Consistent Response Generation with Controlled Specificity

Junya Takayama and **Yuki Arase**
Graduate School of Information Science and Technology, Osaka University
{takayama.junya, arase}@ist.osaka-u.ac.jp

Abstract

We propose a method to control the specificity of responses while maintaining the consistency with the utterances for open-domain conversation systems. We first design a metric based on pointwise mutual information, which measures the co-occurrence degree between an utterance and a response. To control the specificity of the generated responses, we add the distant supervision based on the co-occurrence degree and a PMI-based word prediction mechanism to a sequence-to-sequence model. Using these mechanisms, our model outputs the words with desired specificity for a given specificity level. In experiments with open-domain dialogue corpora, automatic and human evaluation results confirm that our model controls the specificity of the responses more sensitively than the conventional model and can generate highly consistent responses.

1 Introduction

Open-domain response generation is a task for generating a human-like responses to chit-chatting. There are many end-to-end response generation models (Vinyals and Le, 2015; Sordoni et al., 2015; Mei et al., 2017) that apply a sequence-to-sequence (Seq2Seq) (Sutskever et al., 2014) architecture, which allows the generation of fluent responses. However, the Seq2Seq model suffers from a tendency to generate safe but overly typical responses (*i.e*. dull responses), such as “Yes” and “I don’t understand.” To solve this problem, several studies proposed methods to increase the specificity of the generated responses (Li et al., 2016a; Zhang et al., 2018b; Jiang et al., 2019); however, simply maximizing the specificity of the response results in a degenerative solution that generates a specific but inconsistent responses.

In this study, we define the conditions that an automatically generated response is expected to satisfy as (i) being consistent with an input utterance, (ii) being specific to provide informative contents, and (iii) being controllable. As shown in Figure 1, in a human conversation, an utterance could have various responses with different specificity (Csáky et al., 2019). Then, humans control the specificity of the response as necessary. Thus, instead of only generating highly specific responses, the specificity should be controllable in response generation tasks.

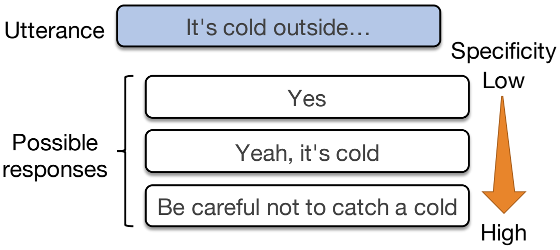

Figure 1: An example of the relationship between utterance and response. There are several possible responses to an utterance with various specificity.

We propose a method to control the specificity of responses while maintaining their consistency with the utterances. Following the observation that a response uniquely co-occurring with a specific utterance in a corpus is both specific and consistent for the utterance, we design a metric called MaxPMI, which measures the co-occurrence degree between an utterance and a response on the basis of positive pointwise mutual information (PPMI). We apply the distant supervision into our model using automatically annotated MaxPMI scores of the training set. At the inference, the specificity of the generated responses can be controlled by inputting a desired specificity level. We also propose a method to automatically set the specificity level by estimating the maximum MaxPMI score for an input utterance, which allows the generation of a response which has the maximum mutual information with the input.

We conducted both automatic and human eval-

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 4418–4427
November 16 - 20, 2020. ©2020 Association for Computational Linguistics

uations using DailyDialog and Twitter corpora. The results confirmed that our method largely outperformed the methods in previous studies and achieved sensitive control of the specificity of the output responses.

2 Related Work

Previous studies focus on addressing the dull response problem generated by Seq2seq models. Li et al. (2016a) rerank the N-best generated responses using an objective function to maximize the mutual information between the utterance and generated sentences. Because this method is post-processing, it ceases to be effective if there are no appropriate response candidates among the N-best responses. To directly improve the specificity of each response generated, previous studies devised training mechanisms of Seq2seq models by penalising for the generation of dull responses and eventually training models to generate specific responses. Yao et al. (2016) and Li et al. (2016b) apply reinforcement learning, and Xu et al. (2017) and Zhang et al. (2018b) apply generative adversarial networks, to directly generate specific responses. Based on the hypothesis that the specificity of sentences increases with the number of low-frequency words, Nakamura et al. (2019) and Jiang et al. (2019) propose loss functions weighted by word frequency. In contrast, to ensure both specificity and consistency, Takayama and Arase (2019) propose a model that directly promotes the generation of words that co-occur with uttered sentences on the basis of PPMI. Their model includes a mechanism for deciding whether or not to generate words of high co-occurrence with the utterance at each decoding step. In this study, we apply this method to our model for proactively generating specific words in a response.

Controlling the properties of generated responses is also related to our study. Xu et al. (2019) and Ko et al. (2019) allow for the control of dialogue-acts, length, and specificity of responses; however, they are resource intensive and thus require an external annotated corpus. In contrast, SC-Seq2Seq (Zhang et al., 2018a) achieves control of response specificity without dependence on external resources, which is most relevant to our study. Moreover, SC-Seq2Seq applies distant supervision, but uses word frequency in responses as a measure of specificity. At inference, SC-Seq2seq requires to input a desired specificity realized in the response. We measure specificity based on PPMI between an utterance and response, hence, our method can maintain both specificity and consistency to the utterance. Additionally, our method can estimate the maximum specificity for each input utterance, and automatically adjust the specificity of generated responses.

3 Proposed method

The proposed method is depicted in Figure 2. In the proposed method, first, a label that indicates the co-occurrence degree between utterance and response is automatically annotated by MaxPMI score (Section 3.1). The model generates sentences on the basis of previously calculated PPMI and MaxPMI (see Section 3.2). The training is performed using the framework of distant supervision based on the utterance–response pair and the MaxPMI score given beforehand (Section 3.3). At the inference, responses are generated using one method of inputting a manually determined specificity level or automatically estimated specificity level considering the input utterance (see Section 3.4).

Since we aim to explicitly control the amount of information in response to utterances, we use the decoder architecture of Takayama and Arase (2019) which has an output gating mechanism that controls whether or not to generate specific words at each decoding time-step.

3.1 MaxPMI: Co-occurrence measure between response and utterance

We propose a simple PPMI-based co-occurrence measure, called MaxPMI, which is based on the observation that a consistent and highly specific response contains words that highly co-occur with a specific utterance.

First, the PPMI of each word is calculated in advance using the all training corpus. $X = \{x^1, x^2, \ldots, x^{|\boldsymbol{X}|}\}$ is a word sequence in an utterance sentence, and $Y = \{y^1, y^2, \ldots, y^{|\boldsymbol{Y}|}\}$ is a word sequence in a response sentence. If the probabilities of word x of appearing in the utterance and response sentences are $p_X(x)$ and $p_Y(x)$, respectively, and if the probability of words x and y of simultaneously appearing in a certain utterance–response pair is $p(x, y)$, then the PPMI is calculated as follows:

$$\mathrm{PPMI}(x, y) = \max\left(\log_2 \frac{p(x, y)}{p_X(x) \cdot p_Y(y)}, 0\right).$$

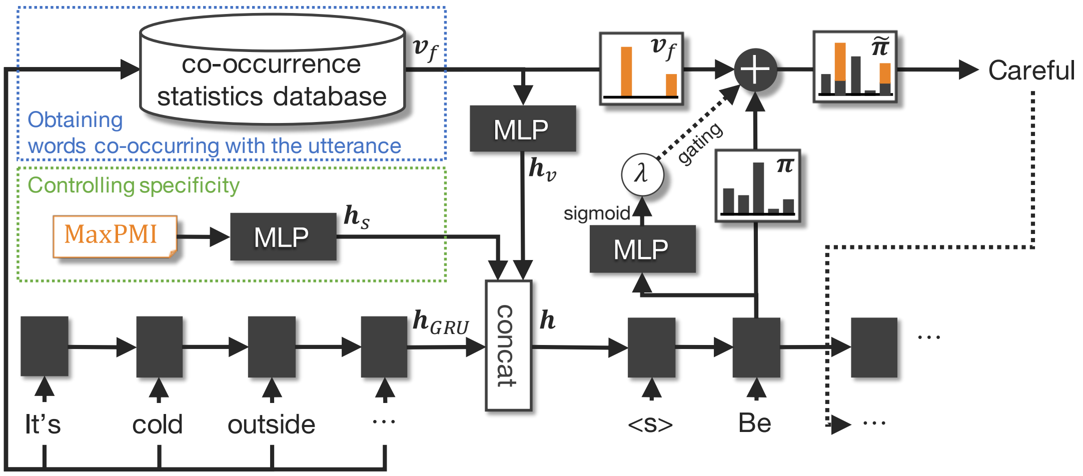

Figure 2: Model architecture

MaxPMI is defined as follows:

$$\mathrm{MaxPMI}(\boldsymbol{X}, \boldsymbol{Y}) = \max_{x \in \boldsymbol{X}, y \in \boldsymbol{Y}} \mathrm{PPMI}(x, y).$$

When training the model, MaxPMI shall be normalized to the range of $[0, 1]$ by using min-max normalization.

3.2 Model Architecture

Our model is based on Seq2seq architecture, which consists of an encoder and decoder, as follows.

Encoder Like in normal Seq2Seq, the tokens in the input sentence are first vectorized using the embedding layer, following which the input sentence is encoded using the gated recurrent units (GRU) (Cho et al., 2014) to obtain the vector $\boldsymbol{h}_{GRU}$. In addition, the proposed method includes a multilayer perceptron (MLP), which encodes the input MaxPMI score ($\mathrm{MaxPMI}(\boldsymbol{X}, \boldsymbol{Y})$) as $\boldsymbol{h}_s$. Subsequently, $\boldsymbol{h}_{GRU}$ and $\boldsymbol{h}_s$ are concatenated to form a vector $\boldsymbol{h}_e = \{\boldsymbol{h}_{GRU}; \boldsymbol{h}_s\}$, which is input to the decoder. The vector $\boldsymbol{h}_s$ conveys the decoder to the level of specificity with which the response should be generated.

Decoder The decoder has the same architecture as that in Takayama and Arase (2019), which promotes the generation of words of high co-occurrence with an input utterance. Let V be the vocabulary of the decoder. A word co-occurrence degree d_v between a word $v \in V$ and an input sentence X is defined as follows:

$$d_v = \sum_{x \in X} \mathrm{PPMI}\,(x, v)\,.$$

The decoder first receives a vector $\boldsymbol{v}_f = [d_0, \ldots, d_{|V|}] \in \mathbb{R}^{|V|}$ that contains the word co-occurrence degrees of all the vocabulary words. It then encodes $\boldsymbol{v}_f$ into a vector $\boldsymbol{h}_v$ using the multi-layer perceptron (MLP).

The initial state $\boldsymbol{h} = \{\boldsymbol{h}_e; \boldsymbol{h}_v\}$ of the decoder is concatenation of $\boldsymbol{h}_v$ and the encoder output $\boldsymbol{h}_e$. Consequently, the decoder can obtain the information of a word that co-occurs easily with the input. In addition, $\boldsymbol{v}_f$ is added with weighting to the output vector $\boldsymbol{\pi}^i$ of the decoder in each time step i to amplify the output probability of a word having a high amount of mutual information with the input sentence. The final output $\tilde{\boldsymbol{\pi}}^i$ of the decoder is given as follows:

$$\tilde{\boldsymbol{\pi}}^i = (1 - \lambda^i) \cdot \boldsymbol{\pi}^i + \lambda^i \cdot \boldsymbol{v}_f,$$

where generation of specific words is controlled by a parameter λ. We employ a gating mechanism using a sigmoid function (See et al., 2017) to determine the value of λ. Although previous literature discussed that the vanishing gradient problem could be caused by a sigmoid function (Goldberg and Hirst 2017, on page 46), See et al. (2017) have shown that the sigmoid-based gating is highly stable. λ^i is computed according to the decoder's current intermediate state $\boldsymbol{h}_i$ as follows:

$$\lambda^i = sigmoid\left(W_{gate}\boldsymbol{h}^i + \boldsymbol{b}_{gate}\right).$$

where W_{gate} is the trainable weight matrix and $\boldsymbol{b}_{gate}$ is the bias term.

3.3 Distant Supervision

MaxPMI score of an utterance–response pair $(\boldsymbol{X}, \boldsymbol{Y})$ in the training corpus is calculated for the distant supervision beforehand (Section 3.1). These scores are then input to the the decoder as $\boldsymbol{h}_s$ for training. The cross-entropy loss is used as the loss function:

$$\mathcal{L} = \sum_{(\boldsymbol{X},\boldsymbol{Y})\in\mathcal{D}} \log P(\boldsymbol{Y}|\boldsymbol{X}, \mathrm{MaxPMI}(\boldsymbol{X},\boldsymbol{Y});\theta),$$

where $\mathcal{D}$ denotes a training set and the model parameters are θ. Intuitively, this loss function allows the model to learn what response should be generated conditioned on an utterance and a specificity level.

3.4 Inference

At the inference, we can control the specificity of a response by inputting the score $s \in [0,1]$ to the model. The larger s makes the response more specific, *i.e.* the response contains words that frequently co-occurred among the utterances and responses of the training corpus. Users of our conversation model determine the desired specificity according to their use cases.

Situations also arise in which users prefer automatic control of the response specificity (rather than controlling it themselves). An appropriate value of s depends on an input utterance, *i.e.* there are utterances that could have specific responses or only typical responses. For example, the utterance in Figure 1 may have specific responses as depicted, but the utterance "Hello." most likely has typical responses like "Hi." Hence, we propose a method for estimating the appropriate s to generate a maximally specific response possible for the utterance. We define the upper bound of MaxPMI, s_{max}, for input sentence $\boldsymbol{X}$ as:

$$s_{max} = \max_{x\in\boldsymbol{X}, v\in\boldsymbol{V}} \mathrm{PPMI}(x,v),$$

which can be calculated using the precomputed PPMI values. By using s_{max}, the most specific response among possible responses of varying specificity to $\boldsymbol{X}$ is expected to be generated (referred to as *information-maximization* decoding).

4 Experimental Settings

To evaluate whether our model can control the specificity of the responses while maintaining their consistency with the utterances, we conducted response-generation experiments using Japanese and English chit-chat dialogue corpora.

4.1 Experiment Corpora

We used two corpora, Twitter (Japanese) and DailyDialog (English). The details of each corpus are as follows.

Twitter We crawled online conversations on Japanese Twitter by using the mentions of "@" as clues. A single-turn dialogue corpus was constructed by considering a tweet and its reply as an utterance–response pair. The sizes of the training/validation/test sets were $1,383,424$/$24,123$/$25,010$ utterance–response pairs, respectively. Each utterance–response pair was divided into subwords using a BertJapaneseTokenizer (bert-base-japanese) in transformers[1] (version = 2.5.1).

DailyDialog This corpus was constructed by Li et al. (2017) by crawling various websites that taught users English dialogues for daily usage. This consists of multi-turn dialogues, which we converted to a single-turn dialogues by considering two consecutive utterances as an utterance–response pair. The sizes of the training/validation/test sets were $76,052$/$7,069$/$6,740$ utterance–response pairs, respectively. Each utterance–response pair was divided into subwords using a BertTokenizer (BERT-base-uncased) in transformers.

As pre-processing, the subwords with frequencies less than 50 for both corpora were excluded for calculating the PPMI.

4.2 Comparison Methods

We compared our model to previous models. The baseline is the standard Seq2Seq (**Seq2Seq**). We also compared our model to **SC-Seq2Seq** (Zhang et al., 2018a) as it is the most relevant method for controlling the specificity of responses.

SC-Seq2Seq is a response generation model that can control the specificity of output sentences using the distant supervision. It hypothesizes that the lower the frequencies of words in a sentence, the higher the specificity of the sentence. As a measure of sentence specificity, it uses a frequency-based metric; inverse frequency of words. Moreover, SC-Seq2Seq also has a word prediction mechanism based on the Gaussian kernel layer in addition to the output layer of the decoder. Unlike our

[1] https://github.com/huggingface/transformers/

model, which takes into account the co-occurrence between utterances and responses, this word prediction layer takes into account the rarity of words. At the inference, the specificity of a response is controlled by inputting the specificity score $\in [0, 1]$.

4.3 Metrics for Automatic Evaluation

We employed several automatic-evaluation metrics typically used in the evaluation of conversation systems.

Metrics for Validity First, we evaluated the validity of the generated sentences in comparison with the reference sentences (responses) using **BLEU** and **NIST**. BLEU (Papineni et al., 2002) measures the correspondence between the n-grams in generated responses and those in the reference sentences. Liu et al. (2016) empirically show that BLEU has a higher Spearman's correlation with 5-scale human evaluation than some other reference-based metrics in experiments using the English Twitter corpus. NIST (Doddington, 2002) also measures the correspondence between generated responses and reference sentences. Unlike BLEU, NIST places lower weights on frequent n-grams, *i.e.* NIST regards content words as more important than function words. Thus, we regard that NIST is more suitable for evaluating the specificity aspects of the responses. We used Natural Language Toolkit[2] for calculation of BLEU and NIST scores.

Metrics for Diversity Second, we evaluated the diversity of the generated responses using **dist** and **ent**. Dist (Li et al., 2016a) is defined as the number of distinct n-grams in the generated responses divided by the total number of generated tokens. On the other hand, ent (Zhang et al., 2018b) considers the frequency of n-grams in generated responses as follows:

$$\text{ent} = -\frac{1}{\sum_w F(w)} \sum_{w \in Y} F(w) \log \frac{F(w)}{\sum_w F(w)},$$

where Y is a set of n-grams output by the system, and $F(w)$ computes the frequency of each n-gram. Compared to dist, which simply focuses on the number of types of words used in a response, ent focuses on the specificity of the response.

Metrics for Fluency Finally, we evaluated the repetition rate (Le et al., 2017) on the test set, which measures the meaningless repetition of words:

$$\text{repetition\_rate} = \frac{1}{N} \sum_{i=1}^{N} \frac{1 + r\left(\widetilde{Y}^i\right)}{1 + r(Y^i)},$$

where $\widetilde{Y}^i$ is the i-th generated sentence, Y^i is its reference, and N is the total number of test sentences. The function $r(\cdot)$ measures the repetition as the difference between the number of words and that of unique words in a sentence:

$$r(Y) = \text{len}(Y) - \text{len}(\text{set}(Y)),$$

where Y means a sentence, $\text{len}(Y)$ computes the number of words in Y, and $\text{set}(Y)$ removes the duplicate words in Y.

4.4 Human Evaluation Settings

Because appropriate responses for a certain utterance are diverse, human evaluation is crucial to properly evaluate conversation systems. We conducted human evaluation using the Japanese Twitter corpus. Specifically, we recruited six raters via crowd-sourcing, who were all Japanese native speakers and active users of Twitter. The raters evaluated the quality of 300 responses that were generated for randomly sampled utterances from the test set. All raters annotated the same set in parallel; each rater evaluated all the systems. In addition, we shuffled the set of responses to an utterance so that the raters did not distinguish which model each response was output from. The raters were recruited using Lancers,[3] a popular Japanese crowd-sourcing service.

The evaluation criteria were the same as those used in (Zhang et al., 2018a): +2: the response is not only semantically consistent and grammatical, but also specific; +1: the response is grammatical and can be used as a response to the utterance, but is too trivial (e.g., "I don't know"); +0: the response is semantically inconsistent or ungrammatical (e.g., grammatical errors). After collecting results from the raters, we adopted the results of the five raters and excluded one who had extremely low agreements with the others.

4.5 Model Settings

We used Adam (Kingma and Lei Ba, 2015) as an optimizer for training all the models with the learning rate to 0.0002. We also used gradient clipping

[2] https://www.nltk.org/

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

	BLEU-1	BLEU-2	NIST	dist-1	dist-2	ent-4	rep	length
Proposed ($s = s_{max}$)	**6.90**	**4.22**	**0.66**	**0.063**	**0.19**	**8.47**	2.68	6.08
Proposed ($s = 0.5$)	6.71	4.09	0.64	0.057	0.17	8.26	2.90	6.51
SC-Seq2Seq ($s = 0.8$)	6.54	4.00	0.62	0.010	0.02	5.65	1.90	5.45
Seq2Seq	5.36	3.53	0.41	0.008	0.02	4.00	**1.56**	4.08
Reference	100.00	100.00	16.85	0.110	0.51	11.17	1.00	6.11

Table 1: Automatic evaluation results on the Twitter corpus (Japanese)

	BLEU-1	BLEU-2	NIST	dist-1	dist-2	ent-4	rep	length
Proposed ($s = s_{max}$)	**22.30**	**17.62**	**2.87**	0.083	**0.41**	**10.77**	1.46	11.89
Proposed ($s = 0.5$)	22.06	17.41	2.85	0.085	0.41	10.74	1.41	11.63
SC-Seq2Seq ($s = 0.5$)	13.32	8.18	1.40	**0.098**	0.36	10.34	1.29	10.09
Seq2Seq	13.75	9.00	1.54	0.096	0.37	10.31	**1.26**	9.70
Reference	100.00	100.00	16.70	0.127	0.54	10.91	1.00	11.67

Table 2: Automatic evaluation results on the DailyDialog corpus (English)

to avoid the exploding gradient problem, with a threshold of 5. For all the models, the number of dimensions of the hidden and embedding layers was 512 and 256, respectively. The training was performed up to 40 epochs on Twitter corpus and 200 epochs on DailyDialog corpus, and the evaluation was conducted using the model with the highest BLEU score on the validation set.

SC-Seq2Seq has a hyper-parameter σ^2, which determines the variance of the Gaussian kernel layer. σ^2 was set to 0.1 for Twitter and 0.2 for DailyDialog, chosen from 0.1, 0.2, 0.5, and 1.0 to maximise the BLEU score on the validation set.

All the code used in the experiment was written using PyTorch[4] (version = 1.0.0). We use a single GPU (NVIDIA Tesla V100 SXM2, 32 GB memory) for both training and testing.

5 Results and Discussion

5.1 Automatic Evaluation Results

The automatic evaluation results on the test sets are presented in Tables 1 (Twitter) and 2 (DailyDialog), where the last columns show the average number of words per response. The proposed method ($s = s_{max}$; information-maximization decoding) achieved the highest scores on validity and diversity metrics (BLEU, NIST, dist, and ent) for most cases. These results confirms that the information-maximization decoding can generate a highly specific response by estimating the appropriate specificity level s. Compared with other methods, our model achieved much higher BLEU and NIST scores on DailyDialog. We hypothesize that this was because our model explicitly incorporates the co-occurrence statistics of words, which may complement the training of Seq2seq with a smaller corpus.

SC-Seq2seq showed comparable BLEU and NIST scores to our model on the Twitter corpus; however, its dist and ent scores were as low as Seq2seq. In contrast, SC-seq2seq scored high for dist and ent on the DailyDialog corpus, but its BLEU and NIST scores were lower than the standard Seq2seq. These results indicate that the effectiveness of SC-Seq2seq is domain dependent. We conjecture this is caused by the specificity estimation based on word frequencies regardless of utterances and responses, which is easily affected by occurrence of rare words.

As an adverse effect of the proposed method, the repetition rate is higher than that of Seq2Seq and SC-Seq2Seq in both corpora. The longer average length of responses and higher NIST and BLEU scores of the proposed model indicates that highly co-occurring words (in references) are repeatedly generated. This is because the probability of generating such words is always high, regardless of the state of the decoder, and it will be generated repeatedly. We will address this problem by adjusting $\boldsymbol{v}_f$ at each time-step in future.

5.2 Controllability Evaluation Results

We evaluated the controllabiity of the specificity of the generated responses using the automatic evalu-

[4] https://pytorch.org/

		BLEU-1	BLEU-2	NIST	dist-1	dist-2	ent-4	rep	length
Proposed	$s = 0.0$	0.05	0.03	0.00	0.007	0.03	2.39	0.94	1.28
	$s = 0.2$	4.71	2.96	0.36	0.035	0.10	6.39	1.81	4.09
	$s = 0.5$	6.95	4.26	0.68	0.058	0.17	8.15	2.93	6.54
	$s = 0.8$	5.91	3.45	0.56	0.046	0.15	8.12	3.97	8.25
	$s = 1.0$	5.63	3.23	0.53	0.039	0.13	8.09	4.23	8.72
	$s = s_{max}$	**7.20**	**4.46**	**0.70**	**0.064**	**0.19**	**8.41**	2.68	6.06
SC-Seq2Seq	$s = 0.0$	3.72	2.68	0.13	0.013	0.04	6.06	0.98	2.99
	$s = 0.2$	4.05	2.88	0.17	0.013	0.04	6.01	0.99	3.11
	$s = 0.5$	5.34	3.65	0.40	0.013	0.03	5.48	1.42	3.89
	$s = 0.8$	6.74	4.16	0.66	0.011	0.03	5.71	1.82	5.36
	$s = 1.0$	6.31	3.86	0.57	0.009	0.02	5.66	2.85	6.36

Table 3: Controllability Evaluation on Twitter corpus (Japanese)

Models	Rate (%)			Kappa
	+2	+1	+0	
Proposed ($s = s_{max}$)	24.8	19.8	55.4	0.42
Proposed ($s = 0.5$)	**26.6**	17.9	55.5	0.41
Proposed ($s = 0.0$)	0.6	53.8	45.6	0.02
Seq2Seq	10.1	59.7	30.3	0.42
SC-Seq2Seq ($s = 1.0$)	11.5	33.7	54.8	0.56
SC-Seq2Seq ($s = 0.8$)	9.8	54.7	35.5	0.50
SC-Seq2Seq ($s = 0.0$)	10.9	56.5	32.6	0.44
Proposed (hybrid)	22.0	38.2	39.8	–
SC-Seq2Seq (hybrid)	12.1	50.2	37.7	–

Table 4: Human evaluation results on the test set of Twitter corpus (Japanese)

ation metrics. For each utterance of the validation set, responses were generated using our model and SC-Seq2Seq, respectively.

The results are summarized in Table 3 (Twitter). Our model shows more sensitive variation for changing s than SC-Seq2Seq. Particularly, in the range of $s \leq 0.5$, as s increases, dist, which indicates diversity, and NIST, which indicates validity of responses, increase. However, in the range of $s \geq 0.5$, as s increases, almost all the scores decrease. These results show that it is impossible to generate an appropriate response when the inputted specificity level s is beyond the possible range for input utterances. It is evident that the repetition rate ('rep' in Table 3) and average length of responses increased as s became larger. This is because the decoder prefers words co-occurring with the utterance in accordance with a large s; and consequently, it repeatedly generated highly specific words for utterances.

The results of the proposed method ($s = s_{max}$) show the highest scores for all of BLEU, NIST, dist, and ent. Further, it achieves the lower repetition rate than the proposed method ($s = 0.5$), which performed best among different settings of s. This results show that the optimal s for each input utterance can be estimated by using information-maximization decoding. The same tendency was also observed in the DailyDialog corpus, whose results are omitted due to the space limitation.

5.3 Human Evaluation Results

The human evaluation results on the test set of Twitter corpus are presented in Table 4. Except for the proposed method ($s = 0.0$), the Kappa values for all the methods exceed 0.4. These Kappa values are similar to those obtained in the human evaluations performed in Zhang et al. (2018a). The low kappa value of 0.02 for the proposed method ($s = 0.0$) is caused by the frequent output of very short responses[5] such as "?" and "huh?", thereby making it difficult to determine whether a response is acceptable.

The proposed method ($s = 0.5$) and the proposed method ($s = s_{max}$) have more "+2"s than the proposed method ($s = 0.0$), which shows that our model generates specific responses by increasing s. The change in the ratio of the number of "+2"s to the change in s is more pronounced for our model than for each of the SC-Seq2Seq results. Thus, our model possesses more sensitive specificity control than SC-Seq2Seq. However, both of the proposed methods and SC-Seq2Seq show a significant increase in the rate of "+0" upon increasing s, compared to Seq2seq. This is because the fluency of the responses was deteriorated by forcing to output a larger number of specific words, which negatively affected to the language generation ability of the decoder. Particularly, as mentioned in Section 5.2, many responses might have lost their fluency because of repeated words.

To address this problem, we tried a simple heuristic to switch the proposed method and the plain Seq2seq. If the proportion of unique words in a re-

[5] the average sentence length is 1.46 words

Utterance	I’m always watching cat movies in free times. Cats of other families all look cute. (暇さえあれば猫動画観てる。他所の猫かわいく見える)
Proposed ($s = s_{max}$)	Movies of cat cafes* remind me how cute they are (猫カフェの動画見てるだけで可愛い)
Proposed ($s = 1.0$)	Cats are cute (猫可愛いですよね)
Proposed ($s = 0.8$)	Movies of cat cafes remind me how cute they are (猫カフェの動画見てるだけで可愛い)
Proposed ($s = 0.5$)	Movies of cat cafes remind me how cute they are (猫カフェの動画見てるだけで可愛い)
Proposed ($s = 0.2$)	Cute (可愛い)
Proposed ($s = 0.0$)	Thank you (ありがとう)
SC-Seq2Seq ($s = 1.0$)	Can’t agree more (それな)
SC-Seq2Seq ($s = 0.8$)	Are you interested in me? (私のことが気になるの?)
SC-Seq2Seq ($s = 0.5$)	Can’t agree more (それな)
SC-Seq2Seq ($s = 0.2$)	Can’t agree more! (それな!)
SC-Seq2Seq ($s = 0.0$)	Can’t agree more! (それな!)
Seq2Seq	Cute (可愛い)

Utterance	But, I’m afraid it’s going to be a nightmare as tomorrow’s a national holiday (でもあした祝日だからやばそう)
Proposed ($s = s_{max}$)	It can’t be helped. A national holiday is always like that. (祝日なんだから仕方ない)
Proposed ($s = 1.0$)	It’s not because of the national holiday, right? (祝日は関係ないからな)
Proposed ($s = 0.8$)	It can’t be helped. A national holiday is a national holiday. (祝日は祝日だから仕方ないね)
Proposed ($s = 0.5$)	It can’t be helped. A national holiday is always like that. (祝日だから仕方ない)
Proposed ($s = 0.2$)	Hang in there (がんばれがんばれ)
Proposed ($s = 0.0$)	Thank you (ありがとう)
SC-Seq2Seq ($s = 1.0$)	No way, I can’t stand it (それは無理だわ)
SC-Seq2Seq ($s = 0.8$)	I should go to a part-time job now (俺は今からバイトだから)
SC-Seq2Seq ($s = 0.5$)	No way, I can’t stand it (それは無理だわ)
SC-Seq2Seq ($s = 0.2$)	I have to work tomorrow (明日は仕事だよ)
SC-Seq2Seq ($s = 0.0$)	Good morning! (おはよー!)
Seq2Seq	Can’t agree more (それな)

Table 5: Examples of generated responses in test set of Twitter corpus. The English sentences in the table was translated from the original Japanese sentences, written in parentheses. (*A “cat cafe” is a cafe where people can play with cats.)

sponse sentence generated by our model falls below a threshold T (we set T to 0.95), *i.e.* the response contains repetitive words, we switch to the plain Seq2seq and use its response instead. The results obtained after applying this heuristic to the proposed method ($s = s_{max}$) as well as SC-Seq2Seq ($s = 1.0$) are listed in Table 4 as the proposed method (hybrid) and SC-Seq2Seq (hybrid), respectively. For both the proposed method (hybrid) and SC-Seq2Seq (hybrid), the ratio of “+0” decreases by more than 15 percentage points, while that of “+2” remains almost unchanged. This problem will be addressed using a more sophisticated approach in future work.

5.4 Case Study

Table 5 presents two examples of generated responses sampled from the test set of the Twitter corpus. In the range of $s \geq 0.5$, our model generated highly specific responses to the utterances. However, it repeatedly generated the same phrase when s was too large, *i.e.* the response on $s = 0.8$ for the second case. As mentioned in the Section 5.1, this is an adverse effect of forcing to output a larger number of specific words than possible. In contrast, the information-maximization decoding ($s = s_{max}$) avoids this problem by adaptively setting an appropriate s value for each input utterance.

SC-Seq2Seq often produced more specific responses than Seq2Seq as shown in the second example. However, the change in the specificity of responses is limited even though inputting a large value of s, like the first example. Specifically, the response by SC-Seq2Seq ($s = 0.8$) in the first case ignores the input utterance and thus is inconsistent. We conjecture this is caused by that the specificity in SC-Seq2Seq is estimated regardless of utterances and responses. For the same example, our model can output words that are associated with the utterance, such as “cat”, “movie”, and “cute”.

6 Conclusion

We empirically showed that the co-occurrence relationship between words in an utterance and words in its response helps to control the specificity in response generation. The conventional specificity control model often generates responses with less consistency with the utterances. In contrast, our model can control specificity of the responses while maintaining the consistency with the utterance.

As future work, we shall improve the proposed method to maintain the fluency in responses by addressing the repeated word problem. Further, an appropriate specificity level of a response depends on the previous utterances and responses, *i.e.* conversation systems that always return highly specific responses are annoying. Hence, we intend to propose a method to adjust the specificity level considering the conversation history.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number JP18K11435.

References

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734.

Richárd Csáky, Patrik Purgai, and Gábor Recski. 2019. Improving Neural Conversational Models with Entropy-Based Data Filtering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 5650–5669.

George Doddington. 2002. Automatic Evaluation of Machine Translation Quality Using N-gram Co-Occurrence Statistics. In *Proceedings of the second international conference on Human Language Technology Research (HLT)*.

Yoav Goldberg and Graeme Hirst. 2017. *Neural Network Methods in Natural Language Processing*. Morgan & Claypool Publishers.

Shaojie Jiang, Pengjie Ren, Christof Monz, and Maarten de Rijke. 2019. Improving Neural Response Diversity with Frequency-Aware Cross-Entropy Loss. In *Proceedings of the Web Conference*.

Diederik P Kingma and Jimmy Lei Ba. 2015. ADAM: A Method for Stochastic Optimization. In *The 3rd International Conference on Learning Representations 2015 (ICLR)*.

Wei-Jen Ko, Greg Durrett, and Junyi Jessy Li. 2019. Linguistically-Informed Specificity and Semantic Plausibility for Dialogue Generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Volume 1 (Long and Short Papers)*, pages 3456–3466, Minneapolis, Minnesota.

An Nguyen Le, Ander Martinez, Akifumi Yoshimoto, and Yuji Matsumoto. 2017. Improving Sequence to Sequence Neural Machine Translation by Utilizing Syntactic Dependency Information. In *Proceedings of the The 8th International Joint Conference on Natural Language Processing (IJCNLP)*, pages 21–29.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. A Diversity-Promoting Objective Function for Neural Conversation Models. In *Proceedings of The 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 110–119.

Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. 2016b. Deep Reinforcement Learning for Dialogue Generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1192–1202.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. DailyDialog: A manually labelled multi-turn dialogue dataset. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (IJCNLP), Volume 1: Long Papers*, pages 986–995.

Chia-Wei Liu, Ryan Lowe, Iulian V Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How NOT To Evaluate Your Dialogue System: An Empirical Study of Unsupervised Evaluation Metrics for Dialogue Response Generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2122–2132.

Hongyuan Mei, Mohit Bansal, and Matthew R. Walter. 2017. Coherent Dialogue with Attention-Based Language Models. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, San Francisco, CA.

Ryo Nakamura, Katsuhito Sudoh, Koichiro Yoshino, and Satoshi Nakamura. 2019. Another Diversity-Promoting Objective Function for Neural Dialogue Generation. In *Proceedings of The Second AAAI Workshop on Reasoning and Learning for Human-Machine Dialogues (DEEP-DIAL)*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318.

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 (ACL), Volume 1: Long Papers*, pages 1073–1083.

Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 196–205.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of The Twenty-eighth Conference on Neural Information Processing Systems (NeurIPS)*, pages 3104–3112.

Junya Takayama and Yuki Arase. 2019. Relevant and Informative Response Generation using Pointwise Mutual Information. In *Proceedings of the First Workshop on NLP for Conversational AI*, pages 133–138.

Oriol Vinyals and Quoc V Le. 2015. A Neural Conversational Model. In *Proceedings of the 32nd International Conference on Machine Learning (ICML)*.

Can Xu, Wei Wu, Chongyang Tao, Huang Hu, Matt Schuerman, and Ying Wang. 2019. Neural Response Generation with Meta-words. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 5416–5426.

Zhen Xu, Bingquan Liu, Baoxun Wang, Chengjie Sun, Xiaolong Wang, Zhuoran Wang, and Chao Qi. 2017. Neural Response Generation via GAN with an Approximate Embedding Layer. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 617–626.

Kaisheng Yao, Baolin Peng, Geoffrey Zweig, and Kam-Fai Wong. 2016. An Attentional Neural Conversation Model with Improved Specificity. *arXiv preprint arXiv:1606.01292*.

Ruqing Zhang, Jiafeng Guo, Yixing Fan, Yanyan Lan, Jun Xu, and Xueqi Cheng. 2018a. Learning to control the specificity in neural response generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 1108–1117.

Yizhe Zhang, Michel Galley, Jianfeng Gao, Zhe Gan, Xiujun Li, Chris Brockett, and Bill Dolan. 2018b. Generating Informative and Diverse Conversational Responses via Adversarial Information Maximization. In *Proceedings of 32nd Conference on Neural Information Processing Systems (NeurIPS)*.