Stylistically User-specific Response Generation

Abdurrisyad Fikri **Hiroya Takamura** **Manabu Okumura**
Department of Information and Communications Engineering
Tokyo Institute of Technology, Japan
fikri@lr.pi.titech.ac.jp, {takamura, oku}@pi.titech.ac.jp

Abstract

Recent neural models for response generation show good results in terms of general responses. In real conversations, however, depending on the speaker/responder, similar utterances should require different responses. In this study, we attempt to consider individual user's information in adjusting the notable sequence-to-sequence (seq2seq) model for more diverse, user-specific responses. We assume that we need user-specific features to adjust the response and we argue that some selected representative words from the users are suitable for this task. Furthermore, we prove that even for unseen or unknown users, our model can provide more diverse and interesting responses, while maintaining correlation with input utterances. Experimental results with human evaluation show that our model can generate more interesting responses than the popular seq2seqmodel and achieve higher relevance with input utterances than our baseline.

1 Introduction

Human-computer conversation is a challenging task in Natural Language Processing (NLP). The aim of conversation models is to generate fluent and relevant responses given an input in a free format, *i.e.,* not just in the form of a question. A large amount of available data on the Internet has sparked the shift in conversation models. Starting with Ritter et al. (2011), completely data-driven models are now commonly used to generate responses. Furthermore, the sequence-to-sequence (seq2seq) model initiated by Sutskever et al. (2014) has been adapted to many NLP tasks, notably to machine translation (MT) and response generation.

input	*how are you ?*
user1	*good morning how are you*
user2	*i'm doing ok*
user3	*i'm good ! ! !*
user4	*not really good*
input	*i am excited !*
user1	*are you sure ? !*
user2	*come to the party ?*
user3	*yay ! ! !*
user4	*are you gonna do it ?*

Table 1: Sample responses from our proposed model involving four different users.

Actual conversations involving humans would be more engaging and the responses are not always general and monotonic. However, neural conversation models tend to generate safe, general, and uninteresting responses, *e.g.*, *I don't know* or *I'm OK* (Sordoni et al., 2015; Vinyals and Le, 2015; Li et al., 2016b). We argue that, aside from adding or understanding the context of a conversation, speaking style and response diversity also play an important role in delivering a more interesting conversation.

Recent studies addressed the response diversity and engagement issues and have attempted to generate responses better than the common and general ones. Some tackled this issue by defining and emphasizing context; previous utterances are commonly used as context in a conversation (Sordoni et al., 2015; Li et al., 2016a). Other studies have attempted to diversify or manipulate responses using specific attributes such as user identification (Li et al., 2016b), profile information sets (Zhang et al., 2018; Wang et al., 2017; Herzig et al., 2017), topics (Xing et al., 2017), and speci-

Proceedings of The 11th International Natural Language Generation Conference, pages 89–98,
Tilburg, The Netherlands, November 5-8, 2018.

fied mechanisms (Zhou et al., 2017).

In this study, we focus on the issue of "response style." We intend to let the model learn to generate responses that resemble those of a real person. Given an input utterance and user-specific information, the model will generate a response relevant to the input utterance based on the given user-specific information.

The existing methods that exhibit the use of user-specific information (Li et al., 2016b; Zhang et al., 2018), usually require that the users appear in the training data. Therefore, these existing methods cannot handle the unseen users, *i.e.*, users that do not exist in the training data. This is a limitation that we want to address in this study. As we intend to make our model versatile, we want to cover also the users that are not present in the training data. Hence, in this study, we propose a model that also works with unseen users.

Since we need identifiers of users, we rely on Twitter as the source of datasets. The dataset used in this work was constructed by collecting tweets and replies, *i.e.*, responses to other tweets. Aside from the user identity, to construct user-specific information, we retrieved individual public tweets from each account that are not replies to other tweets. We assume that some selected representative words from the retrieved individual tweets are suitable as the user's information. Therefore, we use two types of user-specific information: user identities and collections of users' representative words.

Unlike other tasks that can assume a finite set of expected outputs, *e.g.*, machine translation, in response generation, an input utterance can elicit various responses. Thus, measuring the quality of the output becomes a formidable issue. To measure the quality of generated responses, we rely on human judgment. Three evaluation criteria are provided to the judges: *fluency*, *relevance*, and *style*. The results show that our model is significantly better than the baseline in *relevance* and *style*. Some examples of generated responses from our model are shown in Table 1.

2 Related Work

Attempts to develop neural response generation models have been increasing rapidly, providing several options to further improve neural conversation models. Some notable studies in this field (Vinyals and Le, 2015; Shang et al., 2015; Sordoni et al., 2015) follow the encoder-decoder framework of Sutskever et al. (2014). For response generation, the encoder-decoder models are usually supplemented by the attention mechanism, following the implementation of Bahdanau et al. (2015) or Luong et al. (2015).

As for response diversity, earlier researches have acknowledged that responses to one input utterance could be varied (Shang et al., 2015; Li et al., 2016a). To address this issue, several approaches have been proposed; some of these attempts incorporate style or a persona into the model while others focus only on increasing the variety.

Li et al. (2016b) proposed a persona-based model that uses a feature called *speaker embeddings* that are based on an individual user's identity. They have integrated these embeddings into the decoding phase. Despite showing positive results, this approach works only for the persona or user identity that appears in the training data. If a persona is absent from the training data, it would behave like the normal seq2seq model. Our work is similar to them in that we use the speaker identity in the decoding phase, but our work can generate user-specific responses even for unseen or unknown users.

Similar efforts have been made by Zhang et al. (2018), who attempted to personalize the output style using a set of introductory sentences as the user's profile. They combined the encoder-decoder model with the memory network, aiming to enhance the model's ability to "memorize" the profile. A study from Wang et al. (2017) has also attempted to "steer" the output style using additional information called *scenting datasets*. These *scenting datasets* consist of a corpus, or a collection of particular sentences, with each dataset being exclusive to one character. In their study, Wang et al. (2017) only focused on one character (*scenting dataset*) for each model. Hence, their model can only generate responses of one particular style at a time. We also use an additional dataset to control the style, but we differ from them in that we can deal with multiple characters in one model.

A model focusing on increasing diversity without using specific characteristic was devised by Zhou et al. (2017). They defined some mechanisms and generated latent features to divert the context of input utterances before feeding them

to the decoder. They also presented some corresponding words to each mechanism.

3 Sequence-to-Sequence Setup

Following the popular approach in neural response generation, we base our encoder-decoder model on the seq2seq model with attention mechanism. Given the input sequence $X = (x_1, x_2, ..., x_{n_X})$, the model will attempt to produce the output sequence $Y = (y_1, y_2, ..., y_{n_Y})$ as a generated response. For the *encoder*, we adopt the LSTM (Hochreiter and Schmidhuber, 1997) unit to compute the representation of the input sequence. We keep all the hidden states produced by the encoder. Here, we use the notation $\bar{h}_s$ for each hidden state. Then, we adopt an attention-based model (Luong et al., 2015; Bahdanau et al., 2015) for the *decoder*. In general, the *decoding* process for each time step can be interpreted through the following equations:

$$
\begin{aligned}
p(y_t|y_{<t}, x) &= \text{softmax}(W_s\tilde{h}_t), && (1)\\
\tilde{h}_t &= \tanh(W_c[c_t; h_t]), && (2)\\
h_t &= LSTM(y_{t-1}, h_{t-1}), && (3)\\
c_t &= \sum_{s=1}^{S} a_t(s)\bar{h}_s, && (4)\\
a_t(s) &= \text{softmax}(h_t^\top W_a \bar{h}_s), && (5)\\
&= \frac{\exp(h_t^\top W_a \bar{h}_s)}{\sum_{s'} \exp(h_t^\top W_a \bar{h}_{s'})}.
\end{aligned}
$$

The attention-based model used in this work is based on Luong et al. (2015). The weights W_s and W_c are the learned parameters of the decoder. With $a_t(s)$ as the vector containing the alignment score for each hidden state $\bar{h}_s$ of the encoder, c_t is the context for the current decoder at time step h_t. In addition to the attention-based model, we also apply the *input-feeding* approach by Luong et al. (2015) as an attempt to make the model capture the previous alignment. Input-feeding is done by concatenating the current attentional vector $\tilde{h}_t$ to the input to the decoder at the next time step. For both the encoder and the decoder, we employ two-layer LSTM architectures.

4 Response Generation with Attention to Speaker Information

As mentioned in Section 1, we argue for the importance of diversity in response style in creating a more compelling conversation. Our intention is to capture the characteristics of the users, *i.e.,* the responders, and to take them into account in response generation. Our work can be considered as an attempt to improve the persona-based model by Li et al. (2016b). Their model represents individual users, or in their term *speakers*, in the training data as a vector or embedding of speaker-specific information. Adapting their work, we pick usernames as one of the user-specific attributes, and then convert them to embeddings to allow the model to distinguish between users' characteristics. However, this approach can only accommodate users present in the training data. To overcome this issue, we suggest a small dataset for each user to serve as another characteristic feature.

4.1 User-Specific Information

In this study, we define two kinds of user-specific information: user embeddings and user-info embeddings. User embeddings are derived from usernames in the training data, while user-info embeddings are derived from separate collections of words used by the users. User embeddings are only useful for users present in the training data, while user-info embeddings are independent of the training data. The details about how these data are retrieved will be explained in Section 5.

Following the setup described in Section 3, let I_{train} denote the set of users (responders) in the training data, K_{word} the dimension of word embeddings, and K_{user} the dimension of user embeddings. We convert words in each input sequence X to embeddings with size K_{word}. Then, we define a user identity, embedding u_i with size K_{user} for each user $i \in I_{train}$. The user embedding u_i is shared to all conversations involving user i.

The second type of user information involves a collection of users' selected words. In order to capture the characteristic, especially the speaking style, of each user, we argue that we need to define a feature or a set of information that can let the model learn about the characteristic. Thus, we assume that a carefully selected set of words from each user's conversation history is suitable for this task.

Let I denote the set of users. Note that I_{train} is a subset of I. For each user in I, several sentences can be collected. From this collection of sentences, we then extract N words to represent the characteristics of the user. To select those N words, we need a particular approach to score the

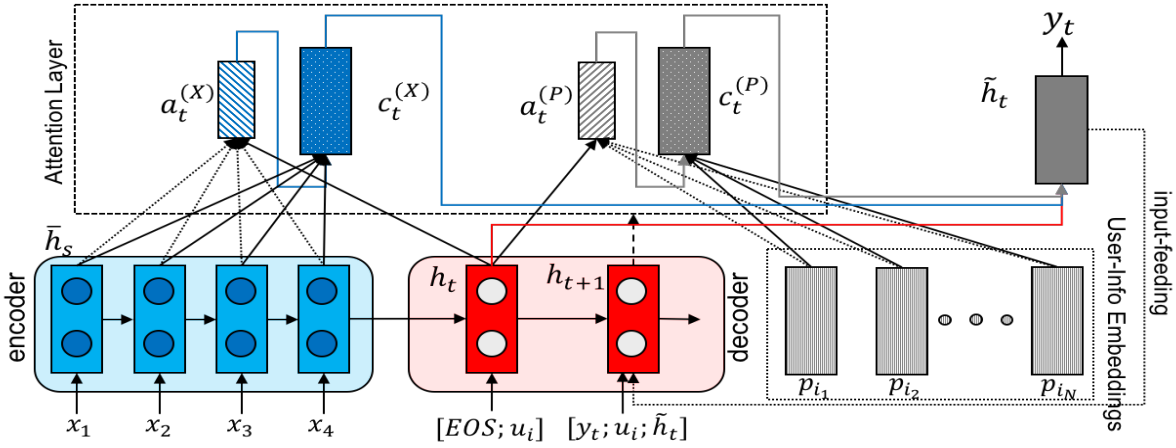

Figure 1: Overview of our neural conversation model with attention to user-specific information. We use two-layer LSTM for both the encoder and the decoder. The attention layer attends to source hidden states $\bar{h}_{s'}$
and user-info embeddings P_i for user i. User embeddings u_i are concatenated with the decoder input at every step.

words.

We compared several scoring methods that are simple enough to employ: word frequency, TF-IDF (Sparck Jones, 1988), and Pointwise Mutual Information (PMI). To compare them, we treated all the words in the selected sentences as the input for every method. Then we ranked the words according to the scores by every method and took N words with the highest ranks for each method. Hence, we have three sets of selected N words, and then deployed them in the training and evaluated the results preliminarily.

Two fluent speakers of English were asked to compare the quality of generated responses. We provided the two evaluators with three sets of generated responses using three different sets of N words, then asked them to evaluate the fluency and relevance to the input message. Based on their evaluation, we decided to choose TF-IDF as the scoring method to extract N words as the user-info dataset. Each of these words is further converted to an embedding of dimension K_{word}.

4.2 Attentional Conversation Model

Our attentional LSTM model takes three features as input: input word embeddings, user embeddings, and user-info embeddings. Both user embeddings and user-info embeddings are used in the decoder of the encoder-decoder model. Since our model also incorporates the *input-feeding* approach, the input for the decoding phase is the concatenation of the output of the previous time step y_{t-1}, user embedding u_i, and input-feeding $\tilde{h}_{t-1}$. The user-info embeddings will be used later as the input for additional attention mechanism. Hence, the decoding process can be described as follows:

$$h_t = LSTM([y_{t-1}; u_i; \tilde{h}_{t-1}], h_{t-1}). \quad (6)$$

The user-info embeddings are constructed from the collection of top N ranked words uttered by intended users, where the users are not necessarily present in the training data. Using the same embeddings as input word embeddings, we compose $P_i = \{p_{i_1}, ..., p_{i_N}\}$ $(\forall k,\ p_{i_k} \in \mathbb{R}^{K_{word}})$ as user-info embeddings for user i.

The model is trained to attend not only to the input source, *i.e.,* the hidden states of the encoder, but also to the user-info embeddings. Therefore, since this model uses two contexts, we need to adjust Equation (3) to

$$\tilde{h}_t = \tanh(W_c[c_t^{(X)}; c_t^{(P)}; h_t]), \quad (7)$$

where we define $c_t^{(X)}$ as the context for input source and $c_t^{(P)}$ as the context for user-info embeddings. This proposed model is illustrated in Figure 1.

5 Datasets

Since our target is to incorporate and emphasize the response styles of actual human responders, we need to include user identification attributes in the datasets. Therefore, for datasets, we collected

tweets from Twitter API. Then, we constructed two types of datasets: *conversation dataset* and *user-info dataset*.

5.1 Conversation Dataset

This dataset is designated to train the model to generate a response to a given input utterance in general. We extracted this dataset from Twitter, and retrieved only those tweets that satisfy the following conditions. We set a filter to select only reply tweets, *i.e.*, responses to other tweets, from users who had engaged in conversations with a minimum of three turns. We paired each reply with the tweet that it is a response of, as *response* and *input utterance*, respectively. We then used the responders' usernames as the user identification attribute, hence user embeddings. Note that the user embeddings can only be obtained from this conversation dataset.

To improve data quality, we further cleaned up the retrieved tweets to remove some noises, such as tweets with non-ASCII characters, duplications, and non-English tweets. We also removed URLs, hashtags, and mentions from tweets. The final conversation dataset consists of around 230,000 pairs of input utterances and responses.

5.2 User-Info Dataset

This dataset is an effort to capture more characteristic of the users and also to handle the unseen users in the training dataset. User-info embeddings mentioned in Section 4.1 are derived from this dataset. To construct user-info dataset, we retrieved tweets from the accounts of every username in the conversation dataset. To ensure that this dataset is independent from the conversation dataset, we retrieved only individual tweets, *i.e.,* non-replies as opposed to the reply tweets for conversation dataset. We retrieved all public tweets, via Twitter API, from each account and then applied TF-IDF to find the most important words for each user. For an individual user, we treated one tweet (sentence) as one document and hence computed the TF-IDF score for each word across all sentences. Then, we kept the top 50 words according to the TF-IDF scores.

The usage of this dataset is independent of the conversation dataset. We can pair the user information in the user-info dataset with the one, the same user, in the conversation dataset or we can disregard the relationship.

6 Experiments

6.1 Implementation Details

Both our encoder and decoder employed two-layer stacked LSTMs. Some hyper-parameter details are as follows:

- Each LSTM layer contains 300 hidden units.
- Embedding size is set to 300.
- Network parameters are initialized with uniform distribution $[-0.05, 0.05]$.
- Training batch size is set to 128.
- Learning rate for the encoder is set to 0.0001, multiplied by 2.5 for the decoder.
- Dropout rate is set to 0.1.
- Vocabulary size is 35,000.

We trained the model by using the Adam optimizer (Kingma and Ba, 2014) with different learning rates between the encoder and decoder. We conducted several procedures to determine the training stop condition. We observed the decrease in loss $H_{y'}(y) := -\sum_i y'_i \log(y_i)$. When the decrease was starting to converge, at around less than 7%, we asked two English fluent speakers to evaluate the generated responses. Finally, we stopped the training at the 47th epoch. We also limited the maximum length of an utterance to 15 words per sentence. The training was run on a single Titan X GPU for about three days.

The input utterance and user-info embeddings were initialized with GloVe embeddings (Pennington et al., 2014). We replaced the words not in the vocabulary with *UNK* tokens. The same treatment was applied to unseen users in user embeddings. We set the *UNK* token embeddings to a vector of all zeroes at the initial stage. To select the prediction, we opted to use the greedy approach.

6.2 Baseline and Comparison Models

We adopted the speaker model of Li et al. (2016b) to serve as the benchmark for our model. Their work used persona (user-identification attribute) in the decoding phase to let the model assimilate the style of that user, or "nearby" users, into the responses.

In terms of using user embeddings in the decoding phase, our model and theirs are similar. However, as mentioned in Section 5, user embeddings

cannot cover unseen users. Our model overcomes that issue by using user-info embeddings. The decoder input of both the models can be represented by Equation (6). Since the baseline model does not have user-info embeddings, our model's attentional hidden $\tilde{h}_t$ is different from theirs. The attentional hidden of the baseline model would be the same as Equation (3), while our model's $\tilde{h}_t$ is represented by Equation (7).

We also prepare a variant of our proposed model, using unseen (*UNK*) users for user embeddings. The rationale for this setting is to investigate whether our model could generate better responses against our baseline's handicap. The last comparison model was a vanilla seq2seq model (without user and user-info embeddings). For simplicity, we labeled the four models as **User + Info** for our main model with *user embeddings* and *user-info embeddings*, **UserOnly** for baseline, **UNK + Info** for our variant model with *unseen users* and *user-info embeddings*, and **seq2seq** for vanilla seq2seq model.

Bio
Journalist. Writer. Broadcaster. For Hire. #AllBlackLivesMatter. Everything is wrestling. Header by @censored

Sample Tweets
- Breaking in a new pair of jeans today. Pray for yer boi.
- Nah it's asocial cold now. Don't invite me to any events. I'm not trekking. I'm not traveling unless there free food, booze or you paying me. It's blitz.
- Shouts to all my freelancers who are getting more work now the full time peeps are taking their winter/Christmas holiday time. Rumble workers, rumble.
- Nah someone needs to put you in the sin bin for 10 minutes. You are out of control today.
- Had creamed corned for the first time yesterday. Looked like sick, tasted alright

Figure 2: Example of a user's Twitter bio and sample tweets used in style evaluation. We censored any mentions of other accounts.

6.3 Evaluation Setup

Many previous studies on dialogue or response generation models (Li et al., 2016b,a; Sordoni et al., 2015; Xing et al., 2017) relied on BLEU (Papineni et al., 2002) as their automatic evaluation metric. To compute the score, BLEU measures the overlapping words or n-grams between the generated output (hypothesis) and the target output (reference). BLEU was initially intended for machine translation, which tends to have a finite target; therefore, it might not be suitable for evaluating conversation models.

According to Liu et al. (2016), BLEU is lowly correlated with human judgments of dialogue systems. Additionally, some other work on response generation (Shang et al., 2015; Li et al., 2016c; Wang et al., 2017; Zhou et al., 2017) did not use BLEU for their evaluation method, relying on human judgment instead. Thus, we opted to use only human evaluation in our work.

We hired judges from Amazon Mechanical Turk (AMT) to evaluate the quality of our generated responses. The following three judgment criteria were defined:

- ***fluency*** or ***naturalness***: Whether the response could be produced by (an English speaking) human.
- ***relevance*** or ***adequacy***: Whether the response could be accepted as a suitable answer or contained useful information regarding the input utterance.
- ***style***: Whether the response could be produced by the same person if some profile information was provided.

The rationale behind measuring these criteria is as follows. Even though our goal is to integrate styles to the generated responses, we also want to assure that the generated responses are correct and useful to the input. Since we supposed that ***style*** is significantly harder to evaluate, the evaluation task was done in two stages: the first stage was for ***fluency*** and ***relevance***, and the second stage was for ***style***.

We randomly picked 12 users from the conversation dataset and retrieved tweets that they replied to. For each user, 5–10 tweets were obtained to be used as input utterances. In total, 100 tweets were collected, and each pair of an input utterance and its response was then evaluated by 10 judges.

Models	Fluency (%)			Relevance (%)		
	bad	enough	good	bad	enough	good
UserOnly (Baseline)	19.5	27.3	53.2	51.8	25.2	23.0
seq2seq	8.2	25.8	66.0	40.1	29.4	30.5
User + Info	17.5	26.4	56.1	44.9	28.2	26.9
UNK + Info (with unseen users)	9.0	23.7	**67.3**	37.4	31.2	**31.4**

Table 2: Human evaluation results for *fluency* and *relevance*, presented as raw score percentages. Our UNK + Info model with unseen users gains 26.5% more for fluency and 36.5% more for relevance compared to the baseline.

For the first stage, we provided the judges with only input utterance-response pairs. There were four models in total, so one utterance had four response alternatives. We employed a three-point Likert scale, labeled $\{bad, enough, good\}$, which were later converted to $\{-1, 0, +1\}$, respectively, and asked the judges to score every response alternative in terms of ***fluency*** and ***relevance***.

In the second stage, the judges were provided with Twitter user bio, *i.e.*, a user's short biography or profile information that commonly contains keywords, and some sample tweets from the respective users. We asked the judges to evaluate the response alternatives on the basis of the provided information and to score them in the range from 1 to 5, where a smaller number is better. Since this time the judges have provided information to compare to, we assume that ranking is more appropriate to measure the similarity between response alternatives and provided samples. Ties in the score were permitted. For ***style*** evaluation, since we intended to investigate the influence of user-specific information to the response, we excluded the vanilla seq2seq model. An example of the provided information is shown in Figure 2.

7 Results and Analysis

7.1 Human Judgment

We first evaluated the *fluency* and *relevance* of the responses. In this stage, one utterance received four responses from all models. We let the judges score using three choices: bad, enough, and good.

To decide which model is the better one, first, we counted the number of each score label every model received. We call it raw scores. The summary of raw scores by the judges is shown in Table 2. According to these results, for both criteria, UNK + Info (with unseen users) received the highest *good* score, followed by the seq2seq model. The UNK + Info gains 26.5% more *fluency* point compared to the baseline. To calculate this gain, we simply compared the percentage obtained by UNK + Info (67.3%) against UserOnly (53.2%).

While it is obvious that high *good* score is desirable, we also want to see if our model can generate more "good enough" response compared to other models. Then, we combined *enough* and *good* scores as *acceptable*. As shown in Table 4, seq2seq seems to achieve the highest score in terms of *fluency*, even though it only gains 0.87% more point than UNK + Info model. Therefore, we argue that our UNK + Info can be considered as fluent enough.

For *relevance*, the *bad* scores of all models are higher than the *good* scores. This implies that *relevance* is harder to achieve than *fluency*. Yet our variant model, UNK + Info, achieved the highest acceptable score in this criteria, gaining 36.5% compared to the baseline.

We calculated the average of the scores by converting them to $\{-1, 0, +1\}$ for each model and criteria, as seen in Table 5. In this table, our variant model, UNK + Info, wins in both criteria. Nevertheless, our model can get higher scores in both criteria than the baseline model, UserOnly. Using one-way ANOVA as significance test, we confirmed that our model is significantly better than the baseline in the *relevance* criteria.

In the second stage, we measured the similarity in style among generated responses and users' sample tweets. As in the first stage, each of the 100 input-response pairs was evaluated by 10 judges, resulting in 1,000 samples, from which we removed some results that did not show consistency, *e.g.*, the results with identical responses with a different score. As explained in Section 6.3, we evaluated only three models. This was done to

User	Input	Responses	
		User + Info	UNK + Info
User1	*i'm going to have a cardiac episode and my wife is not here*	*i love you*	*what if i do*
	hate to burst your bubble it snowed last night & this morning	*and now you get to the game*	*ahh ! ! ! i was so excited !*
	they cut so much out of the finale !	*i was thinking it was*	*it was a good season ! !*
User2	*if lebron loses this series does he leave and where does he go ?*	*he scored the house and got his back house*	*he was playing with his rb / 3*
	he was five for his last 50 you literally had to pound the middle of the zone three times	*are you saying that were no reason ?*	*he has a good draft ?*
	hey guess what ? the bills will draft a few football players this weekend	*lol you got that man ! ! ! ! ! !*	*you beat me ?*
User3	*this is such a mood and i can't even explain why*	*did you get braces or walmart ?*	*hello ! ! !*
	you can skip iron man 3	*do i get some rest ?*	*yeah i am*
	heard he cheated on her	*don't forget*	*she was a lesbian*

Table 3: Examples of responses from different users generated by our model, using known users and their user-info respectively, and its variant model, using unseen users and the same user-infos.

investigate the influence of user-specific information. Additionally, we intended to perform a comparison with a baseline with the same objective. The results can be observed in Table 6.

Based on the style evaluation results in Table 6, the average scores appear to be positioned in the middle of the range, *i.e.*, around rank three. It suggests that, in general all models only generate "good enough" responses in term of style. Nevertheless, our variant model is significantly better than the baseline, proved by the Friedman Test. Sample responses generated by our model are shown in Table 3.

Models	**Fluency** (%)	**Relevance** (%)
	acceptable	acceptable
UserOnly	80.5	48.2
seq2seq	**91.8**	59.9
User + Info	82.5	55.1
UNK + Info	91.0	**62.6**

Table 4: Acceptable or "Good enough" results with *good* and *enough* scores combined. seq2seq tops *fluency*, but our model with unseen users gets the highest *relevance* score.

Models	Fluency	Relevance
UserOnly	0.337 ± 0.06	-0.28 ± 0.06
seq2seq	0.578 ± 0.05	-0.09 ± 0.06
User + Info	0.386 ± 0.06	-0.18 ± 0.06
UNK + Info	**0.583 ± 0.05**	**-0.06 ± 0.06**

Table 5: Average scores for *fluency* and *relevance* criteria. For ***relevance***, our model achieved significantly better scores than the baseline (one-way ANOVA, $p < 0.05$).

Models	Style Rank
UserOnly	3.37 ± 0.09
User + Info	3.29 ± 0.09
UNK + Info	**3.16 ± 0.09**

Table 6: Results of style evaluation. Smaller values are better. Our variant model was significantly better than the baseline (Friedman Test, $p < 0.05$).

7.2 Analysis: External Resources and Response Style

Our main intention is to incorporate an individual user's characteristics to generated responses. We specifically attempted to incorporate more information to emphasize the response style of different users. Therefore, we conducted an experiment to incorporate additional information, and the evalu-

ation we performed proved that the judges recognized a better change in style.

Furthermore, one aspect that distinguishes our model from others is the application of external resources. Usually, if a model was trained to pick up some specific traits or characteristics, such features should be included in the training. Our work also serves as an evidence of usability of external resources for response generation models. With simple mechanisms such as attention, our model can adjust the responses to be better with a small "plug and play" dataset.

An interesting finding is that the variant UNK + Info model achieved better scores than our User + Info model. Through manual observation, we conceived that a model with more injected information can become too "stylized" and lose some relevance to the input utterance. However, the baseline, with less information, still received lower scores. This indicates the strength of the attention mechanism.

In conclusion, a problem still persists in styling generated responses. Regardless of the results being better than the baseline for the previous work, generating fluent and relevant responses with an expected style is still challenging. It might be the common case that either the responses are good but general and timid, or they are interesting but lacking some relevance.

8 Conclusion and Future Work

In this study, we conducted experiments to address the response diversity issue, particularly in response style. We employed user-specific information to drive the generated responses to resemble real user's utterances. We considered usernames and the user-info dataset as user-specific information.

Evaluation through human judgment showed that the outputs of our model are better than the baseline overall, especially our variant model with unseen users. Our model also showed the potential of using external resources in encoder-decoder models. Although we cannot declare that our model architecture is sophisticated, our experiments can serve as the evidence that a simple but appropriate architecture can improve response quality.

The remaining challenge is how to properly emphasize the response style without damaging the content (context) or its relevance. If we can make a good compromise between response content and style and can control the use of these two elements, we argue that it would substantially increase the quality of conversation models.

Acknowledgements

This work is partially supported by JST PRESTO (Grant Number JPMJPR1655).

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Jonathan Herzig, Michal Shmueli-Scheuer, Tommy Sandbank, and David Konopnicki. 2017. Neural response generation for customer service based on personality traits. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 252–256. Association for Computational Linguistics.

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

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. 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. Association for Computational Linguistics.

Jiwei Li, Michel Galley, Chris Brockett, Georgios 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 (Volume 1: Long Papers)*, pages 994–1003, Berlin, Germany. Association for Computational Linguistics.

Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. 2016c. Deep reinforcement learning for dialogue generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202. Association for Computational Linguistics.

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. Association for Computational Linguistics.

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. Association for Computational Linguistics.

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, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

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, Doha, Qatar. Association for Computational Linguistics.

Alan Ritter, Colin Cherry, and William B. Dolan. 2011. Data-driven response generation in social media. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 583–593. Association for Computational Linguistics.

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)*, pages 1577–1586. Association for Computational Linguistics.

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*, pages 196–205, Denver, Colorado. Association for Computational Linguistics.

Karen Sparck Jones. 1988. Document retrieval systems. chapter A Statistical Interpretation of Term Specificity and Its Application in Retrieval, pages 132–142. Taylor Graham Publishing, London, UK, UK.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

Oriol Vinyals and Quoc V. Le. 2015. A neural conversation model. In *Proceedings of the 31th ICML Deep Learning Workshop*, Lille, France.

Di Wang, Nebojsa Jojic, Chris Brockett, and Eric Nyberg. 2017. Steering output style and topic in neural response generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2140–2150. Association for Computational Linguistics.

Chen Xing, Wei Wu, Yu Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. 2017. Topic aware neural response generation. In *AAAI*.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? *CoRR*, abs/1801.07243.

Ganbin Zhou, Ping Luo, Rongyu Cao, Fen Lin, Bo Chen, and Qing He. 2017. Mechanism-aware neural machine for dialogue response generation. In *AAAI*.