1K3-2

日本語の単語依存構造解析のための長単位解析

Japanese Long Unit Word analysis for dependency parsing

田中貴秋 Takaaki Tanaka　林克彦 Katsuhiko Hayashi　永田昌明 Masaaki Nagata

NTT コミュニケーション科学基礎研究所
NTT Communication Science Laboratories

We present a shift-reduce parsing method of jointly deciding Long Unit Word (LUW) chunks and dependency structures, which are usually processed in two separate steps. LUW-based analysis facilitates capturing syntactic structures and makes parsing results more precise than Short Unit Word (SUW)-based analysis. Moreover, the joint analysis improves the performance of identifying major syntactic relations.

1. はじめに

日本語の構文解析は，文節依存構造 (係り受け) 解析として行われることが多いが，Universal Dependencies (以下 UD) [Nivre 15, 金山 15] のような言語横断的な言語アノテーションを考えると，適用可能な言語が限定される文節単位の構造よりも，単語単位の依存構造の方が扱いやすい．

一方で,「単語」という概念自体は明確な定義を行うことは容易ではなく，対象言語の特性にある程度依存することは避けられない．例えば，英語などのように基本的な単位（本稿ではトークンと呼ぶ）を空白で区切って記述する言語においては，便宜上トークンを単語と同一視する場合が多いが，複数のトークンで一つの統語機能を担う multiword の扱いを考慮する必要がある．UD では，一つのトークンを依存構造の単位として扱い，multiword を構成するトークン間には，*mwe* や *compound* などの関係タイプを用いて multiword の依存構造を構成する．

日本語や中国語等のように文が空白で区切られていない言語の場合は，何をトークンとするかという問題から考える必要がある．トークンへの分割の仕方としては，形態素解析器の使用する辞書等の体系によって，様々な基準が用いられている．このトークン分割の要件としては，出来るだけ区切りの揺れが少なく曖昧性の少ない定義が望まれる．日本語では，現代日本語書き言葉均衡コーパス (BCCWJ) [Maekawa 14] に採用されている短単位 (Short Unit Word, SUW) がその要件を満たしていると考えられる．この短単位は，解析用辞書である UniDic [*1] の見出し語として登録されているものと一致し，MeCab 等の形態素解析結果として，利用できることが大きな利点である．ただし，短単位は，統語的機能を担う単位として扱うには細かすぎる傾向があるため，本稿では，同じく BCCWJ で採用されている長単位 (Long Unit Word, LUW) を依存構造の基本単位とする．長単位は，文節を内容語と機能語に分割したそれぞれを構成要素とするもので，まとまった統語的機能をもつ単位として，短単位よりも扱いやすい．これは，UD でのトークンを短単位，*mwe* や *compound* で構成される multiword を長単位と対応させることに近いと考えられる．

長単位を基本単位として依存構造を構成するためには，長単位を同定する長単位解析が必要となる．短単位への分割（形態素解析）– 長単位の同定（長単位解析）– 依存構造解析 というパイプライン処理が，最も単純な解析方法になるが，本来長単位は統語構造を考慮して同定されることが望ましいと考えらえるため，本稿では長単位解析と依存構造解析を同時に行う手法について提案する．長単位解析と依存構造解析をパイプライン処理として行った場合や，短単位の依存構造を直接解析した場合との比較についても述べる．

連絡先: 田中貴秋，NTT コミュニケーション科学基礎研究所, 京都府相楽郡精華町光台 2-4, tanaka.takaaki@lab.ntt.co.jp

*1 http://download.unidic.org/

2. 短単位と長単位

BCCWJ では，長単位と短単位という 2 種類の定義からなる階層的な単語単位を採用している．これらの大雑把な説明としては，短単位は，形態素情報を保持する最小単位で，UniDic の見出し語とほぼ一致しており，一方，長単位は文節を構成する基本単位で，文節は 1 つの内容語の長単位と 0 個以上の機能語の長単位から構成される．

図 1 は，長単位と短単位による単語分割の例である．内容語の長単位には複合名詞「予備調査結果」，機能語の長単位には複合辞「について」が含まれる．また，短単位の品詞は語彙として単一的に定義された品詞が与えられるが，長単位の品詞には，文脈における用法に従った品詞が与えらえる．例えば，「昨日」の短単位品詞は NN（BCCWJ の品詞は,「名詞-副詞可能」）[*2] であるが，長単位品詞は文中での用法に従って ADV（BCCWJ の品詞は「副詞-一般」）である．

3. 日本語の単語依存構造

日本語において単語間の依存構造を扱った研究は多くないが，Mori らの短単位間依存構造 [Mori 14]，Uchimoto らの短単位間（文節内）依存構造 [Uchimoto 08]，Tanaka らの長単位依存構造 [Tanaka 15]，UD の日本語版 [金山 15] などがある．Mori らは短単位を基本単位として，さらに活用語の語尾を分割して依存構造を定義している．この依存構造では関係ラベルを定義していないが，短単位（あるいはさらに分割した単位）は統語的機能を表す単位としては細かいため，そのまま関係ラベルを付加するのは難しいと考えられる．Uchimoto らは，日本語話し言葉コーパス (CSJ) に現れる言いよどみや言い直しなどをアノテーションするために，文節内の短単位間の依存構造を定義している．文節の境界を保持した依存構造であ

*2 本稿では，BCCWJ の品詞の第 1 階層に基づいて定義した 23 種類の品詞タグを使用している

短単位 (SUW)	昨日 NN B	予備 NN B	調査 NN I	結果 NN Ia	に PCS B	つい VB I	て PCJ I	も PBD Ba	報告 NN B	し VB Ia	た AUX Ba
↓											
長単位 (LUW)	昨日 ADV	予備調査結果 NN			について PCS			も PBD	報告し VB		た AUX

図 1: 短単位，長単位の分割と，小澤らの長単位解析手法で用いるタグ (B,I,Ba,Ia) の例．品詞タグは，NN(名詞)，VB(動詞), ADV(副詞), PCS(格助詞), PBD(係助詞), PCS(接続助詞)，AUX(助動詞)．

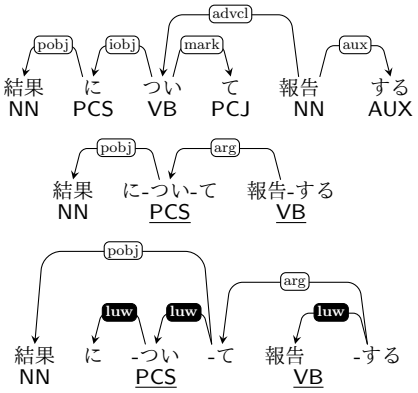

図 2: 短単位による依存構造 (上) と長単位による依存構造 (中)，長単位の依存構造を短単位に分割した構造 (下) の例．

るので，統語的機能を表す関係ラベルを付与するには必ずしも適していない．

一方，前述した長単位は，統語的機能として一つの単位である複合名詞や複合辞等がまとまる傾向があり，統語構造の基本単位として扱いやすいと考えられる．図 2 は，短単位ベースと長単位ベースの依存構造の違いを示している．短単位の依存構造では，複合辞「について」に短単位の動詞「つい」が含まれるため，冗長な構造を構成しているのに対し，長単位の依存構造では,「について」をフラットな構造として一つの助詞と同様に扱っているために，本動詞である「報告する」と「結果」の関係が明確に表示されている．

UD 日本語版は現時点の仕様 (ver. 1.2) では，短単位を依存構造の基本単位として採用しているが，複合辞等一部の複合表現については長単位を採用する考えもあり，*mwe*, *compound* 等で間接的に長単位相当の単位がアノテーションされている．

長単位解析と依存構造解析をパイプライン的に行う場合には，入力の短単位列から長単位解析が長単位列を同定し，その長単位列を依存構造解析の入力として，図 2 の中段のような構造を作る．長単位解析と依存構造解析を同時に行う場合には，短単位列を入力として，図 2 の下段のような構造を作る．「について」や「報告する」等の複数の短単位からなる長単位は，特別な関係ラベル *luw* を使ったフラットな構造で表し，長単位間をそれ以外の関係ラベルを持つ依存関係で構成する構造となる．

4. 解析手法

4.1 系列ラベリングによる長単位解析

小澤らは，短単位列を入力として長単位に分割するチャンキングモデルと，分割された長単位に対して長単位品詞を推定するカテゴリ推定モデルを構築して，2 段階で解析する手法を提案している [小澤 14]．手法の概要は以下の通りである．

1. 与えられた短単位列について，長単位の先頭の短単位 “B” タグ，長単位の先頭以外の短単位 “I” タグを同定する系列ラベリングを行う．ただし，単独で長単位を構成しかつ長単位品詞が短単位品詞 *3 と一致する短単位には “Ba” タグを，複数短単位から構成される長単位の末尾かつ長単位品詞が短単位と一致する短単位には，“Ia” タグが与えられる，
2. 前手順で，“Ba” タグあるいは “Ia” タグのついた短単位を持つ長単位は，その短単位の品詞を長単位品詞とする．“Ba” タグも “Ia” タグも含まない長単位に対しては，前後 1 長単位の情報から構築されたカテゴリ推定モデルを用いて，対象の長単位の品詞を推定する．ただし,「助詞」「助動詞」は，別に設けた複合辞テーブルと表層が一致しない限り，候補の品詞に含めない．

図 1 の例では，短単位で “Ia” タグのついた短単位「結果」を持つ長単位「予備調査結果」は短単位「結果」の品詞 NN をそのまま引き継いで長単位の品詞としている．一方，長単位「昨日」と「について」を構成する短単位には，“Ba” タグあるいは “Ia” タグを持つものが存在せず，元の短単位とは別の品詞を長単位に付与する必要があることを示している．この 2 つの長単位に対しては，2 のカテゴリ推定によって，それぞれ長単位品詞 ADV，PCS が付与されることになる．

4.2 長単位と依存構造の同時解析（提案手法）

短単位を入力として，長単位のチャンキングと長単位間の依存構造を同時に解析することにより依存構造と整合する長単位解析を行うことを考える．解析アルゴリズムは，Shift-reduce による解析をベースとして，長単位解析のための操作と，依存構造解析のための操作を定義し，各状態において，どの操作を行うかを決定することにより解析を行う．これは，中国語における同時解析手法 [Zhang 14, Hatori 12] と類似した方法である．

長単位解析に関するに関する操作は次の通りである．以下，stack を $S = (s_0, s_1, ...)$，queue を $Q = (q_0, q_1, ...)$ とする．初期状態では，S は空，Q は入力文の短単位列である．

ShiftLUW(*pos*) q_0 (SUW) を取り出し，S の最上位 s_0 に入れる．この SUW を先頭として構成される LUW*4 の品詞を *pos* と推定する．この操作は，s_0 が LUW のときのみ実行可能である．

ShiftSUW q_0 (SUW) を取り出し，S の最上位 s_0 に入れる．この SUW はすでに s_0 に存在している SUW の部分木に後続して LUW を構成する要素となる．この操作は，s_0 が LUW として確定されていない SUW(から構成される部分木) であるときのみ実行可能である．

*3 元の手法では活用型，活用形を考慮するが本稿では品詞タグのみを対象とする．

*4 SUW 単独から構成される場合も含まれる

Step	Action	Stack				Queue		
		s_3	s_2	s_1	s_0	q_0	q_1	q_2
0	–				ϕ	昨日	予備	調査
1	ShiftLUW(ADV)				昨日	予備	調査	結果
2	PopLUW				昨日/ADV	予備	調査	結果
3	ShiftLUW (NN)			昨日/ADV	予備	調査	結果	に
4	ShiftSUW		昨日/ADV	予備	調査	結果	に	つい
5	ReduceSUW$_L$			昨日/ADV	予備←調査	結果	に	つい
6	ShiftSUW		昨日/ADV	予備←調査	結果	に	つい	て
7	ReduceSUW$_L$			昨日/ADV	予備←調査←結果	に	つい	て
8	PopLUW			昨日/ADV	予備←調査←結果/NN	に	つい	て
9	ShiftLUW(PCS)		昨日/ADV	予備←調査←結果/NN	に	つい	て	報告
10	ShiftSUW	昨日/ADV	予備←調査←結果/NN	に	つい	て	報告	し
11	ReduceSUW		昨日/ADV	予備←調査←結果/NN	に←つい	て	報告	し
12	ShiftSUW	昨日/ADV	予備←調査←結果/NN	に←つい	て	報告	し	た
13	ReduceSUW		昨日/ADV	予備←調査←結果/NN	に←つい←て	報告	し	た
14	PopLUW		昨日/ADV	予備←調査←結果/NN	に←つい←て/PCS	報告	し	た
15	ReduceLUW$_L$($pobj$)			昨日/ADV	に←つい←て/PCS	報告	し	た
				予備←調査←結果/NN$\swarrow^{pobj}$				
…				…				

図 3: 長単位解析と依存構造解析の同時解析例．“/” の右側の品詞は確定した長単位に付与された長単位品詞，←は長単位内部の短単位間の弧，↙ は長単位間の弧を表す．

長単位素性	
$s_0.h_{Lw} \circ s_1.h_{Lw}$	$s_0.h_{Lt} \circ s_1.h_{Lw}$
$s_0.h_{Lt} \circ s_1.h_{Lt} \circ s_0.l_{Lt}$	$s_0.h_{Lt} \circ s_1.h_{Lt} \circ s_0.r_{Lt}$
$s_0.h_{Lw} \circ s_1.h_{Lw} \circ s_2.h_{Lw}$	$s_0.h_{Lt} \circ s_1.h_{Lt} \circ s_2.h_{Lt}$
$s_0.h_{Lt} \circ s_1.h_{Lt} \circ q_0.t$	$s_0.h_{Lw} \circ s_1.h_{Lt} \circ q_0.t$
$s_0.h_{Lt} \circ s_1.h_{Lw} \circ q_0.t$	$s_0.h_{Lw} \circ s_1.h_{Lw} \circ q_0.w$
複合辞素性	
$q_0.f_{comp} \circ s_0.h_{\cdot Lt}$	$q_0.f_{comp} \circ s_0.h_{Lt} \circ q_0.t$
$q_0.f_{comp} \circ s_0.h_{Lt} \circ q_0.t \circ q_1.t$	$q_0.f_{comp} \circ s_0.h_{Lw}$
$q_0.f_{comp} \circ s_0.h_{Lw} \circ q_0.w$	$q_0.f_{comp} \circ s_0.h_{Lw} \circ q_0.w \circ q_1.w$

図 4: 追加した素性テンプレート

JP Dep		all deps		w/o luw deps	
		UAS	LAS	UAS	LAS
LUW-based	SR joint	95.0	91.4	93.7	89.3
	Coma + SR single	94.9	91.3	93.5	88.9
	Coma + Malt	94.7	91.4	93.3	89.0
	Coma + MST	94.9	91.3	93.5	88.9
SUW-based	SR single	93.6	89.6	92.3	87.5
	Malt	92.9	89.2	90.9	86.7
	MST	93.5	89.4	91.8	86.9

表 1: 解析結果の比較

ReduceSUW$_L$ s_0 と s_1 の間で，LUW の内部構造となる左向きの弧を結ぶ．

ReduceSUW$_R$ s_0 と s_1 の間で，LUW の内部構造となる右向きの弧を結ぶ．

PopLUW s_0 (SUW から構成される部分木) を LUW として確定する．確定された LUW には，ShiftLUW のときに推定された品詞 pos が付与される．

依存構造解析に関する操作は次の通りである．

ReduceLUW$_L$(dep) s_0(LUW) と s_1(LUW) の間で，関係ラベル dep の左向きの弧を結ぶ．

ReduceLUW$_R$(dep) s_0(LUW) と s_1(LUW) の間で，関係ラベル dep の右向きの弧を結ぶ．

図 3 に，本アルゴリズムによる解析例を示す．

5. 評価実験

単語依存構造による構文解析について，短単位の依存構造を直接作る方法，長単位のチャンキングを行ってから長単位の依存構造を作る方法，長単位のチャンキングと長単位の依存構造解析を同時に行う方法を比較した．

		述語項	連体	連用	並列
LUW-based	SR joint	76.6	68.5	65.4	66.5
	Coma + SR single	75.9	65.9	65.3	65.9
	Coma + Malt	75.3	68.2	64.6	65.7
	Coma + MST	75.5	65.8	63.4	65.8
SUW-based	SR single	74.2	63.8	60.9	63.5
	Malt	73.2	63.5	58.4	59.7
	MST	73.2	62.2	58.6	63.9

表 2: 依存関係タイプのカテゴリ別の結果 (F 値)

5.1 設定

評価には，日本語用に設計した体系によるタイプ付き単語依存構造コーパス (以下，JP Dep) を使用した．JP Dep は，京大コーパス [Kurohashi 03] 4 万文のうち，2 万文について，長単位ベースの依存構造がアノテーションされている [Tanaka 15]．表 3 に使用したコーパスの統計量を示す *5．

短単位列の正解データを入力として，以下の解析方法を比較した．

- 長単位解析，長単位依存構造同時解析 (SR joint)
- パイプライン型解析 (Comainu + SR / Malt / MST)
- 短単位依存構造解析 (SR / MST / Malt)

長単位解析と短単位解析の同時解析は，forest reranking に基づく shift-reduce パーザ [Hayashi 13] を長単位解析の操作に対応するように拡張した解析器を実装して用いた (以下，SR joint)．ビーム幅は 12 とし，解析器の素性には，Huang[Huang 12], Hayashi らの素性に加えて，図 4 に示す長単位や複合辞に関する素性を使用した．stack に関して，添字 $_{Lw}$, $_{Lt}$ は，主辞 h，最左の子 l，最右の子 r に対して，それぞれ長単位の単語出現形，品詞を表している．queue に関しては，w, t は（短単位の）単語出現形，品詞である．また，複合辞の一部となりうる短単位に対するフラグを導入した．$q_0.f_{comp}$ は，q_0 の短単位を先頭とする複合辞が存在するかどうかを複合語辞書との照合を行い，存在する場合 1，存在しない場合 0 とするフラグである．

単独の長単位解析器として， Comainu[小澤 14]，依存構造解析器として，MST Parser [McDonald 06], MaltParser [Nivre 07]，SR joint から長単位解析機能を除いたもの (SR single) を用いた．パイプライン型の解析では， 長単

*5 JP Dep は，UD 規格を日本語化した仕様 [金山 15] に比較して，日本語の構文解析用に統語的な情報を保持するように作られている．

		♯Sent	♯SUW	♯LUW
JP Dep	test	2,000	53,193	41,192
	train	17,953	497,309	383,797

表 3: コーパス統計量

Chunking w/o POS	Prec	Recall	F
SR joint	98.87	99.25	99.06
Comainu	99.38	99.55	99.47

表 4: 長単位解析結果の比較

位解析モデルを Comainu により構築し，長単位ベース依存構造解析モデルを，各依存構造解析器で構築した．解析は，短単位列を入力として，Comainu で長単位解析し，解析結果の長単位列を入力として，各依存構造解析器が長単位ベースの依存構造解析結果を出力する．また，短単位ベースの依存構造解析モデルを各依存構造解析器で構築して，短単位列から直接依存構造解析も行った．

5.2 結果

二つの異なるタスクの組み合わせ結果を比較するため，単純には依存構造解析精度の比較ができない．長単位ベースの依存構造解析結果については，それぞれの解析結果を短単位ベースの依存構造に変換して，短単位間の依存構造として比較を行った．複数の短単位からなる長単位は，関係ラベル *luw* を使用して，フラットな短単位間の構造に変換した．

表 1 に解析結果を示す．UAS, LAS は，すべての依存関係について計算したもの (all deps) と，長単位の内部構造 (*luw*) を除いた依存関係について計算したもの (w/o luw deps) を分けている．同時解析で行う場合も，パイプラインで行う場合も，長単位をベースにして依存構造解析を行う方が高い精度が得られている．

一方，長単位ベースの依存構造解析器の精度に関しては，提案手法 (SR joint) が若干高いがあまり差は出ていない．ただし，表 2 にあるように依存関係タイプのうち主要なカテゴリに属するものの結果を見ると傾向が異なることがわかる．特に述語項 (*nsubj* など) や連用修飾 (*advmod* など), 並列 (*conj* など) については，同時解析方式はパイプライン方式に比較して，0.7 ポイント以上の向上が見られている．

また，長単位解析分割のみの結果は表 4 で示すようになった．提案手法の長単位解析単独の精度は，系列ラベリングによるチャンキング (Comainu) に僅かに及んでいないが，依存構造解析精度で比較するとパイプライン手法と同等以上の結果となっている．これは，曖昧性のある長単位の決定を依存構造解析時まで遅らせている効果と考えられる．

6. おわりに

日本語の単語依存構造解析の方法として，長単位に分割する解析と長単位に基づく依存構造解析を同時に行う方法を提案し，短単位に基づく依存構造解析よりも高精度で行えることを示した．今後，日本語以外の言語を含めた UD の複合語表現への適応するように拡張する予定である．

参考文献

[Hatori 12] Hatori, J., Matsuzaki, T., Miyao, Y., and Tsujii, J.: Incremental Joint Approach to Word Segmentation, POS Tagging, and Dependency Parsing in Chinese, in *Proceedings of ACL 2012*, Vol. 1, pp. 1045–1053 (2012)

[Hayashi 13] Hayashi, K., Kondo, S., and Matsumoto, Y.: Efficient Stacked Dependency Parsing by Forest Reranking, *TACL*, Vol. 1, pp. 139–150 (2013)

[Huang 12] Huang, L. and Sagae, K.: Dynamic Programming for Linear-time Incremental Parsing, in *Proceedings of ACL 2010*, pp. 1077–1086 (2012)

[Kurohashi 03] Kurohashi, S. and Nagao, M.: *Building a Japanese Parsed Corpus – while Improving the Parsing System*, chapter 14, pp. 249–260, Kluwer Academic Publishers (2003)

[Maekawa 14] Maekawa, K., Yamazaki, M., Ogiso, T., Maruyama, T., Ogura, H., Kashino, W., Koiso, H., Yamaguchi, M., Tanaka, M., and Den, Y.: Balanced Corpus of Contemporary Written Japanese, *Language Resources and Evaluation*, Vol. 48, No. 2, pp. 345–371 (2014)

[McDonald 06] McDonald, R., Lerman, K., and Pereira, F.: Multilingual Dependency Analysis with a Two-stage Discriminative Parser, in *Proceedings of CoNLL 2006*, pp. 216–220 (2006)

[Mori 14] Mori, S., Ogura, H., and Sasada, T.: A Japanese Word Dependency Corpus, in *Proceedings of LREC 2014*, pp. 753–758 (2014)

[Nivre 07] Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryiğit, G., Kübler, S., Marinov, S., and Marsi, E.: MaltParser: A Language-independent System for Data-driven Dependency Parsing, *Journal of Natural Language Engineering*, Vol. 13, No. 2, pp. 95–135 (2007)

[Nivre 15] Nivre, J.: Towards a Universal Grammar for Natural Language Processing, in *Proceedings of CICLing 2015*, pp. 3–16 (2015)

[Tanaka 15] Tanaka, T. and Nagata, M.: Word-based Japanese Typed Dependency Parsing with Grammatical Function Analysis, in *In Proceedings of ACL2015*, Vol. 2, pp. 237–242 (2015)

[Uchimoto 08] Uchimoto, K. and Den, Y.: Word-level Dependency-structure Annotation to Corpus of Spontaneous Japanese and its Application, in *Proceedings of LREC 2008*, pp. 3118–3122 (2008)

[Zhang 14] Zhang, M., Zhang, Y., Che, W., and Liu, T.: Character-Level Chinese Dependency Parsing, in *Proceedings of ACL 2014*, Vol. 1, pp. 1326–1336 (2014)

[金山 15] 金山博, 宮尾祐介, 田中貴秋, 森信介, 浅原正幸, 植松すみれ：日本語 Universal Dependencies の試案, 言語処理学会第 21 回年次大会予稿集, pp. 505–508 (2015)

[小澤 14] 小澤 俊介, 内元 清貴, 伝 康晴：長単位解析器の異なる品詞体系への適用, 自然言語処理, 第 21 巻, pp. 379–401 (2014)