

類似用例の利用により多義性に対応した 日本語述語項構造解析

芝原 隆善^{a)} 濱口 拓男^{b)} 松本 裕治^{c)}

概要：本研究では文内の日本語述語項構造解析に取り組む。述語項構造解析は述語とその項の間の関係を推定するタスクであり、その推定には文の構造を把握するだけでなく、指定された単語がどのような意味を持つか把握することが重要である。単語の意味に関して、先行研究は2つのアプローチを取っている。1つは用例の分類として定められた意味を利用するアプローチであり、もう1つは単語埋め込みに意味を学習させる事前訓練によるアプローチである。これらのアプローチはそれぞれ利点を持つ反面、1つの共通した課題を持つ。それは単語の意味を静的に決めてしまう点である。用例の分類として語義を定めるアプローチは典型的な意味以外の用法や用例を捉えるのが難しい。事前学習による単語埋め込みを用いるアプローチは、語義を固定長のベクトルに全て押し込めるため、多くの語義を持つ単語の意味を取りこぼしてしまう懸念がある。これらの課題を解決するため、本研究では外部コーパスから動的に単語表現を獲得するアプローチを取る。より具体的には、着目した単語に類似する用例を外部コーパスから集め、それぞれに対して Attention を用いることで、似た用例とそれに基づく単語表現を獲得することで、日本語述語項構造解析をより効果的に解くことを目指す。

キーワード：日本語述語項構造解析

TAKAYOSHI SHIBAHARA^{a)} TAKUO HAMAGUCHI^{b)} YUJI MATSUMOTO^{c)}

1. はじめに

述語項構造解析は述語とその項の間の関係を推定するタスクである。例えば「太郎は花子にご飯を作った。」という文があった時、「太郎」は「作る」の主格である。このような述語項構造の獲得は、一般的にはパーズングなどの文の構造を利用して解かれることが多いが、しかし指定された単語がどのような意味を持つか把握することもまた重要である。前述の例では、「太郎」は一般に人名であり、「太郎を作る」というような内容として記述されることは少ない。このような単語の意味を把握することは、特にゼロ照応などを含む述語項構造解析においては重要な役割を持つ。

近年の語義を述語項構造解析に役立てる先行研究には、次の2つのアプローチが存在する [4], [7], [8]。1つ目は、単語の用例をクラスタリングしそれらを単語の代表的な意味として扱うアプローチである [7], [8]。特に格フレ

ム [1] はこのアプローチの最たる例であり、多くの研究がなされている。これらの研究の利点は、シソーラスを元にした語義を利用することにより、解釈をすることが容易である点を挙げることができる。もう1つのアプローチは事前学習を通じて単語埋め込みにコーパスの情報を覚えさせる方法である [4]。これらは、例えば cBoW [5] などを用いることにより、アノテーションなしのコーパスなど、より多くの情報を活用することができる。

先行研究のアプローチは、語義を明示的・暗黙的に扱えるなどそれぞれ利点を持つ反面、同時に課題も持つ。それは単語の意味を静的に決めてしまう点である。用例のクラスタリングを事前に行うアプローチは、クラスタリングの粒度の選定に恣意性があり、その粒度が別のタスクで最適な粒度であるとは限らない。例えば、形態素解析と意味解析における、品詞と意味役割のように、用例を同一であると見なす基準はタスクにより異なる。単語埋め込みによる語義の活用は、アノテーションなしのコーパスから情報を抜き出すことはできるが、1つのベクトルに単語の意味を全て押し込める難しさを持つ。例えば、「持つ」などの一般

ⁱ¹ 現在, Nara Institute of Science and Technology

a) shibahara.takayoshi.sk4@is.naist.jp

b) takuo-h@is.naist.jp

c) matsuo@is.naist.jp

的な単語は多くの語義を持ち、「棚引く」などの専門性の高い単語は限られた語義しか持たない。それら一様ではない語義を持つ単語を全て固定長のベクトルとして表現することは、情報の取りこぼしや、過剰な表現力での学習をもたらす。このように、静的に語義を表現することは述語項構造解析にとっては問題を引き起こす。

本研究では、我々は動的に語義を表現することで、先行研究が陥った問題の解決を目指す。我々は格フレームや単語埋め込みによるアプローチと同じように、アノテーションなしのコーパスを活用することを考える。ただし、先行研究のように格フレームや単語表現を事前学習するのではなく、attention mechanism[3]を用いて注目している単語に関連した用例を動的に抽出し、述語項構造解析に役立てることを考える。例えば、ある文における「作る」という述語に注目することを考える。我々の提案する手法では、まず「作る」を含んだ文を外部コーパスから収集し、収集された文に attention mechanism を用いることで注目している「作る」に関連した用例を選び出す。これにより、コーパス内でどのようにその単語が用いられているのかの用例を収集することと、今タスクを解くために必要な情報の取捨選択を実現する。同様の手続きを注目している項に対しても行い、それら選ばれた用例を用いて述語項構造を推定するのが我々のアプローチである。これにより、先行研究では難しかった動的な語義の扱いを可能にし、よりよい述語項構造解析を模索する。

2. 手法

本節では、解析対象述語に類似した用例をその類似度に応じて利用するための attention[3]を用いたモデルについて説明する。

2.1 モデルの入出力

本研究で取り組む述語項構造解析は、入力文の語系列に対し3つの格要素を返すことを目標とする。より具体的には、入力として T 語の語系列 $w_{1:T}$ と述語位置 $p \in \{1, \dots, T\}$ があり、それらに対してガ・ヲ・ニ格に対応する単語の組 (w_i, w_j, w_k) を出力する。この時、我々が取り組む述語項構造解析は次のように定義できる。語系列 $w_{1:T}$ とそれに含まれる述語が指定された時、他の単語 w_t がそれぞれ述語のガ・ヲ・ニ格になるスコアを評価する。この評価は全ての単語に対して行われるため、全体として各格ごとのスコア関数 $\text{score}_{\text{frame}} : (w_{1:T}, p) \mapsto \mathbb{R}^T$ を設計するタスクと捉える。ただし $\text{frame} \in \{\text{ga}, \text{o}, \text{ni}\}$ はそれぞれガ・ヲ・ニ格を表すものとする。また先行研究 [6] を参考に、次のような追加的なラベル $\{\text{PRED}, \text{NONE}\}$ とそのスコア関数も用いる。これは、注目している述語に格が割り振られないようするためのものであり、 $\{\text{PRED}\}$ はそれぞれの述語が注目している述語であることを、 $\{\text{NONE}\}$ は単語 w_t

が述語でも項でもないことを示すラベルである。またこれら追加的なラベル $\{\text{PRED}, \text{NONE}\}$ は評価時には用いず、 $\{\text{ga}, \text{o}, \text{ni}\}$ だけを用いて予測を行う。

2.2 提案モデル

本モデルは次の5つの部分に分けることができる。

- (1) 入力層 (embedding): 単語のベクトル表現を獲得する
- (2) 文解析層 (BiLSTM): 解析文中での各単語の表現を獲得する
- (3) 用例層 (CNN): 各用例の表現を獲得する
- (4) attention 層: 解析文中での単語表現に応じて用例の単語表現を利用する
- (5) 出力層: 5つのラベル上での確率値を出力する

2.3 入力層

入力層では単語系列 $w_{1:T} \in V^T$ と述語位置 $p \in \{1, \dots, T\}$ の情報をベクトルとして埋め込む。各単語のベクトル \mathbf{x}_t は $\mathbf{x}_t := \mathbf{x}_t^{\text{arg}} \oplus \mathbf{x}_t^{\text{mark}}$ と定義される。ここで $\mathbf{x}_t^{\text{arg}} \in \mathbb{R}^{d_{\text{word}}}$ は単語のベクトル表現であり、 $\mathbf{x}_t^{\text{mark}} \in \mathbb{R}^{d_{\text{mark}}}$ は単語 w_t が述語であるかどうかを示すフラグのベクトル表現である。

2.4 文解析層

文解析層は、入力層の $\mathbb{R}^{d_{\text{word}}+d_{\text{mark}}}$ の出力を受け取り、 n_l 層 Stacked BiLSTM を用いて出力 $\mathbf{m}_{1:T}$ を獲得する。ただし出力次元は d_o する。

2.5 用例層

用例層ではある単語 w_t の用例一つ一つの単語表現の束を獲得する。具体的には、まず $w_t \in w_{1:T}$ のそれぞれに対して、 w_t を含む文の集合 $\text{snts}(w_t) := \{\text{snt} \in \mathcal{D} | w_t \in \text{snt}\}$ を取得する。 $\text{snt} \in \text{snts}(w_t)$ における w_t のベクトル表現 $\mathbf{v}_{\text{snt}}(w_t)$ の集合 $\text{contexts}(w_t) := \{\mathbf{v}_{\text{snt}}(w_t) | \text{snt} \in \text{snts}(w_t)\}$ として用法を表現する。 $\mathbf{v}_{\text{snt}}(w_t)$ の獲得には入力層と同じ単語のベクトル表現と1層 CNN を用いた。具体的には、文 snt 中の w_t を中心とする前後 s_w 個、計 $2s_w + 1$ 個の単語を入力とした1次元 CNN の出力として $\mathbf{v}_{\text{snt}}(w_t)$ を獲得した。また CNN はフィルターサイズ $2s_w + 1$ 出力チャンネル数 $2d_o$ とする1層 CNN である。ただし、実際にある単語を含む文を全て取ってくると計算コストが高くなるため、各単語 w_t に対してその単語を含む文 $\text{snts}(w_t)$ から事前に s_s 個サンプルしておいた文を用いた。

2.6 attention 層

attention 層では、文解析層の出力に応じて用例層の出力に対して attention mechanism[3] を用いることで解析文中の用法に類似した用例を利用する。より具体的には、文解析層の出力 $\mathbf{m}_{1:T}$ と述語位置 $p \in \{1, \dots, T\}$ を入力と

し、素性列 $\phi_{1:T}$ を次のような方法で出力する。特に、述語と項ごとに attention を行う 4 つの関数

$$\phi_t := \phi^{pp} \oplus \phi^{pt} \oplus \phi^{tp} \oplus \phi^{tt}$$

$$\phi^{pp} := \text{Attention}(\mathbf{m}_p, \text{contexts}(w_p))$$

$$\phi^{pt} := \text{Attention}(\mathbf{m}_p, \text{contexts}(w_t))$$

$$\phi^{tp} := \text{Attention}(\mathbf{m}_t, \text{contexts}(w_p))$$

$$\phi^{tt} := \text{Attention}(\mathbf{m}_t, \text{contexts}(w_t))$$

を用いる。

2.7 出力層

出力層では ϕ_t にアフィン変換をかけ、得られたベクトルを用いて格を予測するためのスコアを計算する。具体的には、訓練時には 5 つのスコア ($\text{score}_{\text{ga}}(w_t)$, $\text{score}_{\text{o}}(w_t)$, $\text{score}_{\text{ni}}(w_t)$, $\text{score}_{\text{PRED}}(w_t)$, $\text{score}_{\text{NONE}}(w_t)$) を計算し、評価時には 3 つのスコア ($\text{score}_{\text{ga}}(w_t)$, $\text{score}_{\text{o}}(w_t)$, $\text{score}_{\text{ni}}(w_t)$) を計算する。これらのスコアをソフトマックス関数に渡すことで確率値に変換し、負の対数尤度を取ったものをロス関数とした。

3. 実験

本実験では、各単語に対しどれくらいの用例を集めれば良いか、さらに用例に応じて周辺の単語をどれくらい見れば良いかを検証することを目的として検証を行った。具体的には本モデルの主要なパラメータである context からのサンプル数と CNN のウィンドウ幅の変化に応じて、解析結果がどう変化するかを検証した。

3.1 実験設定

提案手法には評価用データセットとアノテーションなしテキストコーパスが用いられる。評価用データセットとして NAIST Text Corpus を、アノテーションなしテキストコーパスとして NAIST Text Corpus と同一のカテゴリである新聞のコーパスを用いた。NTC Corpus のうち 2048 文を train, 512 文ずつを dev・test として利用した。また新聞のコーパスとして、毎日新聞 91~03 年度分 (41,949,938 単語)、日経新聞 90~00 年度分 (580,558,510 単語) を mecab[2] で形態素解析し、句読点を利用して文分割を行った。

先行研究と同様に、今回用いたデータセットにおける述語との間に係り受け関係がある項 (Dep), 係り受け関係がない項 (Zero), 対応する項が文の中に存在しない格 (None) の数を図 1 に示した。

モデルのハイパーパラメータは表 4 に記載する。ただし今回の二種類の実験ではハイパーパラメータを変化させた時の挙動を確認するため、ここの実験で操作したパラメー

タは図の値に従い、それ以外の値は表 4 に従うものとする。

実験結果には、サンプル数と、ウィンドウ幅を変えた時のガ・ヲ・ニそれぞれに対して、文内 Zero・Dep・全体の F 値とそれぞれの F 値を述べた。

3.2 実験結果

まず図 2 から、基本的にサンプルサイズが大きい方が値が良いという傾向が見て取れる。さらにサンプルサイズ 4 とサンプルサイズ 8 の全体の F 値の間に大きな差がある。特にこの二つの条件の間でガ格の F 値の向上が多いことがわかる。表 1 が明らかにしている通り、ガ・ヲ・ニ格の中でガ格が最も Dep に対する Zero の項の比率が高いことから、ガ格は述語と直接係り受けをしない傾向にあり、そのため語義的知識が有効に働いていることが考えられる。一方で、それ以上サンプルサイズを増やした時には大きな精度の向上は見られない。

次に図 3 を見てみると、大きいウィンドウサイズの方が良い値を出しているが、結果にばらつきが多い。収束が上手くいっていない可能性がある。収束が比較的上手くいったと思われる 2 と 30 の数値を比較してみると、Zero のガ格の同定の精度が高くなっている一方で、Dep についてはウィンドウ幅 2 でも上手く考慮できていることが多く、特に Dep の条件の多いヲ格はウィンドウ幅によらず高い傾向にある。この傾向はサンプルサイズについても同様であり、ヲ格の特定に提案手法は効果が薄いのではないかと思われる。またニ格には大きな傾向性が見にくいだが、この理由の一つとして、ニ格がそもそもこのモデルで考慮できない None の要素が多いからであると考えられる。

4. おわりに

本研究では、日本語述語項構造解析におけるラベルなしデータからの語義知識の活用に取り組んだ。先行研究では語義知識を事前に学習してから利用する。このため、現在着目している文の情報に合わせて動的に多義性に対応するために、現在着目している単語の用法との類似性に応じて用例を利用する方法を提案した。さらに提案手法の特徴を理解するために、提案手法において重要なパラメータであるサンプルサイズと、ウィンドウ幅を調整し、その結果を分析した。

本手法がガ格など Zero の条件が多いものに効く可能性が高い一方で、None の条件が多いニ格や、そもそも直接係り受けがあり当てやすいヲ格に対しての効果が低いということが実験より明らかになった。これらの欠点には None を意味するダミーの項を追加することや、文法情報を捉える CNN や BiLSTM などにより表現力のあるモデルにすることで対処できると考えられる。そのため今後は提案手法を改善し、データ数を増やし先行研究との比較実験に取り組みたい。

表 1 3つの格と文内 Dep, Zero, None 条件下での存在個数

	train			dev			test		
	Dep	Zero	None	Dep	Zero	None	Dep	Zero	None
ga	1023	387	557	277	114	102	275	101	116
o	733	55	1187	204	14	279	201	14	286
ni	329	50	1632	82	15	410	85	14	410

表 2 サンプルサイズを変えた時の格ごとの文内 Dep・Zero・全体条件下での F 値

サンプルサイズ	文内 Zero				文内 Dep				全体			
	ガ格	ヲ格	二格	All	ガ格	ヲ格	二格	All	ガ格	ヲ格	二格	All
2	0.198	0.071	0.357	0.102	0.465	0.607	0.388	0.576	0.341	0.350	0.124	0.284
4	0.188	0.143	0.357	0.166	0.353	0.736	0.471	0.699	0.267	0.425	0.150	0.286
8	0.317	0.143	0.429	0.174	0.538	0.711	0.588	0.696	0.412	0.409	0.182	0.348
12	0.257	0.071	0.429	0.109	0.567	0.736	0.541	0.710	0.423	0.425	0.173	0.355
16	0.356	0.071	0.500	0.117	0.593	0.697	0.553	0.678	0.456	0.407	0.182	0.365
64	0.267	0.214	0.500	0.245	0.473	0.731	0.506	0.700	0.368	0.425	0.169	0.332
128	0.277	0.143	0.429	0.174	0.571	0.746	0.553	0.720	0.434	0.434	0.173	0.362

表 3 ウィンドウ幅を変えた時の格ごとの文内 Dep・Zero・全体条件下での F 値

ウィンドウ幅	文内 Zero				文内 Dep				全体			
	ガ格	ヲ格	二格	All	ガ格	ヲ格	二格	All	ガ格	ヲ格	二格	All
2	0.218	0.214	0.500	0.245	0.495	0.751	0.565	0.726	0.374	0.442	0.182	0.344
5	0.218	0.071	0.500	0.117	0.400	0.736	0.541	0.710	0.304	0.425	0.176	0.308
8	0.307	0.214	0.429	0.237	0.538	0.721	0.341	0.666	0.407	0.423	0.117	0.333
10	0.267	0.214	0.500	0.245	0.473	0.731	0.506	0.700	0.368	0.425	0.169	0.332
20	0.287	0.214	0.357	0.229	0.462	0.687	0.212	0.618	0.363	0.404	0.078	0.299
30	0.426	0.071	0.571	0.125	0.644	0.766	0.729	0.764	0.513	0.447	0.235	0.415

意味	記号	値
単語埋め込み次元	d_w	64
述語記号埋め込み次元	d_p	16
BiLSTM 出力次元	d_o	16
BiLSTM 層数	n_l	2
CNN window size	s_w	10
entity のサンプルサイズ	n_{ent}	64
ミニバッチサイズ	s_m	8
最適化アルゴリズム		Adam
学習率	α	0.001
Stacked BiLSTM の dropout rate	r_d	0.25

表 4 基本的なハイパーパラメータ

- Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.
- [6] 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)*, Vol. 1, pp. 1591–1600, 2017.
- [7] Ryohei Sasano, Daisuke Kawahara, and Sadao Kurohashi. A fully-lexicalized probabilistic model for Japanese zero anaphora resolution. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pp. 769–776, 2008.
- [8] Ryohei Sasano and Sadao Kurohashi. A discriminative approach to Japanese zero anaphora resolution with large-scale lexicalized case frames. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pp. 758–766, 2011.

参考文献

- [1] Daisuke Kawahara and Sadao Kurohashi. Fertilization of case frame dictionary for robust Japanese case analysis. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pp. 1–7. Association for Computational Linguistics, 2002.
- [2] Taku Kudo. Mecab: Yet another part-of-speech and morphological analyzer. <http://mecab.sourceforge.jp>, 2006.
- [3] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.
- [4] Yuichiroh Matsubayashi and Kentaro Inui. Revisiting the design issues of local models for Japanese predicate-argument structure analysis. *arXiv preprint arXiv:1710.04437*, 2017.
- [5] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey