EmoElicitor: An Open Domain Response Generation Model with User Emotional Reaction Awareness

Shifeng Li[1] , **Shi Feng**[1] , **Daling Wang**[1] , **Kaisong Song**[2] , **Yifei Zhang**[1] and **Weichao Wang**[1]
[1]Northeastern University, Shenyang, China
[2]Alibaba Group, Hangzhou, China
valar000@outlook.com, {fengshi, wangdaling}@cse.neu.edu.cn, kaisong.sks@alibaba-inc.com,
zhangyifei@cse.neu.edu.cn, wangwecha@gmail.com

Abstract

Generating emotional responses is crucial for building human-like dialogue systems. However, existing studies have focused only on generating responses by controlling the agents' emotions, while the feelings of the users, which are the ultimate concern of a dialogue system, have been neglected. In this paper, we propose a novel variational model named EmoElicitor to generate appropriate responses that can elicit user's specific emotion. We incorporate the next-round utterance after the response into the posterior network to enrich the context, and we decompose single latent variable into several sequential ones to guide response generation with the help of a pre-trained language model. Extensive experiments conducted on real-world dataset show that EmoElicitor not only performs better than the baselines in term of diversity and semantic similarity, but also can elicit emotion with higher accuracy.

1 Introduction

Emotional interaction is a key factor in interpersonal communication and has become a crucial concern in building human-like dialogue agents [Picard, 1997]. Ample evidence [Partala and Surakka, 2004; Prendinger and Ishizuka, 2005] has shown that agents capable of expressing emotions can significantly improve user satisfaction during human-computer interactions. Emotional response generation (ERG) is an emotional interaction task for generating appropriate conversational responses conditioned on a given emotion. Early studies [Skowron, 2009] manually designed rules to select the desired emotional responses from a corpus. More recently, great achievements have been witnessed along this line of research due to the availability of large-scale dialogue data [Zhou and Wang, 2018; Rashkin *et al.*, 2019] and the development of Seq2Seq models [Sutskever *et al.*, 2014; Zhou *et al.*, 2018; Song *et al.*, 2019; Zhong *et al.*, 2019].

Although the previous methods have achieved promising results, these models have attempted only to control the emotion of the agent's response. Meanwhile, the feelings of the user during the interaction, which are actually the ultimate concern when designing a dialogue agent, are neglected. In Figure 1, suppose that we wish to design an agent that can reassure users and that the emotion label *Happiness* is provided to two agents. The traditional ERG method (**Agent A**) can generate the topic-relevant (cat) response with the right emotion. However, **Agent A** fails to comfort the user because it does not take the user's emotional reaction into account. By contrast, **Agent B** can elicit the emotion of *Happiness* from the user because it considers the perspective of the user and generates a more appropriate response based not only on the backward context but also on the user's likely next utterance, with the corresponding emotion.

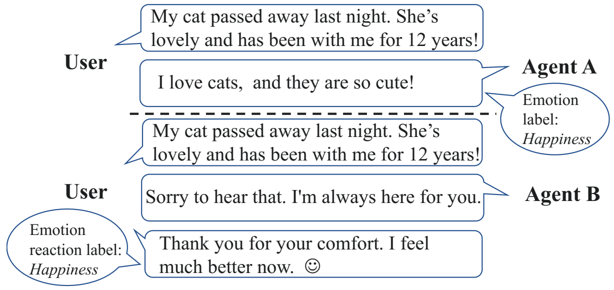
My cat passed away last night. She's lovely and has been with me for 12 years!
User
I love cats, and they are so cute!
Agent A
Emotion label: *Happiness*
My cat passed away last night. She's lovely and has been with me for 12 years!
User
Sorry to hear that. I'm always here for you.
Agent B
Emotion reaction label: *Happiness*
Thank you for your comfort. I feel much better now. ☺

Figure 1: An example of a response (from B) that successfully elicits the emotion *Happiness* from the user, compared with a generated response (from A) that merely exhibits the specified emotion.

In this paper, we focus on the task of open domain response generation for emotion elicitation (RGEE), in which, given the backward context and a desired emotional reaction (considering the next-round utterance and its emotional reaction label), the objective is to elicit a topic-coherent response that can elicit the specific desired emotion. The RGEE task is fundamentally different from the ERG task. (i) Different goals: The goal of ERG is to generate a response that exhibits a specific emotion, whereas RGEE focuses on eliciting a specific emotion to enhance the interactiveness of chatting. Thus, RGEE requires responding from a user-oriented perspective and is more proactive than traditional ERG. (ii) Different inputs: The emotion labels are conceptually different for the two tasks. Furthermore, during the training phase, RGEE considers not only the backward context but also the next-round utterance of the user, which has rarely been considered in previous dialogue generation models.

RGEE facilitates the construction of a more believable and human-like chat agent that can conduct empathetic interactions [Rashkin *et al.*, 2019] or meet a user's emotional needs [Picard and Klein, 2001]. However, RGEE is a highly challenging task since the generated response is not directly associated with the emotional reaction label. Moreover, although incorporating user's next-round utterance during training facilitates the generation of more coherent responses, the next-round utterance is not available during the inference phase.

To tackle these challenges, we propose a variational model EmoElictor, which is built upon a pre-trained language model [Yang *et al.*, 2019], and can capture the relationship between emotional reactions and responses. The original single latent variable is decomposed into sequential ones at each time step during the generation, which helps to enhance the topic coherence and emotional consistency of the response.

Our three main contributions can be summarized as follows: (i) We formulate the RGEE problem as the problem of generating a response that can elicit a specific emotion from a user conditioned on the backward context and next-round utterance. (ii) We propose a variational model EmoElicitor which, to best of our knowledge, is the first to leverage sequential latent variables to capture context information and guide response generation with the help of pre-trained language model. (iii) We construct a large-scale dataset for the RGEE task[1]. Experimental results show that our model consistently outperforms strong baseline methods.

2 Related Work

Emotion-aware dialogue systems have become an emerging area of research in recent years. Zhou *et al.* [2018] first incorporated emotional factors into a dialogue generation task using an end-to-end neural learning framework. Zhong *et al.* [2019] considered the VAD affect model and the effects of negators and intensifiers via an attention mechanism in conversation modeling. Zhou and Wang [2018] proposed a reinforced CVAE-based model called Mojitalk, which could generate responses based on emojis. Rashkin *et al.* [2019] focused on empathethic dialogue generation, in which each conversation contained only one emotion label.

Most previous studies, however, have considered only the emotion of the agent's response while neglecting the user's emotional reaction. Lubis *et al.* [2018; 2019] generated responses that could elicit positive emotions. Hasegawa *et al.* [2013] leveraged a statistical machine translation model to generate responses that could elicit predicted emotions from users. In contrast to the above two methods, our model not only utilizes finer-grained emoji labels but also considers the user's next-round utterance to generate more topic-coherent and emotionally consistent responses.

For a given context, there could be multiple appropriate responses; hence, response generation is known to be a one-to-many problem. Variational autoencoders (VAEs) are one of the most successful types of models for solving such problems [Serban *et al.*, 2017]. However, traditional VAE-based models utilize only a single latent variable to encode an entire response sequence. Our work is inspired by variational recurrent neural networks (VRNNs) [Chung *et al.*, 2015], which perform variational inference at every time step during decoding. Variational models suffer from posterior collapse issues; however, these can be alleviated by means of Kullback-Leibler (KL) annealing and auxiliary loss.

3 Our Approach

3.1 Problem Formulation

Given a dialogue context C and an emotional reaction label e for the next-turn speaker, we aim to generate a response Y that not only is coherent with the context C but also can elicit the desired emotion e from the next-turn speaker.

$$Y = \underset{Y'}{\arg\max} P\left(Y'|C, e\right) \tag{1}$$

The dialogue context is $C = \{U_1^b, U_2^b, U_3^a, \ldots, U_i^b\}$, where U_i^b denotes the utterance of person b in round i. The response is $Y = \{U_{i+1}^a\}$. The emotional reaction e is the emotion label of the next-round utterance, U_{i+2}^b; hence, U_e denotes the next-round U_{i+2}^b.

3.2 CVAE

Our method is built upon conditional VAE (CVAE) [Kingma and Welling, 2014], for where the generation of response y is based on the given context C, the next-round utterance U_e, the emotional reaction label e, and a latent variable z that is intended to capture the distribution of responses.

$$\begin{aligned}\mathcal{L} =& \mathbb{E}_{q_\theta(z|C,Y,U_e)}\left[p(Y|z,C,e)\right] \\ & - \mathrm{KL}\left(q_\theta(z|C,Y,U_e) \| p_\phi(z|C,e)\right)\end{aligned} \tag{2}$$

where KL is the Kullback-Leibler divergence, $p(Y|z, C, e)$ is a decoder that generates Y from the latent variable z, conditional context C and emotional reaction e; $p_\phi(z|C, e)$ is the prior model used to sample z from the prior distribution; $q_\theta(z|C, Y, U_e)$ is the posterior network used to approximate the true posterior distribution of the latent variable z; θ and ϕ are the parameters of the model. In the training phase, the latent variable z sampled from the posterior model is used to generate a response $p(Y|z, C, e)$ given conditions C and e. In the inference phase, only the latent variable z sampled from the prior model is utilized to generate a response $p(Y|z, C, e)$.

Inspired by variational recurrent neural network, our model uses a latent variable z_t in each time step to generate a response as $p_\phi(Y|z, C, e) = \prod_t p\left(Y_t|Y_{<t}, z_t, C, e\right)$.

3.3 Model Framework

Input Representation

The inputs to our multi-turn dyadic dialogue model are the conversation context C, the response Y, and the next-round utterance U_e with emotional reaction label e. All input texts are tokenized using SentencePiece[2]. To capture fine-grained and realistic sentiment labels, we regard emojis as the labels of utterances, and these emojis are treated as plain text during processing. The input embedding for each token includes a word embedding, a type embedding and a position embedding, as visualized in Figure 3.

[1] https://github.com/neuChatbotDS/emoElicitorData

[2] https://github.com/google/sentencepiece

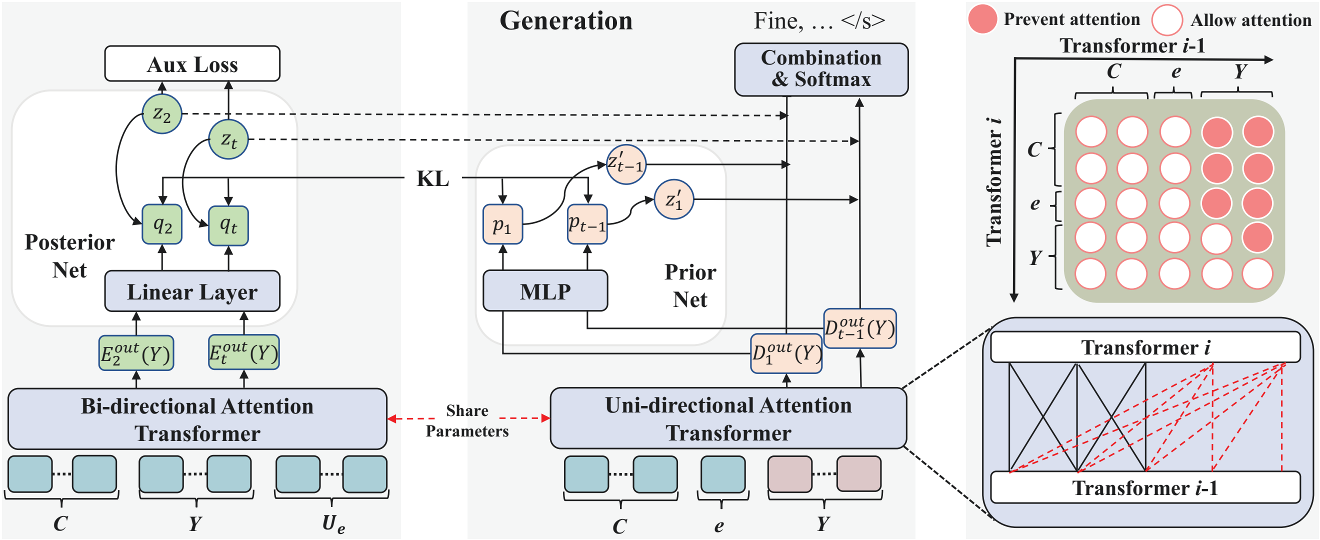

Figure 2: The overall framework of our EmoElicitor model. In the training phase, the latent variable z_t sampled from q_t is fed into the combination layer with $D_{t-1}^{out}(Y)$, as represented by a dashed line, and encodes the context C, the response Y, and the next utterance U_e. In the inference phase, only generation box is executed, and the prior net is used in place of the posterior net, as shown by a solid line. The right part of the figure shows the details of the uni-directional attention Transformer, which is also used in the XLNet model.

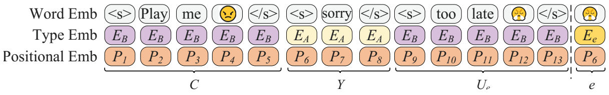

Figure 3: Input representation for the bi-directional Transformer.

(i) The input representation is the concatenation of C, Y and U_e with a special beginning token [s] and end token [/s]. For the context C, we concatenate all the utterances in C with a special token [SEP]. We add an auxiliary emoji vocabulary for emojis. (ii) Type embeddings are employed to differentiate the speakers (e.g. A and B) and the emotion label in the dyadic conversation data. Suppose that speaker A's utterance elicits an emotion from speaker B. Thus, type embeddings of Y and U_e are E_A and E_B, respectively. E_e denotes emotional reaction label type embedding. (iii) The position embeddings are consistent with those of the pre-trained model.

Our input representation is compatible with the Transformer-based pre-trained language models that include token, segment and position embeddings. Note that we replace segment embedding with type embedding, which is consistent with the features of dyadic conversations and fine-tune the type embeddings on the training dataset.

Framework

Figure 2 shows the overall framework of our proposed model, which is a combination of a pre-trained language model and a variational model. The backbone of our infrastructure is inspired by [Lample and Conneau, 2019], a technology that flexibly supports bi-directional encoding and uni-directional decoding through specific self-attention masks. To better capture the backward and forward context representations, we use a bi-directional attention Transformer to model the context C and the emotional reaction label e. A uni-directional attention Transformer is leveraged to model the response Y to be generated. The encoder and decoder Transformers share the same set of weights.

Specifically, the input and output are respectively denoted by E^{in} and E^{out} for the encoder, and D^{in} and D^{out} for the decoder. All inputs are sequences of tokens, including token embeddings (TOK), type embeddings (TYP), and position embeddings (POS):

$$E_i^{in}(C;e) = \{TOK_i(C;e), TYP_i(C;e), POS_i(C;e)\}$$

where $C;e$ denotes the concatenation of C and e. The encoder can attend its own tokens (e.g., $E_i^{in}(C;e)$ can attend arbitrary $E_j^{in}(C;e)$), and each token in the decoder can attend only those ahead of it.

$$E^{out}(C;e); D^{out}(Y) = \text{Transformer}(E^{in}(C;e); D^{in}(Y))$$

$$O_j = D_j^{out}(Y)W_{oo} + b_o \tag{3}$$

$$P_j = \text{softmax}(O_j) \tag{4}$$

where $E^{in}(C;e); D^{in}(Y)$ denotes the concatenation of $E^{in}(C;e)$ and $D^{in}(Y)$; $D_j^{out}(Y)$ is the final output of the decoder when taking response Y as input at time step j; $W_{oo} \in \mathbb{R}^{d_{\text{model}} \times d_{\text{vocab}}}$ and $b_o \in \mathbb{R}^{d_{\text{vocab}}}$ are parameters of the pre-trained model, with d_{vocab} being the vocabulary size, which is the same as the size of the original vocabulary because our responses do not contain emojis; and P_j is the probability distribution of the word to be generated at time step j.

3.4 EmoElicitor

Traditional CVAE-based methods employ only one latent variable z, which make it difficult to model the distribution of the response $p(y|z,c)$. To more effectively use latent variables to capture the relationship between emotional reactions and responses, we decompose z into sequential variables z_t at each time step t during the generation process. Moreover, we add an auxiliary loss to avoid *posterior collapse*.

EmoElicitor Model Details

On the basis of a CVAE and a pre-trained language model, we build EmoElicitor which leverages the context C, the response Y and the next utterance U_e to guide response generation at every time step t. The posterior net encodes C, Y and U_e. Since we assume that z_t follows an isotropic Gaussian distribution $N\left(\mu, \sigma^2 \mathbf{I}\right)$, we have

$$E^{out}(C;Y;U_e) = \text{Transformer}(E^{in}(C;Y;U_e)) \tag{5}$$

$$E^{out}(Y) = \{E_2^{out}(Y), \ldots, E_n^{out}(Y)\} \tag{6}$$

$$\left[\begin{array}{c} \mu_t \\ \sigma_t^2 \end{array}\right] = E_t^{out}(Y)\, W_q + b_q \tag{7}$$

$$z_t \sim N\left(\mu_t, \sigma_t^2 \mathbf{I}\right) \tag{8}$$

where $E^{in}(C;Y;U_e)$ is the input representation for the bi-directional attention Transformer, which concatenates the context C, the response Y and the next utterance U_e as described in Section 3.3. Note that the position embeddings are also sequential. $E^{out}(Y)$ is the bi-directional attention output representation of the response, which is a sequential representation $E_t^{out}(Y)$ of length $n-1$ because the beginning-of-sentence token [s] is not needed in the output representation of response Y for the computation of the posterior latent variables. z_t denotes the posterior latent variable at time step t. $W_q \in \mathbb{R}^{d_{model} \times d_z}$ and $b_q \in \mathbb{R}^{d_z}$ are weight parameters, where d_z is the dimensionality of the latent variable and d_{model} is the dimensionality of the pre-trained model output at each time step t. The prior net encodes the context C and the reaction emotion e. Similarly, we have

$$E^{out}(C;e); D^{out}(Y) = \text{Transformer}(E^{in}(C;e); D^{in}(Y))$$

$$D^{out}(Y) = \{D_1^{out}(Y), D_2^{out}(Y), \ldots, D_{n-1}^{out}(Y)\} \tag{9}$$

$$\left[\begin{array}{c} \mu'_{t-1} \\ \sigma'^2_{t-1} \end{array}\right] = \text{MLP}_p(D_{t-1}^{out}(Y)) \tag{10}$$

$$z'_{t-1} \sim N\left(\mu'_{t-1}, \sigma'^2_{t-1} \mathbf{I}\right) \tag{11}$$

where $E^{in}(C;e)$ is the input representation for the bi-directional attention Transformer, which concatenates the context C and the next-utterance emotion e. $D^{in}(Y)$ is the input representation for the uni-directional attention Transformer, which attends those tokens ahead of it. $D^{out}(Y)$ is the uni-directional attention output representation of the response, which is a sequential representation $E_t^{out}(Y)$ of length $n-1$, because the response Y includes an end-of-sentence token [/s] that is not needed during the generation process. MLP$_p$ is a multi-layer perceptron.

We incorporate time step latent z_t into the pre-trained model's output by a combination layer to predict Y_t by computing $p\left(Y_t | Y_{<t}, z_t, C, e\right)$.

$$G_{t-1} = \tanh\left(\left[D_{t-1}^{out}(Y), z_t\right] W_g\right) \tag{12}$$

$$O_{t-1} = \left[D_{t-1}^{out}(Y), G_{t-1}\right] W_o + b_o \tag{13}$$

$$p\left(Y_t | Y_{<t}, z_t, C, e\right) = \text{softmax}\left(O_{t-1}\right) \tag{14}$$

where $W_g \in \mathbb{R}^{(d_{model}+d_z) \times d_z}$ is the weight parameter used to combine z_{t+1} and $D_t^{out}(Y)$ into G_t. $W_o \in \mathbb{R}^{(d_{model}+d_z) \times d_{vocab}}$ includes $W_{o1} \in \mathbb{R}^{d_{model} \times d_{vocab}}$ and $W_{o2} \in \mathbb{R}^{d_z \times d_{vocab}}$: the first part loads the parameters of original pre-trained model W_{oo}, and the second part is randomly initialized by Xavier method [Glorot and Bengio, 2010].

Learning

A VAE-based model will often ignore the latent variables, causing the posterior to collapse. For a text generation task, Zhao *et al.* [2017] utilized the bag-of-word (BOW) loss to alleviate this problem and thus achieved improved performance. In this paper, we introduce a new auxiliary loss that uses the posterior latent variable z_t to predict the corresponding word in every generation step to preserve information.

$$P_t^{aux}(Y_t | z_t) = \text{softmax}\left(z_t W_z + b_o\right) \tag{15}$$

$$\mathcal{L}_t^{aux} = \mathbb{E}_{q_\theta(z_t | C, Y, U_e)}\left[\log P_t^{aux}(Y_t | z_t)\right] \tag{16}$$

Our final loss function is a weighted sum of $\mathcal{L}_t^{aux}$ and $\mathcal{L}_t^{elbo}$ at each time step t:

$$\mathcal{L} = \sum_t \left[\mathcal{L}_t^{elbo} + \alpha \mathcal{L}_t^{aux}\right] = \sum_t \left[\left(\mathcal{L}_t^{LM} - \mathcal{L}_t^{KL}\right) + \alpha \mathcal{L}_t^{aux}\right]$$

where α is the weight controlling auxiliary loss, $\mathcal{L}_t^{LM}$ is the log-likelihood loss when predicting Y_t, and $\mathcal{L}_t^{KL}$ is KL divergence of the approximate posterior distribution q_t and prior distribution p_{t-1}. The two losses are calculated as follows:

$$\mathcal{L}_t^{LM} = \mathbb{E}_{q_\theta(z_t | C, Y, U_e)}\left[\log p\left(Y_t | Y_{<t}, z_t, C, e\right)\right] \tag{17}$$

$$\mathcal{L}_t^{KL} = \text{KL}\left(q_\theta(z_t | C, Y, U_e) \| p_\phi\left(z'_{t-1} | Y_{<t}, C, e\right)\right) \tag{18}$$

In the generation phase, we predict the token Y_t by computing $p\left(Y_t | Y_{<t}, z'_{t-1}, C, e\right)$, in which the posterior latent variable z_t is replaced with the prior latent variable z'_{t-1}.

4 Experiment

Since there is no off-the-shelf multi-turn emoji-rich dialogue dataset available, we collect a large corpus of Twitter conversations with emojis. We use 58 common emojis as the labels, consistent with the selection used in Mojitalk [Zhou and Wang, 2018] except that the six emojis with the lowest frequencies are removed. Each conversation in our corpus consists of at least three rounds, where the first round belongs to the context and the last round is U_e containing an emoji. If there is more than one emoji in U_e, we select the most frequently used emoji as the emotional reaction label. When the frequencies are equal, we select the emoji with the lowest frequency in the corpus because it is most distinctive.

4.1 Data Preprocessing

During preprocessing, all mentions and hashtag tokens were removed, and repeated letters and symbols were shortened. To better capture the emotions of the conversations, we have added an external emoji vocabulary of size 1k, which is shared with the label space of the emotional reactions e.

The distribution of our emotional reaction labels e is close to that of Mojitalk (e.g., 😂 accounts for 34%). We have removed dialogues with fewer than 6 words and more than 34 words in the response Y and the emotional reaction utterance U_e. Dialogues with fewer than 6 words and more than 100

Models	**Y-avg%**	**U_e-avg%**	**U_e-gre%**	**Dist1%**	**Dist2%**	**BLUE%**	**B1%**	**B2%**	**Avg-len**	**Acc5%**
Human	-	70.62	42.15	10.6	47.50	-	-	-	15.22	65.1
S2S*	71.11	68.68	39.11	0.52	1.422	15.75	23.15	20.35	14.50	50.1
ECM*	70.98	68.56	39.59	0.30	1.085	13.57	21.05	17.83	10.80	48.0
Mojitalk*	71.03	68.70	39.72	4.16	13.18	15.04	22.61	19.35	12.90	50.1
XLNet	71.77	69.03	40.32	4.13	13.82	14.55	22.84	19.16	11.52	49.1
T-CVAE	71.99	69.25	40.26	4.96	21.62	15.99	24.31	20.58	12.92	50.2
w/o U_e	71.51	68.99	39.95	**5.10**	21.73	15.12	23.20	19.56	12.20	49.6
Ours	**73.18**	**70.34**	**41.92**	4.08	21.97	**18.54**	**28.08**	**23.53**	**15.73**	**51.7**
w/o U_e	72.77	70.04	41.36	4.56	**22.00**	17.71	27.02	22.73	14.45	51.1
w/o pre-train	72.54	70.11	40.80	2.46	21.50	17.29	25.52	21.87	15.20	50.8

Table 1: The automatic evaluation results for the generated response. The symbol * means the method is without pre-trained model.

words in the context C have also been eliminated. We randomly split the corpus into 137,421/4,661/4,739 conversation pairs for train/validation/test set. For building validation and test dataset, we choose the response Y with emoji as label so as to test the emotional accuracy of the generated response. Note that the emotion labels of the response Y is not considered in train dataset, and all emojis in Y are all removed.

4.2 Implementation Details

We chose XLNet-base [Yang *et al.*, 2019] as our pre-trained model; thus, d_{model} is 768, and the input and output vocabulary sizes are 33k and 32k, respectively.

The Adam optimizer with an initial learning rate of $1e^{-5}$ was applied to all models. The batch size was set to 15, and greedy search was used for all methods. For the variational model, d_z was set to 64, and the temperature used in the KL annealing strategy was varied from 0 to 1 in 10k steps. All experiments were conducted on a single 11 GB NVIDIA GeForce RTX 2080 Ti GPU card and took approximately seven hours at most to reach convergence.

4.3 Baselines

For all of the baseline methods, we use the same input vocabulary for comparison.

S2S: A simple seq2seq model based on GRU.

ECM: The emotional chatting machine which uses internal emotion memory and external emotion memory for emotion expression [Zhou *et al.*, 2018].

Mojitalk: An encoder-decoder based CVAE model incorporated with emotion embedding [Zhou and Wang, 2018].

XLNet: A seq2seq model with an emotional reaction label e, and initialized with XLNet. Uni-directional attention is utilized for generation, as shown in Figure 2.

T-CVAE: A Transformer-based CVAE model. In contrast to the original T-CVAE implementation [Wang and Wan, 2019], our implementation is based on XLNet; therefore, the combination layer of T-CVAE is the same as that of EmoElicitor. We also add the BOW loss to alleviate model collapse.

4.4 Automatic Evaluation

Results for Response Generation

Embedding-based metrics: Y-avg [Liu *et al.*, 2016] calculate the semantic similarity between the generated response and the ground truth response by averaging word embeddings based on Twitter word embeddings [Godin, 2019]. **U_e-avg** and **U_e-Gre** calculate the semantic similarity between the generated response and the ground truth reaction utterance U_e based on average and greedy matching, respectively.

Models	**Acc10%**	**Pro**	**Pro@5%**
XLNet	21.5	1.03	35.8
T-CVAE	22.0	1.05	36.2
T-CVAE w/o U_e	21.8	1.04	35.9
Ours	23.0	**1.13**	**36.5**
w/o U_e	22.5	1.06	36.4
w/o pre-train	**24.3**	1.06	36.3

Table 2: Automatic evaluation results for emotion elicitation

Model	**Gram%**	**TC%**	**EC%**
XLNet	79.1	61.1	38.2
T-CVAE	85.8	66.2	43.2
Ours	**87.8**	**71.6**	**55.7**

Table 3: Manual evaluation

Dist1 and Dist2: The proportions of distinct unigrams and bigrams [Li *et al.*, 2016] in the generated responses.

BLEU and B1/2: Word-overlap scores against gold-standard responses. BLEU [Papineni *et al.*, 2002] refers to BLEU-4, and B1 and B2 refer to other n-gram scores.

Acc5: We adopt the top-5 emoji accuracy [Zhou and Wang, 2018] to evaluate the agreement between the expected emoji category of the real response and the emotion category of the generated response predicted by a well-trained classifier. Note that the responses in the validation and test datasets have emoji labels that are conceptually different from the emotional reaction labels of U_e.

Avg-len: The average response length.

As shown in Table 1, our model performs best in terms of Y-avg, U_e-avg, U_e-Gre, Dist2, BLEU, Avg-len and Acc5. The models based on Transformer perform better than that base on RNN. In this table, 'w/o U_e' denotes the case in which U_e is eliminated and only the emotional reaction label e is used in the posterior net. The models that incorporate latent variables (i.e., T-CVAE and EmoElicitor) can generate more diverse responses. We observe that T-CVAE performs better than our model in terms of Dist1; this is because the

Context	U_1^b: Aaah how sweet!! They all ok? Xx		
Emoji	😂	😭	😓
Ours	They're not .. they need to have the flu shot on their asses	I'm good. They're just having a heart attack	They all are !! I'm still in shock about what they did to the poor boys
T-CVAE	They are ok, just fed up with the cold weather xx	They're all good thanks! I'll be in a bit of pain for a while, but I'll be fine x	They are, just fed up with the cold weather. They are now going home to bed. Xx
XLNet	I'm good thanks. I'm just tired of the cold. I'm just tired of the cold. Xx	I'm good thanks. I'm just tired of the cold. I'm just tired of the cold. Xx	I'm good thanks. I'm just tired of the cold. I'm just tired of the cold. Xx

Table 4: Some examples of generated responses. The emojis represent the emotions that are to be elicited from user b based on the responses.

longer responses of our model result in a larger denominator in the formula Dist1 $= N^{dist1}/N^{total}$.

The models without U_e perform more poorly in terms of Y-avg, U_e-avg, U_e-Gre, BLEU, Acc5 and Avg-len, indicating that considering U_e can help to generate more coherent responses in conversation. However, the Dist metrics do not decrease as the others do. We conjecture that the posterior latent variables can benefit from U_e but that U_e has limited effects on the prior net, which contributes more to the diversity. On the other hand, the Dist metrics of our model without the pre-trained model do decrease. This finding confirms the effectiveness of the pre-trained model in learning the dependency on context.

Results for Emotion Elicitation

To evaluate model's emotion elicitation ability, for each selected 58 emojis we generate one response Y_e in reply to each context C in test set.

Acc10 is the prediction accuracy for the emotion e that is elicited based on the generated response Y_e and context C. Note that we employ Acc10 (top-10 emoji accuracy) because elicited emotion prediction is much more difficult than response emotion classification.

Pro $= \sum_e p(e|C, Y_e)$ is the sum of probabilities $p(e|C, Y_e)$.

Pro@5 is the sum of the probabilities of the top 5 emotion labels for every context C. We built another classifier to predict the emotional reaction e based on a given context C.

In Table 2, the Acc10 score of our method increases without the pre-trained model. This is because our classifier can more easily judge emotional words (e.g., *love you* ❤️), whereas the words generated by the pre-trained model are more diverse. Pro directly calculates the probability of eliciting emotions, which is fair across all categories, while Acc10 often ignores some difficult-to-judge categories (e.g., 👀). Our model scores highest in terms of Pro and Pro@5, indicating that it is the model that is most effective in eliciting all desired emotions.

4.5 Manual Evaluation

To better evaluate the quality of the generated responses, we performed manual evaluation. Given a post and an emotional reaction label, responses generated from all the Transformer-based models were randomized and presented to five graduate students majoring in sentiment analysis. We randomly sampled 200 posts from the test set and selected a specific emoji as the target emotional reaction label to generate a response in accordance with the emoji distribution.

Each annotator was asked to score each response in terms of each of the following metrics with a rating of 0 or 1: (i) **Grammar (Gram):** whether the generated response is natural and fluent. (ii) **Topic coherency (TC):** whether the response is topically coherent to the context and reasonable in logic. (iii) **Emotional consistency (TC):** whether the response can elicit the given emotion.

Results: We calculated the Fleiss' kappa [1971] to measure interrater consistency. Fleiss' kappa for Gram, TC and EC are 0.84, 0.60 and 0.58, indicating substantial, moderate and moderate agreement respectively. As shown in Table 3, the performance of our model is consistently in line with the human perspective. Notably, XLNet can elicit emotions with only 38.2% success because of the imbalance of the emotion categories (e.g., 😂 accounts for 34%).

4.6 Case Study

We sample some generated responses from all three models in Table 4. The XLNet model simply generates the same responses for different emoji reaction labels because it is difficult for a single emoji to represent the user's reaction, which causes XLNet to ignore the emojis. T-CVAE always generates the same words at the beginning of each sentence (e.g., *They are*). We speculate that the single latent variable z has difficulty affecting the decoder output at the beginning. By contrast, our model generates diverse responses for different emoji (e.g., *shot on their asses* 😂, *heart attack* 😭), thus demonstrating that latent variable z_t at time step t is able to capture dependency between words and emotional reactions.

5 Conclusion

In this paper, we investigate emotional reactions, including next-round utterances and the corresponding elicited emotion labels, in dyadic dialogue generation and propose a novel model called EmoElicitor to generate responses with emotional reaction awareness. By incorporating a latent variable z_t in every time step, we can capture the dependency between words and emotional reactions, thus allowing our model to generate coherent, diverse responses with the intent of eliciting different emotional reactions.

Acknowledgments

The work was supported by the National Key R&D Program of China under grant 2018YFB1004700, National Natural Science Foundation of China (61872074, 61772122).

References

[Chung *et al.*, 2015] Junyoung Chung, Kyle Kastner, Laurent Dinh, Kratarth Goel, Aaron C. Courville, and Yoshua Bengio. A recurrent latent variable model for sequential data. In *ACL*, pages 2980–2988, 2015.

[Fleiss, 1971] Joseph L Fleiss. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378, 1971.

[Glorot and Bengio, 2010] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*, pages 249–256, 2010.

[Godin, 2019] Fréderic Godin. *Improving and Interpreting Neural Networks for Word-Level Prediction Tasks in Natural Language Processing*. PhD thesis, Ghent University, Belgium, 2019.

[Hasegawa *et al.*, 2013] Takayuki Hasegawa, Nobuhiro Kaji, Naoki Yoshinaga, and Masashi Toyoda. Predicting and eliciting addressee's emotion in online dialogue. In *ACL Volume 1: Long Papers*, pages 964–972, 2013.

[Kingma and Welling, 2014] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

[Lample and Conneau, 2019] Guillaume Lample and Alexis Conneau. Cross-lingual language model pretraining. *CoRR*, abs/1901.07291, 2019.

[Li *et al.*, 2016] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. In *NAACL*, pages 110–119, 2016.

[Liu *et al.*, 2016] Chia-Wei Liu, Ryan Lowe, Iulian Serban, Michael Noseworthy, Laurent Charlin, and Joelle Pineau. How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *EMNLP*, pages 2122–2132, 2016.

[Lubis *et al.*, 2018] Nurul Lubis, Sakriani Sakti, Koichiro Yoshino, and Satoshi Nakamura. Eliciting positive emotion through affect-sensitive dialogue response generation: A neural network approach. In *AAAI-18*, pages 5293–5300, 2018.

[Lubis *et al.*, 2019] Nurul Lubis, Sakriani Sakti, Koichiro Yoshino, and Satoshi Nakamura. Positive emotion elicitation in chat-based dialogue systems. *IEEE/ACM Trans. Audio, Speech & Language Processing*, 27(4):866–877, 2019.

[Papineni *et al.*, 2002] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002.

[Partala and Surakka, 2004] Timo Partala and Veikko Surakka. The effects of affective interventions in human-computer interaction. *Interacting with Computers*, 16(2):295–309, 2004.

[Picard and Klein, 2001] Rosalind W. Picard and Jonathan Klein. Computers that recognise and respond to user emotion: theoretical and practical implications. *Interacting with Computers*, 14(2):141–169, 2001.

[Picard, 1997] RW Picard. Affective computing. 1997.

[Prendinger and Ishizuka, 2005] Helmut Prendinger and Mitsuru Ishizuka. The empathic companion: A character-based interface that addresses users' affective states. *Applied Artificial Intelligence*, 19(3-4):267–285, 2005.

[Rashkin *et al.*, 2019] Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *ACL Volume 1: Long Papers*, pages 5370–5381, 2019.

[Serban *et al.*, 2017] Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron C. Courville, and Yoshua Bengio. A hierarchical latent variable encoder-decoder model for generating dialogues. In *AAAI*, pages 3295–3301, 2017.

[Skowron, 2009] Marcin Skowron. Affect listeners: Acquisition of affective states by means of conversational systems. In *Development of Multimodal Interfaces: Active Listening and Synchrony, Second COST 2102 International Training School, Dublin, Ireland, March 23-27, 2009, Revised Selected Papers*, pages 169–181, 2009.

[Song *et al.*, 2019] Zhenqiao Song, Xiaoqing Zheng, Lu Liu, Mu Xu, and Xuanjing Huang. Generating responses with a specific emotion in dialog. In *ACL*, pages 3685–3695, 2019.

[Sutskever *et al.*, 2014] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112, 2014.

[Wang and Wan, 2019] Tianming Wang and Xiaojun Wan. T-CVAE: transformer-based conditioned variational autoencoder for story completion. In *IJCAI*, pages 5233–5239, 2019.

[Yang *et al.*, 2019] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. *CoRR*, abs/1906.08237, 2019.

[Zhao *et al.*, 2017] Tiancheng Zhao, Ran Zhao, and Maxine Eskénazi. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *ACL Volume 1: Long Papers*, pages 654–664, 2017.

[Zhong *et al.*, 2019] Peixiang Zhong, Di Wang, and Chunyan Miao. An affect-rich neural conversational model with biased attention and weighted cross-entropy loss. In *AAAI*, pages 7492–7500, 2019.

[Zhou and Wang, 2018] Xianda Zhou and William Yang Wang. Mojitalk: Generating emotional responses at scale. In *ACL Volume 1: Long Papers*, pages 1128–1137, 2018.

[Zhou *et al.*, 2018] Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *AAAI*, pages 730–739, 2018.