語彙知識を取り入れた韓国語語節分割

金山 博　　小比田涼介
hkana@jp.ibm.com　　kohi@ibm.com
日本アイ・ビー・エム株式会社 東京基礎研究所

1 はじめに

韓国語の表記においては、語節（어절）と呼ばれる単位ごとに空白が挿入される。人間が読む場合も典型的な機械処理でも空白の存在を前提としているが、入力を急ぐ場合や装置の都合などのためか、空白が挿入されていない韓国語の文書が存在する。また、文末の改行文字や空白が、語節の区切りなのか、一行の文字数の制限によるものなのかが曖昧なデータもあり、空白を含む記法でも語節の単位として信頼できない場合もある。これらの問題の解決のため、空白を自動挿入するタスクが試みられてきた [6][4]。また、モバイルデバイスの入力支援の目的の省メモリの手法 [3][7] も提案されている。

(1a) は医療保険への対応に関する記述の実例で、空白が一切書かれていない。これは、本来は (1b) のように空白を伴って表記される 3 つの文である。

(1a) 입원치료후완치그후치료력없으시고현재이상없습니다인수심사바랍니다
(1b) 입원치료 후 완치 ('入院治療後完治')
그 후 치료력 없으시고 현재 이상 없습니다 ('その後治療歴無く現在は異常ありません')
인수심사 바랍니다 ('引受審査お願いします')

(1a) のような文に既存の形態素解析器を施すと、単語分割や品詞推定の性能が大きく低下する。特に、Universal Dependencies のコーパス [5] は、空白や約物のみを語の区切りと規定しているため、その上で訓練された UDPipe[8] などの解析器では、(1a) の入力全体が一つの語として扱われてしまう。

本稿では、現実に存在する空白を伴わない韓国語文への対応を目的とした、形態素解析の前処理としての語節分割について述べる。文字単位の系列ラベル付け問題として、双方向の LSTM と CRF を用いて実装をし、学習データ削減と分野適応のために、直感的に人手で語彙を追加する機構を追加した。また、出力結果より、LSTM モデルの中で学習されている事象を考察する。

2 韓国語の語節と空白

韓国語の語節は日本語の文節と類似した単位で、名詞や動詞などの内容語に、助詞や語尾などの付属語が付加された単位である（内容語のみの場合もあり）。その区切り方は正書法として定められているものの、国家や時代による正書法の差異があり、特に複合名詞や補助動詞の扱いなどにおいて揺れがある。

例えば「勉強することができる」は (2a) のように、依存名詞「수」の前後に空白を置くのが正書法に則った表記であるが、(2b)(2c) のように空白が省略されることがあり、特にマイクロブログなど非公式な場面ではその傾向が強まる。

(2a) 공부할 수 있다 ('勉強することができる')
(2b) 공부할수 있다
(2c) 공부할수있다

複合名詞は要素ごとに分割するのが原則であり、「後遺障害 (후유장해)」は (3a) のように「後遺 (후유)」と「障害は (장해는)」に分割されるが、分野に特化した文書では (3b) のように空白を置かないことも多く、いずれも誤りとは言えない。

(3a) 후유 장해는 없고 ('後遺障害は無く')
(3b) 후유장해는 없고

上記のような本質的な揺れがあるため、空白挿入を完全に予測することは不可能であるが、複数の書き方が頻繁に現れる事象においては、後段の処理も不都合なく行えるので、前処理としての実用上は、明らかに不適切な空白を挿入しなければ有用となる。

空白挿入のタスクは、純粋な語節の単位の認定のほか、引用符の配置の解決も含む。文 (4) のように開閉で同じ引用符が使われることが多く、空白が引用符の左右どちら側に入るか、また引用助詞を含むかの認識にはより広い範囲の文字を見る必要がある。

(4) 학생들도 "재미있는 책이었다"고 술회했다.
('学生たちも「面白い本だった」と述懐した。')

3 手法

3.1 系列ラベリング

語節への分割を、文字単位の系列ラベリングの問題として捉えて、それぞれの文字の後に空白を挿入すべきか否かを推定する。例として、文 (2a) に対しては、表 1 のように、空白を取り除いた文字毎に、語節の始まり（文頭または空白の後）であれば B、そうでなければ I のタグを付けて、これを推定するモデルを作る。

表 1　文字ごとのタグの例。

原文	공부할 수 있다.						
文字単位	공	부	할	수	있	다	.
ラベル	B	I	I	B	B	I	I

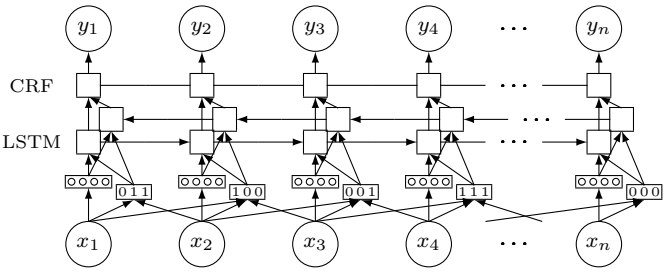

図 1　文字の系列 $x_1 \ldots x_n$ から空白挿入のラベル $y_1 \ldots y_n$ を推定するネットワークの図。

なお、空白が挿入された通常の韓国語の文から自明にラベル付きデータが作れるため、訓練・テストに用いるデータは極めて容易に収集できる。

3.2　双方向 LSTM モデル

本研究の系列ラベリングで用いるモデルを図 1 に図示する。それぞれの文字の埋め込み表現と、辞書とのマッチングの結果の素性 (3.3 節で後述) を用いて、双方向 LSTM[1] を使って層を形成する。これにより、当該文字の左側と右側の任意長の文脈を考慮することができる。

さらに、出力されたラベルの関係を最適化するために、CRF の層を加える。これにより、他の出力との相互関係から語長を均一化させることが期待できる。

3.3　辞書の導入

今回のタスクで重視したいことが、語彙知識を使って利用者が出力を制御する余地を与えることである。特に、分析対象とする分野に特化した語があった時に、その語を含む訓練データを大量に用意することなく、語の単位だけを与えることによって簡単に分野適応ができることが望ましい。

Zhang らは中国語のタギングのために、双方向 LSTM の入力に辞書の情報を組み合わせる手法を提案した [9]。ここではその方法に倣って以下のような素性を追加する。

入力文中で注目する文字 x_i があった時に、m 文字前からの文字列 $x_{i-m} \ldots x_i$ と、m 文字先までの文字列 $x_i \ldots x_{i+m}$ が辞書にあるか否かをエンコードした二値素性のベクトルを用いる。Zhang ら [9] は n が 1 から 4、すなわち 2-gram から 5-gram の文字を使っていたが、中国語の単語と異なり、韓国語の語節の場合、単語に対して語尾が付いたものが語節の単位となること、また一文字で単語となる文字が中国語と比べて限られていることから、単独の文字からなる辞書にも意味があると考え、m=0、すなわち単独の文字も認めることとし、$2n+1$ の長さの素性ベクトルを用いることにした。$n = 3$ とした場合、文 (5) の左から 5 文字目の「전」について、辞書に基づいた素性ベクトルの作り方を図 2 に示す。

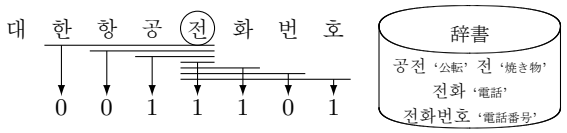

図 2　文 (5) の 5 文字目の辞書素性ベクトル。

(5) 대한항공전화번호 ('大韓航空電話番号')

4　実験

3 節の手法を TensorFlow で実装したもので実験を行った。訓練には、Wikipedia 韓国語版のダンプに対して WikiExtractor[*1]を使ってテキスト部分を取得したものを用いる。テストには公的機関や IBM 社の web サイトの文 (以下 web コーパス) を用いた。さらに、訓練・テストや辞書の構築に Universal Dependencies[5] のうち UD_Korean-GSD のコーパスを用いた[*2]。いずれの場合も、テストに用いるファイルからはすべての空白を除去してあり、一行 (概ね一文) 毎に解析を行う。以降の実験では、文字の埋め込み、LSTM 層の次元は 100 に固定している。

4.1　モデルと訓練量

まず、Wikipedia の文で訓練をして、web コーパスへの空白挿入をした時の正解率を表 2 に示す。学習に用いる文数は XS から XL まで変化させた。なお、ここでは 3.3 節で説明した辞書は用いていない。

XS の CRF 有の 69.79% は全てに I タグが出力された結果であるが、それ以降は訓練量の増加に伴い精度が上がっている。M や L のデータで訓練した場合、CRF の層を加えた場合に精度が上回っている。以降では CRF を用いて実験を行う。

Wikipedia で学習したモデルを、韓国語 UD のテストデータに適用した時の結果を表 3 に示す。また、後段の処理への影響を見るため、空白挿入の結果に対して UDPipe [8] の tokenize をした場合の、Word 同定の F1 値を計測した。表 3 中の「UDPipe」は、空白を除去した UD の訓練コーパス (4400 文) のアノテーションを使って UDPipe の単語分割モデルを再学習した場合の性能である。これらの結果を比較すると、提案手法により 3 万文以上の生コーパスを用いた訓練をすれば、UDPipe の性能を上回る前処理ができることがわ

[*1] https://github.com/attardi/wikiextractor
[*2] 他方、UD_Korean-KAIST のコーパスは、句読点の前後に空白が人工的に挿入されているので今回の実験には不適である。

表 2　web コーパスへの空白挿入の正解率。CRF 有に付された $^{+}$, $^{-}$ は、CRF 無の場合より有意に性能が高い/低いことを示す。

	訓練量	CRF 無	CRF 有
XS	118 文	71.83%	69.79%$^{-}$
SS	868 文	86.99%	85.12%$^{-}$
S	6,355 文	90.40%	90.02%
M	32,141 文	93.16%	93.88%
L	149,905 文	94.76%	95.31%$^{+}$
LL	704,929 文	95.20%	94.76%
XL	1,525,473 文	96.58%	95.70%$^{-}$

表 3　UD コーパスの空白挿入の正解率と、後段の UDPipe によるトークナイズの性能。

訓練量	空白挿入	Word F1
XS	69.73%	0.199
SS	83.95%	0.560
S	89.50%	0.658
M	93.51%	0.809
L	94.74%	0.829
LL	94.26%	0.812
XL	95.61%	0.844
UDPipe		0.787

ID	表層	正規形	UPOS	XPOS
1	소이현의	소 + 이현 + 의	NOUN	XPN+NNP+JKG
2	매력이	매력 + 이	NOUN	NNG+JKS
3	담겨	담기 + 어	VERB	VV+EC

図 3　UD コーパスの例（1〜5 カラム目のみ表示）。

かる。

4.2　辞書の利用

次に、3.3 節で説明した辞書の素性を導入する。以下の通り、UD コーパスからの抽出と、既存の形態素解析で用いる辞書の流用を試みた。

UD 表層: UD コーパス (train 及び dev) の語節の表層形。空白で区切られた単位そのものを機械的に列挙した。図 3 の例では '소이현의', '매력이', '담겨' を抽出。33,082 語、最大長 36。

UD 内容語: UD コーパスの内容語。lemma として分割された内容語 (XPOS が N, V, M で始まるもの) を抽出。図 3 では '이현','매력', '담기' が抽出される。13,009 語、最大長 12。

UD 名詞: UD コーパスの lemma から一般名詞 (NNG)・固有名詞 (NNP) を抜き出したもの。図 3 では '이현' と '매력'。9,399 語、最大長 11。

システム内容語: 形態素解析器の辞書のうち、体言・用言の語幹・副詞・数詞を含む全ての語。168,649 語、最大長 53。

システム名詞: 形態素解析器の辞書のうち、一般名詞と固有名詞。102,202 語、最大長 41。

各リソースの語の最大長に合わせて素性を追加する。すなわち辞書エントリの最大長が 36 の場合は $n = 35$ として、長さ 71 の素性ベクトルを用いる。

表 4　辞書を導入した場合の空白挿入の正解率（%）。$^{+}$, $^{-}$ は辞書無しの場合との有意な差を示す。

訓練	辞書無し	UD 表層	UD 内容語	UD 名詞	sys 内容語	sys 名詞
XS	69.79	72.33^{+}	69.62	69.79	69.79	69.79
SS	85.12	91.29^{+}	89.47^{+}	86.83	87.87^{+}	88.04^{+}
S	90.02	92.01^{+}	92.83^{+}	90.24	92.45^{+}	91.07
M	93.88	92.83	94.43	94.54	94.54^{+}	95.09^{+}
L	95.31	95.37	95.37	95.48	95.20	95.59
LL	94.76	94.54	96.53^{+}	96.42^{+}	94.37	95.53^{+}
XL	95.70	94.70^{-}	96.56	96.58^{+}	96.80^{+}	96.80^{+}

表 4 に辞書を加えた時の性能を示す。この結果より、訓練データが小さい場合には、「UD 表層」の辞書のように、空白を挿入すべき文字列の単位を直接与えることに効果がある一方で、充分な訓練ができる時には、名詞や内容語の辞書の効果が大きいことがわかる。これは、名詞や固有名詞の単位を与えて、文法的な傾向とともに知識を汎化させることができたためと考えられる。テストセットが異なるので直接の比較はできないものの、先行研究の正解率 94.7%[6]〜97.5%[4] に近い水準が得られている。

5　考察

5.1　分野適応

辞書の追加による分野適応の効果を調べるために、Wikipedia 韓国版から選んだ生物学[*3]・スポーツ[*4]・芸能[*5]のページを用いて、次の方法で実験を行った。

- 各ページから、テスト文を無作為に 40 文ずつ抽出
- 当該ページにある Wikipedia のリンクの語から追加辞書を作る。空白を含む場合は空白ごとに区切った単位を登録
- 「UD 名詞」の辞書と XL のコーパスで訓練したモデルをベースとし、元の辞書を使った場合と、そこに前プロセスで作成した辞書を追加したものを使って空白挿入の正解率を比較

その結果を表 5 に示す。簡便かつ客観的な方法で辞書を追加したが、いずれの分野でも正解率が向上しており、特に外来語を含む語節などが正しく分割されるようになった。名詞・固有名詞など分野に特有の語を追加する作業のみで、再学習も不要なので、実応用でも有用であると期待できる。

5.2　機能語の認識

中国語のタギングに辞書を適用する場合 [9] は、辞書の単位が単語の切れ目と一致していたのに対し、本稿

[*3] https://ko.wikipedia.org/wiki/미토콘드리아
[*4] https://ko.wikipedia.org/wiki/요코하마_베이스타스
[*5] https://ko.wikipedia.org/wiki/모닝구무스메

表 5　各分野の辞書を加える前後の空白挿入の正解率の比較。

分野	追加語彙数	辞書追加前	辞書追加後
生物学	395	96.96%	97.51%
スポーツ	587	98.30%	98.85%
芸能	906	96.82%	97.23%

表 6　名詞＋助詞が語節として認識されるかどうかの観測。○は成功、×は失敗。

モデル / 助詞	辞書追加無	UD 表層	UD 内容語	UD 名詞
이 (‘が’)	×	×	○	○
을 (‘を’)	×	○	○	○
에서 (‘から’)	×	○	○	○
보다는 (‘よりは’)	×	×	○	○
한테 (‘(人) に’)	×	×	×	×

表 7　距離毎の引用符前後の空白挿入の正解率。

距離	正解率
0〜5	94.9%
6〜10	92.3%
11〜	90.3%

で用いる辞書は空白の挿入を直接的には示唆しない[*6]。そこで、上の実験の「生物学」で考慮した「네안데르탈인」(‘ネアンデルタール人’) という語が複数の辞書に追加された時、それらに助詞が付加されたものが語節として認識されるかを観察した。表 6 に結果を示す。

空白区切りの単位そのものを辞書化した「UD 表層」よりも、内容語や名詞の部分だけを扱う辞書のほうが、追加した語が名詞であり助詞を伴うと認識されやすいこと、また頻度が高い助詞ほど各種モデルで適切に学習されていることが確認された。

5.3　長距離の文脈

本手法では韓国語の文字だけでなく、アルファベットや漢字などの他言語の文字、句読点や引用符等の記号も一律に扱い、注目する文字の前後に空白を挿入するかどうかを推定している。引用符は (4) のように語節のさまざまな位置に現れ、特に左右で同じ文字を用いる二重引用符については、文全体の文脈を見ないと空白挿入の判断が難しい。

そこで、引用符の前後の空白が正しく推定できた割合を、当該引用符から別の引用符ないし文頭・文末までの文字数[*7]ごとに調べたのが表 7 である。長距離になるほど推定が難しくなるのは直感通りであるが、10 文字を超える距離であっても 90% 以上の正解率は保っており、LSTM が引用符の出現を記憶して長距離にわたって引き継いでいることがわかる。

[*6] 「UD 表層」の辞書は例外。

[*7] 例えば (4) にある 2 つの引用符の場合、1 つ目は文頭からの文字数 4、2 つ目は文末からの文字数 6（いずれも直近の引用符までの文字数 8 より小さい）と見なす。

6　おわりに

本研究では、従来研究で文字 n-gram の HMM などで解決していた韓国語の語節分割を、LSTM と CRF を用いて実現し、一定の精度が得られた。また、辞書を用いて性能の向上を図り、簡便な分野適応ができることを示した。

さらに、双方向 LSTM のモデルで、文法の特性や長距離の関係がどれだけ捉えられているかを観察した。このようにニューラルネットの特性の理解は近年興味を集めており [2]、このようにデータが入手しやすいタスクを題材として調査することには意義があると考えられる。

参考文献

[1] Alex Graves and Jürgen Schmidhuber. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, Vol. 18, No. 5-6, pp. 602–610, 2005.

[2] Urvashi Khandelwal, He He, Peng Qi, and Dan Jurafsky. Sharp nearby, fuzzy far away: How neural language models use context. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pp. 284–294, 2018.

[3] Shinil Kim, Seon Yang, and Youngjoong Ko. A novel discriminative probabilistic model for automatic word spacing on mobile devices. *International Journal of Innovative Computing, Information and Control*, Vol. 8, No. 7, pp. 1–11, 2012.

[4] D. Lee, H. Rim, and D. Yook. Automatic word spacing using probabilistic models based on character n-grams. *IEEE Intelligent Systems*, Vol. 22, No. 1, pp. 28–35, Jan 2007.

[5] Joakim Nivre, et al. Universal Dependencies 2.3, 2018.

[6] Seong-Bae Park, Yoon-Shik Tae, and Se-Young Park. Self-organizing n-gram model for automatic word spacing. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pp. 633–640, Sydney, Australia, July 2006. Association for Computational Linguistics.

[7] Yeongkil Song and Harksoo Kim. Lightweight word spacing model based on short text messages for social networking in smart homes. *International Journal of Distributed Sensor Networks*, Vol. 10, No. 2, p. 532759, 2014.

[8] Milan Straka, Jan Hajič, and Jana Straková. UDPipe: trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, Portorož, Slovenia, 2016. European Language Resources Association.

[9] Qi Zhang, Xiaoyu Liu, and Jinlan Fu. Neural networks incorporating dictionaries for chinese word segmentation. In *Thirty-Second AAAI Conference on Artificial Intelligence (AAAI)*, pp. 5683–5689, 2018.