Hierarchical Word Structure-based Parsing: A Feasibility Study on UD-style Dependency Parsing in Japanese

Takaaki Tanaka and **Katsuhiko Hayashi** and **Masaaki Nagata**
NTT Comminication Science Laboratories
2-4, Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0237, Japan
{tanaka.takaaki,hayashi.katsuhiko,nagata.masaaki}@lab.ntt.co.jp

Abstract

In applying word-based dependency parsing such as Universal Dependencies (UD) to Japanese, the uncertainty of word segmentation emerges for defining a word unit of the dependencies. We introduce the following hierarchical word structures to dependency parsing in Japanese: morphological units (a short unit word, SUW) and syntactic units (a long unit word, LUW). This paper describes the results of a feasibility study on the ability and the effectiveness of parsing methods based on hierarchical word structure (LUW chunking+parsing) by comparing them with methods using single layer word structure (SUW parsing). We also show joint analysis of LUW-chunking and dependency parsing improves the performance of identifying predicate-argument structures, while there is not much difference between overall results of them.

1 Introduction

Some research has recently been introducing word-based dependency schemes into Japanese syntactic parsing from a cross-lingual standpoint such as Universal Dependencies (UD) (Nivre et al., 2016; Kanayama et al., 2015; Tanaka et al., 2016), although syntactic structures are traditionally represented as dependencies between chunks called *bunsetsus*.

However, for languages like Japanese where words are not segmented by white spaces in orthography, word-based dependency parsing is problematic due to difficulties in defining a word unit. Actually, in Japanese several word unit standards exists that can be found in corpus annotation schemes or in the outputs of morphological analyzers. The word unit must be more consistently defined in word-based dependencies than bunsetsu-based dependencies, since the inconsistency of word units directly affects the discordance of the syntactic structure. UD for Japanese adopted a "short unit word" (SUW) defined for building the Balanced Corpus of Contemporary Written Japanese (BCCWJ) (Maekawa et al., 2014), since the word unit is designed to maintain internal consistency in the corpus.

An SUW is the smallest token that conveys morphological information, and generally corresponds to the head word of a morphological analysis dictionary called UniDic, which is compiled based on linguistic analysis and is used for morphological analyzers. Even though SUWs are well-organized as morphological units, they are sometimes too short to represent syntactic construction. Therefore, we also introduce another unit named "long unit word" (LUW), which consists of one or more SUWs with a single syntactic function, and is also defined for BCCWJ. For constructing an LUW-based syntactic structure, we need two types of analyses: LUW chunking and LUW-based dependency parsing. Note that LUWs include two kinds of multiwords: lexicalized phrases and institutionalized phrases (Sag et al., 2001), and for syntactic parsing, it is essential to discriminate functional multiwords that are classified into the latter in particular. Even though a pipeline process is a simple way of combining these two analyses, it may cause inconsistency between dependency parsing and chunking. Therefore, we introduce a joint analysis method of parsing and chunking to unify these two analyses by deciding dependency structures and chunks in the same process.

We describe two methods of hierarchical word-based parsing in Section 3: a pipeline analysis using a current LUW-chunking method and a joint analysis method. We present our evaluation of the

Proceedings of the 15th International Conference on Parsing Technologies, pages 56–60,
Pisa, Italy; September 20–22, 2017.

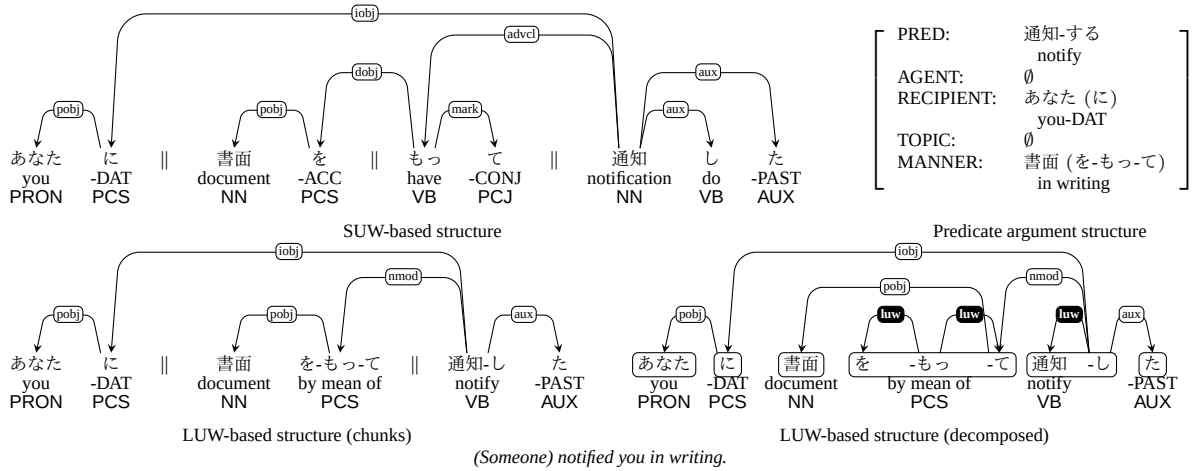

Figure 1: Examples of word-based dependencies. "luw" is a special dependency type that denotes intra-dependencies in an LUW. The symbols '||' denote the borders of bunsetsu chunks.

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

Table 1: Corpus statistics.

results of hierarchical word-based parsing (LUW-based) and single layer word-based (SUW-based) parsing in Section 4.

2 Hierarchical Word Dependencies

We employed two levels of word unit definitions as described in Section 1. A sentence is consistently segmented into morphological units of SUWs, while a syntactic structure is constructed based on syntactic units of LUWs since compound nouns and functional multiwords have a single syntactic function and are usually treated as single LUWs. The relationship between SUW and LUW almost correspond to the one between single word and multiword in other language. Note that an LUW that consists of just an SUW exists, and about 20% of LUWs belongs to a multiword.

Figure 1 shows the differences between SUW- and LUW-based dependency structures. Note that the scheme (described in Section 4.1) in the figure is similar to UD, but they differ in the manners of head selection. For instance, the scheme selects the case particle に -DAT as the head of the noun あなた *you*, while UD treats the noun as the head of the particle. In SUW-based dependencies (top left), SUW verb もつ *have*, a component of functional multiword を-もっ-て [1] *by mean of*, is treated as a main verb, creating a spurious complex structure between a verb and an argument. The pseudo predicates hinder the extraction of *true* predicate argument structures as shown in the top right of the figure. In an LUW-based dependency structure (bottom left), multiword を-もっ-て is considered an LUW with a flat structure, which clearly indicates the relation between main verb 通知-する *notify* and argument あなた に *you*-DAT. The conversion from SUW sequences into LUWs contains ambiguity. For example, sequence を も っ て in the sentence, "彼 は その 本 を もっ て いる", *lit. He has the book.*, is not just a single LUW but three LUWs with a main verb.

The amount of research on Japanese word-based dependency parsing is much less than bunsetsu-based parsing. Tanaka and Nagata (2015) proposed LUW based analysis using a scheme that resembles Stanford typed dependencies (SD) (de Marneffe and Manning, 2008), however, they do not treat LUW-chunking problems. Kato et al. (2017) explored English dependency parsing models that predict multiword expression (MWE)-aware structure. We treat broader categories of multiword in this paper, e.g. LUWs contain ordinary compound nouns as well as named entities. The test set has 8,291 multiwords (LUWs) in 2,000 sentences, while their corpus has 27,949 MWE instances in 37,015 sentences.

3 Analysis Methods

3.1 Pipeline analysis

The pipeline method sequentially runs two analyses; multiword analysis chunks an input SUW sequence into an LUW sequence, and parsing analysis constructs LUW-based dependency structures,

[1] The SUW boundaries in an LUW are denoted by "'-", a symbol that is not actually used in orthography.

LUW transition			Cond.
ShLUW(*p*)	$(\sigma_S, \sigma_L, \beta \mid x_k, A, L)$	$\Rightarrow(\sigma_S \mid p\langle x_k\rangle, \sigma_L, \beta, A, L)$	$\lvert\sigma_S\rvert = 0$
ReLUW$_L$(*r*)	$(\sigma_S, \sigma_L \mid p\langle x_k\rangle \mid q\langle x_m\rangle, \beta, A, L)$	$\Rightarrow(\sigma_S, \sigma_L \mid p\langle x_k\rangle, \beta, A \cup \{r(p\langle x_k\rangle, q\langle x_m\rangle\}, L)$	$\lvert\sigma_L\rvert \geq 2$
ReLUW$_R$(*r*)	$(\sigma_S, \sigma_L \mid p\langle x_k\rangle \mid q\langle x_m\rangle, \beta, A, L)$	$\Rightarrow(\sigma_S, \sigma_L \mid q\langle x_m\rangle, \beta, A \cup \{r(q\langle x_m\rangle, p\langle x_k\rangle\}, L)$	$\lvert\sigma_L\rvert \geq 2$
PopLUW	$(\sigma_S \mid p\langle x_k\rangle, \sigma_L, \beta, A, L)$	$\Rightarrow\{\sigma_S, \sigma_L \mid p\langle x_k\rangle, \beta, A, L \cup \{p\langle x\rangle\})$	$\lvert\sigma_S\rvert = 1$
SUW transition			Cond.
ShSUW	$(\sigma_S \mid p\langle x_k\rangle, \sigma_L, \beta \mid x_m, A, L)$	$\Rightarrow(\sigma_S \mid p\langle x_k\rangle \mid x_m, \sigma_L, \beta, A, L)$	
ReSUW$_L$	$(\sigma_S \mid p\langle x_k\rangle \mid x_m, \sigma_L, \beta, A, L)$	$\Rightarrow(\sigma_S \mid p\langle x_k\rangle, \sigma_L, \beta, A, L \cup \{\ell(x_k, x_m)\})$	$\lvert\sigma_S\rvert \geq 2$
ReSUW$_R$	$(\sigma_S \mid p\langle x_k\rangle \mid x_m, \sigma_L, \beta, A, L)$	$\Rightarrow(\sigma_S \mid p\langle x_m\rangle, \sigma_L, \beta, A, L \cup \{\ell(x_m, x_k)\}$	$\lvert\sigma_S\rvert \geq 2$

x_k SUW
$p\langle x\rangle$ POS of an LUW (head: x)
$r(x, y)$ syntactic dep. (head: x, rel: r)
$\ell(x, y)$ internal dep. in an LUW
Initial state $([], [], [x_0, ..., x_n], \emptyset, \emptyset)$
Final state $([], [], [], A_f, L_f)$

Figure 2: Transitions in our joint parsing algorithm.

as shown in the bottom left of Figure 1.

Kozawa et al. (2014) proposed a method that creates an LUW sequence from an SUW sequence in two steps: chunking an SUW sequence using an LUW-chunking model and assigning an LUW POS to each LUW with an LUW POS estimation model. LUW chunking is done by assigning each SUW in a given sequence either a "B" tag or an "I" tag by a sequence labeling method using CRF.

3.2 Joint analysis

The joint method simultaneously processes an SUW sequence with LUW chunking and syntactic parsing so that the LUW chunking is consistent with the syntactic analysis. The method directly constructs a dependency structure from an SUW sequence, as shown at the bottom right of Figure 1. LUWs consisting of multiple SUWs such as を-もっ-て and 通知-する are represented as a flat structure with a special dependency type *luw*.

We employed an algorithm based on shift-reduce parsing and defined two types of transitions: LUW chunking and dependency parsing. This algorithm is devised by applying a joint analysis method of word segmentation and dependency parsing in Chinese (Zhang et al., 2014; Hatori et al., 2012), or a method which combines lexical and syntactic analysis (Constant and Nivre, 2016). One of features of our algorithm is that a shift transition (ShLUW) assigns a leftmost SUW of an LUW with a POS. We found this obtains better scores than a pop transition (PopLUW) does.

Two stacks, σ_S and σ_L, are provided for SUWs to be processed and chunked LUWs respectively. The algorithm outputs an LUW sequence and an LUW-based parsed tree to a set of internal dependencies in LUW chunks L, and a set of dependencies A. A parsing status is represented as quintuple $(\sigma_S, \sigma_L, \beta, A, L)$, where β is a buffer that initially contains all SUWs in an input sentence, $(x_0, ..., x_n)$. Figure 2 shows transitions used in our algorithm. The necessary condition for each action is shown in the rightmost column. The notion $|\sigma|$ denotes depth of stack σ. For example, $|\sigma_S| = 0$ represents the condition that stack $|\sigma_S|$ is empty.

JP Dep		all deps		w/o luw deps	
Method		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

Table 2: Parsing results.

JP Dep Method	Pred Args	Adnom	Adverb	Coord
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

Table 3: F_1 scores of individual categories of dependency types.

4 Evaluation

We compared two LUW-based parsing methods and an SUW-based parsing method. A simple SUW-based parsing method directly constructs a dependency structure without considering LUWs. The SUW-based method regards "luw" as an ordinary dependency type.

4.1 Setting

Since the current UD Japanese corpora are SUW-based and do not have complete LUW information[2], we used another typed word dependency treebank in Japanese described in (Tanaka and Nagata, 2015)(JP Dep). JP Dep is annotated with LUW-based dependencies in accordance with

[2]They have partly compound word information by annotating dependencies with relation types "mwe"(UD1.2), "fixed"(UD 2.0) and so on.

a scheme that resembles SD, and consists of 20,000 sentences (Table 1) from a newspaper corpus, Kyoto Corpus (Kurohashi and Nagao, 2003).

SR joint employs a shift-reduce parser based on dynamic programming (Huang and Sagae, 2010; Hayashi et al., 2013) that is expanded with LUW-chunking transitions. We used the features related to LUW and the function compound words, in addition to the original features. Moreover, we employ features with flag where SUW may form an LUW of a function compound word. The flag becomes 1 only if a function compound word that begins with a target SUW exists in dictionaries, and otherwise is 0. The features are similar to the additional features used for the joint model (Joint+dict) proposed in (Kato et al., 2017) in terms of utilizing lexical knowledge in dictionaries. We chose 12 for the beam width based on trial results.

For the pipeline methods, we used Comainu (Coma) (Kozawa et al., 2014) as an LUW chunker that is independent of a syntactic parser. We compared the following three parsers by combining them with Comainu: MST Parser (McDonald et al., 2006), MaltParser (Nivre et al., 2007), and SR joint without LUW-chunking transition (SR single). The LUW-chunking model and the LUW-based dependency parsing models were built with the training division of JP Dep.

The SUW-based dependency parsing models were also trained to directly test the parsing of the SUW sequence. The model was trained with LUW-based structures decomposed into SUWs as a structure shown at the bottom right of Figure 1.

4.2 Results

The parsing results are shown in Table 2 [3]. UAS and LAS are calculated on two conditions: the scores of all the dependencies (all deps) and only the scores of the dependencies between LUWs (w/o luw deps), i.e., ignoring the internal dependencies in LUWs. Since the internal dependencies in LUWs are right-to-left and monotonous structures, as shown in Figure 1, and easier to be estimated than the inter-LUW dependencies, the scores of all the dependencies tend to be higher than those of inter-LUW dependencies.

Overall, the results of LUW-based dependency parsing outperformed the SUW-based ones as shown in Table 2. Regarding the LUW-based parsing results, we found few differences between SR joint and the pipeline methods. Nevertheless, the differences between the scores of the inter-LUW dependencies (w/o luw deps) is larger than those between the scores of all the dependencies. This indicates SR joint preferentially obtained better results of syntactic parsing instead of the results of LUW chunking. The differences between the results in the major dependent types are clearer as shown in Table 3. We can see the F_1 scores of the individual categories of the dependency types in the table, where predicate-argument categories (Pred Args) include "nsubj," "dobj," and "iobj." The SR joint improved more than 0.7 points over the pipeline methods in such major categories as Pred Args and adverbial modification, while we found few differences between overall results of the SR joint methods and the pipeline methods.

[3] We converted LUW-based dependencies into SUW-based dependencies by decomposing each LUW into SUWs with a flat structure to compare the results.

Multiword	Freq	SR joint		SR single	
		UAS	LAS	UAS	LAS
case particle					
に-つい-て *about*	19	89	58	84	47
と-いう *(a bird,) called (swallow)*	138	94	58	88	88
conjunctive particle					
と-し-て *by way of (explanation)*	83	84	74	81	74
に-よる-と *according to*	21	91	71	81	67
に-よっ-て *by*	12	92	83	75	67

Table 4: Attachment scores of dependencies including functional multiwords.

Treatment with the functional multiwords of a parsing method affected the scores of the dependency types in such categories as Pred Args and adverbial, where they tend to be long-distance dependencies. Table 4 shows the scores of the dependencies including major functional multiwords, and we found that SR joint obtained better scores than SR single as a whole. This suggests the advantages of identifying functional multiwords contribute the higher scores of the specific types.

5 Conclusion

We presented methods for processing word dependency parsing by treating hierarchical word structures by combining LUW chunking and LUW-based dependency parsing for Japanese syntactic parsing. LUW-based parsing outperformed the SUW-based method, and the joint analysis method is superior to the pipeline methods in identifying the major syntactic relations.

We are planning to apply our joint analysis method on an UD corpus for Japanese and other languages to handle multiword units in syntactic parsing based on UD schemes.

References

Matthieu Constant and Joakim Nivre. 2016. A transition-based system for joint lexical and syntactic analysis. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. volume 1 of *ACL 2016*, pages 161–171.

Marie-Catherine de Marneffe and Christopher D. Manning. 2008. The stanford typed dependencies representation. In *Proceedings of COLING 2008 Workshop on Cross-framework and Cross-domain Parser Evaluation*. pages 1–8.

Jun Hatori, Takuya Matsuzaki, Yusuke Miyao, and Jun'ichi Tsujii. 2012. Incremental joint approach to word segmentation, pos tagging, and dependency parsing in chinese. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*. volume 1 of *ACL 2012*, pages 1045–1053.

Katsuhiko Hayashi, Shuhei Kondo, and Yuji Matsumoto. 2013. Efficient stacked dependency parsing by forest reranking. *Transactions of the Association for Computational Linguistics* 1:139–150.

Liang Huang and Kenji Sagae. 2010. Dynamic programming for linear-time incremental parsing. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. ACL 2010, pages 1077–1086.

Hiroshi Kanayama, Yusuke Miyao, Takaaki Tanaka, Shinsuke Mori, Masayuki Asahara, and Sumire Uematsu. 2015. A draft of universal dependencies for japanese (in japanese). In *the 21st annual meeting of the Association for Natural Language Processing*. pages 505–508.

Akihiko Kato, Hiroyuki Shindo, and Yuji Matsumoto. 2017. English multiword expression-aware dependency parsing including named entities. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. volume 2 of *ACL 2017*, pages 427–432.

Shunsuke Kozawa, Kiyotaka Uchimoto, and Yoshiharu Den. 2014. Adaptation of long-unit-word analysis system to different part-of-speech target (in japanese). In *Journal of Natural Language Processing*. volume 21, pages 379–401.

Sadao Kurohashi and Makoto Nagao. 2003. *Building a Japanese Parsed Corpus – while Improving the Parsing System*, Kluwer Academic Publishers, chapter 14, pages 249–260.

Kikuo Maekawa, Makoto Yamazaki, Toshinobu Ogiso, Takehiko Maruyama, Hideki Ogura, Wakako Kashino, Hanae Koiso, Masaya Yamaguchi, Makiro Tanaka, and Yasuharu Den. 2014. Balanced corpus of contemporary written Japanese. *Language Resources and Evaluation* 48(2):345–371.

Ryan McDonald, Kevin Lerman, and Fernando Pereira. 2006. Multilingual dependency analysis with a two-stage discriminative parser. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*. CoNLL 2006, pages 216–220.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Haji , Christopher Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal Dependencies v1: A multilingual treebank collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*. LREC 2016, pages 1659–1666.

Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülşen Eryiğit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. 2007. Maltparser: A language-independent system for data-driven dependency parsing. *Journal of Natural Language Engineering* 13(2):95–135.

Ivan A. Sag, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger. 2001. Multiword expressions: A pain in the neck for nlp. In *Proceedings of the 3rd International Conference on Intelligent Text Processing and Computational Linguistics*. CICLing-2002, pages 1–15.

Takaaki Tanaka, Yusuke Miyao, Masayuki Asahara, Sumire Uematsu, Hiroshi Kanayama, Mori Shinsuke, and Yuji Matsumoto. 2016. Universal dependencies for Japanese. In *Proceedings of 10th edition of the Language Resources and Evaluation Conference*. LREC 2016.

Takaaki Tanaka and Masaaki Nagata. 2015. Word-based Japanese typed dependency parsing with grammatical function analysis. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*. volume 2 of *ACL 2015*, pages 237–242.

Meishan Zhang, Yue Zhang, Wanxiang Che, and Ting Liu. 2014. Character-level Chinese dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. volume 1 of *ACL 2014*, pages 1326–1336.