DBSJ Journal, Vol.8, No.1
June 2009

ブログ的表記を正規化するためのルール自動生成方式の提案と評価

An Automatic Rule Generation Method for Modifying Informal Expression in Blog Documents

池田 和史♥　柳原 正♥
松本 一則♠　滝嶋 康弘♠

Kazushi IKEDA　Tadashi YANAGIHARA
Kazunori MATSUMOTO　Yasuhiro TAKISHIMA

ブログ上の文書には口語的な表現や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、十分な精度の解析が行えない。これらのブログ的表現は人手により辞書に登録されることが一般的であるが、人的コストの大きさや高度なスキルを必要とするという問題がある。本稿では、ブログ的表現を文語的な表記に修正するルールを自動的に生成する手法を提案する。提案手法は、(1)修正ルールの適用前後における形態素解析結果の比較から、修正ルールを正解率や適用数によりスコアリングする手法、(2)形態素解析結果の比較に基づき、修正ルールの特殊化と結合を行う手法、(3)能動学習と修正ルールの汎用化により、効率的に修正ルールを学習する手法、を用いる。提案手法を実装し、大規模ブログコーパスを用いて従来手法との性能比較評価実験を行った。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、提案手法では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 27.5%軽減するなど、大幅な性能向上が確認された。

Large amount of colloquial and free expressions in blog documents decrease the accuracy of morphological analysis. Manual registration task of many typical blog expressions for morphological dictionary requires large costs and specialized knowledge. In this paper, we propose an automatic rule generation method which corrects blog expressions to well-organized expressions. Our method has 3 steps, (1) estimate the precision accuracy of morphological analysis and the number of applicable sentences for both cases of sentences modified by a rule and original ones, (2) generate a new specific rule from an abstract one and by merging abstract rules using the above estimations, and (3) update rule set effectively by the active learning method and the rule generalization method. We implemented our method and compared its performance to conventional methods. Quantitative evaluation for the number of unknown words, the accuracy of segmentation, and the cost of manual work was conducted. The evaluation shows the miss segmentation of our method is 27.5% lower than that of conventional one.

♥ 正会員　KDDI 研究所
{kz-ikeda, td-yanagihara}@kddilabs.jp
♠ 非会員　KDDI 研究所
{matsu, takisima}@kddilabs.jp

1. まえがき

近年、インターネットの普及により、一般ユーザによるWeb上での情報発信の手段としてブログが注目されており、ブログを対象とした話題抽出や検索、ランキングなどの研究が盛んに行われ、興味深い研究成果も報告されている[1], [2]。しかし、ブログ上の文書には「じゃん」や「したよ～ん」のような口語的な表現や「ヵゎぃぃ」や「ぁたUぉ」（「あたしは」と読む）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られないことが問題となっている。現在ではこれらの表現を人手により辞書登録す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課題となる。

本稿ではこれらの問題を解決するため、上記のようなブログ的表記を文語的な表記へと修正する手法を提案する。提案手法では文字列変換のルールを用いて、文書の修正を行う。例えば、文字列「ヵゎぃぃ」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。しかし、このような具体的な修正ルールは適用できる場合の数(**適用数**)が少ないため、人手で全て登録することは難しい。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（**プリミティブルール**）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することで、修正ルールの適用数や正解率を算出し、それらにより修正ルールをスコアリングする手法、(2)修正ルールのスコアリング結果を利用して、ルールをより具体的にする**特殊化手法**と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**、(3)修正ルールを効率的に学習するための**汎用化手法**と**能動学習手法**、を提案する。

提案手法を実装し、性能評価実験を行った。性能評価実験はWeb上のブログ記事200万件、約1700万文からなるブログコーパスを用いた。提案手法と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を行い、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。提案手法では従来手法と比べて、文節の区切り方の誤りを27.5%軽減するなど、大幅な性能向上を確認した。また、提案手法における学習コーパス量やプリミティブルール数などの学習条件を様々に変化させたときの計算時間や未知語の減少数を評価することで、学習に要する計算時間と文章修正の性能という提案手法が持つトレードオフについても評価した。

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[3]では、チャットの文章を分析し、人手によって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、辞書登録の人的コストは軽減されるが、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、主観に基づきやすい。我々の予備実験では文献[3]を参考にルールを作成し、200 万文のブログ文章を形態素解析したところ、53488 文に文節区切りの変化が見られたが、そこから 600 文をサンプリングして評価したところ、37.2%の文はルール適用前と比べて文節区切りが悪化していた。

この他にも、口語的表現や話し言葉を言語的な観点などか

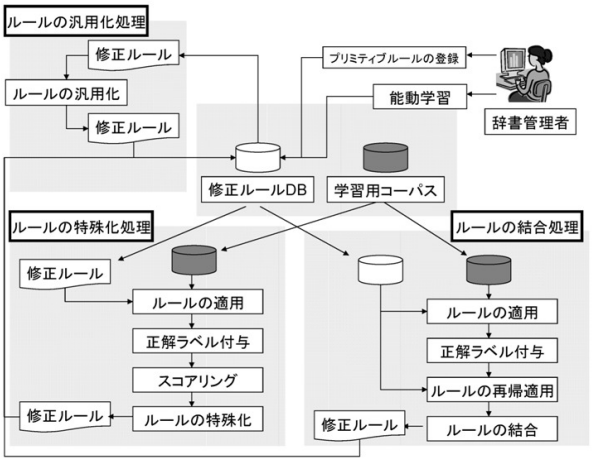

図 1 提案手法における機械学習の全体像
Fig.1 Overview of the Proposed Machine Learning Algorithms

ら分析した形態素解析精度向上のための手法が提案されている。文献[4]では、「～しちゃう」などの口語特有の言い回しを分析し、人手により辞書登録を行うことで、口語の形態素解析精度が向上することが報告されている。同様に、文献[5],[6]では、話し言葉の形態素解析を対象とした研究成果が報告されており、様々な観点から口語的表現を分析し、特徴を列挙している。しかし、上記のような口語的表現の分析は極めて専門的なスキルを必要とする点や辞書登録に多くの労力を要するなど、人的コストの大きさが課題となっている。

形態素解析における未知語の解消についても盛んに研究が行われており、カタカナ語の表記の揺れを解消する手法[7]やWebから新語を獲得する手法[8]、未知語の品詞推定を行う手法[9]などが提案されている。これらの手法は未知語の解消に大きく貢献するが、我々が対象としている口語的な表現やブログ特有の表記の解決を対象としたものではない。

3. 提案手法

提案手法における機械学習の全体像を図 1 に示す。辞書管理者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して、適用することができる。各処理の詳細について説明する。

3.1 正解ラベルの付与

修正ルールのスコアリングを行うため、修正前後の文をそれぞれ形態素解析し、その結果を比較することで、修正が正しいかどうかを判定する。修正ルールの適用前後の文に未知語が含まれるかどうかによって、4 通りの修正結果があり、それぞれに異なるラベルを付与する。表 1 は修正ルール「ょ」⇒「よ」を単語に適用した例であり、修正前後で未知語、既知語（辞書に登録されている語）のいずれであるかによって、"○","×","△","□"のラベルを付与している。

3.2 ルールのスコアリング

修正ルールをコーパスの文に適用し、正解ラベルを付与した結果を用いて、ルールのスコアリングを行う。表 2 は修正ルール「ちゃ」⇒「ては」の適用例であり、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 3 のように、修正ルール「ちゃ」⇒「ては」のスコアリングを行うことができる。同時に修正ルールの特殊化も行うことができ、これについては 3.4 節で説明する。

表 1 正解ラベル付与の例（修正ルール「ょ」⇒「よ」）
Table1 Example of Labeling a Modification Rule

	修正前	修正後	ラベル
（例）	未知語 行こうょ	既知語 行こうよ	○
（例）	既知語 びょうき	未知語 びようき	×
（例）	未知語 行こぅょ	未知語 行こぅよ	△
（例）	既知語 びょういん	既知語 びよういん	□

表 2 修正ルール適用例（「ちゃ」⇒「ては」）
Table2 Example of Sentence Modification

修正前	修正後	ラベル
行かなくちゃ	行かなくては	○
ちゃんとして	てはんとして	×
見なくちゃ	見なくては	○
赤ちゃん	赤てはん	×
おちゃわん	おてはわん	×
しなくちゃ	しなくては	○
メールしなくちゃ	メールしなくては	○
ゲームしなくちゃ	ゲームしなくては	○
くちゃくちゃ	くてはくては	×
困っちゃう	困ってはう	×

表 3 修正ルールスコアリング例（「ちゃ」⇒「ては」）
Table3 Example of Scoring Modification Rules

修正ルール	適用数	正解数	正解率
「ちゃ」 ⇒ 「ては」	10	5	0.5
「くちゃ」 ⇒ 「くては」	6	5	0.83
「なくちゃ」⇒「なくては」	5	5	1.0
「ちゃん」 ⇒ 「てはん」	2	0	0
「おちゃ」 ⇒ 「おては」	1	0	0
「ちゃわ」 ⇒ 「てはわ」	1	0	0
「ちゃう」 ⇒ 「てはう」	1	0	0
「っちゃ」 ⇒ 「っては」	1	0	0

ここで、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。例えば、表 1 の"△"となる例では修正後が「行こうよ」となれば正解であり、修正前と比べると正解に近づいているため、未知語は解消されなかったものの、修正の方向性は正しいと考えられる。一方、"□"となる多くの場合は経験的に修正ルールの適用は不適切である場合が多い。

3.3 プリミティブルールの登録

修正ルールの生成に必要な基本的なルールを人手により事前に登録しておく。ここで登録するのは「ぁ」⇒「あ」のような極めて抽象的なルールであり、専門的なスキルが無くても作成できる。プリミティブルールの例を表 4 に示す。「カタカナをひらがなにする」などは機械的に記述することができ、

表 4 プリミティブルールの例
Table4 Example of Initial Rules

母音を「ー」にする	「あ」 ⇒ 「ー」
小文字を大文字にする	「ぃ」 ⇒ 「い」
カタカナをひらがなにする	「ウ」 ⇒ 「う」
「ー」を削除する	「ー」 ⇒ 「」
発音が似ている語に置換	「ぢ」 ⇒ 「じ」
典型的な口語を文語に置換	「ちゃう」⇒「てしまう」

「ゴキゲン」を「ごきげん」に修正するといった具体的なルールも学習することができる。

3.4 ルールの特殊化

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。特殊化はコーパスに付与された正解ラベルを用いて行う。表 2 のように、「ちゃ」⇒「ては」の修正に対する正解ラベルには「くちゃ」⇒「くては」の修正に対する正解ラベルも含まれる。これをもとに、表 3 のように「くちゃ」⇒「くては」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「見なくちゃ」に修正ルールを適用するとき、「ゃ」⇒「や」や「ちゃ」⇒「ち」などの正解率の低いルールではなく、より正解率の高い「なくちゃ」⇒「なくては」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。ルールを適用するスコアの閾値を設定することで、過剰なルールの適用を抑制することができる。

一般に特殊化ルールは元となったルールと比べ、適用数が少ない。また、特殊化ルールの生成目的上、元のルールよりも正解率が高いことが望ましい。元のルールよりも正解率が低いルール（表 3 の「おちゃ」⇒「おては」など）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。特殊化は 1 つの汎用なルールから複数の特殊化ルールを生成する。ルール数の爆発を防ぐため、文字長が一定値以上になる場合や特殊化しても正解率が向上しない場合はルールを特殊化しないといった制限を設けている。

3.5 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がある。このように、複数の修正ルールを組み合わせて再帰的に適用し、未知語が解消した場合、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 2 に示す。既存のルールとして、「ヵ」⇒「か」、「ゎ」⇒「わ」、「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 2 の状態(1))。このとき、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 2 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 2 の状態(4))。図 2 の状態(3)のように、不適切なルールは生成しない。

3.6 ルールの汎用化

結合ルールや後述のルールの能動学習により得られたルールは極めて具体的であるため、正解率は高いものの、適用数が少なく、そのままでは有用とはいえない。ルールを効率的に学習するため、提案手法ではルールの汎用化を行う。

ルールの結合や能動学習により、具体的な修正ルールが得られたとき、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。

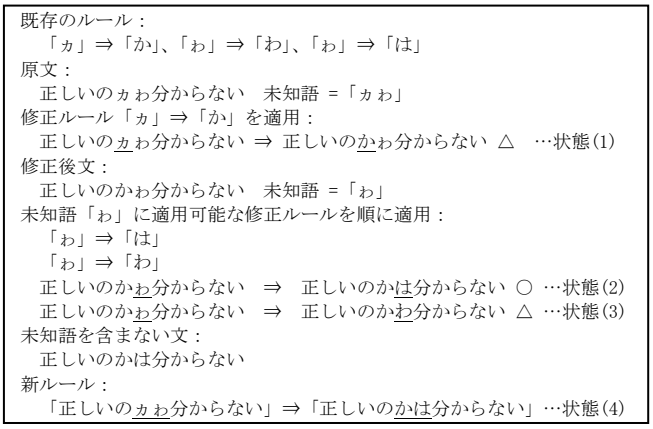

図 2 ルール結合の例
Fig.2 Example of the Merge Algorithms

得られた「ヵゎ」⇒「かは」を特殊化することで、「なのヵゎ」⇒「なのかは」など、新たなルールを得ることができる。このように、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。

3.7 ルールの能動学習

ルールの特殊化や結合、汎用化を用いて繰り返し学習しても未知語が解消されない場合、辞書の管理者に対して、能動的に正解を尋ねる。出現頻度の高い未知語を優先的に尋ね、入力された正解に対して汎用化と特殊化、結合を繰り返し行うことで、能動学習の効果を有効に利用する。

4. 性能評価実験

提案手法を実装し、2 種類の性能評価実験を行った。**実験 1** では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[3]の性能とを比較評価した。従来手法の課題として、人的コストの大きさとルールの過剰適用による文節区切りの誤りが挙げられるため、文献[3]で提示されているルールを作成し、従来手法と提案手法を用いてブログコーパスの文を形態素解析したときの未知語の減少数や文節区切りの正しさについてそれぞれ評価した。また、ルールの作成に要した人的コストについても評価した。

実験 2 では提案手法における学習コーパス量やプリミティブルール数などの学習条件を様々に変化させたときの学習に要する計算時間や未知語の減少数などを評価することで、提案手法が持つ計算時間と性能などのトレードオフについて評価した。各実験について、実験手順と実験結果について説明する。

4.1 実験 1:従来手法と提案手法の比較評価

4.1.1 実験の手順と環境

従来手法を評価するため、文献[3]で提示されているルールと、それを発展させたルールの作成を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依頼した。作業者は文献[3]を参考に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 1000 万文から検出された未知語データを学習用データとして参考にしながら、辞書拡張ルールを追加する（**従来手法 B**）。これらの辞書拡張ルールを機械的に形態素解析辞書に反映する。提案手法も上記の学習用データ 1000 万文を用いて、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 200 万文を用意した。拡張前の形態素解析辞書を**基本辞書**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)文中に出現する未知語の総数（**未知語出現数**）の減少、(2)文節区切りが変化した文数（適用数）(3)適用数に対する文節区切りが向上した割合（**向上率**）、(4)適用数に対する文節区切りが悪化した割合（**悪化率**）、をそれぞれ評価した。文節区切りの向上とは、基本辞書では不正解であった文節区切りが正解になった場合を指す。基本辞書では正解であった文節区切りが不正解になった場合を悪化とする。文節区切りの正解判定は文献[3]と同様に、文献[10]の手法を用いた。人手により文節区切りの正解を付与し、各手法を用いた場合の形態素解析結果の各文節を正解と比較して評価する。例えば、「困っちゃう」は正しくは「困っ/ちゃう」と区切られるべきであるが、基本辞書において「困/っ/ちゃ/う」、従来手法 B において「困っ/ちゃ/う」と区切られていた場合、従来手法 B では「困っ」の文節を正しく区切ることができたと考え、文節区切りが向上したものとする。提案手法により「困ってしまう」に修正された場合「困っ/て/しまう」のように、修正後の文節区切りが正しければ正解とする。それぞれの手法で文節区切りに変化のあった文のうち、600 文に対して評価を行った。

実験では形態素解析器に MeCab [11] (Ver.0.97)を用いた。MeCab 標準の IPADIC 辞書(Ver. 2.7.0)では未知語数が膨大で作業者がルール作成を到底行えないという問題があった。従来手法と提案手法は共に人名などの固有名詞や流行語などが未知語として検出される場合を対象としていないことから、性能評価実験では固有名詞や新語など名詞のみ 18 万語を追加登録した拡張 IPADIC 辞書を用いた。

4.1.2 実験結果

提案手法により生成した修正ルール(**生成ルール**)の例を表 5 に示す。プリミティブルールとして 150 件のルールを与え、生成したルール数は約 130 万件であった。ルールの自動学習には 17 時間を要し、メモリ使用量は最大で 1.3GB 程度であり、提案手法は 1000 万文の学習コーパスに対しても現実的な計算時間で動作することが確認された。プリミティブルールの作成に要した人的工数は 2 時間であった。一方、図 3 は従来手法 B において、人手により作成された辞書拡張ルールの例であり、662 件の辞書拡張ルールから約 178 万語を新たに辞書登録した。辞書拡張ルールの作成に要した人的工数は学習用データの解析に 7 人日、ルールの作成に 10 人日であった。

提案手法と従来手法の未知語出現数、向上率、悪化率を表 6 に示す。提案手法と比較すると、従来手法 A は未知語出現数が多く、文節区切りの悪化率も高い。従来手法 B は未知語出現数が少なく、適用数も多いが、文節区切りの悪化率が高いことから、未知語は解消したが、その一部は誤った文節区切りで形態素解析されていることが分かる。作成したルールが作業者の意図していない過剰適用を起こし、副作用を発生させたと考えられる。提案手法では学習データをもとにルールのスコアリングを行うことで、複数あるルールの中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。スコアの差が一定値以下の場合は曖昧性が高いので、ルールを適用しないなど、ルールの適用を決定するスコアについてはパラメータにより設定することも可能である。

これらの実験結果から、提案手法は従来手法の課題であったルールの過剰適用を大幅に軽減すると共に、極めて少ない人的コストで従来手法と同性能以上の未知語解消能力を発揮できることを確認した。

表 5 提案手法による生成ルールの例
Table5 Example of Generated Rules

修正前	修正後	正解率	適用数
わ	は	0.630	81353
わ	わ	0.489	81353
私わ	私は	0.735	1936
私わ	私わ	0.540	1936
私わいつに	私はいつに	1.000	1
私わいつに	私わいつに	0.000	1
かわいい	かはいい	0.400	20
かわいい	かわいい	1.000	20

1. 品詞が「名詞・形容動詞語幹」ならば語尾に「ッ」が付属する
   例 ″サイアク″ ⇒ ″サイアクッ″, ″ザンネン″ ⇒ ″ザンネンッ″
2. 品詞が「感動詞」かつ表層が「ヨ」で始まるならば「ョ」に置換
   例 ″ヨロシク″ ⇒ ″ョロシク″, ″ヨシ″ ⇒ ″ョシ″
3. 品詞が「名詞・接尾・人名」かつ表層が「母音(a)+ん」で終わるならば「ぁ」を挿入
   例 ″～さん″ ⇒ ″～さぁん″, ″～ちゃん″ ⇒ ″～ちゃぁん″
4. 品詞が「形容詞・自立」かつ表層が「す」で始まるならば「しゅ」に置換
   例 ″すごい″ ⇒ ″しゅごい″, ″すばらしい″ ⇒ ″しゅばらしい″
5. 品詞が「形容詞・自立」かつ表層が「い」で終わるならば「す」に置換
   例 ″キモイ″ ⇒ ″キモス″, ″かわゆい″ ⇒ ″かわゆす″

図 3 従来手法 B による辞書拡張ルールの例
Fig.3 Example of Dictionary Expansion Rules

表 6 各手法の性能比較
Table6 Performance Comparison of Each Method

手法	未知語出現数(件)	適用数(文)	向上率(%)	悪化率(%)
基本辞書	356,232	-	-	-
従来手法 A	326,854	53,488	48.7	37.2
従来手法 B	223,712	265,679	47.3	31.2
提案手法	325,384	58,523	54.0	9.7

4.2 実験 2:提案手法の学習条件と性能の評価

次に、提案手法における学習条件と性能について評価実験を行った。

4.2.1 実験の手順と環境

提案手法において、計算時間や文章修正性能に影響を与えると考えられる要因として、(1)学習コーパス量、(2)プリミティブルール数、(3)特殊化文字長の最大値、(4)能動学習ルール数、(5)アルゴリズムの実行方法、の 5 種類の要因を考慮し、それぞれ表 7 のようにパラメータ値を用意した。各パラメータ値における(a)生成ルール数、(b)修正ルールの適用数、(c)未知語出現数、(d)修正後の文節区切りの向上率と悪化率、(e)ルール学習とルール適用に要するそれぞれの計算時間、について評価した。(5)アルゴリズムの実行方法では、特殊化、結合、汎用化を 1 ラウンドとし、ルール数が増加しなくなるまでラウンドを繰り返して学習を行うという提案手法のアルゴリズムに対して、ラウンド数を 1 回、2 回、3 回のみと制限した場合や、特殊化のみを行った場合、結合、汎用化、特殊化という異なる順序で学習した場合について評価した。(d)文節区切りの向上率、悪化率では前節と同様に文節区切りに変化のあった文のうち、600 文を人手により評価した。(e)ルール適用では学習アルゴリズムによって生成した修正ルールが与えられているとき、学習コーパスとは異なるブログ 200 万文の修正に要する時間を計測した。

上記(1)から(5)の各パラメータの基準値を表 7 において太字で示した値とし、各パラメータを単独で変化させて、評価を行った。具体的には、学習コーパスは 100 万文、プリミティブルール数は 150 件、特殊化文字長の最大値は 5 文字、能

表 7 各パラメータの値
Table7 Value of Each Parameter

学習コーパス量（文）	1 万	5 万	10 万	50 万	**100 万**	500 万	1000 万
プリミティブルール数（件）	50	100	**150**	200	250		
特殊化文字長の最大値（文字）	3	**5**	7	9	11		
能動学習ルール数（件）	**0**	50	100	150	200		
アルゴリズムの実行方法	順序違い	特殊化のみ	1 回	2 回	**3 回**	終了まで	

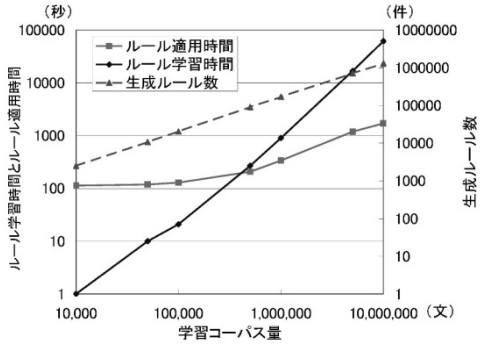

図 4 学習コーパス量と生成ルール数、学習時間、ルール適用時間
Fig.4 Amount of Corpus vs # of Generated Rules, Time of Machine Learning , and Time of Modification

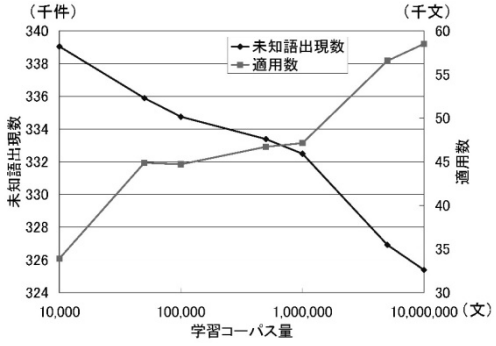

図 5 学習コーパス量と未知語出現数、ルール適用数
Fig.5 Amount of Corpus vs # of Unknown Words and # of Modification

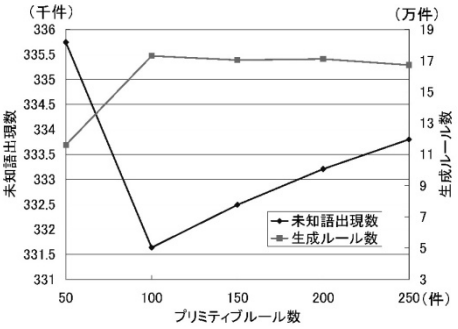

図 6 プリミティブルール数と生成ルール数、未知語出現数
Fig.6 Amount of Initial Rules vs # of Generated Rules and # of Modification

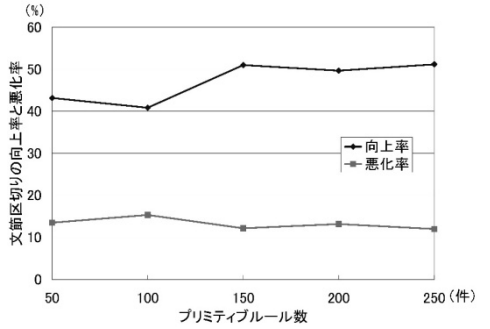

図 7 プリミティブルール数と文節区切りの向上率、悪化率
Fig.7 Amount of Initial Rules vs Miss and Success Ratio of Segmentation

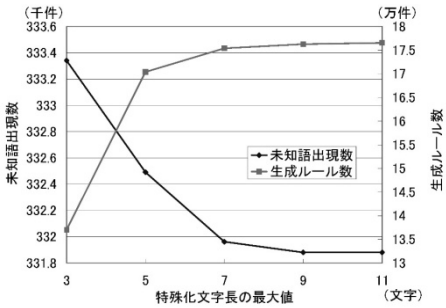

図 8 特殊化文字長の最大値と生成ルール数、未知語出現数
Fig.8 Maximum Word Length of Rule Specification vs # of Unknown Words and # of Modification

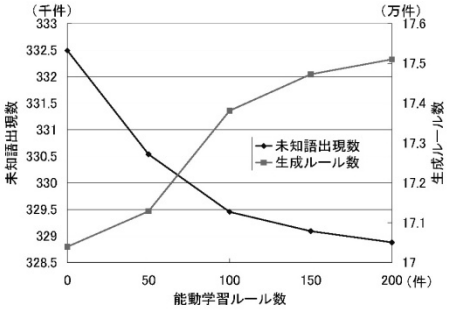

図 9 能動学習ルール数と生成ルール数、未知語出現数
Fig.9 # of Active Learning Rules vs # of Unknown Words and # of Modification

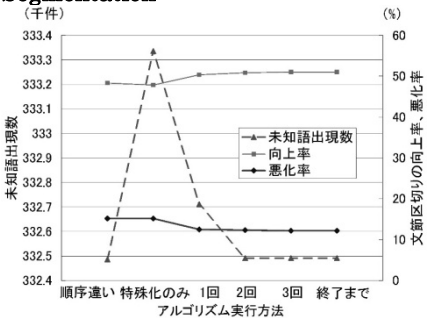

図 10 アルゴリズム実行方法と未知語出現数、文節区切りの向上率、悪化率
Fig.10 Types of Algorithms vs # of Unknown Words and Miss and Success Ratio of Segmentation

動学習ルール数は 0 件、アルゴリズムの実行方法は 3 ラウンドという学習条件を基本とし、学習コーパス量のみを 1 万文から 1000 万文まで 7 通りに変化させて評価を行う、といった手順で実験を行った。評価実験環境は前節と同様とした。

4.2.2 実験結果

図 4 に学習コーパス量を変化させたときのルールの適用時間とルールの学習時間、生成ルール数の関係を示す。生成ルール数は学習コーパス量に比例して増加する。ルールの適用時間は生成ルール数の増加に伴い、増加していると考えられる。学習コーパス量が 1000 万文の場合、ルールの適用に要する時間は 1000 文あたり 1 秒未満であった。ルールの学習時間は学習コーパス量のおよそ二乗に比例して増加するものの、数千万文の学習コーパスに対して、数日間という現実的な計算時間で動作する点は十分に有用であるといえる。図 5 は学習コーパス量とルールの適用数、未知語出現数を比較している。学習コーパス量の増加に伴い、ルールの適用数は増加し、未知語出現数は減少することが分かる。基本辞書にお

ける未知語数が 356232 件であるのに対し、学習コーパスが 1000 万文のとき、未知語の減少率は約 10%と非常に高い性能を発揮することが確認できた。

図 6 にプリミティブルール数と生成ルール数、未知語出現数の関係を示す。プリミティブルール数が 100 件のとき、生成ルール数が多く、未知語も減少しているため、最も性能が良いように見える。しかし、文節区切りの向上率と悪化率を評価した図 7 を見ると、プリミティブルール数が少ないときは文節区切りの精度が悪く、ルールを過剰適用していたことが分かる。プリミティブルール数が増加することで、誤ったルールは他のルールよりも正解率が低くなり、淘汰され、適用されにくくなる。例えば表 5 では、「私ぁ」という表現に対しては「私ぁ⇒私わ」のルールよりも「私ぁ⇒私は」のルールを優先的に適用すべきであることを提案手法で学習できることが示されている。このように、プリミティブルール数を豊富に与えることで、より高精度な修正が可能となる。

図 8 は特殊化文字長の最大値と生成ルール数、未知語出現数の関係を表す。特殊化文字長が増加するに従って、未知語数は減少する。文字長が 9 文字付近を上限に、生成ルール数や未知語出現数に変化の無い、収束状態になっている。文節区切りの精度も同様に、特殊化文字長の増加に伴い、精度が向上し、9 文字付近で収束した。これは提案手法では特殊化してもスコアが向上しないルールはそれ以上特殊化しないことで、不要なルールの生成を抑制し、計算量を削減する機能が有効に働いたためである。

図 9 は能動学習と生成ルール数、未知語出現数の関係を表し、能動学習が未知語数の減少に貢献していることが分かる。追加した能動学習ルール数に比べ、生成ルール数は大幅に大きい。これは能動学習ルールが特殊化、結合、汎用化されることで、有効に利用されていることを示している。

最後に、アルゴリズムの実行方法に関する評価実験結果を図 10 に示す。未知語数の減少と文節区切り精度は 3 ラウンド以降変化が無いことが分かった。学習コーパス量やプリミティブルール数が増加すると収束までのラウンド数は変化すると考えられるが、ルール数の爆発等は起こらないことが確認できた。特殊化のみでは未知語出現数が多く、このことは提案手法の結合と汎用化による学習の有効性を示している。アルゴリズムの順序を変えて学習を行った場合、文節区切りの精度が低下している。これは特殊化される前のルールはスコアリングされていないため、不適切な結合ルールを生成してしまうことが原因と考えられる。

これらの実験から、提案手法が持つ様々なパラメータが性能に与える影響を評価することができた。これにより、提案手法が持つ様々なトレードオフを考慮したパラメータ設定が可能となる。例えば、豊富なプリミティブルールと能動学習ルールを与えることで、人的コストは増加するものの、文章修正の精度や未知語の減少数といった性能は大幅に増加することが期待できる。同様に、学習コーパス量を増やすことで、計算時間は増加するが性能は向上する。また、一度大規模コーパスを用いて学習を行い、生成ルールを取得すれば、ルールの適用に要する時間は生成ルール数に比例する十分短い時間であることなども確認された。

5 まとめ

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、あらかじめ与えられた少量の汎用な修正ルールの特殊化、結合、汎用化を繰り返すことで効率的に学習を行い、自動的に多数の修正ルールを生成する。

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大規模ブログコーパスを用いて実施した。提案手法ではわずか 150 件のプリミティブルールを元に、1000 万文の学習コーパスを用いて現実的な計算時間で学習を行い、未知語を約 10%減少させることに成功した。性能比較評価実験により、従来手法の課題であったルールの過剰適用が 27.5%軽減し、人的コストも大幅に減少することが確認された。加えて、提案手法が持つ様々なパラメータが性能に与える影響についても評価し、豊富なプリミティブルールと能動学習ルールを与えることで、飛躍的に性能が向上することなどを確認した。

[謝辞]

日頃ご指導いただく KDDI 研究所秋葉重幸所長, 松本修一副所長, および菅谷史昭執行役員に深く感謝致します。

[文献]

[1]. 中島伸介，稲垣陽一，草野奉章：高信頼性情報の提示を目指した熟知度に基づくブログランキング方式の提案，日本データベース学会論文誌， Vol. 7, No. 1, pp.257-262 (2008).
[2]. 関口裕一郎，川島晴美，奥田英範，奥雅博：ブログ発信者の特徴を利用した話題抽出手法，日本データベース学会論文誌, Vol. 5, No. 1, pp.9-12 (2006).
[3]. 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).
[4]. 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).
[5]. 竹下敦，福永博信：話し言葉に対する形態素解析，情報処理学会第 42 回全国大会，1C-3 (1991).
[6]. 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).
[7]. 増山毅司，関根聡：大規模コーパスからのカタカナ語の表記の揺れリストの自動構築，言語処理学会第 10 回年次大会論文集，pp.29-32 (2004).
[8]. 三枝優一，古井陽之助，速水治夫 : Web から新語を動的に獲得する形態素解析用辞書拡張方式，情報処理学会データベース・システム研究会報告，pp.77-82 (2007).
[9]. 森信介，長尾眞 : n グラム統計によるコーパスからの未知語抽出，電子情報通信学会技術研究報告 NLC，pp.7-12 (1995).
[10]. Nagata, M.: A Stochastic Japanese Morphological Analyzer Using a Forward-DP Backward-A N-Best Search Algorithm, Proc. of the 15th International Conference on Computational Linguistics, pp. 201-207 (1994).
[11]. 工藤拓 : MeCab. http://mecab.sourceforge. net/.

池田 和史 Kazushi IKEDA
2006 阪大・基礎工・情報科飛び級のため中退。2008 同大大学院博士前期課程了。同年 KDDI(株)入社、研究所所属。自然言語処理などの研究に従事。日本データベース学会正会員。

柳原 正 Tadashi YANAGIHARA
2002 慶大・環境情報卒。2004 同大大学院修士課程了。2005 KDDI(株)入社、研究所所属。リコメンダシステム，テキストマイニング等の研究に従事。日本データベース学会正会員。

松本 一則 Kazunori MATSUMOTO
1984 京大・工・情報工学卒。1986 同大大学院修士課程了。同年国際電信電話(株)入社、研究所所属。現在、KDDI 研究所知能メディアグループにて、マルチメディア検索、コンテンツ配信の研究開発に従事。電子情報通信学会会員。

滝嶋 康弘 Yasuhiro TAKISHIMA
1986 東大・工・電気卒。1988 同大大学院電子工学修士課程了。同年国際電信電話(株)(現 KDDI(株)) 入社。現在、(株)KDDI 研究所知能メディアグループリーダ。この間、動画像の符号化方式、動画通信システム、情報理論の研究・開発に従事。電子情報通信学会、映像情報メディア学会、画像電子学会会員。工博。