Improving Neural Response Diversity with Frequency-Aware Cross-Entropy Loss

Shaojie Jiang
University of Amsterdam
Amsterdam, The Netherlands
s.jiang@uva.nl

Pengjie Ren
University of Amsterdam
Amsterdam, The Netherlands
p.ren@uva.nl

Christof Monz
University of Amsterdam
Amsterdam, The Netherlands
c.monz@uva.nl

Maarten de Rijke
University of Amsterdam
Amsterdam, The Netherlands
derijke@uva.nl

ABSTRACT

Sequence-to-Sequence (Seq2Seq) models have achieved encouraging performance on the dialogue response generation task. However, existing Seq2Seq-based response generation methods suffer from a *low-diversity problem*: they frequently generate generic responses, which make the conversation less interesting. In this paper, we address the low-diversity problem by investigating its connection with *model over-confidence* reflected in predicted distributions. Specifically, we first analyze the influence of the commonly used Cross-Entropy (CE) loss function, and find that the CE loss function prefers high-frequency tokens, which results in low-diversity responses. We then propose a Frequency-Aware Cross-Entropy (FACE) loss function that improves over the CE loss function by incorporating a weighting mechanism conditioned on token frequency. Extensive experiments on benchmark datasets show that the FACE loss function is able to substantially improve the diversity of existing state-of-the-art Seq2Seq response generation methods, in terms of both automatic and human evaluations.

CCS CONCEPTS

• **Information systems → Users and interactive retrieval**.

KEYWORDS

Chatbot; Dialogue system; Sequence-to-sequence model

ACM Reference Format:
Shaojie Jiang, Pengjie Ren, Christof Monz, and Maarten de Rijke. 2019. Improving Neural Response Diversity with Frequency-Aware Cross-Entropy Loss. In *Proceedings of the 2019 World Wide Web Conference (WWW'19), May 13–17, 2019, San Francisco, CA, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3308558.3313415

1 INTRODUCTION

Recently, dialogue response generation has attracted a lot of attention due to its potential for applications, e.g., within intelligent customer service agents and personal assistants. Most state-of-the-art approaches to this task are based on *Sequence-to-Sequence* (Seq2Seq) frameworks [3, 11–13, 17, 19]. Current Seq2Seq frameworks suffer from a *low-diversity problem.* As a result, these approaches frequently generate generic responses such as "*I don't know*" or "*I'm sorry*" [3, 11, 12, 19].

To address the issue of low response diversity, previous studies have produced several hypotheses about the cause of the low-diversity problem with corresponding solutions. For example, Li et al. [3] argue that the usual max a posteriori (MAP) objective function might favor frequent responses. Instead, they propose to use a Maximum Mutual Information (MMI) objective function to encourage responses with higher mutual information regarding the user's utterance. The downside of this method is that it relies heavily on beam search and/or an inverse model that is trained by swapping inputs and outputs. Tao et al. [15] assume that the single attention layer commonly used in Seq2Seq models is only able to focus on a single semantic aspect of the input sequence, so they propose a Multi-Head Attention Mechanism (MHAM) to allow the decoder to generate more diverse responses. There are also studies that try to improve response diversity by introducing randomness [12, 19].

Recently, Jiang and de Rijke [1] have shown that there is a strong connection between the low-diversity problem and what they call *model over-confidence* following Pereyra et al. [10], the phenomenon that a model incorrectly assigns most probability to only a few tokens. However, the cause of this phenomenon remains unknown. In §2, we investigate and conclude that the model over-confidence problem is caused by an imbalanced training of tokens with variable frequencies, which favors frequent tokens and results in low-diversity. To correct for this, we propose a *Frequency-Aware Cross-Entropy* (FACE) loss function that improves the traditional *Cross-Entropy* (CE) loss function by taking token frequency into consideration. More specifically, we first analyze the influence of the commonly used CE loss function, and find that it prefers high-frequency tokens, which results in model over-confidence and low-diversity responses. Then we propose a FACE loss function that improves over the CE loss function by incorporating a weighting mechanism conditioned on token frequency.

The primary differences between FACE and previous studies in addressing the low-diversity problem are two-fold: (1) The diversity improvements brought by FACE do not rely on beam search or

WWW '19, May 13–17, 2019, San Francisco, CA, USA

ACM ISBN 978-1-4503-6674-8/19/05.
https://doi.org/10.1145/3308558.3313415

Table 1: Frequency ranks of leading tokens and their percentage (%). Validation rank and Test rank are for the validation and test model responses, Training rank is for the ground-truth responses in the training data.

	i	the	you	we	he	it
Validation rank	1 (74)	2 (5)	3 (5)	4 (3)	5 (3)	6 (3)
Test rank	1 (72)	2 (7)	3 (5)	5 (3)	6 (3)	4 (2)
Training rank	1 (14)	7 (12)	3 (6)	6 (3)	8 (3)	4 (3)

randomness; and (2) FACE does not introduce new layers or hyper-parameters to Seq2Seq models.

We perform extensive experiments on two benchmark datasets, namely OpenSubtitles database (OSDb) and Twitter. Compared with deterministic Seq2Seq-based methods, like MMI [3] and MHAM [15], FACE achieves the highest diversity performance and it does so with minimum modifications to the original Seq2Seq model structure and existing hyper-parameters.[1]

The main contributions we make in this paper are:

- We examine the influence of token frequency on model over-confidence and response diversity.
- We propose a *Frequency-Aware Cross-Entropy* (FACE) loss function to balance the per-token loss, which alleviates model over-confidence and, hence, improves response diversity.
- We investigate two token frequency calculation methods and corresponding frequency-based weighting mechanisms for FACE.

2 LOW DIVERSITY, MODEL OVER-CONFIDENCE AND LOSS IMBALANCE

As illustrated by Jiang and de Rijke [1], model over-confidence [10] can cause Seq2Seq-based conversation models to have low response diversity. In this section, we show that model over-confidence and low-diversity are statistical and empirical symptoms of the same problem. Imbalanced training is the actual underlying reason.

Existing Seq2Seq-based conversation models tend to be over-confident during prediction and place most probability on only a few tokens. We define a *leading* token in a response to be a token that appears first. From Table 1, we can see that all of the most frequent leading tokens in model responses are actually very likely to appear first in training ground-truth responses, but the percentages of the top-ranked token "i" are much higher in the validation and test sets. This suggests that the model is over-fitting for frequent leading tokens. We also find that the frequency of some subsequent tokens (e.g., *'m* and *not*) is much higher in model responses than in the training ground truth, as illustrated in Table 2. This is also due to model over-fitting, since these tokens are *relatively* more frequent compared to other tokens. For example, given that the previous token is "i", the frequency of the following token "'m" is much higher than others in the training data.

To understand this phenomenon and introduce our solution, we first look into the commonly used *max a posteriori* (MAP) objective function of Seq2Seq models. Given a dataset of message-response pairs (X, Y), where $X = (x_1, x_2, \dots, x_{|X|})$ and $Y = (y_1, y_2, \dots, y_{|Y|})$

Table 2: Frequency ranks of example tokens and their percentage (%) in validation and test model outputs, and ground truth outputs of the training data.

	.	i	'm	not	n't
Validation rank	1 (11)	2 (10)	3 (5)	4 (5)	8 (3)
Test rank	1 (10)	2 (9)	3 (5)	4 (5)	8 (3)
Training rank	1 (7)	4 (3)	35 (1)	32 (1)	13 (1)

are the input and output sequences, respectively, the goal of Seq2Seq training is to maximize the conditional probability $P(Y|X)$. Since the decoder Recurrent Neural Network (RNN) can only give one output at each time step t, and to generate grammatical responses, we are actually maximizing token-wise probability at each time step during training:

$$\max P(Y|X) = \max \prod_{t=1}^{|Y|} P(y_t|y_{<t}, X), \tag{1}$$

where $y_{<t} = (y_1, y_2, \dots, y_{t-1})$ are tokens generated in previous time steps. At test time, the response is generated with respect to:

$$\hat{Y} = \arg\max_Y P(Y|X). \tag{2}$$

In practice, we usually maximize the aforementioned conditional probability by minimizing the prediction loss at each step t, which is actually the Cross-Entropy (CE) loss:

$$\mathrm{CE}(y_t) = -\sum_{i=1}^{N} \delta_i(y_t) \log(P(c_i|y_{<t}, X)), \tag{3}$$

where $(c_1, c_2, \dots, c_N)$ is the search space of y_t, $\delta_i(y_t) = 1$ if $y_t = c_i$ and 0 otherwise; $P(c_i|\cdot)$ is the predicted probability of candidate token c_i and is calculated using the softmax function:

$$P(c_i|y_{<t}, X) = \frac{\exp(f_\theta(h_{t-1}^{dec}, y_{t-1}, c_i, X))}{\sum_{j=1}^{N} \exp(f_\theta(h_{t-1}^{dec}, y_{t-1}, c_j, X))}, \tag{4}$$

where $f_\theta(\cdot)$ is a non-linear scoring function with parameters θ; $f_\theta(\cdot)$ takes the hidden state h_{t-1}^{dec}, last generation y_{t-1} and X as inputs, and calculates a score for each possible candidate c_i.

During training, the loss calculated using Eq. (3) is back-propagated through the whole network. The effect is that f_θ will assign higher scores for $c_i = y_t$ in the future, so that the loss in Eq. (3) will decrease, and meanwhile the predicted probability in Eq. (4) will increase. The ultimate effect is to maximize the probabilities of ground-truth outputs given input sequences, as illustrated in Eq. (1). The model is frequently penalized by a small number of frequent tokens, as a result of which the total loss (TL) for these tokens is higher than for less-frequent ones:

$$\mathrm{TL}(c_i) = \sum_{t=1}^{N_t} \mathrm{CE}(y_t = c_i), \tag{5}$$

where t denotes the time step, with maximum training steps N_t:

$$N_t = \sum_{i=1}^{N} \mathrm{freq}(c_i). \tag{6}$$

Here, $\mathrm{freq}(c_i)$ represents the frequency that c_i appears in the training ground truth. Assume that the expected value $\mathbb{E}[\mathrm{CE}(c_i)]$ is roughly the same for $c_i, \forall i \in \{1, 2, \dots, N\}$, then tokens with a higher frequency will have a larger total loss during training. We refer to this phenomenon as *loss imbalance*.

Due to loss imbalance, a Seq2Seq model favors frequent tokens and thus is over-confident about them. This is especially true for the leading token as observed in Table 1, since the decoder language

[1] Source code for both FACE and the baselines, together with the validation and test sets used in our experiments can be found at https://github.com/ShaojieJiang/FACE.

model does not have a strong effect in the beginning of prediction. When a frequent token is selected as leading token, the search space for subsequent tokens is hugely restricted by the language model, and this will likely result in a frequent *generic* response.

3 FREQUENCY-AWARE CROSS-ENTROPY LOSS

Now that we have identified *loss imbalance* as a cause of low-diversity problems, we propose to balance the total loss for each token by applying a weight factor to TL:

$$\mathrm{WTL}(c_i) = w_i \sum_{t=1}^{N_t} CE(y_t = c_i), \tag{7}$$

where w_i is the weight corresponding to c_i. By absorbing w_i into the CE loss function, we obtain the FACE loss function:

$$\mathrm{FACE}(y_t) = -\sum_{i=1}^{N} w_i \delta_i(y_t) \log(P(c_i | y_{<t}, X)). \tag{8}$$

Our key solution to the low-diversity problem lies in the weight factors w_i. Based on the analysis in §2, one straightforward way to learning w_i is to take advantage of the token frequency $\mathrm{freq}(c_i)$, so that frequent tokens will have lower weights. Below we propose two methods to estimate $\mathrm{freq}(c_i)$: *ground-truth or GT frequency* and *output frequency.*

GT frequency: Token frequency in the ground-truth responses. As illustrated in Eq. (6), the number of training steps N_t equals the sum of frequencies of tokens in the training ground-truth responses, so it is intuitive to use these token frequencies to adjust the weight in Eq. (8). However, during training, the model is given data sequentially in random order. As a result, the real-time token frequency seen by the model is likely to differ from that of the entire training data. Our solution is to calculate the *batch* token frequency instead:

$$\mathrm{freq}_b(c_i) = \mathrm{freq}_{b-1}(c_i) + \mathrm{freq}_o(c_i), \tag{9}$$

where b is the number of training batches seen so far and o represents the newly *observed* batch.

Output frequency: Token frequency in model responses. Alternatively, we can employ a train-and-refine strategy: the responses of a pre-trained Seq2Seq model can reflect which tokens the model is already overfitted for; by directly penalizing those tokens with a fine-tuning procedure, we can improve the response diversity without retraining it from scratch. The output frequency may have a more obvious effect on improving response diversity than the GT frequency, because diversity is directly exhibited by model outputs.

3.1 Weight calculation

Given the frequency of each token, we introduce the following two methods to calculate the weight factor.

3.1.1 *Pre-weight.*

This method derives the weight factor w_i prior to seeing new training examples, i.e., *pre-weight function*:

$$w_i = a \times RF_i + 1. \tag{10}$$

Here we formulate it as a linear function of *relative frequency*: $RF_i = \mathrm{freq}(c_i) / \sum_j \mathrm{freq}(c_j)$, and $a = -1/\max RF_j, \forall j \in \{1, \ldots, N\}$, is the slope, and the bias is 1 so that w_i falls in $[0, 1]$. We then normalize $\{w_1, w_2, \ldots, w_N\}$ to have a mean of 1. The pre-weight function can make sure that tokens with a higher RF value will get lower weights. In other words, the influence of high frequency tokens will be penalized by using the pre-weight function.

3.1.2 *Post-weight.*

This method tries to penalize the model's *conservativeness*: if the output token y_t has a higher frequency than the ground truth c_i, which indicates the model conservatively picked a "safe" token, then its loss will be scaled up by $w_i > 1$, otherwise $w_i = 1$ due to ReLU activations:

$$w_i = 1 + \frac{ReLU(\mathrm{freq}(y_t) - \mathrm{freq}(c_i))}{\sum_j^N \mathrm{freq}(c_j)}. \tag{11}$$

Since we can only apply this weighting function after obtaining the model outputs, we refer to it as the *post-weight function.*

3.2 Empowered by confidence penalty

As illustrated in [1], *Confidence Penalty* (CP) methods can alleviate the low-diversity problem. In the experiments of this paper, we also test the performance of the CP function [1]:

$$\mathrm{CP}(y_t) = \mathrm{CE}(y_t) - \beta H(p(y_t | y_{<t}, X)), \tag{12}$$

where $p(y_t | y_{<t}, X)$ is the predicted distribution at t, and $H(\cdot)$ is its entropy. However, during experiments we found that the parameter β needs to be carefully chosen, otherwise the loss $\mathrm{CP}(y_t)$ will be negative, which is counter-intuitive since losses should be greater than 0. Instead, we propose a *parameter-free* CP function:

$$\mathrm{CP}_{free}(y_t) = \mathrm{CE}(y_t) + \frac{1}{H(p(y_t | y_{<t}, X))}. \tag{13}$$

FACE and CP can be easily combined to further improve response diversity. A trivial combination is to replace the CE loss function with the FACE loss function in Eq. (12) and Eq. (13). We also propose a CP weighting function using the entropy in Eq. (13):

$$w = 1 + \frac{1}{H(p(y_t | y_{<t}, X))}, \tag{14}$$

where w (without subscript) is assumed to be independent of c_i, which is different from that in Eq. (10) and Eq. (11). By using w as the weight of FACE in Eq. (8), we can penalize the model confidence by adjusting the weight of FACE.

4 EXPERIMENTAL SETUP

To prove the effectiveness of our proposed methods, we design experiments to answer the following questions: (Q1) Which combination of our proposed frequency methods (GT frequency and output frequency) and weighting functions (pre- and post-weight) performs best? (Q2) Does FACE improve the diversity of Seq2Seq conversation models? (Q3) Does CP improve the diversity of Seq2Seq conversation models? (Q4) Does the combination of FACE and CP further improve performance? (Q5) Does FACE improve the response quality besides diversity?

4.1 Baselines and datasets

The first baseline is a vanilla Seq2Seq model with *general* attention [6] as implemented within the ParlAI platform [7]. We also choose the MMI models proposed in [3] and the MHAM models proposed in [15]. These models are all deterministic methods. In Table 3, we

[2] The MMI-bidi method needs a reverse model which is pre-trained using inverse training examples: (response, message).

Table 3: Components of different models.

Model	Greedy	BS	#Attn	Extras
Seq2Seq	Yes	No	1	No
MMI	No	Yes	1	Reverse[2]
MHAM	Yes	No	5	No
FACE	Yes	No	1	No
CP	Yes	No	1	No

list the main differences between our methods and the baselines in terms of greedy decoder, beam search (BS), number of attention layers (#Attn), and other mandatory components (Extras). To allow for a fair comparison, we implement all our methods and baselines using ParlAI. We choose two publicly available benchmark datasets: OSDb and Twitter, for evaluating the baselines and our proposed methods. We follow Li et al. [3], Tao et al. [15] to use short-history conversations; see below for details.

4.1.1 OSDb. The OSDb dataset [16] is an online-available corpus of movie subtitles.[3] Here, we use the 2011 version and set aside ~60M lines, which constitutes ~30M message-response pairs for training. Following Li et al. [3], we randomly select 2K pairs from the IMSDB dataset [18] for validation and test sets, respectively, and we filter out pairs whose responses or messages are shorter than 6 tokens.

4.1.2 Twitter. Twitter is a commonly used source of data in dialogue generation research. For ease of reproducibility of the results reported in this paper, we did not follow Li et al. [3], Sordoni et al. [13] who used ~130M context-message-response triples before preprocessing. Instead, we use the version released by [12] for training, which contains ~4M tweet IDs in total and can be scraped in 3 days.[4] After formatting the tweets as *context-message-response* triples, we have 904K training examples and we concatenate context-message as input. For the training, validation and test sets, we exclude all overlapping IDs. Then we restrict the sequence length in both validation and test sets to the range [6, 18], resulting in 18,162 validation and 1,897 test triples. Furthermore, since tuning the MMI models is quite time-consuming, we randomly select 2K triples from the validation set for hyper-parameter selection.

4.2 Evaluation

4.2.1 Automatic evaluation. Although reported not to correlate well with human judgments [5], we still report BLEU scores [8] for fair comparisons with baselines. To evaluate response diversity, we use the *d-1* and *d-2* metrics proposed in [3] that are calculated as the number of distinct uni- and bigrams, divided by the total number of tokens generated.

4.2.2 Human evaluation. Following Sordoni et al. [13], we recruit crowd-source workers to perform pairwise qualitative comparisons. Since the conversation history of the OSDb dataset is only a single turn, which is hard for human annotators to judge, we choose the Twitter dataset for human evaluation, which has two-turn histories. We randomly select 1K test examples from the Twitter test set, and paired model responses (*FACE* vs. *baseline*) are shown to 3 evaluators in random order. Evaluators are told to choose a better response in terms of *relevance*, *interestingness* and *grammar*. Ties are allowed. We then carry out Welch's t-test on the human preferences obtained in this manner.

Table 4: Performance (%) on the OSDb dataset of different variants of FACE. Highest scores in bold face.

Model	d-1	d-2	BLEU
FACE-OPR	4.32	**20.47**	**8.03**
FACE-OPO	4.56	14.96	6.76
FACE-GPR	2.87	10.51	7.56
FACE-GPO	**5.03**	19.66	6.92

Table 5: Performance (%) on the Twitter dataset of different variants of FACE. Highest scores are in bold face.

Model	d-1	d-2	BLEU
FACE-OPR	**6.23**	**24.18**	8.33
FACE-OPO	5.73	17.95	**8.80**
FACE-GPR	4.13	15.03	7.99
FACE-GPO	5.69	17.78	8.72

4.3 Implementation details

For the OSDb and Twitter datasets, we keep the most frequent 25,000 tokens and replace other tokens with *_UNK_*. For the OSDb corpus, we use a 4-layer Long Short-Term Memory (LSTM) network for both encoder and decoder. The hidden size (HS) and word embedding size are set to 1,000. On the smaller Twitter corpus, we use a smaller network with a 2-layer LSTMs for encoder and decoder. The HS and word embedding sizes are set to 512 and 200, respectively. For both networks, we randomly initialize the model parameters from a uniform distribution $\mathcal{U}(-\sqrt{1/\text{HS}}, \sqrt{1/\text{HS}})$ and the word embeddings from a normal distribution $\mathcal{N}(0, 1)$. We also employ dropout [14] with drop ratio $p = 0.1$. We use Adam [2] as our optimization method. For the hyper-parameters of the Adam optimizer, we set the learning rate $\alpha = 0.001$, two momentum parameters $\beta 1 = 0.9$ and $\beta 2 = 0.999$, respectively, and $\epsilon = 10^{-8}$. We also clip gradients [9] to 5 during training to avoid gradient explosion. To speed up training and convergence, we use mini-batches of size 256. Since the model chosen with minimum training loss usually has very low diversity, we choose *d-1* as the early stopping criterion. A scheduler is used to reduce the learning rate by a factor of 0.5 when a *d-1* plateau is detected with $patience = 3$. β in Eq. (12) is set to 0.01.

5 RESULTS AND ANALYSES

5.1 FACE variants

To answer Q1, we first identify the best performing variant of FACE to be used in later experiments. In Tables 4 and 5, we list the scores for four variants of FACE, on the OSDb and Twitter datasets, respectively: Output token frequency & PRe-weight (*FACE-OPR*), Output token frequency & POst-weight (*FACE-OPO*), GT frequency & PRe-weight (*FACE-GPR*) and GT frequency & POst-weight (*FACE-GPO*).

[3] http://www.opensubtitles.org/

[4] On July 11 2018, when we finished scraping, only 2.6M IDs in total are still valid.

Table 6: Results (%) on the OSDb dataset. Highest scores in each column are highlighted using bold face.

Model	d-1	d-2	BLEU
Seq2Seq	2.70	8.63	7.34
Seq2Seq-refine	3.61	14.16	7.61
MMI-antiLM	2.73	10.68	7.83
MMI-bidi	3.06	12.19	7.01
MHAM	3.03	9.47	7.13
CMHAM	4.10	12.92	6.88
FACE	4.32	**20.47**	**8.03**
CP	4.18	15.59	7.27
CP_{free}	**5.48**	18.59	7.08
FACE-CP	4.63	18.32	7.89
$FACE\text{-}CP_{free}$	4.69	18.67	6.98

Table 7: Results (%) on the Twitter dataset. Highest scores in each column are highlighted using bold face.

Model	d-1	d-2	BLEU
Seq2Seq	5.87	17.73	8.77
Seq2Seq-refine	5.69	17.54	8.82
MMI-antiLM	**6.23**	18.21	6.51
MMI-bidi	5.42	15.16	8.20
MHAM	5.52	17.04	**8.96**
CMHAM	4.99	14.91	8.71
FACE	**6.23**	**24.18**	8.33
CP	5.97	18.15	8.84
CP_{free}	6.00	18.67	8.82
FACE-CP	6.07	23.50	8.25
$FACE\text{-}CP_{free}$	5.89	17.81	8.85

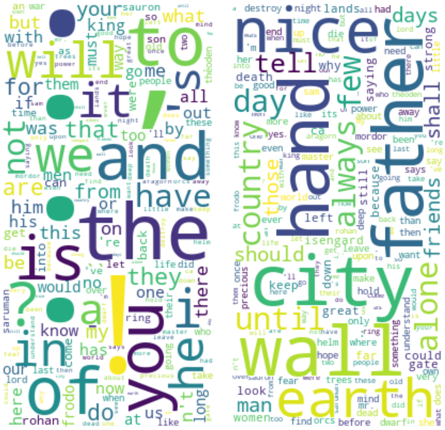

Figure 1: Word cloud showing top-200 frequent tokens of model responses on the OSDb dataset. Left: the larger the font, the higher the frequency. Right: the larger the font, the higher the weight (pre-weight used). (Best viewed in color.)

On the OSDb dataset, we can see from Table 4 that *FACE-OPR* and *FACE-GPO* perform better than the other two in terms of diversity scores. Similar results are shown on the Twitter dataset in Table 5. By comparing the results in Table 4 and 5, we find that the performance of *FACE-OPR* is very stable. We therefore choose *FACE-OPR* as our primary model and refer to it as *FACE* in the following sections. In contrast, *FACE-GPR* performs worst on both datasets, which indicates that GT frequency does not work well with pre-weight. The BLEU scores in Table 4 and 5 show no obvious patterns.

It is worth noting that all variants reported in Table 4 and 5 are using the train-and-refine strategy: first training the Seq2Seq model with CE, then fine-tuning it with FACE. We also tried training from scratch using FACE. However, the performance did not show consistent improvements, which is probably because our *equal expected loss* assumption (under Eq. (6)) is violated in the early training stages.

5.2 Automatic evaluation

We turn to Q2, Q3, and Q4. The automatic evaluation results are shown in Tables 6 and 7. *Seq2Seq* is vanilla-Seq2Seq with *general* attention mechanism [6]. *Seq2Seq-refine* is a fine-tuned version of *Seq2Seq* with a smaller batch size of 30. *MMI-antiLM* adjusts the *Seq2Seq* prediction probability using the decoder language model, while *MMI-bidi* utilizes a reverse model. Both methods use beam size 200. *MHAM* projects encoder hidden-states to 5 different semantic spaces, and *CMHAM* forces those 5 spaces to be perpendicular to each other. All of our models are fine-tuned *Seq2Seq* models with a smaller batch size of 30. *FACE* is fine-tuned using the FACE function and output token frequency. *CP* and CP_{free} are fine-tuned using Eq. (12) and (13), respectively. *FACE-CP* is a linear combination of FACE and CP. $FACE\text{-}CP_{free}$ is the multiplicative combination of FACE and the CP weighting function in Eq. (14).

As shown in Table 6, all the methods proposed in this paper outperform the baselines in terms of diversity metrics on the OSDb dataset. *FACE* achieves the highest d-2 score (increase of 6.3%). *FACE-CP* and $FACE\text{-}CP_{free}$ achieve slightly higher d-1 and lower d-2 scores than *FACE*, which can be viewed as the trade-off between *FACE* and *CP*'s. Although the penalty strength β of *CP* is carefully selected, it is interesting to see that our hyper-parameter free method CP_{free} outperforms *CP* by a large margin and achieves the highest d-1 score (1.9% improvement), which demonstrates the effectiveness of Eq. (13). *FACE* performs best in terms of BLEU scores, which suggests that *FACE* can generate higher-quality responses.

Although the MMI-based and MHAM-based methods all exhibit various improvements over *Seq2Seq* in terms of d-1 and d-2 scores, they mostly perform worse than *Seq2Seq-refine*. This suggests that on the OSDb dataset, carefully designed MMI-based and MHAM-based methods are not able to outperform a fine-tuned *Seq2Seq* baseline.

Similar to Table 6, Table 7 shows that all our methods can increase the diversity of *Seq2Seq* on the Twitter dataset. *FACE* achieves the highest d-1 score (0.4% increase) together with *MMI-antiLM*, but the highest d-2 score of *FACE* (6.5% improvement) demonstrates that our method fares better.

In Table 7 however, *Seq2Seq-refine* decreases the diversity of *Seq2Seq*, indicating that fine-tuning does not help to address the low-diversity problem on the Twitter dataset. This is probably because

Table 8: Results of the pairwise human evaluation (%) on the Twitter dataset. “Win”, “Lose” and “Gain” correspond to “FACE wins”, “Baseline wins” and their difference ($Win - Lose$), respectively. Highest scores are highlighted in bold face, and *,** ,*** symbols indicate significant improvements with *p-value* < 0.05, < 0.01, < 0.005, respectively.

Comparison	Win	Lose	Gain
FACE vs Seq2Seq	**38.61*****	21.54	17.07
FACE vs MMI-antiLM	**51.30*****	19.35	31.95
FACE vs MMI-bidi	**61.91*****	20.92	40.99
FACE vs MHAM	**50.93****	42.56	8.37
FACE vs CMHAM	**43.75***	38.85	4.90

the diversity of *Seq2Seq* is already relatively high on this dataset, and there is limited space for further improvements.

While *MMI-bidi* performs better than *MMI-antiLM* on the OSDb dataset, on the Twitter dataset, however, *MMI-bidi* performs much worse and degrades the diversity of the *Seq2Seq* model. The reason for this phenomenon is probably because of the reverse model of *MMI-bidi*: the input consisting of Twitter triples contains two turns from different speakers (*context-message*), thus given the *response* only, the prediction probability of *context-message* is very unreliable.

Although it achieves the highest BLEU score, *MHAM* slightly hurts diversity. Similarly, *CMHAM* degrades the diversity even more. Closer inspection reveals that most of the attention weights of both methods are on the first several tokens of the input sequence (i.e., *context*). It is unlikely to be able to properly learn the relation between *context-message* and *response* by only paying attention to the *context*.

To illustrate the effectiveness of the weighting function, we display the frequencies and the corresponding weights of some tokens in Figure 1. From this figure, we can see that the most frequent tokens have very small weights, and less frequent tokens receive larger weights. Please note that the tokens with the largest weights are in the scope of top-200 frequent tokens.

5.3 Human evaluation

We now turn to Q5. Human evaluation results are reported in Table 8. We can see that *FACE* is significantly better than all baselines. This means that by increasing response diversity, FACE can improve relevance and interestingness of responses, without sacrificing grammatical accuracy. Specifically, by penalizing frequent tokens, *FACE* gives more opportunities to less frequent tokens, so the interestingness is higher than *Seq2Seq*. While high-frequency tokens may be appropriate responses in more situations than low-frequency ones, it is for the same reason that higher frequency tokens convey less information, thus making the responses seem generic. Therefore, by encouraging low-frequency tokens, *FACE* has the potential to increase relevance of responses. By looking at the human evaluation scores in Table 8 together with the BLEU scores in Table 7, we can re-confirm that BLEU scores do not correlate well with human qualitative evaluation [5].

Since *FACE* learns a language model relying entirely on training data, the grammatical accuracy of *FACE* is as least as good as that of *Seq2Seq*. The examples in Table 9 show that *FACE* can actually generate more relevant and interesting responses. In contrast, although *MMI-antiLM* improves response diversity, it increases the risk of grammatical errors since it penalizes the language model. For example, the response of *MMI-antiLM* in the second example of Table 9 is ungrammatical and irrelevant to the conversation history. Similarly, *MMI-bidi* uses a reverse model to adjust the model outputs, in the hope that the responses can be more diverse and relevant. However, as explained earlier, the reverse model is highly unreliable, making many of the responses look weird, see, e.g., the third row of Table 9.

Table 9: Example responses from different models.

History	Responses
Context: i need 4 mo followers .. **Message**: u ever watched criminal minds suspect behavior??	***FACE***: no i do n't think so .. i 'm watching it now .. ***Seq2Seq***: i do n't know what to say ..
Context: dont you dare steal my idea. i call shenanigans **Message**: Totally not stealing. Everyone collaborated on that idea.	***FACE***: well then you should have said it like that . ***MMI-antiLM***: come on . come on go back to you*
Context: I never watched Spongebob's movie tho, was it good? You know Hey Arnold? **Message**: yes it is good u should watch :D haha hey arnould with footballegg face :P	***FACE***: haha , i 'm watching it now , it 's so good : d ***MMI-bidi***: oh , that
Context: perhaps a recipe is in order? I have room in my fridge... ;) **Message**: trying this one for 1st time LOL http://bit.ly/nrFMIU start from bottom recipe & work up	***FACE***: sounds good ! i 'll be sure to check it out ! thanks for sharing the mocha cakes with the butter cakes ;) ***MHAM***: i 'm sure you 'll find a recipe for me !
Context: Don't laugh at my picture! That's not nice. **Message**: i'm laughing because you actually posted the pic. I believed you:-) lol	***FACE***: i 'm not a fan of the picture but it 's not that bad , it 's just a pic of me saying " i 'm a girl " haha : -) ***CMHAM***: i 'm not laughing at you .

Although *MHAM* and *CMHAM* can improve response diversity to some extent, it is not guaranteed that the multiple attention heads are well distributed throughout the whole history. In fact, there are many cases where the attention weights are on the **Context** part of the history, making the responses irrelevant to the **Message**, such as the fourth and fifth examples in Table 9.

6 CONCLUSION

In this work, we have proposed a Frequency-Aware Cross-Entropy (FACE) function for tackling the low-diversity problem of Seq2Seq-based conversation models. Experiments on short-history conversation datasets demonstrate that the FACE loss function can effectively improve the diversity and quality of responses. FACE achieves the improvements with minimum modifications to the original Seq2Seq model, which makes it flexible to extend. We also propose a hyper-parameter free CP_{free}, which exhibits better performance than the original parameter-dependent CP.

A limitation of FACE is that the learning procedure is not as stable as CE, which increases the difficulty of training. In future work, we would like to investigate this phenomenon in depth. We also hope to test FACE in a long-history conversation setting by applying FACE to hierarchical Seq2Seq [11]. Besides, we also plan to apply FACE to stochastic models [12, 19] and examine how FACE can be used in an adversarial dialogue generation setup [4].

ACKNOWLEDGMENTS

This research was supported by the China Scholarship Council, Ahold Delhaize, the Association of Universities in the Netherlands (VSNU), and the Innovation Center for Artificial Intelligence (ICAI). All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

REFERENCES

[1] Shaojie Jiang and Maarten de Rijke. 2018. Why are Sequence-to-Sequence Models So Dull? Understanding the Low-Diversity Problem of Chatbots. In *Proceedings of the 2018 EMNLP Workshop SCAI: The 2nd International Workshop on Search-Oriented Conversational AI (SCAI '18)*.

[2] Diederik P Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations (ICLR '15)*.

[3] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. 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 (NAACL-HLT '16)*. 110–119.

[4] Ziming Li, Julia Kiseleva, and Maarten de Rijke. 2019. Dialogue generation: From imitation learning to inverse reinforcement learning. In *AAAI 2019: 33rd AAAI Conference on Artificial Intelligence*. AAAI.

[5] Chia-Wei Liu, Ryan Lowe, Iulian Serban, Michael 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 (EMNLP '16)*. 2122–2132.

[6] 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 (EMNLP '15)*. 1412–1421.

[7] Alexander Miller, Will Feng, Dhruv Batra, Antoine Bordes, Adam Fisch, Jiasen Lu, Devi Parikh, and Jason Weston. 2017. ParlAI: A Dialog Research Software Platform. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP '17)*. 79–84.

[8] 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 on Association for Computational Linguistics (ACL '02)*. 311–318.

[9] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the Difficulty of Training Recurrent Neural Networks. In *Proceedings of the 30th International Conference on Machine Learning (ICML '13)*. 1310–1318.

[10] Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. 2017. Regularizing Neural Networks by Penalizing Confident Output Distributions. In *International Conference on Learning Representations (ICLR '17)*.

[11] Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C Courville, and Joelle Pineau. 2016. Building End-To-End Dialogue Systems Using Generative Hierarchical Neural Network Models. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI '16)*. 3776–3784.

[12] Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron C Courville, and Yoshua Bengio. 2017. A Hierarchical Latent Variable Encoder-Decoder Model for Generating Dialogues. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence (AAAI '17)*. 3295–3301.

[13] 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. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT '15)*. 196–205.

[14] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *The Journal of Machine Learning Research* 15, 1 (2014), 1929–1958.

[15] Chongyang Tao, Shen Gao, Mingyue Shang, Wei Wu, Dongyan Zhao, and Rui Yan. 2018. Get The Point of My Utterance! Learning Towards Effective Responses with Multi-Head Attention Mechanism. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI '18)*. 4418–4424.

[16] Jörg Tiedemann. 2009. News from OPUS-A Collection of Multilingual Parallel Corpora with Tools and Interfaces. In *Recent Advances in Natural Language Processing (RANLP '09)*, Vol. 5. 237–248.

[17] Oriol Vinyals and Quoc V. Le. 2015. A Neural Conversational Model. In *ICML Deep Learning Workshop*.

[18] Marilyn A Walker, Ricky Grant, Jennifer Sawyer, Grace I Lin, Noah Wardrip-Fruin, and Michael Buell. 2011. Perceived or Not Perceived: Film Character Models for Expressive NLG. In *International Conference on Interactive Digital Storytelling (ICIDS '09)*. 109–121.

[19] Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning Discourse-level Diversity for Neural Dialog Models using Conditional Variational Autoencoders. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL '17)*. 654–664.