Relevant and Informative Response Generation using Pointwise Mutual Information

Junya Takayama[†] and **Yuki Arase**[†‡]
[†]Graduate School of Information Science and Technology, Osaka University
[‡]Artificial Intelligence Research Center (AIRC), AIST
{takayama.junya, arase}@ist.osaka-u.ac.jp

Abstract

A sequence-to-sequence model tends to generate generic responses with little information for input utterances. To solve this problem, we propose a neural model that generates relevant and informative responses. Our model has simple architecture to enable easy application to existing neural dialogue models. Specifically, using positive pointwise mutual information, it first identifies keywords that frequently co-occur in responses given an utterance. Then, the model encourages the decoder to use the keywords for response generation. Experiment results demonstrate that our model successfully diversifies responses relative to previous models.

1 Introduction

Neural networks are common approaches to building chat-bots. Vinyals and Le (2015) have proposed a neural dialogue model using sequence-to-sequence (Seq2Seq) networks (Sutskever et al., 2014) and achieved fluent response generation. Because a Seq2Seq model uses a word-by-word loss function at the time of training, any words outside the reference are penalized equally. Consequently, the Seq2Seq model tends to generate generic responses that consist of frequent words, such as "Yes" and "I don't know." This is a central concern in neural dialogue generation. To tackle this problem, Li et al. (2016) proposed a model for considering mutual dependency between an utterance and response modeled by maximum mutual information (MMI). However, their model disregarded the aspect of informativeness of responses, which is also important for user experience of chat-bots.

To solve this problem, we propose a response generation model that outputs diverse words while preserving relevance in response to the input utterance. In our model, Positive Pointwise Mutual Information (PPMI) identifies keywords from a large-scale conversational corpus that are likely to appear in the response to an input utterance. Then, the model modifies the loss function in a Seq2Seq model to reward responses using the identified keywords. In order to calculate the loss function using the words output by the decoder, we need to sample words from the probability distribution of the output layer. Hence, we apply the Gumbel-Softmax trick (Jang et al., 2017) as a differentiable pseudo-sampling method.

Experiments using a Japanese dialogue corpus crawled from Twitter and OpenSubtitles revealed that the proposed model outperformed (Li et al., 2016) for all automatic evaluation metrics for correspondence to references and diversity in outputs.

2 Related Work

The generic response problem has been actively studied. Yao et al. (2016) and Nakamura et al. (2019) proposed models that constrain decoders to directly suppress generation of frequent words. Yao et al. (2016) diversified the response by a loss function in which words with high inverse document frequency values are preferred. Nakamura et al. (2019) proposed a loss function that adds weights based on the inverse of the word frequency. Xing et al. (2017) proposed a model using topic words extracted from utterances. Their model ensembles words predicted using the topic words and the words predicted by the decoder.

All of the methods described above only focus on the amount of a information in a response. Therefore, generated responses tend to lack relevance to input utterances. MMI-bidi (Li et al., 2016) solves this problem by approximating the PMI between the utterance Q and the generated

Proceedings of the 1st Workshop on NLP for Conversational AI, pages 133–138
Florence, Italy, August 1, 2019.

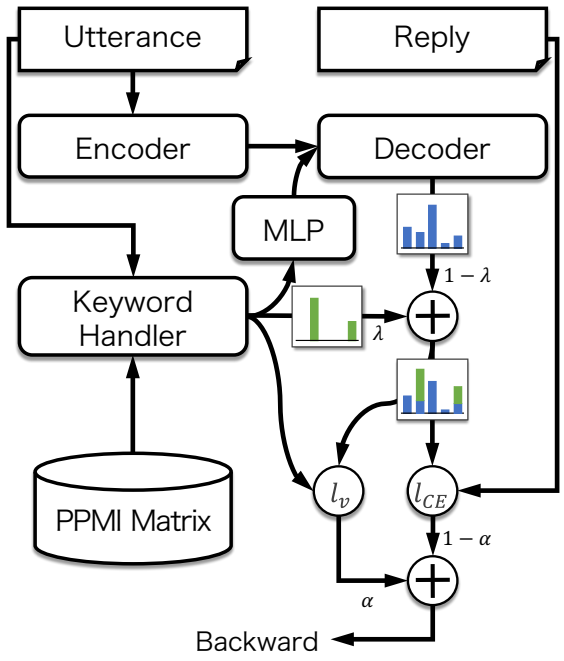

Figure 1: Outline of the proposed model

response R as follows:

$$\mathrm{PMI}(Q,R) = (1-\lambda)\log P(R|Q) + \lambda \log P(Q|R). \quad (1)$$

Here, both $P(R|Q)$ and $P(Q|R)$ are computed by independent Seq2Seq models. Specifically, the N-best candidate responses generated by the former model are re-ranked by Equation (1). MMI-bidi exhibited a strong performance for diversifying responses while preserving relevance to an input utterance. However, its effects depend on the diversities of the N-best candidate responses. If these responses are diverse, MMI-bidi can improve futher.

3 Proposed Model

Figure 1 shows the outline of the proposed model. It first identifies keywords that strongly co-occur between utterances and their responses in a training corpus using PPMI (section 3.1). The decoder then uses Gumbel-Softmax to sample words in the output layer (section 3.3). Finally, it computes the proportion of output words matching the keywords, and add weights to the loss function (section 3.4).

3.1 Keywords Retrieval Based on Positive Pointwise Mutual Information

The keyword handler retrieves words that are likely to appear in the response to a certain input utterance based on PPMI, calculated in advance from an entire training corpus. Let $P_Q(x)$ and $P_R(x)$ be probabilities that the word x will appear in a certain utterance and response sentences, respectively. Also, let $P(x,y)$ be the probability that the words x and y exist in the utterance and response sentence pair. PPMI is calculated as follows:

$$\mathrm{PPMI}(x,y) = \max\left(\log_2 \frac{P(x,y)}{P_Q(x)\cdot P_R(y)}, 0\right).$$

The pair of x and y and its PPMI score are saved in the PPMI Matrix in Figure 1. At the time of response generation, the keyword handler looks up the PPMI Matrix. Let the word set of a certain utterance sentence be $Q = \{q_1, q_2, \ldots, q_L\}$, and the vocabulary in the decoder be $V_R = \{v_{R_1}, v_{R_2}, \ldots, v_{R_N}\}$. The keyword-score of a word $v_{R_n} \in V_R$ is defined as follows:

$$\sum_{q\in Q} \mathrm{PPMI}\left(q, v_{R_n}\right).$$

Keyword-scores are calculated for all words in V_R. Then top-k words are set as keywords V_{Pred} used in the loss function.

3.2 Decoding Response Sentences using Retrieved Keywords

The decoder first receives a vector $\mathbf{v_f}$ consisting of keyword-scores for all words in the vocabulary, and non-linearly transforms v_f through a multilayer perceptron (MLP). This vector is concatenated with the output of the encoder, and then set to the initial state of the decoder. By doing so, we expect that the decoder considers the keyword-scores. In order to directly boost the probability to output the keywords, we add weighted $\mathbf{v_f}$ to the decoder output vector $\pi_\mathbf{i}$ at each time step i. The final decoder output $\tilde{\pi}_\mathbf{i}$ is represented by the following equation:

$$\tilde{\pi}_\mathbf{i} = (1-\lambda_i)\cdot \pi_\mathbf{i} + \lambda_i \cdot \mathbf{v_f}.$$

λ_i balances the effects of the decoder output and v_f. λ_i is calculated as follows based on the current intermediate state $\mathbf{h}_i$ of the decoder:

$$\lambda_i = \sigma\left(W^{gate}\mathbf{h}_i + \mathbf{b}^{gate}\right),$$

where W^{gate} is a trainable weight matrix, $\mathbf{b}^{gate}$ is a bias term, and $\sigma(\cdot)$ is a sigmoid function.

3.3 Pseudo-sampling of Generated Words using Gumbel-Softmax

In order to determine whether the decoder generated words in V_Pred, it is necessary to sample words generated by the decoder. However, sampling based on argmax, which is generally used at the decoder, disallows back propagation because of its discrete nature. Jang et al. (2017) proposed Gumbel-Softmax which performs pseudo sampling from the probability distribution to allow back propagation. Gumbel-Softmax performs the following calculations for a probability distribution π (corresponding to the output layer in the decoder) for k classes:

$$y_i = \frac{\exp((\log(\pi_i) + g_i)/\tau)}{\sum_{j=1}^{k} \exp((\log(\pi_j) + g_j)/\tau)}.$$

Here, τ is a hyperparameter called temperature. Smaller ${}^t au$ makes the vector closer to one-hot but the dispersion of the gradient becomes larger. g_i is obtained by the following calculation using uniform distribution $u_i \sim \mathrm{Uniform}(0, 1)$:

$$g_i = -\log(-\log(u_i)).$$

In the proposed model, Gumbel-Softmax is applied to the final decoder output vector $\tilde{\pi}$ at each time step i as in Equation (2). Then, we obtain the differentiable pseudo-bag-of-words vector $\mathbf{B}$.

$$\mathbf{B} = \sum_{i=1}^{T} \mathrm{GumbelSoftmax}(\tilde{\pi}_i). \quad (2)$$

3.4 Loss function

We design a loss function l_v which value decreases as words contained in V_{Pred} are generated. Thus, the decoder outputs more words that strongly co-occur with the input utterance. Specifically, when $t(b_n)$ is the word corresponding to the n-th index in $\mathbf{B}$, l_v is defined as follows.

$$l_v = -\sum_{n=0}^{N} f(b_n, V_{\mathrm{Pred}}),$$

$$f(b_n, V_{\mathrm{Pred}}) = \begin{cases} \min(b_n, 1) & (t(b_n) \in V_{\mathrm{Pred}}), \\ 0 & (\mathrm{otherwise}). \end{cases} \quad (3)$$

We use $\min(b_n, 1)$ in Equation (3) to avoid adding a reward when a keyword is generated multiple times. This aims to suppress the decoder outputs the same word many times.

Finally, the loss function $\mathcal{L}$ is defined as a liner interpolation of l_{CE} of the cross-entropy error and the l_v:

$$\mathcal{L} = (1 - \alpha) \cdot l_{CE} + \alpha \cdot l_v.$$

α is a hyperparameter that balances the degree of rewards based on the keywords.

4 Experiments

We empirically evaluate how our model avoids generic responses to generate relevant and informative responses.

4.1 Datasets

We used two datasets, OpenSubtitles (English) and Twitter (Japanese). The details of each dataset are as follows.

OpenSubtitles OpenSubtitles (Tiedemann, 2009) is a large scale open-domain corpus composed of movie subtitles.

Like Vinyal et al. (Vinyals and Le, 2015) and Li et al (Li et al., 2016), we assumed that each line of the subtitles represents an independent utterance, and constructed a single-turn dialogue corpus by regarding two consecutive utterances as an utterance-response pair. We randomly sampled 2 million utterance-response pairs. All sentences were tokenized using the Punkt Sentence Tokenizer of nltk [1].

Twitter We crawled conversations in Japanese Twitter using "@" mention as a clue. A single-turn dialogue corpus was constructed by regarding a tweet and its reply as an utterance-response pair. The dataset consists of about 1.3 million utterance-response pairs. All sentences were tokenized by MeCab [2].

In both datasets, 10k utterance-response pairs were separated as validation data, another 10k were separated as test data, and the rest were used as training data.

4.2 Comparison Methods

We compared our model to previous models. The baseline is the standard Seq2Seq (**Seq2Seq**). We also compared to MMI-bidi (**Seq2Seq + MMI**)

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

[2] http://taku910.github.io/mecab/

	BLEU	NIST	dist-1	dist-2	ent-4	length	repetition
Proposed + MMI	**1.577**	**0.872**	**0.050**	**0.187**	**8.536**	8.064	1.551
Proposed	1.569	0.837	0.044	0.148	7.327	7.520	1.377
Seq2Seq + MMI	1.373	0.739	0.009	0.032	5.600	7.566	1.223
Seq2Seq	1.374	0.687	0.005	0.015	4.070	8.025	1.095
Reference	100.000	16.498	0.086	0.482	10.647	7.671	1.000

Table 1: Results on the OpenSubtitle corpus (English)

	BLEU	NIST	dist-1	dist-2	ent-4	length	repetition
Proposed + MMI	**2.611**	0.573	**0.071**	**0.204**	**8.979**	7.913	1.832
Proposed	2.591	**0.583**	0.068	0.188	8.738	8.044	1.902
Seq2Seq + MMI	2.262	0.304	0.043	0.102	7.578	6.791	1.416
Seq2Seq	2.237	0.318	0.040	0.091	7.103	6.920	1.518
Reference	100.000	16.562	0.105	0.496	11.311	12.262	1.000

Table 2: Results on the Twitter corpus (Japanese)

because it is the most relevant method for diversifying responses. In addition, we combined our model with MMI-bidi (**Proposed + MMI**) to see whether it contributes to diversification of the N-best candidates.

4.3 Evaluation Metrics

We employed several automatic evaluation metrics. **BLEU** and **NIST** measure the validity of generated sentences in comparison with references. BLEU (Papineni et al., 2002) measures the correspondence between n-grams in generated responses and those in reference sentences. Following Papineni et al. (2002), we used the average of BLEU scores from 1-gram to 4-gram in the experiment. 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. In the experiment, we used the average of NIST from 1-gram to 5-gram.

In addition, **dist** and **ent** measure the diversity of generated responses. Dist (Li et al., 2016) is defined as the number of distinct n-grams in generated responses divided by the total number of generated tokens. On the other hand, ent (Zhang et al., 2018) considers the frequency of n-grams in generated responses:

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

where X is a set of n -grams output by the system, and $F(w)$ computes the frequency of each n-gram.

In this paper, we focus on automatic evaluation. Human evaluation is our future work.

4.4 Parameter Settings

For all models, we implemented the encoder and decoder of each model using 1-layer GRUs. The dimension of the GRU was set to 512. However, only the decoder of the proposed model used 1024-dimensional GRU. This is because the initial state of the decoder is the concatenation of the keyword-score vector and the output from the encoder (512-dimension for each). Both the encoder and decoder had a word embedding layer of 256 dimensions.

The vocabulary consisted of words that appeared more than 15 times in the training data. Words that occured less than 15 times were replaced with the "<unk>" token. The vocabulary size was 41.5k for the Twitter model and 20.9k for the OpenSubtitles model.

4.5 Results and Error Analysis

The left sides of Tables 1 and 2 show BLEU, NIST, dist, and ent scores for OpenSubtitles and Twitter, respectively. Our model (Proposed) outperformed Seq2Seq and MMI-bidi (Seq2Seq) in all evaluation metrics across the datasets. Furthermore, our model combined with MMI-bidi (Proposed+MMI) achieved the best performance, except for NIST, on the Twitter dataset. This result demonstrates that our method successfully gener-

Input	oh, we just actually wanted to be alone
Reference	get me out of here
Proposed+MMI	i wanted to be alone
Seq2Seq+MMI	you know what ?
Input	should we call an ambulance ?
Reference	praise the lord !
Proposed+MMI	we 'r e going to have a call from the ambulance
Seq2Seq+MMI	i don 't think so
Input	are you tired already ?
Reference	you 'r e such a weakling
Proposed+MMI	i 'm tired of getting warmed up
Seq2Seq+MMI	yes i 'm fine
Input	i have no idea how it was spread
Reference	not from me
Proposed+MMI	i don 't know how much longer i can stand up
Seq2Seq+MMI	i <unk>never been able to find a way
Input	why do you have to carry the burden of someone else 's life ?
Reference	he 's my father
Proposed+MMI	because you 'r e the best commodity in the living room
Seq2Seq+MMI	i don 't know
Input	the little guy was kinda funny lookin
Reference	in what way ?
Proposed+MMI	i kinda like him kinda funny funny
Seq2Seq+MMI	i don 't know what you 'r e talking about
Input	bishop wood at st <unk>'s cathedral
Reference	holy christ blanche
Proposed+MMI	he was baptized in a parson 's church and he was baptized
Seq2Seq+MMI	he was the king of <unk><unk>

Table 3: Example responses generated by the proposed model and (Li et al., 2016)

ates diverse responses, which effectively improves the N-best candidates reranked by MMI-bidi. It is notable that improvements on NIST, which appreciates less frequent n-grams, support this idea the proposed model improves the informativeness of responses. The improvement is larger on the Twitter dataset, where the proposed method (Proposed) achieved NIST score 0.265 points higher than Seq2Seq even though MMI-bidi is inferior to Seq2Seq.

The example responses generated by Proposed+MMI and Seq2Seq+MMI using OpenSubtitles are shown in Table 3. The examples from the top to the third rows show that the proposed model generates more content words relevant to the content words in the utterance. On the other hand, Seq2Seq+MMI ended up generating fewer informative responses using generic words. The fourth and fifth examples show that the proposed model generated responses with little relevance to the input, although they were more informative than the responses generated by Seq2Seq+MMI.

The last two examples show a drawback of the proposed model, *i.e.*, which is over-generation of the same word. For quantitative evaluation, we computed the repetition rate (Le et al., 2017) on the test data, which measures the meaningless repetition of words. The repetition rate is defined as:

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

where $\widetilde{y}_i$ is the i-th generated sentence in the test data, 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:

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

where X means words in a sentence, $\text{len}(X)$ computes the number of items in X, and $\text{set}(X)$ re-

moves duplicate items in X. The average lengths of generated responses and repetition rates are shown on the right sides of Tables 1 and 2. The results show that the proposed models (Proposed and Proposed+MMI) tend to generate longer responses than Seq2Seq, but their repetition rates are also higher. This may be caused by time-invariant keyword-scores, despite the fact that the decoder output changes over time. In the future, we will update the keyword-score vector to avoid repetition in responses.

5 Conclusion

Aiming at generating diverse responses while preserving relevance to the input, we proposed a model that identifies keywords using PPMI and promoted their generation in the decoder. Evaluation results using English and Japanese conversational corpora show that in comparison with (Li et al., 2016), the proposed model achieved better performance in terms of correspondence to references and diversity of output. On the other hand, we found that the proposed model has a tendency of over-generation.

As future work, we will conduct human evaluation and qualitative analysis. We will also investigate the effects of the hyper-parameter α on overall performance. We also plan to develop a mechanism for suppressing over-generation.

Acknowledgments

This project is funded by Microsoft Research Asia, Microsoft Japan Co., Ltd., and JSPS KAKENHI Grant Number JP18K11435.

References

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 2002)*.

Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical Reparametarization with Gumbel-Softmax. In *Proceedings of The 5th International Conference on Learning Representations (ICLR 2017)*.

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 2017)*, pages 21–29.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. 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 2016)*, pages 110–119.

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 2002)*, pages 311–318.

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 2014)*, pages 3104–3112.

Jörg Tiedemann. 2009. News from opus-A collection of multilingual parallel corpora with tools and interfaces 1 Index of Subjects and Terms 13 vi News from OPUS-A Collection of Multilingual Parallel Corpora with Tools and Interfaces. *Recent advances in natural language processing*, 5:237–248.

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

Chen Xing, Wei Chung Wu, Yu Ping Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. 2017. Topic Aware Neural Response Generation. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI 2017)*.

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

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