Generating Repetitions with Appropriate Repeated Words

Toshiki Kawamoto, Hidetaka Kamigaito, Kotaro Funakoshi and Manabu Okumura
Tokyo Institute of technology
{kawamoto,kamigaito,funakoshi,oku}@lr.pi.titech.ac.jp

Abstract

A repetition is a response that repeats words in the previous speaker's utterance in a dialogue. Repetitions are essential in communication to build trust with others, as investigated in linguistic studies. In this work, we focus on repetition generation. To the best of our knowledge, this is the first neural approach to address repetition generation. We propose Weighted Label Smoothing, a smoothing method for explicitly learning which words to repeat during fine-tuning, and a repetition scoring method that can output more appropriate repetitions during decoding. We conducted automatic and human evaluations involving applying these methods to the pre-trained language model T5 for generating repetitions. The experimental results indicate that our methods outperformed baselines in both evaluations.

1 Introduction

Dialogues can build a trusting relationship with others, thus are essential in our daily lives (Schein, 1993; Searle and Vanderveken, 1985). There are several types of responses in dialogues, and the one we focus on is repetitions (Tannen, 1987). A repetition is a response that uses the previous speaker's words or phrases. Figure 1 shows an example. The phrases "a bear" and "came out" are repeated. Repetitions frequently appear in a conversation with diverse roles, e.g., to indicate attentive listening, confirm the previous utterance, and show agreement or sympathy (Machi, 2019; Shimojima et al., 2002). Many linguistic studies investigating repetitions have concluded that they are important for building and strengthening relationships between speakers (Tannen et al., 1989; Johnstone, 2002; Norrick, 1987; Brown, 1999). From the above linguistic point of view, we can say that repetitions are indispensable in dialogues.

Repetitions are similar to paraphrases and reflections, which are component skills of counseling (Theron, 2008), in terms of using the previous speaker's utterance. Paraphrases and reflections have been generated using a template-based method (Han et al., 2013).

Speaker: When I was driving, a bear suddenly came out .
Listener: Oh. A bear came out !?

Figure 1: Example repetition. Listener's response uses words from previous speaker's utterance. Yellow words indicate those that are repeated and green words indicate those in the repetition.

While many studies have tackled general response generation with neural network-based frameworks (Adiwardana et al., 2020; Zhang et al., 2020), less attention has been paid to repetitions. This might be because they are buried in a huge amount of response data. Therefore, we focus on automatically generating repetitions. To the best of our knowledge, this is the first study on generating repetitions with a neural approach. We used the pre-trained language model T5 (Raffel et al., 2019) for generating repetitions because it has performed well in language generation in past few years (e.g., Radford et al.; Raffel et al., 2019; Lewis et al., 2020).

In generating repetitions, it is important to take into account which words should be repeated from the previous utterance. The repeated words might represent objective facts, names of people and places, and the speaker's experiences and emotions, though they are different depending on the language (Machi, 2008). When we use a pre-trained language model, however, the model cannot explicitly learn the repeat likelihood among words during fine-tuning because it is difficult to directly teach which words are likely to be repeated at this step.

To solve this problem, we propose Weighted Label Smoothing (WLS), which is an improvement upon Label Smoothing (LS) (Szegedy et al., 2016). The method enables a language model-based re-

Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 852 - 859

sponse generator to learn the words it should use for each input utterance during fine-tuning. We also propose the repetition scoring method (RSM) to expand the scoring method proposed in Wu et al. (2016) for selecting repetitions that contain appropriate repeated words during decoding.

We evaluated the proposed methods on a dataset we created in Japanese for automatic and human evaluations. Our methods outperformed baselines, i.e., fine-tuned pre-trained language models without our methods, in both evaluations. This indicates that our methods can generate repetitions that contain appropriate words to repeat.

Our contributions are as follows:

1. To the best of our knowledge, this is the first study to use a neural model for generating repetitions.
2. We will release our code and the dataset of repetitions we created.[1]
3. We propose WLS, that takes into account words that should be repeated during fine-tuning, for generating repetitions.
4. We propose RSM to select repetitions containing appropriate repeated words during decoding.

2 Proposed Methods

Repetitions do not necessarily mean we repeat any word. For the utterance "Today's dinner was pizza.", the repetition "Oh, you ate pizza." is more appropriate than "Oh, you ate today." However, a fine-tuned pre-trained language model alone may not be enough to generate repetitions with appropriate repeated words. Therefore, to generate a response that repeats more appropriate words, we introduce *repeat scores* (§2.1) to calculate how likely a word is repeated and incorporate the scores into WLS (§2.2) for fine-tuning and RSM (§2.3) for beam search in decoding.

2.1 Repeat Score

We should give high scores to words that tend to be used in repetitions and low scores to words that should not be. Since only content words (nouns, verbs, adjectives, or adverbs) are repeated in Japanese, we define a *repeat score* only for them. Since subwords are used as a unit in a pre-trained language model, all the subwords in the same content word receive the same *repeat score*.

We use BERT (Devlin et al., 2019) to construct a model for scoring the *repeat scores* in the range of [0, 1]. We pass the final hidden state of BERT through SpanExtractor (Lee et al., 2017) for each word and then convert the vector to a scalar value through a multi-layer perceptron, which has a sigmoid function as the last layer. In the training data, the label is set to 1 if the target content word was repeated, and 0 if it was not. The output is then normalized by applying min-max scaling.

2.2 Weighted Label Smoothing (WLS)

In this section, we explain how to learn words to repeat when fine-tuning a pre-trained language model for repetition generation. Neural response generation models try to optimize cross-entropy loss. Let X be a previous utterance and Y be a response, where Y is divided into subwords as $Y = y_1, \ldots, y_T$. Letting K be the total number of subwords and v_k be the k-th subword, the cross-entropy loss is defined as follows:

$$L(q, p) = -\sum_{k=1}^{K} q(v_k) \log\{p(v_k|y_{<t}, X)\},$$

where $p(v_k|y_{<t}, X)$ is the probability of v_k that the model outputs at time step t given X, and $q(v_k)$ is the probability of v_k in a target distribution that the model aims for. When a one-hot distribution is used, $q(v_k)$ is as follows with a function δ_{v_k,y_t}, which becomes 1 when $v_k = y_t$:

$$q(v_k) = \delta_{v_k,y_t}.$$

When LS is used, however, $q(v_k)$ is as follows with uniform distribution $u(v_k) = 1/K$:

$$q(v_k) = (1 - \epsilon)\delta_{v_k,y_t} + \epsilon u(v_k),$$

where ϵ is a hyperparameter.

A one-hot distribution and LS cannot learn a subword to repeat explicitly because there are labels other than the target, i.e., v_k when $v_k \neq y_t$, that have the same $q(v_k)$. Therefore, we propose WLS, which takes into account how likely a subword is repeated. We use *repeat scores*, explained in §2.1, instead of $u(v_k)$. The $q(v_k)$ of WLS is defined as follows:

$$q(v_k) = (1 - \epsilon)\delta_{v_k,y_t} + \epsilon \frac{r(v_k)^\gamma}{K},$$

[1] https://github.com/titech-nlp/repetition-generation

Dialogue	運動やってました. (I played sports.) へーそしたら中学校高校はクラブは何か？ (Oh, did you participate in any clubs in junior high or high school?) 中学高校大学まで陸上部でした. (I was a member of a track and field club in junior high, high school, and college.)
Repetition1	陸上部ですか. (Track and field club?)
Repetition2	陸上部だったんですね. (You were in the track and field club.)
Repetition3	中学高校大学まで陸上とは, 長く続けられたんですね！ (You were in the track and field club for a long time, from junior high through high school and college.)

Table 1: Examples from repetition dataset. There are at most three repetitions for one dialogue.

where $r(v_k)$ is the *repeat score* for v_k, and γ is a hyperparameter. We use the $q(v_k)$ of WLS as the distribution in the cross-entropy loss function. Subwords in the previous speaker's utterance are weighted in accordance with their $r(v_k)$. Note that if we set $\gamma = 0$, WLS is the same as LS.

2.3 Repetition Scoring Method (RSM)

Pre-trained language models usually use beam search in decoding. We propose a scoring method, RSM, to select more appropriate repetitions in the beam search. RSM is an extension of a scoring method for machine translation in Wu et al. (2016). The original scoring method uses a length normalization procedure and coverage penalty (Tu et al., 2016). Length normalization treats sentences of different lengths equally. The coverage penalty gives a high score to a sentence that is most likely to cover all the words in the source sentence. Since the original scoring method cannot select a repetition with appropriate repeated words, we modify the method by adding *repeat scores*, which indicate words to repeat. Letting Y be a candidate response during beam search and X be the previous utterance, the generation probability is $P(Y|X)$. The scoring function $s(Y, X)$ of RSM is as follows:

$$
\begin{aligned}
s(Y,X) &= log\{P(Y|X)\}/lp(Y)+cp(X,Y)+rs(X,Y), \\
lp(Y) &= \frac{(5+|Y|)^{\alpha}}{(5+1)^{\alpha}}, \\
cp(Y,X) &= \beta * \sum_{i=1}^{|X|} log(\sum_{j=1}^{|Y|} p_{i,j}), \\
rs(Y,X) &= log \sum_{j=1}^{|Y|} r(v_j),
\end{aligned}
$$

where α and β are hyperparameters for length normalization and coverage penalty, respectively. We carry out two modifications to the original scoring method to yield RSM. First, we use the attention value of $p_{i,j}$ without suppression. In contrast to machine translation, in which an input and output have a one-to-one relationship, lengths of an input and output are not the same in repetition generation, and so it is not suitable to suppress the attention value under 1.0. Second, we add the term $rs(Y, X)$, which represents the sum of *repeat scores* for subwords in the response.

3 Dataset

We manually created pairs of a speaker's utterance and its repetition as our dataset using a crowdsourcing service.[2] Since repetitions often occur when a listener replies to a speaker, we used utterances in a corpus of listening dialogues (Yoshino et al., 2018) between an elderly person and caregiver or clinical psychologist as the speaker's utterances in our dataset.[3] In this corpus, the elderly person tends to be a speaker and the others are listeners. We extracted the elderly person's utterances containing content words for creating a repetition. The number of extracted utterances was 5,548. We asked three crowdsourcing workers to create repetitions for each utterance. Specifically, a worker was shown two utterances before each target utterance and asked to create a repetition, that supports the creation of context-aware repetitions. When the workers found it difficult to create a repetition for an utterance, they could discard it. The total number of workers was 333.

Examples from the dataset are given in Table 1. The size and statistics of our repetition dataset are shown in Tables 2 and 3. The word overlap rate is the percentage of words in an utterance that are

[2] https://www.lancers.jp/

[3] We attempted to extract repetitions from the corpus using a rule-based approach and found it is difficult to obtain a sufficient amount of such repetitions.

	Train.	Valid.	Test
Utterance (Dialogue)	4106	489	490
Repetition	10677	1305	1312

Table 2: Size of repetition dataset.

Total Dialogues	5085
Total Repetitions	13294
Average # of Repetitions per Utterance	2.61
Average # of Tokens per Utterance	26.25
Average # of Tokens per Repetition	11.74
Word Overlap Rate	36.48%
Content-word Overlap Rate	38.14%

Table 3: Statistics of repetition dataset.

repeated in a repetition. The content-word overlap rate is the percentage of content words of an utterance that are repeated. Comparing the average numbers of tokens, repetitions are much shorter than utterances. This may indicate that repetitions cannot be produced simply by copying the utterances, and we need to select information that is worth repeating from the utterances.

To understand what types of words overlap, Table 4 shows the percentage of all words' parts-of-speech and overlapped words' parts-of-speech in utterances. Since "postpositional particles" and "auxiliary verbs" tend to accompany content words in a Japanese unit called 'bunsetsu', it might be natural that they also appear in repetitions in high percentages.

While we can have at most three repetitions for an utterance in our dataset, we used only one randomly selected repetition for an utterance in the training data. We used all repetitions for an utterance for the evaluation on the validation and test data to consider the diversity of responses.

4 Experiments

4.1 General Setup

Repeat scores were calculated from the training data. SentencePiece (Kudo and Richardson, 2018) was used to segment the dataset into subwords. With WLS, the hyperparameter ϵ was set to 0.1 following a previous study (Szegedy et al., 2016), and γ was tuned to 4 with the validation data, as explained in Appendix A. With RSM, we used $\alpha = 0.2$ and $\beta = 0.2$, following a previous study (Wu et al., 2016), and a beam size of 5. We used MeCab[4] as a tokenizer to identify content words.

[4] https://taku910.github.io/mecab/

PoS	All(%)	Overlap(%)
Postpositional Particle	27.64	39.02
Noun	23.85	32.70
Auxiliary Verb	9.34	13.09
Verb	13.25	10.03
Adjective	1.86	2.52
Adverb	4.57	1.61
Filler	0.37	0.01

Table 4: The ratios of words and overlapped words of different parts-of-speech (PoS) in utterances.

Utterance	Rule-Based
それとやっぱり深さ, 魚のどの辺におるかとかが難しいんですわ. (It's hard to know where fish are and what depths they are at.)	難しいですか. (Hard, is it?)
先生の話をしっかり聞く言う事が大事. (It is important to listen carefully to what a teacher says.)	先生ですか. (The teacher, is it?)
色々ありましたからね, 国際的なニュースもね. (There's been a lot going on, and international news, too.)	ニュースですか. (News, is it?)

Table 5: Examples of utterance and rule-based response.

4.2 Compared Methods

The baseline methods were as follows:

Rule-Based is a rule-based method, with which a response is created with a content word in the speaker's utterance + *"desuka"* ("is it?"). The content word is randomly selected from the utterance. Examples of rule-based responses are given in Table 5. Responses made with Rule-Based always contain a repeated word and have few grammatical errors. However, *"desuka"* cannot cover all situations. *"desuka"* was chosen because 52% of repetitions in our dataset ends with *"desuka"*, and 6.1% of repetitions are a single word + *"desuka"*.

BertSumAbs (Liu and Lapata, 2019) is a model trained with BERT[5] as the encoder and randomly initialized Transformer as the decoder.

T5[6] (Raffel et al., 2019) is a model that was fine-tuned with the repetition dataset.[7]

LS is T5 fine-tuned with LS.

Copy is T5 fine-tuned with the copy mechanism (See et al., 2017). Since the copy mechanism can be considered similar to the repetition model in that it is used to generate the same words as in an input sentence, we used it for comparison.

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

[6] https://huggingface.co/sonoisa/t5-base-japanese

[7] While another possible model for comparison is GPT-2 (Radford et al.), we did not use it since it was known that T5 is superior to GPT-2 in generation performance (Kale and Rastogi, 2020; Zhao et al., 2020).

	RG-1	RG-2	RG-L	%
Rule-Based	35.26	14.03	35.11	58.24
BertSumAbs	30.73	10.97	29.94	52.51
T5	45.34	22.34	44.59	81.67
LS	45.89	23.08	45.12	81.83
Copy	45.83	23.32	45.07	81.67
WLS	47.88†	24.56	47.14†	85.77†
RSM	46.96	24.66	46.13	84.38†
WLS + RSM	**49.16**†	**26.58**†	**48.28**†	**89.56**†

Table 6: Results of automatic evaluation. % is percentage of outputs containing correct repeated words. Results with † are significantly different from LS, best baseline, using Wilcoxon rank sum test ($p < 0.05$).

	RG-1	RG-2	RG-L	%
w/o lp	46.81	24.52	45.98	83.91
w/o cp	46.93	**24.66**	46.11	84.30
w/o rs	44.97	22.25	44.13	81.28
RSM	**46.96**	**24.66**	**46.13**	**84.38**

Table 7: Ablation study for RSM. lp, cp, and rs were explained in §2.3.

Note that the T5 and BERT were versions pretrained in Japanese. Our methods are as follows:
WLS is T5 fine-tuned with WLS, as mentioned in §2.2.
RSM is T5 using RSM during beam search, as mentioned in §2.3.
WLS + RSM is T5 fine-tuned with WLS and using RSM during beam search.

4.3 Automatic Evaluation

The evaluation metrics were ROUGE (RG-1, RG-2, RG-L) (Lin, 2004) and the percentage of outputs containing correct repeated words. The correct repeated words are content words repeated in the gold response. The experimental results are listed in Table 6. WLS + RSM obtained the highest scores for all metrics, confirming the effectiveness of both WLS and RSM.

We conducted an ablation study to analyze the results of RSM. The results are listed in Table 7. Since w/o rs received the lowest scores, rs was considered the most effective.

Examples of an input and generated responses from the baseline and our model are shown in Table 8. The proposed model (WLS + RSM) successfully generated a response that was close to the correct response, focusing on "having friends who play Go".

4.4 Human Evaluation

We also conducted a human evaluation by comparing three types of response generation methods:

Utterance	昨日は同じ僕らの仲間で囲碁する人いたからね. (Yesterday there were our friends who play Go.)
Gold	仲間で囲碁する人いたんですね. (There were friends who play Go.)
Rule-Based	囲碁ですか. (Go, is it?)
T5	囲碁をしてくれたんですね. (You played Go.)
Ours	仲間内で囲碁する人いたんですか. (There were friends who play Go.)

Table 8: Examples of generated responses from different models.

	Gram	Rel	Cohe	Rep
Rule-Based	2.63	2.49	2.37	2.64
T5	2.82	2.77	2.62	2.79
WLS + RSM	**2.85**†	**2.80**	**2.64**	**2.88**†

Table 9: Results of human evaluation. Results with † are significantly different from T5, the best baseline, using Wilcoxon rank sum test ($p < 0.05$).

Rule-Based, T5, and our model (WLS + RSM). The evaluation measures were grammaticality (Gram), relevance (Rel), coherence (Cohe), and whether repeated words are included (Rep). Two hundred pairs were randomly selected from the test data. The responses were shown to five workers and evaluated on a three-point Likert scale. The response was evaluated with the previous speaker's utterance and one turn before the speaker's utterance as context, meaning the context helps in determining whether the response is an appropriate repetition. The total number of evaluators was 110.

Average scores from the evaluation are listed in Table 9. WLS + RSM outperformed the other methods for all measures, confirming its effectiveness.

5 Conclusion

We focused on repetition generation. Although repetitions play an important role in dialogues, there has been no neural approach for this task to the best of our knowledge. We proposed WLS, which is an improvement upon LS, during fine-tuning and RSM, which is an extended scoring method, during decoding for repetition generation. Through automatic and human evaluations, we confirmed that our model can generate repetitions that contain more appropriate words to repeat than baseline models. For future work, we will take into account synonyms and multiple gold repetition instances to calculate *repeat scores* for improving the diversity of responses. We are also planning to incorporate our repetition model into a general response generation framework.

Acknowledgements

We thank Dr. Koichiro Yoshino of RIKEN for providing the listening dialogue data.

Ethical Statement

Neural generative models have the potential to generate unexpected responses such as violent remarks. As we focused on repetition generation, its model repeats a user's utterance, and so there is little chance of causing unintended responses compared with chit-chat dialogue systems. However, this does not mean that unintended responses will never appear, e.g., when a user's utterance is an unintended expression. Thus, the same consideration must be taken as with other dialogue systems.

Our dataset was created to repeat from utterances in a privacy-secured dataset, and so there is no privacy issue. Since the license of the original dataset is CC BY-NC 4.0, we could use it for this study. We define that the license of our dataset is also CC BY-NC 4.0.

References

Daniel Adiwardana, Minh-Thang Luong, David R. So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, and Quoc V. Le. 2020. Towards a human-like open-domain chatbot.

Penelope Brown. 1999. Repetition. *Journal of Linguistic Anthropology*, 9(1/2):223–226.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Sangdo Han, Kyusong Lee, Donghyeon Lee, and Gary Geunbae Lee. 2013. Counseling dialog system with 5W1H extraction. In *Proceedings of the SIGDIAL 2013 Conference*, pages 349–353, Metz, France. Association for Computational Linguistics.

Barbara Johnstone. 2002. *Discourse analysis*. John Wiley & Sons.

Mihir Kale and Abhinav Rastogi. 2020. Text-to-text pre-training for data-to-text tasks. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 97–102, Dublin, Ireland. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Saeko Machi. 2008. How repetition operates in japanese and english: Introducing different cultural orientations towards conversation. International spring forum.

Saeko Machi. 2019. Managing relationships through repetition: How repetition creates ever-shifting relationships in japanese conversation. *Pragmatics*, 29(1):57–82.

Neal R Norrick. 1987. Functions of repetition in conversation. *Text-Interdisciplinary Journal for the Study of Discourse*, 7(3):245–264.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *ArXiv preprint*, abs/1910.10683.

Edgar H Schein. 1993. On dialogue, culture, and organizational learning. *Organizational dynamics*, 22(2):40–52.

John R Searle and Daniel Vanderveken. 1985. Speech acts and illocutionary logic. In *Logic, thought and action*, pages 109–132. Springer.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Atsushi Shimojima, Yasuhiro Katagiri, Hanae Koiso, and Marc Swerts. 2002. Informational and dialogue-coordinating functions of prosodic features of japanese echoic responses. *Speech communication*, 36(1-2):113–132.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2818–2826. IEEE Computer Society.

Deborah Tannen. 1987. Repetition in conversation: Toward a poetics of talk. *Language*, pages 574–605.

Deborah Tannen et al. 1989. *Talking Voices: Repetition, Dialogue and Imagery in Conversational Discourse*, volume 6. Cambridge University Press.

Michael John Theron. 2008. *A manual for basic relational skills training in psychotherapy*. Ph.D. thesis, Citeseer.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 76–85, Berlin, Germany. Association for Computational Linguistics.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *ArXiv preprint*, abs/1609.08144.

Koichiro Yoshino, Hiroki Tanaka, Kyoshiro Sugiyama, Makoto Kondo, and Satoshi Nakamura. 2018. Japanese dialogue corpus of information navigation and attentive listening annotated with extended ISO-24617-2 dialogue act tags. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020. DIALOGPT : Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278, Online. Association for Computational Linguistics.

Xueliang Zhao, Wei Wu, Can Xu, Chongyang Tao, Dongyan Zhao, and Rui Yan. 2020. Knowledge-grounded dialogue generation with pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3377–3390, Online. Association for Computational Linguistics.

γ	0.0	0.1	0.5	1.0	2.0	3.0	4.0	5.0	10.0
%	82.06	82.06	81.28	83.37	82.21	83.52	85.07	82.90	84.53

Table 10: Percentage of generated responses containing a correct repeated word in the development data when γ was changed. $\epsilon = 0.1$. $\gamma = 0$ indicates LS. The best score was obtained when $\gamma = 4.0$.

A Exploring Hyperparameter γ

We explored the effect of γ on the percentage of responses containing a correct repeated word. The model we used for experiments was the pre-trained model T5, fine-tuned with the training data in §3. We generated repetitions on the development data. The results are listed in Table 10. The best score was recorded when $\gamma = 4.0$. Therefore, we used this value.

B P-values

We now discuss the p-values in the experimental results. To obtain p-values, we conducted the Wilcoxon rank sum test to compare the effectiveness between baseline models and our proposed models. Table 11 shows the p-values for Table 6 from LS. Table 12 shows those for Table 9 from T5.

	RG-1	RG-2	RG-L	%
WLS	0.026	0.159	0.031	0.003
RSM	0.225	0.141	0.294	0.047
WLS + RSM	0.000	0.001	0.001	0.000

Table 11: P-values in Wilcoxon rank sum test between LS and our proposed models in Table 6.

	Gram	Rel	Cohe	Rep
WLS + RSM	0.000	0.055	0.170	0.000

Table 12: P-values in Wilcoxon rank sum test between T5 and WLS + RSM in Table 9.