Vol.2017-NL-231 No.12
Vol.2017-SLP-116 No.12
2017/5/15

日本語Twitter文書を対象とした系列ラベリングによる表記正規化

大崎彩葉[1,a)]　北川善彬[1]　小町守[1]

概要：
本研究では，Twitter 上で見られる口語的表現やタイピングミス等による一般的でない表記（以下「崩れた表記」と呼ぶ）による形態素解析精度低下を改善するためのアプローチとして，崩れた表記の正規化を試みる．英語のようなスペース区切りの言語の正規化では単語単位で崩れた表現を置き換える手法が考えられるが，日本語のようなわかち書きが必要な言語では，処理対象に崩れた表記が含まれることで単語境界の検出に誤りが生じ，未知語が多く出現するため，単語情報が利用しづらく，単語単位での置き換えは適さない．また，Twitter 文書から作られた大規模なアノテーションデータは入手が困難であるため，大量の教師データを必要とするシステムを使ってこの問題を解くことは難しい．そこで，本研究では文字単位の系列ラベリング問題として正規化を解く．系列ラベリング問題として正規化だけを解くことで，形態素解析と表記正規化を同時に学習するような手法に比べ小規模のデータでの学習が可能になる．また，正規化された表記が付与された実際の Twitter 文書のデータを使って学習，実験を行い，Twitter 文書を対象とするのに適した正規化を学習できることを示す．そうして正規化処理を施した文書と，正規化前の文書，人手で正規化を施した文書を形態素解析にかけて比較し，崩れた表記やその正規化結果が形態素解析の精度に与える影響を分析する．

1. はじめに

近年のソーシャルメディアの普及から Twitter 文書を始めとするウェブ文書の分析の需要が高まり，自然言語処理の研究対象として取り上げられるようになった．しかし，Twitter のような不特定多数が言語情報を発信する場では，新聞などの校正のされた文章に比べ，口語的表現等による一般的な辞書にある表記とは異なる表記が多く見られることが知られている [2]．このような一般的でない表記を本研究では便宜的に「崩れた表記」と呼ぶこととする．これらの崩れた表記は形態素解析を誤らせる原因となり（表 1），後続タスクにも悪影響を与える．崩れた表記に関する言語資源は未だ充分ではないため，大規模な学習データや辞書を必要とする従来の形態素解析器でこの問題に対応することは困難である．

日本語のようなわかち書きが必要な言語では，処理対象に崩れた表記が含まれることで単語境界の検出に誤りが生じることがあることを考慮すると，単語分割と正規化を同時に行う方法と，入力文に正規化を施してから単語分割を行う方法が考えられる．Twitter 文書では崩れた表記以外にも新語のような未知語が出現するため，前者のような辞書情報や言語モデルを利用する方法は適切に機能しない．また，前者のような方法は形態素解析と正規化を同時に学習する必要があるため，小規模なデータから正しく学習することは難しい．そこで，本研究では後者の方法の一つとして，系列ラベリングによる表記正規化を行う．

本研究では，まず日本語 Twitter 文書における崩れた表記について，崩れ方の種類や正規の表記と比較した文長の変化等の観点から分析する．また，文字単位の系列ラベリング問題として正規化を解き，その結果について誤りの原因や改善方法を前述の分析内容を踏まえて考察する．最後に，正規化処理を施した文書を形態素解析にかけ，その結果を分析することで，崩れた表記の正規化が形態素解析の精度に貢献することを示す．

2. 関連研究

Twitter などのウェブ文書は一般的な辞書にある表記とは異なる表記を含むことがある．ウェブ上に存在する表記のバリエーションはネットスラングやタイピングミス，口語表記等多岐にわたるが，口語表記の元となる話し言葉の書き起こしに対しては今ほどソーシャルメディアが普及する前から研究が行われてきた [9]．このような崩れた表記

[1] 首都大学東京
[a)] osaki-ayaha@ed.tmu.ac.jp

表 1 ウェブ文書でみられる崩れた表記に関する解析誤りの例と文長変化

崩れの種類	崩れた表記	本来の表記	文長変化
略語・ ネットスラング	おはです お（接頭辞）/は（名詞）/です（助動詞）	おはようです おはよう（感動詞）/です（助動詞）	+2
タイピングミス	こんんいちは こ（動詞）/ん（助動詞）/ん（名詞）/いち（名詞）/は（助詞）	こんにちは こんにちは（感動詞）	-1 (-2+1)
口語表記	てゆーか て（名詞）/ゆーか（名詞）	というか と（助詞）/いう（動詞）/か（助詞）	0 (-3+3)

は形態素の解析誤りの原因となることが知られており [2]，形態素解析の誤りが伝播することで後続タスクの精度低下の一因となっている．

崩れた表記に起因する解析誤りの対応策の一つとして，崩れた表記を含む文書を用いて形態素解析の学習を行うという方法が考えられる．[2] では，実際の Twitter 文書を元に辞書を拡張することで Twitter 文書の形態素解析を行っている．しかし，このような方法で精度を向上させるには形態素解析の学習に使えるだけの大規模なアノテーションデータが必要となる．日本語のコーパスは，日本語書き言葉均衡コーパス [13] が広く利用されており，Yahoo!知恵袋，Yahoo!ブログといったウェブ文書も含まれる各ジャンルに対して，単語境界と，付加情報として，品詞，活用，基本形，読みなどがアノテーションされている．その他に，大学生によって書かれたブログ 249 記事，4,186 文からなる，京都大学ブログコーパス [12]，ウェブ文書 15,000 文に対して，形態素・固有表現・構文・格関係，照応・省略関係，共参照の情報を付与した京都大学ウェブ文書リードコーパス [1] などがある．しかし，以上で挙げられるウェブ文書は比較的綺麗な文書である．崩れた表記を含む Twitter 文書を対象としたコーパス作成の研究もあるが [8]，人手でのコーパス構築はコストが高く，数千文程度の小規模のものしか存在しないため，形態素解析器の学習データとして利用するには前述の大規模なコーパスと併せて拡張辞書のような形で利用するといった工夫が必要である．

別の方法として，崩れた表記に対して正規化処理を施す方法が挙げられる．ルールベースの形態素解析器 JUMAN [3] では，小文字書きや長音記号などの崩れた表記に対して，シンプルな正規化ルールを定めることで対処している [6]．人手でルールを定めるには，データ分析に多くの時間や労力がかかる．対して，コーパスから正規化を学習する方法ならば，人手でルールを定める方法よりは分析のために必要なコストは少ない．正規化を用いる手法では，形態素解析に関して既存のコーパスやツールを利用できるため，新たに用意すべきデータやシステムが少なくて済むという利点がある．

英語で書かれたウェブ文書を対象とした正規化に関して，単語単位の言い換えを用いた Wang らの研究 [11] があるが，英語のようにあらかじめ単語境界が明確に示されている言語と違い，日本語文書の正規化では単語境界の情報を正確に得ることが難しいため，そのまま適用することができない．単語境界情報に依存しない正規化の研究として character embedding を利用した Chrupala らの研究 [7] がある．これは文字 n-gram と文字の分散表現を素性とした CRF を用いた正規化手法で，文単位での正規化が可能な手法である．単語境界の情報を必要としない手法は日本語文書正規化にも応用しやすいが，日本語と英語の崩れた表記の出現傾向や文字種の違い等が解析精度に影響を与える可能性が考えられる．

日本語ウェブ文書を対象にしたテキスト正規化の研究としては，Saito らの研究 [10] では人手で正規化ルールを定めるコストを削減するために，対象を口語的表現と異表記に絞って文字列アライメントをとり統計的に正規化ルールを求め，単語ラティスを構築することによって正規化を行っている．本研究は正規化の対象を限定せず，文字単位の系列ラベリングによって単語情報を必要としない正規化を行う．佐々木らの研究 [15] では，Twitter 文書の表記正規化を，本研究と同様に「入力文にどのように手を加えると正規化された文が出力されるか」という内容をラベルとして付与する系列ラベリング問題として解いている．一方，佐々木らの研究では 1 文字から 1 文字への置換と削除についてのみ考えており，正規化後の文が入力文より長くなる場合については考慮していない．Ikeda らの研究 [4] では，機械翻訳分野でしばしば用いられる Encoder-Decoder モデルを使って崩れた表記を含む文から正規化された文を生成するシステムを考案している．Encoder-Decoder モデルは学習に大規模で高品質なデータを必要とするため，彼らの実験では佐々木らの研究と同様の文字単位の CRF による系列ラベリング手法と比較して高い精度とならなかった．

3. 崩れた表記の分析

この節では，システム設計や実験結果の分析の準備として，崩れた表現の分類や正規化に関する諸統計を示す．集計対象としたのは首都大日本語 Twitter コーパス [8] にてアノテーションされた 1,403 文 19,179 形態素である．このデータセットは 2015 年 10 月頃に収集された実際の Twitter 文書に対し短単位 [*1] を基準に単語境界・品詞・正

[*1] http://pj.ninjal.ac.jp/corpus_center/bccwj/morphology.html

表 2 「おっはょぉ〜」から「おはよう」への正規化ラベル系列

表層形	お	っ	は	ょ	ぉ	</s>
ラベル	NIL	DEL	NIL	DEL	DEL	INS(よう)
正規形	お		は			よう</s>

規形についての情報を付与したものである．首都大日本語コーパスではウェブ文書特有の表記や語彙に焦点を当てたアノテーションがされており，ネットスラングや略語等の品詞付与や，過剰に挿入された長音・母音・感嘆符の削除，片仮名と平仮名の置き換え，「〜じゃね？」→「〜じゃない？」「しちゃう」→「してしまう」等の口語と文語の置き換え等の単語単位での正規化情報の付与がされているため，本研究の実験に採用した．

まず，コーパス中に出現する崩れた表現を含む形態素を分類分けして出現回数を調べたものを表 3 左に示す．出現回数が最も多い長音・母音の挿入は，比較的単純な正規化ルールで正規化できる崩れた表記であるが，口語的表現やタイプミス・常用外の変換のような，人手で正規化ルールを書き出すのは難しい種類の崩れた表記も，高い割合で出現することがわかった．

表 3 右に崩れた表記の種類ごとの単語長の変化の集計を示す．原因としては「はーい」「〜だなぁ」等，長音や母音の挿入が多く行われる傾向があることが考えられる．本研究で定めた崩れた表記の分類上では，正規化前より正規化後の方が単語長が長くなることが多いものは口語的表現，方言，母音の欠落，ネット用語・略語・俗語の 4 種類だけであった．また，増減共に 1 単語あたりに 4 文字以上の文長変化は 27 件しか見られなかったことから，正規化による文長の変化は概ね 3 文字以下であると言える．

また，表 1 に例示するような崩れた表記を含む文に対する正規化後の文長の変化を算出し，統計を取った．図 1 に正規化による文長の変化の分布を示す．正規化前に比べ 1 文字分文長が短くなるものが最も多く見られ，全体でも文長が長くなるものより短くなるものが多く見られた．正規化処理における文字の削除，挿入の内訳を集計したところ，挿入が 800 回，削除が 3,132 回であった．

4. 文字単位の系列ラベリングによる表記正規化手法

本研究では表記正規化の手法として，未知語が多く出現する可能性のある Twitter 文書の処理では単語情報を利用する方法は十分に働かない可能性を考慮して，Chrupala ら [7] や佐々木ら [15] のような文字単位のアプローチを考える．また，入力された文字から直接正規化後の文字を当てようとすると，学習データの量に対し出力側のクラス数が増えすぎてしまうことを考慮し，直接文字を当てるのではなく，「入力文にどのように手を加えると正規化された文が出力されるか」という内容をラベルとして付与する系列ラベリング問題として正規化を試みる．

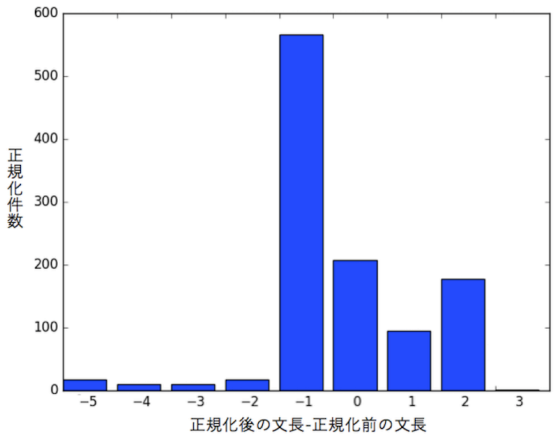

図 1 正規化前と正規化後の文長の変化の分布

表 2 に「おっはょぉ」という文を「おはよう」に正規化する際の正規化のラベリングを例示する．「NIL」はラベルに対応する文字を操作しない，「DEL」はラベルに対応する文字を削除する，「INS()」はラベルに対応する文字の前に文字を挿入する操作を表す．一般に編集距離について議論する際は削除，挿入，置換の 3 つの処理を考える．佐々木らの研究 [15] では，削除と 1 文字対 1 文字の置換のみラベルを付与しているが，3 節で示したように Twitter 文書の正規化の性質として正規化処理の前後で文長が変化しない場合の件数より変化する件数の方が多いことがわかっている．また，出力のラベル数を減らす観点から，本研究では置換には個別のラベルを割り当てないこととする．

5. 日本語 Twitter 文書の正規化実験

5.1 データ

本研究ではデータセットとして，首都大日本語 Twitter コーパス [8] を使用する．まずコーパスを 1,124 文と 282 文に分割し，それぞれのデータを，崩れた表記を含む表層形の文と，正規化情報が付与されている崩れた表記の語を正規化された形に直した正規形の文の対に変換する．レーベンシュタイン距離を求めるアルゴリズムと同じ要領で動的計画法を使い，1,124 文の方のデータの表層形の文を正規形の文に正規化する際の正規化ラベルを求め，それを学習データとする．282 文の方は評価データとする．

5.2 評価尺度

正規化の精度について，本研究では正規化後の文を正解データの文に編集する際に増減する文字数を用いて以下の式で評価を行う．D は削除された文字数，I は挿入された文字数，N は編集操作をしなかった文字数である．

$$character\ error\ rate = \frac{D + I}{N + D + I}$$

また，正規化処理を施した文に形態素解析を行い，その

表 3 崩れた表記の種類と各出現回数，単語長変化

崩れた表記の種類	例	出現回数	単語長変化							
			≦-4	-3	-2	-1	0	1	2	3
長音・母音の挿入	美味しーい（美味しい） くるぞおおお（くるぞ）	195	23	9	19	142	1	1	0	0
口語的表現	じゃね？（じゃない？） ゆって（いって）	141	1	0	2	5	38	90	1	0
記号の挿入	起床！！！！！！！！！！（起床！！） ト☆イ☆レ（トイレ）	114	24	8	5	44	3	0	0	0
タイプミス・常用外の変換	大事ｙぉ鵜部（大丈夫） ホンキ（本気）	81	2	2	5	17	50	3	2	0
方言	取れへんかった（取れなかった）	57	0	0	0	4	37	11	5	0
母音⇔長音の置換	おはよー（おはよう）	47	0	0	3	3	39	1	0	0
母音の欠落	ありがと（ありがとう）	35	0	0	0	1	2	31	1	0
音便化	ぶっこんで（ぶちこんで）	29	0	0	0	0	29	0	0	0
促音の挿入	ほっっっっとんど（ほとんど）	25	2	3	2	17	1	0	0	0
ネット用語・略語・俗語	だお（だよ） おは（おはよう）	20	0	0	0	1	6	3	7	3
小文字	てめぇ（てめえ）	10	0	0	0	2	8	0	0	0
似た発音の文字との置換・ 崩れた発音の書き起こし	まぢ（まじ） ふぃぎゃー（フィギュア）	9	0	2	0	0	5	1	0	0
オノマトペやフィラーの 過剰な繰り返し	ペロペロペロペロペロ（ペロペロ） うふふふふふふふふふ（うふふ）	5	3	0	0	1	0	1	0	0

結果から正規化を施したことで形態素解析に与えられた影響を観察する．ただし，使用したデータセットには正規化後の文の単語分割境界や品詞の情報は付与されておらず，正規化による形態素解析精度の変化を数値として測ることは困難であるため，本研究ではあくまで処理結果の一部について考察をするに留める．

5.3 モデルとツールキット

本実験では窓幅 5（前後 2 文字）でターゲットの周辺の文字を見て，その範囲内にある全ての文字 1-gram から 3-gram を素性テンプレートとする linear-chain CRF [14] で学習を行った．ツールキットとして CRF++ ver.0.58[*2] を使用した．

JUMAN ver.0.996 [3] をベースラインとし，正規化処理を施す前の文（NO-OP），JUMAN による正規化処理を施した文，本手法による正規化処理を施した文の，それぞれについて 5.2 節の評価尺度で正規化精度を評価する．

正規化処理を施した文に対する携帯素解析の評価実験に用いる形態素解析ツールとして KyTea ver.0.4.7 [5] を使用する．本来 KyTea の解析結果は語幹と語尾が分割された超短単位の形で出力されるが，本研究ではデータセットのアノテーション基準と揃える観点から，KyTea の出力結果を語幹と語尾を繋げたものを一形態素とする短単位基準に変換したものを分析する．

[*2] https://taku910.github.io/crfpp/

表 4 Twitter コーパスにおける正規化精度と出力されたラベルの件数

手法	character error rate	NIL	INS	DEL
NO-OP	0.0585	(8,377)	(0)	(0)
JUMAN	0.0551	8,263	21	53
本手法	0.0254	8,197	6	134
gold	(0.0000)	7,899	79	359

5.4 実験結果

表 4 に，5.2 節の評価尺度に基づいた各モデルに対する正規化の精度と出力されたラベルの種類別の件数を示す．JUMAN の実際の出力は正規化された単語でありラベルではないが，本研究の手法と比較するために便宜上，入力文を JUMAN の出力文に編集する際に付与されるラベルを JUMAN の出力ラベルとする．ベースラインのエラーレートが 0.0551 であるのに対し，本手法で正規化を行った結果 0.0254 まで改善した．

正規化に失敗した箇所について，形態素単位で数えて 67 個の正規化できなかった箇所があった．内訳を表 5 に示す．

6. 考察

6.1 正規化処理について

表 6 に本実験で正規化ラベリングが正しく行われた例を示す．日本語 Twitter 文書でもっとも出現頻度が高かった余分な長音記号や記号の削除や，オノマトペの過剰な繰り返しの削除等については正規化が正しく行われていることが確認できた．

表 5 正規化できなかったものの内訳

種類	件数
口語的表現	30
タイプミス・常用外の変換	10
長音・母音の欠落	6
ネット用語・略語・俗語	5
音便化	4
小文字	3
似た音の文字との置換・崩れた発音の書き起こし	3
長音・母音の挿入	2

表 7 に正規化ラベリングが誤ったラベルを出力した例を示す．余分な文字を削除した後に別の文字を挿入しなければならない箇所について，挿入が行われなかったことによる誤りが多く，27 箇所見られた．INS ラベルの付与が正しくされない理由として，学習に用いたデータが非常に小さかったことに加え，3 節の分析より実際の編集操作には削除が多いことがわかっているので，DEL ラベルの付与についての学習に比べ INS ラベルの付与についての学習が十分にされなかった可能性が挙げられる．また，INS ラベルで挿入され得る文字列の種類数だけ INS ラベルの種類も増えるので，INS ラベルは学習データ内での同じラベルの出現回数が少なく，INS ラベルの予想は NIL ラベル・DEL ラベルより難しいことが考えられる．INS ラベルを正しく付与できるようにするための改善策として，INS ラベルと同時に挿入すべき文字列も予想させる方法ではなく，先に NIL・DEL・INS の 3 値分類を解いてから，ラベルが INS だった場合は挿入する文字列を予想するという 2 段階のシステムを用いることが考えられる．

表 5 で示したように，正規化できなかったものでは口語的表現が最も多く，原因としては DEL ラベルと INS ラベルが両方付与されなければいけないものが口語的表現の正規化では多く，前段落で述べたような理由で正規化ができなかったと考えられる．正規化が成功している口語的表現は「もんのすごく」→「ものすごく」のような削除のみのものがほとんどであった．ネット用語・略語・俗語については元の件数に対し正規化できなかった件数が多かった．考えられる理由としては，そもそもネット用語や俗語は普通の単語に比べて使われにくいので，首都大 Twitter コーパスのような小規模なコーパスでは出現回数が少なく十分な学習ができなかったことが考えられる．

6.2 形態素解析について

表 8 に，正規化処理を施したことで形態素解析結果が向上した例を示す．長音等の挿入により単語分割が誤ってしまう様子と，正規化によりそれが解消された例が見られた．

表 9 に，正規化処理を施したが解析誤りが起こってしまった例を示す．一般に漢字で表記されるようなものを平仮名表記することは解析誤りの原因となり得ることが知られているが [2]，本実験で用いたデータセットでは，平仮名を漢字に直すような正規化情報は扱われていないため，そのような処理は学習されていない．そのため，不自然に平仮名が連続する文が生成され，そのような箇所での解析誤りが見られた．また，文章として元々成り立っていない，文脈が不自然な文章に対しては，表記が正しくても解析誤りが発生しやすいことが考えられる．このような文章に関しては表記正規化の観点からアプローチすることは困難であり，文破綻検出等の処理を挟む必要が考えられるが，本実験の結果ではほとんど見られなかった．

7. おわりに

本研究では崩れた表現を含むウェブ文書の形態素解析精度向上のアプローチとして，CRF による文字単位の系列ラベリングを用いて崩れた表記の正規化に取り組んだ．本手法は正規化に関して，既存手法を上回る精度を示した．また，正規化を施した文を形態素解析器にかけた結果を分析し，正規化が形態素解析に与えた影響を確認できた．

今後は CRF を用いた学習に日本語特有の特徴や正規化における傾向を利用したモデルの考案，より大規模なコーパスでの実験と分析，正規化処理による形態素解析の精度向上の数値計測のためのデータ拡張等の発展を図りたい．

参考文献

[1] 萩行正嗣, 河原大輔, 黒橋禎夫. 多様な文書の書き始めに対する意味関係タグ付きコーパスの構築とその分析. 自然言語処理 2014, Vol. 21, No. 2, pp. 213-247, 2014.
[2] Nobuhiro Kaji and Masaru Kitsuregawa. Accurate word segmentation and POS tagging for Japanese microblogs: Corpus annotation and joint modeling with lexical normalization. In Proceedings of EMNLP 2014, pp. 99-109, 2014.
[3] Sadao Kurohashi and Daisuke Kawahara. Japanese morphological analysis system. JUMAN version 5.1 manual, 2005.
[4] Taishi Ikeda, Hiroyuki Shindo, and Yuji Matsumoto. Japanese text normalization with encoder-decoder model. In Proceedings of WNUT 2016, pp. 118-126, 2016.
[5] Graham Neubig, Yosuke Nakata, and Shinsuke Mori. Pointwise prediction for robust, adaptable Japanese morphological analysis. In Proceedings of ACL Volume 2, pp. 529-533. 2011.
[6] 笹野遼平, 黒橋禎夫, 奥村学. 日本語形態素解析における未知語処理の一手法—既知語から派生した表記と未知オノマトペの処理—. 自然言語処理, Vol. 21, No. 6, pp. 1183-1205, 2014.
[7] Grzegorz Chrupała. Normalizing tweets with edit scripts and recurrent neural embeddings. In Proceedings of ACL Volume 2, pp. 680-686, 2014.
[8] 大崎彩葉, 唐口翔平, 大迫拓矢, 佐々木俊哉, 北川善彬, 堺澤勇也, 小町守. Twitter 日本語形態素解析のためのコーパス構築. 言語処理学会 第 22 回年次大会 発表論文集, pp. 16-19, 2016.
[9] Kiyotaka Uchimoto, Chikashi Nobata, Atsushi Ya-

表 6 正規化の成功例

入力：	ペ	ロ	ペ	ロ	ペ	ロ	ペ	ロ	ペ	ロ				
出力：	NIL	NIL	NIL	NIL	DEL	DEL	DEL	DEL	DEL	DEL				
入力：	き	み	ー	の	ー	そ	ー	ば	ー	で	ー	み	る	ー
出力：	NIL	NIL	DEL	NIL	DEL	NIL	DEL	NIL	DEL	NIL	DEL	NIL	NIL	DEL

表 7 正規化の失敗例

入力：	そ	う	ゆ	う	問	題	じ	ゃ	ね	ー	し
出力：	NIL	NIL	DEL	NIL	NIL	NIL	NIL	NIL	DEL	DEL	NIL
（正解：	NIL	NIL	DEL	INS(い)	NIL	NIL	NIL	NIL	DEL	DEL	INS(ない)）
入力：	届	か	ん	の	？						
出力：	NIL	NIL	DEL	NIL	NIL						
（正解：	NIL	NIL	DEL	INS(ない)	NIL）						

表 8 形態素解析の成功例

正規化前：ハッスルー！	ハッ/スルー/！
正規化後：ハッスル！	ハッスル/！
正規化前：思い出すわぁ	思い出す（動詞）/わぁ（名詞）
正規化後：思い出すわ	思い出す（動詞）/わ（助詞）

表 9 形態素解析の失敗例

正規化前：いいないいないいなぁ〜	いい/ないい/な/いい/なぁ/〜
正規化後：いいないいないいな	いい/ないい/な/い/い/な

mada, Satoshi Sekine, and Hitoshi Isahara. Morphological analysis of a large spontaneous speech corpus in Japanese. In Proceedings of ACL Volume 1, pp. 479-488, 2003.

[10] Itsumi Saito, Kugatsu Sadamitsu, Hisako Asano, and Yoshihiro Matsuo. Morphological analysis for Japanese noisy text based on character-level and word-level normalization. In Proceedings of COLING, pp. 1773-1782, 2014.

[11] Wang Ling, Chris Dyer, Alan W Black, and Isabel Trancoso. Paraphrasing 4 microblog normalization. In Proceedings of EMNLP 2013, pp. 73-84, 2013.

[12] 橋本力, 黒橋禎夫, 河原大輔, 新里圭司, 永田昌明. 構文・照 応・評判情報つきブログコーパスの構築. 言語処理学会第 15 回年次大会発表論文集, pp. 614-617, 2009.

[13] 前川喜久雄. KOTONOHA『現代日本語書き言葉均衡コーパス』の開発 (< 特集 > 資料研究の現在). 日本語の研究, Vol. 4, No. 1, pp. 82-95, 2008.

[14] Lafferty, John, Andrew McCallum, and Fernando Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Proceedings of ICML. Vol. 1, pp.282-289, 2001.

[15] 佐々木彬, 水野淳太, 岡崎直観, 乾健太郎. 機械学習に基づくマイクロブログ上のテキストの正規化. 人工知能学会第 27 回全国大会, 4B1-4, 2013.