Human Performance in Incremental Dependency Parsing: Dependency Structure Annotations and their Analyses

Hiroki Unno[1], Tomohiro Ohno[2], Koichiro Ito[1], Shigeki Matsubara[1,3]
[1]Graduate School of Informatics, Nagoya University
[2]Graduate School of Science and Technology for Future Life, Tokyo Denki University
[3]Information Technology Center, Nagoya University
unno.hiroki.t9@s.mail.nagoya-u.ac.jp
ohno@mail.dendai.ac.jp
{ito.koichiro.z5, matsubara.shigeki.z8}@f.mail.nagoya-u.ac.jp

Abstract

Incremental dependency parsing identifies the dependency structure as each component in a sentence is inputted. Since this task needs to predict non-inputted parts of the sentence, it is challenging not only for machines but also for humans. Although comparing machines and humans in this task is interesting, human performance in incremental dependency parsing has not been well studied due to lack of sufficient evaluation data. This study presents the construction of a large-scale data annotated with human incremental dependency parsing and string prediction and evaluates the human performance on these tasks. The data includes 3,639 written and 1,935 spoken sentences incrementally annotated by humans as each word was inputted. The dependency structure produced incrementally by humans was designed based on the intuition that they simultaneously predict non-inputted words and establish dependencies between previously inputted and non-inputted words. This study contributes to reveal the difficulty of incremental dependency parsing and certain aspects of human behavior in this task.

1 Introduction

Real-time language processing systems have applications for spoken and written languages. Applications for spoken language include simultaneous machine interpretation (Liu et al., 2021), spoken dialogue modeling (Nguyen et al., 2023), and real-time captioning (Piperidis et al., 2004; Ohno et al., 2009). For written language, applications, such as text input support systems (Murata et al., 2010), could be provided. A common requirement of these systems is to execute processing simultaneously with time-continuous input of sentence components. Incremental dependency parsers provide these systems with syntactic information for the input up to that point each time the input is received. In other words, these parsers identify dependencies between components of a sentence even when the input is still in progress (Kato and Matsubara, 2009; Ohno and Matsubara, 2013).

In incremental dependency parsing, whenever a component in a sentence is inputted, the dependency structure for the sequence of inputted components needs to be identified. The dependency structure that should be output at each point depends on what the speaker/writer inputs subsequently. For this reason, accurately performing is highly challenging even for humans. Understanding human performance in this task is meaningful, as it can guide the performance achieved by incremental parsing systems. However, existing research has made little attempt to reveal the difficulty of this task for humans, and has been limited to assessing comparisons between parsers based on their agreement with the correct structure. This is due to the lack of data to evaluate human performance in incremental dependency parsing.

In recent years, advances in large-scale language models have led to the development of datasets for various tasks to evaluate their effectiveness (Kurihara et al., 2022; Reid et al., 2022; García-Ferrero et al., 2023). These evaluations often include comparison with human performance (Lee et al., 2023). Additionally, many analyses have been conducted to identify potential differences in language comprehension processes between the models and humans (Shaitarova et al., 2023; Rodriguez et al., 2024). One possible approach to quantitatively analyze the human language comprehension process is to collect a large-scale data of incremental dependency parsing process by humans.

This study presents the construction of a large-scale data annotated with incremental dependency parsing results by humans. We evaluates human performance on incremental dependency parsing and reveal certain aspects of human behavior. The data were constructed by annotating 3,639 and 1,935 sentences of written and spoken languages

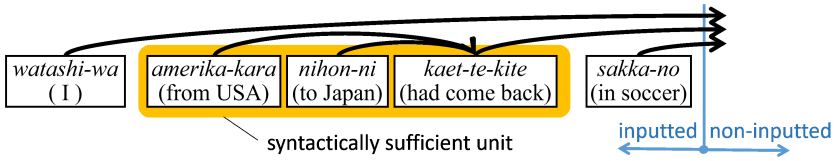

Figure 1: Dependency structure which expresses the fact that some bunsetsus do not depend on any inputted bunsetsus.

with dependency structures, respectively. The annotators identified these structures one by one as each word was inputted in sequence from the beginning of the sentence. The structures include not only the dependencies between previously inputted *bunsetsus*[1] but also those between previously inputted and non-inputted bunsetsus. It also captures predicted non-inputted words. The design is based on the intuition that humans simultaneously predict words in non-inputted bunsetsus and the dependencies between previously inputted and non-inputted bunsetsus.

The remainder of this paper is organized as follows. Section 2 describes previous works and our designed structure for incremental dependency parsing. Section 3 outlines the annotation on human incremental dependency parsing and presents the annotation results. Section 4 discusses the analysis of the data. Finally, Section 5 summarizes this research and suggests directions for future works.

2 Incremental Dependency Parsing

2.1 Previous Works

Many works have focused on incremental dependency parsing, which identifies dependency relationships between components of a sentence in the middle of the input one (Kato et al., 2001; Kato and Matsubara, 2009; Ohno and Matsubara, 2013). However, there has been little discussion of the specific information that should be included in a parser's output structure. The previous parsers (Kato et al., 2005; Johansson and Nugues, 2007; Nivre, 2008) update the parsing results midstream whenever a new word is inputted. They output pairs of modifiers and modifyees whenever they detect such pairs. Therefore, these parsers can only output a dependency relation after the modifier and modifyee have been inputted.

To solve this problem, Ohno and Matsubara (2013) proposed a structure that a Japanese incremental dependency parser should output in terms of the requirements of real-time language processing systems. Their proposed dependency structure requires the parser to clarify that a bunsetsu whose modified bunsetsu has not yet been inputted does not depend on any previously inputted bunsetsu. Figure 1 illustrates the dependency structure that a parser outputs immediately after the bunsetsu "*sakka-no* (in soccer)" is inputted while incrementally parsing the sentence "*watashi-wa amerika-kara nihon-ni kaet-te-kite sakka-no warudokappu-wo mimashi-ta* (I watched the World Cup in soccer after I had come back to Japan)." If it becomes clear that the modified bunsetsu of a bunsetsu has not been inputted yet, the higher layer applications can identify *syntactically sufficient units*[2] in the inputted sequence of bunsetsus and effectively use this information. For example, in Figure 1, the sequence of bunsetsus enclosed in the orange box "*amerika-kara nihon-ni kaet-te-kite* (after I had come back to Japan)" is identified as a syntactically sufficient unit. In fact, information on a syntactically sufficient unit is crucial for detecting the timing to start interpreting in simultaneous machine interpretation (Ryu et al., 2006) and determining the proper linefeed position in captioning (Ohno et al., 2009).

Additionally, several studies have focused on predicting specific words in the non-inputted parts of a sentence to support real-time language processing systems, as described in Section 1 (Grissom II et al., 2014; Tsunematsu et al., 2020; Cai et al., 2022). In the incremental process of human language understanding, we can intuitively assume that humans simultaneously predict specific words

[1] *Bunsetsu* is a linguistic unit in Japanese that roughly corresponds to a basic phrase in English. A bunsetsu consists of one independent word and zero or more ancillary words. A dependency relation in Japanese is a modification relation in which a modifier bunsetsu depends on a modified bunsetsu. In other words, the modifier bunsetsu and the modified bunsetsu work as modifier and modifyee, respectively.

[2] A syntactically sufficient unit is defined as a sequence of bunsetsus of which the dependency structure is closed, that is, any bunsetsu except the final bunsetsu does not depend on a bunsetsu outside the sequence.

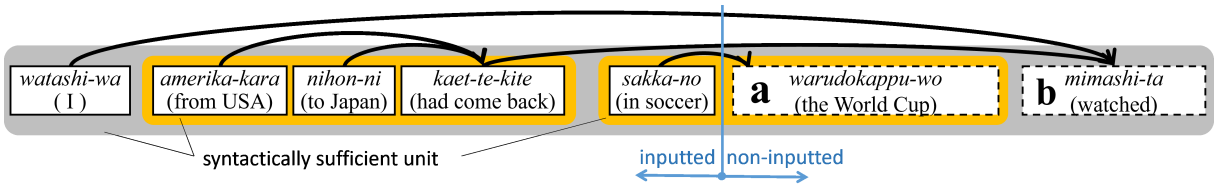

Figure 2: Dependency structure, which includes dependency relationships between inputted and non-inputted bunsetsus and the predicted specific strings of the non-inputted bunsetsus.

in non-inputted parts and parse dependencies between previously inputted and non-inputted parts. Based on this intuition, a study on incremental parsing partly exists, which adds a pseudo node representing the part of speech (POS) of the word to be next inputted and identifies syntactic relations between already inputted components and the added node (Köhn and Menzel, 2014). However, to the best of our knowledge, no study has simultaneously addressed incremental dependency parsing and prediction of specific words in non-inputted parts.

2.2 Dependency Structure for Incremental Parsing

In this section, we describe a new dependency structure that we introduce in this research. This structure is defined by integrating the dependency structure shown in Figure 1 (Ohno and Matsubara, 2013) with the prediction of specific words in non-inputted parts of a sentence.

Our new dependency structure can explicitly express the dependency relationships between inputted and non-inputted bunsetsus. When multiple bunsetsus depend on any of non-inputted bunsetsus (Figure 1), they may depend on different bunsetsus or the same bunsetsu. Our new dependency structure clarifies whether those bunsetsus depend on the same non-inputted bunsetsu. Furthermore, the specific strings of the non-inputted bunsetsus are predicted.

Figure 2 shows our new dependency structure in the same situation as Figure 1. This dependency structure clarifies that the bunsetsus "*watashi-wa* (I)" and "*kaet-te-kite* (after I had come back to Japan)" depend on the same non-inputted bunsetsu **b**, whereas the bunsetsu "*sakka-no* (in soccer)" depends on a different non-inputted bunsetsu **a**. Additionally, the non-inputted strings of bunsetsu **a** and **b** are predicted as "*warudokappu-wo* (the World Cup)" and "*mimashi-ta* (watched)," respectively. If such a dependency structure is identified, syntactically sufficient units can be detected in greater detail, as shown by the orange and gray boxes in Figure 2.

3 Annotation on Human Incremental Dependency Parsing

In this section, we describe the construction of a large-scale data annotated with results of human performance in incremental dependency parsing. In the construction, whenever a bunsetsu in a sentence included in the existing corpus is displayed one by one, the already displayed sequence of bunsetsus is annotated with the identified dependency structure of the format in Figure 2 and strings of the non-inputted bunsetsus in the structure are predicted. This study provides annotations for written and spoken Japanese. In what follows, we describe the existing corpus of our target for annotation and explain the data construction.

3.1 Target Data of Annotation

In our research, we used 3,639 sentences in Kyoto University Text Corpus Version 4.0 (Kyoto Corpus) (Kawahara et al., 2002), which consists of approximately 40,000 sentences from Japanese newspaper with morphological and syntactic annotations, for written language, and all sentences in Japanese lecture speech of Simultaneous Interpretation Database (SIDB) (Tohyama et al., 2005), which consists of 1,935 sentences from Japanese lecture speech transcriptions with morphological and syntactic annotations, for spoken language, as target data for annotation.

The difficulty of incremental dependency parsing and string prediction can be influenced by readability of the inputted sentences. In our research, we focus on human language processing in written and spoken language, which are relatively well readable. Newspaper articles in Kyoto Corpus are consistently written in a style familiar to the general audience, thus ensuring a consistent level of readability. The lecture manuscripts in SIDB were prepared in advance, and the transcribed texts are

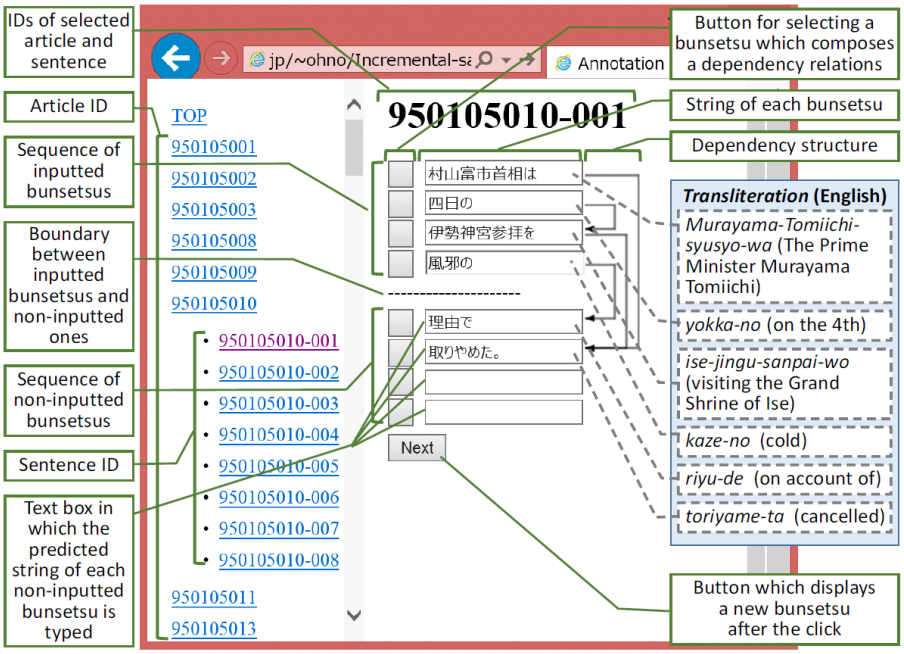

Figure 3: Web interface for annotation.

relatively well readable within the spoken language domain.

3.2 Outline of Data Construction

Two annotators (annotator A and B), who are native Japanese speakers, annotated 3,639 sentences (including 36,824 bunsetsus) in Kyoto Corpus. Annotator A also annotated 1,935 sentences (including 23,598 bunsetsus) in SIDB. Each annotator completed the whole annotation by iterating the annotation for a sentence after reading the annotation manual. The procedure for annotating a sentence using the Web interface shown in Figure 3 is as follows:

(1) Selection of a target sentence: An annotator selects an article's/lecture's ID in order from the top on the left side of the interface, and then the list of IDs of sentences included in the article/lecture is displayed. After that, the annotator selects a sentence ID in order from the top. This is because humans are generally thought to predict non-inputted bunsetsu using context.

(2) Annotation of the selected sentence: Following the selection of a sentence, the annotator proceeds with annotations of the sentence on the right side of the interface, where a bunsetsu in the sentence is displayed one by one. Whenever a new bunsetsu is displayed, the annotator conducts the following two annotation steps for the sequence of bunsetsus, which has already been displayed, with no time restriction.

(2a) The annotator annotates the inputted sequence of bunsetsus with the dependency structure of the format in Figure 2 by deciding each modified bunsetsu for all the inputted bunsetsus. Here, a non-inputted bunsetsu is allowed to become a modified bunsetsu. In Figure 3, an arrow means a dependency relation.

(2b) The annotator predicts strings of the non-inputted modified bunsetsus in the dependency structure determined by (2a) and then types each string in the corresponding text box. Annotators can choose not to type a string if they cannot think of one. In Figure 3, strings of the two non-inputted modified bunsetsus are predicted as "*riyu-de* (on account of)" and "*toriyame-ta* (canceled)," respectively.

After the two annotations, the annotator clicks the button "Next." Then, a new bunsetsu is displayed, and the annotator repeats the two annotations until the sentence-end bunsetsu is displayed.

(3) Confirmation of annotation results: After completing the annotation of a sentence, the score of the annotation results and the correct dependency structure is displayed. The annotator compares their own annotation results with the correct answer and confirms the writing style of newspaper articles or transcripts of lectures, the specification of dependency grammar, and so on. Displaying the score is performed to maintain the motivation of annotators.

	annotator A		annotator B
corpus	SIDB	Kyoto	Kyoto
dependencies	216,204	262,426	262,426
strings	17,762	22,723	27,512

Table 1: The number of dependencies and strings of annotator A and annotator B in the annotation results.

3.3 Annotation Results

Table 1 presents the number of dependencies and strings in the annotation results. The annotation is iteratively performed for the already inputted sequence of bunsetsus whenever a bunsetsu in a sentence is displayed. In Table 1, we counted the dependencies and strings many times each time of the iteration. Additionally, the counted strings were only ones, which the annotators predicted and actually typed into the text boxes.

4 Analysis of Human Performance on Incremental Language Processing

We revealed aspects of human performance on incremental language processing based on analyses of the constructed data. Our analyses are based on two perspectives: dependency parsing and string prediction.

4.1 Human Performance on Dependency Parsing

We evaluated human performance on dependency parsing in terms of the following three points.

- **Sentence-based parsing:** We measured the agreement rate between the correct dependency structure and the dependency structure with which an annotator annotated a whole sentence after a sentence-end bunsetsu was displayed.
- **Incremental parsing I:** We evaluated the dependency structure provided by an annotator by seeing it as the dependency structure of the format of Figure 1. In other words, we ignore the information on whether or not other modifier bunsetsus depend on the same modified bunsetsu in the evaluation of dependency relations whose modified bunsetsu has not been inputted.
- **Incremental parsing II:** We evaluated the dependency structure provided by an annotator by seeing it as the dependency structure of the format of Figure 2. First, we establish correspondences between modified bunsetsus of the annotation results and modified bunsetsus of the correct data so that the agreement rate on dependency relations becomes the highest. After that, we measure the agreement rate.

4.1.1 Analytical Findings of Human Performance on Dependency Parsing

Table 2 shows the accuracy of dependency parsing by the two annotators at each evaluation point described above. The second, fourth, and sixth columns present the **dependency accuracy**[3], defined as the percentage of correctly analyzed dependencies out of all dependencies. The third, fifth, and seventh columns present **sentence accuracy**, defined as the percentage of sentences in which all dependencies are correctly analyzed. Table 2 shows that the incremental parsing II is the most difficult evaluation point compared to the other two parsing. This is easy to imagine because, in incremental parsing II, it is necessary to identify the greatest amount of information compared to other parsing methods.

We also separately measured the recall, precision, and f-measure of incremental parsing I and II for the case that the modified bunsetsu was inputted or not. The results are shown in Table 3. This table indicates that although it is less difficult for a human to identify that the modified bunsetsu has not been inputted, it becomes very difficult for a human to identify the dependency relationships between the inputted bunsetsus and the non-inputted ones.

Furthermore, we assessed the inter-annotator agreement between annotators A and B using the Kappa coefficient. The Kappa coefficients for sentence-based parsing, incremental parsing I, and incremental parsing II were 0.54, 0.51, and 0.43, respectively. According to Landis and Koch (1977), $0.41 \leq \kappa \leq 0.60$ indicates moderate agreement. The agreement gradually decreased as the evaluation point became more difficult. Additionally, the difference between the values of sentence-based parsing and incremental parsing I is smaller than the difference between those of incremental parsing I and II. This indicates that the performance of identifying dependency relationships between the inputted bunsetsus and the non-inputted bunsetsus

[3]Dependency accuracies of sentence-based parsing, incremental parsing I and II are measured based on the accuracies defined in the literatures (Uchimoto et al., 1999; Ohno and Matsubara, 2013).

	annotator A				annotator B	
corpus	SIDB		Kyoto		Kyoto	
eval. metrics	dependency	sentence	dependency	sentence	dependency	sentence
sentence-based	0.897	0.462	0.947	0.633	0.950	0.654
incremental I	0.887	0.395	0.945	0.546	0.942	0.489
incremental II	0.852	0.229	0.918	0.315	0.896	0.186

Table 2: Accuracy of two annotators' dependency parsing, evaluated by dependency and sentence accuracy.

		annotator A						annotator B		
corpus		SIDB			Kyoto			Kyoto		
eval. metrics		R	P	F	R	P	F	R	P	F
incremental parsing I	inputted	0.891	0.884	0.887	0.958	0.940	0.949	0.953	0.943	0.948
	non-inputted	0.923	0.900	0.911	0.959	0.957	0.958	0.960	0.939	0.949
incremental parsing II	inputted	0.891	0.884	0.887	0.958	0.940	0.949	0.953	0.943	0.948
	non-inputted	0.794	0.774	0.784	0.867	0.865	0.866	0.804	0.786	0.795

Table 3: Recall (R), precision (P), and the f-measure (F) of two annotators' incremental dependency parsing, separately for the case that the modified bunsetsu has not been inputted, and the case that the one has been inputted.

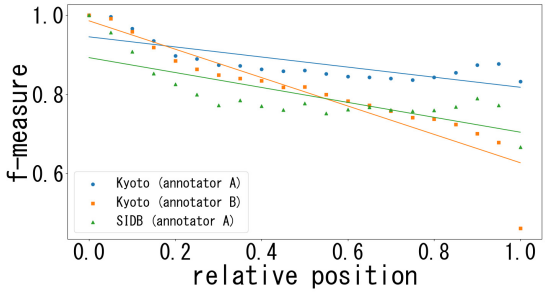

Figure 4: F-measure of incremental parsing II (non-inputted) by relative position.

varies significantly greatly from person to person.

4.1.2 Effect of the Number of Inputted Bunsetsus on Incremental Parsing

The difficulty of incremental parsing is expected to vary depending on the number of inputted bunsetsus. This section examines the effect of the number of inputted bunsetsus on incremental parsing. To account for the length of the entire sentence, we defined a relative position as the number of inputted bunsetsu divided by the total number of bunsetsu in the entire sentence. We classified the annotation results based on the relative positions of each bunsetsu input in 0.05 increments and then calculated the F-measure of incremental parsing II (non-inputted) for each class.

Figure 4 shows the f-measures of incremental parsing II by relative position. The straight lines represent the regression lines. The figure shows that the f-measure declined as the relative position increased. However, for the Kyoto Corpus and SIDB annotated by annotator A, the f-measure increased when the relative position exceeded 0.8. There are two factors that explain these trends. First, as the number of inputted bunsetsu increased, the number of possible modified bunsetsu increased. Therefore, dependency parsing becomes more complicated. Second, as more bunsetsu were inputted, the understanding of the sentence improved, making it easier to identify correct non-inputted modified bunsetsu. When the relative position was below 0.8, the f-measure was lower due to the stronger influence of the first factor. In contrast, the f-measure was higher when the relative position exceeded 0.8 due to the stronger influence of the second factor.

4.2 Human Performance on String Prediction

We evaluated human performance on string prediction. Specifically, we evaluated how accurately the two annotators predicted the string of a non-inputted bunsetsu.

4.2.1 Analytical Findings of Human Performance on String Prediction

Table 4 shows the recall and precision values of string prediction. Recall is the percentage of correctly predicted bunsetsus out of all bunsetusus in the correct dependency structure. Precision is the percentage of correctly predicted bunsetsus out of all bunsetsus whose strings are predicted by an-

	annotator A				annotator B	
corpus	SIDB		Kyoto		Kyoto	
eval. point	exact	partial	exact	partial	exact	partial
recall	0.043	0.125	0.057	0.117	0.036	0.119
precision	0.086	0.249	0.128	0.262	0.067	0.219

Table 4: Accuracies of string prediction by only exact match (exact) and including partial match (partial).

POS	Kyoto	SIDB
noun	27.17	20.07
verb	61.88	64.59
adjective	6.15	4.75
adverb	0.52	0.33
pre-noun adj	0.02	0.04
conjunction	0.10	0.05
interjection	0.01	0.07
copula	4.09	9.82
demonstrative	0.06	0.29

Table 5: Percentage distribution of POS in the head of modified bunsetsus.

	POS	R	P
Kyoto (annotator A)	verb	0.070	0.265
	noun	0.098	**0.267**
	all	**0.117**	0.262
Kyoto (annotator B)	verb	0.074	**0.240**
	noun	0.097	0.234
	all	**0.119**	0.219
SIDB (annotator A)	verb	0.070	**0.265**
	noun	0.105	0.254
	all	**0.125**	0.249

Table 6: Recall (R) and precision (P) of verb and noun for partial match.

notators. The "exact" columns show the results for which the prediction was correct only if they exactly matched the correct string. The "partial" columns show the results where the prediction was correct if they either exactly or partially matched[4] the correct string. The results indicate that predicting strings of non-inputted bunsetsus is challenging, even for humans. But at the same time, it suggests that humans have the ability to predict strings of some non-inputted bunsetsus.

Next, we evaluated the inter-annotator agreement on the string prediction between annotator A and B. The κ values of string prediction were 0.27 and 0.29 for an exact and partial match, respectively. According to Landis and Koch (1977), $0.21 \leq \kappa \leq 0.40$ indicates fair agreement. The agreement is lower than that of dependency parsing. Therefore, we can see that the performance of string prediction varies more greatly from person to person.

4.2.2 Analysis of String Prediction by POS of Non-inputted Modified Bunsetsus

We investigated how string prediction accuracy varies with the POS of the head[5] of the non-inputted modified bunsetsu in the correct structure. First, we examined the percentage distribution of POS in the head of the modified bunsetsu. Table 5 shows that verbs and nouns made up over 80% of the total, indicating that many modified bunsetsu heads were verbs or nouns. Therefore, we focused on verbs and nouns in this section.

Table 6 shows the performance of string prediction for verbs and nouns. The recall of verbs and nouns was lower than the micro-recall of all POS. This is because the frequency of occurrence for verbs and nouns is higher, leading to more instances where the number of inputted bunsetsu was insufficient to predict string of non-inputted modified bunsetsu, compared to other POS. However, the precision of verbs and nouns was higher than the micro-precision of all POS. This means that humans can more easily predict a string of the non-inputted modified bunsetsu whose head is a verb or noun than another POS when focusing on the strings annotated by the annotators.

4.2.3 Effect of the Number of Inputted Bunsetsus on String Prediction

We assumed that the closer a newly inputted bunsetsu is to the end of the sentence, the more context is available, and the string prediction accuracy will increase. To examine the assumption, we analyzed the effect of relative position on string prediction

[4] A partial match is judged when a predicted string includes the surface of the head of the correct string.

[5] Each bunsetsu has a head, which serves as the primary expression of its content and is determined with reference to the definition by Uchimoto et al. (1999).

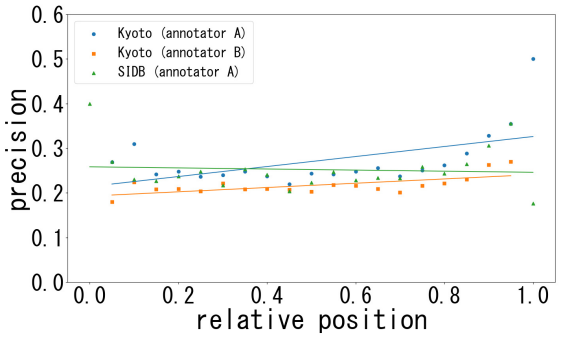

Figure 5: Precision of string prediction for partial match by relative position.

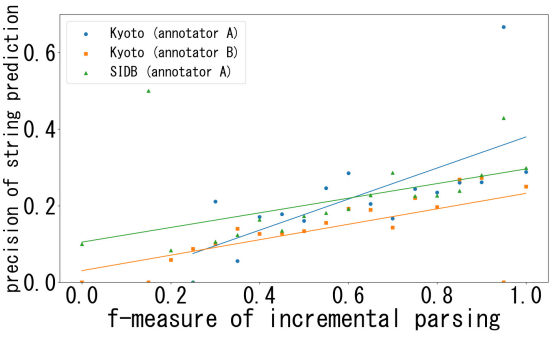

Figure 6: Precision of string prediction for partial match by each f-measure of incremental parsing Ⅱ (non-inputted).

in a similar manner to Section 4.1.2. Here, we used precision for partial match as the evaluation index of string prediction to focus on the strings predicted by the annotators.

Figure 5 shows the precision of string prediction. The blue and orange lines represent the regression lines for the Kyoto Corpus annotated by annotators A and B, respectively; the green line represents the regression line for the SIDB annotated by annotator A. The regression lines show a positive slope, except for the SIDB (annotator A). Notably, the string prediction precision increased sharply when the relative position exceeded 0.8. This phenomenon can be attributed to the structural characteristics of the Japanese language. As a subject-object-verb language, Japanese often places verbs at the end of sentences. We assume that precision rapidly increases because annotators can predict a sentence-end verb using richer contexts when approaching the end of a sentence.

The results demonstrate that humans can make string predictions more accurately as the number of inputted bunsetsu increases, particularly as the sentence approaches its end.

4.3 Relationship between Incremental Dependency Parsing and String Prediction

Both incremental dependency parsing and string prediction in common require prediction of non-inputted parts of a sentence based on contextual understanding. We have the intuition that humans simultaneously perform these two tasks while predicting non-inputted parts and thus the two tasks are related to each other. In this section, we investigate the relationship between string prediction and incremental parsing Ⅱ (incremental dependency parsing).

Figure 6 illustrates the relationship between the f-measure of incremental parsing Ⅱ (non-inputted) on the x-axis and the precision of string prediction (partial) on the y-axis, measured each time a new bunsetsu was inputted and rounded in 0.05 increments. The blue and orange lines represent the regression line for Kyoto Corpus annotated by annotator A and B, respectively; the green line represents the regression line for the SIDB annotated by annotator A. The regression lines have positive slopes, indicating a positive correlation between the two tasks. This suggests that when humans accurately parse dependencies, they also tend to predict strings of modified bunsetsu more accurately.

5 Conclusion

In this study, we presented the annotation results, capturing human performance in incremental language processing. The annotators performed incremental dependency parsing and string prediction of some non-inputted bunsetsus whenever a new bunsetsu was inputted. Using this annotated data, we analyzed human performance in incremental dependency parsing and string prediction of non-inputted modified bunsetsus.

In the future, we intend to conduct a more detail analysis of the constructed data to further understand human performance in incremental dependency parsing. For example, we aim to investigate factors such as the content of inputted bunsetsus and the context, which could potentially influence incremental dependency parsing and string prediction.

Acknowledgments

This work was supported by JSPS KAKENHI Grand Number JP19K12127, JP24K15076.

References

Shanqing Cai, Subhashini Venugopalan, Katrin Tomanek, Ajit Narayanan, Meredith Morris, and Michael Brenner. 2022. Context-aware abbreviation expansion using large language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2022)*, pages 1261–1275.

Iker García-Ferrero, Begoña Altuna, Javier Alvez, Itziar Gonzalez-Dios, and German Rigau. 2023. This is not a dataset: A large negation benchmark to challenge large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP 2023)*, pages 8596–8615.

Alvin Grissom II, He He, Jordan Boyd-Graber, John Morgan, and Hal Daumé III. 2014. Don't until the final verb wait: Reinforcement learning for simultaneous machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 1342–1352.

Richard Johansson and Pierre Nugues. 2007. Incremental dependency parsing using online learning. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007)*, pages 1134–1138.

Yoshihide Kato and Shigeki Matsubara. 2009. Incremental parsing with monotonic adjoining operation. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2009) Short Papers*, pages 41–44.

Yoshihide Kato, Shigeki Matsubara, Katsuhiko Toyama, and Yasuyoshi Inagaki. 2001. Efficient incremental dependency parsing. In *Proceedings of the 7th International Workshop on Parsing Technologies (IWPT 2001)*, pages 225–228.

Yoshihide Kato, Shigeki Matsubara, Katsuhiko Toyama, and Yasuyoshi Inagaki. 2005. Incremental dependency parsing based on headed context-free grammar. *Systems and Computers in Japan*, 36:63–77.

Daisuke Kawahara, Sadao Kurohashi, and Kôiti Hasida. 2002. Construction of a Japanese relevance-tagged corpus. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC 2002)*, pages 2008–2013.

Arne Köhn and Wolfgang Menzel. 2014. Incremental predictive parsing with TurboParser. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*, pages 803–808.

Kentaro Kurihara, Daisuke Kawahara, and Tomohide Shibata. 2022. JGLUE: Japanese general language understanding evaluation. In *Proceedings of the 13th Language Resources and Evaluation Conference (LREC 2022)*, pages 2957–2966.

J. Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33 1:159–74.

Hwaran Lee, Seokhee Hong, Joonsuk Park, Takyoung Kim, Meeyoung Cha, Yejin Choi, Byoungpil Kim, Gunhee Kim, Eun-Ju Lee, Yong Lim, Alice Oh, Sangchul Park, and Jung-Woo Ha. 2023. SQuARe: A large-scale dataset of sensitive questions and acceptable responses created through human-machine collaboration. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL 2023)*, pages 6692–6712.

Dan Liu, Mengge Du, Xiaoxi Li, Ya Li, and Enhong Chen. 2021. Cross attention augmented transducer networks for simultaneous translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP 2021)*, pages 39–55.

Masaki Murata, Tomohiro Ohno, and Shigeki Matsubara. 2010. Automatic comma insertion for Japanese text generation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP 2010)*, pages 892–901.

Tu Anh Nguyen, Eugene Kharitonov, Jade Copet, Yossi Adi, Wei-Ning Hsu, Ali Elkahky, Paden Tomasello, Robin Algayres, Benoît Sagot, Abdelrahman Mohamed, and Emmanuel Dupoux. 2023. Generative spoken dialogue language modeling. *Transactions of the Association for Computational Linguistics*, 11:250–266.

Joakim Nivre. 2008. Algorithms for deterministic incremental dependency parsing. *Computational Linguistics*, 34(4):513–553.

Tomohiro Ohno and Shigeki Matsubara. 2013. Dependency structure for incremental parsing of Japanese and its application. In *Proceedings of the 13th International Conference on Parsing Technologies (IWPT 2013)*, pages 91–97.

Tomohiro Ohno, Masaki Murata, and Shigeki Matsubara. 2009. Linefeed insertion into Japanese spoken monologue for captioning. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2009)*, pages 531–539.

Stelios Piperidis, Iason Demiros, Prokopis Prokopidis, Peter Vanroose, Anja Hoethker, Walter Daelemans, Elsa Sklavounou, Manos Konstantinou, and Yannis Karavidas. 2004. Multimodal, multilingual resources in the subtitling process. In *Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004)*, pages 205–208.

Machel Reid, Victor Zhong, Suchin Gururangan, and Luke Zettlemoyer. 2022. M2D2: A massively multi-domain language modeling dataset. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP 2022)*, pages 964–975.

Amilleah Rodriguez, Shaonan Wang, and Liina Pylkkänen. 2024. Do neural language models inferentially compose concepts the way humans can? In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 5309–5314.

Koichiro Ryu, Shigeki Matsubara, and Yasuyoshi Inagaki. 2006. Simultaneous English-Japanese spoken language translation based on incremental dependency parsing and transfer. In *Proceedings of the Joint Conference of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (COLING-ACL 2006) Poster Sessions*, pages 683–690.

Anastassia Shaitarova, Anne Göhring, and Martin Volk. 2023. Machine vs. human: Exploring syntax and lexicon in German translations, with a spotlight on anglicisms. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa 2023)*, pages 215–227.

Hitomi Tohyama, Shigeki Matsubara, Nobuo Kawaguchi, and Yasuyoshi Inagaki. 2005. Construction and utilization of bilingual speech corpus for simultaneous machine interpretation research. In *Proceedings of Interspeech 2005*, pages 1585–1588.

Kazuki Tsunematsu, Johanes Effendi, Sakriani Sakti, and Satoshi Nakamura. 2020. Neural Speech Completion. In *Proceedings of Interspeech 2020*, pages 2742–2746.

Kiyotaka Uchimoto, Satoshi Sekine, and Hitoshi Isahara. 1999. Japanese dependency structure analysis based on maximum entropy models. In *Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics (EACL 1999)*, pages 196–203.