applied sciences

MDPI

Article

Knowledge-Grounded Chatbot Based on Dual Wasserstein Generative Adversarial Networks with Effective Attention Mechanisms

Sihyung Kim [1], Oh-Woog Kwon [2] and Harksoo Kim [3,*]

[1] Computer and Communications Engineering, Kangwon National University, Chuncheon 24341, Korea; sureear@kangwon.ac.kr
[2] Language Intelligence Research Lab., Electronics and Telecommunications Research Institute, Daejeon 34129, Korea; ohwoog@etri.re.kr
[3] Computer Science and Engineering, Konkuk University, Seoul 05029, Korea
* Correspondence: nlpdrkim@konkuk.ac.kr; Tel.: +82-2-450-3499

Received: 16 April 2020; Accepted: 8 May 2020; Published: 11 May 2020

Featured Application: Core technology for intelligent virtual assistants.

Abstract: A conversation is based on internal knowledge that the participants already know or external knowledge that they have gained during the conversation. A chatbot that communicates with humans by using its internal and external knowledge is called a knowledge-grounded chatbot. Although previous studies on knowledge-grounded chatbots have achieved reasonable performance, they may still generate unsuitable responses that are not associated with the given knowledge. To address this problem, we propose a knowledge-grounded chatbot model that effectively reflects the dialogue context and given knowledge by using well-designed attention mechanisms. The proposed model uses three kinds of attention: Query-context attention, query-knowledge attention, and context-knowledge attention. In our experiments with the Wizard-of-Wikipedia dataset, the proposed model showed better performances than the state-of-the-art model in a variety of measures.

Keywords: knowledge-grounded chatbot; multi-turn chatbot; context-knowledge attention mechanism

1. Introduction

The ultimate goal of natural language processing is to let humans and machines freely communicate with each other. In human-to-human conversations, people communicate based on contextual knowledge (i.e., knowledge obtained from previous utterances) or external knowledge (i.e., knowledge obtained from various media). A knowledge-grounded chatbot is a dialogue system that can communicate by recalling internal and external knowledge (similar to a human). Knowledge-grounded chatbots must understand the language, store internal/external knowledge during the conversations, and respond in various ways based on the stored knowledge. Table 1 lists the differences between the responses of a conventional chatbot (utterances 1-1 and 2-1) and a knowledge-grounded chatbot (utterances 1-2 and 2-2). As shown in Table 1, the conventional chatbot provides general responses such as "I see" and "Yes, it is" because it has no knowledge about "Avengers." However, the knowledge-grounded chatbot provides specific responses (such as "Sure, I like Iron Man" and "Oh! Anthony and Joe Russo directed it") based on external knowledge obtained from the documents retrieved by an information retrieval system.

Table 1. Examples of differences between conventional chatbots and knowledge-grounded chatbots.

ID	Speaker	Utterance
1	User	Hi! Do you know Avengers is coming out today?
1-1	General chatbot	I see.
1-2	Knowledge-grounded chatbot	Sure, I like Iron Man.
2	User	It looks so funny.
2-1	General chatbot	Yes, it is.
2-2	Knowledge-grounded chatbot	Oh! Anthony and Joe Russo directed it!

To generate these specific responses, the knowledge-grounded chatbot must possess the following three abilities.

- Knowledge retrieval: A chatbot should be able to search for documents associated with the conversation topics.
- Knowledge extraction: A chatbot should be able to extract knowledge from the retrieved documents.
- Response generation: A chatbot should be able to generate responses that reflect the extracted knowledge.

In open-domain conversations, it is difficult for a chatbot to find documents that are closely associated with the current conversation topic, because a multi-turn conversation references many topics. Although we assume that the chatbot can find some documents associated with the current conversation topic by using a state-of-the-art information retrieval system, it is difficult for the chatbot to extract knowledge from the search results because the document may contain diverse knowledge on the given topic [1]. In addition, it is difficult for the chatbot to generate appropriate responses that reflect the acquired knowledge because the conversation history (i.e., contextual information based on previous utterances) should also be considered. In this study, we ignore the knowledge retrieval problem; instead, we focus on addressing the problems of knowledge extraction and response generation. In particular, we propose a knowledge-grounded multi-turn chatbot model that generates responses by considering new knowledge from previous utterances and a given document. This paper is organized as follows. In Section 2, we review the previous works on generative chatbot models and describe the proposed knowledge-grounded chatbot model in Section 3. In Section 4, we explain the experimental setup and report on some experimental results. Section 5 concludes our study.

2. Related Work

Chatbot models are divided into retrieval-based models and generative models. The retrieval-based models predefine a large amount of question-answer pairs. Then, they match users' queries against predefined questions by using well-designed information retrieval models. Next, they return answers of the matched questions as responses of users' queries [2,3]. Since the retrieval-based models predefine well-purified question-answer pairs in advance, they do not return responses including grammatical errors. However, they have some deficiencies that chatbots' responses are restricted into a predefined set and are not sensitive to changes of users' queries. To overcome these deficiencies, generative models that automatically select word sequences proper to a response have been recently proposed. Most recent studies on generative chatbots primarily use the sequence-to-sequence (Seq2Seq; encoder-decoder) model [4,5]. Generally, Seq2Seq models consist of two recurrent neural networks (RNNs) called an encoder for embedding input sentences and a decoder for generating output sentences. Previous studies have some deficiencies that Seq2Seq-based chatbots often generate safe and short responses such as "I don't know" and "OKay" [6,7]. To overcome this problem, a maximum mutual information (MMI) model and a variational autoencoder (VAE) model have been proposed [6–9]. To diversify responses, in the MMI model, the standard objective function (the

log-likelihood of a response given a query) is replaced with a new objective function based on mutual information between a response and a query [6]. The VAE model is possible to give diversity to a response by learning latent variables from inputs and outputs. However, the VAE model has collapse problems that the decoder is trained to ignore the latent variable by simplifying them to a standard normal distribution [10,11]. In other words, although the real response has a very complex distribution, the latent variable of the VAE model has a posterior collapse problem because it learns a simple prior distribution as the normal distribution. Some previous studies partially solved this problem by using adversarial learning [11]. However, adversarial learning has generated a discrete token that cannot be differentiated and, thus, cannot be learned [12,13]. To solve this problem, various studies have proposed using adversarial learning not for generating responses but for generating latent variables.

There have been various studies on knowledge-grounded chatbots (i.e., chatbot models that actively acquire and use knowledge for generating responses). The memory-to-sequence (Mem2Seq) model acquires structured knowledge and contextual information through a memory network [14]. The neural knowledge diffusion model automatically acquires knowledge associated with the entities in the previous utterances by looking up a knowledge base [15]. However, existing models have some limitations. First, they require considerable time to build new external knowledge bases. Second, the knowledge included in their responses is restricted to predefined knowledge bases [14–18]. Therefore, most open-domain knowledge-grounded chatbots extract knowledge from unstructured texts and generate responses using the extracted knowledge [19–21].

3. Knowledge-Grounded Chatbot Model

The proposed model comprises four submodules: A context encoder, knowledge encoder, latent variable generator, and response generator, as shown in Figure 1.

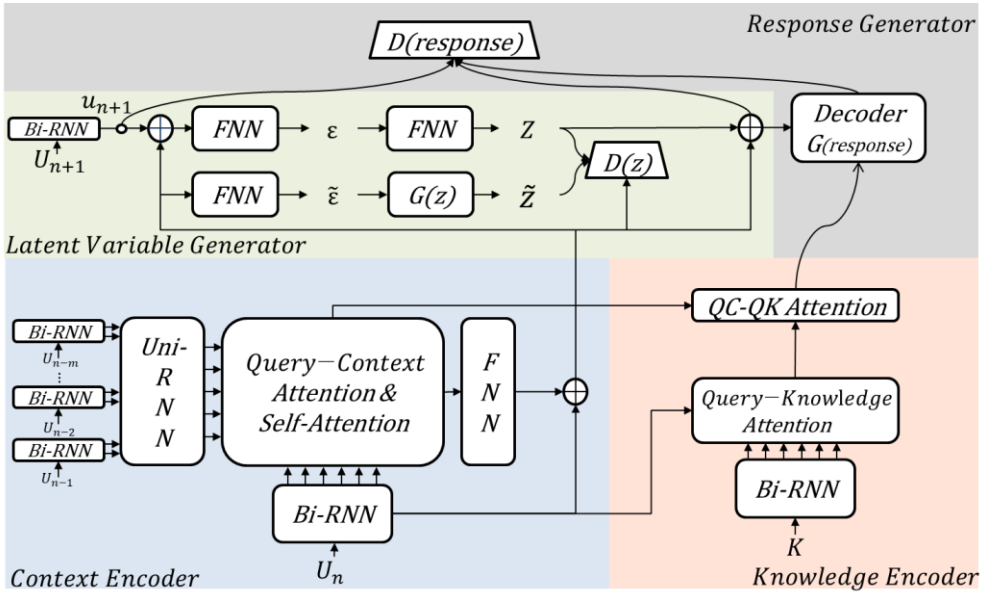

Figure 1. Overall architecture of the proposed knowledge-grounded chatbot. RNN: Recurrent neural network; FNN: Fully connected neural network.

The context encoder takes m previous utterances $U_{n-m}, \ldots, U_{n-2}, U_{n-1}$ (i.e., a dialogue context) and a current utterance U_n (i.e., a user's query) as inputs. Then, it calculates the degrees of association between the previous utterances and the current utterance based on a scaled dot product attention mechanism [22]. Finally, it generates a context vector. The knowledge encoder takes knowledge K (i.e., a document containing evidence to generate a response) and the current utterance U_n as inputs. Then, it calculates the degrees of association between knowledge and the current utterance based on a scaled dot product attention mechanism [22], and it generates a knowledge vector. During the training,

the latent variable generator takes the context vector and the next utterance U_{n+1} (i.e., a chatbot's response) as inputs. Then, it creates a context vector similar to the RNN-encoded next-utterance vector using an adversarial learning method [23]. Next, it decodes an encoded next-utterance vector using an autoencoder learning method. Finally, the trained latent variable vector $\widetilde{z}$ is used at the inference time. To generate various responses, the response generator creates an encoded response vector (i.e., a gold response vector) similar to a response vector decoded by the RNN (i.e., a generated response vector) using an adversarial learning scheme.

3.1. Context Encoder

The context encoder calculates the degrees of association between a dialogue context $U_{n-m}, \ldots, U_{n-2}, U_{n-1}$ and a current utterance U_n. The current and previous utterances in the dialogue context are individually encoded with bidirectional gated recurrent units (Bi-GRU; a kind of bidirectional RNN) [24]:

$$\begin{aligned} \overrightarrow{h_i} &= GRU(x_i, \overrightarrow{h_{i-1}}), \\ \overleftarrow{h_i} &= GRU(x_i, \overleftarrow{h_{i+1}}), \\ h_i &= [\overrightarrow{h_i}; \overleftarrow{h_i}], \end{aligned} \tag{1}$$

where x_i denotes the i-th word vector in an utterance, $\overrightarrow{h_i}$ and $\overleftarrow{h_i}$ denote the i-th word vectors encoded by the forward and the backward state of Bi-GRU, and $[\ ;\ ]$ denotes concatenation. Then, by analogy, each utterance in a dialogue context is encoded by unidirectional gated recurrent units (Uni-GRU; a kind of unidirectional RNN) [25]:

$$\begin{aligned} u_j &= [h_1; h_{last}], \\ c_j &= GRU(u_j, c_{j-1}), \end{aligned} \tag{2}$$

where u_j denotes the concatenation of the first word vector and the last word vector in the j-th utterance encoded by Bi-GRU, and c_j denotes the j-th utterance vector encoded by Uni-GRU. After encoding the dialogue context and the current utterance, the context encoder computes attention scores A^c between the current utterance and each utterance in the dialogue context by using scaled dot products [22]:

$$\begin{aligned} A^c &= g(softmax\left(\frac{W_c * C + \left(W_{h_1} * H\right)^T}{\sqrt{d}}\right) * H + C), \\ g(x) &= x * \sigma(W_g * x), \end{aligned} \tag{3}$$

where C denotes a matrix of an encoded dialogue context, $[c_1, c_2, \ldots, c_m]$, and H denotes a matrix of an encoded current utterance, $[h_1, h_2, \ldots, h_{last}]$. Then, W_c, W_{h_1}, and W_g are weights; d is a normalizing factor, which is set to 300; and $g(x)$ is a sigmoid gate function. After passing through the self-attention layer between A^c and A^c [20], the final attention matrix A^{qc}(called QC-Attention) is calculated as follows:

$$A^{qc} = A^c * softmax\left(\sum_{i=1}^{N} tanh(A_i^c * h_i)\right) \tag{4}$$

3.2. Knowledge Encoder

The knowledge encoder calculates the degrees of association between a current utterance U_n and given knowledge K. We assume that knowledge is represented as unstructured text (a sequence of

words). Knowledge K is encoded with Bi-GRU as in the context encoder. After encoding knowledge and the current utterance, the knowledge encoder computes attention scores A^k as in the context encoder:

$$\begin{aligned} A^k &= g(softmax\left(\frac{W_k * K + \left(W_{h_2} * H\right)^T}{\sqrt{d}}\right) * H + K), \\ A^{qk} &= A^k * softmax\left(\sum_{i=1}^{N} tanh(A_i^k * h_i)\right), \end{aligned} \tag{5}$$

where H denotes a matrix of an encoded current utterance, $[h_1, h_2, \ldots, h_k]$, and K denotes a matrix of encoded knowledge, $[k_1, k_2, \ldots, k_{last}]$, in which k_j is a concatenation of the i-th forward state and the i-th backward state of Bi-GRU. Then, W_c, W_{h_2}, and W_g are weights; d is a normalizing factor, which is set to 300; and $g(x)$ is a sigmoid gate function. Then, the final attention matrix A^{qk} (called QK-Attention) is calculated. Finally, the knowledge encoder calculates A^{ck} (called QC-QK-Attention) that represents the degrees of association between the dialogue context and knowledge:

$$A^{ck} = g(softmax\left(\frac{W_{A^k} * A^k + \left(W_{A^c} * A^c\right)^T}{\sqrt{d}}\right) * A^k + A^c). \tag{6}$$

3.3. Latent Variable Generator

The latent variable generator is a module that is responsible for generating latent variables that help the response decoder generate various responses. To generate latent variables, we adopted an autoencoder model using Wasserstein distance [4,11,23]. The conventional VAE dialogue model assumes that latent variables follow a simple prior distribution. However, we train latent variables using a Wasserstein generative adversarial network (WGAN) because responses in the real world follow very complex distributions. Formally, we model distributions z and $\widetilde{z}$ in latent spaces as follows:

$$\begin{aligned} z &= G_\theta(\epsilon),\ \epsilon \sim N\left(\epsilon; \mu, \sigma^2 I\right),\ \begin{bmatrix} \mu \\ log\sigma^2 \end{bmatrix} = W * f_\theta\left(\begin{bmatrix} A^{qc} \\ h_i \\ u_{i+1} \end{bmatrix}\right) + b, \\ \widetilde{z} &= Q_\varnothing(\widetilde{\epsilon}),\ \widetilde{\epsilon} \sim N\left(\widetilde{\epsilon}; \widetilde{\mu}, \widetilde{\sigma}^2 I\right),\ \begin{bmatrix} \widetilde{\mu} \\ log\widetilde{\sigma}^2 \end{bmatrix} = \widetilde{W} * g_\varnothing\left(\begin{bmatrix} A^{qc} \\ h_i \end{bmatrix}\right) + \widetilde{b}, \end{aligned} \tag{7}$$

where $f_\theta(\cdot)$ and $g_\varnothing(\cdot)$ are feed-forward neural networks for generating latent variables z and $\widetilde{z}$, respectively. The difference between the two neural networks is whether u_{i+1}, the next utterance (i.e., a chatbot's response) encoded by Bi-GRU, is used as an input or not. In Equation (7), u_{i+1} is represented by the concatenation of the first word vector and the last word vector encoded by Bi-GRU. Our goal is to minimize the divergence between z and $\widetilde{z}$. Therefore, we made the two distributions similar, according to the well-known GAN training scheme [26]:

$$\min_{\theta,\varnothing,\varphi} E\left[D_\varphi(\widetilde{z})\right] - E\left[D_\varphi(z)\right] + \lambda * E\left[\left(\|\nabla_{\widetilde{z}} D_\varphi(\widetilde{z})\|_2 - 1\right)^2\right], \tag{8}$$

where $D(\cdot)$ is a discriminator based on a feed-forward neural network. The discriminator plays the role of distinguishing between z and $\widetilde{z}$.

3.4. Response Generator

First, the response generator generates a response by using a concatenation of an encoded current utterance vector u_n, QC-Attention A^{qc}, QC-QK-Attention A^{ck}, and a generated latent variable (i.e., z for training and $\widetilde{z}$ for inferencing) as the initial state of a decoder. Then, it makes the generated response vector similar to u_{n+1}, which is a gold response vector encoded by Bi-GRU using the Wasserstein autoencoder (WAE) process based on WGAN [23].

3.5. Implementation and Training

We implemented the proposed model by using TensorFlow 1.14.1 [27]. Bi-GRUs have 300 hidden units in each direction, and Uni-GRUs have 600 hidden units. The dimensions of QC-Attention, QK-Attention, and QC-QK-Attention are 600. The discriminators of the latent variable and response generators are three-layer FNNs with 100 hidden units and rectified linear unit activation [28]. The vocabulary size, the word-embedding size, and the dialogue context size was set to 47,186, 300, and 3, respectively. All word-embedding vectors were initialized to random values. Responses were generated by using a greedy decoding algorithm. The model training was performed through three steps [2]. In the first step, we trained the WGAN in the latent variable generator by using an adversarial learning method. In the second step, we trained the entire model, except the WAE in the response generator. At last, we trained the WAE in the response generator by using an adversarial learning method. When we train the discriminant models [26], we used the gradient penalty which was set to 10. Then, we used a cross-entropy function as a cost function to maximize the log-probability. In the inference step, we used $\widetilde{z}$ as a latent variable.

4. Evaluation

4.1. Datasets and Experimental Settings

We evaluated our model on a Wizard-of-Wikipedia dataset [1]. The dataset is used for three tasks: Knowledge prediction (i.e., selecting documents containing proper knowledge from a document collection), response generation (i.e., generating responses using the given knowledge), and an end-to-end task (i.e., both knowledge prediction and response generation). In this study, we focused on response generation. Owing to hardware limitations, we refined the Wizard-of-Wikipedia dataset to use two previous utterances as the dialogue context. Our refined dataset comprises 83,247 utterances for training, 4444 utterances for verification, and 4356 utterances for testing.

We used Bilingual evaluation understudy (BLEU) [29,30], perplexity (PPL) [1,31], and bag-of-words (BOW) embedding [32] as performance measures. BLEU measures a ratio of overlapped words between generated responses and gold responses, as shown in the following equation.

$$BLEU = \min(1, \frac{length\ of\ a\ generated\ sentence}{length\ of\ a\ gold\ sentence})(\prod_{i=1}^{n} precision_i)^{\frac{1}{n}}, \quad (9)$$

where n is the maximum length of n-grams, which is commonly set to 4, and $precision_i$ is a word i-gram precision (i.e., the number of correct word i-grams divided by the number of word i-grams in a generated sentence). The precision of BLEU is the average score of BLEUs for 10 generated sentences (for these experiments, the decoder returned 10 candidate sentences per query by a beam search algorithm) per query. The recall of BLEU is the maximum score among BLEUs for 10 generated sentences per query. As many studies used unigram-F1, we also used word an unigram F1, called F1-score [1,31]. PPL is a measure for evaluating language models, and it is commonly used in traditional generative chatbot models that mainly use an RNN decoder. The BOW embedding metric is the cosine similarity of BOW embedding between generated and gold responses. It consists of three metrics: Greedy [33], average [34], and extrema [35]. In our test, we report the maximum BOW embedding score among the 10 sampled responses.

4.2. Experimental Results

In our first experiment, we evaluated the effectiveness of the knowledge encoder at the architecture level. The results are summarized in Table 2.

Table 2. Performance comparison depending on changes in the architecture (The bolds indicate the highest scores in each evaluation measure).

Model	F1-Score	PPL	BOW Embedding		
			A	G	E
CE	0.210	**75.71**	0.259	0.356	0.162
CE+KE	**0.267**	81.40	**0.320**	**0.411**	**0.210**

CE: Context encoder; KE: Knowledge encoder; PPL: Perplexity; A: Average; E: Extrema; G: Greedy; BOW: Bag-of-words.

In Table 2, CE is a model that only uses QK-Attention vector, and CE + KE uses QC-QK-Attention. As shown in Table 2, CE+KE exhibit better performances for all measures except PPL. PPL is the average number of options when a language model is generating words. With a lower PPL, the model has fewer options for generating words, such as general words ("I know"). Hence, PPL is not important for chatbots that need to generate various responses. This means that QC-QK-Attention improves the quality of responses.

In the second experiment, we compared the performance of the proposed model with those of state-of-the-art models. For a fair comparison, we downloaded the code provided by the authors of each model from GitHub (https://github.com/lizekang/ITDD) [31]. Then, we trained and tested the models by using the refined Wizard-of-Wikipedia dataset. Table 3 compares the performance of these models.

Table 3. Performance comparison between the proposed and previous models. (The bolds indicate the highest scores in each evaluation measure).

Model	F1-Score	PPL	BOW Embedding		
			A	G	E
Dinan-End2End [1]	0.172	70.03	0.127	0.248	0.118
Dinan-TwoStage [1]	0.150	50.55	0.127	0.235	0.104
ITDD [31]	0.193	**11.16**	0.250	0.332	0.136
Proposed Model	**0.267**	81.40	**0.320**	**0.411**	**0.210**

In Table 3, Dinan-End2End and Dinan-TwoStage are the most well-known knowledge-grounded chatbot models proposed by Dinan et al. who deployed the Wizard-of-Wikipedia dataset [1]. Dinan-End2End performs the end-to-end task, and Dinan-TwoStage separately performs knowledge prediction and response generation. ITDD is a state-of-the-art model based on Transformer, and it generates a response using a two-pass decoder. We replaced the document with the context. As shown in Table 3, the proposed model exhibits better performances for all measures except PPL. The compared models show lower PPL because they generate more general words than the proposed model does, as shown in Table 4. Table 4 shows an example of the response of each model. Dinan-End2End, Dinan-TwoStage, and ITDD models generated answers such as "I know" or "I'm not sure." These general responses help lower PPL. However, the proposed model generated a response that mentioned the names of the band members, reflecting the acquired knowledge.

Table 4. Comparison of proposed and previous models.

Knowledge	The first stable line-up consisted of Brian Jones (guitar, harmonica), Mick Jagger (lead vocals), Keith Richards (guitar, backing vocals), Bill Wyman (bass), Charlie Watts (drums), and Ian Stewart (piano).
Context	I used to listen to the Rolling Stones a lot when I was a child.
	Me too. I can't believe they were formed in London as far back as 1962! What's your favorite song?
	I can't even remember, to be honest! Do you know who the band members were?
Gold Response	Mick (of course), Brian Jones, Keith Richards, Bill Wyman, and I don't remember who else.
Dinan-End2End	I know, they were formed in London in 1962.
Dinan-TwoStage	I'm not sure, but I know that the band was formed in 1981 by drummer Lars Ulrich and vocalist James Hetfield.
ITDD	I'm not sure, but I know they were formed in 1962.
Proposed Model	Mick Jagger, Keith Richards, Bill Wyman, Charlie Watts, and Ian Stewart.

5. Conclusions

We proposed a knowledge-grounded multi-turn chatbot model to effectively reflect newly-acquired external knowledge. To generate responses in the context of conversation history, it used an attention mechanism between a query and a context in the query-encoding step. To generate responses that reflect external knowledge, it used the query-knowledge mechanism in the knowledge encoding step. Furthermore, the model effectively mixed the two kinds of attention to consider the degree of association between the dialogue context and the given knowledge. In experiments with the Wizard-of-Wikipedia dataset, the proposed model showed better performances than the previous state-of-the-art models and generated responses more effectively, reflecting external knowledge. However, the proposed model has a limitation; that is, proper external knowledge should be given in advance. To overcome this, we will study an end-to-end knowledge-grounded chatbot model that searches external documents containing proper knowledge, summarizes the retrieved documents, and extracts proper knowledge from the summarized documents. As the future work, we will study a method to generalize an input utterance to a shallow semantic form that is a set of keywords, tense information, and modality information (e.g., "I'll be able to go there." → "[[I, go, there], future, possibility]"). Then, we will study a method to use the semantic form as an input of a Seq2Seq model.

Author Contributions: Conceptualization, H.K.; methodology, H.K.; software, S.K.; validation, S.K.; formal analysis, H.K.; investigation, H.K.; resources, S.K.; data curation, S.K.; writing—original draft preparation, S.K.; writing—review and editing, H.K.; visualization, H.K.; supervision, H.K. and O.-W.K.; project administration, H.K. and O.-W.K.; funding acquisition, H.K. and O.-W.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Institute for Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (2019-0-00004, Development of semi-supervised learning language intelligence technology and Korean tutoring service for foreigners). This research was also supported by the MSIT (Ministry of Science, ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2020-2016-0-00465) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation).

Acknowledgments: We especially thank the members of the NLP laboratory at Konkuk University for technical support.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dinan, E.; Roller, S.; Shuster, K.; Fan, A.; Auli, M.; Weston, J. Wizard of Wikipedia: Knowledge-Powered Conversational Agents. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
2. Qiu, M.; Li, F.-L.; Wang, S.; Gao, X.; Chen, Y.; Zhao, W.; Chen, H.; Huang, J.; Chu, W. Alime chat: A sequence to sequence and rerank based chatbot engine. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 498–503.
3. Yan, Z.; Duan, N.; Bao, J.; Chen, P.; Zhou, M.; Li, Z.; Zhou, J. Docchat: An information retrieval approach for chatbot engines using unstructured documents. In Proceedings of the Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, 7–12 August 2016; pp. 516–525.
4. Kim, J.; Oh, S.; Kwon, O.-W.; Kim, H. Multi-Turn Chatbot Based on Query-Context Attentions and Dual Wasserstein Generative Adversarial Networks. *Appl. Sci.* **2019**, *9*, 3908. [CrossRef]
5. Vinyals, O.; Le, Q.V. A Neural Conversational Model. In Proceedings of the ICML Deep Learning Workshop, Lille, France, 10–11 July 2015.
6. Li, J.; Galley, M.; Brockett, C.; Gao, J.; Dolan, B. A Diversity-Promoting Objective Function for Neural Conversation Models. In Proceedings of the Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA, USA, 12–17 June 2016; Association for Computational Linguistics: San Diego, CA, USA, 2016; pp. 110–119.
7. Sato, S.; Yoshinaga, N.; Toyoda, M.; Kitsuregawa, M. Modeling Situations in Neural Chat Bots. In Proceedings of the Proceedings of ACL 2017, Student Research Workshop, Vancouver, BC, Canada, 30 July–4 August 2017; Association for Computational Linguistics: Vancouver, BC, Canada, 2017; pp. 120–127.
8. Zhao, T.; Zhao, R.; Eskenazi, M. Learning Discourse-level Diversity for Neural Dialog Models using Conditional Variational Autoencoders. In Proceedings of the Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 654–664.
9. Shen, X.; Su, H.; Niu, S.; Demberg, V. Improving variational encoder-decoders in dialogue generation. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
10. Goyal, P.; Hu, Z.; Liang, X.; Wang, C.; Xing, E.P.; Mellon, C. Nonparametric Variational Auto-Encoders for Hierarchical Representation Learning. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 5104–5112.
11. Gu, X.; Cho, K.; Ha, J.W.; Kim, S. Dialogwae: Multimodal response generation with conditional Wasserstein auto-encoder. In Proceedings of the 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, 6–9 May 2019.
12. Li, J.; Monroe, W.; Shi, T.; Jean, S.; Ritter, A.; Jurafsky, D. Adversarial Learning for Neural Dialogue Generation. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, 9–11 September 2017; pp. 2157–2169.
13. Yu, L.; Zhang, W.; Wang, J.; Yu, Y. Seqgan: Sequence generative adversarial nets with policy gradient. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
14. Madotto, A.; Wu, C.-S.; Fung, P. Mem2Seq: Effectively Incorporating Knowledge Bases into End-to-End Task-Oriented Dialog Systems. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, 15–20 July 2018; pp. 1468–1478.
15. Liu, S.; Chen, H.; Ren, Z.; Feng, Y.; Liu, Q.; Yin, D. Knowledge Diffusion for Neural Dialogue Generation. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Melbourne, Australia, 15–20 July 2018; pp. 1489–1498.
16. Kim, S.; Kim, H.; Kwon, O.-W.; Kim, Y.-G. Improving Response Quality in a Knowledge-Grounded Chat System Based on a Sequence-to-Sequence Neural Network. In Proceedings of the 2019 IEEE International Conference on Big Data and Smart Computing (BigComp), Kyoto, Japan, 27 February–2 March 2019; pp. 1–4.

17. He, S.; Liu, C.; Liu, K.; Zhao, J. Generating Natural Answers by Incorporating Copying and Retrieving Mechanisms in Sequence-to-Sequence Learning. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BC, Canada, 30 July–4 August 2017; pp. 199–208.
18. Parthasarathi, P.; Pineau, J. Extending Neural Generative Conversational Model using External Knowledge Sources. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 690–695.
19. Yavuz, S.; Rastogi, A.; Chao, G.-L.; Hakkani-Tur, D. DeepCopy: Grounded Response Generation with Hierarchical Pointer Networks. In Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue, Stockholm, Sweden, 11–13 September 2019; pp. 122–132.
20. Ghazvininejad, M.; Brockett, C.; Chang, M.-W.; Dolan, B.; Gao, J.; Yih, W.; Galley, M. A Knowledge-Grounded Neural Conversation Model. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
21. Ye, H.-T.; Lo, K.-L.; Su, S.-Y.; Chen, Y.-N. Knowledge-grounded response generation with deep attentional latent-variable model. *Comput. Speech Lang.* **2020**, *63*, 101069. [CrossRef]
22. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is All You Need. In *Advances in Neural Information Processing Systems 30*; Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R., Eds.; Curran Associates, Inc.: Dutchess County, NY, USA, 2017; pp. 5998–6008.
23. Arjovsky, M.; Chintala, S.; Bottou, L. Wasserstein generative adversarial networks. In Proceedings of the 34th International Conference on Machine Learning-Volume 70, Sydney, Australia, 6–11 August 2017; pp. 214–223.
24. Schuster, M.; Paliwal, K. Bidirectional recurrent neural networks. *Signal Process. IEEE Trans.* **1997**, *45*, 2673–2681. [CrossRef]
25. Cho, K.; van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. 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), Doha, Qatar, 25–29 October 2014; pp. 1724–1734.
26. Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; Courville, A.C. Improved training of wasserstein gans. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5767–5777.
27. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M. TensorFlow: A system for large-scale machine learning. In Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation, Savannah, GA, USA, 2–4 November 2016; pp. 265–283.
28. Nair, V.; Hinton, G.E. Rectified linear units improve restricted boltzmann machines. In Proceedings of the 27th International Conference on Machine Learning (ICML-10), Haifa, Israel, 21–24 June 2010; pp. 807–814.
29. Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.-J. BLEU: A Method for Automatic Evaluation of Machine Translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, PA, USA, 7–12 July 2002; pp. 311–318.
30. Chen, B.; Cherry, C. A Systematic Comparison of Smoothing Techniques for Sentence-Level BLEU. In Proceedings of the Ninth Workshop on Statistical Machine Translation, Baltimore, MD, USA, 26–27 June 2014; pp. 362–367.
31. Li, Z.; Niu, C.; Meng, F.; Feng, Y.; Li, Q.; Zhou, J. Incremental Transformer with Deliberation Decoder for Document Grounded Conversations. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 12–21.
32. Liu, C.W.; Lowe, R.; Serban, I.V.; Noseworthy, M.; Charlin, L.; Pineau, J. 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, Austin, TX, USA, 1–5 November 2016; pp. 2122–2132.
33. Rus, V.; Lintean, M. A Comparison of Greedy and Optimal Assessment of Natural Language Student Input Using Word-to-Word Similarity Metrics. In Proceedings of the Seventh Workshop on Building Educational Applications Using NLP, Montréal, QC, Canada, 7 June 2012; pp. 157–162.

34. Mitchell, J.; Lapata, M. Vector-based Models of Semantic Composition. In Proceedings of the ACL-08: HLT, Columbus, OH, USA, 15–20 June 2008; pp. 236–244.
35. Forgues, G.; Pineau, J.; Larchevêque, J.-M.; Tremblay, R. Bootstrapping dialog systems with word embeddings. In Proceedings of the Nips, Modern Machine Learning and Natural Language ProcessingWorkshop, Geneva, Switzerland, 12 December 2014.