2-3-3　日本語テキスト正規化
2-3-3　Japanese Text Normalization

東山 翔平
HIGASHIYAMA Shohei

ソーシャルメディアやブログなどのユーザ生成テキストでは、標準的な表記法から逸脱した「崩れた」表記が多用されることから、言語処理システムの精度が低下する問題が起こる。本稿では、同問題に対処するための日本語の語彙正規化タスクについて解説し、著者らが取り組んできた二つの研究を紹介する。一つ目の研究では、語彙正規化のための評価用コーパスを構築・公開することで、従来及び将来のシステムの性能比較を可能とした。二つ目の研究では、疑似ラベル付きデータの生成法と、テキスト編集に基づく語彙正規化法を提案し、従来手法よりも高い正規化精度を達成した。

Text normalization is important for overcoming the problem that non-canonical sentences in user-generated text degrade the performance of general natural language processing systems. This paper describes the authors' work on Japanese text normalization that constructed a manually annotated evaluation corpus and proposed a normalization system based on text editing. The evaluation corpus can be a useful benchmark for comparing and analyzing existing and future systems. The proposed normalization system trained with pseudo labeled data outperformed an existing system.

1　はじめに

ブログ、ソーシャルメディア、電子掲示板などへ投稿される文章を指すユーザ生成テキストでは、標準的な表記法から逸脱した「崩れた」表記・表現が多用されることから、「整った」テキストを前提とする言語処理システムの精度が低下する問題が起こる。たとえば、図1のように、一般的な対訳コーパスで学習された機械翻訳システムでは、崩れた表記の文に対する適切な翻訳結果を出力できないことがある。

何らかの自然言語処理タスク（目的タスクと呼ぶ）を実行する際、崩れた表記に対処する方法として、(a) 崩れたテキストを直接処理できるような目的タスクのモデルを学習する方法[*1]と、(b) 崩れた表記（以降、崩れ表記と呼ぶ）を標準的な表記（以降、正規表記と呼ぶ）に変換する正規化処理を事前に適用した上で、目的タスクの一般的なモデルを用いる方法がある。(b) の方法には、多様な崩れ表記への対応の問題と目的タスクの学習の問題とを分離することで、目的タスクのモデルに変更を加えずに様々なタスクへ正規化処理を応用できる利点がある。特に、ユーザ生成テキストの機械翻訳という目的タスクを考えた場合、日本語テキスト正規化を介することで、日本語から任意の言語への翻訳精度向上に貢献することが期待できる。

本稿では、著者らが取り組んできた日本語のテキスト正規化の研究 [2][3] を紹介する。

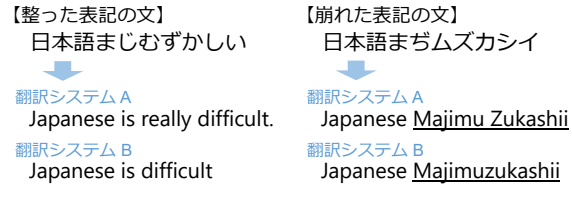

図1　崩れたテキストの機械翻訳結果（オンライン翻訳システム A，B）の例

2　日本語形態素解析・語彙正規化タスク

日本語のテキストを処理する際、単語やそれに準じる処理単位であるトークンへ文を分割する処理が必要であり、その代表的な枠組みの一つが単語分割（自動

＊1　(a) に該当する当研究室での研究として、ユーザ生成テキストの疑似対訳データを生成して機械翻訳モデルを学習した Benjamin らの研究 [1] がある。

分かち書き）である。日本語の単語分割は、図2に示すように、品詞付与や活用・原形推定と合わせた複合的な処理である形態素解析[*2]として解かれることが多い。また、ユーザ生成テキストを処理する際には、単語レベルでのテキスト正規化を指す語彙正規化[*3]を同時に行うことで形態素解析精度も向上することが報告されている [5][6]。著者らの研究でも、単語分割、品詞付与、語彙正規化を同時に行う複合的な処理に取り組んでいる。

3　日本語ユーザ生成テキストコーパスの構築

日本語の語彙正規化の従来研究では非公開データでシステムを検証しているため、異なるシステム間の性能を比較したり、システム横断的な課題を発見したりすることが困難であった。そこで、著者らは、形態素解析及び語彙正規化の精度評価のためのコーパスを構築し、ベンチマークデータとして一般公開することで、上記の問題を解決することを目指した。具体的には、ブログ及び質問サイトの投稿テキストに対し、形態素（単語）情報と正規化情報を付与したコーパス BQNC（Blog and Q&A Site Normalization Corpus）を構築した[*4]。

コーパスの構築は、次の三つの方針に基づき行った。一つ目は、コーパスを公開して第三者が利用可能とすることである。そのために、国立国語研究所の BCCWJ [7] に収録されているウェブテキストの原文に対してアノテーション[*5]を行い、アノテーション情報を公開することで、利用者が原文とアノテーション情報を入手してアノテーション情報付きデータを復元できるようにした。二つ目は、既存の代表的な単語分割基準及び品詞体系に準拠することである。そのために、国語研究所による多くのコーパスで採用されている短単位の基準に従いつつ、正規化を扱う際に問題となる事例を考慮して追加の基準を定義した。三つ目は、ユーザ生成テキスト特有の言語現象を評価・分析可能とすることである。そのために、ユーザ生成テキストに頻繁に出現する特徴的な事例を単語カテゴリとして分類・整理し、カテゴリ情報もテキストにアノテーションした。

以降、本節では、著者らが分類した単語カテゴリの概要と、本コーパスを用いた従来システムの評価実験を紹介する。

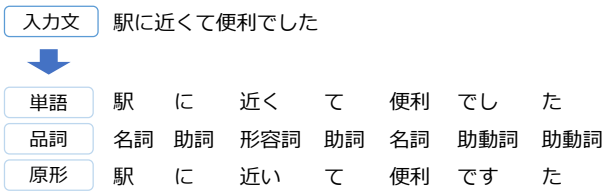

図2　入力文に対する形態素解析（単語分割、品詞付与、原形推定）処理の例

＊2　「形態素」は意味を有する最小の言語単位を指し、言語学的には単語よりも細かい粒度の単位である。ただし、日本語自然言語処理では形態素と単語を厳密に区別しないことが多く、（形態素または単語への）分かち書きを含む一連の処理を形態素解析と呼んでいる。

＊3　以降、単に正規化と呼ぶ場合には語彙正規化を指すものとする。文レベルでのテキスト正規化の研究 [4] も行われているが、入力テキストのどの範囲をどのように変換したかという情報を得ることが難しい点が問題になり得る。

＊4　本コーパスは次のサイトにて公開している。https://github.com/shigashiyama/jlexnorm

＊5　原文テキストに、人間による分析や機械処理に有用な何らかの付加情報を付与することをアノテーションと呼ぶ。本コーパスでは、単語の区切りに相当する位置情報や、単語の品詞の情報などを付加情報として付与した。

表1　BQNC における単語カテゴリ

	カテゴリ	例	正規表記
語彙種別	スラング	コピペ	
	固有名	ドラクエ	
	オノマトペ	キラキラ	
	感動詞	おお	
	方言	ほんま	
	外国語	EASY	
	顔文字・アスキーアート	(^—^)	
異表記種別	異文字種	カワイイ	かわいい / 可愛い
	代用表記	大きぃ	大きい
	音変化	おいしーい	おいしい / 美味しい
	誤表記	つたい	つらい / 辛い

3.1　単語カテゴリの概要

ユーザ生成テキストに特徴的かつ単語分割の誤解析を生じ得るという観点で著者らが分類した単語カテゴリを表 1 に示す。これらのカテゴリは、分類の粒度は異なるものの、従来研究 [8][9] で報告されている現象とおおむね同等の現象をカバーしている。語彙的な種類の観点から 7 カテゴリに分けたものを語彙種別と呼び、異表記の発生過程の観点から 4 カテゴリに分けたものを異表記種別と呼び、各単語は何らかの語彙種別のカテゴリと何らかの異表記種別のカテゴリに同時に該当し得る。

語彙種別の各カテゴリの共通の特徴は、新しい単語が際限なく生まれたり他言語・方言から借用されたりする点にあり、言語処理システムにとって未知の単語がテキストに出現した場合、誤解析の原因となる。異表記種別のカテゴリは、従来からある単語か新しい単語かにかかわらず、標準的な表記とは異なる表記で書かれた単語が該当し、そのような表記は無数に生じ得ることから、やはり誤解析の原因となる。具体的には、日本語の複数の文字種のうち規範的なテキストでの使用が稀である表記を指す「異文字種」(「可愛い」や「かわいい」に対する「カワイイ」など)や、口語の発音を再現したような表記を指す「音変化」(「おいしーい」など)、本来の表記と視覚的に近い文字を用いた表記などを指す代用表記 (「大きぃ」など)、入力誤りなどによる「誤表記」の 4 カテゴリを定めた。

3.2　実験設定

構築したコーパスの統計情報を表 2 に示す。本コーパスは、929 文、延べ語数 12,600 単語から成り、何らかの異表記種別に該当する崩れ表記 767 件を含む。

本コーパスを用いて、条件付き確率場 (Conditional Random Fields) に基づく代表的な日本語形態素解析システムである MeCab [10] と、Sasano らの形態素解析・正規化法 [11] を著者らが MeCab を用いて再現したシステム MeCab+ER[*6] の二つの従来手法の精度を評価した。Sasano らの方法は、人手定義した 5 種類の正規化ルールにより正規化を行う方法であり、たとえば、「冷たーーい」に対しては長音を削除するルールを適用することで「冷たい」が正規表記候補に追加される。両システムとも、短単位に基づく UniDic [12] の形態素解析用辞書 (unidic-cwj-2.3.0) を用いた。UniDic 解析用辞書には BCCWJ コアデータ等から成る訓練コーパスから学習された MeCab のパラメータ値が含まれている。本実験では本コーパスを評価用コーパスとし、2 システムについて追加の学習を行わず、ブログ・質問サイトのテキストに対してどの程度の解析精度を達成できるかを評価した。

評価指標には、単語分割、品詞付与、正規化の Precision (適合率)、Recall (再現率)、F1 値を用いた。Precision はシステムの予測のうち正解と一致したものの割合であり、システムの予測の正確さを表す。Recall は正解のうちシステムが予測できたものの割合であり、システムの予測の網羅性を表す。F1 値は Precision と Recall の調和平均である。品詞付与の評価では、単語分割と品詞付与の両方に正解した場合にのみ正解とし、正規化の評価では、単語分割と正規化の両方に正解した場合にのみ正解とした。なお、本コーパスでは、一つの崩れ表記に対して一つまたは複数の正規表記を付与しているため、正解正規表記のいずれかに一致した場合に正解とした。

3.3　実験結果

本コーパスにおける 2 システムの精度を表 3 に示す。2 システムの単語分割及び品詞付与の解析精度は F1 値 90 ～ 95 % 程度であり、整った書き言葉と比べて解析が難しいと言える[*7]。MeCab+ER は、MeCab の単語分割・品詞付与精度から F1 値 2.5 ～ 2.9 ポイントの向上を達成し、正規化との同時解析が有効であることを示している。同システムの正規化精度は他の 2 タスクに比べて大幅に低く、特に Recall が低い点は、ユーザ生成テキストに出現する多様な異表記に対して、使用した正規化ルールの網羅性が十分でないことを示している。実際、表 4 に示すように、MeCab+ER が正規化に成功した事例を含むカテゴリは、音変化と代用表

*6　Sasano ら [11] のオリジナルの方法は、形態素解析器 JUMAN を拡張したシステムとして実装され、JUMAN 品詞体系に対応している。

*7　たとえば、Kudo ら [10] は新聞テキストに対する単語分割・品詞付与精度について F1 値 98 ～ 99 % と報告している。

表 2　BQNC の統計情報

媒体	文数	単語数 (延べ)	単語数 (異なり)	崩れ表記数 (延べ)	崩れ表記数 (異なり)
質問サイト	379	5,649	1,699	320	221
ブログ	550	6,951	2,231	447	257
全体	929	12,600	3,419	767	420

表 3　2 システムの単語分割、品詞付与、正規化精度

タスク	MeCab			MeCab+ER		
	P	R	F	P	R	F
単語分割	89.2	95.1	92.1	**93.5**	**96.5**	**95.0**
品詞付与	87.5	93.3	90.3	**91.4**	**94.3**	**92.8**
正規化	–	–	–	55.9	25.8	35.3

表 4　MeCab+ER のカテゴリ別の正規化精度 (Recall)

カテゴリ	件数	Recall
音変化	419	37.0
異文字種	248	0.0
代用表記	132	32.6
誤表記	23	0.0

表 5　2 システムのカテゴリ別の単語分割、品詞付与精度 (Recall)

カテゴリ	件数	MeCab		MeCab+ER	
		分割	品詞	分割	品詞
方言	23	91.3	78.3	**95.7**	**82.6**
固有名	103	87.4	84.5	**88.4**	**85.4**
オノマトペ	218	79.8	73.4	**87.2**	**77.1**
外国語	14	78.6	78.6	78.6	78.6
顔文字・アスキーアート	270	**73.7**	**64.1**	73.3	63.3
感動詞	174	64.9	**53.5**	**72.4**	48.9
スラング	37	67.6	67.6	67.6	67.6
音変化	419	50.6	47.5	**82.6**	**76.4**
異文字種	248	71.0	62.9	**78.2**	**69.4**
代用表記	132	65.2	54.6	**76.5**	**69.0**
誤表記	23	47.8	30.4	47.8	30.4
いずれかのカテゴリに該当	1,565	68.9	61.9	79.6	70.4
いずれのカテゴリにも非該当	11,035	98.9	97.7	98.9	97.7

記のみであった。

続いて、2 システムのカテゴリ別の単語分割・品詞付与精度 (Recall) を表 5 に示す。いずれのカテゴリにも該当しない一般的な単語に対しては、両システムともに約 98 ～ 99　% と高い精度を達成している一方、ユーザ生成テキストに特徴的である各カテゴリに対する精度はおおむね 60 ～ 80 % にとどまっている。MeCab+ER の精度は MeCab に比べて全般的に高い傾向があり、特にオノマトペ、異文字種、音変化について顕著に向上している。感動詞は、事例数が多いカテゴリでありながら品詞付与精度 50 % 前後と低く、最も認識が難しいカテゴリの一つであると示唆される。これは、笑い声、泣き声、叫び声などを模した様々な感情の表現が臨時的に創造され用いられるために、新表現の多様性が大きいカテゴリとなっているという点が要因と考えられる。

3.4　まとめ

本節では、著者らによる形態素解析・語彙正規化のためのユーザ生成テキストコーパス構築の研究について紹介した。本コーパスを用いた評価実験により、ユーザ生成テキストに特徴的な言語現象に対して従来システムの解析精度が低下することを示した。本コーパスは、従来及び将来のシステムの比較を可能にする公開ベンチマークデータとして機能すると期待できる。本研究のより詳細な内容は文献 [2] を参照されたい。

4　日本語語彙正規化のための疑似データ生成法とテキスト編集モデル

日本語テキストの語彙正規化タスクにおける課題として、モデルの学習に利用できるラベル付きデータがほとんどない点が挙げられる。著者らは、同課題に対処するための有望な方法として、対を成す崩れ表記と正規表記についての語彙知識を用いて、疑似ラベル付きデータを生成する方法を提案した。ただし、疑似的に生成したデータはノイズも含み、高品質なデータを大量に確保することは難しい。そのため、限られた量のデータから効率的に学習可能な方法として、テキスト編集に基づく方法を採用し、単語分割、品詞付与と語彙正規化を同時に解く方法を提案した。

以降、本節では、本研究におけるタスク定式化方法、

疑似ラベル付きデータの生成方法並びに両方法に基づく提案手法の評価について紹介する。

4.1　タスク定式化

単語分割、品詞付与、語彙正規化の同時解析タスク（Word Segmentation, POS Tagging, and Lexical Normalization: 以降 SPN と呼ぶ）は、文字 x_i の系列である入力文 $x=(x_1, ..., x_n)$ に対し、適切な単語境界、品詞及び正規表記を予測する問題である。たとえば、入力文“日本語まぢムズカシー”に対しては、“日本”、“語”、“まぢ”、“ムズカシー”という単語列に分割し、単語列に“名詞”、“名詞”、“副詞”、“形容詞”という品詞列を割り当て、“NONE”、“NONE”、“まじ”、“難しい”という正規表記を割り当てられれば適切な予測となる（“NONE”は割り当てるべき正規表記がないこと、つまり元の表記のままでよいことを意味する）。

本研究では、SPN タスクを、4 種類のタグ列を予測する問題として定式化した。単語分割タグ（Seg タグ）として、各入力文字に“B”（単語の先頭）、“I”（単語の内部）、“E”（単語の末尾）、“S”（単独で単語となる文字）のいずれかを割り当てる。品詞タグ（POS タグ）として、各入力文字に“名詞”、“動詞”など事前に決められた品詞のいずれかを割り当てる。正規化については、入力文字列を正規表記に編集するための 2 系統のタグを定義し、各入力文字に、文字列編集操作タグ（SEdit タグ）と文字種変換タグ（CConv タグ）をそれぞれ割り当てる。前述の入力文に対して予測すべきタグ列の例を図 3 示す。

このようなテキスト編集の方法を英語の語彙正規化に適用した研究 [13][14] もあるが、英語では文字の種類が英数字やアルファベットなど少数であるのに対し、日本語では漢字を含む数千種類の文字を扱う必要がある。そこで、2 系統のタグセット SEdit、CConv と、仮名漢字変換の機構を組み合わせることで、120 件程度のタグで正規化処理を実現可能とした。

具体的には、SEdit タグセット T_{sedit} を式 (1) で定めた。

$$T_{\text{sedit}} = \{\text{KEEP, DEL, INSL}(c), \text{INSR}(c), \text{REP}(c)\} \cdots\cdots (1)$$

各タグは記載した順に、「変更なし」、「該当文字を削除」、「該当文字の左隣に文字 c を挿入」、「該当文字の右隣に文字 c を挿入」、「該当文字を文字 c で置換」を意味する。また、CConv タグセット T_{cconv} を式 (2) で定めた。

$$T_{\text{cconv}} = \{\text{KEEP, TO\_HIRA, TO\_KATA, TO\_KANJI}\} \cdots\cdots (2)$$

各タグは記載した順に、「変更なし」、「該当文字をひらがなに変換」、「該当文字をカタカナに変換」、「該当文字を含む単語全体を漢字に変換」を意味する。“TO_KANJI”タグについては、このタグが付与されただけではどの漢字に変換すればよいか一意に決まらないが、外部の仮名漢字変換器を用いて最も可能性の高い変換候補に変換することで対処する。例として、崩れた表記の単語を正規表記に変換するためのタグ列を表 6 に示す。

なお、正規化を実現するための方法として、機械翻訳などの言語生成タスクの他、テキスト正規化の従来研究 [4] でも用いられている系列変換（Sequence-to-Sequence）[15] の方法を採用することもできる。系列変換では、任意の入力トークン列を任意の出力トークン列へ直接変換する処理を行うが、出力トークンの種類が数千～数万以上と多いため大規模な学習データ

x：	日	本	語	ま	ぢ	ム	ズ	カ	シ	ー
y^s：	B	E	S	B	E	B	I	I	I	E
y^p：	名詞	名詞	名詞	副詞	副詞	形容詞	形容詞	形容詞	形容詞	形容詞
y^e：	KEEP	KEEP	KEEP	KEEP	REP(じ)	KEEP	KEEP	KEEP	KEEP	REP(い)
y^c：	KEEP	KEEP	KEEP	KEEP	KEEP	HIRA	HIRA	HIRA	HIRA	KEEP
				⇒ まじ		⇒ むずかしい				

図 3　入力文 x に対する Seg タグ列 (y^s)，POS タグ列 (y^p)，SEdit タグ列 (y^e)，CConv タグ列 (y^c) の例

表 6　崩れ表記に対する SEdit タグ列及び CConv タグ列の例

崩れ表記	正規表記	SEdit タグ列	CConv タグ列
まぢ	まじ	K, REP（じ）	K, K
ムズカシー	むずかしい	K, K, K, K, REP（い）	HR, HR, HR, HR, K
すごーいー	すごい	K, K, DEL, K, DEL	K, K, K, K, K
さいこー	最高	K, K, K, REP（う）	KJ, KJ, KJ, KJ

“KEEP”, “TO_HIRA”, “TO_KANJI” タグをそれぞれ “K”, “HR”, “KJ” と略記した。

（数十万～数百万文）が必要となる点などが問題となり得る。テキスト編集の方法を採用する利点は、事前に決めた百種類程度のタグを予測できればよいことから比較的少量の学習データ（数万～数十万文）でのモデルの学習が可能と想定される点にある。

4.2　疑似ラベル付きデータの生成

本研究では、崩れ表記 v_s と正規表記 v_t の異表記対集合 V を用いて、形態素解析済みテキストから疑似ラベル付きデータを生成する二つの方法、DS-T と DS-S を提案した*8。異表記ペア集合 V の具体的な構築方法は後述する。生成する疑似ラベル付きデータは、崩れた表記の語を含む文（Source Sentence）$\boldsymbol{x}_{\mathrm{src}}$ と標準的な表記の語から成る整った文（Target Sentence）$\boldsymbol{x}_{\mathrm{tgt}}$ の対とみなせる。

DS-T では、図 4 のように、V 中の崩れ表記 v_s を含む入力文について、v_s をその正規表記 v_t で置き換えることにより、実在の崩れた文から人工的な整った文を生成する（＝ターゲット文が人工データ）。DS-S では、図 5 のように、V 中の正規表記 v_t を含む入力文について、v_t をその崩れ表記 v_s で置き換えることにより、実在の整った文から人工的な崩れた文を生成する（＝ソース文が人工データ）。各方法で生成されたソース文とターゲット文の対は、ソース文と、ソース文をターゲット文に変換するための SEdit タグ列及び CConv タグ列の形式に自動変換された上で、（さらに Seg タグ列と POS タグ列を付加し）モデル学習の入力に用いられる。

異表記対集合の構築には、辞書ベースの方法と、ルールベースの方法の 2 種類を用いた。辞書ベース異表記抽出法では、形態素解析用辞書 UniDic を用いて、同一の語彙素（例：「大きい」）として登録されている表記の集合（例：「大きい」、「おおきい」、「おっきい」）を取得し、コーパス出現頻度や読み情報を基に崩れ表記と正規表記を決定し、異表記対を構成した。ルールベース異表記抽出法では、辞書ベース法で得た正規表記（例：「大きい」）に人手定義ルールを適用して崩れ表記（例：「大きーい」）を生成し、崩れ表記、正規表記ともコーパス中の（文字 n-gram としての）出現頻度が閾値 10 以上のものを有効な異表記対と認定した。なお、人手定義ルールとしては、Sasano ら [11] 及び Ikeda ら [4] が定義したルールに追加のルールを加えた大分類 10 種類のルールを用いた。

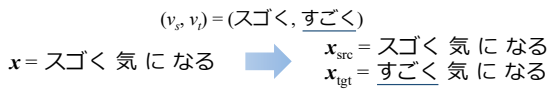

図 4　DS-T による疑似データ生成の例

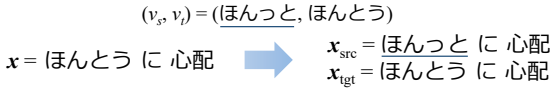

図 5　DS-S による疑似データ生成の例

4.3　実験設定

4.3.1　システム

3 の実験と同様に、従来手法として、日本語形態素解析システム MeCab と、Sasano らの手法を実装したシステム MeCab+ER を用いた。

提案手法では、系列タグ付けタスクで用いられる一般的なニューラルネットワーク構造の一つである BiLSTM（Bidirectional Long Short-Term Memory）[16][17] に、タグ種別（Seg, POS, SEdit, CConv）ごとの Softmax 推論層を追加したモデルを用いた。入力文は、文字ごとの特徴量ベクトルの系列に変換され、BiLSTM での計算を経て各タグの確率分布ベクトルの系列に変換され、文字ごとに確率値が最大のタグが出力される。特徴量として、文字及び発音の分散表現ベクトルと、辞書マッチングに基づく数種類の 2 値ベクトルを連結して用いた。仮名漢字変換には、n-gram 言語モデルに基づく仮名漢字変換器を実装して用いた。

4.3.2　実験データ

BCCWJ コアデータを訓練データ D_t（5.7 万文）と開発用データ D_v（0.3 万文）に分割し、提案手法の単語分割と品詞付与タスクの学習に用いた。テストデータとして、**3** で述べた BQNC を用いた。

辞書由来異表記対集合 V_d として、UniDic から抽出された異表記対候補の BCCWJ 非コアデータ D_u（350 万文）での出現頻度を計測し、頻度上位 20 万対を採用した。ルール由来異表記対集合 V_r として、V_d の正規表記とルールを適用して得られた崩れ表記候補について、Yahoo! 知恵袋データ（880 万文）での n-gram 頻度を計測し、頻度上位 20 万件を採用した。

正規化タスクのための疑似ラベル付きデータとして、D_t, D_u と V_r, V_d を基に 3 種類のデータを生成した。一つ目は、辞書由来異表記対集合 V_d を用いて D_t に DS-T を適用して得られたデータ A_t（5.7 万文）である。二つ目と三つ目は、辞書、ルール由来異表記対集合 V_d, V_r をそれぞれ用いて D_u に DS-S を適用して得られたデータ A_d（17.3 万文）及び A_r（17 万文）である。3 種類の疑似ラベル付きデータの一つまたは複数を用いて提案手法の正規化タスクの学習を行った。

*8　DS は Distant Supervision（遠距離教師あり学習）を意味する。

4.4　実験結果

表 7 に 3 システムの単語分割精度（F1 値）、品詞付与精度（F1 値）、正規化精度（Precision, Recall, F1 値）を示す。提案手法は、三つのデータ A_t, A_r, A_d をすべて用いた場合に最良の F1 値 42.4 を達成しており、異なる種類の疑似ラベル付きデータを用いることが有効であった。なお、提案手法について、2 種類の後処理ルール SegPP 及び NormPP[*9] を適用することでさらに精度が向上し、特に後者のルールにより Precision が 15 ポイント向上した。これは、簡易な後処理ルールが有効である一方で、モデルが不適切な正規表記を多く生成しており、改善の余地があることを示している。他システムと比較すると、提案手法は正規化精度（Recall, F1 値）で MeCab+ER を上回り、より多くの崩れ表記を正規化できていることがわかる。対して、単語分割と品詞付与では MeCab+ER が最良であり、この点については、同システムが崩れ表記の範囲を明示的に考慮していることが他の 2 タスクに有効であった可能性がある。提案手法でも同様の工夫を行うことで単語分割、品詞付与精度を改善できる余地がある。

続いて、提案手法（表 7 の（iii）のモデル）と MeCab+ER のカテゴリ別の正規化精度（Recall）を表 8 に示す。提案手法はいずれのカテゴリでも MeCab+ER の精度を上回り、音変化、異文字種、代用表記について同程度の認識精度を達成している。誤表記の認識精度が低いのは、疑似ラベル付きデータに同カテゴリの事例がほとんど含まれていなかったためと考えられる。

なお、三つの疑似ラベル付きデータ A_t, A_r, A_d を訓練データとした場合に、テストデータ中の崩れ表記トークン 767 件のうち、訓練データに出現したトークンの割合は 63 % であった。これは、一度でも訓練データに出現した崩れ表記を正規化できる理想的なシステムの Recall に相当すると言え、提案手法（表 7 の（iv）のモデル）が達成した約 38 % から開きがある。また、訓練データに出現しなかった 100−63 =37 % の未知トークンは、今回生成した疑似ラベル付きデータでカバーできなかった事例である。したがって、疑似ラベル付きデータの網羅性、モデルの学習能力の両方において改善の余地がある。

4.5　まとめ

本節では、著者らが提案した疑似ラベル付きデータの生成法と、テキスト編集に基づく単語分割、品詞付与、語彙正規化の同時解析法について紹介した。語彙正規化では、異なる種類の疑似ラベル付きデータを組み合わせて学習に用いることで崩れ表記の認識精度が向上し、従来法以上の精度を達成できることを示した。本研究のより詳細な内容は文献 [3] を参照されたい。

5　おわりに

本稿では、日本語テキストの語彙正規化の問題について解説し、著者らが取り組んできた評価用コーパス構築の研究と、単語分割、品詞付与及び語彙正規化の同時解析システムの研究を紹介した。今後の展望を以下に述べる。

構築した評価用コーパスは、システムの性能評価に有用であるものの、現時点では 2 ジャンル（質問サイト、ブログ）で合計 929 文を収録した小規模なデータである。ユーザ生成テキストの様々なジャンルについてより精緻（せいち）な評価を可能とするため、他ジャンルのテキストについてもアノテーションを行い、合計 5 ジャンル以上、1 万文以上となるようにコーパスを拡大することを考えている。

また、著者らが提案した語彙正規化システムは、従来手法以上の精度を達成したものの、現状の精度には

*9　SegPP は、母音または特殊モーラの仮名文字（「ー」「っ」「ん」）が連続する場合にそれらの Seg ラベルを “I” に修正するルールである。NormPP は、予測された正規表記が所定の正規表記辞書（実験では V_d の正規表記を使用）に含まれない場合に正規表記の予測を取り下げる（元の表記そのままとする）ルールである。

表 7　3 システムの単語分割、品詞付与、正規化精度

システム	疑似データ	分割	品詞	正規化		
		F	F	P	R	F
MeCab		92.1	90.3	–	–	–
MeCab+ER		**95.0**	**92.8**	55.9	25.8	35.3
提案手法	（i） A_t	92.6	88.8	50.9	19.4	28.1
	（ii） A_t, A_r	92.3	90.1	42.4	28.0	33.8
	（iii） A_t, A_r, A_d	92.5	89.6	49.7	37.0	42.4
提案手法 +SegPP	（iv） A_t, A_r, A_d	93.5	90.5	50.8	**37.8**	43.4
提案手法 +NormPP	（v） A_t, A_r, A_d	93.5	90.5	**65.8**	36.6	**47.1**

表 8　2 システムのカテゴリ別の正規化精度（Recall）

カテゴリ	件数	MeCab+ER	提案手法
音変化	419	37.0	**37.2**
異文字種	248	0.0	**37.1**
代用表記	132	32.6	**38.6**
誤表記	23	0.0	**4.4**

予測の正確さの点でも網羅性の点でも改善の余地がある。可能な限り予測誤りを減らすとともに網羅性も更に高めることで、実用的な精度の正規化システムを実現するとともに、入力テキストの正規化処理が、機械翻訳を始めとする応用タスクの精度向上に寄与することを実証したいと考えている。

【参考文献】

1 B. Marie and A. Fujita, “Synthesizing Parallel Data of User-Generated Texts with Zero-Shot Neural Machine Translation,” Transactions of the Association for Computational Linguistics, MIT Press, vol.8, pp.710–725, 2020.
2 S. Higashiyama, M. Utiyama, T. Watanabe, and E. Sumita, “User-Generated Text Corpus for Evaluating Japanese Morphological Analysis and Lexical Normalization,” Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp.5532–5541, June 2021.
3 S. Higashiyama, M. Utiyama, T. Watanabe, and E. Sumita, “A Text Editing Approach to Joint Japanese Word Segmentation, POS Tagging, and Lexical Normalization,” Proceedings of the 7th Workshop on Noisy User-generated Text, pp.67–80, Nov. 2021.
4 T. Ikeda, H. Shindo, and Y. Matsumoto, “Japanese Text Normalization with Encoder-Decoder Model,” Proceedings of the 2nd Workshop on Noisy User-generated Text, pp.129–137, Dec. 2016.
5 I. Saito, K. Sadamitsu, H. Asano, and Y. Matsuo, “Morphological Analysis for Japanese Noisy Text Based on Character-level and Word-level Normalization,” Proceedings of the 25th International Conference on Computational Linguistics: Technical Papers, pp.1773–1782, Aug. 2014.
6 N. Kaji and M. Kitsuregawa, “Accurate word segmentation and pos tagging for Japanese microblogs: Corpus annotation and joint modeling with lexical normalization,” Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, pp99–109, Oct. 2014.
7 K. Maekawa, M. Yamazaki, T. Ogiso, T. Maruyama, H. Ogura, W. Kashino, H. Koiso, M. Yamaguchi, M. Tanaka, and Y. Den, “Balanced corpus of contemporary written Japanese,” Language Resources and Evaluation, vol.48, pp.345–371, 2014.
8 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，vol.3，no.3，pp.68–77，Sept. 2020.
9 鍛治 伸裕，森 信介，高橋 文彦，笹田 鉄朗，斉藤 いつみ，服部 圭悟，村脇 有吾，内海 慶，“形態素解析のエラー分析，” 言語処理学会第 21 回年次大会ワークショップ「自然言語処理におけるエラー分析 (兼：Project Next NLP 報告会)」，March 2013.
10 T. Kudo, K. Yamamoto, and Y. Matsumoto, “Applying Conditional Random Fields to Japanese Morphological Analysis,” Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing, pp.230–237, July 2004.
11 R. Sasano, S. Kurohashi, and M. Okumura, “A Simple Approach to Unknown Word Processing in Japanese Morphological Analysis,” Proceedings of the 6th International Joint Conference on Natural Language Processing, pp.162–170, Oct. 2013.
12 伝 康晴，“多様な目的に適した形態素解析システム用電子化辞書，” 人工知能，vol.24，no.5，pp.640–646，Sept. 2009.
13 G. Chrupała. “Normalizing tweets with edit scripts and recurrent neural embeddings,” Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp.680–686, 2014.
14 W. Min and B. Mott, “NCSU_SAS_WOOKHEE: A deep contextual long-short term memory model for text normalization,” Proceedings of the Workshop on Noisy User-generated Text, pp.111–119, 2015.
15 I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to sequence learning with neural networks”, Advances in Neural Information Processing Systems, vol.27, Dec. 2014.
16 S. Hochreiter and J. Schmidhuber. “Long short-term memory,” Neural Computation, vol.9, no.8, pp.1735–1780, 1997.
17 Z. Huang, W. Xu, and K. Yu. “Bidirectional LSTM-CRF models for sequence tagging,” Computing Research Repository, arXiv:1508.01991, 2015.

東山 翔平 （ひがしやま しょうへい）

ユニバーサルコミュニケーション研究所
先進的音声翻訳研究開発推進センター
先進的翻訳技術研究室
研究員
博士 (工学)
自然言語処理

【受賞歴】

2021 年　The 7 th Workshop on Noisy User-generated Text, Best Paper Award
2021 年　言語処理学会 2020 年度論文賞
2014 年　2014 International Conference on Computer & Information Sciences, Best Paper Award