JOINT LEARNING FOR ADDRESSEE SELECTION AND RESPONSE GENERATION IN MULTI-PARTY CONVERSATION

Qi Song *Sheng Li* *Ping Wei* *Ge Luo* *Xinpeng Zhang*⋆ *Zhenxing Qian*⋆

School of Computer Science, Fudan University, 200433 Shanghai, China

ABSTRACT

A large number of multi-party conversation scenarios exist in social networks, which have been seldom studied in the field of human-machine conversation. In this paper, we study a novel task of joint learning for addressee selection and response generation in multi-party conversations. Systems are expected to select whom they address and generate the corresponding response. To solve it, we propose an end-to-end addressee selection and response generation (ASRG) model, containing an addressee selection module and a response generation module. In the selection module, we develop an addressee prediction attention scheme to obtain a unique context vector for each candidate, thereby calculating the probability of the candidate more accurately. In the generation module, we propose a Focus Transformer to generate responses. These two modules are jointly learnt to fully explore the correlations between addressee and response. Experimental results show ASRG remarkably outperforms baselines and generates relevant content for different addressees.

Index Terms— Multi-party Conversation, Virtual Users, Natural Language Processing, Attention Mechanism, Text Generation

1. INTRODUCTION

Human-machine conversation technology is known as the pearl in the crown of artificial intelligence, and generating virtual users that can pass the Turing test [1] is the unremitting pursuit of NLP [2] field. Researches in this area mostly focus on the dialogue with two-interlocutor [3, 4, 5]. However, social networks also have a large number of multi-party conversation scenarios that have multiple interlocutors [6], such as Telegram groups and Twitter comments. Virtual users in these scenarios can act as response assistants to enhance the activeness of multi-party conversations.

Multi-party conversations have many distinctive characteristics compared to two-party conversations. As shown in Table 1, firstly, multiple interlocutors participate in the conversation, and the interaction between them affects each other. Also, their roles (speaker, addressee or others) may change across different dialog turns [7]. Secondly, when the virtual user, namely the system speaks, it needs to select a suitable addressee among multiple interlocutors, and then generate quality responses for the addressee. Finally, usually more dialogue contexts exist in the multi-party conversation, and multi-topic are interspersed among them. When generating responses, it's unreasonable to treat each utterance equally or only consider the utterances related to the addressee. Instead, different weights should be assigned to each utterance according to the addressee, so that the generated contents focus on the selected addressee.

Table 1. An example for multi-party conversation.

Speaker	Addressee	Utterance
user1	-	My computer can't work ...
system	-	Try reinstalling the os
user2	-	View error messages, and
user1	system	How to do it ?
system	[To Whom?] 1. user1 2. user2	[Speak What?] 1. See this URL: http//xxx 2. Can't turn on ...

Traditional two-party conversations methods cannot be directly applied to multi-party conversations. Because ignoring the above characteristics may lead to misunderstanding or even contradictory results. In literature, a few attempts have been devoted to study multi-party conversations. Dynamic-RNN [6] and SI-RNN [8] have studied the addressee selection problem, ICRED [7] and GSN [9] have worked on response generation. However, considering addressee selection and response generation independently is inappropriate. In fact, the addressee affects the generated contents, these two parts are interrelated and should be treated as a whole.

In this paper, we consider the addressee selection and response generation as a joint learning task. To tackle it, we propose the addressee selection and response generation (ASRG) model, including an addressee selection module and a response generation module. An embedding matrix for modeling interlocutors is dynamically updated in the former module, and a prediction attention that calculate the probability of each candidate addressee more accurately is also introduced into this module. Moreover, we propose a novel Transformer variant called Focus Transformer. The focus attention mechanism is introduced in the Decoder of proposed Transformer, for dynamically assigning weights to the utterances according to the addressee. Thus, the generated responses are related to the addressee.

In brief, the main contributions of this paper are as follows:

- We're the first to consider addressee selection and response generation in multi-party conversation as a joint learning task.
- We propose an end-to-end ASRG model. Therein, a prediction attention is introduced for addressee selection and a novel Focus Transformer is designed for response generation.
- We develop a new joint loss to integrate two modules for addressee selection and response generation.

2. RELATED WORK

2.1. Two-party Conversation

Many Human-machine conversation researches have been published. HRED [10] utilizes hierarchical RNN structure to model

⋆Xinpeng Zhang and Zhenxing Qian are the corresponding authors.

This work was supported in part by the National Natural Science Foundation of China under Grants U1936214, U20B2051, 62072114, U20A20178, in part by the Project of Shanghai Science and Technology Commission 21010500200.

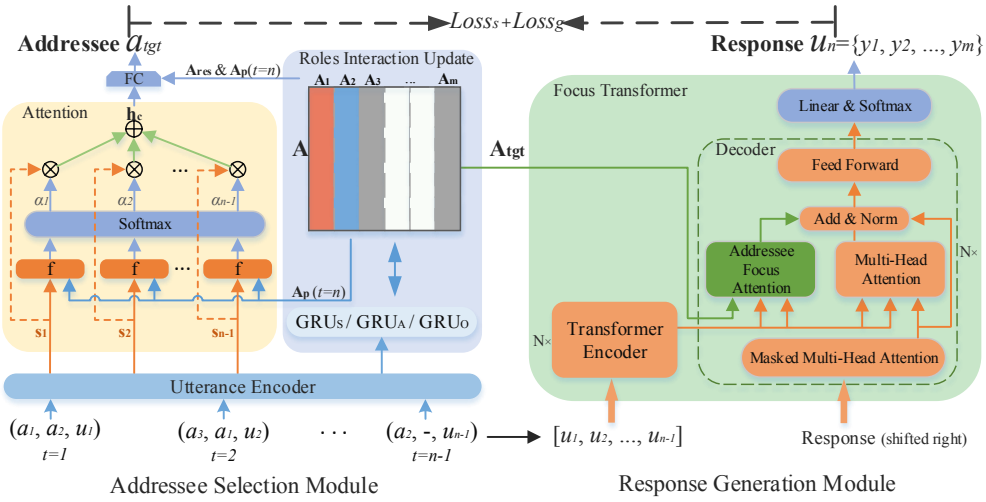

Fig. 1. **The architecture of the Addressee Selection And Response Generation Model (See Section 3).** The selection module on the left predicts the target addressee a_{tgt}, which mainly includes utterance encoder, roles interaction update and addressee prediction attention. Focus Transformer with addressee focus attention on the right, generates the response u_n. Furthermore, the two parts are joint learning.

dialogue context to achieve multi-round conversations. AliMe Chat [11] is an open domain question answering robot, which integrates Seq2Seq [12] generation structure and information retrieval. In recent years, Transformer [13] built entirely based on attention mechanism and its derivatives [14, 15] have been widely used in NLP downstream tasks such as text generation [16] and dialogue system [17, 18]. However, these studies all focus on two-party conversations and are not suitable for multi-party conversations.

2.2. Multi-party Conversation

Researches in multi-party conversations are few. Ouchi et al. [6] first proposes the task of addressee and response selection, and designs a structure call Dynamic-RNN to fulfil the tasks. On this basis, SI-RNN [8] is raised to process interlocutors in a role-sensitive manner, with which the performance of addressee selection is improved. But both of them are retrieval-based approaches and cannot generate diverse responses. Some researches assume that the addressee is known in response generation. Among them, GSN [9] utilizes a graph structure network to represent interlocutors' interaction and generates responses. ICRED [7] models multi-interlocutor according to different roles, and then integrates the addressee information into the response generation.

These studies above take the addressee selection and response generation as two independent task. However, these two parts should be considered jointly, because it's an overall interconnected process, and the generated contents are affected by the selected addressee.

3. PROPOSED METHOD

Given the responding speaker a_{res} and dialog context C, the task is to select a target addressee a_{tgt} and generate the corresponding response u_n at time step n. Here, C is a list ordered by time step: $C = [(a^t_{spk}, a^t_{adr}, u_t)]^{n-1}_{t=1}$, where each triad means speaker a^t_{spk} says utterance u_t to addressee a^t_{adr} at time step t and other interlocutors are defined as others, $n-1$ is the number of dialog turns.

The overview of the proposed ASRG model is shown in Figure 1. The details are as follows.

3.1. Addressee Selection Module

Selection module is to choose the target addressee a_{tgt} through the utterance encoder, roles interaction update module and addressee prediction attention mechanism. Among them, addressee prediction attention is the key to improving the accuracy of prediction.

3.1.1. *Utterance Encoder*

Utterance encoder transforms utterances of dialogue context into distributional representations. For time step t, a utterance with N words $u_t = (w_1, w_2, ..., w_N)$, we leverage the Bi-directional Gated Recurrent Units [19] to encode it.

$$\begin{aligned} \overrightarrow{\mathbf{h_i}} &= \mathrm{GRU}(\overrightarrow{\mathbf{h_{i-1}}}, \mathbf{e_i}) \\ \overleftarrow{\mathbf{h_i}} &= \mathrm{GRU}(\overleftarrow{\mathbf{h_{i+1}}}, \mathbf{e_i}) \end{aligned} \tag{1}$$

where $\mathbf{e_i}$ is embedding of the word w_i. The utterance representation vector $\mathbf{s_t} = [\overrightarrow{\mathbf{h_N}}; \overleftarrow{\mathbf{h_N}}]$, where the final forward hidden state and the backward hidden state are concatenated. Afterwards, the utterance representation could be sent to the roles interaction update module and addressee prediction attention mechanism.

3.1.2. *Roles Interaction Update*

Roles interaction update is leveraged to obtain the information of each interlocutor. Similar to the SI-RNN [8] and ICRED [7], we also use the role-sensitive update method.

An interlocutor embedding matrix $\mathbf{A}$ is used to record all interlocutors' representation, which is initiated with a zero matrix. The i-th column of the matrix, $\mathbf{A_i}$ means the representation of interlocutor a_i. At time step t, each interlocutor will update the representation according to its role (speaker, addressee or others). Embeddings for speaker, addressee and others are updated by following role-differentiated GRUs: $\mathrm{GRU_S}$, $\mathrm{GRU_A}$ and $\mathrm{GRU_O}$, respectively.

$$\mathbf{A^t_{spk}} \leftarrow \mathrm{GRU_S}(\mathbf{A^{t-1}_{spk}}, \mathbf{A^{t-1}_{adr}}, \mathbf{s_t}) \tag{2}$$

$$\mathbf{A^t_{adr}} \leftarrow \mathrm{GRU_A}(\mathbf{A^{t-1}_{adr}}, \mathbf{A^{t-1}_{spk}}, \mathbf{s_t}) \tag{3}$$

$$\mathbf{A^t_{otr}} \leftarrow \mathrm{GRU_O}(\mathbf{A^{t-1}_{otr}}, \mathbf{s_t}) \tag{4}$$

where $\mathbf{A^t_{spk}}$, $\mathbf{A^t_{adr}}$, $\mathbf{A^t_{otr}}$ are the embedding representations of the speaker, addressee and others at time step t, respectively. After $n-1$ steps, the update of $\mathbf{A}$ is complete. The final matrix will be used in the addressee prediction attention mechanism at time step n.

3.1.3. Addressee Prediction Attention

Interlocutor embedding matrix $\mathbf{A}$ captures interlocutors' information. Previous researches directly apply a max pooling to $\mathbf{A}$ and combine the information of responding speaker for addressee selection. However, the dialogue context information is ignored. In fact, the context information is also important for addressee selection, because multi-party conversations include not only multiple interlocutors, but also the utterances between interlocutors. And the context information that everyone pays attention to is different.

Therefore, we design the addressee prediction attention mechanism. It calculates the dialogue context vector $\mathbf{h_c}$ separately for each candidate addressee a_p, where a_p is in $\mathcal{A}(C)$ and $\mathcal{A}(C)$ is the set of speakers appearing in C. For each candidate a_p, the attention mechanism assigns different weights to each utterance in the context according to its embedding representation.

$$\mathbf{h_c} = \sum_{i=1}^{n-1} \alpha_i \mathbf{s_i} \tag{5}$$

$$\alpha_i = Softmax(f(\mathbf{A_p}, \mathbf{s_i})) \tag{6}$$

where f is a feed-forward neural network, and $\mathbf{s_i}$ is the representation of utterance u_i, and $\mathbf{A_p}$ is the representation vector corresponding to a_p in matrix $\mathbf{A}$. The calculated context vector $\mathbf{h_c}$ extracts the dialogue information that the candidate a_p pays attention to.

On this basis, each candidate leverages the corresponding embedding representation $\mathbf{A_p}$, its unique context vector $\mathbf{h_c}$ and the responding speaker's representation $\mathbf{A_{res}}$ for probability calculation. The probability calculated in combination with key context information is more accurate, so that the final selection result is more exact.

$$\mathbb{P}(a_p|C) = \sigma([\mathbf{A_p}; \mathbf{h_c}]^{\mathrm{T}} \mathbf{W_a} \mathbf{A_{res}}) \tag{7}$$

where σ is the sigmoid function, $\mathbf{W_a}$ is a parameter matrix. Finally, among all the candidates, the one with the highest probability is chosen as the target addressee a_{tgt}. The utilization of this attention mechanism greatly improves the accuracy of addressee prediction.

3.2. Response Generation Module

Unlike two-party conversations, multi-party conversations often involve more dialogue context. Traditional RNNs are difficult to process long text sequences due to the long-term dependence [20], and its generated responses cannot make full use of dialogue information. Transformer [13], calculating in parallel, has stronger feature extraction capabilities, and is more suitable for generation task in multi-party conversations.

Generally, when the addressee is fixed, the speaker's response is usually to answer addressee's questions or expand addressee's viewpoints [7]. That's to say, the generated responses should be related to the addressee, and not aimlessly generalize. Unfortunately, raw Transformer doesn't possess such mechanism and can't be directly applied to response generation. Thus, we design a novel Transformer variant by introducing the addressee focus attention mechanism, called Focus Transformer.

Focus Transformer Encoder is consistent with that of the original Transformer. We input all utterances into Encoder to obtain the context vector $\mathbf{Enc_o}$ containing all the information of the dialogue. In the Decoder, we add an Addressee Focus Attention sublayer in each decoder block, which is parallel to the Multi-Head Encoder-Decoder Attention sublayer.

$$\begin{aligned} \mathbf{O_{focus}} &= \mathrm{Att}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) \\ &= \mathrm{Att}(\mathbf{A_{tgt}}, \mathbf{Enc_o}, \mathbf{Enc_o}) \end{aligned} \tag{8}$$

Embedding representation $\mathbf{A_{tgt}}$ is used as the query vector to obtain the key information that a_{tgt} attends to in the dialogue. The output $\mathbf{O_{focus}}$ is spliced with the output of Multi-Head Encoder-Decoder Attention and sent to the subsequent "Add & Norm" module together. Under the guidance of the addressee, the response $u_n = \{y_1, y_2, ...y_m\}$ containing m words is finally generated, which is closely related to the chosen addressee.

3.3. Loss Function

The training of the model is an end-to-end process. Our training loss $Loss$ consists of the selection loss $Loss_s$ and the generation loss $Loss_g$. Each part is calculated by cross-entropy. The goal of optimization is to minimize the loss through back-propagation.

$$Loss_s = -\sum_{j=1}^{|\mathcal{A}(C)|} a_{real} \log({a_{tgt}}^j) \tag{9}$$

$$Loss_g = -\sum_{j=1}^{m} \log P(y_j|y_{<j}, C, a_{tgt}) \tag{10}$$

$$Loss = \alpha Loss_s + \beta Loss_g \tag{11}$$

where a_{real} is the ground truth of target addressee a_{tgt} and $|\mathcal{A}(C)|$ is the number of candidates. α and β are two hyper-parameters.

4. EXPERIMENTS

4.1. Experimental Setup

We use Ubuntu IRC Logs [21], a large-scale Ubuntu IRC log dataset, for training and evaluation. After cleaning, finally 423.32K conversations (about 2.54M utterances) are extracted. Each conversation is composed of 6 triples, and the last one is the ground truth.

For training details, we set vocabulary size to 20k, word embedding to 300d and interlocutor vector in $\mathbf{A}$ to 512d. Also, selection module uses a 2-layer GRU with 256 hidden units and the settings of Focus Transformer are the same as raw Transformer. What needs to be pointed out is that we use the learning rate adjustment strategy of cosine annealing[22] in the training process. A simple learning rate decay strategy may make the model fall into a local optimum.

In our scheme, we utilize the prediction accuracy to evaluate the performance of addressee selection. Following previous work [7], we leverage metrics BLEU [23], ROUGE [24] and LENGTH (length of generated sentences) to evaluate the quality of generated text responses.

4.2. Results and Analysis

Addressee selection part is compared with the Static-RNN [6], Dynamic-RNN [6] and SI-RNN [8] models with the same function. The experimental results on the validation and test dataset are shown in Table 3. Our proposed addressee selection model outperforms the other works greatly. Our model has an accuracy improvement of more than 10% over the other works. We find that our model only achieves the close results like SI-RNN if the attention mechanism

Table 2. An example of different models' responses for the same dialogue context.

	Speaker	Addressee	Utterance
Dialogue Context	mgz	-	it actually builds or not on one of out supported platforms, is probably not a good idea
	finch	mgz	we decided errors with gccgo builds weren't going to block trunk, but maybe
	sinzui	finch	i think the issue is that ubuntu require this to work ans we need to prove it's a compiler
	finch	sinzui	well, it's a compiler issue. it can't find a package from the standard library.
	sinzui	mgz	the log shows gccgo did compile the ppc64el package on stilson
Human Response	finch	sinzui	remove and reinstall golang on that machine, maybe the installation got corrupted somehow
ICRED[7]			it is on gccgo compiler, error happen
Raw Transformer[13]	finch	sinzui	use this compiler to compile it, it can work.
Focus Transformer			Maybe check the compiler, make sure installed correctly

is canceled. This is because addressee prediction attention can help accurately calculate the probability of each candidate being selected, thereby the prediction accuracy is improved greatly.

Table 3. The prediction accuracy for addressee selection models.

Model	Validation	Test
Static-RNN[6]	61.26	60.39
Dynamic-RNN[6]	62.38	63.19
SI-RNN[8]	73.59	74.08
Our Model (without attention)	74.14	74.90
Our Model (with attention)	**84.42**	**84.65**

In terms of generating evaluation, the two-party conversations models Seq2Seq [25], HRED [10], raw Transformer [13] and response generation models ICRED [7] and GSN [9] for multi-party conversations are involved in the comparison. As shown in Table 4, our Focus Transformer has the best performance on all automated evaluation metrics. Meanwhile, our model can generate longer response sentences. Seq2Seq and HRED perform poorly because they cannot handle multi-interlocutor interaction. Other roles-based models such as ICRED, etc., cannot digest plenty of text sequences in multi-party conversations and perform poorly, due to the limitation of traditional RNNs.

Table 4. Experimental results of response generation models.

Model	BLEU-1	BLEU-2	ROUGE	LENGTH
Seq2Seq[25]	8.86	4.13	7.62	9.48
HRED[10]	9.21	4.60	7.67	10.20
ICRED[7]	10.63	4.96	8.73	11.34
GSN[9]	10.93	5.25	9.40	10.68
Raw Transformer[13]	10.47	5.01	8.89	11.96
Focus Transformer	**11.53**	**5.88**	**10.23**	**12.21**

4.3. Visualization of Response Relevance

Our Focus Transformer can generate responses related to the selected addressee a_{tgt}. However, it is difficult to quantitative evaluate the performance. So we use t-SNE [26] to conduct dimensionality reduction visualization experiments on the representation of a_{tgt} and the generation of different models. As shown in the Figure 2, in the low-dimensional space, we visually illustrate the correlation by observing the distance and distribution between them.

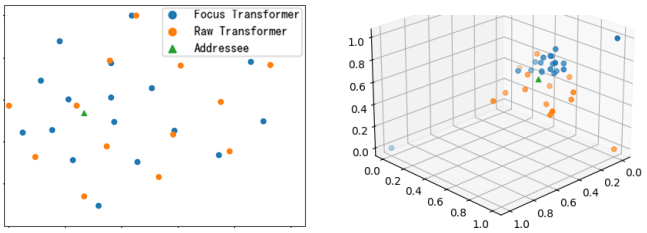

Fig. 2. Low-dimensional space visualization results.

In the two-dimensional figure, the green triangle represents the addressee, and the dots of different colors represent the responses generated by different models. Each dot here corresponds to one word. It can be seen that the blue dots representing Focus Transformer are closer to the green triangle as a whole, indicating that our generated response is more relevant to the chosen addressee. And the results prove that our proposed addressee focus attention is effective. The same conclusion can be obtained in three-dimensional space. And according to our observations, most of the samples in the test dataset meet the above conclusions.

4.4. Case Study

In this part, we show an example of different models' responses for the same dialogue context in Table 2. First of all, our model correctly chooses the target addressee *sinzui*. As conversation responses, our model provides meaningful suggestions for the issue discussed, and a useful conclusion of *possible compiler or installation problems* is drawn. Our generated response is closer to natural human response than the other works, and it is smoother, more appropriate and more relevant to the addressee.

5. CONCLUSION

In this paper, we formalize a joint learning task of addressee selection and response generation in multi-party conversations. We propose an novel end-to-end ASRG model, including the addressee selection module with prediction attention and the Focus Transformer generation module. It provides state-of-the-art performance, which can directly help to create virtual users in social networks.

Our future work is to tackle the problem of multiple addressees selection. In this paper, we assume that there is only one addressee for each conversation round. However, in actual multi-party conversations, sometimes one utterance may refer to multi-interlocutor. Therefore, multiple addressees selection will be the next challenge.

6. REFERENCES

[1] Alan M Turing, "Computing machinery and intelligence," in *Parsing the turing test*, pp. 23–65. Springer, 2009.

[2] Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang, "Pre-trained models for natural language processing: A survey," *Science China Technological Sciences*, pp. 1–26, 2020.

[3] Xipeng Qiu and Xuanjing Huang, "Convolutional neural tensor network architecture for community-based question answering," in *Twenty-Fourth international joint conference on artificial intelligence*, 2015.

[4] Libo Qin, Xiao Xu, Wanxiang Che, Yue Zhang, and Ting Liu, "Dynamic fusion network for multi-domain end-to-end task-oriented dialog," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 6344–6354.

[5] Anbang Xu, Zhe Liu, Yufan Guo, Vibha Sinha, and Rama Akkiraju, "A new chatbot for customer service on social media," in *Proceedings of the 2017 CHI conference on human factors in computing systems*, 2017, pp. 3506–3510.

[6] Hiroki Ouchi and Yuta Tsuboi, "Addressee and response selection for multi-party conversation," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 2133–2143.

[7] Cao Liu, Kang Liu, Shizhu He, Zaiqing Nie, and Jun Zhao, "Incorporating interlocutor-aware context into response generation on multi-party chatbots," in *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, 2019, pp. 718–727.

[8] Rui Zhang, Honglak Lee, Lazaros Polymenakos, and Dragomir Radev, "Addressee and response selection in multi-party conversations with speaker interaction rnns," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018, vol. 32.

[9] Wenpeng Hu, Zhangming Chan, Bing Liu, Dongyan Zhao, Jinwen Ma, and Rui Yan, "Gsn: A graph-structured network for multi-party dialogues," .

[10] Iulian Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau, "Building end-to-end dialogue systems using generative hierarchical neural network models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2016, vol. 30.

[11] Minghui Qiu, Feng-Lin Li, Siyu Wang, Xing Gao, Yan Chen, Weipeng Zhao, Haiqing Chen, Jun Huang, and Wei Chu, "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)*, 2017, pp. 498–503.

[12] Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations, ICLR 2015*, 2015.

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," in *NIPS*, 2017.

[14] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever, "Improving language understanding by generative pre-training," .

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[16] Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu, "Mass: Masked sequence to sequence pre-training for language generation," in *International Conference on Machine Learning*. PMLR, 2019, pp. 5926–5936.

[17] Zhaojiang Lin, Peng Xu, Genta Indra Winata, Farhad Bin Siddique, Zihan Liu, Jamin Shin, and Pascale Fung, "Caire: An end-to-end empathetic chatbot," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, vol. 34, pp. 13622–13623.

[18] Ruxin Tan, Jiahui Sun, Bo Su, and Gongshen Liu, "Extending the transformer with context and multi-dimensional mechanism for dialogue response generation," in *CCF International Conference on Natural Language Processing and Chinese Computing*. Springer, 2019, pp. 189–199.

[19] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," in *EMNLP*, 2014.

[20] Sekitoshi Kanai, Yasuhiro Fujiwara, and Sotetsu Iwamura, "Preventing gradient explosions in gated recurrent units," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 435–444.

[21] Ryan Lowe, Nissan Pow, Iulian Vlad Serban, and Joelle Pineau, "The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems," in *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 2015, pp. 285–294.

[22] Ilya Loshchilov and Frank Hutter, "Sgdr: Stochastic gradient descent with warm restarts," *arXiv preprint arXiv:1608.03983*, 2016.

[23] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

[24] Chin-Yew Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.

[25] Ilya Sutskever, Oriol Vinyals, and Quoc V Le, "Sequence to sequence learning with neural networks," *Advances in neural information processing systems*, vol. 27, 2014.

[26] Laurens Van der Maaten and Geoffrey Hinton, "Visualizing data using t-sne.," *Journal of machine learning research*, vol. 9, no. 11, 2008.