Generating Responses with a Specific Emotion in Dialog

Zhenqiao Song[1,2], **Xiaoqing Zheng**[*1,2], **Lu Liu**[1,2], **Mu Xu**[3] **and Xuanjing Huang**[1,2]
[1]School of Computer Science, Fudan University, Shanghai, China
[2]Shanghai Key Laboratory of Intelligent Information Processing
[3]Department of Computer Science, University of California, Santa Barbara
{zqsong17, zhengxq, l_liu15}@fudan.edu.cn
muxu@ucsb.edu, xjhuang@fudan.edu.cn

Abstract

It is desirable for dialog systems to have capability to express specific emotions during a conversation, which has a direct, quantifiable impact on improvement of their usability and user satisfaction. After a careful investigation of real-life conversation data, we found that there are at least two ways to express emotions with language. One is to describe emotional states by explicitly using strong emotional words; another is to increase the intensity of the emotional experiences by implicitly combining neutral words in distinct ways. We propose an emotional dialogue system (EmoDS) that can generate the meaningful responses with a coherent structure for a post, and meanwhile express the desired emotion explicitly or implicitly within a unified framework. Experimental results showed EmoDS performed better than the baselines in BLEU, diversity and the quality of emotional expression.

1 Introduction

Humans have the unique capacity to perceive complex, nuanced emotions, and also have the unique capability to communicate those experiences to one another with language. Although recent studies (Partala and Surakka, 2004; Prendinger and Ishizuka, 2005) provide much evidence that the systems capable of expressing emotions significantly improve the user satisfaction, it is still a great challenge to make dialogue systems more "emotional" in their responses.

In early representative work (Polzin and Waibel, 2000; Skowron, 2010), manually prepared rules are applied to deliberately select the desired "emotional" responses from a conversation corpus. Those rules were written by persons with expertise after careful investigation in the corpus, which makes it hard to express complex, various emotions, and difficult to scale well to large datasets.

Post:	I bought a beautiful dress yesterday!
Explicit:	Wearing beautiful dress makes me **happy**!
Implicit:	Wow, you must feel walking on air!
Post:	The rose is really beautiful!
Explicit:	I **love** rose!
Implicit:	I am keen on rose.
Post:	I lost my computer today!
Explicit:	It is really an **annoying** thing.
Implicit:	Oh, you must feel hot under the collar.

Table 1: Examples of two (explicit and implicit) ways in emotional expressions. For each post, one emotional response for each way is listed below. The emotional words associated with strong feelings are highlighted in bold blue font.

Most recently, a sequence to sequence (seq2seq) learning framework with recurrent neural networks (RNNs) has been successfully used to build conversational agents (also known as chatbots) (Sutskever et al., 2014; Sordoni et al., 2015; Shang et al., 2015; Vinyals and Le, 2015; Serban et al., 2016a,b; Wen et al., 2016; Li et al., 2017; Shen et al., 2018) due to their capability to bridge arbitrary time lags. Such framework was also tried to address the problem of emotional expression in a chatbot, called emotional chat machine (ECM) by Zhou el al (2018). However, the authors reported that ECM tends to express the emotion category (say "joy" or "neutral") with much more training samples than others, although it is explicitly asked to express another ("anger" for example). It suffers from exploring the overwhelming samples belonging to a certain emotion category.

Language plays an important role in emotion because it supports the conceptual knowledge used to make meaning of sensations in a given context. As shown in Table 1, we found there are at least two ways to put feelings into words. One is to describe emotional states (such as "anger," "disgust," "contentment," "joy," "sadness," etc.) by *explicitly* using strong emotional words associated with the

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3685–3695
Florence, Italy, July 28 - August 2, 2019.

categories; another is to increase the intensity of the emotional experiences not by using words in emotion lexicon, but by *implicitly* combining neutral words in distinct ways on emotion.

In this study, we propose an emotional dialogue system (EmoDS) that is able to put a specific feeling into words with a coherent structure in an explicit or implicit manner. The seq2seq framework has been extended with a lexicon-based attention mechanism that encourages to replace the words of the response with their synonyms in an emotion lexicon. The response generation process is guided by a sequence-level emotion classifier that not only increases the intensity of emotional expression, but also helps to recognize the emotional sentences not containing any emotional word. We also present a semi-supervised method to create an emotion lexicon that is relatively "accurate" representation of the emotional states that humans are prepared to experience and perceive. Experimental results with both automatic and human evaluations show that for a given post and an emotion category, our EmoDS can express the desired emotion explicitly (if possible) or implicitly (if necessary), and meanwhile successfully generate the meaningful responses with a coherent structure.

2 Related Work

Previous studies have reported that dialog systems equipped with the ability to make appropriate emotional expressions in their responses can directly increase user satisfaction (Prendinger and Ishizuka, 2005) and bring improvement in decision making and problem solving (Partala and Surakka, 2004). A few efforts have been devoted to make dialogue systems more "human-like" by imitating emotional expressions. In early representative work (Polzin and Waibel, 2000; Skowron, 2010), manually prepared rules are used to choose the responses associated with a specific emotion from a conversation corpus. Those rules need to be written by well-trained experts, which makes it hard to extend to deal with complex, nuanced emotions, especially for large corpora.

Recurrent neural networks (RNNs) and their applications in the sequence-to-sequence framework have been empirically proven to be quite successful in structured prediction such as machine translation (Sutskever et al., 2014; Cho et al., 2014; Bahdanau et al., 2014), summarization (Rush et al., 2015), or image caption generation (Vinyals et al., 2015). This framework was also applied to build a chatbot, designed to simulate how a human would behave as an interactive agent (Vinyals and Le, 2015). In earlier attempts to develop chatbots by the seq2seq framework, many efforts have been made to avoid generating dull sentences (such as "tell me more", and "go on") in their responses.

Very recently, a little attention has been given to generate responses with the specific properties like sentiments, tenses, or emotions. Hu et al. (2017) proposed a text generative model based on variational autoencoders (VAEs) to produce sentences presenting a given sentiment or tense. Ghosh et al. (2017) presented a RNN-based language model to generate emotional sentences conditioned on their affect categories. This study focused on the text generations only, but not in the case of conversations. Zhou and Wang (2018) collected a large corpus of Twitter conversations including emojis (ideograms and smileys used in electronic messages) first, and then used emojis to express emotions in the generated texts by trying several variants of conditional VAEs.

Zhou et al. (2018) proposed an emotional chat machine (ECM) based on the seq2seq framework, which is more closely related to this study. They disentangle the emotion factors from texts in the form of embeddings. When ECM is asked to express a specific emotion in the response, the corresponding emotion embedding is consumed by the machine until each element of such embedding being set back to zero. An external emotion lexicon is also used to help the response generation. ECM tends to express the emotion category having the most training samples, although it is specifically asked to express another. Such bias is possibly caused by the potential tendency to gradually blur the differences between the emotion embeddings when the training progresses. We use an emotion classifier to guide the response generation process, which ensures that a specific emotion is appropriately expressed in the generated responses. To our knowledge, this study is among the first ones to built an interactive machine capable of expressing the specific emotions either in an explicit (if possible) or implicit (when necessary) way.

3 Method

In this section, we describe EmoDS that can generate the emotional responses with a coherent struc-

Post:	It is a sunny day today!
Joy:	I really like sunny days, and so we can hang out and have fun!
Contentment:	Wow, let us go outside and play tennis!
Disgust:	I would rather like rainy days.
Anger:	It is none of my business.
Sadness:	I think it seems going to rain.

Table 2: Example responses generated by EmoDS.

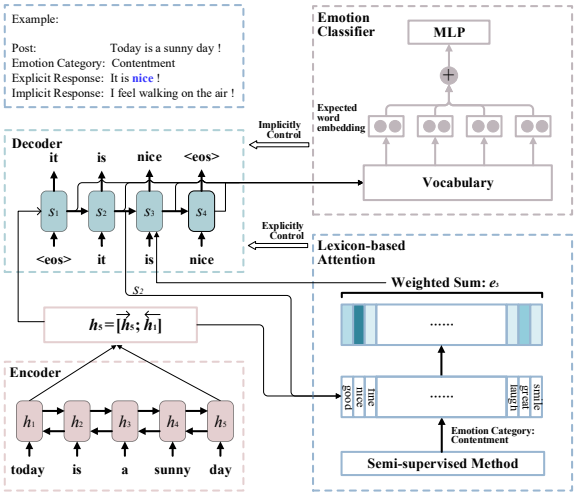

Figure 1: The architecture of an emotional dialogue system (EmoDS). The lower left shows a bidirectional LSTM-based encoder that encodes an input post into its vector representation. This vector representation will be used to initialize a decoder (shown in the upper left) that outputs a meaningful response with a specific emotion in assistance with an emotion classifier (shown in the upper right) and a lexicon-based attention (shown in the lower right). The lexicon-based attention proposes explicitly plugging emotional words into the responses to the encoder at the right time steps, while the emotion classifier provides a global guidance on the emotional response generation in an implicit way by increasing the intensity of emotional expression.

ture in an explicit or implicit manner. The seq2seq framework is extended with a lexicon-based attention mechanism to plug in the desired emotional words. A sequence-level emotion classifier simultaneously helps to recognize the emotional sentences without any emotional word. A diverse decoding algorithm is also presented to foster diversity in response generation. Furthermore, we propose a semi-supervised method to produce an emotion lexicon that can properly represent the mental perceptions of the emotional states.

3.1 Problem Definition

The problem can be formulated as follows: given a post $X = \{x_1, x_2, ..., x_M\}$ and an emotion category e, the objective is to generate a response $Y = \{y_1, y_2, ..., y_N\}$ that is not only meaningful with the content, but also in accordance with the desired emotion, where $x_i \in V$ and $y_j \in V$ are words in the post and response. M and N denote the lengths of the post and response respectively. $V = V_g \bigcup V_e$ is a vocabulary, which consists of a generic vocabulary V_g and an emotion lexicon V_e. We require that $V_g \bigcap V_e = \varnothing$. The lexicon V_e can be further divided into several subsets V_e^z, each of which stores the words associated with an emotion category z. We list an example post with its responses with different emotions in Table 2.

3.2 Dialogue System with Lexicon-based Attention Mechanism

The EmoDS is based on the seq2seq framework that is first introduced for neural machine translation (Sutskever et al., 2014). A lexicon-based attention mechanism (Bahdanau et al., 2014) is also applied to seamlessly "plug" emotional words into the generated texts at the right time steps. The architecture of EmoDS is shown in Figure 1.

Specifically, we use bidirectional long-short term memory (LSTM) (Hochreiter and Schmidhuber, 1997; Schuster and Paliwal, 1997) as an encoder to transform a post, $X = \{x_1, x_2, ..., x_M\}$, into its vector representation. Formally, the hidden states of the encoder can be computed as follows:

$$\begin{aligned} \overrightarrow{h_i} &= LSTM_{forward}(Emb(x_i), \overrightarrow{h_{i-1}}) \\ \overleftarrow{h_i} &= LSTM_{backward}(Emb(x_i), \overleftarrow{h_{i+1}}) \end{aligned} \quad (1)$$

where $i = 1, 2, ..., M$, and $\overrightarrow{h_i}$ and $\overleftarrow{h_i}$ are the i-th hidden states of forward and backward LSTMs respectively. $Emb(x_i) \in \mathbb{R}^d$ is the word embedding of x_i, and d is the dimensionality of word embeddings. We concatenate the corresponding hidden states of the forward and backward LSTMs, namely $h_i = [\overrightarrow{h_i}; \overleftarrow{h_i}]$, as the i-th hidden state produced by the two LSTMs. The last hidden state h_M is fed to a decoder as its initialization.

The decoder module contains a separate LSTM enhanced with a lexicon-based attention mechanism. The LSTM decoder takes as input a previously predicted word y_{j-1} and an emotion vector e_j to update its hidden state s_j as follows:

$$s_j = LSTM_{decoder}([Emb(y_{j-1}); e_j], s_{j-1}) \quad (2)$$

where $j = 1, 2, ..., N$ and $s_0 = h_M$. $Emb(y_{j-1})$ is the word embedding of y_{j-1}, and $[\cdot;\cdot]$ denotes an operation that concatenates the feature vectors

separated with semicolons. The emotion vector e_j is calculated as a weighted sum of embeddings of words in V_e^z with the given category z:

$$\begin{aligned} e_j &= \sum_k a_{jk} \cdot Emb(w_k^z) \\ a_{jk} &= \frac{exp(c_{jk})}{\sum_{t=1}^{T_z} exp(c_{jt})} \\ c_{jk} &= Sigmoid(\alpha^\top h_M + \beta^\top s_{j-1} + \gamma^\top Emb(w_k^z)) \end{aligned} \tag{3}$$

where w_k^z denotes the k-th word in V_e^z, T_z is the number of words for the emotion category z, and α, β and γ are trainable parameters. We compute attention scores using the global attention model proposed by Luong et al. (2015). For each emotional word w_k^z in V_e^z, the attention score a_{jk} at the time step j is determined by three parts: the previous hidden state s_{j-1} of the decoder, the encoded representation h_M of the input post, and the embedding $Emb(w_k^z)$ of the k-th word in V_e^z. Therefore, given the partial generated response and the input post, the more relevant an emotional word is, the more influence it will have on the emotion feature vector at the current time step. In this way, such lexicon-based attention gives higher probability to the emotional words that are more relevant to the current context.

In order to plug the emotional words into the responses, we estimate both a probability distribution $P_e(y_j = w^e)$ over all the emotional words w^e in V_e^z for a given emotion type z, and a probability distribution $P_g(y_j = w^g)$ over all the generic words w^g in V_g as follows:

$$\begin{aligned} P_e(y_j = w^e) &= Softmax(W_e s_j) \\ P_g(y_j = w^g) &= Softmax(W_g s_j) \\ \delta_j &= Sigmoid(\upsilon^\top s_j) \\ y_j \sim P(y_j) &= \begin{bmatrix} \delta_j P_e(y_j = w^e) \\ (1-\delta_j) P_g(y_j = w^g) \end{bmatrix} \end{aligned} \tag{4}$$

where $\delta_j \in (0, 1)$ is a type selector controlling the weight of generating an emotional or a generic word, and W_e, W_g and υ are trainable parameters. The lexicon-based attention mechanism helps to put the desired emotional words into response at the right time steps, which makes it possible to express the expected feelings in the generated texts. The loss function for each sample is defined by minimizing the cross-entropy error in which the target distribution t is a binary vector with all elements zero except for the ground truth:

$$L_{MCE} = -\sum_{j=1}^{N} t_j \log(P(y_j)) \tag{5}$$

3.3 Emotion Classification

The feelings can be put into words either by explicitly using strong emotional words associated with a specific category, or by implicitly combining neutral words to a sequence in distinct ways. Therefore, we use a sequence-level emotion classifier to guide the generation process, which helps to recognize the responses expressing a certain emotion but not containing any emotional word. A straightforward method to introduce such a classifier is to build a sentence-level emotion discriminator as follows:

$$Q(E|Y) = Softmax(W \cdot \frac{1}{N} \sum_{j=1}^{N} Emb(y_j)) \tag{6}$$

where $W \in \mathbb{R}^{K \times d}$ is a weight matrix and K denotes the number of emotion categories. However, it is infeasible to enumerate all possible sequences as the search space is exponential to the size of vocabulary, and the length of Y is not known in advance. Besides, it is non-differentiable if we approximate the generation process by sampling few sequences according to their probabilities.

Following Kočiskỳ et al. (2016), we use the idea of expected word embedding to approximate $Q(E|Y)$. Specifically, the expected word embedding is a weighted sum of embeddings of all the possible words at each time step:

$$Ewe(j; X, z) = \sum_{y_j \in V_g \cup V_e^z} P(y_j) \cdot Emb(y_j) \tag{7}$$

where for each time step j, we enumerate all possible words that are in the union of V_g and V_e^z. The classification loss for each sample is defined as:

$$\begin{aligned} L_{CLA} &= -P(E)\log(Q(E|Y)) \\ Q(E|Y) &= Softmax(W \cdot \frac{1}{N} \sum_{j=1}^{N} Ewe(j; X, z)) \end{aligned} \tag{8}$$

where $P(E)$ is a one-hot vector that represents the desired emotion distribution for an instance.

The introduced emotion classifier can not only increase the intensity of emotional expression, but also help to identify the emotional responses not containing any emotional word. Note that the emotion classifier is used only during training process, and can be taken as a global guidance for emotional expression.

3.4 Training Objective

The overall training objective is divided into two parts: the generation loss and the classification

one, which can be written as:

$$L = L_{MCE} + \lambda L_{CLA} \quad (9)$$

where a hyperparameter λ governs the relative importance of the generation loss compared with the classification term. The generation loss L_{MCE} ensures that the decoder can produce meaningful responses with a coherent structure, while the emotion classification term guides the generation process and guarantees that a specific emotion is appropriately expressed in the generated responses.

3.5 Diverse Decoding Algorithm

Li et al. (2016c) found that most responses in the N-best results produced by the traditional beam search are much similar, and thus we propose a diverse decoding algorithm to foster diversity in the response generation. We force the head words of N-candidates should be different, and then the model continues to generate a response by a greedy decoding strategy after such head words are determined. Finally, we choose the response with the highest emotion score from the best N-candidates. The candidates are scored by the emotion classifier trained in advance on a dataset annotated automatically (see Section 4.1). Therefore, our model can produce the N-best candidates with more diversity, in which the one with the highest emotion score is chosen as the final result.

3.6 Emotion Lexicon Construction

In this section, we describe how to construct the required emotion lexicon in semi-supervised manner from a corpus consisting of the sentences annotated with their emotion categories. The meaning of words is rated on a number of different bipolar adjective scales. For example, scales might range from "strong" to "weak". We only collect the words rated as "strong" for each emotion category and put into the emotion lexicon.

Inspired by Vo and Zhang (2016), each word is represented as $w = (p_w, n_w)$ for an emotion category (i.e. "joy"), where p_w denotes the probability being assigned to this category while n_w denotes the opposite. Given a sentence s that is a sequence of n words, and the estimated emotion probability is simply calculated as $\hat{z_s} = \sum_{i=1}^{n}(\frac{p_{w_i}}{n}, \frac{n_{w_i}}{n})$. If sentence s presents the emotion, it is labeled as a two-dimensional emotion vector $z = (1, 0)$; if not $z = (0, 1)$. Each word is initialized by small random values, and trained by minimizing the cross-entropy error in form of $\{-\sum_{i=1}^{m} z_m \log \hat{z_m}\}$, where m is the number of sentences in a corpus.

Training	Post	$3,992,363$	
	Response	Anger	$204,797$
		Disgust	$535,869$
		Contentment	$344,549$
		Joy	$1,065,689$
		Sadness	$494,962$
		Neutral	$1,346,497$
Validation	All	$221,798$	
Test	All	$221,798$	

Table 3: Statistics of emotion-labeled STC dataset.

Method	Accuracy
Lexicon-based	0.453
RNN	0.572
LSTM	0.597
Bi-LSTM	0.635

Table 4: Classification accuracy on the NLPCC dataset.

We remove all the stop words in the sentences, and map the recognized "digit," "E-mail," "URL," "date," and "foreign word" into special symbols. The words following the negation are transformed to $(-p_w, -n_w)$ before they are used to produce the emotion vector of its sentence. If the words are modified by superlative or comparative adjectives (or adverbs), the value of learning rate used to update their representations will be doubled or tripled accordingly. The training process can be divided into two stages. In the first stage, the standard back-propagation is applied. When the prediction accuracy is greater than a given threshold (say 90%), the second stage starts using the maximum margin learning strategy until arriving at a convergence. After the training stops, we compute an average as $v = \frac{1}{n}\sum_{i=1}^{n}(p_w - n_w)$ and its variance σ. The word with its value $\frac{1}{\sigma}(p_w - n_w - v)$ being greater than a certain threshold will be identified as an emotional word.

4 Experiments

4.1 Data Preparation

There is no large-scale off-the-shelf emotional conversation data, so we constructed our own experimental dataset based on Short Text Conversation (STC) dataset[1] (Shang et al., 2015). Following Zhou et al. (2018), we first trained an emotion classifier on NLPCC dataset[2] and then annotated

[1] Available at http://ntcir12.noahlab.com.hk/stc.htm
[2] Available at http://http://tcci.ccf.org.cn/nlpcc.php

Models	Embedding			BLEU Score	Diversity		Emotional Expression	
	Average	Greedy	Extreme	BLEU	distinct-1	distinct-2	emotion-a	emotion-w
Seq2Seq	0.523	0.376	0.350	1.50	0.0038	0.012	0.335	0.371
EmoEmb	0.524	0.381	0.355	1.69	0.0054	0.0484	0.720	0.512
ECM	0.624	0.434	0.409	1.68	0.0090	0.0735	0.765	0.580
EmoDS-MLE	0.548	0.367	0.374	1.60	0.0053	0.0670	0.721	0.556
EmoDS-EV	0.571	0.390	0.384	1.64	0.0053	0.0659	0.746	0.470
EmoDS-BS	0.614	0.442	0.409	**1.73**	0.0051	0.0467	0.773	0.658
EmoDS	**0.634**	**0.451**	**0.435**	**1.73**	**0.0113**	**0.0867**	**0.810**	**0.687**

Table 5: Results reported in the embedding scores, BLEU, diversity, and the quality of emotional expression.

STC dataset using this classifier.

More specifically, we trained a bidirectional LSTM (Bi-LSTM) classifier on NLPCC dataset for emotion classification, as it achieved the highest classification accuracy compared with other classifiers (Zhou et al., 2018). Accuracies of several neural network-based classifiers are shown in Table 4. NLPCC dataset is composed of emotion classification data in NLPCC2013[3] and NLPCC2014[4]. There are eight emotion categories in this dataset, including Anger (7.9%), Disgust (11.9%), Contentment (11.4%), Joy (19.1%), Sadness (11.7%), Fear (1.5%), Surprise (3.3%) and Neutral (33.2%). After removing the infrequent categories (Fear and Surprise), we have six emotion categories at last: Anger, Disgust, Contentment, Joy, Sadness and Neutral. Next we used the well-trained Bi-LSTM classifier to annotate the STC dataset with the six emotion labels, and thus we obtained the emotion-labeled conversation dataset. Finally we randomly split the emotion-labeled STC dataset into training/validation/test sets with the ratio of 9:0.5:0.5. The detailed statistics are shown in Table 3.

4.2 Training Details

We implemented our EmoDS in Tensorflow[5]. Specifically, we used one layer of bidirectional LSTM for encoder and another uni-directional LSTM for decoder, with the size of LSTM hidden state set as 256 in both the encoder and decoder. The dimension of word embedding was set to 100, which was initialized with Glove embedding (Pennington et al., 2014). Many empirical results show that such pre-trained word representations can enhance the supervised models on a variety of NLP tasks (Zheng et al., 2013; Zheng, 2017; Feng and Zheng, 2018). The generic vocabulary was built based on the most frequent $30,000$ words, and the emotion lexicon for each category was constructed by our semi-supervised method with size set to 200. All the remaining words were replaced by a special token <UNK>. Parameters were randomly initialized by the uniform distribution within $[-3.0/n, 3.0/n]$, where n denotes the dimension of parameters. The size of diverse decoding was set to 20. We tuned the only hyper-parameter λ in {1e-1,1e-2,1e-3,1e-4}, and found that 1e-2 worked best.

We applied the stochastic gradient descent (SGD) (Robbins and Monro, 1985) algorithm with mini-batch for optimization. The mini-batch size and learning rate were set to 64 and 0.5, respectively. We run the training for 20 epoches and the training stage took about 5 hours on a TITAN X GPU card. Our code will be released soon.

4.3 Baseline Models

We conducted extensive experiments to compare EmoDS against the following representative baselines: (1) **Seq2Seq**: We implemented the Seq2Seq model as in Vinyals and Le (2015); (2) **EmoEmb**: Inspired by Li et al. (2016b), we represented each emotion category as a vector and fed it to the decoder at each time step. We call this model emotion embedding dialogue system (EmoEmb). (3) **ECM**: We used the code released by Zhou et al. (2018) to implement ECM.

Additionally, to better analyze the influence of different components in our model, we also conducted ablation tests as follows: (4) **EmoDS-MLE**: EmoDS is only optimized with the MLE objective, without the emotion classification term. (5) **EmoDS-EV**: EmoDS uses an external emotion lexicon[6] instead of producing an internal one. (6) **EmoDS-BS**: EmoDS applies the original beam search rather than our diverse decoding.

[3]Available at http://tcci.ccf.org.cn/conference/2013/
[4]Available at http://tcci.ccf.org.cn/conference/2014/
[5]Available at https://www.tensorflow.org/
[6]http://download.csdn.net/download/abacaba/9722161

Models	Joy		Contentment		Disgust		Anger		Sadness		Overall	
	Cont.	Emot.	Cont.	Emot.	Cont.	Emot.	Cont.	Emot.	Cont.	Emot.	Cont.	Emot.
Seq2Seq	1.350	0.455	**1.445**	0.325	1.180	0.095	1.150	0.115	1.090	0.100	1.243	0.216
EmoEmb	1.285	0.655	1.320	0.565	1.015	0.225	1.160	0.400	0.995	0.190	1.155	0.407
ECM	**1.395**	0.690	1.400	0.615	1.130	0.425	**1.190**	0.330	1.195	0.335	1.262	0.479
EmoDS	1.265	**0.695**	1.260	**0.685**	**1.370**	**0.530**	1.185	**0.505**	**1.265**	**0.625**	**1.269**	**0.608**

Table 6: The results of human evaluation. Cont. and Emot. denote content and emotion, respectively.

Models	2-1	1-1	0-1	2-0	1-0	0-0
Seq2Seq	10.0	8.6	3.2	35.1	25.5	17.6
EmoEmb	20.4	11.4	8.9	23.5	16.3	19.5
ECM	26.5	15.3	7.5	20.4	17.9	12.4
EmoDS	**31.7**	19.3	9.8	17.7	8.8	12.7

Table 7: The distribution (%) of *Content-Emotion* scores.

Pref. (%)	Seq2Seq	EmoEmb	ECM	EmoDS
Seq2Seq	-	44.7	36.9	30.7
EmoEmb	55.3	-	42.4	39.9
ECM	63.1	57.6	-	41.4
EmoDS	**69.3**	**60.1**	**58.6**	-

Table 8: Preference test (%) between any two models.

4.4 Automatic Evaluation

4.4.1 Metrics

We used the following metrics to evaluate the performance of our EmoDS: (1) **Embedding Score**: We employed three embedding-based metrics (average, greedy and extreme) (Liu et al., 2016), which map the responses into vector space and compute the cosine similarity. The embedding-based metrics can, to a large extent, capture the semantic-level similarity between the generated responses and the ground truth. (2) **BLEU Score**: BLEU (Papineni et al., 2002) is a popular metric that calculates the word-overlap score of the generated responses against gold-standard responses. BLEU in this paper refers to the default BLEU-4. (3) **Distinct**: Distinct-1/distinct-2 is the proportion of the distinct unigrams/bigrams in all the generated tokens, respectively (Li et al., 2016a). Distinct metrics can be used to evaluate the diversity of the responses. (4) **Emotion Evaluation**: We designed two emotion-based metrics, emotion-a and emotion-w, to test how well the emotion is expressed in the generated responses. Emotion-a is the agreement between the predicted labels through the Bi-LSTM classifier in Data Preparation and the ground truth labels. Emotion-w is the percentage of the generated responses that contain the corresponding emotional words.

4.4.2 Results

The results are reported in Table 5. The top half is the results of all baseline models, and we can see that EmoDS outperformed the competitors in all cases. Notably, EmoDS achieved significant improvements on emotion-a and emotion-w over EmoEmb and ECM, indicating that our EmoDS can generate coherent responses with better emotional expression. Seq2Seq model performed rather poorly on nearly all metrics, primarily because it does not take any emotion factor into account and tends to generate short generic responses. The ability to express emotions in both explicit and implicit manners makes EmoDS generate more emotional responses.

The bottom half of Table 5 shows the results of ablation tests. As we can see, after removing the emotion classification term (EmoDS-MLE), the performance decreased most significantly. Our interpretation is that without the emotion classification term, the model can only express the desired emotion explicitly in the generated responses and can not capture the emotional sequences not containing any emotional word. Applying an external emotion lexicon (EmoDS-EV) also brought performance decline, especially on emotion-w. This makes sense because an external emotion lexicon shares fewer words with the corpus, causing the generation process to focus on generic vocabulary and more commonplace responses to be generated. Additionally, the distinct-1/distinct-2 decreased most when using the original beam search (EmoDS-BS), indicating that the diverse decoding can promote diversity in response generation.

4.5 Human Evaluation

4.5.1 Evaluation Settings

Following the protocols defined in (Zhou et al., 2018), we employed a human evaluation method designed from the content and emotion levels to better understand the quality of the generated re-

Post	Model		Response	Translated Response
月光下的海真美。 Sea under the moonlight is so beautiful.	Seq2Seq		照片怎么拍出来的?	What do you think of this photo?
	EmoDS	Anger	一点都不美。	It is not beautiful at all.
		Disgust	照片拍的真**难看**。	The photo is really **ugly**.
		Contentment	好想去看看。	I really want to see the scenery.
		Joy	月亮和海都 **好美**！	Both the moon and sea are **so beautiful**!
		Sadness	感觉很**孤单**！	It makes me feel **lonely**!
大自然是最伟大的艺术家 The nature is the greatest artist .	Seq2Seq		大自然的力量无穷。	The power of the nature is endless.
	EmoDS	Anger	我一点也没觉得。	I feel nothing at all.
		Disgust	我**讨厌**大自然。	I **hate** the nature.
		Contentment	当然！大自然 **太美**了！	Sure ! How **beautiful** the nature is!
		Joy	我**爱**大自然！	I **love** the nature!
		Sadness	破坏大自然太**残忍**了！	It is so **cruel** to destroy the nature!
手工制作的帽子和手链。 The handmade hat and bracelet.	Seq2Seq		我也想要一个。	I also want one.
	EmoDS	Anger	一点都不好看。	It does not look nice at all.
		Disgust	这个手链看着像**赝品**。	The bracelet looks like **a fake**.
		Contentment	很有**创意**的设计。	It is a **creative** design.
		Joy	我**喜欢**这个设计。	I **like** the design.
		Sadness	这是一个**悲伤**的故事。	It is a **sad** story.
香香甜甜的美梦，我来咯 Have a sweet dream! I am going to bed.	Seq2Seq		哈哈，这是你的梦。	Haha, that is your dream .
	EmoDS	Anger	赶紧睡吧，我要**疯了**！	Go to bed quickly, and I am **crazy** .
		Disgust	我**讨厌**做梦 。	I **hate** dreaming .
		Contentment	**哇**，好**可爱**的小 姑娘！	**Wow**, what a **lovely** girl .
		Joy	我**喜欢**做梦！	I **like** dreaming .
		Sadness	感觉你会做**噩梦**！	I feel you may have a **nightmare** .
最美的天空来自自由的阳光。 The brilliant and free sunshine brings about the most beautiful sky.	Seq2Seq		自由的阳光。	A free sunshine .
	EmoDS	Anger	你怎么了？	what is wrong with you ?
		Disgust	这样的蓝天真**难看**。	This sky is really **ugly** .
		Contentment	阳光让人**灿烂**。	Sunshine makes me **happy** .
		Joy	我**最喜欢**的阳光灿烂 ！	I **like** the sunshine **very much** .
		Sadness	没有阳光的自由更好。	I feel better without the sunshine .

Table 9: Case study for EmoDS. For each post, one sample response is listed for each emotion category. The emotions of the responses containing emotional words (highlighted in blue font) are expressed explicitly, while those of others are expressed implicitly.

sponses. First, two hundred posts were randomly sampled from the test set and for each of them, all models except Seq2Seq generated six responses for six emotion categories. Instead, Seq2Seq model generated top 6 responses in beam search for each post. Later the triples of (post, response, emotion) were presented to three human judges with order disrupted. They evaluated each response from the content level by 3-scale rating (0, 1, 2) and emotion level by 2-scale rating (0, 1). Evaluation from the content level assesses whether a response is coherent and meaningful for the context. Evaluation from the emotion level decides if a response reveals the desired emotion property.

Agreements to measure inter-rater consistency among three annotators were calculated with the Fleiss's kappa (Fleiss and Cohen, 1973). Finally, the Fleiss's kappa for content and emotion is 0.513 and 0.811, indicating "Moderate agreement" and "Substantial agreement", respectively.

4.5.2 Results

It is shown in Table 6 that EmoDS achieved the highest performance in most cases (Sign Test, with p-value < 0.05). Specifically, for content coherence, there was no obvious difference among most models, but for emotional expression, the EmoDS yielded a significant performance boost. As we can see from Table 6, EmoDS performed well on all categories with an overall emotion score of 0.608, while EmoEmb and ECM performed poorly on categories with less training data, e.g., disgust, anger and sadness. Note that all emotion scores of Seq2Seq were the lowest, indicating that Seq2Seq is bad at emotional expression when generating responses. To sum up, EmoDS can generate meaningful responses with better emotional expression, due to the fact that EmoDS is capable of expressing the desired emotion either explicitly or implicitly.

To better analyze the overall quality of the generated responses at both the content and emotion

levels, we also report the distribution of the combined content and emotion scores in Table 7. It shows that 31.7% of the responses generated by EmoDS were annotated with a content score of 2 and an emotion score of 1, which is higher than all the other three models. This demonstrates that EmoDS is better at generating high-quality responses in the respect of both the content and emotion. Furthermore, the results of preference test are shown in Table 8. It can be seen that EmoDS is significantly preferred against other models (Sign Test, with p-value < 0.05). Obviously, the diverse emotional responses generated by our EmoDS are more attractive to users than the commonplace responses generated by the Seq2Seq.

4.6 Case Study

To gain an insight on how well the emotion is expressed in the generated responses, we provide some examples in Table 9. It shows that the EmoDS can generate informative responses with any desired emotion by putting a specific feeling into words either in an explicit or implicit manner. For example, "难看 (ugly)" is a strong emotional word that is used to explicitly describe the emotional state of disgust, while the words in "好 / 想 / 去 / 看看 / 。 (I really want to see the scenery.)" are all neutral ones, but their combination can express the emotional state of contentment.

5 Conclusion

Observing that emotional states can be expressed with language by explicitly using strong emotional words or by forming neutral word in distinct patterns, we proposed a novel emotional dialog system (EmoDS) that can express the desired emotions in either way, and at the same time generate the meaningful responses with a coherent structure. The sequence-to-sequence framework has been extended with a lexicon-based attention mechanism that works by seamlessly "plugging" emotional words into the texts by increasing their probability at the right time steps. An emotion classifier is also used to guide the response generation process, which ensures that a specific emotion is appropriately expressed in the generated texts. To our knowledge, this study is among the first ones to build an interactive machine capable of expressing the specific emotions either in an explicit (if possible) or implicit (when necessary) way. Experimental results with both automatic and human evaluations demonstrated that EmoDS outperformed the baselines in BLEU, diversity and the quality of emotional expression with a significant margin, highlighting the potential of the proposed architecture for practical dialog systems.

6 Acknowledgements

The authors would like to thank the anonymous reviewers for their valuable comments. We are also grateful to Chenxin An, Geng Hong, Yingshan Yang and Zongyi Li for their suggestions. This work was supported by National Key R&D Program of China (No. 2018YFC0830902), Shanghai Municipal Science and Technology Major Project (No. 2018SHZDZX01) and Zhangjiang Lab.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 2014 International Conference on Learning Representations*.

Kyunghyun Cho, Bart Merrienboer, Caglar Gulcehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*.

Jiangtao Feng and Xiaoqing Zheng. 2018. Geometric relationship between word and context representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

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

Sayan Ghosh, Mathieu Chollet, Eugene Laksana, Louis-Philippe Morency, and Stefan Scherer. 2017. 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)*, volume 1, pages 634–642.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. *arXiv preprint arXiv:1703.00955*.

Tomáš Kočiskỳ, Gábor Melis, Edward Grefenstette, Chris Dyer, Wang Ling, Phil Blunsom, and Karl Moritz Hermann. 2016. Semantic parsing with semi-supervised sequential autoencoders. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1078–1087.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119.

Jiwei Li, Michel Galley, Chris Brockett, Georgios P Spithourakis, Jianfeng Gao, and Bill Dolan. 2016b. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 994–1003.

Jiwei Li, Will Monroe, and Dan Jurafsky. 2016c. A simple, fast diverse decoding algorithm for neural generation. *arXiv preprint arXiv:1611.08562*.

Jiwei Li, Will Monroe, Tianlin Shi, Sėbastien Jean, Alan Ritter, and Dan Jurafsky. 2017. Adversarial learning for neural dialogue generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2157–2169.

Chia-Wei Liu, Ryan Lowe, Iulian Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2122–2132.

Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318. Association for Computational Linguistics.

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

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Thomas S Polzin and Alexander Waibel. 2000. Emotion-sensitive human-computer interfaces. In *ISCA tutorial and research workshop (ITRW) on speech and emotion*.

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

Herbert Robbins and Sutton Monro. 1985. A stochastic approximation method. In *Herbert Robbins Selected Papers*, pages 102–109. Springer.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for sentence summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Iulian V Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2016a. A hierarchical latent variable encoder-decoder model for generating dialogues. *arXiv preprint arXiv:1605.06069*.

Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C Courville, and Joelle Pineau. 2016b. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Proceedings of the 2016 Association for the Advancement of Artificial Intelligence*, volume 16, pages 3776–3784.

Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 1577–1586.

Xiaoyu Shen, Hui Su, Wenjie Li, and Dietrich Klakow. 2018. Nexus network: Connecting the preceding and the following in dialogue generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4316–4327.

Marcin Skowron. 2010. Affect listeners: Acquisition of affective states by means of conversational systems. In *Development of Multimodal Interfaces: Active Listening and Synchrony*, pages 169–181. Springer.

Alessandro Sordoni, Yoshua Bengio, Hossein Vahabi, Christina Lioma, Jakob Grue Simonsen, and Jian-Yun Nie. 2015. A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 553–562. ACM.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Oriol Vinyals, Samy Bengio Alexander Toshev, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*.

Oriol Vinyals and Quoc Le. 2015. A neural conversational model. *arXiv preprint arXiv:1506.05869*.

Duy Tin Vo and Yue Zhang. 2016. Don't count, predict! an automatic approch to learning sentiment lexicons for short text. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Tsung-Hsien Wen, David Vandyke, Nikola Mrksic, Milica Gasic, Lina M Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2016. A network-based end-to-end trainable task-oriented dialogue system. *arXiv preprint arXiv:1604.04562*.

Xiaoqing Zheng. 2017. Incremental graph-based neural dependency parsing. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing*.

Xiaoqing Zheng, Hanyang Chen, and Tianyu Xu. 2013. Deep learning for chinese word segmentation and pos tagging. In *Proceedings of the Conference on Empirical Methods on Natural Language Processing*.

Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2018. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *Proceedings of the 2018 Association for the Advancement of Artificial Intelligence*.

Xianda Zhou and William Yang Wang. 2018. Mojitalk: Generating emotional responses at scale. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 1128–1137.