全文字 n-gram を考慮した Dilated CNN を用いた単語分割

山口修平　　三輪 誠　　佐々木 裕
豊田工業大学
{sd17440, makoto-miwa, yutaka.sasaki}@toyota-ti.ac.jp

1 はじめに

自然言語処理では多くの場合文を単語の系列として扱うため，日本語のような単語ごとに分割されていない言語においては単語分割を最初に行うことが多い．単語の分割を誤ると後に行われる処理に悪い影響を与えるため，単語分割を正確に行うことが重要である．

単語分割はこれまでに多くの研究が行われている．日本語単語分割における伝統的な手法は，ラティス構造と呼ばれる辞書を用いて単語分割の候補を表現したグラフを構築し，ラティス構造上の経路を予測することで単語の境界を決定する手法 [3, 2] である．この辞書は，単語と品詞や読み，活用形などをまとめたものである．このようなラティス構造を用いた手法では，ラティス構造が正しく作成できれば，ラティス構造中の単語情報を利用することにより，高い精度で単語分割が可能である．しかしながら，辞書に登録されていない単語（未知語）を含む文では，正しいラティス構造を作成することが難しく，単語分割の精度が低下する．正しいラティス構造を作成するには，新しい単語が出現するたびに辞書を更新する必要があるため，コストがかかる．

このような辞書からラティス構造への依存を回避する手法として，点予測による単語分割手法 [5] がある．点予測とは，単語の境界決定を他の境界決定に依存せず，独立に予測する手法のことである．単語分割においては，文中の文字ごとに両端の単語境界の有無を予測することで単語境界を決定する．その際，文中の文字列の並びや文字種を判断材料にし，予測過程で生じる単語情報を用いない．単語の境界決定が独立なため，部分的にアノテーションされた文から学習を行うことができる特徴がある．

近年，ラティス構造を用いる単語分割手法と点予測による単語分割手法をニューラルネットワークを用いて拡張したニューラル単語分割 [4, 6] が提案されている．ニューラル単語分割は，従来手法と比較し，高い競争力を持つことが示されている．

本研究では，単語分割の精度向上を目的として，辞書に依存しない新たなニューラル単語分割手法を提案する．この手法は全ての文字 n-gram の単語候補を考慮し，単語単位で境界決定を行うため，各境界決定に決定済みの単語情報を利用でき，文字ベースの手法に比べて判断の回数が少ない．また，Dilated CNN を用いてこれまでに決定した単語情報を考慮することで，少ない計算量で CNN より広範囲の文脈を考慮する．

2 関連研究

辞書に依存しない単語分割手法として有名なものに京都テキスト解析ツールキット（KyTea）[5] がある．KyTea は点予測を用いて単語分割，品詞推定などを行う解析器である．KyTea では，単語分割された文を教師として文字ごとに単語境界の有無を学習することで単語境界を予測する．i 番目の文字と $i+1$ 番目の文字間の境界予測は以下の式で表される．

$$y_i = \mathrm{sign}(\boldsymbol{w} \cdot \phi(\boldsymbol{x}, i) + b) \quad (1)$$

$$= \mathrm{sign}(\sum_{k}^{N} w_k \cdot \phi_k(\boldsymbol{x}, i) + b) \quad (2)$$

$\phi_k(\boldsymbol{x}, i)$ は入力文 $\boldsymbol{x}$ 中の i 番目の文字における k 番目の素性である．$\boldsymbol{w}$，b はそれぞれ重みベクトルとバイアス項を表す．各文字の境界予測に用いる素性は文字や文字列，文字種など，文字ごとに独立な情報である．

3 提案手法

本論文では，辞書に依存せず単語単位で単語の境界を決定するニューラル単語分割を提案する．3.1 節でラティス構造を用いずに単語単位で単語の境界を決定する手法の概要を説明する．次に，3.2 節において，提案した単語分割手法を利用するためのニューラルネッ

トワーク構造について述べ，最後に 3.3 節で，ネットワークの学習手法ついて説明する．

3.1 単語分割の決定手法

本提案では，文の左から順に単語単位で分割を決定することによって単語分割を行う．より具体的には，単語分割の終わっていない文の部分文字列の先頭から，つまり左から，単語の分割候補を生成し，その候補から単語を選択することで単語分割を行う．分割候補には，単語分割されていない文字列の先頭文字が先頭となる全ての部分文字列を用い，その先頭文字からどの文字までが 1 つの単語であるかを決定する．つまり，分割候補はこの分割されていない文字列の文字数と同じになる．この決定手法を繰り返すことにより，本提案では単語単位で単語分割を行う．この単語単位の分割は，文字ベースの手法に比べて，必要な決定が少なくて済むという利点がある．

実験では，利用されない非常に長い部分文字列が生成されるのを避けるため，辞書や学習データ中をもとに，候補として利用する単語の文字数に制限をかけているが，この分割候補の長さの制約の決定方法と長さの制約を超える単語の扱いは今後の課題である．

この手法は，単語を分割候補から選択するという点ではラティス構造を左から貪欲に探索する単語分割の手法と似ているが，辞書を用いていないため，全ての文字列を生成し公平に扱っている．また，事前にラティス構造を作る必要がなく，上記の長さの制約を除き，未知語に対する例外的な処理も必要ない．

3.2 提案モデル

本節では，3.1 節で説明した，単語分割の決定手法を行うためのニューラルネットワークのモデル構造について説明する．提案モデルの単語分割プロセスを図 1 に示す．提案モデルの表現は，大きく分けて分割候補の表現と，分割候補から単語を決定するための表現の 2 つに分けられる．

3.2.1 分割候補の表現

提案モデルはニューラルネットワークによるモデルであるため，分割候補を実数値ベクトルによって表現する．分割候補の先頭文字と末尾文字がそれぞれ文 $\boldsymbol{x}$

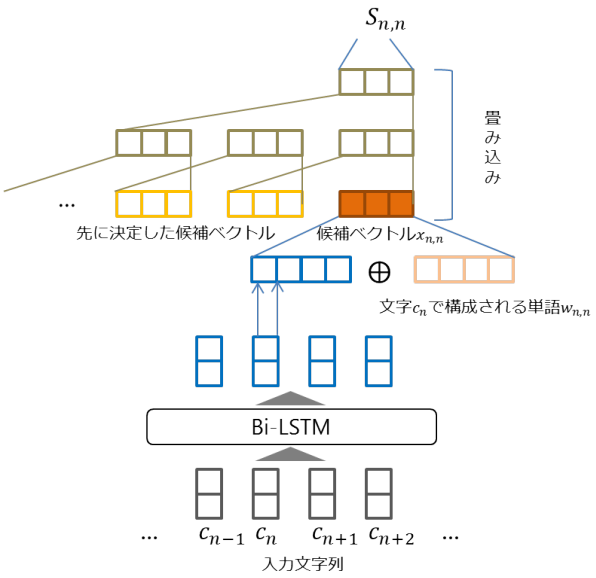

図 1: 提案モデルの単語分割プロセス

中の n，m 番目の文字である場合，分割候補の表現 $\boldsymbol{x}_{n,m}$ は，以下の式で表される．

$$\boldsymbol{h}_n = \mathrm{BiLSTM}(\boldsymbol{c}_n) \quad (3)$$

$$\boldsymbol{h}_m = \mathrm{BiLSTM}(\boldsymbol{c}_m) \quad (4)$$

$$\boldsymbol{x}_{n,m} = \tanh(\mathbf{W}[\boldsymbol{h}_n : \boldsymbol{h}_m : \boldsymbol{w}_{n,m}] + \boldsymbol{b}) \quad (5)$$

$\mathbf{W}$ は重み，$\boldsymbol{b}$ はバイアスであり，$\boldsymbol{c}$，$\boldsymbol{w}$ はそれぞれ対応する文字，単語に対応する実数値ベクトルである．この BiLSTM は，文全体を対象とした文字列上の双方向 LSTM (Long Short-Term Memory) であり，$\boldsymbol{w}_{n,m}$ は分割候補に対応するベクトル表現である．このような表現にすることで，分割候補の表現ベクトルが文中の文字列全てと分割候補の単語の情報に依存した表現となることを期待している．また，この BiLSTM は全ての分割候補で共有されており，1 文につき，一度だけ計算される．この文字情報の表現方法は既存の文字情報を利用したニューラル単語分割 [6] のベクトル表現を双方向にしている．

また，本提案では文字・単語に対応した実数値ベクトル $\boldsymbol{c}$，$\boldsymbol{w}$ を，学習の過程で更新するベクトルと事前学習済みの固定ベクトルの 2 種類で表現する．事前学習済みの固定ベクトルは大規模なコーパスで提案モデルの訓練前に学習し，単語分割の訓練データ中に出現しない文字・単語の分割精度向上を目的としている．事前学習していない文字・単語については全ての事前学習済みの文字・単語ベクトルの平均を用いる．2 種類のベクトルをともに用いるときはニューラルネット

ワークに入力する前にベクトルを結合することによって行う．その際，ネットワーク構造はベクトルの入力となる次元数のみ変更する．

3.2.2 分割候補から単語分割の決定

分割候補からの単語分割の決定では，各候補毎にスコア関数を用いて分割候補のスコア $S_{n,m}$ を算出し，高いスコアが得られる分割候補を選択する．スコア関数の表現には，以下の 2 通りの手法を提案する．

Dilated CNN を用いて決定済みの単語情報を考慮する手法

提案した単語分割手法は単語単位の単語分割を行うため，分割候補から単語分割を決定する際，それ以前の単語は既に決定している．本手法では，単語分割の過程で既に決定した単語を考慮しながら，分割候補のスコアを算出する．スコア関数は以下の式で表される．

$$S_{n,m} = \boldsymbol{w} \tanh(\mathrm{DilatedCNN}([\mathbf{E}; \boldsymbol{x}_{n,m}])) + h \quad (6)$$

$\boldsymbol{w}, h$ は重みベクトルと閾値，$\mathbf{E}$ は既に決定した単語に対応する分割候補ベクトルの列である．DilatedCNN は多層の CNN を Dilation と呼ばれるスキップ幅を増やしながら積み上げることによって計算される．各 CNN 層の間には活性化関数 tanh を用いる．このスコア関数によって考慮する決定済みの単語の数は，DilatedCNN の層数によって決まる．

決定済みの単語情報を考慮しない手法

分割候補の表現ベクトルのみからスコアを算出する．

$$S_{n,m} = \boldsymbol{w} \tanh(\boldsymbol{x}_{n,m}) + h \quad (7)$$

ここで，$\boldsymbol{w}$ と h はそれぞれ重みベクトルと閾値を表す．分割候補のスコアがそれまでの単語分割の決定に依存せず独立であるため，分割候補を構成する文字や文中の文字列のみを判断材料にする．しかし，分割候補となる部分文字列の先頭文字はそれまでの単語分割の決定によって決まるため，点予測とは異なる．

3.3 学習

提案モデルの学習は，以下に示す損失関数 L を最小化するように更新する．

$$L = -\sum \boldsymbol{y}_n \log \mathrm{softmax}(S_{n,n}, S_{n,n+1}, ...) + \frac{\lambda}{2}\|\boldsymbol{\theta}\|_2^2 \quad (8)$$

ここで $\boldsymbol{y}_n$ は n 番目の文字から考慮される分割候補の数の次元数を持ったベクトルで，正解単語の文字数 −1 番目の要素が 1 で他が 0 の one-hot ベクトルである．$\boldsymbol{\theta}$ はモデルパラメータ，$\frac{\lambda}{2}\|\boldsymbol{\theta}\|_2^2$ は正則化項であり，λ はハイパーパラメータである．式 6，7 は各分割候補に対して独立に計算されるため，式 8 は文中の単語数にかかわらず並列に計算できる．

学習時は提案モデル中の文字と単語に対応した実数値ベクトルと候補のスコアを算出する重みベクトル $\boldsymbol{w}$ の入力ベクトルにドロップアウトを行う．

4 実験設定

4.1 テキストコーパス

実験で使用したテキストデータは京都大学テキストコーパス [7] である．京都大学テキストコーパスは新聞記事から作成されたテキストデータで，日本語文が人手によって単語分割され品詞などの情報が与えられている．提案モデルの学習データ，開発データ，評価データにそれぞれ 35,900 文，500 文，2,000 文用いた．

また，提案モデルに用いる文字ベクトル，単語ベクトルの事前学習には Wikipedia 日本語版を用いた．

4.2 文字ベクトル, 単語ベクトルの事前学習

提案モデルに用いる文字ベクトル，単語ベクトルとして事前学習したベクトルを用意した．ウィキペディア日本語版 19,642,053 文を学習コーパスに用いた．学習の事前準備として，京都大学テキストコーパスに合わせるため，全ての半角文字を全角文字に変更した．単語ベクトルの学習のため，京都大学テキストコーパスの学習データで学習した KyTea を用いて分かち書きを行った．

ベクトルの学習には Skip-gram を用い，使用したハイパーパラメータを表 1 に示す．学習した文字種数，語彙数はそれぞれ 3,936，306,358 である．

4.3 学習設定

文字ベクトルの次元数，LSTM 層の出力の次元数は Chen ら [1] の設定を用い，それぞれ 100，150 とした．単語ベクトルの次元数は 200，CNN の入出力の次元数はともに 200 次元とし，ウィンドウサイズは 2 とした．CNN の Dilation は CNN 層を積み上げる毎に 1，2 と，分割候補単語を合わせて 4 単語まで，指数的に

増やした．単語の候補として考慮する最長の文字長は 22 文字とした．訓練時のドロップアウトは 20 %の確率で行った．正則化項の係数は 5×10^{-4} を用いた．最適化手法に用いた Adam の学習率は 0.001，モーメントの減衰率は 0.9，0.999 を使用した．

5 結果と考察

京都大学テキストコーパスを用いて学習し，F 値で評価を行った結果を表 2 に示す．提案モデルの比較では，Dilated CNN を用いて決定済みの単語情報を考慮したモデルが最も精度の高い結果となった．さらに，表中の提案モデル（Dilated CNN）+ PT は，3.2.1 項で述べたように，文字ベクトルと単語ベクトルを学習中に更新するベクトルと 4.2 節で事前に学習したベクトルを結合したものを用いた場合である．事前学習済みベクトルを用いることで開発データでは高い精度を記録したが，評価データでは精度が落ちている．事前学習済みベクトルを用いる動機は，訓練データ中に存在しない単語の精度を向上させることであったが，結果としては事前学習済みベクトルによる影響は無いとわかる．提案モデルは全体として，評価データの F 値が開発データの F 値より下がっており，汎化性能が高くないと考えられる．

既存の単語分割システムとの比較として，JUMAN と KyTea を用いた．KyTea は京都大学テキストコーパスの学習データで学習したものを用いている．JUMAN，KyTea はともに開発データと評価データの F 値に差が少なく，頑健なシステムだと分かる．KyTea は評価データにおいて，F 値が最も高い結果となった．

表 1: Skip-gram のハイパーパラメータ

パラメータ	値
文字ベクトルの次元数	100
単語ベクトルの次元数	200
低頻度文字を省く閾値	500
低頻度単語を省く閾値	20
コンテキストサイズ	15
ネガティブサンプリングに用いる数	5
学習率	0.025
学習回数	5

表 2: 実験結果

	F 値（%）	
モデル	開発データ	評価データ
JUMAN	97.681	97.700
KyTea	98.441	98.452
提案手法 (単語情報無し)	98.332	97.918
提案手法 (Dilated CNN)	98.475	98.207
提案手法 (Dilated CNN)+PT	98.566	98.130

6 おわりに

本論文では，単語分割の精度向上を目的として，ラティス構造を用いず単語単位で分割を行うニューラル単語分割を提案した．今後の課題としては，単語分割の決定を貪欲におこなっているため，ビーム探索を取り入れ，より構造を考慮したモデルとすることが挙げられる．

参考文献

[1] Xinchi Chen, et al. Long short-term memory neural networks for chinese word segmentation. In *EMNLP*, pp. 1197–1206, 2015.

[2] Taku Kudo, et al. Applying conditional random fields to japanese morphological analysis. In *EMNLP*, pp. 230–237, 2004.

[3] Sadao Kurohashi, et al. A method of case structure analysis for japanese sentences based on examples in case frame dictionary. *IEICE*, Vol. 77, No. 2, pp. 227–239, 1994.

[4] Hajime Morita, et al. Morphological analysis for unsegmented languages using recurrent neural network language model. In *EMNLP*, pp. 2292–2297, 2015.

[5] Graham Neubig, et al. Pointwise prediction for robust, adaptable japanese morphological analysis. In *ACL*, pp. 529–533, 2011.

[6] 池田大志ら．辞書情報と単語分散表現を組み込んだリカレントニューラルネットワークによる日本語単語分割．In *NLP2016*, pp. 879–882, 2016.

[7] 黒橋禎夫ら．京都大学テキストコーパス・プロジェクト．人工知能学会全国大会論文集, 第 11 巻, pp. 58–61, 1997.