Progressive Open-Domain Response Generation with Multiple Controllable Attributes

Haiqin Yang[1] , **Xiaoyuan Yao**[1] , **Yiqun Duan**[1] , **Jianping Shen**[1] , **Jie Zhong**[1] and **Kun Zhang**[2]
[1]Ping An Life Insurance Company of China
[2]Carnegie Mellon University
hqyang@ieee.org, yaoxiaoyuan617@pingan.com.cn, duanyiquncc@gmail.com, jpshen2008@qq.com,
zhongjie@pingan.com.cn, kunz1@cmu.edu

Abstract

It is desirable to include more controllable attributes to enhance the diversity of generated responses in open-domain dialogue systems. However, existing methods can generate responses with only one controllable attribute or lack a flexible way to generate them with multiple controllable attributes. In this paper, we propose a Progressively trained Hierarchical Encoder-Decoder (PHED) to tackle this task. More specifically, PHED deploys Conditional Variational AutoEncoder (CVAE) on Transformer to include one aspect of attributes at one stage. A vital characteristic of the CVAE is to separate the latent variables at each stage into two types: a global variable capturing the common semantic features and a specific variable absorbing the attribute information at that stage. PHED then couples the CVAE latent variables with the Transformer encoder and is trained by minimizing a newly derived ELBO and controlled losses to produce the next stage's input and produce responses as required. Finally, we conduct extensive evaluations to show that PHED significantly outperforms the state-of-the-art neural generation models and produces more diverse responses as expected.

1 Introduction

Developing human-like conversational agents is a long-lasting goal of artificial intelligence. Recently, thanks to the availability of a plethora of conversation data on the Internet and the booming of deep learning technologies, researchers have been attracted to explore end-to-end data-driven approaches to building social chatbots [Huang *et al.*, 2020].

Nowadays, sequence-to-sequence (Seq2Seq) models [Serban *et al.*, 2016] have been adopted to generate conversations due to their scalability and promising capability in capturing language-independence to implicitly learn semantic and syntactic relations between message-response pairs and contextual dependencies. However, they usually tend to generate "safe responses", such as "I do not know" and "OK", because the vanilla Seq2Seq models are prone to only memorize high-frequency responses in the data [Serban *et al.*, 2017; Xing *et al.*, 2018]. Various neural generation methods have been proposed to incorporate different controllable attributes or rich information into the Seq2Seq framework to enhance the generation diversity. The attributes may include length [Kikuchi *et al.*, 2016], sentiment and emotion [Hu *et al.*, 2017; Zhou and Wang, 2018; Zhou *et al.*, 2018; Rashkin *et al.*, 2019], tone [Ke *et al.*, 2018; Bi *et al.*, 2019], specificity [Zhang *et al.*, 2018; See *et al.*, 2019], and meta-words [Xu *et al.*, 2019]. Recently, it is desirable to generate responses with multiple controllable attributes because it can allow social chatbots to create more human-like responses and manifest more intelligence from different angles [Huang *et al.*, 2020; Zheng *et al.*, 2020]. However, existing methods usually generate responses with only one controllable attribute or fail to provide a flexible way to generate them with multiple controllable attributes [See *et al.*, 2019].

Post	The match is over. Udinese 2-1 Milan.
R_{H}	$\overleftrightarrow{\text{Congratulations}}$ Udinese, keep going in the new season!
R_{HI}	$\overleftrightarrow{\text{Haha}}$, $\underline{\text{is}}$ Udinese going to offence God's will$\underline{?}$
R_{HIL}	$\overleftrightarrow{\text{Haha}}$, don't know $\underline{\text{why}}$ this time I am still $\overleftrightarrow{\text{very happy}}$

Figure 1: An example of generated responses with progressively fed attributes: The word with ↔ on top indicates its high specificity to the Happy emotion. The $\underline{\text{underlined word}}$ denotes the $\underline{\text{I}}$nterrogative tone while the third character of L requires generating a long response.

In this paper, we develop a new framework, the Progressively trained Hierarchical Encoder-Decoder (PHED), to tackle this task. As illustrated in Fig. 1, PHED effectively generates responses with three different aspects of controllable attributes in a progressive way: the Happy emotion in the first aspect, the Interrogative tone in the second one, and the long response generation requirement in the third one. PHED enjoys prominent properties: (1) It acts as an interface for developers to customize responses by tailoring the attributes partially or fully. In [Xu *et al.*, 2019], all controllable attributes need to be preset. Differently, our PHED can output each stage of responses with one desired attribute at one stage. (2) The framework is extensible and scalable. More aspects of attributes can be easily incorporated in the generation procedure. This is different from existing work on text generation with multiple attributes [Logeswaran *et al.*, 2018; Lample *et al.*, 2019; Shao *et al.*, 2019].

To ensure the relevance of a response to the message and fidelity of the response to the controlled attributes, PHED designs subtle losses under rigorous mathematical derivation. Specifically, we utilize Transformer because it facilitates the self-attention mechanism for many NLP applications [Vaswani *et al.*, 2017]. To ensure the diversity of the generated responses with controllable attributes, we apply Conditional Variational AutoEncoder (CVAE) and separate the CVAE latent variables into two meaningful types of variables: a joint latent variable capturing semantic features shared among all data and specific latent variables, each of which controls the attribute at the corresponding stage. The learned CVAE latent variables are then coupled with the encoding information learned at previous stages to *explicitly* promote the effect of the specific attributes in generating responses. Here, we borrow the idea of story completion in [Wang and Wan, 2019] to utilize the proved effective architecture of Transformer-based CVAE (T-CVAE) to implement the coupling procedure. Different from T-CVAE, PHED does not share the parameters in the encoder and the decoder, but contains more CVAE latent variables, which are optimized by a newly derived evidence lower bound (ELBO) and controlled losses. We conduct extensive evaluations and demonstrate that PHED can generate more diverse responses.

The contribution of our work is threefold: (1) a first work to generate diverse responses with multiple controllable nesting attributes; (2) a unified framework to include only one aspect of controllable attributes at one stage, relying on a hierarchical structure that enjoys flexibility and extensibility with rigorous theoretical guarantee; (3) empirical evaluations clearly demonstrating the effectiveness of PHED.

2 Our Proposal

We present PHED with the theoretical results and its training procedure.

2.1 Preliminaries

Given a corpus, $\mathcal{D} = \{(\mathbf{x}_i, c_i, \mathbf{y}_i)\}_{i=1}^N$, where N is the number of message-response pairs, $\mathbf{x}_i = x_{i1}\, x_{i2}\, \ldots\, x_{i|\mathbf{x}_i|}$ is a message with $|\mathbf{x}_i|$ characters or words, c_i denotes the associated attributes on the response of $\mathbf{y}_i = y_{i1}\, y_{i2}\, \ldots\, y_{i|\mathbf{y}_i|}$, the objective is to learn the conditional probability $p(\mathbf{y}|\mathbf{x}, c)$ from the corpus. Here, the attribute $c = l_1, \ldots, l_K$ enforces the attribute l_i at the i-th stage from K pre-defined aspects, e.g., the emotion of happy or sad [Jiao *et al.*, 2019], and the tone of Declarative, Interrogative, or Imperative [Ke *et al.*, 2018]. After obtaining $p(\mathbf{y}|\mathbf{x}, c)$, given a message $\mathbf{x}$ and a specific attribute c, we will generate response $\mathbf{y}$ accordingly.

We propose Progressively trained Hierarchical Encoder-Decoder (PHED), shown in Fig. 2(a), to enforcing controlling only one aspect of attributes from the data at one stage. The basic structure of PHED resembles T-CVAE [Wang and Wan, 2019], but distinguishes the CVAE variables to $\mathbf{z}_c \in \mathbb{R}^{d_{\mathbf{z}_c}}$ for capturing common semantic features and $\mathbf{z}_i \in \mathbb{R}^{d_{\mathbf{z}_i}}$ for capturing individual features at the i-th stage, where $d_{\mathbf{z}_c}$ and $d_{\mathbf{z}_i}$ denote the size of $\mathbf{z}_c$ and $\mathbf{z}_i$, respectively.

To relieve the burden of the model expression, we define the vanilla Transformer layer for the encoder ($\mathcal{T}_e$) and the decoder ($\mathcal{T}_d$) as follows:

$$\mathbf{h}^i = \mathcal{T}_e(\mathbf{h}^{i-1}) := \begin{cases} \mathbf{A} = \mathbf{MH}_{\mathcal{E}_e}(\mathbf{h}^{i-1}, \mathbf{h}^{i-1}, \mathbf{h}^{i-1}), \\ \mathbf{B} = \mathrm{LN}(\mathbf{h}^{i-1} + \mathbf{A}), \\ \mathbf{h}^i = \mathrm{LN}(\mathrm{FFN}(\mathbf{B}) + \mathbf{B}), \end{cases} \quad (1)$$

$$\mathbf{h}_d^i = \mathcal{T}_d(\mathbf{h}_d^{i-1}, \mathbf{h}_e) := \begin{cases} \mathbf{f} \ = \mathbf{MH}_{\mathcal{E}_d}(\mathbf{h}_d^{i-1}, \mathbf{h}_e, \mathbf{h}_e), \\ \mathbf{g} \ = \mathrm{LN}(\mathbf{h}_d^{i-1} + \mathbf{f}), \\ \mathbf{h}_d^i = \mathcal{T}_e(\mathbf{g}). \end{cases} \quad (2)$$

Here, all the hidden size in the Transformer layer is H. $\mathbf{h}^i, \mathbf{h}_d^i \in \mathbb{R}^H$ denotes the output of the encoder and the decoder at the i-th Transformer layer, respectively. $\mathbf{MH}_{\mathcal{E}_e}$ and $\mathbf{MH}_{\mathcal{E}_d}$ is the multi-head attention network with the input of query, key, and value in the encoder and the decoder, respectively. LN denotes the operation of layer normalization and FFN is a feed forward neural network.

Borrowing T-CVAE in [Wang and Wan, 2019], we define:

$$\mathbf{z} = \mathcal{C}(\Psi, \mathbf{a}) := \begin{cases} \text{I. } \mathbf{v} = \mathbf{MH}_{\Psi}(\boldsymbol{\alpha}, \mathbf{a}, \mathbf{a}), \\ \text{II. } \begin{bmatrix} \boldsymbol{\mu} \\ \log(\boldsymbol{\sigma}^2) \end{bmatrix} = \mathbf{MLP}(\mathbf{v}), \\ \text{III. } \mathbf{z} \sim N(\boldsymbol{\mu}, \boldsymbol{\sigma}^2\mathbf{I}), \end{cases} \quad (3)$$

Hence, $\mathbf{z}$ is sampled from an isotropic Gaussian distribution with mean ($\boldsymbol{\mu}$) and variance ($\boldsymbol{\sigma}$) computed in two steps. In Step I, a hidden feature $\mathbf{v}$ is computed from a multi-head attention network on Ψ, $\mathbf{MH}_{\Psi}$, which takes three inputs, i.e., $\boldsymbol{\alpha}$ (a random initialized context vector) for the query, and $\mathbf{a}$ for the key and the value, respectively. In Step II, $\mathbf{v}$ is fed to a multi-layer perceptron ($\mathbf{MLP}$) to determine the mean and the variance simultaneously.

2.2 Model and Theory

Let $\mathbf{h}^i \in \mathbb{R}^H$ and $E_{dec}^{d_i} \in \mathbb{R}^H$ be the output of the encoder and the decoder at the i-th stage, respectively. d_i is the number of layers in the decoder up to i-th stage, $i = 1, \ldots, K$. When $i = 0$, we compute $\mathbf{h}_{\mathbf{x}}^0$ and $\mathbf{h}_{\mathbf{y}}^0$ by the first Transformer layer:

$$\mathbf{h}^0 = \mathbf{h}_{\mathbf{x}}^0 = E_{enc}(\mathbf{x}) := \mathcal{T}_e(\tilde{\mathbf{x}}), \quad \mathbf{h}_{\mathbf{y}}^0 = E_{enc}(\mathbf{y}), \quad (4)$$

where $\tilde{\mathbf{x}}$ is the sum of token embedding and position embedding on $\mathbf{x}$, i.e., $\mathrm{WE}(\mathbf{x}) + \mathrm{PE}(\mathbf{x})$.

The decoder of the first Transformer layer is computed by

$$E_{dec}^{d_0} = \mathcal{T}_d(\tilde{\mathbf{y}}, \mathbf{h}_{\mathbf{x}}^0), \text{ where } \tilde{\mathbf{y}} = \mathrm{WE}(\mathbf{y}) + \mathrm{PE}(\mathbf{y}). \quad (5)$$

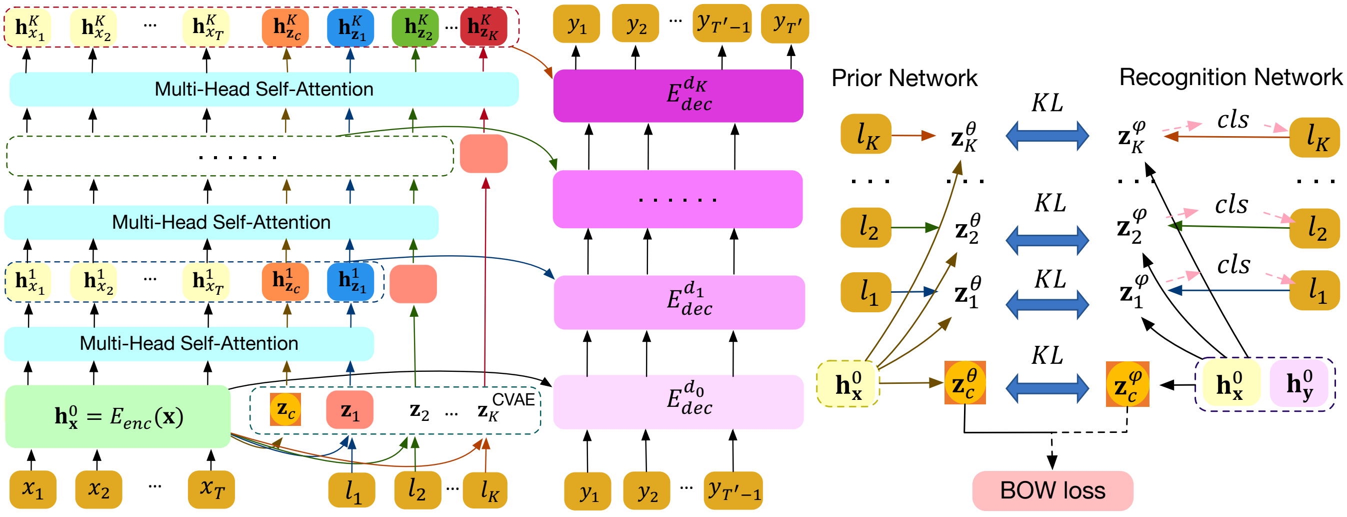
The corresponding CVAE variables at the i-th stage of the *recognition network* and the *prior network* are:

$$\mathbf{z}_c^i(\boldsymbol{\psi}) = \mathcal{C}(q_{\boldsymbol{\psi},c}^i, [\mathbf{h}_{\mathbf{x}}^0; \mathbf{h}_{\mathbf{y}}^0]), \quad \mathbf{z}_i(\boldsymbol{\psi}) = \mathcal{C}(q_{\boldsymbol{\psi}}^i, [\mathbf{h}_{\mathbf{x}}^0; \mathbf{h}_{\mathbf{y}}^0]), \quad (6)$$

$$\mathbf{z}_c^i(\boldsymbol{\theta}) = \mathcal{C}(p_{\boldsymbol{\theta},c}^i, \mathbf{h}_{\mathbf{x}}^0), \quad \mathbf{z}_i(\boldsymbol{\theta}) = \mathcal{C}(p_{\boldsymbol{\theta}}^i, \mathbf{h}_{\mathbf{x}}^0). \quad (7)$$

Obviously, the difference between the two networks lies in whether the multi-head attention network attends to the decoder $\mathbf{h}_{\mathbf{y}}^0$ or not. It is noted that $\mathbf{h}_{\mathbf{x}}^0$ and $\mathbf{h}_{\mathbf{y}}^0$, rather than $\mathbf{h}^{i-1}$, are applied to learn the parameters of both networks because $\mathbf{h}^{i-1}$ has absorbed the attribute information in all previous stages and may contaminate the original data.

We train PHED in the following progressive way:


$\mathbf{h}_{x_1}^K$ $\mathbf{h}_{x_2}^K$ ··· $\mathbf{h}_{x_T}^K$ $\mathbf{h}_{\mathbf{z}_c}^K$ $\mathbf{h}_{\mathbf{z}_1}^K$ $\mathbf{h}_{\mathbf{z}_2}^K$ ··· $\mathbf{h}_{\mathbf{z}_K}^K$
Multi-Head Self-Attention
· · · · · ·
Multi-Head Self-Attention
$\mathbf{h}_{x_1}^1$ $\mathbf{h}_{x_2}^1$ ··· $\mathbf{h}_{x_T}^1$ $\mathbf{h}_{\mathbf{z}_c}^1$ $\mathbf{h}_{\mathbf{z}_1}^1$
Multi-Head Self-Attention
$\mathbf{h}_{\mathbf{x}}^0 = E_{enc}(\mathbf{x})$
$\mathbf{z}_c$ $\mathbf{z}_1$ $\mathbf{z}_2$ ... $\mathbf{z}_K$ CVAE
x_1 x_2 ··· x_T
l_1 l_2 ··· l_K
y_1 y_2 ··· $y_{T'-1}$ $y_{T'}$
$E_{dec}^{d_K}$
· · · · · ·
$E_{dec}^{d_1}$
$E_{dec}^{d_0}$
y_1 y_2 ··· $y_{T'-1}$
Prior Network
Recognition Network
l_K $\mathbf{z}_K^\theta$ KL $\mathbf{z}_K^\varphi$ cls l_K
l_2 $\mathbf{z}_2^\theta$ KL $\mathbf{z}_2^\varphi$ cls l_2
l_1 $\mathbf{z}_1^\theta$ KL $\mathbf{z}_1^\varphi$ cls l_1
$\mathbf{h}_{\mathbf{x}}^0$ $\mathbf{z}_c^\theta$ KL $\mathbf{z}_c^\varphi$ $\mathbf{h}_{\mathbf{x}}^0$ $\mathbf{h}_{\mathbf{y}}^0$
BOW loss

(a) PHED (b) CVAE

Figure 2: The architecture of the PHED framework: Different colors on different variables distinguish the effect of attributes at the corresponding stage. For the variables and the training procedure, please see the main text.

1. The CVAE variables $\mathbf{z}_c^1$ and $\mathbf{z}_1$ are sampled from the *recognition network* defined in Eq. (6) by setting $i = 1$. Next, $\mathbf{h}^1$ and $E_{dec}^{d_1}$ are then computed by the newly stacked Transformer layer on the concatenation of $\mathbf{h}^0$, $\mathbf{z}_c^1$, and $\mathbf{z}_1$ (i.e., $\tilde{\mathbf{h}}^1 = [\mathbf{h}^0; \mathbf{z}_c^1; \mathbf{z}_1]$):

$$\mathbf{h}^1 = \mathcal{T}_e(\tilde{\mathbf{h}}^1), \quad E_{dec}^{d_1} = \mathcal{T}_d(E_{dec}^{d_0}, \tilde{\mathbf{h}}^1), \tag{8}$$

   We highlight two remarks: (1) The effect of the CVAE variables $\mathbf{z}_c$ and $\mathbf{z}_1$ is realized by the multi-head self-attention on $\tilde{\mathbf{h}}^1$. (2) The input of $\mathcal{T}_d$ is slightly different from the standard Transformer. That is, the self-attention in PHED is applied at the same stage, not from scratch. It can then enhance the impact of the CVAE variables at the corresponding stage.
2. At the i-th stage ($i \geq 2$), we fix the parameters learned at previous stages and sample $\mathbf{z}_c^i$ and $\mathbf{z}_i$ from the *recognition network* defined in Eq. (6) and compute $\mathbf{h}^i$ and $E_{dec}^{d_i}$ by a newly stacked Transformer layer:

$$\mathbf{h}^i = \mathcal{T}_e(\tilde{\mathbf{h}}^i), \quad E_{dec}^{d_i} = \mathcal{T}_d(E_{dec}^{d_{i-1}}, \tilde{\mathbf{h}}^i), \tag{9}$$

   where $\tilde{\mathbf{h}}^i = [\mathbf{h}^{i-1}; \mathbf{z}_i]$. Note that $\tilde{\mathbf{h}}^i$ does not include $\mathbf{z}_c^i$ because it has been absorbed in $\mathbf{h}^{i-1}$.
3. Step 2 continues until we reach the K-th stage. The parameters are learned by the **Multi-stage Training** procedure detained in Sec. 2.3.

By the above generation mechanism, we can derive the following theorem to compute the conditional probability:

Theorem 1. *Given the above defined notations, the conditional generation probability can be computed by*

$$\begin{aligned} &p(\mathbf{y}|\mathbf{x}, c = l_1 \ldots l_K) \\ =&p(\mathbf{y}|\mathbf{h}^K, l_K) \cdot \prod_{k=1}^{K} p(\mathbf{h}^k|\mathbf{h}^{k-1}, l_k) \cdot p(\mathbf{h}^0|\mathbf{x}), \end{aligned} \tag{10}$$

and the evidence lower bound at the i-th ($i \geq 1$) stage is

$$\log p_{\boldsymbol{\theta}}^i(\mathbf{y}|\mathbf{x}, l_1 \ldots l_i) \geq \underbrace{-L_{KL}^{c,i} - L_{KL}^i - L_M^i}_{L_{ELBO}}, \text{ where} \tag{11}$$

$$L_{KL}^{c,i} := D_{KL}(q_{\boldsymbol{\psi},c}^i(\mathbf{z}_c^i|\mathbf{h}^0, \mathbf{y}) \parallel p_{\boldsymbol{\theta},c}^i(\mathbf{z}_c^i|\mathbf{h}^0)), \tag{12}$$

$$L_{KL}^i := D_{KL}(q_{\boldsymbol{\psi}}^i(\mathbf{z}_i|\mathbf{h}^0, \mathbf{y}, l_i) \parallel p_{\boldsymbol{\theta}}^i(\mathbf{z}_i|\mathbf{h}^0, l_i)), \tag{13}$$

$$L_M^i := -E_{\mathbf{z}_c \sim q_{\boldsymbol{\psi},c}^i, \mathbf{z}_i \sim q_{\boldsymbol{\psi}}^i}[\log p_{\boldsymbol{\theta}}^i(\mathbf{y}|\mathbf{h}^{i-1}, l_i)]. \tag{14}$$

The proof is provided in the Appendix. Eq. (10) holds due to the variable dependency and the Markov chain on $\mathbf{h}^i$. Here, we only consider the Markov property and leave the variants of including more hidden states as a future work. Note that in Eq. (11), the derived ELBO (L_{ELBO}) consists of not only the expected log-likelihood estimator, L_M^i, but also two separated KL divergences, $L_{KL}^{c,i}$ and L_{KL}^i to control $\mathbf{z}_c^i$ and $\mathbf{z}_c$.

2.3 Losses and Training

Other than the derived KL divergence in Eq. (11), we need the following losses to constrain CVAE latent variables sampled from Eq. (6). First, $\mathbf{z}_c^i$ and $\mathbf{z}_i$ should be as dissimilar as possible. Moreover, to balance the effect of $\mathbf{z}_c$ and $\mathbf{z}_i$, we force their length to be nearly the same and yield the following loss:

$$L_{\mathbf{z}^i} = \frac{\mathbf{z}_c^{iT}\mathbf{z}_i}{\|\mathbf{z}_c^i\|\|\mathbf{z}_i\|} + \left(\|\mathbf{z}_c^i\| - \|\mathbf{z}_i\|\right)^2. \tag{15}$$

Second, we expect that $\mathbf{z}_c$ changes little across two consecutive stages and enforce it by minimizing the *Fréchet Inception Distance* (FID) [Heusel *et al.*, 2017]:

$$L_{\mathbf{z}_c^i} = \text{FID}(\mathbf{z}_c^{i-1}, \mathbf{z}_c^i). \tag{16}$$

This loss is also equivalent to minimizing the Wasserstein-2 distance on two isotropic Gaussian distributions, i.e., the sum of the difference of mean and standard deviation on two Gaussian distributions.

Third, to guarantee encoding meaningful information, we follow the idea of [Zhao *et al.*, 2017] to enforce the bag-of-word (BOW) loss on $\mathbf{z}_c^i$:

$$L_{\mathbf{z}_c^i}^{bow} = E_{\mathbf{z}_c^i \sim q_{\boldsymbol{\psi},c}^i}[\log p(\mathbf{y}_{bow}|\mathbf{h}^i, \mathbf{z}_c^i)], \tag{17}$$

where $\mathbf{y}_{bow}$ are the words in response $\mathbf{y}$ without order, and $p(\mathbf{y}_{bow}|\mathbf{h}^i, \mathbf{z}_c^i)$ is obtained by a single layer fully-connected network $\mathbf{h}^b = \mathbf{MLP}_b([\mathbf{h}^i; \mathbf{z}_c^i])$.

Fourth, the cross entropy loss is placed to guarantee the effect of the fed attribute:

$$L_{cls}^i = -\mathbf{y}_{l_i} \log\left(\mathbf{MLP}_i(\mathbf{z}_i)\right). \tag{18}$$

Multi-stage Training

We train PHED progressively: at the first stage, we estimate the model parameters by minimizing the following loss:

$$L_{stage}^1 = \lambda(L_{KL}^{c,1} + L_{KL}^1) + L_M^1 + L_{\mathbf{z}^1} + L_{\mathbf{z}_c^1}^{bow} + L_{cls}^1, \tag{19}$$

where λ is gradually increased from 0 to 1 via the annealing technique [Ke *et al.*, 2018] because L_M^1, $L_{\mathbf{z}_c^1}^{bow}$, and L_{cls}^1 are cross entropy losses with nearly the same scale while the effect of $L_{\mathbf{z}^1}$ is small as observed.

Next, at the i-th stage ($i \geq 2$), we freeze previously learned parameters and seek new parameters by minimizing

$$L_{stage}^i = \lambda(L_{KL}^{c,i} + L_{KL}^i) + L_M^i + L_{\mathbf{z}^i} + L_{\mathbf{z}_c^i}^{bow} + L_{cls}^i + L_{\mathbf{z}_c^i}, \tag{20}$$

where the loss $L_{\mathbf{z}_c^i}$ is specially included to guarantee the smoothness of the change of $\mathbf{z}_c^i$. The above minimization procedure continues until i reaches K.

After training PHED, given a message $\mathbf{x}$, we can then generate each type of responses with the associated controlled attribute at each stage. That is, we sample $\mathbf{z}_c^i$ and $\mathbf{z}_i$ from Eq. (7) and concatenate them with $\mathbf{h}^i$ to construct the input of Transformer at each stage, i.e., $\tilde{\mathbf{h}}^1 = [\mathbf{h}^0; \mathbf{z}_c^1; \mathbf{z}_1]$ for $E_{dec}^{d_1}$ as in Eq. (8) and $\tilde{\mathbf{h}}^i = [\mathbf{h}^{i-1}; \mathbf{z}_i]$ for $E_{dec}^{d_i}$ as in Eq. (9), where $i = 2, \dots, K$. Let $E_{dec,t}^k$ be the k-th stage decoder at the t-th step, we can generate the response by

$$\mathbf{y}_t^k \sim \mathrm{softmax}(E_{dec,t}^k W_o), \tag{21}$$

where $W_o \in \mathbb{R}^{H \times |V|}$ is the parameter shared with the embedding layers and $|V|$ is the vocabulary size.

3 Experiments

We conduct experiments to address the following questions: (1) What is the performance of PHED in both automatic and human evaluations? (2) What is the effect of the losses in PHED? (3) What are the generation results?

3.1 Data

The data is the short-text conversation dataset (**STC**) [Shang *et al.*, 2015], collected from Sina Weibo, a Chinese social platform. After setting the maximum number of characters in a response to 30, we obtain around 3.9 million dialog pairs and split them into the set of training, validation, and test with the ratio of 90%, 5%, and 5%, respectively. We pick three independent aspects of attributes, Emotion (**Emo.**), **Tone**, and length (**Len.**). The emotion aspect consists of six categories: angry (A), disgust (D), happy (H), like (L), sad (S), and others (O). The tone aspect considers three types: declarative (D), interrogative (I), and imperative (M). The emotion classifier and the tone classifier is trained as in [Zhou *et al.*, 2018; Ke *et al.*, 2018]. Based on the typical length generated by the baselines, we set the length of a response as long, denoted by L, when the number of characters is greater than 12 and others as short, denoted by S. Table 1 reports the data statistics.

	#. Pairs	min.	max.	avg.	#. Char.
Train$_{\mathrm{m}}$	3,542,103	13	59	38.0±10.6	10,261
Train$_{\mathrm{r}}$		5	30	28.9 ±10.5	6,603
Valid.$_{\mathrm{m}}$	196,783	13	59	38.1±10.6	5,693
Valid.$_{\mathrm{r}}$		9	30	28.9±10.6	6,064
Test$_{\mathrm{m}}$	196,783	13	59	38.1±10.6	5,714
Test$_{\mathrm{r}}$		7	30	28.9±10.5	6,126

	Type	Train	Test	Type	Train	Test
Emo.	A	4.2	4.2	D	23.0	23.3
	H	5.1	5.2	L	22.0	22.1
	S	10.8	10.8	O	34.8	34.6
Tone	D	61.5	61.6	I	18.0	17.9
	M	20.6	20.5			
Len.	L	58.0	58.1	S	42.0	41.9

Table 1: Statistics of the data.

3.2 Methods and Implementation

We compare our PHED with the following strong baselines: (1) **MMI-bidi** [Li *et al.*, 2016]: a popular Seq2Seq LSTM model applying the maximum mutual information objective to re-rank responses from beam search; (2) **SC-S2S** [Zhang *et al.*, 2018]: an Seq2Seq LSTM model with the specificity controlling; (3) **DCVAE** [Gao *et al.*, 2019]: a newly proposed Seq2Seq LSTM model with CVAE to generate responses through two-stage sampling. (4) **T2T** [Vaswani *et al.*, 2017]: a Transformer-based baseline, which is also the base of PHED without controllable attributes.

Our implementation is in PyTorch[1]. We apply the default setup for the baselines with the best tuned parameters on the highest BLEU score. Following [Zhang *et al.*, 2018], SC-S2S applies Jieba to construct a vocabulary dictionary in 40,000 most frequent Chinese words. The same as [Gao *et al.*, 2019], DCVAE constructs a vocabulary dictionary with 50,000 most frequent Chinese words and characters. MMI-bidi and T2T apply the same Chinese characters as the input to attain a dictionary with 9,500 Chinese characters out of total 10,549 characters extracted from the dataset, where the low-frequented 1,049 characters are denoted by <UNK>. Moreover, three additional special tokens, <EOS>, <BOS>, and <PAD> are introduced to denote the start and the end of a sentence, and the padding character, respectively. By this setting, we obtain a much smaller size of the dictionary in T2T and PHED, yielding a lighter effort to learn precise representations for the input tokens. For each Transformer block, we set the number of self-attention heads to 8 and the hidden size (H) to 512. T2T consists of 6 Transformer layers in both the encoder and the decoder, pre-trained on 40

[1] https://www.dropbox.com/s/1376kmhvuaxqe5h/PHED.zip?dl=0

Method	**Relevance**				**Diversity (%)**		**Human Evaluation**			**Len.**
	BLEU-1	BLEU-2	BLEU-3	BLEU-4	Dist. 1	Dist. 2	Quality	Good	Accept	**Avg.± Std.**
MMI-bidi	10.26	6.03	4.18	3.20	0.15	2.61	1.56±0.57	16%	48%	11.3±1.9
SC-S2S	9.63	5.92	3.87	2.95	$\underline{0.17}$	2.47	1.50±0.55	18%	48%	10.1±2.2
DCVAE	11.72	6.97	4.82	3.88	**0.18**	3.07	1.73±0.94	24%	54%	9.1±2.6
T2T	17.11	9.26	6.16	4.63	0.13	2.81	1.83±0.90	24%	56%	12.0±1.9
PHED$_{\mathrm{H}}$	16.75	9.03	6.00	4.43	0.14	3.72	2.03±0.90	**42%**	66%	12.9±3.1
PHED$_{\mathrm{S}}$	17.69	$\underline{9.69}$	$\underline{6.55}$	**4.95**	0.13	$\underline{4.26}$	1.98±0.89	39%	58%	14.4±3.8
PHED$_{\mathrm{HD}}$	17.04	9.17	6.08	4.51	0.14	3.74	$\underline{2.08\pm0.79}$	40%	$\underline{70\%}$	13.3±3.1
PHED$_{\mathrm{HI}}$	15.61	8.11	5.21	3.78	0.12	2.96	2.01±0.82	33%	54%	13.6±3.2
PHED$_{\mathrm{SD}}$	18.28	**9.82**	**6.57**	$\underline{4.94}$	0.13	$\underline{4.26}$	2.02±0.86	39%	56%	14.6±3.8
PHED$_{\mathrm{SI}}$	14.43	8.02	5.44	4.14	0.13	3.64	1.96±0.88	30%	61%	13.6±3.5
PHED$_{\mathrm{HDS}}$	14.26	7.56	4.95	3.64	0.15	3.52	2.01±0.96	$\underline{41\%}$	65%	11.2±1.5
PHED$_{\mathrm{HDL}}$	**18.69**	**9.82**	6.37	4.65	0.11	3.62	1.95±0.95	35%	64%	15.6±3.1
PHED$_{\mathrm{HIS}}$	12.48	6.47	4.12	2.99	0.14	2.92	**2.10±0.82**	31%	**71%**	11.3±2.1
PHED$_{\mathrm{HIL}}$	16.67	8.53	5.47	3.88	0.11	2.90	1.93±0.86	25%	59%	16.0±3.0
PHED$_{\mathrm{SDS}}$	14.47	8.07	5.41	4.21	0.16	**4.61**	1.87±1.06	35%	54%	11.3±1.9
PHED$_{\mathrm{SDL}}$	$\underline{18.06}$	9.53	6.28	4.65	0.15	4.06	1.86±0.99	31%	54%	16.8±3.3
PHED$_{\mathrm{SIS}}$	11.86	6.67	4.67	3.59	0.13	3.83	1.81±0.97	28%	53%	11.2±1.8
PHED$_{\mathrm{SIL}}$	15.88	8.57	5.69	4.52	0.11	3.53	1.77±0.89	22%	47%	16.4±3.2

Table 2: Evaluation results of all compared methods, where PHED reports 14 cases by selecting two typical types in each aspect of attributes. The best two results are highlighted by bold and an underline, respectively.

million message-response pairs. PHED applies CVAE on top of T2T. The size of CVAE latent variables is set to 128, i.e., $d_{\mathbf{z}_c} = d_{\mathbf{z}_i} = 128$. We stack two more Transformer layers for each aspect of attributes in the order of emotion, tone, and length and yield a total of 12 Transformer layers in PHED. The order of fed attributes is not optimized but follows the proved humans' effective learning procedure [Spiering and Ashby, 2008], from difficult tasks to easy ones. Except for the parameters in the initial Transformer block, the remaining parameters are initialized by the **Xavier** method and trained by **ADAM** with the learning rate 0.0001 and the batch size of 32. In the inference, we set the beam search size to 5. Under the above settings, we train PHED 10 epochs at each stage on a Tesla V100 GPU and cost about 51 hours.

3.3 Evaluation Metrics

We evaluate the models by: (1) **BLEU**: BLEU-n measures the average n-gram precision on a set of reference sentences. As DCVAE [Gao *et al.*, 2019], we set $n = 1, 2, 3, 4$. (2) **Dist. 1 & Dist. 2** [Li *et al.*, 2016]: the ratios of distinct unigrams and bigrams in the generated responses to the total generated unigrams and bigrams, measuring the diversity of the responses. For a fair comparison, all metrics are evaluated by the Chinese-character-level tokenization. (3) **Human evaluation**: Three expert labelers were recruited to evaluate the generated responses for 300 randomly selected posts based on the following 4-point criteria: 1) +3: the response is not only semantically relevant and grammatically correct, but also informative and interesting; 2) +2: the response is grammatically correct and can be used as a response to the utterance, but is too general (e.g., "OK"); 3) +1: the response is grammatically correct, but semantically irrelevant; 4) +0: the response contains mistakes (e.g., grammatical errors or <UNK>). Though it is different from the 3-point criteria in [Zhang *et al.*, 2018; Gao *et al.*, 2019], the 4-point criteria allows us to further distinguish meaningful responses from general and irrelevant responses. The values of the Fleiss' Kappa [Fleiss and Cohen, 1973] are great than 0.3 in all cases, which indicate the inter-rater consistency among three labelers.

3.4 Experimental Results

Table 2 reports 14 cases of PHED by selecting two typical types for each aspect of attributes and shows that

- **Relevance.** PHED attains the best two BLEU scores, which imply the generation relevance. Moreover, long responses can get significantly higher BLEU scores ($p < 0.01$ in t-test) than the short responses and the LSTM-based methods. Even short responses can attain competitive BLEU scores compared to the baselines.
- **Diversity.** PHED attains relatively lower scores in Dist. 1 because usually PHED generates longer responses than baselines, yielding a larger number of unigrams. In terms of Dist. 2, PHED attains significantly higher scores than the baselines (e.g., 4.61 vs. 3.07, around 50% gain). This again shows that PHED generates more diverse responses.
- **Length.** According to the results in the last column, PHED tends to generate longer responses, with more powerful expression ability than the baselines. Even when setting to generate short responses, PHED can generate longer responses than SC-S2S and DCVAE and similar length for MMI-bidi and T2T. When setting to generate long responses, PHED can generate 3 to 5 more characters for each response, significantly longer than the baselines.

The human evaluation results in the eighth to tenth columns of Table 2 are consistent with the automatic evaluation: (1) PHED generates significantly more relevant responses, where the values of 42% vs. 24% and 71% vs. 56% indicate that

	BLEU	Dist.	Emo.	Tone	Len. A.	Len.
T2T	4.63	2.81	–	–	–	12.0±1.9
PHED$_{\mathrm{S}}$	4.96	4.26	54.6	–	–	14.4±3.8
PHED$_{\mathrm{SI}}$	4.14	3.64	57.6	93.8	–	13.6±3.5
PHED$_{\mathrm{SIS}}$	3.59	3.83	60.3	89.3	84.0	11.2±1.8
$-L^i_{cls}$	5.57	5.67	9.7	21.6	48.6	13.6±2.8
$-L_{\mathbf{z}^i}-L^i_{cls}$	3.19	3.84	62.4	90.6	88.2	10.9±2.1
PHED$_{\mathrm{SIL}}$	4.52	3.53	59.6	90.8	94.9	16.4±3.2
$-L^i_{cls}$	5.57	5.68	9.8	21.6	51.4	10.9±2.8
$-L_{\mathbf{z}^i}-L^i_{cls}$	3.91	3.45	61.8	91.4	96.4	16.5±3.0

Table 3: Results on ablation study.

PHED generates more good (scoring over 3-point) and acceptable (scoring over 2-point) responses. (2) DCVAE and T2T are competitive while MMI-bidi and SC-S2S attaining much lower scores than them. Overall, PHED generates more good responses than baselines.

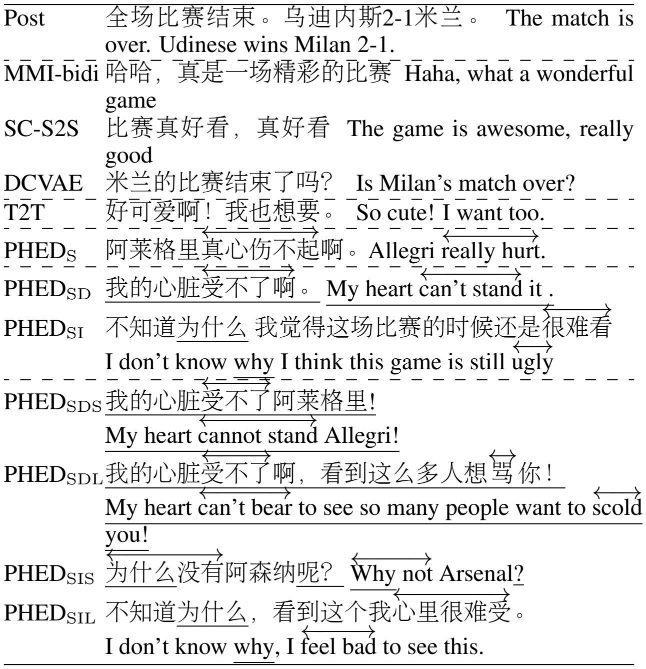
3.5 Ablation Study

We conduct ablation studies on PHED. The test shows that PHED produces similar results on all the combinations of attributes. In Table 3, we only report the scores of BLEU-4, Dist. 2, the accuracy of emotion (Emo.), tone (Tone), and the length prediction (Len. A.), and the average length (Len.) in five models: T2T, PHED$_{\mathrm{S}}$, PHED$_{\mathrm{SI}}$, PHED$_{\mathrm{SIS}}$ and PHED$_{\mathrm{SIL}}$; see more results in the Appendix.

The results show that: (1) The Dist. 2 scores in PHED on different fed attributes are all higher than that in T2T. This means that PHED attains more diverse responses than T2T. By examining more details, we can observe that the emotion accuracy increases slightly after adding more other attributes. The tone accuracy is around 90% while the length accuracy is at least 84%. (2) By removing the losses related to $\mathbf{z}_c$ in PHED, we obtain similar BLEU-4 and Dist. 2 scores to PHED with all restricted losses but attain slightly higher accuracy on all three aspects of attributes. The results imply that by removing $\mathbf{z}_c$, we can promote the effect of individual attributes and yield better attribute accuracy. (3) By examining the results of only removing classification loss (i.e., $-L^i_{cls}$), we observe that the corresponding attribute's accuracy slashes largely and becomes normal when removing both $L_{\mathbf{z}^i}$ and L^i_{cls}. The observation means that L^i_{cls} plays an essential role in controlling the attributes when PHED needs to satisfy other minimization requirements.

3.6 Case Study

Figure 3 illustrates a complete examination on the compared methods in Table 2. Our PHED clearly generates responses with the specific attributes progressively, including not only the corresponding emotion aspect, but also the exact tone and the length. For example, in the Happy emotion, PHED frequently generates "Congratulations" and "Haha". While in the Interrogative tone, it generates the related words, e.g., "What" or "Why". Moreover, by changing the required response length from short to long, more characters can be produced. For example, in Post#1, PHED$_{\mathrm{SDL}}$ generates similar words in the beginning to PHED$_{\mathrm{SDS}}$ but produces one more sentence to enrich the expression than PHED$_{\mathrm{SDS}}$. By examining all responses generated by PHED, the fidelity to the attribute(s) is clearly confirmed. In terms of the responses generated by MMI-bidi, SC-S2S, and DCVAE, they are usually shorter and blank. Responses generated by T2T are a little fluctuation and cannot deliver any attribute effect. More examples can be shown in the Appendix.



Post	全场比赛结束。乌迪内斯2-1米兰。 The match is over. Udinese wins Milan 2-1.
MMI-bidi	哈哈，真是一场精彩的比赛 Haha, what a wonderful game
SC-S2S	比赛真好看，真好看 The game is awesome, really good
DCVAE	米兰的比赛结束了吗？ Is Milan's match over?
T2T	好可爱啊！我也想要。 So cute! I want too.
PHED$_{\mathrm{S}}$	阿莱格里真心伤不起啊。Allegri really hurt.
PHED$_{\mathrm{SD}}$	我的心脏受不了啊。 My heart can't stand it .
PHED$_{\mathrm{SI}}$	不知道为什么 我觉得这场比赛的时候还是很难看 I don't know why I think this game is still ugly
PHED$_{\mathrm{SDS}}$	我的心脏受不了阿莱格里! My heart cannot stand Allegri!
PHED$_{\mathrm{SDL}}$	我的心脏受不了啊，看到这么多人想骂你！ My heart can't bear to see so many people want to scold you!
PHED$_{\mathrm{SIS}}$	为什么没有阿森纳呢？ Why not Arsenal?
PHED$_{\mathrm{SIL}}$	不知道为什么，看到这个我心里很难受。 I don't know why, I feel bad to see this.

Figure 3: Responses generated by the compared methods. In PHED, the words with ↔ on top and the words in underline of the generated responses indicate a high specificity to the first attribute and the second attribute, respectively.

4 Conclusion

We propose Progressively trained Hierarchical Encoder-Decoder to generate responses with multiple controllable attributes. By incorporating CVAE into Transformer, we represent the controlled attributes by a joint latent variable and further specific latent variables, where the CVAE latent variables are then coupled with the encoding information to generate responses. The model is then effectively trained progressively by maximizing the evidence lower bound while minimizing several subtly designed losses. Empirical results with both automatic and human evaluations demonstrate that PHED significantly outperforms the state-of-the-art neural generation models and is able to generate more diverse responses.

Several challenging but interesting directions will be considered in the future. First, we only exploit three aspects of attributes in one order of generation. It is practicable and useful to further take into account other aspects and other orders. Second, we only apply CVAE under the Markov assumption. It is interesting to explore more dependencies in CVAE. Third, the current task only focuses on open-domain response generation. It would be worthwhile to probe other tasks, e.g., text generation in the same spirit.

References

[Bi *et al.*, 2019] Wei Bi, Jun Gao, Xiaojiang Liu, and Shuming Shi. Fine-grained sentence functions for short-text conversation. In *ACL*, pages 3984–3993, 2019.

[Fleiss and Cohen, 1973] Joseph L. Fleiss and Jacob Cohen. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and psychological measurement*, 33(3):613– 619, 1973.

[Gao *et al.*, 2019] Jun Gao, Wei Bi, Xiaojiang Liu, Junhui Li, Guodong Zhou, and Shuming Shi. A discrete CVAE for response generation on short-text conversation. In *EMNLP-IJCNLP*, pages 1898–1908, 2019.

[Heusel *et al.*, 2017] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*, 2017.

[Hu *et al.*, 2017] Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. Toward controlled generation of text. In *ICML*, pages 1587–1596, 2017.

[Huang *et al.*, 2020] Minlie Huang, Xiaoyan Zhu, and Jianfeng Gao. Challenges in building intelligent open-domain dialog systems. *ACM Trans. Inf. Syst.*, 38(3):21:1–21:32, 2020.

[Jiao *et al.*, 2019] Wenxiang Jiao, Haiqin Yang, Irwin King, and Michael R. Lyu. Higru: Hierarchical gated recurrent units for utterance-level emotion recognition. In *NAACL-HLT*, pages 397–406, 2019.

[Ke *et al.*, 2018] Pei Ke, Jian Guan, Minlie Huang, and Xiaoyan Zhu. Generating informative responses with controlled sentence function. In *ACL*, pages 1499–1508, 2018.

[Kikuchi *et al.*, 2016] Yuta Kikuchi, Graham Neubig, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. Controlling output length in neural encoder-decoders. In *EMNLP*, pages 1328–1338, 2016.

[Lample *et al.*, 2019] Guillaume Lample, Sandeep Subramanian, Eric Michael Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. Multiple-attribute text rewriting. In *ICLR*, 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-HLT*, pages 110–119, 2016.

[Logeswaran *et al.*, 2018] Lajanugen Logeswaran, Honglak Lee, and Samy Bengio. Content preserving text generation with attribute controls. In *NeurIPS*, 2018.

[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*, pages 5370–5381, 2019.

[See *et al.*, 2019] Abigail See, Stephen Roller, Douwe Kiela, and Jason Weston. What makes a good conversation? how controllable attributes affect human judgments. In *NAACL-HLT*, pages 1702–1723, 2019.

[Serban *et al.*, 2016] Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C. Courville, and Joelle Pineau. Building end-to-end dialogue systems using generative hierarchical neural network models. In *AAAI*, 2016.

[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.

[Shang *et al.*, 2015] Lifeng Shang, Zhengdong Lu, and Hang Li. Neural responding machine for short-text conversation. In *ACL*, pages 1577–1586, 2015.

[Shao *et al.*, 2019] Zhihong Shao, Minlie Huang, Jiangtao Wen, Wenfei Xu, and Xiaoyan Zhu. Long and diverse text generation with planning-based hierarchical variational model. In *EMNLP-IJCNLP*, pages 3255–3266, 2019.

[Spiering and Ashby, 2008] B. J. Spiering and F. G. Ashby. Initial training with difficult items facilitates information integration, but not rule-based category learning. *Psychol Sci.*, 19(11):1169–1177, 2008.

[Vaswani *et al.*, 2017] 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*, pages 5998–6008, 2017.

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

[Xing *et al.*, 2018] Chen Xing, Yu Wu, Wei Wu, Yalou Huang, and Ming Zhou. Hierarchical recurrent attention network for response generation. In *AAAI*, pages 5610–5617, 2018.

[Xu *et al.*, 2019] Can Xu, Wei Wu, Chongyang Tao, Huang Hu, Matt Schuerman, and Ying Wang. Neural response generation with meta-words. In *ACL*, 2019.

[Zhang *et al.*, 2018] Ruqing Zhang, Jiafeng Guo, Yixing Fan, Yanyan Lan, Jun Xu, and Xueqi Cheng. Learning to control the specificity in neural response generation. In *ACL*, pages 1108–1117, 2018.

[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*, pages 654–664, 2017.

[Zheng *et al.*, 2020] Yinhe Zheng, Rongsheng Zhang, Minlie Huang, and Xiaoxi Mao. A pre-training based personalized dialogue generation model with persona-sparse data. In *AAAI*, pages 9693–9700, 2020.

[Zhou and Wang, 2018] Xianda Zhou and William Yang Wang. Mojitalk: Generating emotional responses at scale. In *ACL*, 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.