English Multiword Expression-aware Dependency Parsing Including Named Entities

Akihiko Kato and **Hiroyuki Shindo** and **Yuji Matsumoto**
Graduate School of Information and Science
Nara Institute of Science and Technology
8916-5, Takayama, Ikoma, Nara, 630-0192, Japan
{kato.akihiko.ju6, shindo, matsu} @is.naist.jp

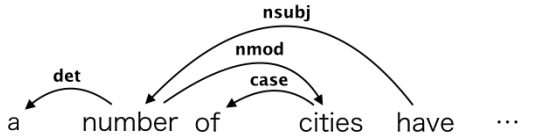
Abstract

Because syntactic structures and spans of multiword expressions (MWEs) are independently annotated in many English syntactic corpora, they are generally inconsistent with respect to one another, which is harmful to the implementation of an aggregate system. In this work, we construct a corpus that ensures consistency between dependency structures and MWEs, including named entities. Further, we explore models that predict both MWE-spans and an MWE-aware dependency structure. Experimental results show that our joint model using additional MWE-span features achieves an MWE recognition improvement of 1.35 points over a pipeline model.

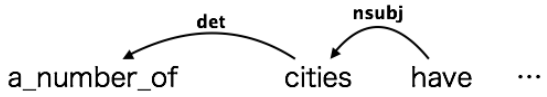
1 Introduction

To solve complex Natural Language Processing (NLP) tasks that require deep syntactic analysis, various levels of annotation such as parse trees and named entities (NEs) must be consistent with one another (Finkel and Manning, 2009). Otherwise, it is usually impossible to combine these pieces of information effectively.

However, the standard syntactic corpus of English, Penn Treebank, is not concerned with consistency between syntactic trees and spans of multiword expressions (MWEs). In Penn Treebank, that is, an MWE-span does not always correspond to a span dominated by a single non-terminal node. Therefore, word-based dependency structures converted from Penn Treebank are generally inconsistent with MWE-spans (Figure 1a). To mitigate this inconsistency, Kato et al. (2016) establishes each span of functional MWEs [1] as a subtree of a phrase structure in the Wall Street Journal portion of Ontonotes (Pradhan et al., 2007).



(a) a word-based dependency structure



(b) an MWE-aware dependency structure

Figure 1: A word-based and an MWE-aware dependency structure. In the former, a span of an MWE ("a number of") does not correspond to any subtree. The MWE is represented as a single node in the latter structure.

To pursue this direction further, we construct a corpus such that dependency structures are consistent with MWEs, by extending Kato et al. (2016)'s corpus [2]. As is the case with their corpus, each MME is a syntactic unit in an MWE-aware dependency structure from our corpus (Figure 1b). Moreover, our corpus includes not only functional MWEs but also NEs. Because NEs are highly productive and occur more frequently than functional MWEs, they are difficult to cover in a dictionary.

Consistency between NE-spans and phrase structures is not guaranteed because they are independently annotated in most syntactic corpora.

[1] By functional MWEs, we mean MWEs that function either as prepositions, conjunctions, determiners, pronouns, or adverbs.

[2] We release our dependency corpus at https://github.com/naist-cl-parsing/mwe-aware-dependency. MWE-aware phrase structures will be distributed from LDC as a part of LDC2017T01.

Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 427–432
Vancouver, Canada, July 30 - August 4, 2017.
https://doi.org/10.18653/v1/P17-2068

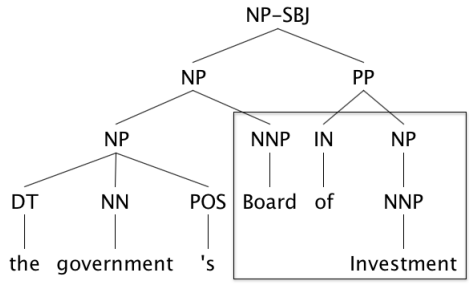


Figure 2: Example of inconsistency between NE-spans and phrase structures. A rectangle shows an NE-span.

MWE_POS	NNP	RB	IN	others
MWE Instances	20,992	3,796	2,424	737
MWE Types	11,875	377	92	52

Table 1: Corpus statistics.

For instance, in Figure 2, an NE-span is "Board of Investment," which is inconsistent with the syntactic tree. Therefore, we resolve this inconsistency by modifying phrase structures locally and establishing each NE as a subtree.

Furthermore, to evaluate the constructed corpus, we explore pipeline and joint models that predict both MWE-spans and an MWE-aware dependency tree [3]. Our experimental results show that the proposed joint model with additional MWE-span features achieves an MWE recognition improvement of 1.35 points over the pipeline model.

2 MWE-aware Dependency Corpus

To ensure consistency between MWE annotations and dependency structures, we first integrate NE annotations on Ontonotes [4] into phrase structures such that functional MWEs are established as subtrees. Subsequently, we convert phrase structures to dependency structures. We construct our corpus by extending Kato et al. (2016)'s corpus [5], which is itself built on a corpus by Shigeto et al. (2013). Regarding MWE annotations, Shigeto et al. (2013) first constructed an MWE dictionary by extracting functional MWEs from the English-language Wiktionary [6], and classified their occurrences in Ontonotes into either MWE or literal usage. Kato et al. (2016) integrated these MWE annotations into phrase structures and established functional MWEs as subtrees.

Type of MWEs	Non-terminal	Contiguous children	Crossing brackets
Functional MWEs	3,466	1,663	1,799
NEs	18,625	2,252	144

Table 2: Histogram tabling the consistency between MWE-spans and phrase structures.

Next, we describe the establishment of each NE as a subtree. If an NE-span does not correspond to any non-terminal in a phrase structure, there are two possibilities: (A) the NE-span corresponds to multiple contiguous children of a subtree, or (B) the NE-span has crossing brackets with the spans in the parse tree (Finkel and Manning, 2009; Kato et al., 2016). In Case (A), we insert a new non-terminal ("MWE_NNP") that governs the NE-span [7]. In Case (B), many instances correspond to a noun phrase (NP) comprised of a nested NP and a prepositional phrase (Figure 2). In the main NP, a modifier, such as a determiner, an adjective, or a possessive NP, precedes an NE. For these instances, according to Finkel and Manning (2009), we reduce Case (B) to Case (A) by moving the modifier from the nested NP to the main NP. Then, we establish each NE as a subtree by inserting an MWE-specific non-terminal. Furthermore, in some instances it is more reasonable to enlarge NE-spans than to modify phrase structures. As a typical example, there is an NE annotation that covers only part of a coordination structure, such as "Peter and Edward Bronfman," where "Edward Bronfman" is annotated as an NE. In this case, we extend an original NE-span to the whole coordination structure. We show the statistics for the corpus in Table 1 [8]. This corpus has 27,949 MWE instances in 37,015 sentences. A histogram

[3] Although Kato et al. (2016) conducts experiments regarding MWE-aware dependency parsing, they use gold MWE-spans. This is not a realistic scenario. By contrast, our parsing models do not use gold MWE-spans.

[4] We exploit NE annotations on Ontonotes Release 5.0 (LDC2013T19). We address traditional NEs, such as persons, locations, and organizations, while omitting the following: DATE, TIME, PERCENT, MONEY, QUANTITY, ORDINAL, and CARDINAL. Note that we only focus on multi-word NEs.

[5] https://catalog.ldc.upenn.edu/LDC2017T01

[6] https://en.wiktionary.org

[7] We do not require manual annotations for Case (A).

[8] NEs have NNP as an MWE-level POS tag.

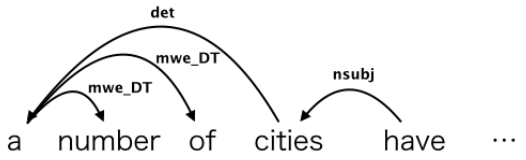


Figure 3: In the joint model, we directly infer an MWE-aware dependency tree in which an MWE ("a number of") is represented as a head-initial structure by a dependency parser.

tabling the consistency between MWE-spans and phrase structures is shown in Table 2. For tree-to-dependency conversion, we first replace a subtree corresponding to an MWE by a preterminal node and its child node. The preterminal node has an MWE-level POS (MWE_POS) tag. The child node is generated by joining all components of the MWE with underscores. We then convert a phrase structure into a Stanford-style dependency structure (Marneffe and Manning, 2008) (Figure 1b).

3 Models for MWE identification and MWE-aware dependency parsing

In this section, we explore models that predict both MWE-spans and an MWE-aware dependency structure (Figure 1b).

3.1 Pipeline Model

The pipeline model involves the following three steps. First, BIO tags encoding MWE-spans and MWE_POS tags, such as "B_NNP" and "I_DT" are predicted by a sequential labeler based on Conditional Random Fields (CRFs) (Lafferty et al., 2001). Second, tokens belonging to each predicted MWE-span are concatenated into a single node. Finally, an MWE-based dependency structure (Figure 1b) is predicted by an arc-eager transition-based parser. For the CRFs, in addition to word-form and character-based features, we use 1- to 3-gram features based on dictionaries of functional MWEs and NEs within 5-word windows from a target token. For a dictionary of functional MWEs, we use the dictionary by Shigeto et al. (2013) (Section 2). Meanwhile, we create a dictionary of NEs from a title list of English Wikipedia articles, excepting stop words, provided by UniNE [9]. Regarding parsing features, we use baseline features and rich non-local features proposed by Zhang and Nivre (2011).

[9] http://members.unine.ch/jacques.savoy/clef/englishST.txt

3.2 Joint Model

In the proposed joint model, MWE-spans and MWE_POS tags are encoded as dependency labels, and conventional word-based dependency parsing is performed by an arc-eager transition-based parser. We use the same parsing features used in the pipeline model. We convert MWEs in MWE-aware dependency structures (Figure 1b) to head-initial structures (Figure 3) that encode MWE-spans and MWE_POS tags. Note that this representation is similar to Universal Dependency (McDonald et al., 2013). When parsing, we use constraints based on a history of transitions and the dictionary of functional MWEs. This is done to avoid invalid dependency trees. Because NEs are highly productive, we do not use a constraint regarding NEs.

Joint(+dict)

We designed additional features based on matches with dictionaries of NEs and functional MWEs. Hereafter, we refer to the joint model coupled with these additional features as joint(+dict). For instance, given a sentence that starts with "a number of cities," the additional features are as follows: a / B_DT, number / I_DT, of / I_DT, cities / O. Based on these additional features, we extend the baseline features proposed by Zhang and Nivre (2011) to develop MWE-specific features whose atomic features include not only words and word-level POS tags, but also BIO tags encoding MWE-spans and MWE_POS tags.

Joint(+pred_span)

Because dictionary matching is not concerned with context, in this setting, we use MWE-spans and MWE_POS tags predicted by CRF, rather than dictionary matching. Hereafter, we refer to this as joint(+pred_span). By using features extracted from CRF predictions, we can mitigate error propagation from sequential labeling and consider information from a full sentence. Moreover, we can alleviate difficulties in predicting MWE-spans and MWE_POS tags encoded as head-initial structures (Figure 3) by the parser.

4 Experimental Setting

We split the Wall Street Journal (WSJ) portion of Ontonotes, using sections 2-21 for training, and section 23 for testing. For all models, we used

	Dependency Parsing				MWE Recognition	
	All sentences		First tokens of MWEs			
Model	**UAS**	**LAS**	**UAS**	**LAS**	**FUM**	**FTM**
Pipeline	91.39	89.42	84.06	78.22	91.40	91.32
Joint	91.15	89.18	81.93	77.74	89.03	88.79
Joint(+dict)	91.36	89.37	84.45	80.74	91.93	91.78
Joint(+pred_span)	91.50	89.51	84.85	81.29	92.75	92.60

Table 3: Experimental results on the test set.

	Dependency Parsing (First tokens of MWEs)				MWE Recognition		
	Functional MWEs		NEs		Functional MWEs		NEs
Model	**UAS**	**LAS**	**UAS**	**LAS**	**FUM**	**FTM**	**FUM**
Pipeline	78.89	64.01	85.58	82.41	96.76	96.42	89.81
Joint	71.28	65.05	85.07	81.49	91.01	89.93	88.47
Joint(+dict)	79.93	73.70	85.79	82.82	97.94	97.25	90.16
Joint(+pred_span)	81.31	74.74	85.89	83.23	97.59	96.91	91.32

Table 4: Breakdown of experimental results by type of MWE. Note that UAS / LAS are calculated regarding first tokens of MWEs. For NEs, the FTM is the same as the FUM because each NE always takes NNP as an MWE-level POS tag, and is not repeated.

the POS tags predicted by the Stanford POS tagger (Toutanova et al., 2003) [10]. For the pipeline model and joint(+pred_span), we used MWE-spans and MWE_POS tags predicted by CRF [11]. For dependency parsing, we used Redshift (Honnibal et al., 2013) for all models, with a beam size of 16 for decoding. For training, we removed non-projective dependency trees. For testing, we parsed all sentences. To evaluate parsing, we used unlabeled and labeled attachment scores (UAS/LAS) [12]. For the pipeline model, we converted each concatenated token corresponding to an MWE into a head-initial structure and compared this with the gold tree. For the joint model, we directly compared a predicted tree with the gold tree. To evaluate MWE recognition, we used the F-measure for untagged / tagged MWEs (FUM/FTM) [13]. For the pipeline model, we compared the gold MWEs with predictions by CRF. For the proposed joint model, we compared the gold MWEs with predicted MWE-spans and MWE_POS tags represented as dependency labels.

5 Experimental Results and Discussion

We present the experimental results in Table 3. Comparing the joint model with the pipeline model, there is not much difference between these models regarding UAS / LAS for all sentences. However, the former is 2.13 / 0.48 points worse than the latter in terms of UAS / LAS regarding the first tokens of MWEs (1269 in 34,526 tokens), and 2.37 / 2.53 points worse than the latter regarding FUM / FTM. These results suggest that the performance of the joint model with no additional features at predicting dependencies inside and around MWEs is worse than the pipeline model. One of the reasons for this is that the exploitation of head-initial structures in the joint model (Figure 3) involves the addition of MWE-specific labels. This results in an increase in the total number of dependency labels from 41 to 50. Because of this broader output space, more search errors can occur in the joint model compared with the pipeline model. Moreover, a breakdown by type of MWE (Table 4) shows that most differences in performance between these two models are related to functional MWEs. These results suggest that constraints regarding functional MWEs during parsing (3.2) are harmful to the joint model with no additional features in terms of its performance with

[10] We used 20-way jackknifing for the training split. The test split was automatically tagged by the POS tagger trained on the training split.

[11] We used 20-way jackknifing for the training split. The test split was automatically tagged by the sequential labeler trained on the training split.

[12] When calculating UAS/LAS, we removed punctuation.

[13] FUM only focuses on MWE-spans, whereas FTM focuses on both MWE-spans and MWE_POS tags.

respect to functional MWEs.

By adding MWE-specific features to the joint model, however, we observe at least a 2.52 / 3.00 point improvement in terms of UAS / LAS regarding the first tokens of MWEs, and a 2.90 / 2.99 point improvement regarding FUM / FTM. As a result, we obtain a 1.35 / 1.28 point improvement with joint(+pred_span) compared with the pipeline model in terms of FUM / FTM. A breakdown by type of MWE shows that the addition of MWE-specific features leads to a performance improvement, especially for functional MWEs (Table 4). These results suggest that MWE-specific features are effective at both MWE recognition through dependency parsing and the prediction of dependencies connecting inside and outside of MWEs.

Comparing the joint(+pred_span) with the joint(+dict), the former is 0.40 / 0.55 points better than the latter in terms of UAS / LAS regarding the first tokens of MWEs, and 0.82 / 0.82 points better than the latter regarding FUM / FTM. We can attribute this gain in performance to the additional features extracted from more accurate predictions of MWE-spans and MWE_POS tags by CRF than those by dictionary matching.

6 Related Work

Whereas French Treebank is available for French MWEs (Abeillé et al., 2003), there have been only limited corpora for English MWE-aware dependency parsing. Schneider et al. (2014) constructs an MWE-annotated corpus on English Web Treebank (Bies et al., 2012). However, this corpus is relatively small as training data for a parser, and its MWE annotations are not consistent with syntactic trees. By contrast, our corpus covers the whole of the WSJ portion of Ontonotes and ensures consistency between MWE annotations and parse trees.

Korkontzelos and Manandhar (2010) reports an improvement in base-phrase chunking by pre-grouping MWEs as words-with-spaces. They focus on compound nouns, adjective-noun constructions, and named entities. However, they use gold MWE-spans, and this is not a realistic setting. By contrast, we use predicted MWE-spans.

Three works concerned with a French MWE-aware syntactic parsing are relevant. First, Green et al. (2013) proposes a method for recognizing contiguous MWEs as a part of constituency parsing by using MWE-specific non-terminals. They investigate a CFG-based model and a model based on tree-substitution grammars. Second, Candito and Constant (2014) compares several architectures for graph-based dependency parsing and MWE recognition, in which MWE recognition is conducted before, during, and after parsing. Finally, Nasr et al. (2015) explores a joint model of MWE recognition and dependency parsing. They focus on complex function words. In terms of data representation, they adopt one similar to ours, insofar as the components of an MWE are linked by dependency edges whose labels are MWE-specific.

7 Conclusion

We constructed a corpus that ensures consistency in Ontonotes between dependency structures and English MWEs, including named entities. Furthermore, we explored models that can predict both MWE-spans and an MWE-aware dependency structure. Our experiments show that by using additional MWE-span features, our joint model achieves an MWE recognition improvement of 1.35 points over the pipeline model.

Acknowledgments

This work was partially supported by JST CREST Grant Number JPMJCR1513 and JSPS KAKENHI Grant Number 15K16053. We are grateful to members of the Computational Linguistics Laboratory at NAIST, and to the anonymous reviewers for their valuable feedback. Regarding the preparation of a title list from English-language Wikipedia articles, we are particularly grateful for the assistance given by Motoki Sato.

References

Anne Abeillé, Lionel Clément, and François Toussenel. 2003. *Building a Treebank for French*, Springer Netherlands, Dordrecht, pages 165–187.

Ann Bies, Justin Mott, Colin Warner, and Seth Kulick. 2012. English web treebank. *Technical Report LDC2012T13, Linguistic Data Consortium, Philadelphia, Pennsylvania, USA.* .

Marie Candito and Matthieu Constant. 2014. Strategies for contiguous multiword expression analysis and dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, pages 743–753. https://doi.org/10.3115/v1/P14-1070.

Rose Jenny Finkel and D. Christopher Manning. 2009. Joint parsing and named entity recognition. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, pages 326–334. http://aclweb.org/anthology/N09-1037.

Spence Green, Marie-Catherine de Marneffe, and Christopher D Manning. 2013. Parsing models for identifying multiword expressions. *Computational Linguistics* 39(1):195–227.

Matthew Honnibal, Yoav Goldberg, and Mark Johnson. 2013. *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, Association for Computational Linguistics, chapter A Non-Monotonic Arc-Eager Transition System for Dependency Parsing, pages 163–172. http://aclweb.org/anthology/W13-3518.

Akihiko Kato, Hiroyuki Shindo, and Yuji Matsumoto. 2016. Construction of an english dependency corpus incorporating compound function words. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association (ELRA), Paris, France.

Ioannis Korkontzelos and Suresh Manandhar. 2010. Can recognising multiword expressions improve shallow parsing? In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, Los Angeles, California, pages 636–644. http://www.aclweb.org/anthology/N10-1089.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. 18th International Conf. on Machine Learning*. Morgan Kaufmann, San Francisco, CA, pages 282–289. citeseer.ist.psu.edu/lafferty01conditional.html.

Marie-Catherine Marneffe and D. Christopher Manning. 2008. *Coling 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation*, Coling 2008 Organizing Committee, chapter The Stanford Typed Dependencies Representation, pages 1–8. http://aclweb.org/anthology/W08-1301.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, and Jungmee Lee. 2013. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics, pages 92–97. http://aclweb.org/anthology/P13-2017.

Alexis Nasr, Carlos Ramisch, José Deulofeu, and André Valli. 2015. Joint dependency parsing and multiword expression tokenization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, Beijing, China, pages 1116–1126. http://www.aclweb.org/anthology/P15-1108.

Sameer S. Pradhan, Eduard H. Hovy, Mitchell P. Marcus, Martha Palmer, Lance A. Ramshaw, and Ralph M. Weischedel. 2007. Ontonotes: A unified relational semantic representation. In *Proceedings of the First IEEE International Conference on Semantic Computing (ICSC 2007), September 17-19, 2007, Irvine, California, USA*. pages 517–526. https://doi.org/10.1109/ICSC.2007.83.

Nathan Schneider, Spencer Onuffer, Nora Kazour, Emily Danchik, Michael T. Mordowanec, Henrietta Conrad, and Noah A. Smith. 2014. Comprehensive annotation of multiword expressions in a social web corpus. In Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. European Language Resources Association (ELRA), Reykjavik, Iceland, pages 455–461. ACL Anthology Identifier: L14-1433.

Yutaro Shigeto, Ai Azuma, Sorami Hisamoto, Shuhei Kondo, Tomoya Kouse, Keisuke Sakaguchi, Akifumi Yoshimoto, Frances Yung, and Yuji Matsumoto. 2013. *Proceedings of the 9th Workshop on Multiword Expressions*, Association for Computational Linguistics, chapter Construction of English MWE Dictionary and its Application to POS Tagging, pages 139–144. http://aclweb.org/anthology/W13-1021.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*. http://aclweb.org/anthology/N03-1033.

Yue Zhang and Joakim Nivre. 2011. Transition-based dependency parsing with rich non-local features. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, pages 188–193. http://aclweb.org/anthology/P11-2033.