同一指示 PRO を考慮した空範疇検出の性能分析

竹野 峻輔 † 永田 昌明 ‡ 山本 和英 †
† 長岡技術科学大学
‡ NTT コミュニケーション科学基礎研究所
†{takeno, yamamoto}@jnlp.org, ‡nagata.masaaki@ntt.co.jp

1 はじめに

日本語の文書では主語などの省略現象が頻繁に発生する．これらは質問応答, 自動要約, 機械翻訳といった言語処理の応用タスク上に大きく影響を及ぼすことから, 省略検出を高精度を達成することは日本語の言語処理の基礎を築く上で重要である．しかしながら, 現状では実用に足るだけの十分な性能を達していない．

従来の日本語の省略検出は, 述語項構造解析の部分問題として, 省略検出とその照応先の同定を同時にすることが多い [8,9]．他方で, 日本語と同様に文中の省略が多く発生する中国語では, 省略検出は句構造構文解析の部分問題として取り扱われることが多い．この問題は空範疇検出と呼ばれ, 研究が進められている [7]．

中国語と同様に日本語でも, 欅ツリーバンク [2] と呼ばれる日本語句構造ツリーバンクのの開発に伴い, 句構造構文解析の部分問題として, 照応先の同定を伴わずとも省略検出に取り組めるようになった．我々の先行研究 [6] においては, 欅ツリーバンク中に含まれるゼロ代名詞 pro(small pro) と名詞句の移動の痕跡を表す T(trace) に関する空範疇検出手法の提案を行った．

この先行研究の問題点に, 空範疇の同一指示ゼロ代名詞 PRO(big pro) の検出性能の評価を行うことができていない点がある．これは欅ツリーバンクのアノテーション方針から PRO が非明示的であるためである．その同定方法については, Butler ら [1] により既に提案されている．そこで我々は, 同一指示ゼロ代名詞 PRO を考慮した日本語の空範疇検出の性能を再評価した．これに加え, 現存の述語項構造解析器と我々の空範疇検出手法についてその省略検出性能の比較・分析したのでこれを報告する．

2 句構文解析における省略検出手法

本節では, 我々の先行研究で報告した空範疇検出手法 [6] について 説明する．

欅ツリーバンク中の文は平坦な句構造で表され, 空範疇は IP ノード直下の子ノードとして, 文法機能 (SBJ, OB1, OB2 など) を伴って, 空範疇の種類 (pro, PRO, T) が終端記号に付与される．これら空範疇はその種類・文法機能・位置により一意に定まる．従って, これらで符号化を行い, 直上の IP ノードのラベルとすることで, 空範疇検出は, 句構造木中の IP ノードに対するラベル分類問題として取り扱える．

Xiang らの確率モデル [7] を参考に, 本手法は空範疇検出を以下の対数線形モデルの確率モデルで定式化する．

$$P(e_1^n|T) = \prod_{i=1}^{n} P(e_i|e_1^{i-1}, T) = \prod_{i=1}^{n} \frac{\exp(\boldsymbol{\theta} \cdot \boldsymbol{\phi}(e_i, e_1^{i-1}, T))}{Z(e_1^{i-1}, T)}$$

ここで $\boldsymbol{\phi}$ は素性ベクトル, $\boldsymbol{\phi}$ への $\boldsymbol{\theta}$ は重みベクトル,Z は, 以下で求められる正規化係数を表す．

$$Z(e_1^{i-1}, T) = \sum_{e \in \mathcal{E}} \exp(\boldsymbol{\theta} \cdot \boldsymbol{\phi}(e_1^{i-1}, T, e))$$

$\mathcal{E}$ は空範疇がないことを表す空ラベルと符号化された空範疇ラベル全てを合わせた集合を表す．また, 句構造木の根からの行きがけ順 (preorder) にノードを巡回した系列を $T = t_1 t_2 \cdots t_n$ とし,t_i に関連付けられた空範疇ラベルを e_i としている．

2.1 パス素性を利用した素性設計

我々の手法では, 句構造木中のノードを行きがけ順に巡回し, 各 IP ノードに対しその空範疇を推定する．IP ノードごとに主辞素性・子ノード素性・空範疇素性の 3 つの素性とパス素性を抽出し, これらを組み合わせたものを素性として利用する．本節では, この推定に利用するそれぞれの素性について説明をする．

パス素性は, 現在のノードから, 根ノードまたは CP ノードまでの祖先のノードに対する非終端記号ラベルの系列の集合を表す (i.e. IP-MAT, IP-MAT → PP, IP-MAT → PP → NP etc ...)．主辞単語素性は, 現在のノードの語彙的主辞となる単語の表層形を表す．子ノード素性は, 現在のノードの子ノードのラベルの集合である．ただし, 子ノードの右端に機能語が来る場合にはラベルに機能語を付加する (i.e. PP-は, PP-が etc...)．空範疇素性は, 現在のノードの祖先となる IP ノードで検出された空範疇の種類である．

我々の手法では, このように取り出された主辞素性, 子ノード素性, 空範疇素性とパス素性と組み合わせた

表現 (i.e. [IP-MAT → PP] × [PP-は] etc...) を各 IP ノードの素性として抽出し, 対数線形モデルでその重みを学習する.

2.2 分散表現による格フレーム辞書の近似

格フレーム辞書は, 述語が取り得る項の情報を与えるため, 日本語の省略検出においては明らかに有効である. しかし, 処理対象となる領域において, 大規模かつ網羅的な格フレーム辞書を予め用意することは難しい. そこで, 我々は日本語のように格関係を明示する機能語 (格助詞) をもつ言語において利用できる, 分散表現を用いた格フレーム辞書の近似表現を提案した.

Pennington ら [5] が提案する分散表現の GloVe は, 二つの分散表現の内積がそれらの共起頻度の対数を近似するように設計されている. このことから, 述語の格フレームを近似する素性を簡単に作ることができる. 述語の分散表現を w_i,N 個の格助詞 (および格助詞に相当する単語) の分散表現を $q_1, q_2, \cdots, q_N \in Q$ とすれば, この述語に関する, 分散表現による格フレーム近似は, 述語とそれぞれの格助詞の内積を正規化した N 次元のベクトルとして以下で定義できる.

$$\tilde{v_i} = w_i \cdot (q_1, q_2, \cdots, q_N)$$
$$v_i = \frac{\tilde{v_i}}{||\tilde{v_i}||}$$

提案手法では, 集合 Q として高頻度の格助詞および格助詞に相当する単語である { が, は, も, の, を, に, へ } を用いた.

3 PRO を含めた空範疇検出の評価

先行研究 [6] と同様の実験設定にて, 手法の空範疇検出の性能を評価する. ただし訓練および開発、評価に用いる欅ツリーバンクは 2015 年 11 月 10 日のものを利用する. 欅ツリーバンクでは, pro の再分類として hearer や speaker などがあるが, 単純化のため, あらかじめ pro 系統の空範疇は pro に変換しておく. このデータセットに対して, Butler ら [1] らが提案する手法を用いて同一指示ゼロ代名詞 PRO を同定したものを実際の訓練および開発、評価に用いる. 表 1 に利用したデータセットの統計情報を示す.

開発セットおよび評価セットには, 1995 年の毎日新聞コーパス (newswire) 217 文, KNB コーパス (blog) 611 文, 2002 年の CIAIR の話し言葉コーパス (transcript) 1,182 文をそれぞれランダムに等分割し, 一方を開発セット, 他方を評価セットとした. そして残りのデータを訓練セットとしてモデルの学習を行った.

実験設定としては, 入力が空範疇が含まれない理想的な句構造木である場合 (GOLDEN) と句構造構文解析器からの出力である場合 (SYSTEM) の 2 つの条件のもと実験を行う. SYSTEM 条件下で利用する句構造構文解析器には, 欅ツリーバンクを BerkleyParser に学習させた HARUNIWA[1] を用いる.

表 1: 欅ツリーバンク (2015 年 11 月 10 日時点)

		全て	開発/評価セット		
			blog	newswire	transcript
#PRO	SBJ	10,082	104	339	76
	OB1	206	1	1	0
	OB2	7	0	0	0
#T	SBJ	5,231	41	269	6
	OB1	1,032	11	36	0
	OB2	16	1	2	0
	other	466	9	21	15
#pro	SBJ	17,716	188	345	588
	OB1	2,156	1	27	43
	OB2	59	0	0	2
	other	14	1	1	0
#IP node		61,619	534	1,800	1,105
#sent.		30,872	217	611	1,182

我々の手法 (OURS) に対し, 比較手法には HARUNIWA に付属のルールベース空範疇付与器 (RULE) に加え, PRO, T に向けた構造パターンマッチ手法の Johnson の手法 [4], 中国語における句構造空範疇検出性能で最高性能を誇る Xiang の手法 [7] の手法, それぞれを欅ツリーバンク向けに修正を加えたものを採用する.

評価は先行研究と同様に予測された空範疇の位置・種類・文法機能をもとに適合率・再現率を計算し, F 値にて評価する.

得られた結果を表 2 に示す. 表 2 より GOLDEN 条件下では我々の手法は 78.1%を記録し既存手法よりも 2.4 ポイント高い. しかし, 空範疇検出の性能は句構造構文解析器の性能に強く依存しており, SYSTEM 条件下では性能が大きく低下し, F 値は 53.4%となっている. 以上から先行研究 [6] の報告同様, 我々の手法の有効性を確認することができた.

4 省略検出手法の性能比較

本節では, 我々の日本語空範疇検出手法と現存の日本語述語項構造解析器について, その省略検出性能について性能比較を行う.

代表的な述語項構造解析器 SynCha[2], KNP[3] では, 省略検出と照応解析を同時に解析し述語項の同定を行うている. SynCha の省略検出では 省略検出モデル, 文内先行詞同定モデル, 文間先行詞同定モデル等の出力結果を整数計画法に基づいて最適化することで最終的な省略検出の結果を得ている. KNP は大規模格フレームを利用した識別モデルの格同定により, 省略検出を行っている. これらの解析器においては直接係り受けにない項をゼロ照応とし, 文内ゼロ照応 (INTRA_Z)

[1] http://www.compling.jp/haruniwa/
[2] http://www.cl.cs.titech.ac.jp/ ryu-i/syncha/
[3] http://nlp.ist.i.kyoto-u.ac.jp/?KNP

表 2: PRO を含めた空範疇検出手法の比較 (F 値 [%])

空範疇	PRO		T				pro			ave.
文法機能	SBJ	OB1	SBJ	OB1	OB2	other	SBJ	OB1	OB2	
GOLDEN										
RULE	78.2	66.7	90.2	44.9	0.0	0.0	75.4	0.0	0.0	75.4
modified (Jonson,2002)	30.2	0.0	42.4	33.3	0.0	0.0	46.3	0.0	0.0	44.8
modified (Xiang et al.,2013)	77.8	100.0	89.9	53.7	0.0	0.0	75.3	0.0	0.0	75.7
OURS	74.1	100.0	91.7	50.0	0.0	4.76	79.2	0.0	0.0	**78.1**
SYSTEM										
RULE	28.6	33.2	45.9	26.1	0.0	0.0	55.3	0.0	0.0	45.6
modified (Jonson,2002)	16.3	0.0	35.8	33.3	0.0	0.0	42.3	0.0	0.0	33.2
modified (Xiang et al.,2013)	36.3	100.0	46.0	32.6	0.0	0.0	62.1	0.0	0.0	51.5
OURS	38.5	100.0	45.6	50.0	0.0	0.0	65.1	0.0	0.0	**53.4**

および文外ゼロ照応 (INTER_Z) を省略検出の対象として取り扱っている.

表 3: 構築した共通データ

	共通	NTC	KTB
ガ格/SBJ	578	1,399	951
ヲ格/OB1	80	374	59
二格/OB2	2	222	2
総数	660	1,995	1,012

表 4: 省略アノテーションの対応関係

		PRO	T	pro
ガ格/SBJ	INTRA_Z	122	142	67
	INTER_Z	82	23	142
ヲ格/OB1	INTRA_Z	0	61	5
	INTER_Z	0	1	13
ニ格/OB2	INTRA_Z	0	2	0

既存の述語項構造解析器である SynCha や KNP と, 我々の提案手法の比較を行うためには, 前提とする文構造や単語単位の問題が異なるため直接の比較が難しい. しかしながら, 我々は,NAIST テキストコーパス v1.5 [3] と欅ツリーバンクで共通する毎日新聞コーパス 593 文の省略に関する共通のアノテーションを機械的に抽出し, これらのデータについて、それぞれの解析器の性能を測定することで省略解析の性能について比較を行う. 本稿において, この対応関係の抽出方法については紙面の都合より詳細を省略する. このデータセットは, 異なる解析器の性能を比較できるよう, 単語単位の差異や受け身における格交替などのアノテーションの差異の吸収を行ったものである. そして, 述語に対し付与されている省略表現を NAIST テキストコーパスと欅ツリーバンク それぞれをガ格/SBJ・ヲ格/OB1・ニ格/OB2 とまとめたものである. これらのデータセットは Web 上にて公開予定である.

評価データ中の NAIST テキストコーパスの省略関係 (文外照応:INTER_Z, 文内照応:INTRA_Z) と欅ツリーバンク中の省略要素 (PRO, T, pro) の抽出結果を表 3 と表 4 に示す.

表 3 は NAIST テキストコーパス (NTC) と欅ツリーバンク (KTB), それぞれの省略アノテーション数と共通データの比較を表す. 表 4 は共通データの省略の対応を表す. ヲ格/OB1 のデータ数が実際の KTB の省略要素数より多いが, これはコーパス間での受動表現の取り扱いの差異を吸収のため表記の数になる.

作成した共通データセットを用いて, これら手法について評価を行う. 評価軸として次を考えた.

評価 A : 文中の省略要素が検出できるか否か
評価 B : 検出した省略要素の型まで含め判別可能か

評価 A では 特定の述語の格関係について省略要素を持つか否かについてのみ評価する. つまり, 述語項構造解析器出力を評価するの場合はガ格・ヲ格について直接係り受け関係か否かのみを確認し, 文外照応と文内照応の区別はつけない, 空範疇検出器の評価の場合には, IP ノード中の述語に対する主格 (SBJ), 目的格 (OB1) の項について空範疇の有無のみを評価し, 空範疇の種類については区別をしない.

評価 B では 特定の述語の格関係について特定の省略要素を持つか否かを評価する. つまり, 述語項構造解析器の評価の場合は, ガ格・ヲ格について直接係り受け関係, 文内照応 (INTRA_Z), 文外照応 (INTER_Z; 外界照応を含む) のみを評価し, 照応先の正否は問わない. 空範疇検出器の評価の場合には IP ノード中の述語に対する主格・目的格について, それぞれで空範疇の種類が正しく付与されているかを評価する.

述語項構造解析器 SynCha と KNP および我々の空範疇検出手法 (OURS) の省略検出の性能比較の結果を表 4 に示す. 表 4 では, 我々の手法 OURS については前節の実験と同様の GOLDEN と SYSTEM の条件のもと評価した結果を示している.

省略解析に焦点を当てた場合, 既存の述語項構造解析器 SynCha, KNP よりも句構造空範疇検出器 OURS(SYSTEM) の方が, ガ格/SBJ に関して評価 A,B の両方で上回る結果となった. ヲ格/OB1 にお

表 5: 共通データにおける省略性能の比較 [%]. () 内の数字が省略の有無のみを判定する評価 A の結果.

ガ格/主格

手法	SynCha		KNP		OURS (SYSTEM)		OURS (GOLDEN)	
KTB/NTC ※	INTER_Z	INTRA_Z	INTER_Z	INTRA_Z	INTER_Z	INTRA_Z	INTER_Z	INTRA_Z
PRO	15.9(39.0)	45.9 (52.4)	22.0 (50.0)	54.1 (70.5)	59.8 (79.3)	63.9 (77.9)	76.8(96.3)	81.1(93.4)
T	8.7(26.1)	28.2 (43.5)	21.7 (30.4)	69.0 (73.9)	60.9 (91.3)	73.2 (88.0)	91.3(100.0)	96.5(97.2)
pro	8.45(19.7)	40.3 (47.8)	16.9 (19.0)	38.8 (50.7)	48.6 (71.1)	29.9 (49.3)	61.3(79.6)	46.3(68.7)
ave.	25.9(36.4)		40.1(54.3)		57.8(76.0)		75.7(88.6)	

ヲ格/目的格

KTB/NTC ※	INTER_Z	INTRA_Z	INTER_Z	INTRA_Z	INTER_Z	INTRA_Z	INTER_Z	INTRA_Z
T	0.0(0.0)	0.0(0.0)	0.0 (0.0)	18.0(0.0)	0.0(0.0)	16.4(16.4)	0.0(0.0)	72.1(72.1)
pro	0.0(0.0)	0.0(0.0)	15.4(15.4)	0.0(0.0)	0.0(0.0)	0.0(0.0)	23.1(23.1)	20.0(20.0)
ave.	0.0(0.0)		17.1(17.1)		12.2(12.2)		58.5(58.5)	

※ KTB:欅ツリーバンクと NTC:NAIST テキストコーパス, それぞれ省略アノテーションの分類項目

いては, 構造解析を含めた評価では KNP が最も性能が良い結果となった.

述語項構造解析の結果において, 文外照応 (INTER_Z) となるような省略アノテーションに関する部分について SynCha と KNP は共通して性能が低いのに対し, OURS では比較的高い性能を保持していることがわかる. 特に文内に照応先が存在しないという点でアノテーションが一致している pro と INTER_Z のおいては句構造空範疇検出器 OURS は, 48.6%と既存のツールよりもその性能は高い.

また文外照応 (INTER_Z) と同一指示ゼロ代名詞 PRO の省略が対応する点は, 複数の述語が項を共有しており, さらに PRO の指示先が文外となる場合を表す. そのような省略は, 表 4 より欅ツリーバンクから見ると PRO の 40%, NAIST テキストコーパスから見ると文外ゼロ照応の 33%の数を占めており, 省略検出性能全体に対する影響は決して小さくない. このような場合, 述語項構造解析器は, 独立に述語項構造を同定する. そこで複数の述語が項を共有することを検出することで性能の改善することが報告されている [8, 9]. 一方で, 我々の句構造空範疇検出器では, この問題をアノテーションレベルで解決していることがわかる. その検出性能も, OURS(SYSTEM) では 59.8%と比較的高い性能を記録している.

OURS の GOLDEN と SYSTEM の結果を比較すると, 句構造空範疇検出器の精度は, 句構造構文解析の精度に強く依存することが確認できた. 構文解析の精度向上および空範疇検出の頑健性の向上が今後の課題である.

5 おわりに

我々は欅ツリーバンクに同一指示ゼロ代名詞 PRO を考慮した空範疇検出の性能について再度評価を行った. また, 現存の述語項構造解析器と我々の手法について省略検出の性能に関する比較を試みた. 今後は, NAIST テキストコーパスと欅ツリーバンクという全く異なるアノテーション基準のもとで作成された言語資源の対応関係について, 更に分析を進め, 省略検出に関する, より厳密な精度評価を実現したい。

参考文献

[1] Alastair Butler, Shota Hhiyama, and Kei Yoshimoto. Coindexed null elements for a Japanese parsed corpus. In *Proceedings of the 21th Annual Meeting of the Association for Natural Language Processing*, pp. 708–711, 2015.

[2] Alastair Butler, Tomoko Hotta, Ruriko Otomo, Kei Yoshimoto, Zhen Zhou, and Hong Zhu. Keyaki Treebank : phrase structure with functional information for Japanese. In *Text Annotation Workshop*, 2012.

[3] Ryu Iida, Mamoru Komachi, Naoya Inoue, and Inui Kentaro. Annotating Predicate-Argument Relations and Anaphoric Relations: Findings from the Building of the NAIST Text Corpus. *Journal of Natural Language Processing*, Vol. 17, No. 2, pp. 1–26, 2010.

[4] Mark Johnson. A simple pattern-matching algorithm for recovering empty nodes and their antecedents. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pp. 136–143, 2002.

[5] Jeffrey Pennington, Richard Socher, and Christopher D Manning. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pp. 1532–1543, 2014.

[6] Takeno Shunsuke, Nagata Masaaki, Yamamoto Kazuhide, Shunsuke Takeno, Masaaki Nagata, and Kazuhide Yamamoto. Empty Category Detection using Path Features and Distributed Case Frames. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1335–1340, 2015.

[7] Bing Xiang, Xiaoqiang Luo, and Bowen Zhou. Enlisting the ghost: Modeling empty categories for machine translation. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 822–831, 2013.

[8] 大村舞, 進藤裕之, 松本裕治. 複数の述語間関係を考慮した日本語述語項構造解析. 言語処理学会 第 21 回年次大会 発表論文集, pp. 67–70, 2015.

[9] 大内啓樹, 進藤裕之, Duh Kevin, 松本裕治. 複数の述語項構造の同時解析手法に関する調査. 言語処理学会 第 21 回年次大会 発表論文集, pp. 289–292, 2015.