Structure Modeling Approach for UD Parsing of Historical Modern Japanese

Hiroaki Ozaki[1] **Mai Omura**[2]
Komiya Kanako[1] **Masayuki Asahara**[3,4] **and Toshinobu Ogiso**[3,4]
[1]Tokyo University of Agriculture and Technology, Japan
[2]Osaka Shoin Women's University, Japan
[3]National Institute for Japanese Language and Linguistics, Japan
[4]The Graduate University for Advanced Studies, Japan
hiroaki-ozaki@st.go.tuat.ac.jp, omura.mai@osaka-shoin.ac.jp,
kkomiya@go.tuat.ac.jp, {masayu-a,togiso}@ninjal.ac.jp

Abstract

This study shows the effectiveness of structure modeling for transferability in diachronic syntactic parsing. The syntactic parsing for historical languages is significant from a humanities and quantitative linguistics perspective to enable annotation support and analysis on unannotated documents. We compared the zero-shot transfer ability between Transformer-based Biaffine UD parsers and our structure modeling approach. The structure modeling approach is a pipeline method consisting of dictionary-based morphological analysis (MeCab), a deep learning-based phrase (bunsetsu) analysis (Monaka), SVM-based phrase dependency parsing (CaboCha) and a rule-based conversion from phrase dependencies to UD. This pipeline closely follows the methodology used in constructing Japanese UD corpora. Experimental results showed that the structure modeling approach outperformed zero-shot transfer from the contemporary to the modern Japanese. Moreover, the structure modeling approach outperformed several existing UD parsers in contemporary Japanese. To this end, the structure modeling approach outperformed in the diachronic transfer of Japanese by a wide margin and was useful to those applications for digital humanities and quantitative linguistics.

1 Introduction

Dependency parsing has long been studied as a core task in natural language processing. In recent years, dependency annotation corpora created under multilingual unified annotation frameworks such as Universal Dependencies (UD; Zeman et al. 2018) have been published, enabling the development of deep learning-based dependency parsers that operate across multiple languages.

On the other hand, syntactic structure analysis, including dependency parsing, is beneficial for humanities research as well as traditional NLP applications. This is because structure modeling, which consists of a layered pipeline of NLP tasks, can preserve low-level linguistic structures such as phrases required for humanities research. In contrast, the high zero-shot transfer performance of recent deep learning-based parsers (Kondratyuk and Straka, 2019) would also be helpful to support annotation tasks and quantitative linguistic analysis on unannotated corpora such as historical literature, which no longer has native speakers.

Thus, this study focuses on the diachronic application of UD dependency parsing and compares structure modeling with end-to-end deep learning in the context of zero-shot transfer. Specifically, Japanese UD corpora exist for both contemporary and Meiji-period (modern) Japanese, allowing us to investigate the transfer performance from contemporary Japanese, which has sufficient training resources, to modern Japanese. The research questions regarding the structure modeling in this context are: (1) whether it demonstrates a performance advantage and (2) whether it is effective when considering practical annotation and application use cases.

For our structure modeling approach to Japanese UD parsing (see Figure 1), we first applied a morphological analysis (MeCab), and then, we applied deep learning-based phrase (bunsetsu) segmentation (Monaka) and bunsetsu dependency parsing (CaboCha). After this, we employed a rule-based transformation from bunsetsu dependencies to UD annotation, simulating actual Japanese UD annotations. This structure modeling approach closely follows the standard workflow used for constructing Japanese UD linguistic resources, involving morphological annotation, bunsetsu dependency annotation, and subsequent rule-based conversion into the UD format. As a comparison, we trained and used a graph-based Biaffine parser (Attardi et al., 2021), which is a representative UD parsing method.

The comparison results showed that (1) the

Proceedings of the 1st Joint Workshop on Large Language Models and Structure Modeling (XLLM 2025), pages 106–114

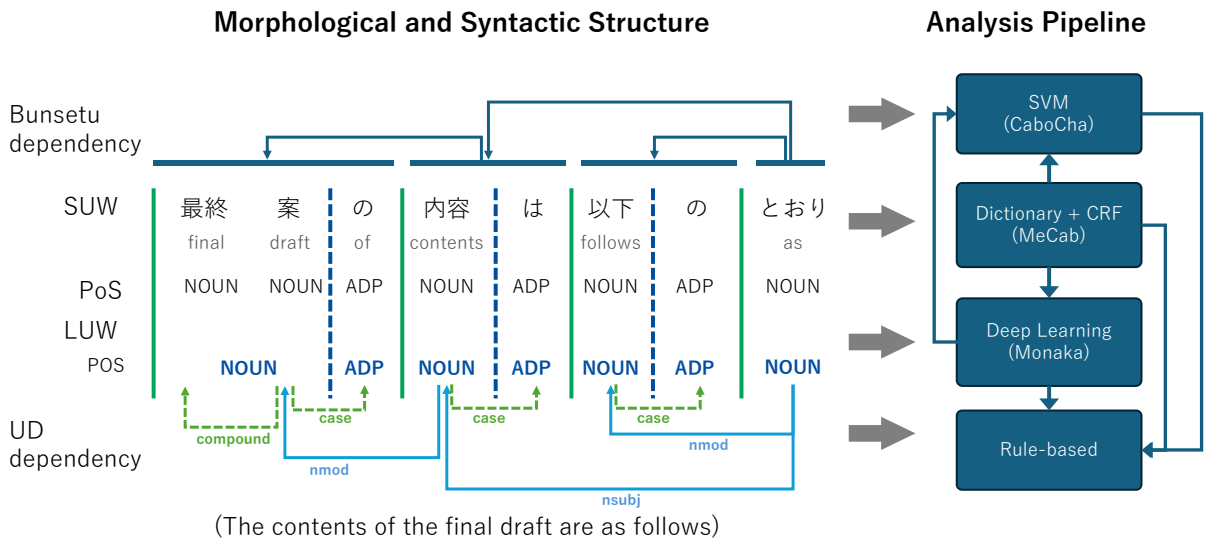


Figure 1: The overview of the task and structural approach. Green solid bars represent bunsetsu boundaries, and blue dotted bars represent the boundaries of long unit words. The left side of the figure depicts the morphological and syntactic information of the sentence "最終案の内容は以下のとおり (The contents of the final draft is as follows)." The right side of the picture shows the analysis pipeline of the structure modeling approach.

structure modeling approach not only outperformed deep learning-based zero-shot transfer in accuracy but also achieved high performance on contemporary Japanese before transfer. (2) In practical annotation scenarios, zero-shot transfer using deep learning alone was impractical due to the inconsistency of phrase (bunsetsu) structures, whereas the structure modeling approach produced reasonable results, preserving morphological and phrase (bunsetsu) structures.

2 Related Work

2.1 UD Treebanks of Japanese

There are two major UD treebanks for contemporary Japanese. UD_Japanese-BCCWJ[1] (Asahara et al., 2018; Omura and Asahara, 2018) is a treebank built on the Balanced Corpus of Contemporary Written Japanese (Maekawa, 2008). UD_Japanese-GSD[2] (Tanaka et al., 2016) contains the sentences from Google Universal Dependency Treebanks v2.0 (legacy)[3]. These two Japanese UD treebanks are annotated with mostly the same methods and criteria.

UD_Japanese-Modern[4] (Omura and Asahara, 2017) is a deprecated UD treebank annotated on Meiroku-zasshi that was published in the Meiji period (C.E. 1868-1912). This was the only official annotated (test set only) UD treebank for historical Japanese, containing 822 sentences.

2.2 Parsing Methods for UD

For syntactic parsers (not limited to UD parsers), there are two major approaches, namely graph- and transition-based parsers. In UD parsing, graph-based parsers (often called Biaffine Parser; Dozat et al. 2017; Qi et al. 2018; Che et al. 2018) won the competitions of the CoNLL shared task in 2017 and 2018 (Zeman et al., 2017, 2018). Their Biaffine parsers are available as Stanza [5] models.

Because of the success of the graph-based approach, there have been many investigations performed to improve the parsing performance, for example, DiaParser (Attardi et al., 2021) extends the architecture of the Biaffine Parser by exploiting both embeddings and attentions provided by transformers and achieved high performance.

On the other hand, before the competition, Straka et al. (2016) provides a transition-based parser, UDPipe[6], which reconstructs parsed trees based on estimated action sequences applying word tokens. And another popular NLP tool spaCy [7] also provides transition-based parsers.

[1] https://github.com/UniversalDependencies/UD_Japanese-BCCWJ
[2] https://github.com/UniversalDependencies/UD_Japanese-GSD
[3] https://github.com/ryanmcd/uni-dep-tb
[4] https://github.com/UniversalDependencies/UD_Japanese-Modern
[5] https://stanfordnlp.github.io/stanza/
[6] https://github.com/ufal/udpipe
[7] https://spacy.io/

In contrast, this paper focuses on structure modeling of UD parsing, including deep learning-based phrase segmentation.

Multilingual Transfer of Deep UD parsers Biaffine parsers have high transfer ability, especially for low-resource languages. Kondratyuk and Straka (2019) shows their single Biaffine model named UDify trained on 75 UD treebanks with high performances for those low-resource treebanks.

UD Parsers for Contemporary Japanese There are a lot of parsers which support contemporary Japanese UD. For example, the spaCy [8] supports Japanese UD parsing. GiNZA (Matsuda, 2020), which is also a spaCy-based parser, trained specifically for contemporary Japanese.

UD Parser for Modern Japanese For UD parsing methods applied to modern Japanese, Yasuoka (2020) examined an approach that combines morphological information conversion using the UniDic designed for modern Japanese with an existing Japanese UD dependency parser. While this morphological conversion significantly improves accuracy, it has been reported that the accuracy does not reach the level achieved when trained directly on the UD_Japanese-Modern (Meirokuzasshi) corpus. Additionally, this parser has been released as `unidic2ud`[9]. In the `unidic2ud` repository, Yasuoka (2020) provides a few UD annotated sentences from famous literature written in modern Japanse (Yukiguni, Maihime, and Koyayori).

3 Bunsetsu Dependency for Syntactic Structure of Japanese

The left side of Figure 1 shows an example of Japanese's morphological and syntactic structure. In Japanese, bunsetsu dependency relations (shown in the top dependency tree of Figure 1) are widely used for representing syntactic structure. A bunsetsu is the smallest and natural phrase unit for native Japanese speakers, and syntactic structure is expressed through the dependency relations between bunsetsu phrases.

A bunsetsu consists of one or more words. However, since Japanese lacks a whitespace separation of words in its writing system, there are multiple word unit definitions, such as short unit words (SUWs) and long unit words (LUWs) defined by the National Institute for Japanese Language and Linguistics (NINJAL). In this study, we explain bunsetsu structures based on the SUWs and LUWs, which are commonly used in Universal Dependencies (UD).

3.1 Bunsetsu (Base-phrase)

As mentioned above, a bunsetsu is a (natural) minimal phrase that consists of a Japanese sentence. An example of bunsetsu boundaries is shown in Figure 1 as green solid lines. Generally, a bunsetsu boundary appears after a particle or a sequence of particles. This is because Japanese functional words typically follow their content words, on which they depend. In Figure 1, all LUW noun (NOUN) and adposition (ADP) pairs are composed into bunsetsu segments.

3.2 Short Unit Word

Short Unit Word (SUW) is a token close to the granularity of typical Japanese word tokens. A dictionary (UniDic) was established for SUWs, enabling high-performance morphological analysis based on UniDic (Den et al., 2008). As shown in the overview Figure 1, bunsetsu and LUWs are also composed of SUWs.

3.3 Long Unit Word

The Long Unit Word (LUW) is a lexical unit corresponding to a bunsetsu. Identification of LUW involves identifying bunsetsu and then dividing each bunsetsu into independent and attached LUWs. In Figure 1, blue dotted lines represent LUWs' boundaries, which divide bunsetsu into independent and attached LUWs.

4 Structure Modeling Approach for UD Parsing

In standard Japanese UD annotation (Asahara et al., 2018; Omura and Asahara, 2018), bunsetsu dependency information is used as a basis for rule-based conversion into UD annotation, referencing the SUWs and LUWs contained within each bunsetsu.

Therefore, by estimating the SUW, LUW, and bunsetsu boundaries, along with the dependency relations between bunsetsu, it is possible to obtain UD parsing results by applying the same conversion rules.

Figure 1 shows the pipeline of the structure modeling approach. The pipeline starts from

[8] https://spacy.io/

[9] https://github.com/KoichiYasuoka/UniDic2UD

a CRF-based SUW analysis (MeCab), and then the results of SUW analysis are sent to deep learning-based LUW and bunsetsu analysis (Monaka). Next, all SUW, LUW and bunsetsu information are sent to bunsetsu dependency parsing (CaboCha), and finally, with all the information, rule-based UD conversion is performed.

4.1 SUW Analysis

For SUW analysis, MeCab [10], which uses the Conditional Random Field (CRF; Lafferty et al. 2001) with a dictionary, was generally used. The actual analysis was performed using fugashi[11], a Cython wrapper for MeCab (McCann, 2020). In the MeCab-based analysis, UniDic was used as the SUW dictionary. UniDic supports not only modern Japanese [12], but also various periods of the Japanese language from old Japanese (Nara-period; C.E. 710-) onward.

4.1.1 Bunsetsu Analysis (Monaka)

For bunsetsu analysis, the parser named Monaka[13] proposed by Ozaki et al. (2024) was used. Monaka simultaneously predicts bunsetsu boundaries, LUW boundaries, and part-of-speech tags of LUWs from a sequence of SUWs. As described in the previous section, SUWs can be analyzed using MeCab, allowing us to perform all necessary analyses except for bunsetsu dependency parsing.

The method proposed by Ozaki et al. (2024) targets Japanese from the Heian (C.E. 794-1185) to Muromachi (C.E. 1336-1573) periods, as stored in the Corpus of Historical Japanese (CHJ). Because their method provides publicly available code, we newly built a one-model bunsetsu and LUW parser covering both the Heian–Muromachi periods and contemporary Japanese. Building the model, we referenced the bunsetsu and LUW information included as UFeat in UD_Japanese-BCCWJ and UD_Japanese-GSD. The hyperparameters to train the model are also the same as the original ones.

4.1.2 Bunsetsu Dependency Parsing (CaboCha)

For bunsetsu dependency parsing, we used CaboCha (Taku Kudo, 2002). CaboCha is a bunsetsu dependency parser based on Support Vector Machines (SVM). It consists of multiple analysis layers, including SUW analysis, bunsetsu segmentation, and bunsetsu dependency parsing, allowing for layer-specific parser customization. The default SUW analysis layer in CaboCha uses MeCab.

Since CaboCha's bunsetsu dependency parsing is performed based on the features of SUWs within each bunsetsu, it is possible to conduct only bunsetsu dependency parsing by passing SUWs and bunsetsu information to CaboCha in its designated format.

4.1.3 Rule-based UD Conversion for Bunsetsu Dependency

For rule-based UD conversion of bunsetsu dependencies, we employed the method proposed by Asahara et al. (2018); Omura and Asahara (2018). This method was used in the creation of the Japanese UD corpus for the Balanced Corpus of Contemporary Written Japanese (BCCWJ)[14].

In this rule-based conversion, dependencies between bunsetsu are transformed into dependencies between SUWs that represent the meaning of the bunsetsu, such as content words (shown as light blue solid arrows of UD dependency in Figure 1). Each SUW within a bunsetsu is then set to depend on the representative SUW of that bunsetsu (shown as green dotted arrows of UD dependency in Figure 1).

In Japanese bunsetsu dependency parsing, only dependency relations between bunsetus are defined; no relation label is assigned (see Figure 1. However, in the UD framework, dependency relations must always have labels. Therefore, in this rule-based conversion, UD dependency labels are assigned by referencing morphological information such as the part-of-speech tags of SUWs and LUWs.

Although the conversion can be performed using only SUW morphological information, LUW morphological information will improve the accuracy of the transformation.

5 Evaluation

5.1 Target Corpora

We used UD_Japanese-GSD and UD_Japanese-BCCWJ, which are contemporary Japanese UD corpora, for training. For evaluation, UD_Japanese-Modern (Meiroku-zasshi), a UD

[10] https://taku910.github.io/mecab/
[11] https://github.com/polm/fugashi
[12] https://clrd.ninjal.ac.jp/unidic/download_all.html
[13] https://github.com/komiya-lab/monaka
[14] https://github.com/UniversalDependencies/UD_Japanese-BCCWJ

	Heian	Kamakura	Muromachi	Contemporary	
				BCCWJ	GSD
Bunsetsu	97.35	97.38	97.86	93.85	97.88
LUW span	99.69	99.44	98.99	97.87	98.84
+ PoS	99.33	99.03	98.00	96.73	98.23

(a) The one-model

	Heian	Kamakura	Muromachi	Contemporary	
				BCCWJ	GSD
Bunsetsu	97.03	97.69	97.87	94.04	98.01
LUW span	99.64	99.47	98.95	98.00	99.02
+ PoS	99.29	99.08	98.08	97.09	98.32

(b) Trained on each period

Table 1: The evaluation results of the one-model bunsetsu parser. The results for models trained on each period on Heian to Muromachi periods are from Ozaki et al. (2024).

corpus from the Meiji-period, and corpora (Yukiguni and Maihime) independently created by Yasuoka (2020), included with `unidic2ud`, were used.

5.2 Models

5.2.1 Bunsetsu and LUW

We trained the one-model bunsetsu parser explained in §4.1.1. We compared models trained on corpora from each period. Evaluation results for historical Japanese were from Ozaki et al. (2024). We newly trained contemporary bunsetsu parsers for each UD_Japanese-BCCWJ and UD_Japanese-GSD. These bunsetsu parsers were trained on each training set of UD treebanks and tested on their corresponding test sets for each treebank.

5.2.2 UD

As a deep learning-based parser, we used DiaParser, a graph-based Biaffine parser model (Attardi et al., 2021). For word embeddings, we used Japanese BERT provided by Tohoku University[15], and these embeddings were kept frozen during training. The models used for comparison are as follows:

Structure: The structure modeling approach proposed in this study.

jBERT: A DiaParser model utilizing Japanese BERT provided by Tohoku University.

UDify: A Biaffine parser model trained on 75 UD treebanks (Kondratyuk and Straka, 2019).

GiNZA: A transition-based parsing model provided by spaCy[16], trained on BCCWJ. For contemporary Japanese, we compared GiNZA as is (Matsuda, 2020).

Unidic2ud: Unidic2ud[17] provides a UDPipe (Straka et al., 2016) model trained on modern Japanese (UD_Japanese-Modern (Meiroku-zasshi)), morphological analysis was performed using MeCab with modern UniDic.

5.3 Evaluation Method

5.3.1 Bunsetsu and LUW

We used span-based f1-value evaluation (same as the evaluation used for the original bunsetsu parser; Ozaki et al., 2024).

5.3.2 UD

To focus solely on the evaluation of dependency parsing, we compared only dependency by ignoring tokenization errors (AlignedAcc). The evaluation metric was the Labeled Attachment Score (LAS), which is an extraction performance metric including the relationship label between two words in a dependency relation. Additionally, the Unlabeled Attachment Score (UAS), which measures the extraction performance of two words in a dependency relation, was also used for comparison. The evaluation script used was the one employed in the CoNLL Shared Task 2018 (Zeman et al., 2018) [18].

[15] https://huggingface.co/cl-tohoku/bert-base-japanese

[16] https://spacy.io/

[17] https://github.com/KoichiYasuoka/UniDic2UD

[18] https://universaldependencies.org/conll18/conll18_ud_eval.py

Period	Contemporary				Modern					
Corpus	BCCWJ		GSD		Yukiguni		Maihime		Meiroku-zasshi	
Model	UAS	LAS	UAS	LAS	UAS	LAS	UAS	LAS	UAS	LAS
Structure	**92.52**	**91.32**	92.42	91.18	**89.29**	**85.71**	**92.45**	**77.36**	**83.40**	**63.91**
jBERT	90.19	88.82	90.88	89.70	78.18	74.55	75.00	65.38	79.41	57.88
UDify	-	-	**94.37**	**92.08**	83.93	78.57	79.25	58.49	74.99	55.62
GiNZA	87.52	85.89	88.52	87.12	-	-	-	-	-	-
Unidic2ud	-	-	-	-	89.09	87.27	88.46	75.00	(88.20)	(72.87)

Table 2: UAS/LAS evaluation results. **UDify** results in the contemporary Japanese and Meiroku-zasshi are from their paper (Kondratyuk and Straka, 2019). Other results were evaluated by the parsed outputs. Because **Unidic2ud** was trained on Meiroku-zasshi, we show their performances surrounded by parentheses.

Period	Contemporary		Modern		
Corpus	BCCWJ	GSD	Yukiguni	Maihime	Meiroku-zasshi
Words	98.94	98.68	100.	100.	99.50
UPOS	98.21	96.80	94.64	94.34	87.61
XPOS	98.13	96.65	69.64	79.25	73.13

Table 3: UPOS/XPOS evaluation results.

5.4 Evaluation Results

5.4.1 Bunsetsu and LUW

Table 1 shows the evaluation results of the one-model bunsetsu parser, which was trained on the CHJ (Heian (C.E. 794-1185), Kamakura (C.E. 1185-1336), and Muromachi (C.E. 1336-1573) periods), UD_Japanese-BCCWJ, and UD_Japanese-GSD. Compared to the models trained on each period, the one-model approach achieved comparable results. Since modern Japanese (Meiji period) is in between the contemporary and Muromachi periods, the one-model bunsetsu parser is expected to perform well in modern Japanese.

5.4.2 UD

Dependency Table 2 shows UAS and LAS values for each corpus. Bold values indicate the highest value for each corpus and metric.

The structure modeling approach achieved the highest performance in BCCWJ and other modern corpora. The structure modeling approach outperformed the existing Japanese UD parser GinZA. This indicates the structure modeling approach is effective in improving UD parsing performance.

UDify reported the highest performance on GSD, however, its performance on Meiroku-zasshi was the worst. As well as **UDify**, **jBERT** struggled to perform in modern Japanese, despite their strong cross-lingual transfer ability. This indicates phrase (bunsetsu) or morphological level transfers are required for diachronic syntactic analysis.

Notably, it demonstrated high transfer performance in UAS, whereas LAS for Meiroku-zasshi (UD_Japanese-Modern) tended to be lower overall. This suggests that Meiroku-zasshi was created based on a different annotation standard compared to contemporary Japanese, Yukiguni, and Maihime.

SUW Accuracy Table 3 shows accuracy of SUW analysis. Because we use the same SUW analyzer (MeCab/`unidic2ud`), we compared accuracies for each corpus. Words, UPOS, and XPOS values were calculated by the CoNLL Shared Task 2018 evaluation script [19].

The Words value represents tokenization accuracy. From modern to contemporary Japanese, tokenization has been performed without significant issues. However, focusing on UPOS, performance declines in modern Japanese, with particularly low values observed in Meiroku-zasshi. Since UPOS represents the accuracy of language-independent PoS tags in UD, the rule-based conversion from bunsetsu dependency parsing to UD, which uses PoS information to estimate dependency labels, may contribute to the decreased accuracy of dependency labels. This corresponds to the overall significantly lower LAS in Meiroku-zasshi and

[19] https://universaldependencies.org/conll18/conll18_ud_eval.py

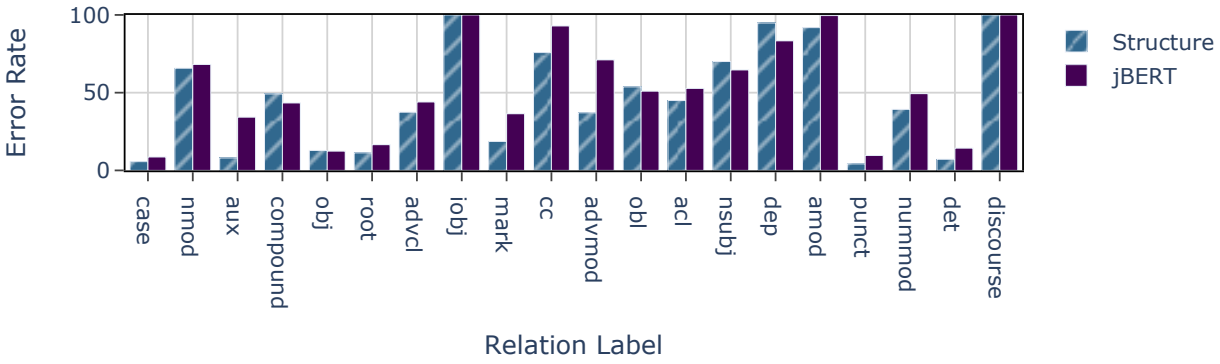
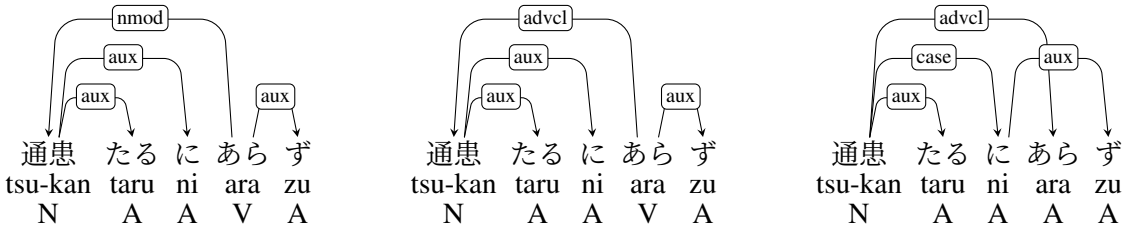


Figure 2: Comparison with dependency labels of models.



(a) Gold (b) Structure (c) jBERT

Figure 3: An example of parse results for BERT-based, the structure models. "N" represents nouns, "A" represents auxiliary verbs, and "V" represents verbs, respectively. The example is picked from Meirokku-zasshi.

suggests that, despite the availability of dictionaries for early modern Japanese, further improvements in SUW performance are necessary.

On the other hand, XPOS represents the accuracy of more detailed, language-specific PoS labels in UD, but its values have significantly declined compared to UPOS. This decline is not due to SUW analyze errors but rather inconsistencies in XPOS labeling during the annotation process of the UD corpus. Therefore, improving XPOS performance requires a normalization process for XPOS labels.

5.5 Comparison by Dependency Labels

Figure 2 shows the error rate comparison between **jBERT** and **Structure** models for each UD dependency label. We investigated all dependency relations based on dependency labels. The structure modeling approach achieves a lower error rate for most dependency labels, especially for aux dependencies. Because identifying aux relations, bunsetsu and PoS tags is important, the structure modeling approach can estimate them appropriately, resulting in the high performance of aux dependencies. However, for dependency labels that represent case relations between bunsetsu, such as obl (oblique nominal) and nsubj (nominal subject), the **jBERT** model has a slightly lower error rate.

This suggests that the **jBERT** model may have a better ability to transfer knowledge for semantic relationships compared to the structure modeling approach. It also indicates that incorporating features from BERT into the structure modeling approach for bunsetsu dependency parsing could potentially improve accuracy.

5.6 Case Study

Figure 3 shows an example of parse results for **jBERT** and **Structure** models compared to their gold annotation. The phrase "通患たるにあらず (tsu-kan taru ni ara zu: it is not generally a problem)" consists of two bunsetsu "通患たるに (tsu-kan taru ni)" and "あらず (ara zu)." In the structure modeling approach, intra-bunsetsu dependencies are preserved, and the bunsetsu "通患たるに (tsu-kan taru ni)" is correctly dependent on the bunsetsu "あらず (ara zu)", resulting in a valid dependency structure under UD. However, there was a mismatch in dependency labels between bunsetsu. Since the PoS tags of the SUW composing the phrases were also correctly predicted, this discrepancy in inter-bunsetsu dependency labels is likely due to differences in annotation standards between the contemporary and the modern UD corpora.

On the other hand, in the parsing result using

the **jBERT** model, the dependency within the bunsetsu, such as "ず (zu)" being associated with "に (ni)," is not preserved. Moreover, despite the fact that the "に (ni)" related to "tsu-kan" is an auxiliary verb, the dependency label is predicted as "case," which seems to be a confusion with particles. Additionally, similar to the structure modeling approach, "あら (ara)" is treated as modifying "通患 (tsu-kan)" in the adverbial clause (advcl), which can be considered a natural result from the perspective of contemporary Japanese annotation.

6 Conclusion

This study shows the effectiveness of structure modeling for transfer ability in diachronic syntactic parsing. We compared the zero-shot transfer ability between Transformer-based Biaffine UD parsers and our structure modeling approach. The structure modeling approach is a pipeline method consisting of dictionary-based morphological analysis (MeCab), a deep learning-based phrase (bunsetsu) analysis (Monaka), SVM-based phrase dependency parsing (CaboCha), and a rule-based conversion from phrase dependencies to UD, which closely follows the methodology used in constructing Japanese UD corpora. Experimental results showed that the structure modeling approach outperformed zero-shot transfer from the contemporary to the modern Japanese by a wide margin. The structure modeling approach outperformed several existing UD parsers in contemporary Japanese. Moreover, for other languages as well, it may be beneficial to adopt an analysis approach based on an understanding of resource construction methods, such as how UD resources are created or how parsed trees are transformed into UD format using head rules. From a case study, the structure modeling approach can preserve low-level information such as morphology and phrases (bunsetsu). On the other hand, the Biaffine parser has slightly better transfer performances of case relations. To this end, the structure modeling performed well on diachronic transfer in Japanese.

Limitations

Our structure modeling approach and compared models use "base" size BERT models; thus, by using larger models, the conclusion might differ from that we achieved. Since SUWs, LUWs, and bunsetsu analysis have been established for historical Japanese, we can easily apply our structure modeling approach. However, this is a rather unique case, and it might be harder to apply a similar approach to diachronic transfer for other languages.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number 22K12145, and a collaborative research project of National Institute for Japanese Language and Linguistics Joint Research Project "Extending the Diachronic Corpus through an Open Co-construction Environment" and "Evidence-based Theoretical and Typological Linguistics."

References

Masayuki Asahara, Hiroshi Kanayama, Takaaki Tanaka, Yusuke Miyao, Sumire Uematsu, Shinsuke Mori, Yuji Matsumoto, Mai Omura, and Yugo Murawaki. 2018. Universal Dependencies version 2 for Japanese. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Giuseppe Attardi, Daniele Sartiano, and Maria Simi. 2021. Biaffine dependency and semantic graph parsing for EnhancedUniversal dependencies. In *Proceedings of the 17th International Conference on Parsing Technologies and the IWPT 2021 Shared Task on Parsing into Enhanced Universal Dependencies (IWPT 2021)*, pages 184–188, Online. Association for Computational Linguistics.

Wanxiang Che, Yijia Liu, Yuxuan Wang, Bo Zheng, and Ting Liu. 2018. Towards better UD parsing: Deep contextualized word embeddings, ensemble, and treebank concatenation. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 55–64, Brussels, Belgium. Association for Computational Linguistics.

Yasuharu Den, Junpei Nakamura, Toshinobu Ogiso, and Hideki Ogura. 2008. A proper approach to Japanese morphological analysis: Dictionary, model, and evaluation. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Timothy Dozat, Peng Qi, and Christopher D. Manning. 2017. Stanford's graph-based neural dependency parser at the CoNLL 2017 shared task. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 20–30, Vancouver, Canada. Association for Computational Linguistics.

Dan Kondratyuk and Milan Straka. 2019. 75 languages, 1 model: Parsing Universal Dependencies universally. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2779–2795, Hong Kong, China. Association for Computational Linguistics.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Kikuo Maekawa. 2008. Balanced Corpus of Contemporary Written Japanese. In *Proceedings of the 6th Workshop on Asian Language Resources*.

Hiroshi Matsuda. 2020. Ginza - practical japanese nlp based on universal dependencies. *Journal of Natural Language Processing*, 27(3):695–701.

Paul McCann. 2020. fugashi, a tool for tokenizing Japanese in python. In *Proceedings of Second Workshop for NLP Open Source Software (NLP-OSS)*, pages 44–51, Online. Association for Computational Linguistics.

Mai Omura and Masayuki Asahara. 2017. Universal dependency for japanese modern. In *Japan Association for Digital Humanities*.

Mai Omura and Masayuki Asahara. 2018. UD-Japanese BCCWJ: Universal Dependencies annotation for the Balanced Corpus of Contemporary Written Japanese. In *Proceedings of the Second Workshop on Universal Dependencies (UDW 2018)*, pages 117–125, Brussels, Belgium. Association for Computational Linguistics.

Hiroaki Ozaki, Kanako Komiya, Masayuki Asahara, and Toshinobu Ogiso. 2024. Long unit word tokenization and bunsetsu segmentation of historical Japanese. In *Proceedings of the 1st Workshop on Machine Learning for Ancient Languages (ML4AL 2024)*, pages 48–55, Hybrid in Bangkok, Thailand and online. Association for Computational Linguistics.

Peng Qi, Timothy Dozat, Yuhao Zhang, and Christopher D. Manning. 2018. Universal Dependency parsing from scratch. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 160–170, Brussels, Belgium. Association for Computational Linguistics.

Milan Straka, Jan Hajič, and Jana Straková. 2016. UDPipe: Trainable pipeline for processing CoNLL-U files performing tokenization, morphological analysis, POS tagging and parsing. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4290–4297, Portorož, Slovenia. European Language Resources Association (ELRA).

Yuji Matsumoto Taku Kudo. 2002. Japanese dependency analysis using cascaded chunking. In *CoNLL 2002: Proceedings of the 6th Conference on Natural Language Learning 2002 (COLING 2002 Post-Conference Workshops)*, pages 63–69.

Takaaki Tanaka, Yusuke Miyao, Masayuki Asahara, Sumire Uematsu, Hiroshi Kanayama, Shinsuke Mori, and Yuji Matsumoto. 2016. Universal Dependencies for Japanese. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1651–1658, Portorož, Slovenia. European Language Resources Association (ELRA).

Koichi Yasuoka. 2020. Keitaiso kaiseki-bu no tsukekae ni yoru kindai nihongo(kyu-jitai kyu-kana) no kakari-uke kaiseki (in japanese) dependency parsing of modern japanese (kyujitai and kyukana) by replacing morphological analysis units. Technical Report 3, Information Processing Society of Japan.

Daniel Zeman, Jan Hajič, Martin Popel, Martin Potthast, Milan Straka, Filip Ginter, Joakim Nivre, and Slav Petrov. 2018. CoNLL 2018 shared task: Multilingual parsing from raw text to Universal Dependencies. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–21, Brussels, Belgium. Association for Computational Linguistics.

Daniel Zeman, Martin Popel, Milan Straka, Jan Hajič, Joakim Nivre, Filip Ginter, Juhani Luotolahti, Sampo Pyysalo, Slav Petrov, Martin Potthast, Francis Tyers, Elena Badmaeva, Memduh Gokirmak, Anna Nedoluzhko, Silvie Cinková, Jan Hajič jr., Jaroslava Hlaváčová, Václava Kettnerová, Zdeňka Urešová, Jenna Kanerva, Stina Ojala, Anna Missilä, Christopher D. Manning, Sebastian Schuster, Siva Reddy, Dima Taji, Nizar Habash, Herman Leung, Marie-Catherine de Marneffe, Manuela Sanguinetti, Maria Simi, Hiroshi Kanayama, Valeria de Paiva, Kira Droganova, Héctor Martínez Alonso, Çağrı Çöltekin, Umut Sulubacak, Hans Uszkoreit, Vivien Macketanz, Aljoscha Burchardt, Kim Harris, Katrin Marheinecke, Georg Rehm, Tolga Kayadelen, Mohammed Attia, Ali Elkahky, Zhuoran Yu, Emily Pitler, Saran Lertpradit, Michael Mandl, Jesse Kirchner, Hector Fernandez Alcalde, Jana Strnadová, Esha Banerjee, Ruli Manurung, Antonio Stella, Atsuko Shimada, Sookyoung Kwak, Gustavo Mendonça, Tatiana Lando, Rattima Nitisaroj, and Josie Li. 2017. CoNLL 2017 shared task: Multilingual parsing from raw text to Universal Dependencies. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–19, Vancouver, Canada. Association for Computational Linguistics.