系列ラベリングBERTモデルを用いた述語項構造解析

力石 翔太　　白山 晋
東京大学大学院 工学系研究科
chikaraishi-shota405@g.ecc.u-tokyo.ac.jp, sirayama@sys.t.u-tokyo.ac.jp

1 はじめに

述語項構造解析（Predicate Argument Structure Analysis, PASA）は，述語に対する項と，その役割（格）を推定する技術である．つまり，ある述語に対して，その主語（主格）や直接目的語（対格），間接目的語（与格）が，どの単語であるかを推定することができる．

こうした情報は，英語の構文解析では依存関係と同時に推定される．しかしながら，従来の日本語の係り受け解析ではエッジラベルを用いないため，述語項構造解析を用いて別途推定する必要がある．このような直接の係り受け関係にある項の格を推定する技術を，特に格解析と呼ぶ．

述語項構造解析に関する研究では，日英機械翻訳の精度向上が動機のひとつとして挙げられることが多い．日本語は，英語に比べて項の省略が多いため，これを補完することで精度向上が期待されている．省略項を補って格の推定まで行う処理のことをゼロ照応解析と呼び，述語項構造解析は，格解析とゼロ照応解析を総称した技術を指す．

しかし，現在では，ニューラルネットワークによる系列変換を用いた翻訳モデルが高い性能を示しているため，翻訳精度向上という動機は以前ほど強くないと考えられる．

一方で，述語と項の格関係を推定することは，生文に対する正規化・構造化とも考えられ，本研究はその方向性で進めたものである．この点に関しては，参考文献 [1] においても，情報抽出に用いる考えが述べられている．また，PASA を用いて生文を構造的に扱う実用的な研究も存在する [2, 3]．

生文から品質のよい構造化データを得るためには，PASA の精度向上が不可欠である．そこで本研究では，強力な事前学習済みモデルである BERT[4] を用いた PASABERT を提案し，その有効性を確認する．

2 関連研究

Zhou ら (2015) は，PASA に類似したタスクである英語の意味役割付与を，RNN と CRF を用いた系列ラベリングで解くモデルを提案した [5]．Ouchi ら (2016) による，このモデルの日本語述語項構造解析への拡張 [6] が，優れた性能を示したため，その後，系列ラベリングモデルを用いた研究が盛んになっている．

この系列ラベリングをベースに，文中の複数の述語を同時に考慮する [7]，Pointer Network[8]，事態性名詞とマルチタスク学習 [9] 等の工夫を加えた研究が行われている．これらの先行研究は，後述のように主として同一文内における文内述語項構造解析を対象としている．

3 提案モデル

テキストに対する系列ラベリングの性能は，単語埋め込みから得られる分散表現と，RNN などによる単語同士の相互作用機構の品質に依存していると考えられる．そのため，この部分をより良いモデルで代替することで，精度が向上する可能性がある．

本研究では，これに事前学習済みの BERT モデルを用いる．BERT モデルは，自然言語処理の種々のベンチマークで記録を更新しており，述語項構造解析においても高い性能向上が期待できる．

BERT の提案論文 [4] で，すでに BERT を固有表現認識のための系列ラベリングモデル（トークン単位の分類問題）として使用する方法が示されている．日本語に関連する研究では，Shibata ら (2019) の構文解析への適用がある [10] が，BERT を用いて日本語述語項構造解析を解く研究は著者らの知る限り存在しない．

これらのモデルの構成を踏襲し，BERT で述語項構造解析を扱うモデルについて述べる．以降，我々の提案モデルを PASABERT と呼ぶことにする．

3.1 述語の指定方法

述語項構造解析では，解析対象とする述語を指定する必要がある．これに対し，Segment Embeddings を用いる方法と，述語を含む文節を 2 文目として追加する方法を考える．

方法 1　Segment Embeddings を用いる方法

各トークンに対して，解析対象の述語か否かのフラグ $f = f_1, ...f_n$ を付与することで，モデルに述語を識別させる．これは先行研究 [6, 7] でも標準的な方法になっている．

$$f_t = \begin{cases} 1 & (predicate) \\ 0 & (otherwise) \end{cases}$$

一方，BERT は 2 文入力が可能であり，1 文目と 2 文目を区別するために，各トークンに対して segment id というフラグを利用する．

$$segment\ id = \begin{cases} 0 & (first\ sentence) \\ 1 & (second\ sentence) \end{cases}$$

segment id は，segment embeddings 層で変換され，トークンの分散表現に加算される．

2 文入力を用いるタスクには，2 文の意味的類似の算出や，含意関係認識などがあり，これらのタスクでは，2 文を区別できる形でモデルに入力する必要がある．

一方で，多くの先行研究で扱っている文内述語項構造解析では，文字通り 1 文のみを扱うため，segment id を用いて文を区別する必要がない．

そこで，述語フラグ f を，この segment embeddings 層に入力することで，BERT モデルに対して述語部分を指定する方法を提案する．これを PASABERT1 と称することにする．

述語フラグを，BERT 入力層の段階から入力することで，述語位置情報を，BERT 内部の Attention 機構によるトークン同士の相互作用に利用できる．

また，NTC では，述語や格のタグは形態素ごとに付与されているが，サ変名詞（例: 行動する）に対しては「する」に述語タグが付与されている．しかし，「する」の前にある名詞（「行動」など）が，具体的な意味を担っていると考えられる．このため，述語部の意味をとらえるという点では，この名詞も含めて述語フラグを立てるほうが有効だと考えられる．したがって，述語を含む文節全体に述語フラグを立てる．

PASABERT1 では，単語 ID 列 $w = w_1, ..., w_n$ と，述語フラグ列 $f = f_1, ..., f_n$ に対し，格ラベル確率ベクトル p_t を予測する．

$$h = BERT(w, f)$$

$$p_t = softmax(W_o h_t + b_o)$$

BERT の出力は，$h \in \mathbb{R}^{768 \times n}$ である．各トークンの格ラベル確率ベクトルは，$p_t \in \mathbb{R}^4$ であり，ガ格（GA），ヲ格（WO）ニ格（NI），いずれにも該当しない（O），それぞれに対する確率値を要素にもつ．

方法 2 述語を含む文節を 2 文目として追加する方法

BERT は 2 文入力が可能であるが，文内述語項構造解析は 1 文を扱うため 2 文目は存在しない．そこで述語を含む文節を文末に追加し，これを 2 文目として BERT に入力する方法を提案する．これを PASABERT2 と称することにする．

PASABERT2 では，単語 ID 列 $w = w_1, ..., w_n$ と，述語文節 ID 列 $w_p = w_{p1}, ..., w_{pm}$ に対し，格ラベル確率ベクトル p_t を予測する．

$$h = BERT([w; w_p])$$

$$p_t = softmax(W_o h_t + b_o)$$

ただし，2 文目 (述語文節) は，精度評価には使用しない．

4 実験

4.1 実験設定

4.1.1 事前学習済みモデル

東北大学で公開されているモデル (BERT-base_mecab-ipadic-bpe-32k_whole-word-mask) を使用した[1]．

4.1.2 NAIST テキストコーパス

学習・評価には NAIST テキストコーパス ver1.5 を用いる．NTC は，2,979 記事，34,800 文からなる．

学習・開発・評価セットへの分割は，多くの先行研究で採用されている Taira ら (2008) の分割に従う [11]．

[1] https://github.com/cl-tohoku/bert-japanese

この結果，学習 24,283 文, 開発 4,833 文, 評価 9,284 文になる．

また，先行研究にならい，係り受け有の項に対する格解析 (以降，Dep と略す) および文内ゼロ照応 (以降，Zero と略す) を扱い，文間ゼロ照応および外界ゼロ照応は扱わない．

NTC では，係り受け情報は文節単位で付与されている．Taira らは，項を含む文節が述語を含む文節に係るケース，述語を含む文節が項を含む文節に係るケース，の両方を係り受け有として扱っているため，この点も同様に従う．

PASABERT では，述語文節全体を述語として扱うため，正解の項と述語が同一文節内にあるケースも扱わないものとする．同様の設定を採用している先行研究も存在する [9]．

4.1.3 学習・評価に用いる情報

単語境界および文節境界

NTC に付与されている分割を使用する．

述語位置

述語タグのついた語のみを学習・評価対象とする．NTC では，述語を「動詞」「形容詞」「サ変名詞＋する」「名詞句＋だ」と定義している．こういった情報は，運用時において形態素解析等の技術で取得可能であるため，既知の情報として扱う．

述語タグが付与されていても，係り受け有の項や，文内ゼロ照応の項を持たない場合（文間ゼロ照応や，外界照応のみを持つ）場合，前処理によって述語フラグはつくものの正解の項がないデータ（格ラベルがすべて O のデータ）が生じる．

文内に正解の項がないという情報は，アノテーションされた正解ラベルから得られる情報なので，これらのデータも除外せず，学習や評価に用いる．学習データからのみ取り除く場合は問題ないが，文内に必ず正解の項が含まれるデータのみで学習すると，そのようなデータに過学習する恐れがある．

評価データや，モデルを実際に運用する際に入力されるデータには，文内に正解の項がないデータも当然存在する．このようなデータに対して，すべての項について格ラベル O を出力することもモデルの振る舞いとして重要であり，より適合率の高いモデルになると考えられる．

一方で，述語文節の位置および，述語文節内のトークンに正解の格ラベルが存在しないことは既知であるため，述語文節に対する予測値は精度評価には用いないとする．

係り受け

提案手法において，学習時には，明示的に係り受け情報をモデルに入力していない．

学習・予測する格ラベルは，GA, WO, NI, O の 4 種類であるが，評価時には，先行研究との比較のため，Dep/Zero で個別に精度を示す必要があり，係り受け情報を使用している．

正解ラベル，予測ラベルともに，評価時に，係り受け有の文節内トークンに対するラベルには末尾に 1 を，それ以外の文節内トークンに対するラベルには末尾に 2 を追加する．具体的には，GA1, GA2, WO1, WO2, NI1, NI2 とする後処理を加える．ただし，ラベル O にはこの処理は行わない．

4.1.4 サブワードの扱い

学習時の辞書の構築方法に依存するが，BERT は分かち書き処理で，単語をさらにサブワードに分割するモデル/実装が多く，今回使用するモデルもこれに該当する．

ある単語が複数のサブワードに分割される場合，関連研究に習い [4, 10]，先頭トークンの予測値のみで精度評価を行い，2 つ目以降のトークンは無視する実装にする．これらは元はひとつの単語なので，先頭トークンのラベルが予測できれば，以後のサブワードのラベルも同一ラベルと見なすことができるためである．

4.1.5 学習時のパラメータ

- Epoch: 3
- Learning rate: 3e-5
- Batch size: 32
- Optimaizer: Adam
- Maximum sequence length[2]: 256

[2]学習データの最大トークン数が，BERT の前処理によるサブワードも含めて 205 トークンであったため．

		Dep			Zero		
MODEL	ALL	GA	WO	NI	GA	WO	NI
PASABERT1	87.63	93.54	96.37	72.14	67.69	55.84	29.18
PASABERT2	87.59	93.44	96.30	72.82	67.85	55.16	20.71
Omori 2019 (ens. of 5)	86.01	92.15	95.80	72.95	57.84	45.20	0.00
M&I 2018 (ens. of 10)	85.34	91.84	95.57	70.80	60.21	52.50	26.00
Ouchi 2017	81.42	88.75	93.68	64.38	50.65	32.35	7.52

表 1: **NTC1.5 評価セットに対する F 値．参考として同様に NTC1.5 を用いた先行研究から，F 値（全体）が最大のモデルの精度を示す [9, 7, 12].**

4.2 実験結果

実験の結果を表 1 に示す．係り受けの有無および格ごとの F1 スコアで評価した．Dep の NI 格を除き，先行研究よりも高い精度を達成した．

特に，Zero に関しては，精度が大きく向上した（表 1 における先行研究の最高性能に比べて，PASABERT1 は，GA+7.48, WO+3.34, NI+3.18）．

これは BERT の内部で使用している Attetion 機構が，RNN 系のモデルに比べ，長距離の依存関係を捉えるのに上手く作用しているためと考えられる．先行研究においても，Zero の精度が高い Matsubayashi と Inui の研究 [7] では，RNN（GRU）に Self-Attetion を適用したモデルを使用している．

5 おわりに

本研究では，系列ラベリング BERT モデルを述語項構造解析に適用する手法を提案した．

述語部の指定方式に，segment embeddings を利用する方法と，2 文入力を利用する方法を考え．segment embeddings の利用がより高い性能を示すことを確認した．どちらの方法も，BERT の仕組みを有効に活用しており，自然な拡張になっていると考えられる．また，これらを実装した結果，課題となっていて文内ゼロ照応に関して，特に大きく精度が改善した．

今後は，関連研究で提案されている種々のテクニックを，本提案手法に自然な形で取り込む方法を考えることで精度向上を図りたい．

参考文献

[1] 飯田龍, 小町守, 井之上直也, 乾健太郎, 松本裕治. 述語項構造と照応関係のアノテーション: NAIST テキストコーパス構築の経験から. **自然言語処理**, 17(2):2_25–2_50, 2010.

[2] 岡崎健介, 村田真樹, 馬青. 複数文書からの重要情報の抽出と表の生成. **言語処理学会第 *24* 回年次大会発表論文集**, 2018.

[3] 村山友理, 小林一郎, 森田武史, 中野有紀子, 山口高平. 自然言語の SPARQL クエリ変換に基づく構造化知識へのアクセス手法の開発. **人工知能学会全国大会論文集**, JSAI2018:1J104–1J104, 2018.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[5] Jie Zhou and Wei Xu. End-to-end learning of semantic role labeling using recurrent neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1127–1137, 2015.

[6] 大内啓樹, 進藤裕之, 松本裕治. 深層リカレントニューラルネットワークを用いた日本語述語項構造解析. **研究報告自然言語処理** *(NL)*, 2016(21):1–8, 2016.

[7] Yuichiroh Matsubayashi and Kentaro Inui. Distance-free modeling of multi-predicate interactions in end-to-end Japanese predicate-argument structure analysis. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 94–106, 2018.

[8] 高橋啓吾, 大森光, 小町守. Pointer networks を用いた文内述語項構造解析. **人工知能学会全国大会論文集**, JSAI2019:4Rin112–4Rin112, 2019.

[9] 大森光, 小町守. マルチタスク学習を用いたニューラル文内述語項構造解析. **言語処理学会第 *25* 回年次大会発表論文集**, 2019.

[10] 柴田知秀, 河原大輔, 黒橋禎夫. BERT による日本語構文解析の精度向上. **言語処理学会第 *25* 回年次大会発表論文集**, 2019.

[11] Hirotoshi Taira, Sanae Fujita, and Masaaki Nagata. A Japanese predicate argument structure analysis using decision lists. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 523–532, 2008.

[12] Hiroki Ouchi, Hiroyuki Shindo, and Yuji Matsumoto. Neural modeling of multi-predicate interactions for Japanese predicate argument structure analysis. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1591–1600, 2017.