仮名漢字変換ログを用いた単語分割の精度向上

高橋 文彦* 森 信介†
* 京都大学大学院情報学研究科 † 京都大学学術情報メディアセンター

1 はじめに

日本語や中国語のような膠着語を扱う自然言語処理の多くは，まずテキストを単語に分割することから始まる．これは，英語のような明示的な単語境界がある言語処理と異なり，単語分割の誤り分が言語処理の精度を割り引くこととなる．単語分割は，日本語なら 98%，中国語なら 95%以上の精度で実行できることが知られているが [1, 2]，言語処理の対象となるドメインが，単語分割の学習コーパスを利用できるドメインと異なることが多々あり，この際，単語分割の精度を低下させ，応用の言語処理に大きく影響する．例えば，特許の機械翻訳，医学書のテキストマイニング，Twitter[1]のツイートを用いたマーケティングなどが挙げられる．これらの研究では，単語分割精度や，単語分割と品詞推定のジョイントタスクである形態素解析の精度が低いことが報告されている [3, 4, 5]．

本研究ではこのような問題に対処するために，仮名漢字変換を利用する過程から，単語分割に利用できる情報を獲得する方法を提案する．日本語のインプットメソッドは仮名漢字変換を用いて入力されるが，文章を作成するときの変換の履歴には，単語境界の情報が含まれる．本研究では，この変換の履歴を仮名漢字変換ログと呼び，仮名漢字変換ログを利用して単語分割を行う．

本研究は，人の自然な行動から有用な情報を獲得する枠組みの一つである．このような研究の他の例として，Wikipedia などの HTML タグのある文章に対してそのタグを単語境界とみなして，単語分割器を学習する方法が提案されている [2, 6, 7]．本論文で利用する仮名漢字変換ログも，人が意図して作成した言語資源ではないという点で，これらの研究と類似している．しかしながら，仮名漢字変換ログはノイズや断片化等の問題があり，その利用方法は自明ではない．本論文では，適切なログの利用方法を提示し，単語分割に有用であることを示す．

2 未知語も提示する仮名漢字変換

この章では，生コーパスの部分文字列も提示する仮名漢字変換について説明する．

2.1 擬似確率的タグ付与コーパス

擬似確率的タグ付与により，一般的な自動タグ付与では出現しない未知語がコーパスに出現する．次の例を用いて説明する．

擬似確率的タグ付与コーパスの例

艦/かん | これ/これ | や/や |っ/っ| た/た
艦これ/かんこれ | 面白/おもしろ | い/い |？/？

１行目の文では「艦これ」を「艦」と「これ」に分割しているが，２行目の文では「艦これ」を１単語としている．これは「艦」と「こ」の間が確率的に分割され，単語境界有無の揺れが生じた結果である．この例では「艦これ」という未知語候補がコーパスに現れるが，インプットメソッド利用者が「艦これ」を変換候補から選択することで，未知語が読みとともにログに記録される．

擬似確率的タグ付与コーパスでは誤った単語も生じる．しかし誤った単語はユーザーに選択されにくくログに記録されにくいため，擬似確率的タグ付与コーパスは未知語の再現率が重要であることに留意したい．

擬似確率的タグ付与コーパスを作成するために，アノテーション情報のないテキストから文献 [8] の方法を用いて単語境界と読み情報を付与した．

2.2 言語モデル

仮名漢字変換サーバーとして，文献 [8] の単語と読みの組を単位とする言語モデルを用いる．確率的モデルによる仮名漢字変換 [9] は，キーボードから直接入力可能な入力記号 $\mathcal{Y}$ の正閉包 $\boldsymbol{y} \in \mathcal{Y}^+$ を入力として，日本語の語彙の正閉包 $\boldsymbol{w} \in \mathcal{W}^+$ を変換結果として出力する．この際，以下の式が示すように，単語 w を入力記号列 $\boldsymbol{y}$ の組 $u = \langle w, \boldsymbol{y} \rangle$ を単位とする言語モデルによる生成確率を評価基準とする．

$$\begin{aligned} \operatorname*{argmax}_w P(\boldsymbol{w}|\boldsymbol{y}) &= \operatorname*{argmax}_w \frac{P(\boldsymbol{w},\boldsymbol{y})}{P(\boldsymbol{y})} \\ &= \operatorname*{argmax}_w P(\boldsymbol{u}) \end{aligned}$$

ここで単語列 $\boldsymbol{w}$ は表記文字である．$P(\boldsymbol{u})$ は，u を単位とする n-gram モデルを用いて，以下のようにモデル化される．

$$P(\boldsymbol{u}) = \prod_{i=1}^{h} P(u_i|\boldsymbol{u}_{i-n+1}^{i-1})$$

*takahasi@ar.media.kyoto-u.ac.jp
†forest@i.kyoto-u.ac.jp
[1] https://twitter.com

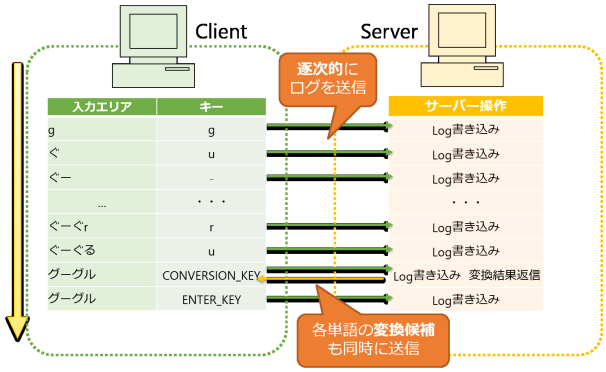

図 1: 変換ログを収集するインプットメソッド

$$P(u_i|\boldsymbol{u}_{i-n+1}^{i-1}) = \begin{cases} P(u_i|\boldsymbol{u}_{i-n+1}^{i-1}) & (u_i \in \mathcal{U}) \\ P(\texttt{UU}|\boldsymbol{u}_{i-n+1}^{i-1}) M_{y,n}(\boldsymbol{y}_i) & (u_i \notin \mathcal{U}) \end{cases}$$

ここで $\mathcal{U}$ は言語モデルの語彙 (単語と入力記号列の組の集合) を表す．この式の中の u_i $(i \leq 0)$ と u_{h+1} は，文頭と文末に対応する記号 BT であり， UU は未知の組を表す記号である．また，$\boldsymbol{y}_i = y(u_i)$ は $u_i = \langle w_i, \boldsymbol{y}_i \rangle$ の入力記号列である．

式 (1) の $P(u_i|\boldsymbol{u}_{i-n+1}^{i-1})$ と $P(\texttt{UU}|\boldsymbol{u}_{i-n+1}^{i-1})$ は，語彙に BT と UU を加えた $\mathcal{U} \cup \{\texttt{BT}, \texttt{UU}\}$ 上の n-gram モデルである．パラメータは，単語に分割されかつ入力記号列が付与されたコーパスから以下の式を用いて最尤推定する．

$$P(u_i|\boldsymbol{u}_{i-n+1}^{i-1}) = \frac{N(\boldsymbol{u}_{i-n+1}^{i})}{N(\boldsymbol{u}_{i-n+1}^{i-1})}$$

ここで，$N(\boldsymbol{u})$ はコーパス中の表記と読みの組列 $\boldsymbol{u}$ の出現回数を表す．また，$M_{y,n}(\boldsymbol{y}_i)$ は入力記号列の n-gram モデルによる未知語モデルである．

3 仮名漢字変換ログの収集

本章では仮名漢字変換ログを収集するインプットメソッドと，収集した変換ログの特性について説明する．

3.1 変換ログを収集するインプット メソッド

仮名漢字変換ログを収集するために，サーバーサイドで仮名漢字変換を行うインプットメソッド KAGAMI[2] を作成した．クライアントとサーバーの動作の様子を図 1 に示す．

インプットメソッドを使う過程は入力過程，変換過程，確定過程の 3 つに分けられる．入力過程はキーボード操作により入力文字列が入力される過程である．この過程における入力文字列が文の読み情報となる．変換過程は入力文字列が表記文字列へ変換される過程であり，変換結果から他の変換候補を選択する操作を含む．この変換過程で文に単語境界情報が付与される．確定過程は表記文字列を決定する過程である．入力過程，変換過程，確定過程の順に進み入力が完了する．ただし，表記文字列が平仮名のみで構成される場合に多いが，変換過程はスキップできる．

クライアントは，各過程のログと共にその時間と IP アドレスを逐次的にサーバーに送信する．サーバーは，仮名漢字変換と，クライアントから受け取った変換ログをログファイルへ書き出しを行う．

3.2 仮名漢字変換ログ

1 つの仮名漢字変換ログは，確定結果 1 つに対する入力過程，変換過程，確定過程のログで構成される．変換ログを収集するインプットメソッド によって得られた変換ログの一部を，確定した時間（確定時間）と確定結果，変換過程の有無と共に表 1 に示す．

変換ログの主要な情報は確定過程における確定結果である．多くの場合，確定結果の単位は完全な文ではなく文断片である（細分化）．さらに，誤まって確定した結果などを含む（ノイズ）．したがって，確定結果はノイズありの単語分割済みかつ読み付与済みの文断片からなるコーパスと見なすことができる．本研究では，このような変換ログを，自動単語分割器から参照する．

4 仮名漢字変換ログによる単語分割

本章では，仮名漢字変換ログを学習コーパスとする方法と，その学習コーパスを利用するために部分的アノテーションから学習できる推定器について説明する．

4.1 仮名漢字変換ログの利用

ノイズありの単語分割済みかつ読み付与済みの文断片からなるコーパスである変換ログを学習データに利用するために，文献 [10] では 3 つの方法が提案されている．本論文では，これに加えて変換操作の多いログを利用する方法を提案する．

AS-IS-log: 確定結果は単語境界情報が付与された部分的アノテーションコーパスと見なすことが出来る．このため，確定結果をそのままコーパスとして利用する．

CHUNK-log: 細分化の問題を回避するために，確定結果の時間を参照して連結する方法を提案する．変換ログの確定時間と次の変換ログの入力過程のログの開始時間の差が s 以下の場合，この確定結果を連結する．本論文では，$s = 0.5$[s] とする．

ALIGN-log: 作成されたツイートに変換ログをアライメントし，単語分割位置と読みの情報を付与する．この方法により，ノイズと細分化の問題を回避できると考えられる．ノイズの変換ログはアライメントされないため学習データから除外され，文として完成しているツイートにアライメントするため細分化の問題を回避出来る．

MCON-log: 本論文では，変換操作の回数でフィルタリングしたログを学習コーパスとする方法を提案する．ユーザーが編集した文は，ノイズを含む可能性が

[2] http://plata.ar.media.kyoto-u.ac.jp/takahasi/kagami/

表 1: ‘それに比べると安めかと’というツイートの仮名漢字変換ログ

時間	確定結果	変換過程有無	備考
18:37:11.21	そ _れ _に	無	変換していない確定結果
18:37:12.60	くらっ/くらっ| ベル/べる	有	誤って確定した結果
18:37:14.94	比べ/くらべ | る/る	有	修正の入力
18:37:15.32	と	無	
18:37:19.82	も _の _の	無	完成したツイートには残らなかった確定結果
18:37:22.42	安め/やすめ | か/か | と/と	有	

低く，未知語を含む可能性が高い，と考えられる．このため，変換過程における変換の操作が n 回以上のログのみを学習コーパスとして使用する．実験では $n = 2$ とする．

CHUNK-MCON-log: MCON-log に対して，CHUNK-log と同様に，時間を参照して確定結果を連結する．

4.2 ログを利用する単語分割

ログは文断片かつ単語境界情報が部分的にアノテーションされているコーパスと見なせるので，このようなコーパスから学習できる自動単語分割器を用いる．このような単語分割器として，点予測による単語分割を採用した．点予測とは，分類器の素性として，周囲の単語境界の推定値を利用せずに，周囲の文字列の情報のみを利用する方法である．点予測による単語分割では，文字 n-gram，文字種 n-gram，単語辞書素性の３種類の素性を参照する線形サポートベクトルマシン [11] による分類を行う．この機能があるテキスト解析器として KyTea[1] を用いる．窓幅は 3 とした．

5 評価実験

4.1 項で述べた仮名漢字変換ログを利用するの手法を実験的に評価する．

5.1 仮名漢字変換システム

アノテーションのないテキストから，2.1 項で説明した擬似確率的タグ付与コーパスを作成した．アノテーションのないテキストとして，ツイートと現代日本語書き言葉均衡コーパス (BCCWJ)[12] の NonCore データを用いた．ツイートは，13,467,927 件のツイートを収集し，786,331 文を得た．BCCWJ の NonCore データは 358,078 文を用いた．これらの２つのテキストを合わせた 1,207,182 文に対して確率的タグ付与を行う．単語境界確率と読み確率を計算するために，KyTea[1] を用いた．文献 [10] と同様の学習データを用いて，ロジスティック回帰 [11] を用いたモデルを学習した．この単語分割・読み推定器を用いて，ツイートと BCCWJ の NonCore データの単語境界確率，読み確率を計算し，擬似確率的タグ付与コーパスを作成した．このコーパスを用いて 2 節の未知語も提示する仮名漢字変換システムを作成した．

表 2: 実験で用いる言語資源

学習コーパス		
記号	文数	単語数
BCCWJ-train	56,753	1,324,951
TWI-train	2,354	29,460
記号	エントリー数	単語境界情報
AS-IS-log	32,119	39,708
CHUNK-log	6,572	63,144
ALIGN-log	1,850	43,820
MCON-log	4,610	10,852
CHUNK-MCON-log	1,218	14,242
テストコーパス		
記号	文数	単語数
TWI-test	588	7,498

この仮名漢字変換システムを公開し配布した．2014/04/13-2014/12/31 の間に収集した変換ログを実験に使用した．なお，ユーザーは最大 17 名であった．

5.2 テストデータ

2014/05/19-2014/05/22, 2014/06/02-2014/06/04 に収集した 2,659,168 件のツイートから無作為に 1,592 件のツイートを選択し，人手でアノテーションを行った．アノテーション基準は BCCWJ の短単位に準拠し，これに加えて活用語尾を分割する．これらのツイートから，2,976 文を得た．これを 8 : 2 に分け，それぞれを TWI-train, TWI-test とした．

5.3 実験の設定

実験で用いるコーパスを表 2 に示す．BCCWJ-train は BCCWJ の学習セット，TWI-train，TWI-test は人手でアノテーションしたツイートの本文である．

BCCWJ-train のみを学習データとしたモデル，AS-IS-log, CHUNK-log, ALIGN-log, MCON-log. CHUNK-MCON-log をそれぞれ BCCWJ-train に追加し学習データとしたモデルの６つのモデルで単語分割を行う．

5.4 ログ利用手法の比較

解析結果と正解データを単語単位でアライメントを取り，再現率，適合率，その調和平均 (F 値) で評

表 3: ツイートに対する単語分割精度

	再現率	適合率	F 値
BCCWJ-train	90.31	94.05	92.14
+ AS-IS-log	90.33	93.77	92.02
+ CHUNK-log	91.04	94.29	92.64
+ ALIGN-log	90.71	94.13	92.39
+ MCON-log	90.62	94.09	92.32
+ CHUNK-MCON-log	91.40	94.45	92.90

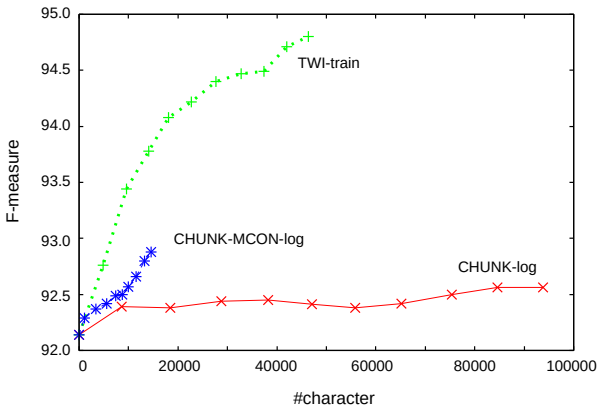

図 2: ログの量に応じた単語分割精度

価した．単語分割精度を表 3 に示す．AS-IS-log は，BCCWJ-train に比べて精度低下が見られた．これは確定結果に含まれるノイズによる影響だと考えられる．これに対して，MCON-log は精度の向上が見られ，確定結果に含まれるノイズが変換操作回数によりフィルタリングされたことを示す．さらに，MCON-log を連結することによって得られる CHUNK-MCON-log において，有意 ($p = 0.01$) に精度が向上した．

5.5 学習コーパスの追加との比較

次に，TWI-train，CHUNK-log，CHUNK-MCON-log について，学習コーパス (ログ) の量と単語分割精度のグラフを図 2 に示す．CHUNK-MCON-log を CHUNK-log と比較すると，同程度の文字量であってもより精度が高いことがわかる．また，今回獲得したログの量程度であっても，800 文程の人手によるアノテーションと同等の精度向上を実現した．仮名漢字変換ログは，アノテーションの訓練を受けていないユーザーが，単語境界情報の付与を意図しない自然なインプットメソッドの使用から獲得できる情報である．グラフ上の CHUNK-MCON-log の精度の推移が向上し続ける傾向にあることからも，より多くの仮名漢字変換ログを集めることで，さらなる精度向上が期待できる．

6 おわりに

本論文は，あらゆる膠着語の言語処理で課題となる単語分割の精度向上を目的として，仮名漢字変換を利用する過程から単語分割に利用できる情報を獲得する方法を提案した．実験により，変換操作の多い仮名漢字変換の確定結果を収集することで自動的に単語分割の精度を向上できることが明らかになった．本手法を用いれば，ドメインに合わせた仮名漢字変換システムを作りユーザーに提供することで，自動的に対象ドメインの単語分割精度が向上することが期待できる．

参考文献

[1] Graham Neubig, Yosuke Nakata, and Shinsuke Mori. Pointwise prediction for robust, adaptable Japanese morphological analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pp. 529–533, 2011.

[2] Fan Yang and Paul Vozila. Semi-supervised Chinese word segmentation using partial-label learning with conditional random fields. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pp. 90–98, 2014.

[3] Shinsuke Mori and Graham Neubig. Language resource addition: Dictionary or corpus? In *Proceedings of the Nineth International Conference on Language Resources and Evaluation*, pp. 1631–1636, 2014.

[4] 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 the 2014 Conference on Empirical Methods in Natural Language Processing*, pp. 99–109, 2014.

[5] Yijia Liu, Yue Zhang, Wangxiang Che, Ting Liu, and Fan Wu. Domain adaptation for crf-based Chinese word segmentation using free annotations. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pp. 864–874, 2014.

[6] Yuta Tsuboi, Hisashi Kashima, Shinsuke Mori, Hiroki Oda, and Yuji Matsumoto. Training conditional random fields using incomplete annotations. In *Proceedings of the 22nd International Conference on Computational Linguistics*, 2008.

[7] Wenbin Jiang, Meng Sun, Yajuan Lu, Yating Yang, and Qun Liu. Discriminative learning with natural annotations: Word segmentation as a case study. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pp. 761–769, 2013.

[8] 森信介, 笹田鉄郎, Neubig Graham. 確率的タグ付与コーパスからの言語モデル構築. 自然言語処理, Vol. 18, No. 2, 2011.

[9] Shinsuke Mori, Daisuke Takuma, and Gakuto Kurata. Phoneme-to-text transcription system with an infinite vocabulary. In *Proceedings of the 21st International Conference on Computational Linguistics*, 2006.

[10] 高橋文彦, 森信介. 仮名漢字変換ログを用いた単語分割読み推定の精度向上. 情報処理学会研究報告, 第 NL219 巻, 2014.

[11] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, Vol. 9, pp. 1871–1874, 2008.

[12] Kikuo Maekawa. Balanced corpus of contemporary written Japanese. In *Proceedings of the 6th Workshop on Asian Language Resources*, pp. 101–102, 2008.