Dependency-Aware Word Prediction Integrated with Incremental Parsing

Hiroki Unno, Tomohiro Ohno, Koichiro Ito, Shigeki Matsubara

Proceedings of the 39th Pacific Asia Conference on
Language, Information and Computation (PACLIC 39)

Emmanuele Chersoni, Jong-Bok Kim (eds.)

2025

Dependency-Aware Word Prediction Integrated with Incremental Parsing

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

This paper proposes a method for dependency-aware word prediction performed simultaneously with word input to enhance the performance of real-time natural language processing tasks, such as complementary response generation in dialogue systems and simultaneous machine interpretation. The characteristics of our method are as follows: 1) the targets of the word prediction are not the immediate next words but non-inputted words that have a dependency relation with any of the inputted words, and 2) the word prediction is integrated with incremental dependency parsing. We performed experiments on predicting non-inputted words that have a dependency relation with any of the inputted words, and compared the results with human performance, which confirmed the feasibility of our method. Furthermore, to verify the usefulness of our method for complementary response generation, we evaluated the agreement between actual complementary responses and the words predicted by our method. In addition, we compared the results with those obtained by a large language model (LLM). The results demonstrated that our method can predict main parts of actual complementary responses with higher performance than the LLM, which indicates that our method can provide informative cues with little noise for the complementary response generation.

1 Introduction

Several natural language processing tasks, such as dialogue systems (Nakano et al., 2000; Chiba and Higashinaka, 2025; Liu et al., 2022) and simultaneous machine interpretation (Wang et al., 2024; Ryu et al., 2006; Gu et al., 2017), require real-time responses, and a common requirement of such systems is to execute processing simultaneously with time-continuous input of sentence components. Previous studies have investigated ways to improve the performance of these real-time tasks by predicting non-inputted parts of a sentence through next word prediction (Alinejad et al., 2018; Tsunematsu et al., 2020; Ouyang et al., 2025; Ekstedt and Skantze, 2021).

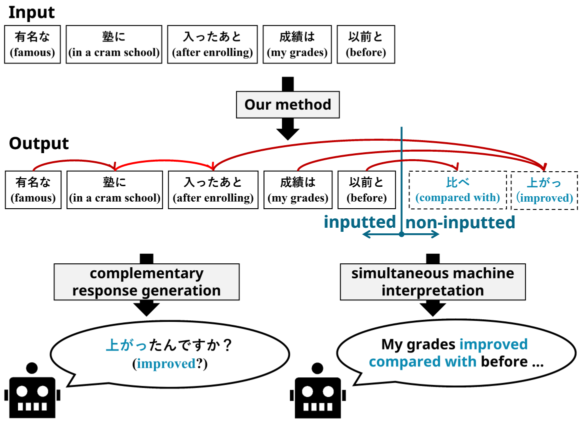

Figure 1: Examples demonstrating the effectiveness of predicting non-inputted words that have dependency relations with any of the inputted words.

In contrast, among these tasks, especially in tasks such as complementary response generation for Japanese and simultaneous machine interpretation between languages with divergent word order (e.g., Japanese-English), situations arise where predicting words that have a dependency relation with any of the inputted words is more beneficial than next word prediction (Oda et al., 2015). For example, in complementary response generation, it is necessary to concisely complement the speaker's utterance; thus, it is important to predict non-inputted words that have a dependency relation with any of the inputted words. In addition, in simultaneous Japanese-English interpretation, predicting predicates that have a dependency relation with any of the inputted words at an early stage enables faster interpretation (Matsubara et al., 2000; Grissom II et al., 2016; Li et al., 2020). As shown in Figure 1, predicting a non-inputted word “上が

つ” (improved), which has dependency relations with the inputted words “入った後” (after enrolling) and “成績は” (my grades), allows systems to generate a complementary response or perform machine interpretation simultaneously. However, to the best of our knowledge, no previous study has focused on non-inputted words that have a dependency relation with any of the inputted words as prediction targets.

Thus, this paper proposes a method for dependency-aware word prediction, i.e., the prediction of non-inputted words that have a dependency relation with any of the inputted words, simultaneously with word input for Japanese. In our method, the word prediction is integrated with incremental dependency parsing, which identifies dependency relations between the inputted and non-inputted words, and both processes are performed jointly in an end-to-end manner. As shown in Figure 1, given a sequence of inputted words, our method identifies dependency relations and the non-inputted words involved in those relations. This approach is based on the following hypothesis: when humans predict non-inputted parts of a sentence, they implicitly anticipate both the syntactic structure and the non-inputted words simultaneously.

The remainder of this paper is organized as follows. Section 2 describes related work. Section 3 explains our method in detail. Section 4 describes experiments conducted to evaluate the performance of our method, and Section 5 examines how accurately the method can predict the heads of complementary responses. Finally, the paper is concluded in Section 6.

2 Related Work

2.1 Next Word Prediction

Numerous previous studies have investigated ways to improve the performance of real-time natural language processing tasks by predicting the non-inputted parts of a sentence based on next word prediction (Alinejad et al., 2018; Tsunematsu et al., 2020; Ouyang et al., 2025; Ekstedt and Skantze, 2021). For example, focusing on real-time tasks, Tsunematsu et al. (2020) were the first to formalize the task of completing the remaining word sequence given the first 25%, 50%, and 75% of a sentence, and they proposed a method for this completion. In addition, in the simultaneous machine interpretation context, Alinejad et al. (2018) proposed a method to predict the next word of an inputted sequence of words based on an RNN, and Ouyang et al. (2025) proposed a method to predict multiple word sequence candidates coming after the inputted word sequence based on a large language model (LLM). Focusing on dialogue systems, Ekstedt and Skantze (2021) proposed a method to predict a specified number of words following the inputted word sequence by GPT-2.

2.2 Application-Side Requests

As discussed in Section 1, in various tasks, including complementary response generation and simultaneous interpretation between languages with different word orders, it is particularly important to predict non-inputted words that have a dependency relation with any of the inputted words. However, the above-mentioned approaches based on next word prediction merely predict a sequence of non-inputted words by repeatedly predicting the next word, without explicitly identifying which of the predicted words has a dependency relation with any of the inputted words. In addition, the approach based on next word prediction has a known issue, where the prediction accuracy tends to degrade due to error propagation (Zhang et al., 2023; Qian et al., 2025).

To address these issues, this study takes an approach that directly predicts non-inputted words that have a dependency relation with any of the inputted words. To the best of our knowledge, no previous study has focused on non-inputted words that have a dependency relation with any of the inputted words as the prediction targets. However, as a related approach, there exist some studies that have proposed methods to predict the final verb of a sentence (Matsubara et al., 2000; Grissom II et al., 2016; Li et al., 2020). For example, Matsubara et al. (2000) performed early prediction of verbs based on noun phrases and reported the effectiveness of this method in simultaneous interpretation. Additionally, Grissom II et al. (2016) compared human performance with a statistical model in the verb prediction using incomplete Japanese and German sentences, and Li et al. (2020) proposed a method to predict the final verb using a neural model. These methods focus solely on the final verb; however, our approach differs from these existing approaches because it aims to predict all non-inputted words that have a dependency relation with any of the inputted words.

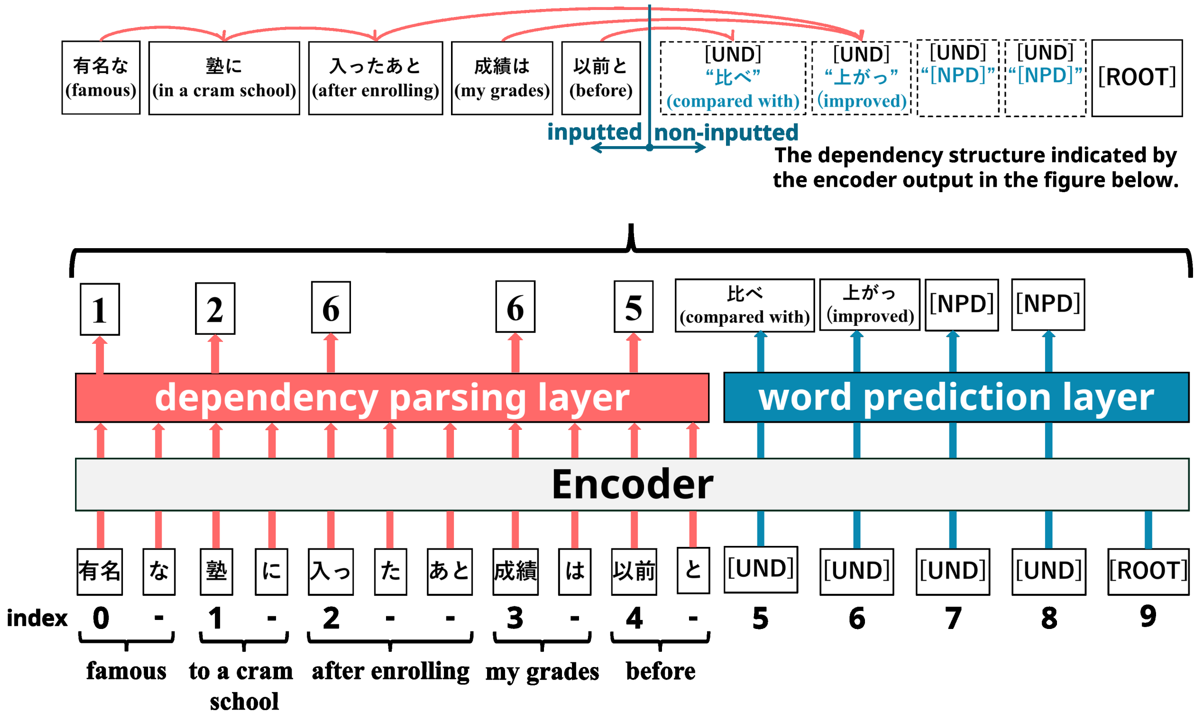

Figure 2: Overview of our method.

3 Word Prediction Integrated with Incremental Dependency Parsing

This section describes our method to predict non-inputted words, i.e., head words[1] of non-inputted *bunsetsus* that have a dependency relation with any of the inputted bunsetsus, for Japanese sentences. A bunsetsu is a linguistic unit in Japanese that roughly corresponds to a basic phrase in English. A bunsetsu comprises one independent word and zero or more ancillary words. In Japanese, a dependency relation is a modification relation in which a modifier bunsetsu is dependent on a modified bunsetsu, which is expressed as "a modifier bunsetsu → a modified bunsetsu" in this paper.

3.1 Overview of Our Method

Our method end-to-end identifies a structure, as shown in the output of Figure 1 whenever a bunsetsu b_i $(1 \leq t < n)$ is inputted for a sentence comprising n bunsetsus $b_1 \cdots b_n$. In other words, our method identifies not only the dependency structure, which also includes the dependency relations between inputted and non-inputted bunsetsus, but also each head word of the non-inputted modified bunsetsus[2].

An overview of our method is shown in Figure 2. The input to the encoder is the inputted bunsetsu sequence $B_t = b_1 \cdots b_t$. More precisely, the token sequence of B_t is inputted with additional special tokens [UND] and [ROOT], which are described later. We append two independent fully connected layers to the final layer of the encoder. Here, the first fully connected layer parses the modified bunsetsu of each inputted bunsetsu, and the second fully connected layer predicts the head word of each non-inputted modified bunsetsu.

The model was trained using multi-task learning, which jointly optimizes word prediction and dependency parsing. In addition, the total loss is calculated as the weighted sum of the losses for each task with the weighting coefficient λ as follows:

$$Loss = \lambda \times Loss_{\text{word\_prediction}} + (1 - \lambda) \times Loss_{\text{dependency\_parsing}} \quad (1)$$

Here, we adopt the cross-entropy loss as the loss for each task.

[1] Our definition of a head follows that of Uchimoto et al. (1999), except that auxiliary verbs are excluded. In addition, non-independent words are generally not considered head words but are accepted as the heads when no other candidate exists within the bunsetsu.

[2] Non-inputted modified bunsetsus are those that have dependency relations with any of the inputted bunsetsus because we assume that no dependency is directed from right to left, which is almost true in Japanese.

3.2 Word Prediction

Following the previous study (Yoshida and Kawahara, 2022), our method introduces a special token [UND], which denotes an undetermined token as a head word of a non-inputted modified bunsetsu. This special token is added as many as the number of inputted bunsetsus, assuming each inputted bunsetsu is dependent on a different non-inputted bunsetsu.

To predict each [UND] token, the model computes the probability distribution over words in the vocabulary and selects the word with the highest probability. Here, to process a special token [UND] that has no dependency relation with any of the inputted bunsetsus, we introduce a special token [NPD] and train the model to output [NPD] as the token for such [UND] tokens. This design is intended to prevent unnecessary word predictions.

3.3 Incremental Dependency Parsing

Following Shibata et al. (2019), our method formulates dependency parsing as a head-selection problem (Zhang et al., 2017). Specifically, for each head word of an inputted bunsetsu, the model predicts the index of the corresponding modified bunsetsu. We model the dependency relations between bunsetsus through word-level, more precisely, token-level parsing.

We add a [ROOT] token to the end of the input sequence and set [ROOT] as the modified bunsetsu for the head word of the sentence-level root bunsetsu. The index of a [UND] token, which represents a non-inputted bunsetsu, is selected when the model decides that the modified bunsetsu has not yet been inputted.

4 Experiment

To evaluate the feasibility of our method to predict non-inputted modified bunsetsus, we conducted experiments using Japanese lecture speech transcripts and compared the performance of our method with that of a human.

4.1 Dataset

In this study, we used Japanese lecture speech from the Simultaneous Interpretation Database (SIDB) (Matsubara et al., 2002). This dataset is annotated with morphological tags, bunsetsu boundaries, clause boundaries, and dependency relations, all of which have been corrected manually. Note that the morphological tags are annotated based on the IPA dictionary.

We performed 16-fold cross-validation to evaluate the performance of our method. Here, in each fold, one of the 16 lectures served as the test set, and the remaining 15 lectures were used for training. This procedure was repeated once for each lecture. We used two of the 16 lectures as a development set and evaluated the model's performance on the remaining 14 lectures.

We created the training data incrementally. In other words, whenever a new bunsetsu was inputted, we generated a single instance corresponding to the inputted bunsetsu sequence to create the training data. Thus, each sentence produced as many instances as it has bunsetsus. Using all instances to train our model may lead to overfitting. In this experiment, following Yoshida and Kawahara (2022), we prevented overfitting by limiting the incrementally created data used for training to 5% of all generated instances.

4.2 Evaluation Metrics

For performance evaluation, we compared the performance of our model with that of a human (Unno et al., 2024). Here, a single annotator predicted the non-inputted modified bunsetsus and performed incremental dependency parsing on the test set. In other words, the annotator predicted the structure shown in Figure 1.

We evaluated the word prediction performance for non-inputted modified bunsetsus using recall, precision, and F1 score. Here, recall is defined as the percentage of modifier bunsetsus whose modified bunsetsu's head word was predicted correctly out of all modifier bunsetsus whose modified bunsetsus were non-inputted in the gold dependency structure. Precision is defined as the percentage of modifier bunsetsus whose modified bunsetsu's head word was predicted correctly out of all modifier bunsetsus whose modified bunsetsus were non-inputted in the parsed dependency structure. Figure 3 shows an example of the evaluation metrics calculation for word prediction. Even if the top-1 prediction was incorrect, the appearance of the correct word in the higher-ranked list can benefit downstream modules or human users. Thus, we adopted top-k ($k = 1, 2, 3, 4, 5$) and evaluated performance by determining whether the correct words were included in the top-k outputs of our method.

To evaluate the incremental dependency pars-

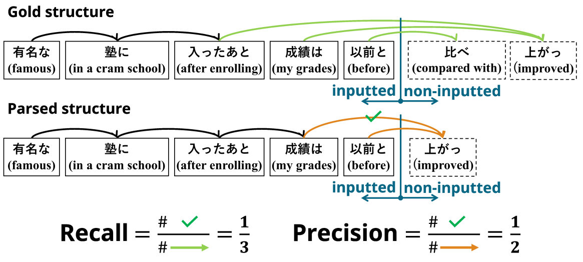

Figure 3: Example of evaluation metrics calculation for word prediction.

Table 1: Results for word prediction of non□inputted modified bunsetsus.

	Recall	Precision	F1
ours (top-1)	7.76	8.04	7.90
ours (top-2)	11.46	11.87	11.66
ours (top-3)	14.42	14.94	14.67
ours (top-4)	16.52	17.11	16.81
ours (top-5)	18.34	19.00	18.67
human (top-1)	11.26	10.91	11.08

ing, we classified each dependency according to whether its modified bunsetsu had already been inputted or was still non-inputted, and we computed the recall, precision, and F1 score values separately for the two categories. Here, recall is defined as the percentage of correctly parsed dependency relations out of all dependency relations in the gold dependency structure. Precision is defined as the percentage of correctly parsed dependency relations out of all dependency relations in the parsed dependency structure.

4.3 Implementation

We implemented our model using PyTorch[3], and we employed the publicly available pre-trained Japanese BERT model released by Tohoku University[4] as the encoder. As a result of hyperparameter tuning on the development set, we set the batch size to 4, the learning rate to 1e-5, the number of training epochs to 10, and the loss-weighting coefficient λ to 0.6. For this evaluation, we varied the random seed, trained five models, and computed the average value of each evaluation metric.

4.4 Experimental Results

4.4.1 Results for Word Prediction

Table 1 shows the evaluation results for word prediction of non-inputted modified bunsetsus. As can be seen, the F1 score obtained by our method increased consistently as the number of candidates increased from top-1 to top-5. In addition, from top-2 to top-5, our method achieved F1 scores that exceeded the human-level performance, which confirms the feasibility of our method.

In contrast, with our method, the F1 score of the top-1 was 3.28 points lower than that of the human, which indicated that our method still falls short of human-level accuracy when limited to a single output. Possible improvements include refining the prediction mechanism and re-ranking candidate outputs. We leave these directions for future work.

4.4.2 Results for Incremental Dependency Parsing

Table 2 shows the evaluation results for incremental dependency parsing. As shown, our method achieved similar performance in cases where the modified bunsetsus were non-inputted (F1 score: 72.30) and where they were already inputted (F1 score: 70.53). These results demonstrate that our method can identify dependency relations involving for non-inputted bunsetsus.

Compared with human performance, the gap in F1 score was 6.19 points when the modified bunsetsu was non-inputted; however, this gap widened to 17.49 points when the modified bunsetsu was already inputted. This result confirms that there is room to improve our method.

4.5 Discussion

Our method formulates word prediction and incremental dependency parsing as two separate tasks, and it accomplishes both tasks in a single framework by solving them simultaneously through multi-task learning. However, the outputs of the two tasks are not always mutually consistent. As shown in Figure 4, two types of inconsistencies can occur. First, our method may output [NPD] as a head word of a non-inputted bunsetsu even though it has determined that the non-inputted bunsetsu has dependency relations with any of the inputted bunsetsus. Second, our method may output

[3] https://pytorch.org/

[4] https://huggingface.co/tohoku-nlp/bert-base-japanese-whole-word-masking

Table 2: Results for incremental dependency parsing.

	inputted			non-inputted		
	Recall	Precision	F1	Recall	Precision	F1
ours	71.35	69.73	70.53	71.05	73.60	72.30
human	87.77	88.28	88.02	79.74	77.29	78.49

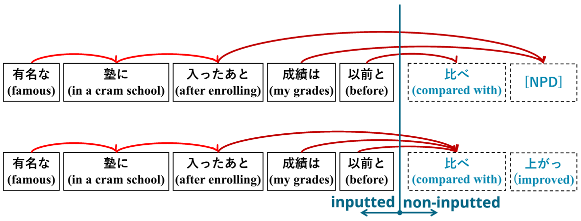

Figure 4: Examples where word prediction and incremental dependency parsing are inconsistent.

a token other than [NPD] as the head word for a non-inputted bunsetsu, that is determined to not have such a relation. In both cases, the outputs of the word prediction and incremental dependency parsing do not align with each other.

To examine the consistency of the outputs of the two tasks in our method, we calculated the following two percentages.

- The percentage of [UND] tokens predicted to be tokens other than [NPD] among those determined parsed to have dependency relations with any of the inputted bunsetsus.
- The percentage of [UND] tokens parsed to have dependency relations with any of the inputted bunsetsus among those predicted to be tokens other than [NPD].

We found that the former and latter percentages were 97.53% and 95.77%, respectively, which indicates that the outputs of the two tasks are mainly consistent. However, a portion remains inconsistent. Addressing such cases will be the focus of future work.

5 Application of Our Method

This section evaluates whether predicting non-inputted modified bunsetsus using the our method is helpful for downstream tasks. Here, we focus on complementary response generation in dialogue systems as a specific downstream task.

A complementary response is a type of responsive utterance that conveys attentive listening (attentive listening response) to a narrative and complements the speaker's narrative. Generating a complementary response requires understanding the content of the speaker's narrative and predicting what the speaker will say; thus, producing such responses is known to be highly effective in conveying an attentive listening attitude. Figure 5 shows an example of a complementary response. In this example, a listener has predicted the non-inputted word "犬派" (a dog person) on the basis that the adversative conjunction "けど" (but) is followed by the clause "猫派だった" (I used to be a cat person), and produced a complementary response. As shown in this example, the element that the listener predicts and provides as a complementary response (i.e., "犬派" (a dog person) in this case) is frequently not the immediate next word. More frequently, it is a modified bunsetsu that has a dependency relations with an inputted bunsetsu, e.g., "猫派だったんですけど" (I used to be a cat person, but) in this figure, and appears later in the speaker's narrative, after at least one intervening word (refer to Appendix A). Thus, in complementary response generation, our method to predict non-inputted modified bunsetsu is expected to be effective.

Accordingly, we assessed the effectiveness of our method for complementary response generation by examining how accurately it predicts the head word of the final bunsetsu that forms the complementary response (referred to as the head word of the complementary response). Specifically, we evaluated the agreement between the head words of the non-inputted modified bunsetsu predicted by our method and the head words of the complementary responses. We demonstrate that our method predicts the head words of the complementary responses with higher accuracy by comparing with an LLM-based method that predicts the non-inputted parts by repeating the next word prediction.

5.1 Responsive Utterance Corpus

In this experiment, we used the Responsive Utterance Corpus (Ito et al., 2022)[5], which contains

[5] We did not use the Responsive Utterance Corpus (Ito et al., 2022) in the experiments discussed Section 4 because

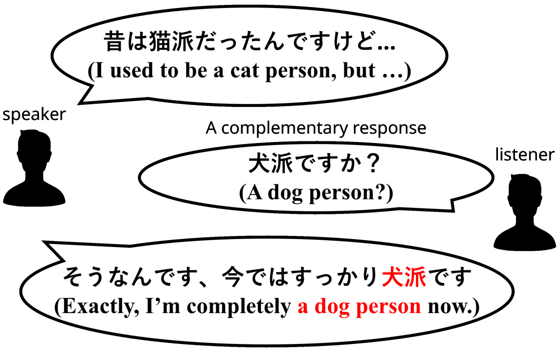
昔は猫派だったんですけど…
(I used to be a cat person, but …)
speaker
A complementary response
犬派ですか？
(A dog person?)
listener
そうなんです、今ではすっかり犬派です
(Exactly, I'm completely a dog person now.)

Figure 5: Example of a complementary response.

Prompt
System Prompt: A partial narrative is provided as input. Predict the sentence that follows this narrative. Output only your prediction. Do not repeat the input narrative. Do not add any character decoration or markup. User Prompt: {narrative}

Figure 6: Prompt used in the experiment (English translation).

148,952 attentive listening responses produced by 11 annotators while listening to the narrative audio. JELico (Aramaki, 2016), which is a corpus of narratives produced by elderly people, was employed for the narrative audio. The collected attentive listening responses are classified into 16 categories, including complementary responses, and the corpus contains a total of 614 complementary responses. Note that annotators produced their responses offline while listening to the prerecorded narrative audio; thus, the content of those responses could not influence the subsequent progression of the narratives. As a result, the speaker never omits or withholds the content of the complementary responses.

5.2 Experimental Settings

In this experiment, we divided the 2,156 narrative sentences in the Responsive Utterance Corpus into 447 sentences for testing, 446 sentences for development, and 1,263 sentences for training. Note that the annotators of the corpus did not always produce a complementary response for each sentence. The 447 test sentences contain 141 complementary responses, and the 446 development sentences contain 123. In this experiment, we treated each point at which a complementary response occurred as a prediction point. Here, we used the narrative from the beginning of the sentence up to the point where the complementary response was produced as the input at each point. Both our method and the LLM-based method predicted the non-inputted parts. For this evaluation, we calculated the agreement between the predicted head words and the head words of the complementary responses for each prediction method.

To train our method, we further fine-tuned the model trained with SIDB in the experiment described in Section 4, using the training part of the Responsive Utterance Corpus. We generated data instances for each of the 1,263 training sentences[6] using the procedure described in Section 4.1, and we trained the model on these data. Based on tuning with the development data, we set the hyperparameters of our method to a batch size of 5, learning rate of 1e-5, three training epochs, and a loss-weight coefficient λ of 0.7.

5.3 LLM-based Method

We employed GPT-4.1[7] as the compared method and instructed it to predict the sentence in the narrative that follows the point at which the complementary response occurs. Table 6 shows the prompt used in this evaluation. Here, we first processed each sentence generated by the LLM using CaboCha (Kudo and Matsumoto, 2003) to apply morphological analysis and bunsetsu segmentation. Then, we determined the head word of each bunsetsu from the resulting parse.

5.4 Evaluation Metrics

We used two evaluation metrics, i.e., recall and precision. Here, recall is defined as the proportion of the 141 complementary responses in the test set whose head words were predicted correctly. A prediction was considered correct if at least one of the head words predicted by each method for the non-inputted bunsetsus matched the head word in the complementary response. Precision is defined as the proportion of the predicted non-inputted bunsetsus whose head word matches the head word of the complementary response. In the evaluation, we used five predictions results obtained by each

it lacks manually annotated gold-standard dependency information.

[6] Here, we used the dependency information parsed by using CaboCha because this data lacks the manual annotation.

[7] https://openai.com/index/gpt-4-1/

Table 3: Results for predicting head word of the complementary responses.

Method	Recall					Precision				
	top-1	top-2	top-3	top-4	top-5	top-1	top-2	top-3	top-4	top-5
ours	**7.56**	**10.08**	**11.76**	**11.76**	**13.45**	**6.87**	**9.16**	**10.69**	**10.69**	**12.21**
LLM	**7.56**	7.56	10.08	**11.76**	12.61	1.43	1.65	2.09	2.31	2.42

method. Specifically, the predictions from the top-1 to top-5 with higher probabilities were selected for our method, and we set the temperature to 0.5 for the LLM and performed five predictions.

5.5 Experimental Results

Table 3 shows the experimental results. As can be seen, the recall values obtained by our method were at least as high as that of the LLM across all top-k levels, which indicates that our method can predict complementary responses with higher coverage than the LLM.

The gap was even greater for precision than for recall; our method achieved considerably higher scores. In addition, we found that increasing k did not narrow the gap between the two methods, and our method consistently retained its advantage across all top-k levels. Since an LLM is based on the iterative next word prediction, it generates not only words that have a dependency relation with any of the inputted bunsetsus but also other many words that do not have a dependency relation with them. Thus, simply using the LLM lowered its precision. In contrast, our method focused on only bunsetsus that have a dependency relation with any of the inputted bunsetsus, effectively discarding irrelevant candidates and identifying the head word of the complementary response more accurately. Thus, compared to simply using an LLM, our method can provide informative cues with little noise for the complementary response generation.

Overall, our method, which directly predicts the head words of non-inputted modified bunsetsus, achieved higher agreement with the head words of the complementary responses than the LLM based on iterative next word prediction. These results highlight the strength of our method in providing relevant, low-noise cues that are useful for complementary response generation.

6 Conclusion

This paper has proposed a method that parses dependency structures incrementally to identify dependency relations between inputted and non-inputted bunsetsus and simultaneously predicts the non-inputted bunsetsus involved in those relations. Our method was evaluated in experiments to predict non-inputted bunsetsus that had a dependency relation with any of the inputted bunsetsus, and compared the results with human performance. The results confirmed both the feasibility of our method and remaining challenges. In addition, applying our method to predicting the head words of complementary responses yielded higher performance than a LLM based on iterative next word prediction. These results further verify the usefulness of our method for the complementary response generation.

Limitations

First, the current model can sometimes produce word predictions that are inconsistent with its dependency parsing outputs because the two tasks are only integrated via a weighted loss, and we impose no constraints to enforce consistency between the word prediction and incremental dependency parsing. Second, the evaluation performed in the current study was limited to Japanese lecture-style narratives (from the SIDB). In other words, we did not evaluate generalization to conversational narratives, other domains, or other languages. Finally, the reported results rely on the top-1 to top-5 candidates. We did not examine selection strategies, e.g., re-ranking, for our method, thereby leaving a gap in terms of practical deployment.

Acknowledgments

This work was supported by JSPS KAKENHI Grand Number JP24K15076.

References

Ashkan Alinejad, Maryam Siahbani, and Anoop Sarkar. 2018. Prediction Improves Simultaneous Neural Machine Translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3022–3027, Brussels, Belgium.

Eiji Aramaki. 2016. Japanese Elder's Language Index Corpus v2. https://figshare.com/articles/dataset/Japanese_Elder_s_Language_Index_Corpus_v2/2082706.

Yuya Chiba and Ryuichiro Higashinaka. 2025. Investigating the Impact of Incremental Processing and Voice Activity Projection on Spoken Dialogue Systems. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 3687–3696, Abu Dhabi, UAE.

Erik Ekstedt and Gabriel Skantze. 2021. Projection of Turn Completion in Incremental Spoken Dialogue Systems. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 431–437, Singapore and Online.

Alvin Grissom II, Naho Orita, and Jordan Boyd-Graber. 2016. Incremental Prediction of Sentence-final Verbs: Humans versus Machines. In *Proceedings of the 20th Conference on Computational Natural Language Learning*, pages 95–104, Berlin, Germany.

Jiatao Gu, Graham Neubig, Kyunghyun Cho, and Victor O.K. Li. 2017. Learning to Translate in Real-time with Neural Machine Translation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1053–1062, Valencia, Spain.

Koichiro Ito, Masaki Murata, Tomohiro Ohno, and Shigeki Matsubara. 2022. Construction of Responsive Utterance Corpus for Attentive Listening Response Production. In *Proceedings of the 13th Language Resources and Evaluation Conference*, pages 7244–7252, Marseille, France.

Taku Kudo and Yuji Matsumoto. 2003. Fast Methods for Kernel-Based Text Analysis. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 24–31, Sapporo, Japan.

Wenyan Li, Alvin Grissom II, and Jordan Boyd-Graber. 2020. An Attentive Recurrent Model for Incremental Prediction of Sentence-final Verbs. In *Findings of the Association for Computational Linguistics: Empirical Methods in Natural Language Processing 2020*, pages 126–136, Online.

Chang Liu, Xu Tan, Chongyang Tao, Zhenxin Fu, Dongyan Zhao, Tie-Yan Liu, and Rui Yan. 2022. ProphetChat: Enhancing Dialogue Generation with Simulation of Future Conversation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 962–973, Dublin, Ireland.

Shigeki Matsubara, Keiichi Iwashima, Nobuo Kawaguchi, Katsuhiko Toyama, and Yasuyoshi Inagaki. 2000. Simultaneous Japanese-English Interpretation based on Early Prediction of English Verbs. In *Proceedings of Symposium on Natural Language Processing*, volume 4, pages 268–273.

Shigeki Matsubara, Akira Takagi, Nobuo Kawaguchi, and Yasuyoshi Inagaki. 2002. Bilingual Spoken Monologue Corpus for Simultaneous Machine Interpretation Research. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation*, pages 153–159, Las Palmas, Canary Islands - Spain.

Mikio Nakano, Noboru Miyazaki, Norihito Yasuda, Akira Sugiyama, Jun-ichi Hirasawa, Kohji Dohsaka, and Kiyoaki Aikawa. 2000. WIT: A Toolkit for Building Robust and Real-time Spoken Dialogue Systems. In *Proceedings of the 1st SIGdial Workshop on Discourse and Dialogue - Volume 10*, page 150–159, USA.

Yusuke Oda, Graham Neubig, Sakriani Sakti, Tomoki Toda, and Satoshi Nakamura. 2015. Syntax-based Simultaneous Translation through Prediction of Unseen Syntactic Constituents. 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)*, pages 198–207, Beijing, China.

Siqi Ouyang, Oleksii Hrinchuk, Zhehuai Chen, Vitaly Lavrukhin, Jagadeesh Balam, Lei Li, and Boris Ginsburg. 2025. Anticipating Future with Large Language Model for Simultaneous Machine Translation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5547–5557, Albuquerque, New Mexico.

Junlang Qian, Zixiao Zhu, Hanzhang Zhou, Zijian Feng, Zepeng Zhai, and Kezhi Mao. 2025. Beyond the Next Token: Towards Prompt-Robust Zero-Shot Classification via Efficient Multi-Token Prediction. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7093–7115, Albuquerque, New Mexico.

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 International Conference on Computational Linguistics/Annual Meeting of the Association for Computational Linguistics 2006 Main Conference Poster Sessions*, pages 683–690, Sydney, Australia.

Tomohide Shibata, Daisuke Kawahara, and Sadao Kurohashi. 2019. BERT ni yoru nihongo koubun kaiseki no seido koujou (In Japanese). In *Proceedings of the 25th Annual Meeting of the Association for Natural Language Processing*, pages 205–208. Improved accuracy of Japanese parsing with BERT [Translated from Japanese.].

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*, pages 196–203, Bergen, Norway.

Hiroki Unno, Tomohiro Ohno, Koichiro Ito, and Shigeki Matsubara. 2024. Human Performance in Incremental Dependency Parsing: Dependency Structure Annotations and their Analyses. In *Proceedings of the 38th Pacific Asia Conference on Language, Information and Computation*, pages 697–706, Tokyo, Japan.

Minghan Wang, Thuy-Trang Vu, Jinming Zhao, Fatemeh Shiri, Ehsan Shareghi, and Gholamreza Haffari. 2024. Simultaneous Machine Translation with Large Language Models. In *Proceedings of the 22nd Annual Workshop of the Australasian Language Technology Association*, pages 89–103, Canberra, Australia.

Airi Yoshida and Daisuke Kawahara. 2022. Kouzouteki aimaisei ni motozuku yomizurasa no kenshutsu (In Japanese). In *Proceedings of 28th Annual Meeting of the Association for Natural Language Processing*, pages 425–429. Detecting readability based on structural ambiguity [Translated from Japanese.].

Xingxing Zhang, Jianpeng Cheng, and Mirella Lapata. 2017. Dependency Parsing as Head Selection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 665–676, Valencia, Spain.

Yizhe Zhang, Jiatao Gu, Zhuofeng Wu, Shuangfei Zhai, Joshua Susskind, and Navdeep Jaitly. 2023. PLANNER: Generating Diversified Paragraph via Latent Language Diffusion Model. In *Advances in Neural Information Processing Systems*, volume 36, pages 80178–80190.

A Analysis of Complementary Responses

To evaluate whether prediction of non-inputted modified bunsetsus can function effectively in complementary response generation, we analyzed using the Responsive Utterance Corpus (Ito et al., 2022). The corpus contains 614 complementary responses in total. We randomly sampled 300 complementary responses from the entire set for analysis.

The analysis showed that out of the 300 sample complementary responses, there were 95 responses whose contents appeared in the narrative utterances after the responses were produced. Among these 95 responses, 25 responses (26.32% = 25/95) contained content that appeared as the immediate next words in the subsequent narrative. Furthermore, among the remaining 70 responses, those where the content appeared after at least one intervening word 56 responses (80.00% = 56/70) contained content that appeared as the modified bunsetsu of one of the inputted bunsetsus. These findings confirm that predicting the non-inputted modified bunsetsus is more valuable than simply predicting the immediate next word when generating complementary responses.