Emotional dialog generation via multiple classifiers based on a generative adversarial network

Wei CHEN, Xinmiao CHEN, Xiao SUN

School of Computer and Information, Hefei University of Technology, Hefei 230601, China

* Corresponding author, sunx@hfut.edu.cn
Received: 14 May 2020 Accepted: 22 June 2020

Citation: Wei CHEN, Xinmiao CHEN, Xiao SUN. Emotional dialog generation via multiple classifiers based on a generative adversarial network. Virtual Reality & Intelligent Hardware, 2021, 3(1): 18—32
DOI: 10.1016/j.vrih.2020.12.001

Abstract Background Human-machine dialog generation is an essential topic of research in the field of natural language processing. Generating high-quality, diverse, fluent, and emotional conversation is a challenging task. Based on continuing advancements in artificial intelligence and deep learning, new methods have come to the forefront in recent times. In particular, the end-to-end neural network model provides an extensible conversation generation framework that has the potential to enable machines to understand semantics and automatically generate responses. However, neural network models come with their own set of questions and challenges. The basic conversational model framework tends to produce universal, meaningless, and relatively "safe" answers. Methods Based on generative adversarial networks (GANs), a new emotional dialog generation framework called EMC-GAN is proposed in this study to address the task of emotional dialog generation. The proposed model comprises a generative and three discriminative models. The generator is based on the basic sequence-to-sequence (Seq2Seq) dialog generation model, and the aggregate discriminative model for the overall framework consists of a basic discriminative model, an emotion discriminative model, and a fluency discriminative model. The basic discriminative model distinguishes generated fake sentences from real sentences in the training corpus. The emotion discriminative model evaluates whether the emotion conveyed via the generated dialog agrees with a pre-specified emotion, and directs the generative model to generate dialogs that correspond to the category of the pre-specified emotion. Finally, the fluency discriminative model assigns a score to the fluency of the generated dialog and guides the generator to produce more fluent sentences. Results Based on the experimental results, this study confirms the superiority of the proposed model over similar existing models with respect to emotional accuracy, fluency, and consistency. Conclusions The proposed EMC-GAN model is capable of generating consistent, smooth, and fluent dialog that conveys pre-specified emotions, and exhibits better performance with respect to emotional accuracy, consistency, and fluency compared to its competitors.

Keywords Emotional dialog generation; Sequence-to-sequence model; Emotion classification; Generative adversarial networks; Multiple classifiers
1 Introduction

Technologies related to human-machine dialog are used in several types of products, including intelligent voice assistants and online customer services. With time, the requirements and expectations of maturity of human-machine dialog have drastically increased. Several related topics have been extensively researched, such as dialog systems with common sense knowledge, dialog systems with audio context, latent-variable task-oriented dialog systems, dialog systems combining texts and images, etc. For a detailed account of the relevant research, please refer to the survey conducted by Ma et al. Currently, dialog generation primarily includes three types of methods: rule-based systems, information retrieval systems, and generation-based systems. This study is based on the final approach. The Seq2Seq model has been extensively researched in the context of the machine translation problem, including its implementations based on Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM) and attention mechanism. Vinyals et al. were the first to apply the Seq2Seq structure to the problem of dialog generation.

The basic Seq2Seq model suffers from a fundamental drawback when it is used to generate conversation—the evaluation of the performance of the model is usually performed at the level of sentences. Since the initial application of Seq2Seq to this topic, several researchers have attempted to solve the problem by using generative adversarial networks (GANs), which have achieved great success in computer vision. Yu et al. proposed a more suitable framework to generate conversations based on a GAN called SeqGAN. By modeling the data generator as a stochastic policy in reinforcement learning, SeqGAN avoids the difference between generators by directly updating its gradient strategy. Li et al. proposed the use of adversarial training based on reinforcement learning for open-domain dialog generation. Cui et al. proposed the Dual Adversarial Learning (DAL) framework that improves both the diversity and overall quality of the generated responses.

People with high emotional intelligence quotients are capable of identifying and expressing their emotions, identifying the emotions of others, controlling their own emotions, and using feelings and emotions to spur adaptive behavior. It is equally essential to endow machines with emotion in the context of human-machine dialog. Ghosh et al. proposed an LSTM-based model to generate text endowed with emotion. Rashkin et al. introduced a new dataset with emotional annotations that was used to provide retrieval candidates or fine-tune the dialog model, leading to more empathetic responses. The Emotional Chatting Machine, proposed by Zhou and Zhang, is capable of generating dialogs with appropriate content as well as emotions. Wang et al. proposed the framework SentiGAN, which enables models to generate diverse, high-quality texts with specific sentiment labels via penalty mechanisms. In our previous work, we had presented a model based on LSTM, in which we altered the training corpus to solve the emotion factors in dialogs. In that model, the input was adapted to the original sentence and a sentence with an emotion label, and the sentence with the emotion label was used as the output.

In this study, we introduce a new emotional dialog generation model (EMC-GAN) based on a generative adversarial network. As it is difficult to express emotional features in dialog via the basic dialog generation model, we solve the problem by decomposing the emotional dialog generation task. Several different models are trained to generate dialogs endowed with different emotions. Each model focuses on creating one kind of emotional dialog. By incorporating such a modular structure, this method successfully excludes the interference and influence of other emotions during the generation of dialog with a specific emotion, thereby improving the accuracy of dialog generation endowed with pre-specified emotions. The proposed framework comprises a generative model and multiple discriminative models. The generative model is constructed based on the basic Seq2Seq dialog generation model, and the aggregate discriminative model of the framework comprises a basic discriminative model, an emotion discriminative...
model, and a fluency discriminative model. Together, they help to distinguish the generated text from the original text and to guide the generator to produce more fluent dialogs that convey specific emotions more accurately. The EMC-GAN model is capable of producing coherent, smooth, and fluent dialog expressing specific emotions, and performs better than existing systems with respect to emotional accuracy, coherence, and fluency (Figure 1).

### 2 Methods

The proposed emotional dialog generation framework, EMC-GAN, comprises one generative and three discriminative models. The generative model $G_e(Y|X; \theta_g)$, is a dialog generation model based on the basic Seq2Seq architecture. It generates coherent and fluent target sentences corresponding to a specified emotion category $e$, based on input source sentences. The aggregate discriminative model of EMC-GAN comprises a basic discriminative model $D_e(X,Y; \theta_e)$, an emotion discriminative model $D_{emot}(X,Y; \theta_e)$, and a fluency discriminative model $D_{fluency}(X,Y; \theta_e)$. The basic discriminative model is identical to the general dialog generation model, which is based on a generative adversarial network, and distinguishes generated fake sentences from real sentences in the training corpus. It also guides the generator to generate dialogs that are closer to human dialogs. The emotion discriminative model is a binary classifier of text sequences, which is capable of determining whether or not the emotion expressed by a generated dialog agrees with a specified emotion $e$. It provides a confidence probability for the emotion category of the input dialog to be identical to a pre-specified emotion category. The fluency discriminator assesses the fluency of input dialog and guides the generator to create more fluent dialog.

#### 2.1 Generative model-oriented emotional dialog

The goal of the generative model $G_e(Y|X; \theta_g)$, is to generate a target sequence corresponding to each input source sequence by endowing it with a pre-specified emotion $e$. $\theta_g$ is a parameter of the generative model. At each time-step $t$, $G_e(Y|X; \theta_g)$ produces a sentence $S_t = Y_{t<} = \{y^{<1>, y^{<2>, \cdots, y^{<t>}}\}$ where $y^{<t>}$ denotes a word token in the existing vocabulary. Eq. 1 and Eq. 2 present the penalty-based loss function:

$$V_c(S_t, y^{<t+1>}) = \lambda_1 \cdot V_c(S_t, y^{<t+1>}) + \lambda_2 \cdot V_{emot}(S_t, y^{<t+1>}) + \lambda_3 \cdot V_{fluency}(S_t, y^{<t+1>})$$ (1)

where $V_c(S_t, y^{<t+1>})$ denotes the total penalty score for the sentence sequence as calculated by multiple discriminative models, $V_c(S_t, y^{<t+1>})$ denotes the penalty score calculated by the basic discriminative
model, $D_e(X, Y; \theta_d)$, $V_{\text{emotion}}^{S_i}(S_i, y^{t+1})$ denotes the penalty score calculated by the emotional discriminative model, $D_e^{\text{emotion}}(X, Y; \theta_d)$, which reflects the emotion conveyed by the generated dialogs, $V_{\text{fluency}}^{S_i}(S_i, y^{t+1})$ denotes the penalty score calculated by the fluency discriminative model, $D_e^{\text{fluency}}(X, Y; \theta_d)$, which reflects the fluency of the sentences, and $\lambda_1 + \lambda_2 + \lambda_3 = 1$. In this study, we set $\lambda_1 = 0.5$ and $\lambda_2 = \lambda_3 = 0.25$.

Thus, the loss function $L(y^{t+1})$ is defined as follows based on the penalty scores:

$$L(y^{t+1}) = G_e(y^{t+1} | S_i; \theta_e) \cdot V_c(S_i, y^{t+1})$$

(2)

where $G_e(y^{t+1} | S_i; \theta_e)$ denotes the probability of choosing the $(t+1)^{th}$ word, which is dependent on the sequence $S_i$. The generative model reference penalty is defined as follows to minimize loss:

$$J_e(\theta_e) = E_{y \sim p_e} [L(y)] = \sum_{t=0}^{t+1} L(y^{t+1})$$

(3)

The penalty is calculated via the following formula:

$$V_c(S_i, y^{t+1}) = \begin{cases} \frac{1}{N} \sum_{i=1}^{N} (1 - D_e(X, Y, i; \theta_d)), & t < T, \\ 1 - D_e(X, Y, i; \theta_d), & t = T, \end{cases}$$

(4)

where $T_i$ denotes the maximum length of the target sequence, and $N$ denotes the size of the Monte Carlo search samples. The penalty score of the partial sequence is calculated by averaging those corresponding to multiple samples to reduce the loss caused by sampling.

### 2.1.1 The baseline model of dialog generation

The basic Seq2Seq model is used as a benchmark model in this study. This model uses the encoder-decoder network with deep LSTM units as the underlying architecture for dialog generation. Adding an effective attention mechanism to the model can help in the extraction of a greater amount of corresponding information between the source sentence and the target sentence. The overall structure of this model has been depicted in Figure 2. The dialog generation model proposed in this study shares its network structure with the Seq2Seq baseline model.

![Encoder-Decoder with attention mechanism](image)

Both the encoder and the decoder are implemented using LSTM. Corresponding to each time-step, a token of the source sequence is treated as the input to the encoder network. After the input is completed, the encoder generates a semantic vector $C$ corresponding to each time-step input, which represents the
input source sequence. The initial state of the decoder is determined by the generated semantic vector \( C \). The decoder decodes the semantic vector and outputs a token \( y^{t+1} \) corresponding to each time-step. Thus, we obtain an output sequence, \( Y(y^{<1>}, y^{<2>}, \ldots, y^{<T_r>}) \), corresponding to the input sequence \( X(x^{<1>}, x^{<2>}, \ldots, x^{<T_r>}) \) using the dialog generation framework based on the encoder-decoder network.

**2.1.2 Attention mechanism of the generative model**

The fundamental function of the attention mechanism is to compute the context vector. The context vector \( context^{t+1} \), directs the decoder \( y \), by highlighting the work tokens of the input sentence which should be focused upon by the decoder via the context vector. Figure 3 depicts the details of the calculation of the context vector in the attention mechanism. In order to save the state of the hidden layer of the source sequence, the Seq2Seq model with attention mechanism uses a bidirectional LSTM network\(^{(23)}\) to extract the hidden state of the source sequence at each time-step. The bidirectional LSTM network is given by Eq. 5.

\[
a^{<t>} = \begin{bmatrix} a^{<t>} \\ a^{<t>}
\end{bmatrix}
\]  

where \( a^{<t>} \) comprises two parts \( a^{<t>} \) and \( a^{<t>} \), representing the positive sequence feature and the reverse sequence feature, respectively. The two parts of the hidden state vector are calculated via the following formulae.

\[
a^{<t>} = \text{BiLSTM}^{fw}(a^{<t>}, x^{<t>})
\]

\[
a^{<t>} = \text{BiLSTM}^{bw}(a^{<t>}, x^{<t>})
\]

Besides providing the historical information of the sequence before time-step \( t \), the intermediate state vector \( e^{<t>} \), also provides the future information of the sequence. To distinguish the intermediate state vector of the decoder from that of the encoder, we denote the former by \( S^{<t>} \) at time-step \( t \). The intermediate state vector of the decoder at the previous time-step \( S^{<t-1>} \) is concatenated with the intermediate state vector of the encoder at each time-step.

\[
e^{<t>} = (S^{<t-1>}, a^{<t>})
\]

where, the vector \( e^{<t>} \) denotes the vector obtained by concatenating the intermediate state vector of the decoder corresponding to the \( t-1 \)th time-step and the intermediate state vector of the encoder at the \( t \)th time-step. As is evident from Eq. 9, the attention vector \( \alpha^{<t,t'>} \) represents the degree to which, at the time-step \( t \), the decoder focuses on the intermediate state vector \( a^{<t>} \) of the encoder at the time-step \( t' \). Further, as is evident from Eq. 10, the context vector, \( context^{<t>} \), is obtained by multiplying the attention vector, \( \alpha^{<t,t'>} \), by its corresponding hidden state vector \( a^{<t>} \), and then summing up the products over the range of time-
The achieved and the fake dialog.

\[ \alpha^{<i'>} = \frac{\exp(e^{<i'>})}{\sum_{j=1}^{T_e} \exp(e^{<j'>})} \]

\[ \text{context}^{<i'>} = \sum_{j=1}^{T_e} \alpha^{<i'>} a^{<j'>} \]

### 2.2 The discriminative model

The deep discriminative models implemented by convolutional neural networks (CNNs)\(^{[26]}\) and recursive convolutional neural networks (RCNNs)\(^{[25]}\) perform well in complex sequence classification tasks. We use CNN as the fundamental structure of the discriminative model proposed in this study. Moreover, a highway network\(^{[28]}\) is added to the discriminative model to improve its training speed. Both the emotion discriminative model and the fluency discriminative model are pre-trained models and do not participate in the adversarial training process of the model.

#### 2.2.1 Basic discriminative model

The text classification model based on CNN, originally proposed by Zhang and LeCun\(^{[29]}\), is used as the fundamental structure of the basic discriminative model \(D_e(X, Y; \theta'_e)\), which is employed to distinguish generated fake sentences from real sentences in the training dataset. The loss function of the basic discriminative model can be expressed as follows.

\[ J_{\theta'_e}(X, Y) = - \left[ y \cdot \log(p) + (1 - y) \cdot \log(1 - p) \right] \]

The adversarial training process of the proposed emotional dialog generation model EMC-GAN has been presented in Table 1.

#### 2.2.2 Emotion discriminative model

The emotion discriminative model \(D^{\text{emotion}}_e(X, Y; \theta'_e)\), guides the generative model to generate a target sentence with a specified emotion based on an input source sentence. The emotional discriminative model is capable of determining whether or not the emotion category of the input dialog agrees with the specified emotion category. The training process of the emotion discriminative model has been presented in Table 2. The real dialog with emotion category \(e\) is denoted by \(R = \{ \text{Dialog}_{<1>} \}\) and the fake dialog with emotion category is denoted by \(F = \{ \text{Dialog}_{<1>}, \text{Dialog}_{<2>}, \ldots, \text{Dialog}_{<n>} \}\). The emotion discriminator is used to discriminate between the real dialog \(R\) and the fake dialog \(F\) and assign a confidence probability to the event that the input dialog is identical to the real dialog. This model is similar to the basic discriminative model and is trained in advance. The achieved emotional accuracies pertaining to the different.

| Table 1 | The adversarial training process of EMC-GAN |
|---------|--------------------------------------------|
| **Input**: \(G_e(\{X, Y; \theta'_e\})\), \(D_e(X, Y; \theta'_e), D^{\text{emotion}}_e(X, Y; \theta'_e), D^{\text{fluency}}_e(X, Y; \theta'_e)\); |
| **Output**: Trained Dialog Generator: \(G_e(\{X, Y; \theta'_e\})\) |
| **Algorithm 1** Adversarial training of the model |
| 1: Initialize \(G_e(\{X, Y; \theta'_e\})\) and \(D_e(X, Y; \theta'_e)\) with random weights; |
| 2: Pre-train \(G_e(\{X, Y; \theta'_e\})\) using MLE on Train Data \(R\{X, Y\}\); |
| 3: Generate fake dialogs \(F\{X, Y\}\) using \(G_e(\{X, Y; \theta'_e\})\); |
| 4: Pre-train \(D_e(X, Y; \theta'_e)\) using \(R\{X, Y\}\); |
| 5: while model not converges do |
| 6: for each generative step do |
| 7: Generate fake dialog \(F\) using \(G_e(\{X, Y; \theta'_e\})\); |
| 8: Calculate penalty \(V_e\) using Eq. 1 |
| 9: Update \(G_e(\{X, Y; \theta'_e\})\) using Eq. 3 |
| 10: end |
| 11: for each discriminative step do |
| 12: Generate fake dialog \(F\) using \(G_e(\{X, Y; \theta'_e\})\); |
| 13: Update \(D_e(X, Y; \theta'_e)\) using \(R\{X, Y\}\) and Eq. 11 |
| 14: end |
| 15: end |
| 16: return |
categories range within 70%–85%, and experimental results demonstrate that this is sufficient to guide the generative model to generate sentences endowed with adequately accurate emotions.

### 2.2.3 Fluency discriminative model

The sentence fluency evaluation algorithm proposed in this study is based on the sentence fluency evaluation method proposed by Liu[30]. The training process of sentence fluency evaluation has been presented in Table 3, and the algorithm employs the N-gram statistical language model[31], which uses the transition probabilities of three tuples to measure the fluency of the entire sentence. Initially, we count the number of binary tuples in the dialogs present in the dataset and the corresponding frequencies of occurrence, save the results in n\_gram2\_count, adopt the binary tuple as the key of the dictionary, and take the occurrence frequency of the binary tuple as the value. Then, all the ternary tuples and their corresponding occurrence frequencies are calculated in a similar manner, and the results are saved in n\_gram3\_count. The transition probabilities of all ternary tuples are calculated using the dictionaries n\_gram2\_count and n\_gram3\_count, as given below.

#### Table 3 The fluency score of dialogs as obtained via the fluency discriminative model

**Algorithm 3** Calculate fluency score

**Input:** corpus, Sentence to be evaluated;

**Output:** the fluency score of the input sentence;

1. Count the number of Bi-gram of dialog corpus
2. Bi-gram count dicn\_gram2\_count={"[x\_s,x\_j]" : the number of [x\_s,x\_j] in corpus}
3. Count the number of Tri-gram of dialog corpus.
4. Tri-gram count dicn\_gram3\_count={"[x\_s,x\_j,x\_i]" : the number of [x\_s,x\_j,x\_i] in corpus}
5. Calculate Tri-gram transition probability by Eq. 12
6. Tri-gram transition probability dicn\_gram3\_prob={"[x\_s,x\_j,x\_i]" : p(x\_s,x\_j,x\_i)}
7. Sort the n\_gram3\_prob by transfer probability, and sorted probability saved to list sorted_n\_gram3\_prob
8. The size of sorted probability list is: size = len(sorted_n\_gram3\_prob)
9. reward\_prob = sorted_n\_gram3\_prob[ int(size*0.4) ]
10. penalty\_prob = sorted_n\_gram3\_prob[ int(size*(1 - 0.2) )]
11. p(x\_i) = p(x\_s,x\_i,...,x\_s+2)
12. fluency(X) = fluency( [x\_s,x\_j,...,x\_n]) = 0
13. if T\_r < 3 then
14. return fluency(X)
15. for i = 1 to m - 2 do
16. if p(x\_i) > reward\_prob then
17. fluency(X) += p(x\_i)/reward\_prob
18. else if p(x\_i) < penalty\_prob then
19. fluency(X) -= p(x\_i)/penalty\_prob
20. end
21. fluency(X) = fluency(X) / (m - 2)
22. return fluency(X)
\[ p(x_i, x_j, x_k) = p(x_j|x_i, x_k) = \frac{\text{count}(x_i, x_j, x_k)}{\text{count}(x_i, x_j)} \]

where, \(x_i, x_j, x_k\) denote adjacent words in the sentence. The calculated result is saved in \(n\_gram3\_prob\).

Finally, the ternary tuples are sorted in descending order of corresponding transition probabilities and the result is saved to the list \(sorted\_n\_gram3\_prob\). In general, sentences corresponding to higher transition probabilities of n-gram tuples are more fluent. The two transition probabilities, \(reward\_prob\) and \(penalty\_prob\), are used to decide whether or not a generated sentence is fluent. The first 40\% ternary tuples in \(sorted\_n\_gram3\_prob\) are smoother than the rest and the constructions of the last 20\% are considerably more awkward. \(reward\_prob\) denotes the minimum transition probability in the first 40\% of the tuples, and \(penalty\_prob\) denotes the maximum transition probability in the last 20\% of the tuples.

During the evaluation of the fluency of a sentence \(X = \{x_1, x_2, \cdots, x_n\}\), an initial fluency score of 0 is assigned to \(\text{fluency}(X)\). Then, all the ternary tuples are traversed. If the length of the sentence is less than 3, the algorithm directly sets its fluency score to 0 because we do not expect the model to use a short sentence as a response to the input source sentence. If the transition probability of the current ternary tuple is higher than \(reward\_prob\), it is implied that the ternary tuple is relatively smooth. In this case, the current fluency score \(\text{fluency}(X)\) adds the ratio of the transition probabilities to \(reward\_prob\). If the transition probability of current ternary tuple is less than \(penalty\_prob\), it is implied that the current ternary tuple is relatively awkward. In this case, the current fluency score \(\text{fluency}(X)\) subtracts the ratio of transition probabilities from \(penalty\_prob\). If the binary tuple corresponding to the ternary tuple does not exist, the ternary tuple is assigned a negligible transition probability (0.02) during the calculation of fluency. Finally, the computed fluency score \(\text{fluency}(X)\) is divided by the number of ternary tuples in the sentence to obtain the final result.

3 Experiments

During the process of emotional dialog generation, the generative model generates a target sentence corresponding to each input source sentence and a specified emotion category. It is essential for the resulting sentence sequence to be consistent, fluent, and pertain to the specified emotion category.

3.2 Datasets

The dialog dataset comprises a series of dialog pairs with emotion category labels \(\{X, Y\}\), where \(X = \{e_s, x_1^{s, \beta}, x_2^{s, \beta}, \cdots, x_T^{s, \gamma}\}\) is the source sentence sequence, \(Y = \{e_t, y_1^{t, \beta}, y_2^{t, \beta}, \cdots, y_T^{t, \gamma}\}\) is the target sentence sequence (dialog response), and \(e_s\) and \(e_t\) are the emotion category labels of the source sentence and the target sentence, respectively. The generative model intends to produce a target sentence with the specified emotion corresponding to each input source sentence with any emotion. To construct the corresponding datasets for the different emotion generative models, we divide the dataset into multiple sub-datasets based on the emotion conveyed by the target sentence. All target sentence sequences belonging to the same dataset share the identical emotion label \(e_t\).

NLPCC Weibo (NLPW): This dataset is constructed based on the conversations extracted from Weibo comments and comprises 1119200 dialog pairs with six emotion category labels (anger, disgust, happiness, liking, sadness, and other), similar to those used in our previous study\(^{23}\).

Xiaohuangji (XHJ): This dataset comprises 454130 dialog pairs in aggregate. However, its corpus does not include corresponding emotion labels. We use the open-source natural language processing tool
HanLP[32], to train an emotion classification model, which is essentially a naïve Bayes classifier trained on the NLPW dataset. The emotion classification model is capable of classifying sentences into the six emotion categories in the NLPW dataset.

The frequency distribution of the two datasets, NLPW and XHJ, with respect to varying emotion categories has been depicted in Figure 4. Table 4 presents the emotion distribution for sub-datasets corresponding to different emotions.

### Table 4 Emotion distribution in sub-datasets

| Dataset | Source Emotion | Target Emotion |
|---------|----------------|----------------|
|         | Other          | Liking | Sadness | Disgust | Anger | Happiness |
| NLPW    | Other          | 0.419  | 0.271   | 0.290   | 0.301 | 0.291      | 0.276     |
|         | Liking         | 0.187  | 0.381   | 0.182   | 0.180 | 0.159      | 0.276     |
|         | Sadness        | 0.095  | 0.083   | 0.212   | 0.098 | 0.111      | 0.099     |
|         | Disgust        | 0.149  | 0.110   | 0.145   | 0.259 | 0.199      | 0.120     |
|         | Anger          | 0.065  | 0.040   | 0.064   | 0.080 | 0.147      | 0.057     |
|         | Happiness      | 0.085  | 0.115   | 0.108   | 0.082 | 0.094      | 0.171     |
| XHJ     | Other          | 0.450  | 0.393   | 0.377   | 0.388 | 0.374      | 0.382     |
|         | Liking         | 0.132  | 0.227   | 0.119   | 0.124 | 0.109      | 0.144     |
|         | Sadness        | 0.087  | 0.079   | 0.172   | 0.080 | 0.072      | 0.106     |
|         | Disgust        | 0.173  | 0.155   | 0.170   | 0.246 | 0.183      | 0.160     |
|         | Anger          | 0.102  | 0.091   | 0.109   | 0.110 | 0.212      | 0.101     |
|         | Happiness      | 0.056  | 0.053   | 0.053   | 0.052 | 0.049      | 0.108     |

### 3.4 Experimental setup

The basic Seq2Seq dialog generation model is used as a benchmark model during experimental comparison. The performances of the hybrid neural network-based emotional dialog generation model EHMCG[23], and the emotional dialog generation model, EM-SeqGAN[33], are compared with that of the proposed EMC-GAN model. This study analyzes and evaluates the dialogs generated by the three different models primarily based on three indicators: emotional accuracy, coherence, and fluency. Tensorflow[33] is used during the construction of our model. During the calculation of the penalty score based on the discriminator coefficient, $\lambda$, as defined in Eq.1, the ratio between $\lambda_1$, $\lambda_2$, and $\lambda_3$ is taken to be 2:1:1. This adequately constrains the weights of the three evaluation parameters and appropriately guides the training of the generative model. The training iterations of the generative model and the discriminative model are taken to be 5 and 10, respectively.

### 3.5 Evaluation of emotional accuracy

Following the generation of dialog, we annotate them with the corresponding categories of emotion via the emotion classifier used in HanLP. If the emotion category of the generated sentence is identical to that of the target sentence, the emotion category of the generated sentence is considered to conform to the expectation.

Table 5 depicts the emotional accuracies of dialogs generated by the different dialog generation models.
Compared to other models, the proposed EMC-GAN model exhibits the highest emotional accuracy corresponding to each sub-dataset pertaining to each emotion. In the case of the NLPW dataset, the EMC-GAN model exhibits high emotional accuracies corresponding to the emotions "Other", "Liking", "Sadness", and "Disgust", which lie in the range of 0.588–0.740, while its emotional accuracy corresponding to the emotions "Anger" and "Happiness" were observed to be only 0.392 and 0.236, respectively. In the case of the XHJ dataset, the EMC-GAN model exhibits higher emotional accuracy corresponding to every emotion, and the accuracies lie in the range 0.701–0.870. Compared to other emotion categories, the emotion "Other" consistently corresponds to the highest emotional accuracy on both datasets, and this may be attributed to the fact that it corresponds to the highest number of training data. As the emotion "Other" represents any emotion that is distinct from the listed ones; the emotion classification model tends to judge the emotion category of most input sentences to be "Other".

| Dataset | Model       | Other | Liking | Sadness | Disgust | Anger | Happiness |
|---------|-------------|-------|--------|---------|---------|-------|-----------|
| NLPW    | Seq2Seq     | 0.286 | 0.121  | 0.089   | 0.128   | 0.212 | 0.191     |
|         | EHMCG       | 0.421 | 0.354  | 0.374   | 0.289   | 0.211 | 0.195     |
|         | EM-SeqGAN   | 0.572 | 0.487  | 0.548   | 0.376   | 0.295 | 0.201     |
|         | EMC-GAN     | 0.740 | 0.687  | 0.723   | 0.588   | 0.392 | 0.236     |
| XHJ     | Seq2Seq     | 0.293 | 0.176  | 0.064   | 0.227   | 0.177 | 0.094     |
|         | EHMCG       | 0.563 | 0.379  | 0.374   | 0.458   | 0.385 | 0.295     |
|         | EM-SeqGAN   | 0.647 | 0.563  | 0.487   | 0.567   | 0.426 | 0.375     |
|         | EMC-GAN     | 0.870 | 0.794  | 0.761   | 0.732   | 0.765 | 0.701     |

### 3.6 Evaluation of coherence

One of the most essential parameters in the evaluation of the performance of a dialog generation model is the coherence of the generated dialog, which represents whether or not it is in consonance with the context of the source sentence. Currently, satisfactory models capable of adequately evaluating the coherence of generated text are unavailable. Therefore, we resort to manual judgment to assess its coherence. The options used for evaluation and the corresponding evaluation scores have been summarized in Table 6. The evaluation scores range from 1 to 5 and higher evaluation scores correspond to higher coherence of the dialog.

| Option  | Very Good | Good | Normal | Bad | Very Bad |
|---------|-----------|------|--------|-----|----------|
| Score   | 5         | 4    | 3      | 2   | 1        |

The coherence evaluation scores of dialogs generated by the different generation models have been depicted in Table 7. The proposed EMC-GAN model exhibits higher coherence evaluation scores corresponding to all emotion categories for both datasets compared to the other models. Further, EMC-GAN exhibits higher coherence evaluation scores on the XHJ dataset than on the NLPW dataset. The generated dialog texts corresponding to the emotions "Other", "Liking" and "Sadness" reflected higher coherence evaluation scores for both datasets. Notably, coherence evaluation scores of 3.407 and 3.180 were achieved in the "Other" and "Sadness" categories, respectively, by EMC-GAN. This indicates that the dialogs generated by the proposed model exhibit satisfactory coherence.
3.7 Evaluation of fluency

Besides coherence, fluency of the generated text is also an essential parameter in the evaluation of the performance of a dialog generation model. Fluency reflects the text production capability of the generator. The fluency discriminative model evaluates the fluency of generated dialogs, and its underlying algorithm has been outlined in Algorithm 3. In addition, to improve the accuracy of the fluency evaluation, we also adopt a manual method.

A fluency score is assigned to each generated sentence by the fluency discriminative model. As is evident from the data presented in Table 8, the proposed EMC-GAN model exhibits higher fluency scores compared to other models corresponding to each dataset and emotion category. In the case of the NLPW dataset, the proposed model exhibits higher fluency scores corresponding to the emotions "Sadness" and "Anger", and the fluency score obtained corresponding to the emotion "Other" is relatively low. The generated dialogs in the XHJ dataset corresponded to higher fluency scores than those of the NLPW dataset, and the fluency of the sentences was observed to improve palpably. In the case of the XHJ dataset, the fluency score corresponding to the emotion "Other" was also observed to be relatively low.

Table 7  The coherence evaluation scores of generated dialog

| Dataset | Model    | Other  | Liking | Sadness | Disgust | Anger  | Happiness |
|---------|----------|--------|--------|---------|---------|--------|------------|
| NLPW    | Seq2Seq  | 1.306  | 1.067  | 1.451   | 1.085   | 1.181  | 1.051      |
|         | EHMCG    | 1.403  | 1.192  | 1.387   | 1.236   | 1.335  | 1.096      |
|         | EM-SeqGAN| 1.732  | 1.563  | 1.734   | 1.522   | 1.403  | 1.225      |
|         | EMC-GAN  | 2.277  | 2.875  | 2.115   | 1.881   | 1.972  | 1.245      |
| XHJ     | Seq2Seq  | 1.127  | 1.361  | 1.229   | 1.111   | 1.147  | 1.263      |
|         | EHMCG    | 1.256  | 1.452  | 1.248   | 1.324   | 1.223  | 1.371      |
|         | EM-SeqGAN| 1.820  | 1.726  | 1.339   | 1.514   | 1.330  | 1.207      |
|         | EMC-GAN  | 3.407  | 2.542  | 3.180   | 1.931   | 1.561  | 2.255      |

Table 8  The fluency score of generated dialog evaluated by algorithm 3

| Dataset | Model    | Fluency Score |
|---------|----------|---------------|
|         |          | Other | Liking | Sadness | Disgust | Anger  | Happiness |
| NLPW    | Seq2Seq  | -0.189| -0.193| -0.192  | -0.193  | -0.192 | -0.194     |
|         | EHMCG    | 0.405 | 0.727  | 1.133   | 0.526   | 1.238  | 0.943      |
|         | EM-SeqGAN| 0.586 | 0.875  | 1.875   | 1.034   | 1.337  | 1.237      |
|         | EMC-GAN  | 0.854 | 1.710  | 2.617   | 1.512   | 2.498  | 1.706      |
| XHJ     | Seq2Seq  | -0.124| -0.123 | -0.124  | -0.125  | -0.124 | -0.123     |
|         | EHMCG    | 3.356 | 5.528  | 7.526   | 6.776   | 5.882  | 4.581      |
|         | EM-SeqGAN| 4.652 | 7.238  | 8.832   | 7.774   | 6.237  | 6.735      |
|         | EMC-GAN  | 6.300 | 9.239  | 11.33   | 10.33   | 10.26  | 11.26      |

Table 9 depicts the fluency scores of the different generation models as assigned by human judgment. These scores are observed to be similar to the coherence evaluation score. Compared to other models, the EMC-GAN model exhibits higher fluency corresponding to each dataset and emotion category. Further, EMC-GAN exhibits a higher fluency score corresponding to each emotion of the XHJ dataset than those in the NLPW dataset. On the XHJ dataset, EMC-GAN exhibits a highest fluency evaluation score of 4.480 corresponding to the emotion "Sadness", and the fluency evaluation score corresponding to "Disgust" and "Anger" are observed to be relatively low at 2.835 and 2.960, respectively.
The proposed model is observed to perform better in terms of emotional and content consistency, they do not fare well consequently and,

It is apparent that the responses generated by Seq and EMC-GAN. The column entitled 'Emotion' depicts the pre-specified emotion category that the dialogs were meant to be generated in. It is apparent that the responses generated by Seq are poorer in terms of emotional and content consistency, and, consequently, they do not fare well with respect to the various evaluation parameters. The proposed model is observed to perform better in terms of emotional accuracy, coherence, and fluency.

### Table 9  The fluency score of generated dialog evaluated by human

| Dataset | Model    | Other | Liking | Sadness | Disgust | Anger | Happiness |
|---------|----------|-------|--------|---------|---------|-------|------------|
| NLPW    | Seq2Seq  | 1.193 | 1.267  | 1.251   | 1.202   | 1.114 | 1.351      |
|         | EHMC-GN  | 1.253 | 1.196  | 1.325   | 1.269   | 1.156 | 1.183      |
|         | EM-SeqGAN| 2.013 | 2.162  | 2.067   | 1.849   | 1.758 | 1.657      |
|         | EMC-GAN  | 2.424 | 2.875  | 2.365   | 2.476   | 2.272 | 1.984      |
| XHJ     | Seq2Seq  | 1.287 | 1.111  | 1.567   | 1.311   | 1.265 | 1.187      |
|         | EHMC-GN  | 1.257 | 1.284  | 1.732   | 1.455   | 1.325 | 1.173      |
|         | EM-SeqGAN| 2.471 | 2.648  | 2.741   | 1.846   | 1.775 | 2.659      |
|         | EMC-GAN  | 3.717 | 3.760  | 4.480   | 2.835   | 2.960 | 3.326      |

### 3.8 Analysis of results

The errors in emotional accuracy revealed by this experiment can be primarily attributed to mistakes in the datasets and on the part of the emotion classification model. As the accuracy of emotion classification on the NLPW dataset is observed to be 64%, some errors were caused in the emotion category of the generated dialog. Since dialogs in the XHJ dataset are not equipped with corresponding emotion category labels, the HanLP tool is used to train an emotion classifier by using the NLPW dataset as a training corpus and, thereby, add emotion labels to dialogs in the XHJ dataset. Further, to reduce the influence of the emotion classification model on the emotion evaluation for dialogs, we use the same emotion classification model to classify all the generated texts.

Based on the experimental results, it can be concluded that the proposed model performs better on the XHJ dataset than on the NLPW dataset. The sentence quality of dialogs in the NLPW dataset were generally assessed to be poor. The lengths of the sentences in the NLPW dataset are observed to be shorter than those in usual conversations, and sentences are more awkward, which may be attributed to the irregularity of comments in Weibo. In the NLPW dataset, a significant number of binary and ternary tuples occurred with singular frequency, which is not conducive to the construction of the language model, as it makes it more difficult for the dialog generation model to achieve convergence. Analysis of dialogs generated by the baseline model revealed that, in this case, the majority of the generated sentences are sequences of random words, and the fluency of the generated sentences is particularly poor. As is evident from Table 8, the dialogs generated by the Seq2Seq model exhibit lower fluency scores, and the fluency scores of sentences corresponding to different emotions are roughly identical. This can be attributed to the fact that, during the process of fluency evaluation, we assign a negligible transition probability (0.02) to ternary tuples when the corresponding binary tuple does not exist. Thus, as the fluency of the generated dialogs are extremely poor, the transition probabilities of most ternary tuples are close to the minimum value.

Certain samples of generated dialog were obtained from the test set. Table 10 depicts the dialogs generated by Seq2Seq and EMC-GAN. The column entitled 'Emotion' depicts the pre-specified emotion category that the dialogs were meant to be generated in. It is apparent that the responses generated by Seq2Seq are poorer in terms of emotional and content consistency, and, consequently, they do not fare well with respect to the various evaluation parameters. The proposed model is observed to perform better in terms of emotional accuracy, coherence, and fluency.
Conclusion

In this study, a new emotional dialog generation framework (EMC-GAN) was proposed, which uses multiple classifiers to generate better dialogs with respect to various evaluation parameters. The generative model generates a target sentence for each input source sentence. The basic discriminative model distinguishes generated fake sentences from real sentences in the training dataset. The emotion discriminative model evaluates whether the emotion of the generated dialog agrees with a pre-specified emotion. Finally, the fluency discriminative model assesses the fluency of the input dialog, and assigns a fluency score to it. Based on the experimental results, it was concluded that EMC-GAN is capable of generating dialogs with pre-specified emotions. Compared to other models, the dialogs generated by EMC-GAN were observed to be more fluent and smoother. However, the accuracy of the emotion classifier should be improved to obtain more realistic dialogs. Further, other features of sentences (such as novelty and variability) need to be considered to make the final dialog more fluent and natural.

Declaration of competing interest
We declare that we have no conflict of interest.

References

1 Young T, Cambria E, Chaturvedi I, Zhou H, Biswas S, Huang M. Augmenting end-to-end dialogue systems with
commonsense knowledge. Thirty-Second AAAI Conference on Artificial Intelligence. New Orleans, Louisiana, USA, 2018, 4970–4977
2 Young T, Pandelea V, Poria S, Cambria E. Dialogue systems with audio context. Neurocomputing, 2020, 388: 102–109
DOI:10.1016/j.neucom.2019.12.126
3 Xu H T, Peng H Y, Xie H R, Cambria E, Zhou L Y, Zheng W G. End-to-End latent-variable task-oriented dialogue system with exact log-likelihood optimization. World Wide Web, 2020, 23(3): 1989–2002
DOI:10.1007/s11280-019-00688-8
4 Zhang Z, Liao L Z, Huang M L, Zhu X Y, Chua T S. Neural multimodal belief tracker with adaptive attention for dialogue systems. In: The World Wide Web Conference on-WWW’19. San Francisco, CA, USA, ACM Press, 2019, 2401–2412
DOI:10.1145/3308558.3313598
5 Ma Y K, Nguyen K L, Xing F Z, Cambria E. A survey on empathetic dialogue systems. Information Fusion, 2020
DOI:10.1016/j.inffus.2020.06.011
6 Paschke A, Boley H. Rule responder: rule-based agents for the semantic-pragmatic web. International Journal on Artificial Intelligence Tools, 2011, 20(6): 1043–1081
DOI:10.1142/s021821310100528
7 Xu M H, Li P J, Yang H R, Ren P J, Ren Z C, Chen Z M, Ma J. A neural topical expansion framework for unstructured persona-oriented dialogue generation. 2020
8 Cho K, van Merrienboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). Doha, Qatar, Association for Computational Linguistics, 2014, 1724–1734
DOI:10.3115/v1/d14-1179
9 Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks. 2014
10 Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. 2014
11 Vinyals O, Le Q. A neural conversational model. 2015
12 Goodfellow I J, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y. Generative adversarial networks. 2014
13 Yu L, Zhang W, Wang J, Yu Y. SeqGAN: sequence generative adversarial nets with policy gradient. In: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence. San Francisco, California, AAAI, 2017, 2852–2858
14 Kaelbling L P, Littman M L, Moore A W. Reinforcement learning: a survey. Journal of Artificial Intelligence Research, 1996, 4(1): 237–285
DOI:10.1613/jair.301
15 Sutton R S, McAllester D A, Singh S P, Mansour Y. Policy gradient methods for reinforcement learning with function approximation. In: Advances in Neural Information Processing Systems 12, NIPS Conference. Denver, Colorado, USA, 1999, 1057–1063
16 Li J W, Monroe W, Shi T L, Jean S, Ritter A, Jurafsky D. Adversarial learning for neural dialogue generation. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Copenhagen, Denmark, Association for Computational Linguistics, 2017, 2157–2169
DOI:10.18653/v1/d17-1230
17 Cui S B, Lian R Z, Jiang D, Song Y F, Bao S Q, Jiang Y. DAL: dual adversarial learning for dialogue generation. 2019
18 Salovey P, Mayer J D. Emotional intelligence. Imagination, Cognition and Personality, 1990, 9(3): 185–211
DOI:10.2190/dugg-p24e-52wk-6cdg
19 Ghosh S, Chollet M, Laksana E, Morency L P, Scherer S. Affect-LM: a neural language model for customizable affective text generation. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). PA, USA, Association for Computational Linguistics, 2017, 634–642
DOI:10.18653/v1/p17-1059
20 Rashkin H, Smith E M, Li M, Bourreau Y L. Towards empathetic open-domain conversation models: a new benchmark and dataset. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Florence,
21 Zhou H, Huang M L, Zhang T Y, Zhu X Y, Liu B. Emotional chatting machine: emotional conversation generation with internal and external memory. 2017
22 Wang K, Wan X. SentiGAN: Generating sentimental texts via mixture adversarial networks. In: Twenty-Seventh International Joint Conference on Artificial Intelligence. Stockholm, Sweden, 2018, 4446-4452 DOI: 10.24963/ijcai.2018/618
23 Sun X, Peng X Q, Ding S. Emotional human-machine conversation generation based on long short-term memory. Cognitive Computation, 2018, 10(3): 389–397 DOI:10.1007/s12559-017-9539-4
24 Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. 2014
25 Graves A, Mohamed A R, Hinton G. Speech recognition with deep recurrent neural networks. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing. Vancouver, BC, Canada, IEEE, 2013, 6645–6649 DOI:10.1109/icassp.2013.6638947
26 Salimans T, Goodfellow I, Zaremba W, Cheung V, Radford A, Chen X. Improved techniques for training GANs. Advances In Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016. Barcelona, Spain, 2016, 2226–2234
27 Lai S, Xu L, Liu K, Zhao, J. Recurrent convolutional neural networks for text classification. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence. Austin, Texas, AAAI, 2015, 2267–2273
28 Srivastava R K, Greff K, Schmidhuber J. Highway networks. 2015
29 Zhang X, LeCun Y. Text Understanding from Scratch. 2015
30 Liu D. Approaches to Chinese word analysis; utterance segmentation and automatic evaluation of machine translation. Beijing: Chinese Academy of Sciences, 2004
31 Doddington G. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In: Proceedings of the second international conference on Human Language Technology Research-. San Diego, California, Morristown, NJ, USA, Association for Computational Linguistics, 2002, 138–145 DOI:10.3115/1289189.1289273
32 Zhang H P, Yu H K, Xiong D Y, Liu Q. HHMM-based Chinese lexical analyzer ICTCLAS. In: Proceedings of the second SIGHAN workshop on Chinese language processing. NJ, USA, Association for Computational Linguistics, 2003 DOI:10.3115/1119250.1119280
33 Sun X, Chen X M, Pei Z M, Ren F J. Emotional human machine conversion generation based on SeqGAN. In: 2018 First Asian Conference on Affective Computing and Intelligent Interaction (ACII Asia). Beijing, China, IEEE, 2018, 1–6 DOI:10.1109/aciiasia.2018.8470388
34 Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M, Kudlur M, Levenberg J, Monga R, Moore S, Derek G. Tensorflow: A system for large-scale machine learning. 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 2016). GA, USA, 2016, 265–283