Affect-Driven Dialog Generation

Pierre Colombo[1*], **Wojciech Witon**[1*], **Ashutosh Modi**[1], **James Kennedy**[1], **Mubbasir Kapadia**[1,2]
[1]Disney Research, [2]Rutgers University
{pierre.colombo, wojtek.witon}@disneyresearch.com
ashutosh.modi@disneyresearch.com
james.kennedy@disneyresearch.com
mubbasir.kapadia@rutgers.edu

Abstract

The majority of current systems for end-to-end dialog generation focus on response quality without an explicit control over the affective content of the responses. In this paper, we present an affect-driven dialog system, which generates emotional responses in a controlled manner using a continuous representation of emotions. The system achieves this by modeling emotions at a word and sequence level using: (1) a vector representation of the desired emotion, (2) an affect regularizer, which penalizes neutral words, and (3) an affect sampling method, which forces the neural network to generate diverse words that are emotionally relevant. During inference, we use a re-ranking procedure that aims to extract the most emotionally relevant responses using a human-in-the-loop optimization process. We study the performance of our system in terms of both quantitative (BLEU score and response diversity), and qualitative (emotional appropriateness) measures.

1 Introduction

Recent breakthroughs in deep learning techniques have had an impact on end-to-end conversational systems (Chen et al., 2017). Current research is mainly focused on functional aspects of conversational systems: keyword extraction, natural language understanding, and pertinence of generated responses (Ilievski et al., 2018). Although these aspects are indeed key features for building a commercial system, most existing solutions lack social intelligence. Conversational systems could benefit from incorporating social intelligence by: (1) avoiding interaction problems that may arise when the system does not understand the user's request (e.g., inappropriate responses that cause user anger) (Maslowski et al., 2017), and (2) building rapport with the user (Strohkorb et al., 2016). Our method makes such conversational systems more social by outputting responses expressing emotion in a controlled manner, without sacrificing grammatical correctness, coherence, or relevance.

Existing sequence-to-sequence (seq2seq) architectures, either recurrent- (Sordoni et al., 2015; Serban et al., 2015), attention- (Vaswani et al., 2017) or convolutional neural network (CNN)-based (Fan et al., 2018), do not provide a straightforward way to generate emotionally relevant output in a controlled manner. We introduce EMOTIonal CONversational System (EMOTICONS), which generates emotion-specific responses. It is based on novel contributions presented in this paper which fall in two main categories: explicit models which allow a controlled emotion-based response generation (e.g., methods based on emotion embeddings, affective sampling, and affective re-ranking), and implicit models with no direct control over the desired emotion (i.e., affective regularizer). We show that EMOTICONS outperforms both the system proposed by Zhou et al. (2018) (current state of the art for our task) and the vanilla seq2seq in terms of BLEU score (Papineni et al., 2002) (improvement up to 7.7%) and response diversity (improvement up to 52%). Additionally, we qualitatively evaluate the emotional content of the generated text (see example responses in Table 1). The user study (22 people) demonstrates that EMOTICONS is able to generate grammatically correct, coherent, emotionally-rich text in a controlled manner.

2 Related Work

Sequence-to-sequence (seq2seq) models have attracted a lot of attention in the past few years, especially in the fields of Neural Machine Translation (Sutskever et al., 2014; Bahdanau et al., 2014) and Neural Dialogue Generation (Sordoni et al., 2015; Vinyals and Le, 2015; Serban et al.,

[*] Both authors contributed equally to this work.

Proceedings of NAACL-HLT 2019, pages 3734–3743
Minneapolis, Minnesota, June 2 - June 7, 2019.

	Input	Good to see you again.	Be careful, I have a knife.	You're the only one who can help us.
EMOTICONS	Baseline	It's good to see you.	Don't worry about it.	There's a lot of people here.
	Fear	What are you doing here?	I'm just trying to get out of here.	What are you going to do with me?
	Anger	I'm not here.	I'm going to kill you.	You're not the only one who can help me.
	Joy	Good to see you.	I'm going to marry you.	I can protect you.
	Surprise	You want to talk to me?	I'm just going to the party.	You've got to be kidding me.

Table 1: Example responses from the baseline (seq2seq) model and the four EMOTICONS models with different emotions.

2015). Prior work has focused on designing architectures that lead to the best performance in terms of BLEU (Papineni et al., 2002) and Perplexity scores. Most seq2seq models are based on gated recurrent neural networks, either Long Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) or Gated Recurrent Unit (GRU) (Serban et al., 2015), but in general it is difficult to conclude which gating mechanism performs better (Chung et al., 2014). In our model, we use GRU because it has fewer parameters to optimize, and it is faster to train.

In order to overcome the problem of generating trivial or mundane responses, there have been developments in inference techniques for encoder-decoder systems. Use of beam search has been shown to improve the general quality of generated answers, while Maximum Mutual Information (MMI) (Li et al., 2016) has improved the diversity of generated answers, leading to more meaningful output. We build on these techniques during affective inference.

Emotion-based (affective) dialog generation systems have received increasing attention in the past few years. Huang et al. (2018) use emotion tokens (special "words" in a dictionary representing specific emotions) at either the encoder or decoder side, forcing the decoder to output a sentence with one specific emotion. Zhou et al. (2018) build their system using external and internal memory, where the former forces the network to generate emotional words, and the latter measures how emotional a generated sequence is compared to a target sequence. Lubis et al. (2018) modeled emotions in Valence-Arousal (VA) space for response generation. We extend this idea by using a Valence-Arousal-Dominance (VAD) Lexicon (Mohammad, 2018), as it has been shown by Broekens (2012) that the third dimension (Dominance) is useful for modeling affect. Asghar et al. (2017) used the VAD Lexicon, but they let the neural network choose the emotion to generate (by maximizing or minimizing the affective dissonance) and their system cannot generate different emotional outputs for the same input, nor generate a specified emotion.

3 System Architecture

Our system (see overview in Figure 1) is divided into three main components: (1) Emotion Labeling – automatic labeling of sentences according to the emotional content they express, using an emotion classifier (§3.2.1); labeling of words with VAD Lexicon values (§4.2), (2) Affective Training – training of two seq2seq networks, which use an encoder-decoder setting. The first network is trained with prompt-response pairs (S-T), whereas the second (used during Affective Inference) is trained with reversed pairs (T-S), (3) Affective Inference – generation of many plausible responses, which are re-ranked based on emotional content.

3.1 Preliminaries

Let $V = \{w_1, w_2, \ldots, w_{|V|}\}$ be a vocabulary, and $\mathcal{X} = (x_1, x_2, \ldots, x_{|\mathcal{X}|})$ a sequence of words (e.g. a sentence). We denote $E_{\mathcal{X}} \in \mathbb{R}^6$ as an emotion vector representing a probability distribution over six emotions associated with the sequence $\mathcal{X}$:

$$E_{\mathcal{X}} = \begin{bmatrix} p_{\text{anger}} \\ p_{\text{surprise}} \\ p_{\text{joy}} \\ p_{\text{sadness}} \\ p_{\text{fear}} \\ p_{\text{disgust}} \end{bmatrix}$$

Note that in this work we focus on six basic emotions proposed by Paul Ekman (Ekman et al., 1983) but the techniques we develop are general and can be extended to a more fine grained list of emotions. $\mathcal{X}$ can be an input sequence, candidate response, final response, or target response (denoted respectively as $\mathcal{S}$, $\mathcal{R}_C$, $\mathcal{R}_{\text{final}}$, $\mathcal{R}_0$). We introduce E_0, which during training, is the representation of the emotion of the target response ($\mathcal{R}_0$). During testing, E_0 indicates a desired emotion for the final response ($\mathcal{R}_{\text{final}}$), and can be set manually. For

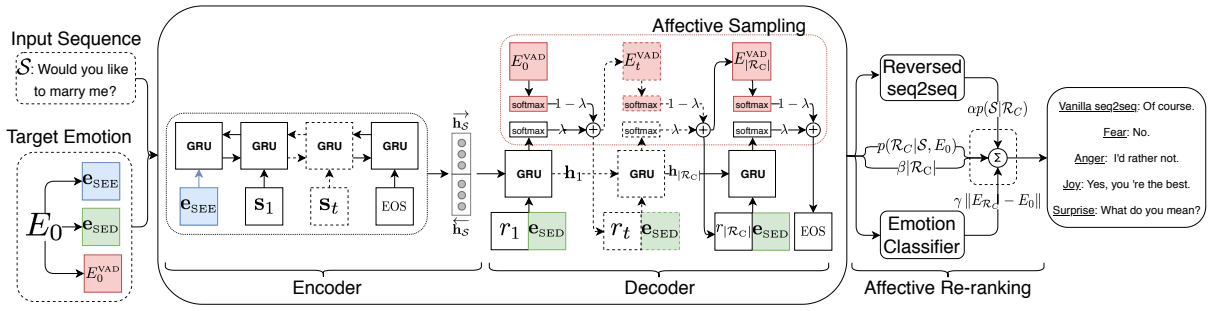

Figure 1: System overview: The input sequence and the target emotion (automatically extracted from the target response using the emotion classifier during training, and set by the user during inference) are fed into the seq2seq. The generated candidate responses are re-ranked based on the output of the reversed seq2seq, the length, and the emotional content.

example, in the case of 'anger', E_0 would be a one-hot vector with 1 at the first position, and 0 elsewhere.

In our work, we extend the standard seq2seq model (Sutskever et al., 2014), that predicts the final response $\mathcal{R}_{\text{final}} = \text{argmax}_{\mathcal{R}_C}\, p(\mathcal{R}_C|S)$. The proposed affective system aims to extend the inference mechanism by incorporating emotions encoded in E_0:

$$\mathcal{R}_{\text{final}} = \underset{\mathcal{R}_C}{\text{argmax}}\, p(\mathcal{R}_C|S, E_0) \qquad (1)$$

3.2 Affect Modeling

We extend the standard seq2seq architecture by including emotion-specific information during the training and the inference. A critical challenge in both generating and evaluating responses is a reliable assessment of emotional state. We use two representations of emotion: (1) a categorical representation with six emotions (anger, surprise, joy, sadness, fear, disgust), and (2) a continuous representation in a VAD space. The latter uses a VAD Lexicon introduced by Mohammad (2018), where each of 20k words is mapped to a 3D vector of VAD values, ranging from 0 (lowest) to 1 (highest) ($\mathbf{v} \in [0, 1]^3$). Valence measures the positivity/negativity, Arousal the excitement/calmness, and Dominance the powerfulness/weakness of the emotion expressed by a word. This expands the work of Lubis et al. (2018), who modeled emotions only in VA space. In the following sections we describe different versions of the proposed model.

3.2.1 Emotion Classifier

Affective training requires E_0, the emotion representation of the target sequence. In order to label all sentences of the corpus with E_0, we use an Emotion Classifier by Witon et al. (2018). The classifier predicts a probability distribution over class of six emotions. The classifier predictions for Cornell Movie-Dialogs Corpus (Cornell) have been shown to be highly correlated with human predictions (Witon et al., 2018).

3.2.2 Sequence-Level Explicit Encoder Model (SEE)

To explicitly generate responses with emotion, this version of the model includes an emotion embedding at the encoder side. We feed the encoder with $\mathcal{S}' = (\mathbf{e}_{\text{SEE}}, s_1, s_2, \ldots, s_{|S|})$, where $\mathbf{e}_{\text{SEE}} = \mathbf{A}_{SEE}E_0$ is an Emotion Embedding ($\mathbf{e}_{\text{SEE}} \in \mathbb{R}^3$), and $\mathbf{A}_{SEE} \in \mathbb{R}^{3\times 6}$ is a mapping (learned during training) from E_0 into an emotion embedding space.

3.2.3 Sequence-Level Explicit Decoder Model (SED)

Another way of forcing an emotional output is to explicitly indicate the target emotion at every step in decoding along with other inputs. Formally, the GRU hidden state at time t is calculated as $\mathbf{h}_t = f(\mathbf{h}_{t-1}, r'_t)$ with $r'_t = [r_{t-1}; \mathbf{e}_{\text{SED}}]$, where $\mathbf{e}_{\text{SED}}$ is defined similarly as $\mathbf{e}_{\text{SEE}}$. It is worth noting that $\mathbf{A}_{SEE}$ and $\mathbf{A}_{SED}$ are different, which implies that the emotion embedding spaces they map to are also different. Compared to a similar approach introduced by Huang et al. (2018), our solution enables the desired emotional content, E_0, to be provided in a continuous space.

3.2.4 Word-Level Implicit Model (WI)

To model the word-level emotion carried by each sequence, we introduce an Affective Regularizer (AR), which expresses the affective distance between $\mathcal{R}_{\text{final}}$ and $\mathcal{R}_0$, in the VAD space. It forces the neural network to prefer words in the vocabulary that carry emotions in terms of VAD. Math-

ematically, we extend the regular Negative Log Likelihood (NLL) loss with an affective regularizer, $\mathcal{L}_{\text{AR}}$:

$$\begin{aligned}\mathcal{L} &= \mathcal{L}_{\text{NLL}} + \mathcal{L}_{\text{AR}} \\ &= -\log p(\mathcal{R}_{\text{final}}|\mathcal{S}) + \mu\mathcal{L}_{\text{VAD}}(\mathcal{R}_{\text{final}}, \mathcal{R}_0)\end{aligned}$$

$$\mathcal{L}_{\text{VAD}}(\mathcal{R}_{\text{final}}, \mathcal{R}_0) = \left\| \sum_{t=1}^{|\mathcal{R}_{\text{final}}|} \frac{\mathbf{E}^{\text{VAD}}\mathbf{s}_t}{|\mathcal{R}_{\text{final}}|} - \sum_{t=1}^{|\mathcal{R}_0|} \frac{\mathbf{e}^{\text{VAD}}_{r_{0_t}}}{|\mathcal{R}_0|} \right\|,$$

where $\mathbf{s}_t = \text{softmax}(\mathbf{h}_t)$ ($\mathbf{s}_t \in \mathbb{R}^{|V|}$) is a confidence of the system of generating words $w_1, \ldots, w_{|V|}$ at time t and $\mu \in \mathbb{R}$. $\mathbf{e}^{\text{VAD}}_x \in \mathbb{R}^3$ is a 3D vector representing emotion associated with a word x in VAD space (note that $\mathbf{e}^{\text{VAD}}_x$ is constant with respect to t), and $\mathbf{E}^{\text{VAD}} \in \mathbb{R}^{3\times|V|}$ is a matrix containing $\mathbf{e}^{\text{VAD}}_{w_v}$ for all $|V|$ words in the vocabulary:

$$\mathbf{E}^{\text{VAD}} = \left[\mathbf{e}^{\text{VAD}}_{w_1} ; \ldots ; \mathbf{e}^{\text{VAD}}_{w_{|V|}} \right]$$

Intuitively, the regularizer penalizes the deviation of the emotional content of the generated response, $\mathcal{R}_{\text{final}}$, from the desired response, $\mathcal{R}_0$. The emotional information carried by $\mathcal{R}_{\text{final}}$ is the weighted sum of emotion representations $\mathbf{e}^{\text{VAD}}_{w_i}$ for all words w_i in the vocabulary, where the weights are determined by the confidence $\mathbf{s}_t$.

3.2.5 Word-Level Explicit Model (WE)

Sequential word generation allows sampling of the next word, based on the emotional content of the current incomplete sequence. If some words in a sequence do not express the target emotion E_0, other words can compensate for this by changing the final affective content, e.g., in a sentence "I think that the cat really loves me!", the first 6 words are neutral, whereas the end of the sentence make it clearly express joy. We incorporate this observation by explicitly generating the next word using an Adaptive Affective Sampling Method:

$$\log p(\mathcal{R}_C|\mathcal{S}, E_0) = \sum_{t=1}^{|\mathcal{R}_C|} \log p(r_t|r_{<t}, \mathbf{e}_{r_{<t}}, \mathbf{h}_{\mathcal{S}}, E_0),$$

$$p(r_t|r_{<t}, \mathbf{e}_{r_{<t}}, \mathbf{h}_{\mathcal{S}}, E_0) = \lambda \,\text{softmax}\, g(\mathbf{h}_t) + (1-\lambda)\,\text{softmax}\, v(E^{\text{VAD}}_t),$$

where $g(\mathbf{h}_t)$ is a linear mapping from GRU hidden state $\mathbf{h}_t$ to an output vector of size $|V|$, and $0 \le \lambda \le 1$ is learned during training. The first term in Equation 3.2.5 is responsible for generating words according to a language model preserving grammatical correctness of the sequence, whereas the second term forces generation of words carrying emotionally relevant content. $E^{\text{VAD}}_t \in \mathbb{R}^3$ is a vector representing the remaining emotional content needed to match a goal (E^{VAD}_0) after generating all words up to time t. It is updated every time a new word r_t with an associated emotion vector $\mathbf{e}^{\text{VAD}}_{r_t}$ is generated:

$$E^{\text{VAD}}_t = E^{\text{VAD}}_{t-1} - \mathbf{e}^{\text{VAD}}_{r_{t-1}}$$

$$E^{\text{VAD}}_0 = \begin{cases} \sum_{t=1}^{|\mathcal{R}_0|} \mathbf{e}^{\text{VAD}}_{r_{0_t}}, & \text{training} \\ \mathbf{M}_{\text{VAD}} E_0 \cdot \max_{\text{length}}, & \text{inference} \end{cases}$$

where $\mathbf{e}^{\text{VAD}}_{r_{0_t}}$ is an emotion vector associated with words r_{0_t} in the target sequence, $\max_{\text{length}}$ is a maximum length set for the seq2seq model, and $\mathbf{M}_{\text{VAD}} \in \mathbb{R}^{3\times 6}$ is a mapping from six-dimensional emotion space into VAD space (every emotion has a VAD vector as introduced by Hoffmann et al. (2012), scaled to a range [0, 1]):

$$\mathbf{M}_{\text{VAD}} = \begin{array}{c} \begin{array}{cccccc} \text{anger} & \text{surprise} & \text{joy} & \text{sadness} & \text{fear} & \text{disgust} \end{array} \\ \left[\begin{array}{cccccc} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0.5 \\ 1 & 0 & 1 & 0 & 0 & 0.5 \end{array}\right] \end{array} \begin{array}{c} \\ V \\ A \\ D \end{array}$$

$v(E^{\text{VAD}}_t)$ is a vector, whose i-th component measures the potential remaining emotional content of the sequence in the case of choosing the i-th word w_i:

$$v(E^{\text{VAD}}_t) = -\begin{bmatrix} \left\| E^{\text{VAD}}_t - \mathbf{e}_{w_0} \right\| \\ \cdots \\ \left\| E^{\text{VAD}}_t - \mathbf{e}_{w_{|V|}} \right\| \end{bmatrix}$$

In the following, we set a constant $\lambda = 1$ after generating the first $\max_{\text{length}}/2$ words, as this setting ensures that the first generated words carry the right emotional content, while not sacrificing the grammatical correctness of the whole response. This leads to an improvement in performance.

3.3 Affective Inference

The methods described in the previous sections aim to improve the seq2seq training/sampling procedure. We hypothesize that a good inference strategy is crucial for generating diverse and emotion-specific responses. As Li et al. (2016) suggest,

traditional objective functions, i.e., likelihood of a response given an input, can be improved by using an N-best list and MMI during inference. We build upon this idea; our hypothesis is that by generating B diverse sequences and re-ranking the responses, we are more likely to infer one best emotion-specific response. The B-best list is found using Beam Search of size B with length normalization.

In the MMI-bidi setting, Li et al. (2016) rank all responses found during beam search based on a score calculated as:

$$\mathcal{R}_{\text{final}} = \underset{\mathcal{R}_C}{\operatorname{argmax}}\, p(\mathcal{R}_C|\mathcal{S}) + \alpha p(\mathcal{S}|\mathcal{R}_C) + \beta|\mathcal{R}_C|, \tag{2}$$

where $p(\mathcal{S}|\mathcal{R}_C)$ is a model with the same architecture as $p(\mathcal{R}_C|\mathcal{S})$ trained on reversed prompt-response pairs (T-S), and $|R_C|$ is the length of the candidate response, R_C. We modify this objective in the following form:

$$\mathcal{R}_{\text{final}} = \underset{\mathcal{R}_C}{\operatorname{argmax}}\, p(\mathcal{R}_C|\mathcal{S}, E_0) + \alpha p(\mathcal{S}|\mathcal{R}_C) + \beta|\mathcal{R}_C| - \gamma \left\| E_{\mathcal{R}_C} - E_0 \right\|, \tag{3}$$

where the last term penalizes the deviation of the emotional content, $E_{\mathcal{R}_C}$, of the candidate response, $\mathcal{R}_C$, from the desired emotional content, E_0. The task is to find optimal values of parameters α, β and γ, which give the best responses in terms of grammatical correctness, diversity (α, β) and emotional content (γ) (see §5 and §6).

4 Model Training

In this section, we describe corpora used for training, the baseline models and the training procedure for the models presented in §3.

4.1 Corpora

Cornell contains around 10K movie characters and around 220K dialogues (Danescu-Niculescu-Mizil and Lee, 2011).

OpenSubtitles2018 is a collection of translated movie subtitles with 3.35G sentence fragments (Tiedemann, 2009). It has been filtered to get pairs of consecutive sequences (containing between 5 and 30 words), with respective timestamps within an interval of 5 seconds, that are part of a conversation of at least 4 turns. The filtered dataset contains 2.5M utterances.

Preprocessing Each dataset is tokenized using the spaCy[1] tokenizer, converted to lowercase, and non-ASCII symbols are removed. To restrain the vocabulary size and correct the typos, we use a default vocabulary of fixed size 42K words from spaCy. Each word in the dataset is then compared with the vocabulary using the difflib library[2] in Python (algorithm based on the Levenshtein distance), and mapped to the most similar word in the vocabulary. If no word with more than 90% of similarity is found, the word is considered a rare word or a typo, and is mapped to the out-of-vocabulary (OOV) word. For Cornell, less than 1% of the unigrams are OOV.

4.2 Affective Dictionary

The VAD lexicon may not have all the words in the vocabulary. Based on the word similarity (using difflib library), each word of the vocabulary is assigned a VAD value of the most similar word in the VAD lexicon. If no word with more than 90% of similarity is found, a "neutral" VAD value ($\mathbf{v} = [0.5, 0.5, 0.5]$) is assigned.

4.3 Baselines

We compare our work to two different baselines: a vanilla seq2seq and the ECM introduced by Zhou et al. (2018). For the external memory we use our affective dictionary and train the model using the default parameters provided by authors.

4.4 Training Details

All the hyper-parameters have been optimized on the validation set using BLEU score (Papineni et al., 2002). For the encoder, we use two-layer bidirectional GRUs (hidden size of 256). The final hidden states from both directions are concatenated and fed as an input to the decoder of one-layer unidirectional GRUs (hidden size of 512). The embedding layer is initialized with pre-trained word vectors of size 300 (Mikolov et al., 2018), trained with subword information (on Wikipedia 2017, UMBC web-base corpus and statmt.org news dataset), and updated during training. We use ADAM optimizer (Kingma and Ba, 2014) with a learning rate of 0.001 for learning $p(\mathcal{R}_C|\mathcal{S}, E_0)$ (resp. 0.01 for $p(\mathcal{S}|\mathcal{R}_C)$), which is updated by using a scheduler with a patience of 20 epochs and a decreasing rate of 0.5. The gradient norm is clipped to 5.0, weight decay is set to $1e^{-5}$, and dropout (Srivastava et al., 2014) is set to 0.2. The maximum sequence length is set to 20 for Cornell and to 30 for OpenSubtitles. The models have been trained on 94%, validated on 1%, and tested on 5% of the data.

[1] https://spacy.io

[2] https://docs.python.org/3/library/difflib.html

	Model	C distinct-1	C distinct-2	OS distinct-1	OS distinct-2	C BLEU	OS BLEU
No re-rank	Baseline	0.0305	0.1402	0.0175	0.1205	0.0096	0.094
	ECM	0.0310	0.1412	0.0180	0.1263	0.0099	0.099
	SEE	0.0272	0.1331	0.0170	0.1100	0.0110	0.093
	SED	0.0303	0.1502	0.0189	0.1231	0.0128	0.103
	WI	0.0316	0.1480	0.0175	0.1235	0.0129	0.100
	WE	0.0310	0.1400	0.0195	0.1302	0.0098	0.095
	WI + WE	**0.0342**	**0.1530**	**0.0198**	**0.1300**	**0.0108**	**0.105**
		(+12.1%)	**(+9.1%)**	**(+13.1%)**	**(+7.9%)**	**(+12.5%)**	**(+11.7%)**
Re-rank	$\text{MMI}_{baseline}$	0.0379	0.1473	0.0200	0.1403	0.0130	0.105
	$\text{EMOTICONS}_{\gamma=0}$	**0.0406**	**0.2030**	**0.0305**	**0.1431**	**0.0140**	**0.110**
		(+7.1%)	**(+37.8%)**	**(+52.5%)**	**(+2.0%)**	**(+7.7%)**	**(+4.8%)**

Table 2: Quantitative results: Results for all proposed models trained on Cornell (C) and OpenSubtitles (OS). distinct-1 and distinct-2 count the number of distinct unigrams and bigrams, respectively, normalized by the total number of generated tokens in 200 candidate responses. The performance boost is computed with respect to the vanilla seq2seq model.

5 Quantitative Evaluation for Model Selection

To evaluate language models, we use BLEU score (computed using 1- to 4-grams), as it has been shown to correlate well with human judgment (Agarwal and Lavie, 2008). Perplexity does not provide a fair comparison across the models: during the training of the baseline seq2seq model, we minimize the cross entropy loss (logarithm of perplexity), whereas in other models (e.g., WI) we aim to minimize a different loss not directly related to perplexity (cross entropy extended with the affective regularizer). Having more diverse responses makes the affective re-ranking more efficient, to evaluate diversity we count the number of distinct unigrams (distinct-1) and bigrams (distinct-2), normalized by the total number of generated tokens.

The performance of different models introduced in §3 are presented in Table 2. $\text{MMI}_{\text{bas.}}$ refers to a system that re-ranks responses based on Equation 2, where both $p(\mathcal{R}_C|\mathcal{S})$ and $p(\mathcal{S}|\mathcal{R}_C)$ are baseline seq2seq models. EMOTICONS is a system based on Equation 3, where $p(\mathcal{R}_C|\mathcal{S}, E_0)$ is computed using a composition of Word-Level Implicit Model (WI) and Word-Level Explicit Model (WE), and $p(\mathcal{S}|\mathcal{R}_C)$ is computed using WI (as we are not interested in explicitly using the input emotion). We optimize α and β on the validation set using BLEU score, since Li et al. (2016) have shown that adding MMI during inference improves the BLEU score. We set $\gamma = 0$ and find optimal values $\alpha_{opt} = 50.0$ and $\beta_{opt} = 0.001$ using grid search.

Improving BLEU score and diversity was not the goal of our work, but the observed improvement (after adding emotions) shows that the different systems are able to extract and use emotional patterns to improve the general language model.

5.1 Response Diversity

From Table 2, we observe that for both Cornell and OpenSubtitles datasets, SED, WI, and WE models outperform the vanilla seq2seq and the ECM for at least one of the two distinct measures. SEE has the worst performance overall and does not compete with either the baseline, nor with SED. This is expected according to the results reported by Huang et al. (2018). It seems that the model is not able to capture the information carried by the additional emotion embedding token – it is treated as just one additional word among 20 others. SED makes better use of the emotion information, as it is used at each time step during decoding. In addition, it is more natural to use these features during the decoding, since the emotion embedding represents the desired emotion of the response. The combination of WI and WE performs best in terms of distinct-1 and distinct-2 measures among all models without re-ranking, yielding an improvement of up to 13.1%. It suggests that the word level emotion models suit the seq2seq architecture better. During training, both models are encouraged not only to match the target words, but also to promote less frequent words that are close to the target words in terms of VAD values (affective regularizer and affective sampling), fostering the model to generate more diverse responses.

As expected, by adding MMI, we observe an

improvement in diversity, but the relative improvement for OpenSubtitles ($\text{MMI}_{\text{bas.}}$) is smaller than the one reported by Li et al. (2016). This could originate from the different data filtering and beam search strategy, and the fact the hyper-parameter optimization has been performed on Cornell. EMOTICONS is a combination of WI + WE (best performing model) for $p(\mathcal{R}_C|\mathcal{S}, E_0)$ and WI for $p(\mathcal{S}|\mathcal{R}_C)$, it is better than $\text{MMI}_{\text{bas.}}$ (up to 52.5% gain in distinct-1).

It is worth noting that we observe higher scores in terms of diversity for the reversed model $p(\mathcal{S}|\mathcal{R}_C)$ compared to the normal model $p(\mathcal{R}_C|\mathcal{S}, E_0)$, while training on Cornell. We can explain this using the data distribution: distinct-2 is higher for the questions than for the answers (0.167 and 0.154 for Cornell, respectively).

5.2 Response Quality

Table 2 shows that, in general, introducing emotional features into the process of generating responses does not reduce the BLEU score. To reduce the potential negative impact of choosing inappropriate first words in the sequence, we compute the BLEU score on the result of beam search of size 200. For example, if the first word is "I", the seq2seq models tend to generate a response "I don't know" with high probability, due to the high number of appearances of such terms in the training set. In certain cases, like WI and SED, we observe an improvement. Such an improvement is expected, since our model takes into account additional (affective) information from the target sequence during response generation.

6 Human-in-the-Loop Hyper-Parameter Estimation

The quantitative evaluation shows that EMOTICONS outperforms the baseline while adding the emotional features during response generation. The re-ranking phase did not take into account the affective term ($\gamma = 0$ in Equation 3). Setting a different value would not necessarily improve any of the available metrics (e.g., BLEU score, diversity), as they do not explicitly take into account affective content in their definition. In this section, we describe an optimization procedure, relying on human judgment, for finding the optimal value of γ.

6.1 Experiment Description

We asked annotators to evaluate (using an AffectButton) the generated responses. We use AffectButton (Broekens and Brinkman, 2013), a reliable affective tool for assigning emotions, which, to our knowledge, has never been used for estimating the emotional content of the generated responses. In our experiment, the AffectButton lets users choose a facial expression from a continuous space (see Figure 3), that best matches the emotional state associated with the sequence, which is then mapped into the VAD space. In order to conduct the experiment, we chose a pool of 12 annotators, who annotated a total of 400 sequences. The prompts were randomly chosen from the test set of Cornell, among the 200 sequences that create the most diverse responses in terms of distinct-2. The more diverse the responses are, the more likely we are to select a response carrying a desired emotion. The responses for the prompts were generated using EMOTICONS where the target emotion was either fear, anger, joy, or surprise; the four corners of the AffectButton. γ was randomly chosen among 20 uniformly sampled values in $[0, 10]$.

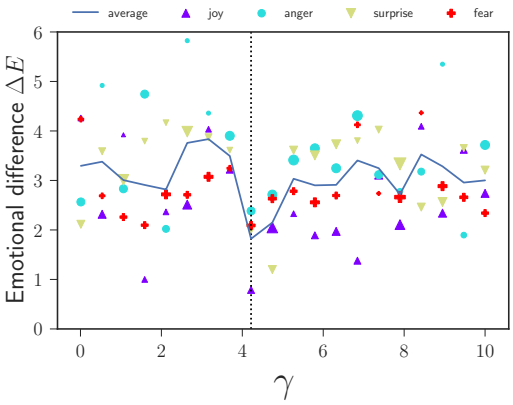

Figure 2: Hyper-parameter optimization: For different values of γ, users assign a face to the generated response. Each point represents an average ΔE of annotations for each emotion. ΔE is the difference between the VAD representation of the face assigned by the user and the desired emotion for the response. The size is proportional to the number of collected annotations. ΔE is at a minimum for $\gamma_{\text{opt}} = 4.2$.

6.2 Experiment Results

In Figure 2, we present the difference between the VAD value according to the face assigned by the user, and the desired emotion for the response. The average curve presents a global minimum at $\gamma_{opt} = 4.2$. The system does not perform equally well at generating different emotions according to the human judgment. On average, we observe lower values for joy compared to anger in Figure 2. This phenomenon is expected, as in the re-ranking

	Model	Grammatical Correctness	User Preference Total	User Preference Majority Vote
	$\text{MMI}_{\text{bas.}}$	83 %	39	8
EMOTICONS	Fear	82 %	96	37
EMOTICONS	Anger	80 %		
EMOTICONS	Joy	84 %		
EMOTICONS	Surprise	79 %		

Table 3: User study results: *Grammatical Correctness* shows the ratio of grammatically correct sentences among all generated responses, whereas *User Preference* shows the number of times each model was preferred by the users.

process $E_{\mathcal{R}_C}$ is estimated using the emotion classifier (Witon et al., 2018) which detects joy more accurately than anger (77% versus 57%), surprise (62%) and fear (69%).

7 Qualitative Evaluation

In this section, we qualitatively evaluate the emotional content and correctness of the responses generated by $\text{EMOTICONS}_{\gamma=\gamma_{opt}}$ compared to the ones from $\text{MMI}_{\text{bas.}}$ through a user study. It consists of three different experiments which measure grammatical correctness, user preference, and emotional appropriateness. For all experiments, we chose prompts from the test set of Cornell, for which the most diverse responses were created by $\text{MMI}_{\text{bas.}}$ in terms of distinct-2. We test EMOTICONS by generating responses according to four emotions: fear, anger, joy, and surprise (beam size of 200).

7.1 Grammatical Correctness

In this experiment, we used 40 prompts. For each prompt, we generated 5 sentences (4 for EMOTICONS, and 1 for $\text{MMI}_{\text{bas.}}$) that were presented in a random order to 3 native English speakers. They assigned either 0 (sentence grammatically incorrect), or 1 (sentence grammatically correct) for all sentences. To measure the agreement across annotators, we calculate Fleiss' $\kappa = 0.4128$, which corresponds to "moderate agreement". Our model does not substantially sacrifice the grammatical correctness of the responses (see Table 3).

7.2 User Preference

In this setting, we quantify how likely the user is going to prefer the response generated by EMOTICONS compared to the one generated by $\text{MMI}_{\text{bas.}}$. We asked 18 annotators to choose their favorite response to the input query among eight proposed answers (top four responses coming from the MMI_{bas} and 4 coming from EMOTICONS with the four different target emotions). Each of 45 sentences were annotated by three different annotators. Results of the experiment (Table 3) indicate that users strongly prefer EMOTICONS over $\text{MMI}_{\text{bas.}}$.

(a) Bas. (b) Fear (c) Anger (d) Joy (e) Surprise

Figure 3: Emotional faces: AffectButton presents faces according to average VAD vectors (in parentheses) obtained for the (a) $\text{MMI}_{\text{bas.}}$ ($[0.47, 0.98, 0.36]$), and for the four EMOTICONS models with different target emotions: (b) Fear ($[0.2, 0.95, 0.38]$), (c) Anger ($[0.54, 0.92, 0.65]$), (d) Joy ($[0.68, 0.97, 0.66]$), and (e) Surprise ($[0.37, 0.97, 0.52]$).

7.3 Emotional Appropriateness

In this experiment, we show that our model is able to generate emotions in a controlled manner. For each of the 5 models, 22 users assign a face via the AffectButton. We generate responses for 120 different prompts. We keep the responses that were annotated with a VAD vector with the norm greater than 2, corresponding to those expressing strong emotions. We compute the average VAD vectors for the annotated sequences for each model, with corresponding AffectButton faces (Figure 3). The majority of user-assigned faces have a high arousal value, which can be explained by the fact that users tend to click in one of the four corners of the AffectButton. The majority of the faces represent an accurate portrayal of the desired emotion. The poor performance of EMOTICONS at expressing surprise comes from the fact that (1) users often mismatch surprise with joy, leading to a neutral dominance value, and (2) surprise is one of the most difficult emotions to judge (see §6).

8 Conclusion

We have presented EMOTICONS, a system that can generate responses with controlled emotions. The flexibility of the presented solution allows it to be used in any kind of neural architecture as long it fits the encoder-decoder framework. Currently, EMOTICONS does not generate different emotions equally well. Future work could include incorporating contextual information that would help EMOTICONS to better capture emotional content.

Acknowledgments

We would like to thank anonymous reviewers for their insightful comments. Mubbasir Kapadia has been funded in part by NSF IIS-1703883, NSF S&AS-1723869, and DARPA SocialSim-W911NF-17-C-0098.

References

Abhaya Agarwal and Alon Lavie. 2008. Meteor, m-bleu and m-ter: Evaluation metrics for high-correlation with human rankings of machine translation output. In *Proc. of the Third Workshop on Statistical Machine Translation*, pages 115–118.

Nabiha Asghar, Pascal Poupart, Jesse Hoey, Xin Jiang, and Lili Mou. 2017. Affective neural response generation. *CoRR*, abs/1709.03968.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Joost Broekens. 2012. In defense of dominance: PAD usage in computational representations of affect. *International Journal of Synthetic Emotions*, 3(1):33–42.

Joost Broekens and Willem-Paul Brinkman. 2013. Affectbutton: A method for reliable and valid affective self-report. *International Journal of Human-Computer Studies*, 71(6):641–667.

Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A survey on dialogue systems: Recent advances and new frontiers. *ACM SIGKDD Explorations Newsletter*, 19(2):25–35.

Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*, abs/1412.3555.

Cristian Danescu-Niculescu-Mizil and Lillian Lee. 2011. Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs. In *Proc. of the Workshop on Cognitive Modeling and Computational Linguistics, ACL*.

Paul Ekman, Robert W Levenson, and Wallace V Friesen. 1983. Autonomic nervous system activity distinguishes among emotions. *Science*, 221(4616):1208–1210.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. *CoRR*, abs/1805.04833.

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

Holger Hoffmann, Andreas Scheck, Timo Schuster, Steffen Walter, Kerstin Limbrecht, Harald C Traue, and Henrik Kessler. 2012. Mapping discrete emotions into the dimensional space: An empirical approach. In *Proc. of IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3316–3320. IEEE.

Chenyang Huang, Osmar Zaiane, Amine Trabelsi, and Nouha Dziri. 2018. Automatic dialogue generation with expressed emotions. In *Proc. of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 2 (Short Papers), pages 49–54.

Vladimir Ilievski, Claudiu Musat, Andreea Hossmann, and Michael Baeriswyl. 2018. Goal-oriented chatbot dialog management bootstrapping with transfer learning. In *Proc. of IJCAI*.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

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

Nurul Lubis, Sakriani Sakti, Koichiro Yoshino, and Satoshi Nakamura. 2018. Eliciting positive emotion through affect-sensitive dialogue response generation: A neural network approach. In *Proc. of AAAI Conference on Artificial Intelligence*.

Irina Maslowski, Delphine Lagarde, and Chloé Clavel. 2017. In-the-wild chatbot corpus: from opinion analysis to interaction problem detection. In *Proc. of ICNLSP*.

Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. Advances in pre-training distributed word representations. In *Proc. of the International Conference on Language Resources and Evaluation*.

Saif M. Mohammad. 2018. Obtaining reliable human ratings of valence, arousal, and dominance for 20,000 English words. In *Proc. of the 56th Annual Meeting of the Association for Computational Linguistics*, Melbourne, Australia.

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

Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C. Courville, and Joelle Pineau. 2015. Hierarchical neural network generative models for movie dialogues. *CoRR*, abs/1507.04808.

Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. *CoRR*, abs/1506.06714.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Sarah Strohkorb, Chien-Ming Huang, Aditi Ramachandran, and Brian Scassellati. 2016. Establishing sustained, supportive human-robot relationships: Building blocks and open challenges. In *AAAI Spring Symposium on Enabling Computing Research in Socially Intelligent Human-Robot Interaction*, volume 2123, page 2016.

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

Jörg Tiedemann. 2009. News from OPUS - A collection of multilingual parallel corpora with tools and interfaces. In *Recent Advances in Natural Language Processing*, volume 5, pages 237–248.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proc. of Advances in neural information processing systems*.

Oriol Vinyals and Quoc V. Le. 2015. A neural conversational model. *CoRR*, abs/1506.05869.

Wojciech Witon, Pierre Colombo, Ashutosh Modi, and Mubbasir Kapadia. 2018. Disney at IEST 2018: Predicting emotions using an ensemble. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 248–253.

Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2018. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *Proc. of The AAAI Conference on Artificial Intelligence*, pages 730–738.