI 研究所

1 はじめに

ブログ上の文書には「じゃん」、「したよ～ん」のような口語的な表現や「ｶわいい」、「ぁたUぁ」のような特有の表記が多く含まれるため、一般の形態素解析器を用いても十分な解析精度を得ることができない。これらの表現は人手により辞書登録されることが一般的であるが、人的コストが大きい点や言語処理に関する高度なスキルを必要とする点が問題となる。

本稿では上記のようなブログ的表記を文字列変換のルール(修正ルール)を用いて文語的な表記へと自動的に修正する手法を提案する。例えば、文字列「ｶわいい」を「かわいい」に変換することで、ブログ的な表記を文語的な表記に修正できる。提案手法では人手により与えられた少量の汎用な修正ルールをもとに、多数のより具体的な修正ルールをコーパスから自動的に学習し、生成する。

自動学習の手法として、(1)修正ルールを適用する前後の文の形態素解析結果を比較し、修正ルールをその適用数や正解率によってスコアリングする手法、(2)修正ルールのスコアを利用して、ルールをより具体的にする特殊化手法、(3)複数のルールを結合し、新たなルールを生成する結合手法、(4)修正ルールを効率的に学習するための汎用化手法を提案する。

提案手法を実装し、従来手法である人手による辞書拡張手法との性能比較評価実験を行った。実験では形態素解析時の未知語の減少数や文節区切りの正確さ、人的コストの大きさなどを定量的に評価した。従来手法の課題であったルールの過剰適用による文節区切りの誤りを提案手法では30%以上軽減するなど、大幅な性能向上が確認された。

2 関連研究

チャットの口語的表現を対象とした形態素解析辞書拡張手法[1]では、チャットの文章を人手により分析し、ルールを作成することで、既存の辞書から派生した語を辞書登録する。例えば、「がっこう」は「がっこー」と表現されるなどの例から、直前の文字の母音が「o」の場合、「お、う、ー、～」は互いに置換可能である、などのルールを提示している。この手法により、複数の語を機械的に辞書登録することができ、人的コストの軽減が期待されるが、人手によるルール作成は作業者が参考にした文例に依存したり、主観に基づきやすく、機械的に登録された語が文節区切りに悪影響を及ぼさないためには高度なスキルが求められる。口語的表現や話し言葉を言語的な観点などから分析した形態素解析手法は他にもいくつか提案されており[2, 3]、口語的表現の形態素解析精度向上に貢献しているが、口語的表現の分析は専門的なスキルと多くの労力を要するため、人的コストの大きさが課題となる。

3 提案手法

提案手法における修正ルールの機械学習処理について説明する。ユーザは始めに少量の修正ルール（プリミティブルール）を登録しておく。提案手法では登録されている修正ルールをもとに、大規模コーパスを用いて機械学習を行い、ルールの特殊化、結合、汎用化を繰り返し行うことで、多数の新たなルールを生成する。生成したルールをスコアリングすることで、複数のルールから文に適したルールを選択できる。

表1: 正解ラベルの付与

修正前	修正後	ラベル
未知語	未知語	△
未知語	既知語	○
既知語	未知語	×
既知語	既知語	□
具体例：修正ルール「ょ」⇒「よ」		
そう思うょ	そう思うよ	○
困ったょ	困ったよ	○
ちょうだい	ちようだい	×
私の勝ちょね	私の勝ちよね	○
行こぅょ	行こぅよ	△
びょういん	びよういん	□

表2: 修正ルールスコアリング例（「ょ」⇒「よ」）

修正ルール	適用数	正解数	正解率
「ょ」⇒「よ」	4	3	0.75
「うょ」⇒「うよ」	1	1	1.0
「たょ」⇒「たよ」	1	1	1.0
「ちょ」⇒「ちよ」	2	1	0.5
「ちょう」⇒「ちよう」	1	0	0
「勝ちょ」⇒「勝ちよ」	1	1	1.0

3.1 正解ラベルの付与とルールのスコアリング

修正ルールの適用前後の文がそれぞれ未知語を含むかどうかによってラベルを付与する。表1は修正ルール「ょ」⇒「よ」の適用例であり、修正前後で未知語を含むか既知語（辞書に登録されている語）のみであるかによって、"○","×","△","□"の4種類の正解ラベルを付与している。

ルールのスコアリングは正解ラベルを用いて行う。表1において修正ルールが正解となる文と不正解となる文の数を数えることで、表2のように、修正ルール「ょ」⇒「よ」のスコアリングを行うことができる。ラベルが"△"や"□"、すなわち文を修正したにもかかわらず、未知語が未知語のままである場合と、既知語が既知語のままである場合のスコアリングは手法の実装方針に依存する。

3.2 ルールの特殊化

プリミティブルールなどの汎用なルールをもとに、より具体的なルールを生成することをルールの特殊化と呼び、生成されたルールを特殊化ルールと呼ぶ。特殊化は正解ラベルを用いて行う。表1のように、「ょ」⇒「よ」の修正に対する正解ラベルには「うょ」⇒「うよ」の修正に対する正解ラベルも含まれる。これをもとに、表2のように「うょ」⇒「うよ」の修正ルールをスコアリングすることができる。

ルールの特殊化は複数のルールから適用すべきルールを選択するときに役立つ。例えば、未知語「見ようょ」に

A Rule Generation Method for Correcting Typical Expressions in Blog Documents
†Kazushi Ikeda †Tadashi Yanagihara †Kazunori Matsumoto
†Yasuhiro Takishima
†KDDI R&D Laboratories

```
既存のルール：
    「ヵ」⇒「か」、「ゎ」⇒「は」
原文：
    正しいのヵゎ分からない　未知語 =「ヵゎ」
修正ルール「ヵ」⇒「か」を適用：
    正しいのヵゎ分からない ⇒ 正しいのかゎ分からない △　… 状態 (1)
修正後文：
    正しいのかゎ分からない　未知語 =「ゎ」
未知語「ゎ」に適用可能な修正ルールを順に適用：
    「ゎ」⇒「は」
    正しいのかゎ分からない　⇒　正しいのかは分からない ○　… 状態 (2)
未知語を含まない文：
    正しいのかは分からない
新ルール：
    「正しいのヵゎ分からない」⇒「正しいのかは分からない」　… 状態 (3)
```

図 1: ルール結合の例

修正ルールを適用するとき、「ょ」⇒「ー」などの正解率の低いルールではなく、より正解率の高い「うょ」⇒「うよ」を適用することで、より高い確率で文を正しく修正できる。特殊化ルールの生成目的上、元のルールよりも正解率が低くなるルールについては、不適切な特殊化ルールを生成していると考えられるため、利用しない。

3.3 ルールの結合

正解ラベル付与において"△"のラベルが付与されるとき、修正後の文に再度修正ルールを適用することで、未知語が解消される場合、適用した修正ルールを結合し、新たな修正ルールとすることができる。

ルール結合の例を図 1 に示す。既存のルールとして、「ヵ」⇒「か」、「ゎ」⇒「は」があるとき、「正しいのヵゎ分からない」という文の未知語「ヵゎ」を 1 回の修正ルールの適用で解消することはできず、"△"のラベルが付与される (図 1 の状態 (1))。修正後文中の未知語（例では「ゎ」）に対して、別の修正ルールを再度適用することで、未知語を解消でき (図 1 の状態 (2))、原文から未知語が解消された文への修正を新しいルールとする (図 1 の状態 (3))。

3.4 ルールの汎用化

提案手法ではルールを効率的に学習するため、結合ルールなどで得られた具体的なルールの汎用化を行う。修正ルールの前後で共通の文字列を先頭と末尾から削除することでルールを汎用化する。例として、「正しいのヵゎ分からない」⇒「正しいのかは分からない」という修正ルールの汎用化では、先頭から「正しいの」を、末尾から「分からない」をそれぞれ削除し、「ヵゎ」⇒「かは」というルールを得る。得られたルール「ヵゎ」⇒「かは」を特殊化することで、「なのヵゎ」⇒「なのかは」というルールを得るなど、汎用化により得られたルールは再度特殊化、結合され、新たなルールの生成に役立つ。

4 性能評価実験

4.1 実験概要

提案手法の性能と従来手法である人手による辞書拡張手法 [1] の性能とを比較する。形態素解析辞書構築に関する技術とノウハウを持つ第三者の作業者により文献 [1] で提示されている辞書拡張ルールを作成する（**従来手法 A**）。次に、ブログコーパス 100 万記事を学習用データとし、辞書拡張ルールを追加する（**従来手法 B**）。提案手法も同じ学習用データを用いて、機械学習を行う。

評価の対象データとして、学習用とは異なるブログコーパス 100 万記事を用意し、そこから 1000 文をランダムに選択した。拡張前の形態素解析辞書を**基本手法**とし、従来手法 A、従来手法 B、提案手法の 3 手法について、対象データの形態素解析を行ったときの各手法における (1) 未知語総数の減少、(2) 文節区切りが変化した文数（適用数）、(3) 文節区切りが向上した文数（向上数）、(4) 文節区切りが悪化した文数 (**悪化数**)、をそれぞれ評価した。文節区切りの向上、悪化は基本手法と比較して判定する。

表 3: 各手法の性能比較

手法	未知語総数	適用数	向上数	悪化数
基本手法	206	-	-	-
従来手法 A	188	20	9	8
従来手法 B	166	51	24	19
提案手法	173	29	27	0

4.2 実験結果

提案手法では 124 件のプリミティブルールを与え、生成されたルール数は約 7 万件であった。ルールの自動学習には 135 分を要し、メモリ使用量は最大で 130MB 程度であり、提案手法は一般的な計算機上でも十分動作することが確認できた。プリミティブルールの作成に要した人的工数は 0.5 人日であった。一方、従来手法 B では 441 件の辞書拡張ルールから約 150 万語を新たに辞書登録した。学習用データの解析に 6 人日、ルールの作成に 9 人日を要した。このことから、提案手法では大幅に人的工数を軽減できることが確認できた。

提案手法と従来手法の比較を表 3 に示す。従来手法 A は未知語数が多く、適用数も少ないため、性能は低いといえる。従来手法 B は未知語数は少ないが、悪化数が多いことから、未知語は解消したが、その一部は誤った区切りで形態素解析されており、作業者が意図していないルールの過剰適用が発生していると考えられる。提案手法ではルールをスコアリングすることで、複数のルールから適用すべきルールを選択して利用することができる。ルール同士のスコアの差が閾値以下の場合は曖昧性が高いので適用しないなど、ルールの適用を決定するスコアをパラメータにより設定することも可能である。

5 まとめと今後の課題

本稿では口語的な表現や特有の表記を多く含むブログを対象とした文書正規化手法を提案した。提案手法では与えられたプリミティブなルールの特殊化、結合、汎用化を繰り返すことで効率的に修正ルールを学習する。従来手法である人手による辞書拡張手法との性能比較評価実験により、従来手法の課題であった人的コストの軽減と過剰適用の軽減が確認された。

今後の課題として、修正ルールのスコアリング精度を向上させることで、よりスコアの差が小さいルールも使い分けることができ、文節区切りの悪化数を増加させずに、適用数を増加させることが可能と考えられる。現在は形態素解析結果に含まれる未知語の情報のみを正解ラベル付与に用いているが、形態素解析により得られる他の情報も利用することで、精度の向上が期待される。

参考文献

[1] 風間淳一，光石　豊，牧野貴樹，鳥澤健太郎，松田晃一，辻井潤一：チャットのための日本語形態素解析，言語処理学会第五回年次大会発表論文集，pp. 509–512 (1999).

[2] 竹元義美，福島俊一：口語的表現を含む日本語文の形態素解析の実現と評価，情報処理学会自然言語処理研究会報告，pp. 105–112 (1994).

[3] 松本裕治，伝　康晴：話し言葉の形態素解析，情報処理学会音声言語情報処理研究会報告，pp. 9–14 (2001).