INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Estimation of Listening Response Timing by Generative Model and Parameter Control of Response Substantialness Using Dynamic-Prompt-Tune

Toshiki Muromachi, Yoshinobu Kano

Shizuoka University, Japan

tmuromachi@kanolab.net, kano@inf.shizuoka.ac.jp

Abstract

A spoken dialogue system is required to continuously listen to a human user for smooth conversation. We propose a method that simultaneously performs response generation and listener response timing estimation. Our proposed method estimates listener response timing by adding pseudo-samples where listener response should be irrelevant, which allows using text-only conversation dataset without audio information. Furthermore, our proposed method can control substantialness of responses by user-specified parameter integrated with the Dynamic-Prompt-Tune method, which uses prompt token embedding dynamically generated from the parameter. Our automatic and manual evaluation showed that the proposed method can generate responses with more natural timing and more in line with the response substantialness parameter compared to the baseline model.
Index Terms: spoken dialogue system, listening response, response timing estimation, response substantialness control

1. Introduction

In human conversation, listeners continuously engage in listening responses. Such responses can make communication smoother, and therefore, a spoken dialogue system also need to continuously provide listening responses to users. To achieve this, the system needs to estimate the timing and content of responses. However, these are influenced by non-linguistic information, thus, we need a mechanism that can change the frequency and content of responses by receiving non-linguistic information. We aim to construct a spoken dialogue system that estimates when a listener should respond and generates a variety of responses along with a user specified response substantialness parameter.

Several methods have been proposed for estimating the timing of possible listener response, including methods based on acoustic features [1], and methods based on the parts of speech at the end of utterances and the duration of the utterance [2]. Similarly, estimation of turn-taking timing is a related research topic to the estimation of listener response timing. Many methods have been proposed for estimating the timing of turn-taking using acoustic and linguistic features [3, 4]. Ekstedt et al. proposed TurnGPT, a GPT-2 [5] based speaker turn-taking prediction model, which takes the transcribed text of the preceding utterance as input and estimates the speaker turn-taking probability immediately following the input [6].

Regarding natural language generation, large-scale models, such as GPT-3 [7], use Prompt Design, which solves a task without re-training by providing an additional string called a Prompt that describes the task description and other information. However, the performance of Prompt Design tends to be inferior to that of Fine-Tune. Prompt-Tune [8] concatenate a trainable vector, called Soft-Prompt, with an embedded representation of the input text, optimizing only the Soft-Prompt vector. Prompt-Tune reduces the training cost by decreasing training parameters by this Soft-Prompt architecture, which can prevent overfittings compared to Fine-Tune.

Because Soft-Prompt is a static embedded representation specialized for a single task, it is difficult to represent everything in a single static prompt token embedding when complex and varied input statements must be supported, such as dialogue tasks. Several methods have been proposed to dynamically change the prompt token embedding for each input, such as a method to handle multi-modal information by inputting images to the prompt token (called Multi-modal Prompt-Tune in this paper) [9] and Control-Prefixes [10] that control generated text in a specific direction using a prompt token embedding dynamically generated from attribute information and other data.

We propose a method that simultaneously performs response generation and listener response timing estimation, which is different from TurnGPT in that it estimates only the speaker turnover probability. While TurnGPT learns speaker turn-taking probabilities from actual corpus data, our proposed method estimates listener response timing by adding pseudo-samples where listener response is impossible.

Regarding the substantialness-specified response generation, we propose to use three levels of parameter to control the response substantialness as auxiliary information for response generation. In this paper, we mean *substantialness* as how much an expression includes concrete contents, e.g. very short answers like ”yeah” are less substantial but phrasal answers could be more substantial. In order to learn the response substantialness with the parameter, Prompt-Tune is performed using a prompt token embedding dynamically generated from the response substantialness parameter. This enables efficient learning even with small amounts of data.

Because the production of high-quality transcribed text is very costly, spoken conversation data is less available and its amount is insufficient than text data. To augment such an insufficient dataset, we generate negative response timings by randomly cutting the speaker's utterance in the middle, adding pseudo-samples that are impossible for the listener to respond. This augmentation comes from our observation that it is normally inappropriate to make a response in the middle of an utterance, which could interrupt the speaker. This makes large amounts of data in various formats trainable, including textual conversation data without audio information, as well as transcription of spoken conversation.

We performed both automatic and manual evaluations, in terms of response timing and the degree to which the generated responses reflect the substantialness parameter. These evaluation results showed that our proposed method was able to es-

10.21437/Interspeech.2023-320

timate response timing with better performance than baseline models, which estimate response timing by clause boundaries. The evaluation results also showed that our proposed method using Dynamic-Prompt-Tune was able to generate responses more in line with the given substantialness parameter than models with Fine-Tune and Prompt-Tune.

In summary, the contributions of this paper are as follows:

- Integrated response timing estimation with response generation by GPT.
- Estimation of response timing through data augmentation using pseudo-samples randomly cut in the middle of utterances.
- Dynamic-Prompt-Tune was used to efficiently learn the response substantialness parameter from a small number of data.

2. Proposed Method

An overview of the proposed method is shown in Figure 1. The proposed model consists of a GPT and a three-layer MLP. The MLP is Prompt-Token-Encoder for generating prompt token embedding for response substantialness control. As a dialogue system, the system takes response substantialness parameter with three levels of values and the speaker's utterances (Figure 2), and generates responses in a substantialness consistent with the parameter. The response substantialness parameter is assumed to be determined by rule-based methods using features such as pause length.

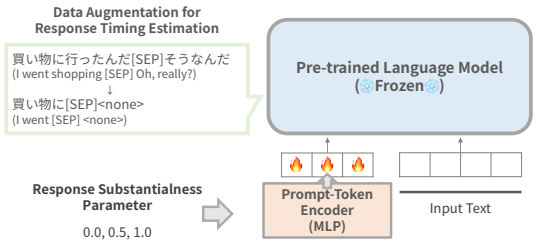


Figure 1: *Conceptual figure of the proposed method*

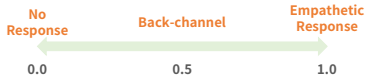


Figure 2: *Response Substantialness Parameter*

2.1. Response Timing Estimation

To augment the insufficient amount of the conversation datasets, we generate negative response timings by randomly cutting the speaker's utterance in the middle, replacing the listener response with a special token (<none>). This augmentation comes from our observation that it is normally inappropriate to make a response in the middle of an utterance, which could interrupt the speaker. Then the GPT model is fine-tuned by these utterance-response pairs to estimate the listener response timing.

2.2. Response Substantialness Control

We generate responses using prompt token embedding that are dynamically generated by our user-specified parameter for response substantialness control.

The proposed method uses only dynamic prompt token embedding excluding the static part of Control-Prefixes. That is, the embedded representation generated by the Prompt-Token-Encoder is used as the prompt token embedding. The model takes the parameter for response substantialness control and utterance-response pairs, which are pairs of speaker utterances and listener responses. During training, only the training parameters of the Prompt-Token-Encoder are updated.

Table 1: *Evaluation results of response timing estimation (Automatic Evaluation)*

	Precision	Recall	F1-Score
Clause Boundaries Model	0.554	**0.893**	0.684
Dynamic-Prompt-Tune	**0.918**	0.869	**0.893**

Table 2: *Evaluation results of response timing estimation (Human Evaluation)*

	Precision	Recall	F1-Score
Clause Boundaries Model	**0.721**	0.708	0.714
Dynamic-Prompt-Tune	0.717	**0.882**	**0.791**

3. Experiment

3.1. Dataset

3.1.1. Response Timing Estimation and Response Generation

For Fine-Tune, we used the Corpus of Everyday Japanese Conversation (CEJC, 168,350 pairs) [11], the Nagoya University Conversation Corpus (33,361 pairs) [12], JEmpatheticDialogues (40,000 pairs) [13], JPersonaChat (61,870 pairs) [13], and independently collected Japanese tweets (5,000,000 pairs). Regarding the spoken conversation corpus, we only used the transcribed text without audio information, which was formatted into utterance-response pairs using the results of the following automatic back-channel determination.

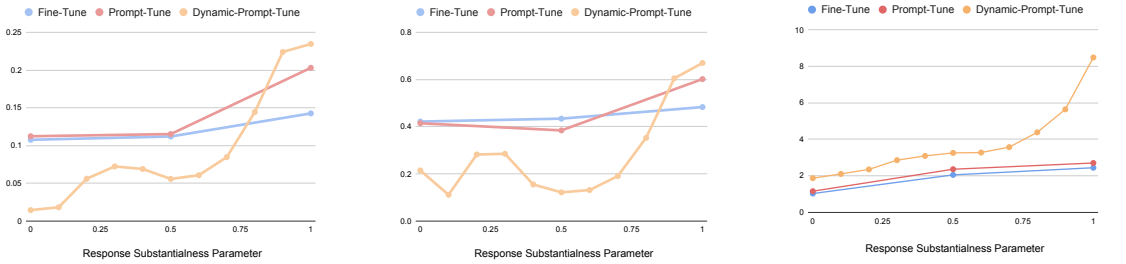
We manually labeled if an utterance was back-channel or not, for 10,000 samples in CEJC evaluation dataset. Then, we fine-tuned BERT [14] to determine whether the input samples were back-channel or not. We used the Japanese Spoken Language Partial Learning BERT[1] as a pretrained BERT model, which was trained by Wikipedia and CSJ [15]. We divided the manually labeled 10,000 samples into 8(training):1(validation):1(evaluation). The accuracy of this evaluation was 0.958. The original corpus texts were divided by these automatically determined back-channels, which became the units of the utterance-response pairs.

As described earlier, we randomly cut off 10% of the speaker's utterances in the middle and replaced them with an additional special token (<none>), using these special tokens as negative samples. We divided the CEJC corpus into training 8: verification 1: evaluation 1, while all other corpora are used as training dataset with the training data taken from CEJC.

3.1.2. Response Substantialness Control

Using 1,200 randomly selected speaker utterances from JEmpatheticDialogues, we manually created gold standard responses

[1]https://www.anlp.jp/proceedings/annual_meeting/2021/pdf_dir/P4-17.pdf



(a) *Distinct-1* (b) *Distinct-2* (c) *Average number of words per response*

Figure 3: *Parameter effectiveness for response substantialness control (Automatic Evaluation)*

Table 3: *Parameter effectiveness for response substantialness control (Human Evaluation)*
F, R, E represent Fluency, Relevance and Effectiveness, respectively.

	0.0			0.5			1.0		
	F	R	E	F	R	E	F	R	E
Fine-Tune	4.78	**3.92**	4.60	4.65	3.96	4.16	4.62	3.79	2.65
Prompt-Tune	4.97	3.67	**4.76**	4.82	4.26	4.12	4.76	4.06	2.83
Dynamic-Prompt-Tune	**4.98**	3.87	4.68	**4.94**	**4.41**	**4.63**	**4.86**	**4.20**	**3.52**

for each of three level response substantialness control parameters: 0.0, 0.5, or 1.0; 0.0: no response, 0.5: only back-channel, 1.0: an empathetic response. The empathetic responses were made based on the original responses of the JEmpatheticDialogues. The training and evaluation data consisted of 600 speech response pairs, 200 pairs for each of the three level parameter. The data preprocessing for response timing estimation described in the previous chapter was applied to 10% of the total data.

3.2. Model

We used *japanese-gpt-1b* [2], a pre-trained Japanese GPT model, with its tokenizer. We fine-tuned *japanese-gpt-1b* with the training dataset for the *response timing estimation and response generation* above. We used this fine-tuned model throughout the following experiment models.

For response substantialness control, we experimented and compared the Fine-Tune, Prompt-Tune, and Dynamic-Prompt-Tune models with 3 additional types of learning, respectively. These three models are used for both experiments of the *response timing estimation and response generation*, and the *response substantialness control*.

3.2.1. Fine-Tune Model

We further fine-tuned the common fine-tuned GPT model, by adding special tokens for the 3 levels of parameter to the end of the input utterance. The number of epochs was set to 3.

3.2.2. Prompt-Tune Model

The above Fine-Tune model was further Prompt-Tuned, with a prompt token embedding size of 100 and an epoch count of 50.

3.2.3. Dynamic-Prompt-Tune Model

Unlike the above two models, the numerical values of the parameter are input directly into the Prompt-Token-Encoder; the prompt token size is set to 20 and the number of epochs to 50.

[2]https://huggingface.co/rinna/japanese-gpt-1b

3.3. Experimental Settings

prompt token sizes were determined from preliminary experiments. In the preliminary experiments, we manually selected which one produces responses in line with the response substantialness parameter, comparing 5, 10, 20, 50, and 100 prompt token sizes. Within the manually selected settings, we adopt the smallest prompt token size for each model. The number of MLP neurons in the Prompt-Token-Encoder was set to 1 for the input layer, 10,000 for the middle layer, and 40,960 for the output layer. The batch size during training was set to 8, and the learning rate was set to 5e-05 for Fine-Tune and 0.3 for Prompt-Tune and Dynamic-Prompt-Tune. AdamW was used as the optimizer, and the number of epochs was set to the value at which loss converged during training.

4. Experiment Evaluation and Results

4.1. Response Timing Estimation

4.1.1. Automatic Evaluation

We randomly cut off the speaker's utterances in the middle of a sample, for half of the evaluation dataset, regarding these as negative samples where we should not make responses. The rest of the dataset includes the original utterances as they are, so we regard these original ones as positive samples, i.e. timings which can make responses. As a result, we obtained a balanced evaluation dataset.

In addition to the proposed Dynamic-Prompt-Tune model, we evaluated a rule-based baseline model which uses clause boundaries determined by the surface forms of parts of speech and words obtained by Japanese morphological analyzer Juman++ [16, 17](Ver.2.0.0-rc3).

Table 1 shows the automatic evaluation results, where results of Dynamic-Prompt-Tune model are the response substantialness parameter pre-fixed at 0.5. The proposed Dynamic-Prompt-Tune model achieved an F1-Score approximately 20% higher than that of the baseline model.

Table 4: *Example of response generation*

Model	Input	Parameter	Response Generation Results
Fine-Tune	今日は友達と遊びに行ったんだ (I went out with friends today)	0.0	そうなんだ楽しかった? (I see... that was fun, wasn't it?)
		0.5	楽しかった? (Did you have a good time?)
		1.0	楽しかった? (Did you have a good time?)
Prompt-Tune	今日は友達と遊びに行ったんだ (I went out with friends today)	0.0	うん (yeah)
		0.5	よかったね (Good for you.)
		1.0	いいなー (I like it.)
Dynamic-Prompt-Tune	今日は友達と遊びに行ったんだ (I went out with friends today)	0.0	<none>
		0.5	そっかぁ (I see.)
		1.0	いいな青春してるね (I like it. You're a young man, aren't you?)

4.1.2. Human Evaluation

For the human evaluation dataset, we extracted transcribed text from the Corpus of Spoken Japanese (CSJ), a total of 30 minutes which consists of approximately 10 minutes of recordings from three different speakers. Three annotators manually labeled binary response timings. We measured the agreement between each annotator and the system's estimation by Precision, Recall, and F1-Score as human evaluation metrics. Table 2 shows the average values of the three annotators. The inter-annotator agreement was 0.465 in Fleiss' Kappa [18].

Similar to the automatic evaluation, the Dynamic-Prompt-Tune model was able to estimate with higher evaluation scores.

4.2. Effectiveness of Response Substantialness Parameter

4.2.1. Automatic Evaluation

It can be said that our gold standard dataset generates more diverse and longer responses as the response substantialness parameter increases. Therefore, we used Distinct-1,2 [19] and length of generated text to evaluate the generated responses whether the parameter changes were effective (Figure 3). For evaluation purposes, we randomly extracted 1,000 samples from the CEJC evaluation dataset that contained empathetic responses, and used only the speaker's utterance portion. Furthermore, the Dynamic-Prompt-Tune model conducted response generation with parameters spaced at 0.1 intervals.

Compared to the Fine-Tune and Prompt-Tune models, the Dynamic-Prompt-Tune model has been suggested to perform response generation more in line with the response substantialness parameter, as significant variability has been observed in both the Distinct score and the length of the generated text.

4.2.2. Human Evaluation

For the human evaluation of the parameter effectiveness of response substantialness control, we randomly extracted 100 samples from the CEJC automatic evaluation dataset. Three annotators manually evaluated responses generated for each of the 3 parameter (0.0, 0.5, and 1.0) by a 5-point scale in Fluency, Relevance, and parameter Effectiveness, respectively (Table 3).

Dynamic-Prompt-Tune model showed better evaluation scores than baselines, particularly for larger parameter values.

5. Discussion

5.1. Example of Response Generation

Table 4 shows examples of response generation for the Fine-Tune model, Prompt-Tune model, and Dynamic-Prompt-Tune model, for each response substantialness parameter. There appears to be little change in the response generation results for the Fine-Tune model and the Prompt-Tune model when the parameter is changed. However, Dynamic-Prompt-Tune model generated a special token indicating no response(<none>) when the parameter is 0.0, and generated back-channel when the parameter is 0.5. Furthermore, when the parameter is 1.0, the Dynamic-Prompt-Tune model was able to generate empathic responses. These results suggest that the Dynamic-Prompt-Tune model was able to generate responses along with the specified parameter.

5.2. Limitations of the Proposed Method

If the input text is grammatically broken, words are omitted, or fillers are inserted, the proposed model often generates irrelevant responses. This is likely to occur because the input does not include a long dialogue history and lacks a mechanism to appropriately summarize the utterance.

6. Conclusions

Dialogue system is required to continuously perform listening responses for smooth communication in spoken dialogue with humans. We proposed a method for estimating timings when the listener should respond, and for generating the responses. In our proposed method, response timing is estimated by inputting response substantialness parameter and utterances into a response generation model, and response substantialness is controlled by a prompt token embedding dynamically generated from the parameter. To augment insufficient conversational dataset, we generate negative response timings by randomly cutting the speaker's utterance in the middle. These methods enable the use of various forms of dialogue data for response timing estimation, either text chat or spoken dialogue, while response substantialness control can be learned efficiently from a small amount of data. From the results of automatic and human evaluation, we confirmed that the proposed method can generate responses with more natural timing and in line with the given parameter compared to the baseline model.

Generating responses that follow non-linguistic information, such as the flow and rhythm of the conversation, is a future work.

7. Acknowledgements

This work was supported by JSPS Kakenhi JP21K18115, JST AIP Accelerated Proposal JPMJCR22U4, and Secom Science and Technology Foundation Grant-in-Aid for Research in Specific Areas. We express our gratitude to those who kindly participated in the experiments.

8. References

[1] D. Lala, P. Milhorat, K. Inoue, M. Ishida, K. Takanashi, and T. Kawahara, "Attentive listening system with backchanneling, response generation and flexible turn-taking," in *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, 2017, pp. 127–136.

[2] M. Takeuchi, N. Kitaoka, and S. Nakagawa, "Generation of natural response timing using decision tree based on prosodic and linguistic information," in *Eighth European Conference on Speech Communication and Technology*, 2003.

[3] K. Hara, K. Inoue, K. Takanashi, and T. Kawahara, "Prediction of Turn-taking Using Multitask Learning with Prediction of Backchannels and Fillers," in *Proceedings of Interspeech*, 2018, pp. 991–995.

[4] S.-Y. Chang, B. Li, T. Sainath, C. Zhang, T. Strohman, Q. Liang, and Y. He, "Turn-Taking Prediction for Natural Conversational Speech," in *Proceedings of Interspeech*, 2022, pp. 1821–1825.

[5] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI blog*, 2019.

[6] E. Ekstedt and G. Skantze, "TurnGPT: a transformer-based language model for predicting turn-taking in spoken dialog," in *Findings of the Association for Computational Linguistics: EMNLP*, 2020, pp. 2981–2990.

[7] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 1877–1901.

[8] B. Lester, R. Al-Rfou, and N. Constant, "The power of scale for parameter-efficient prompt tuning," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 3045–3059.

[9] M. Tsimpoukelli, J. L. Menick, S. Cabi, S. Eslami, O. Vinyals, and F. Hill, "Multimodal few-shot learning with frozen language models," *Advances in Neural Information Processing Systems*, vol. 34, pp. 200–212, 2021.

[10] J. Clive, K. Cao, and M. Rei, "Control prefixes for parameter-efficient text generation," in *Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*. Association for Computational Linguistics, 2022, pp. 363–382.

[11] H. Koiso, H. Amatani, Y. Den, Y. Iseki, Y. Ishimoto, W. Kashino, Y. Kawabata, K. Nishikawa, Y. Tanaka, Y. Usuda, and Y. Watanabe, "Design and evaluation of the corpus of everyday Japanese conversation," in *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, 2022, pp. 5587–5594.

[12] I. Fujimura, S. Chiba, and M. Ohso, "Lexical and grammatical features of spoken and written japanese in contrast: Exploring a lexical profiling approach to comparing spoken and written corpora," in *Proceedings of the VIIth GSCP International Conference. Speech and Corpora*, 2012, pp. 393–398.

[13] H. Sugiyama, M. Mizukami, T. Arimoto, H. Narimatsu, Y. Chiba, H. Nakajima, and T. Meguro, "Empirical analysis of training strategies of transformer-based japanese chit-chat systems," *arXiv preprint arXiv:2109.05217*, 2021.

[14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, 2019, pp. 4171–4186.

[15] K. Maekawa, H. Koiso, S. Furui, and H. Isahara, "Spontaneous speech corpus of Japanese," in *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC)*, 2000.

[16] H. Morita, D. Kawahara, and S. Kurohashi, "Morphological analysis for unsegmented languages using recurrent neural network language model," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 2292–2297.

[17] A. Tolmachev, D. Kawahara, and S. Kurohashi, "Juman++: A morphological analysis toolkit for scriptio continua," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2018, pp. 54–59.

[18] J. L. Fleiss, "Measuring nominal scale agreement among many raters." *Psychological bulletin*, vol. 76, no. 5, p. 378, 1971.

[19] J. Li, M. Galley, C. Brockett, J. Gao, and B. Dolan, "A diversity-promoting objective function for neural conversation models," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016, pp. 110–119.