Relevance-Promoting Language Model for Short-Text Conversation*

Xin Li,[1] **Piji Li,**[2] **Wei Bi,**[2] **Xiaojiang Liu,**[2] **Wai Lam**[1]
[1]Department of Systems Engineering and Engineering Management,
The Chinese University of Hong Kong
[2]Tencent AI Lab, Shenzhen, China
{lixin, wlam}@se.cuhk.edu.hk, {pijili, victoriabi, kieranliu}@tencent.com

Abstract

Despite the effectiveness of sequence-to-sequence framework on the task of Short-Text Conversation (STC), the issue of under-exploitation of training data (i.e., the supervision signals from query text is *ignored*) still remains unresolved. Also, the adopted *maximization*-based decoding strategies, inclined to generating the generic responses or responses with repetition, are unsuited to the STC task. In this paper, we propose to formulate the STC task as a language modeling problem and tailor-make a training strategy to adapt a language model for response generation. To enhance generation performance, we design a relevance-promoting transformer language model, which performs additional supervised source attention after the self-attention to increase the importance of informative query tokens in calculating the token-level representation. The model further refines the query representation with relevance clues inferred from its multiple references during training. In testing, we adopt a *randomization-over-maximization* strategy to reduce the generation of generic responses. Experimental results on a large Chinese STC dataset demonstrate the superiority of the proposed model on relevance metrics and diversity metrics.[1]

Introduction

Short Text Conversation (STC) (Shang, Lu, and Li 2015), also known as single-turn chit-chat conversation, is a popular research topic in the field of natural language processing. It is usually formulated as a sequence translation problem (Ritter, Cherry, and Dolan 2011; Shang, Lu, and Li 2015) and the sequence-to-sequence encoder-decoder (SEQ2SEQ) framework (Cho et al. 2014; Sutskever, Vinyals, and Le 2014; Bahdanau, Cho, and Bengio 2015) is applied for solving this problem. The decoder generates the responses token-by-token, conditioned on the compressed query representations from the encoder. Following this paradigm, many attempts have been conducted to refine the quality of the generated responses (Li et al. 2016a; Xing et al. 2017; Du et al. 2018; Wu et al. 2019).

Despite the effectiveness of these efforts, some intrinsic issues of SEQ2SEQ-based models still hinder further improvement of generation performance. Under the SEQ2SEQ formulation, the auto-regressive decoder is only trained on the gold-standard response text while the query text is ignored, leading to under-exploitation of the training data. Besides, the maximization-based decoding strategies adopted in existing models, such as beam search and greedy search, restrict the search space to the most frequent phrases and thus they have the tendency to generate the generic responses or repetitive responses with unnaturally high likelihood, degrading the conversational experience.

GPT-2 (Radford et al. 2019), a recently proposed Transformer-based language model, provides an alternative solution for language generation. One advantage of GPT-2 is that the transformer language model can not only capture the context of arbitrary length but also make full use of the textual supervision signals because the generator is actually the language model itself. Moreover, GPT-2 adopts top-k sampling (Fan, Lewis, and Dauphin 2018) to diversify the generated texts while preserving the relevance. Obviously, these characteristics are attractive and meaningful for solving the STC task, whose aim is to generate informative and diverse human-like responses given the user queries.

However, due to the essence of language modeling, directly applying GPT-2 on the STC task, a conditional language generation task, may be insufficient because the language model is unable to discriminate the source (query) sentence and the target (response) sentence. The original experimental results of GPT-2 on the abstractive summarization task (Nallapati et al. 2016) also verify this claim. Another potential issue of adapting language model for the STC task comes from ***recency bias*** (Khandelwal et al. 2018) and ***explanation-away*** effects (Yu et al. 2017; Holtzman et al. 2019), where the language model has the tendency to rely overly on the immediate context and explain away from the long-term context[2], yielding fluent but topically irrelevant responses.

*The work described in this paper is substantially supported by a grant from the Research Grant Council of the Hong Kong Special Administrative Region, China (Project Code: 14204418). It was mainly done when Xin Li was an intern at Tencent AI Lab.

[1] Code available at https://ai.tencent.com/ailab/nlp/dialogue/.

[2] Long-term context in language model is roughly equivalent to the source information in SEQ2SEQ framework.

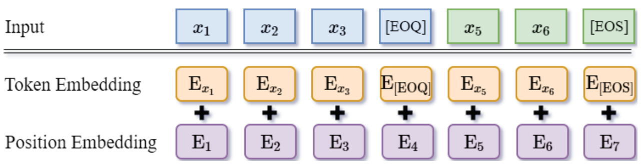

Figure 1: Representations of the example input with $n = 7$ and $m = 4$.

With the motivation of inheriting the merits of transformer language model while alleviating the potential issues under the language model formulation, we carefully design a training strategy to adapt the auto-regressive transformer-based language model[3] for the conditional response generation. First of all, it is observed that the dialog conversation is actually a process of text continuation, in other words, giving the response right after the query. Based on this observation, we can regard the STC task as a language modeling problem on the concatenated sequence of query and response. To discriminate the generation of query tokens and that of response tokens, we inject a special token between query and response, acting as the trigger of response generation. With this formulation, the language model based training objective can make use of the textual data from query, alleviating the under-exploitation issue mentioned above.

Since the transformer-based language model tends to focus on the short-term context and ignore the long-term context, namely, the ***explanation away*** issue, we propose to empower the self-attention with encoder-decoder attention, which enforces the model to pay additional attention to the query, especially the query tokens of user interest, and guides the model to rely on informative query tokens to make good predictions. It is also observed that some response tokens not mentioned in the query are still closely related to the discussed topic in the conversation. In order to exploit such kind of relevance clues hidden behind the responses, we propose a topic inference component to learn a compact source (query) representation encoding the information relevant to the query and feed the query representation into each generation step, encouraging the language model to consider the generation of the topic words potentially related to the query.

As with the decoding strategy, different from the existing STC models, we propose to decode with ***randomization-over-maximization*** method, namely, the top-k sampling, from the transformer language model to generate the relevant response with high originality.

In summary, our contributions are as follows:

- We tailor-make a training strategy to adapt the transformer-based language model for the Short Text Conversation (STC) task.
- We propose two components, namely, Supervised Source Attention (SSA) component and Topic Inference (TI) component to promote the relevance modeling in the language model based response generator.
- To the best of our knowledge, we are the first to introduce top-k sampling, a ***randomization-over-maximization*** strategy, for diverse response generation.[4]

Model

Overview

In our language model formulation, each training query-response pair and the special tokens are concatenated as a single sequence $\mathbf{x} = \{x_1, \cdots, x_m, x_{m+1}, \cdots, x_n\}$ of length n. $\mathbf{x}_{1:m}$ corresponds to the query token sequence of length m and x_m is the special token [EOQ], denoting the end of query. $\mathbf{x}_{m+1:n}$ corresponds to the response and x_n is [EOS], the end symbol of the whole sequence. The training objective of our model is to maximize the unconditional likelihood $p(\mathbf{x}_{1:n})$, similar to the existing language models (Bengio et al. 2003; Merity, Keskar, and Socher 2018).

The architecture of our model is depicted in Fig 2, where L decoder-only transformer layers (Vaswani et al. 2017)[5] are involved. Different from the original transformer layer solely containing the self-attention component, the transformer layer in our model is further empowered with the proposed supervised source attention (SSA) component. The outputs of the l-th transformer layer are the contextualized token representations of size $\dim_h$, denoted as $\mathbf{H}^l \in \mathbb{R}^{n \times \dim_h}$. When predicting the tokens, a Topic Inference (TI) component is introduced to provide the refined query representations encoding the topic information inferred from the reference.

Language Model as Response Generator

To achieve the goal of adapting language model for the STC task, we should carefully design a training strategy different from that in the SEQ2SEQ framework. Based on the observation that the human conversations can be regarded as a process of text continuation (i.e., giving the response/answer right after the query/question), we concatenate the query token sequence and the response token sequence into a single sequence and formulate the STC task as a contextual text continuation problem. One input example of our model is illustrated in Fig 1. The training goal of the model is to minimize the joint negative log likelihood over the whole sequence:

$$\mathcal{L}^{\text{mle}} = -\log P(\mathbf{x}_{1:n}) = -\sum_{t=1}^{n} \log P(\mathbf{x}_t | \mathbf{x}_{<t}) \quad (1)$$

Obviously, it is easy to bridge the gap between the task-specific training and the auto-regressive pre-training (Peters et al. 2018; Radford et al. 2018; 2019) because the formulations of their objectives are almost the same. Another advantage of this language model formulation is that it takes

[3] Without explicit specification, the language model in our paper refers to the "auto-regressive" language model, which is different from those "auto-encoding" language models (Devlin et al. 2019; Dong et al. 2019).

[4] We notice that some concurrent works (Olabiyi and Mueller 2019; Zhang et al. 2019) also adopt the strategy similar to ours after the submission.

[5] For the technical details of transformer, we recommend the reader to read the paper (Vaswani et al. 2017).

the likelihood of query tokens into consideration, which is ignored in the existing works (Shang, Lu, and Li 2015; Xing et al. 2017). Intuitively, the text generated by the language model is more fluent than those generated by SEQ2SEQ framework because the generator of the language model (the language model itself) is not only trained on the response sentence but also the query sentence.

Relevance Modeling Component

The vanilla transformer decoder is equipped with self-attention (Cheng, Dong, and Lapata 2016; Lin et al. 2017) and can theoretically capture the context of arbitrary length. Given the input $\mathbf{H}^{l-1} \in \mathbb{R}^{n \times \dim_h}$, the contextualized representations $\mathbf{h}_t^l$ ($l \in [1, L]$, $t \in [1, n]$) at the t-th time step is built as follows:

$$\begin{aligned} \mathbf{h}_t^l, \boldsymbol{\alpha}_t^l &= \text{SLF-ATT}(\mathbf{q}_t^{l-1}, \mathbf{K}_{\leq t}^{l-1}, \mathbf{V}_{\leq t}^{l-1}) \\ \mathbf{Q}^{l-1} &= \mathbf{H}^{l-1}\mathbf{W}^Q \\ \mathbf{K}^{l-1}, \mathbf{V}^{l-1} &= \mathbf{H}^{l-1}\mathbf{W}^K, \mathbf{H}^{l-1}\mathbf{W}^V \end{aligned} \tag{2}$$

where SLF-ATT is the self-attention layer[6] and $\boldsymbol{\alpha}_t^l \in \mathbb{R}^t$ is the calculated attention vector. $\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{n \times \dim_h}$ respectively denote the query[7], key and value in the self-attention layer. $\mathbf{K}_{\leq t}^{l-1} = \{\mathbf{k}_1^{l-1}, \cdots, \mathbf{k}_t^{l-1}\}$ indicate the leftward elements and the same to $\mathbf{V}_{\leq t}^{l-1}$. Despite its capability of learning global dependency, the transformer-based language model still has the tendency to overly rely on the short-term context and ignore the long-term context when predicting the next word, dubbed as ***explanation away*** problem (Holtzman et al. 2019). This problem is catastrophic for the STC task because the query acts as the long-term context in our language model formulation and not involving the query information is prone to generating the content irrelevant to the query. Therefore, explicitly modeling the relevance and emphasizing the importance of the query are essential. In this paper, we propose two components, namely, **S**upervised **S**ource **A**ttention (SSA) and **T**opic **I**nference (TI), to handle the ***explanation away*** problem.

Supervised Source Attention In the existing SEQ2SEQ-based frameworks, incorporating the query/source information is achieved by applying encoder-decoder attention solely on the encoder hidden representations. Similarly, attending only on the long-term context of language model is presumably beneficial for improving the relevance. Therefore, we propose to introduce another source attention layer on top of the self-attention layer. The computational formula of the t'-th ($t' \geq m$) query-enhanced hidden representation $\hat{\mathbf{h}}_{t'}^l$ is below:

$$\begin{aligned} \hat{\mathbf{h}}_{t'}^l, \boldsymbol{\beta}_{t'}^l &= \text{SRC-ATT}(\hat{\mathbf{q}}_{t'}^l, \hat{\mathbf{K}}^l, \hat{\mathbf{V}}^l) \\ \hat{\mathbf{Q}}^l &= \mathbf{H}^l\mathbf{W}^Q \\ \hat{\mathbf{K}}^l, \hat{\mathbf{V}}^l &= \mathbf{H}_{1:m}^l\mathbf{W}^K, \mathbf{H}_{1:m}^l\mathbf{W}^V \end{aligned} \tag{3}$$

SRC-ATT refers to our source attention layer on top of the self-attention layer. $\boldsymbol{\beta}_{t'}^l \in \mathbb{R}^m$ is the attention scores for the corresponding hidden representations of the query tokens. $\mathbf{H}^l$ is the output of SLF-ATT layer and $\hat{\mathbf{Q}}^l \in \mathbb{R}^{n \times \dim_h}$, $\hat{\mathbf{K}}^l$, $\hat{\mathbf{V}}^l \in \mathbb{R}^{m \times \dim_h}$ are the corresponding query, key, value in the source attention. Note that we only additionally apply source attention when the current token is not query token, i.e., $t' \geq m$, and do nothing in the preceding steps. Learning word alignment from data is possible but may be inaccurate without any supervision or external knowledge (Liu et al. 2016; Mi, Wang, and Ittycheriah 2016), therefore, we employ the keywords as the knowledge and enforce the source attention component to be concentrated on the important query tokens. First of all, we perform **max-over-time** pooling over the attention vectors $\boldsymbol{\beta}_{t'}^l \in \mathbb{R}^m$ ($t' \in [m+1, n]$) and induce the vector $\hat{\mathbf{y}}^{\text{src}} \in \mathbb{R}^m$ reflecting the salience scores of the query/source tokens:

$$\hat{\mathbf{y}}_i^{\text{src}} = \max\{\boldsymbol{\beta}_{m+1,i}^L, \cdots, \boldsymbol{\beta}_{n,i}^L\}, i \in [1, m] \tag{4}$$

Then, given the query keyword indicator vector $\mathbf{y}^{\text{src}} \in \{0, 1\}^m$, we introduce additional source attention loss $\mathcal{L}^{src}$ into Eq (1):

$$\mathcal{L}^{\text{src}} = \frac{1}{m}||\hat{\mathbf{y}}_i^{\text{src}} - \mathbf{y}^{\text{src}}||_2^2 \tag{5}$$

Ideally, the generation process will rely on more important query tokens if the salience score $\hat{\mathbf{y}}^{\text{src}}$ is more close to the keyword vector $\mathbf{y}^{\text{src}}$.

Topic Inference The SSA component attempts to improve the relevance by highlighting the importance of the important query tokens/words in the attention process. However, the range of the words topically related to the query is far more than that of the keywords explicitly mentioned in the query. Considering the query "*what is your favorite fruit?*" and two valid responses "*I like the watermelon very much*" and "*My favorite fruit is pineapple*", "fruit" should be emphasized during the generation but the words used to discuss fruit such as "watermelon" and "pineapple" are also very meaningful for building a response. Inspired by this, we collect the multiple references of each query in the training set and gather all of the keywords extracted from such responses[8]. To exploit the latent topic information, we introduce Topic Inference (KI) component to estimate the global topical word distribution based on the query representation $\mathbf{h}^q$ as follows:

$$\mathbf{h}^q = f(\mathbf{x}_{1:m}), \quad P(z|\mathbf{x}_{1:m}) = \text{Softmax}(\mathbf{W}^o\mathbf{h}^q) \tag{6}$$

where $f : \mathbb{R}^m \to \mathbb{R}^{\dim_h}$ denotes the function mapping the input query tokens to a low-dimensional query representation. Specifically, we feed the last query hidden representation in the transformer, namely, $\mathbf{h}_m^L$, into a linear layer with `tanh` activation and regard the output as the query representation $\mathbf{h}^q$ for simplifying the modeling part. To encode the

[6] The symbols for the feed-forward layer and residual connections are not shown.

[7] Here, the "query" refers to a real-valued vector while the "query" in the STC task is a sentence.

[8] (Xing et al. 2017) extend the keyword set using external corpus. Here, we focus on improving the relevance rather than enriching the topical words in the response, thus, we only utilize the training data to explore more keywords.

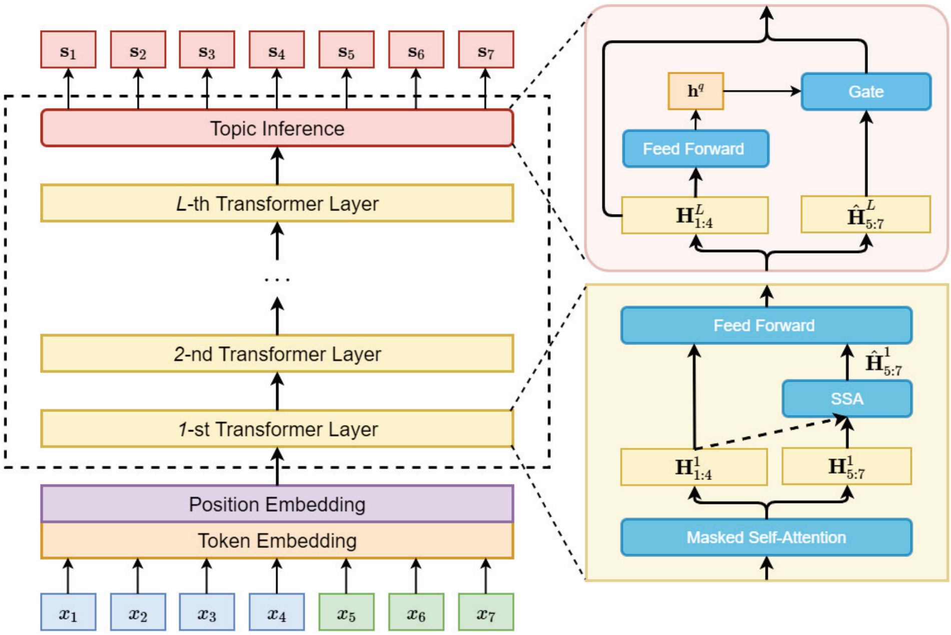

Figure 2: Overall architecture. The Topic Inference (TI) component on top of the transformer layers and the Supervised Source Attention (SSA) component inside the transformer layers are the proposed relevance-promoting components. Training losses are calculated on top of the obtained representation vectors $\mathbf{s}_t$'s.

topic information into the query representation, we employ the global keyword indicator vector $\mathbf{y}^{\text{kwd}} \in \{0,1\}^{|\mathcal{V}|}$ as supervision signals and enforce the components corresponding to keywords/important tokens in the query-based global topic distribution to be up-weighted. The computational formula is as follows:

$$\mathcal{L}^{\text{kwd}} = -\frac{1}{|\mathcal{V}|}\sum_{i=1}^{|\mathcal{V}|} \mathbf{y}_i^{\text{kwd}} \cdot \log P_i(z|\mathbf{x}_{1:m}) \tag{7}$$

where the subscript i denotes the i-th component of a vector and $|\mathcal{V}|$ is the vocabulary size. Note that we attempt to replace the Softmax in Eq 6 with the component-wise Sigmoid, typically used in multi-label classification problem, but the empirical results become worse. Thus, we keep the Softmax probability function unchanged in the experiment. Similar to Eq 5, the $\mathcal{L}^{\text{kwd}}$ will be added in the training loss.

Different from (Yao et al. 2017) and (Gao et al. 2019) regarding the concrete topic/keyword as the trigger of generation, we introduce the query representation encoding the global topic information as the supplementation for each token-level representation to encourage the generation of the relevant topical words. The representation vector $\mathbf{s}_t$ for predicting the output is calculated below:

$$\begin{aligned} \mathbf{s}_t &= \begin{cases} (1-g_t) * \mathbf{h}_t^L + g_t * \mathbf{h}^q & \text{, if } t > m \\ \mathbf{h}_t^L & \text{, Otherwise} \end{cases} \\ g_t &= \sigma(\mathbf{W}^g \mathbf{h}^q + \mathbf{W}^l \mathbf{h}_t^L + \mathbf{b}), \end{aligned} \tag{8}$$

where $g_t \in \mathbb{R}^{\text{dim}_h}$ is the gate value and $\mathbf{W}^g, \mathbf{W}^l \in \mathbb{R}^{\text{dim}_h \times \text{dim}_h}$ are parameter matrices in the TI component.

Model Training

The proposed SSA component and the TI component are jointly trained with the transformer-based language model. Based on Eq 1, Eq 5 and Eq 7, the overall training objective $\mathcal{L}(\theta)$ of the proposed model is as follow:

$$\begin{aligned} \mathcal{L}(\theta) &= \frac{1}{|\mathbb{D}|}\sum_{(\mathbf{x},\mathbf{y}^{\text{src}},\mathbf{y}^{\text{kwd}})\in\mathbb{D}} \mathcal{L}(\mathbf{x},\mathbf{y}^{\text{src}},\mathbf{y}^{\text{kwd}}) \\ \mathcal{L}(\mathbf{x},\mathbf{y}^{\text{src}},\mathbf{y}^{\text{kwd}}) &= \mathcal{L}^{\text{mle}} + \gamma_1\mathcal{L}^{\text{src}} + \gamma_2\mathcal{L}^{\text{kwd}} \end{aligned} \tag{9}$$

Here, γ_1 and γ_2 are the coefficients controlling the proportion of $\mathcal{L}^{\text{src}}$ and $\mathcal{L}^{\text{kwd}}$ involved in the training respectively.

Decoding

Due to the limited search space, it is difficult for the beam search or greedy search to find the interesting and diverse responses. Therefore, we do not adopt them but a "***randomization-over-maximization***" strategy (also know as 'top-k sampling") to perform the decoding, as done in (Fan, Lewis, and Dauphin 2018; Radford et al. 2019). (Holtzman et al. 2019) and (Ippolito et al. 2019) explore the usage of other advanced decoding strategies in the language generation task. Since our aim in this paper is not to compare the performances across the different decoding strategies, we consistently use the top-k sampling.

Experiment

Experiment Setup

We utilize the benchmark STC dataset (Liu et al. 2018) to evaluate the effectiveness of the proposed relevance-promoting transformer language model. This dataset is built based on the real conversations from `Weibo`[9] and contains about 7M high-quality query-response pairs. We split the dataset such that #train:#dev:#test is 7,024,156:2,000:800. Training details are provided in the appendix.

To avoid word segmentation errors and out-of-vocabulary issue, the tokens in our model and the baseline models are Chinese characters and the vocabulary size is about 12,000.

Evaluation Metrics

We introduce the following metrics to evaluate the model's capability of generating relevant and diverse responses:

Relevance Metrics We employ **BLEU-2**, **BLEU-3** & **BLEU-4** (Papineni et al. 2002) to estimate the relevance of the generated responses. Moreover, we also design two more metrics, namely, **HIT-Q** and **HIT-R** to calculate the hit rates of the topical words in the query and the response respectively. Firstly, we build a *high-precision-low-recall* keyword set for each query/response sentence based on keyword extraction toolkit[10] and filter some noisy words based on additional hand-crafted rules. Then, we calculate the HIT-Q$_i$ and HIT-R$_i$ for the i-th predictions as follows:

$$\text{HIT-Q}_i = \frac{|\mathbb{K}^{r_i} \cap \mathbb{K}^{q_i}|}{|\mathbb{K}^{r_i}|}, \text{HIT-R}_i = \frac{|\mathbb{K}^{r_i} \cap \mathbb{K}^{r_i^g}|}{|\mathbb{K}^{r_i}|} \quad (10)$$

where $\mathbb{K}^{q_i}$, $\mathbb{K}^{r_i}$ and $\mathbb{K}^{r_i^g}$ respectively denote the topical word set for the i-th query, predicted response and gold standard response. Then we obtain the HIT-Q and HIT-R by performing the corpus-level average:

$$\text{HIT-Q} = \frac{1}{N}\sum_{i}^{N}\text{HIT-Q}_i, \text{HIT-R} = \frac{1}{N}\sum_{i}^{N}\text{HIT-R}_i \quad (11)$$

Diversity Metrics Following (Li et al. 2016a), we employ **DIST-1** and **DIST-2** to calculate the ratios of the distinct uni-grams and bi-grams in the generated responses.

Human Evaluations We also conduct human evaluations. Specifically, we randomly sampled 100 queries and recruit five helpers to judge *Relevance* (4-scale rating, 0-3), *Fluency* (3-scale rating, 0-2) and *Acceptance* (0 or 1) of the generated responses from our model and the baselines. Details of the rating criteria are stated in the appendix.

Comparison Models

- **LSTM-LM** (Mei, Bansal, and Walter 2017): LSTM-based auto-regressive language model armed with incremental self-attention. We train LSTM-LM using the same strategy mentioned in this paper.
- **LSTM-S2S**: Attention-based LSTM Sequence-to-Sequence model.
- **TFM-S2S**: Transformer Sequence-to-Sequence model where the network components are identical to those in (Vaswani et al. 2017).
- **TFM-LM**: Transformer-based auto-regressive language model. We train TFM-LM using the same strategy mentioned in this paper.
- **MMI** (Li et al. 2016a): LSTM-S2S with Maximum Mutual Information objective in decoding. In this paper, we set the number of responses for re-ranking as 50.
- **CVAE** (Zhao, Zhao, and Eskenazi 2017)[11]: Conditional Variational Auto-Encoder for response generation. We replace the dialogue acts used in the original model with the keywords extracted from the references.
- **MMPMS** (Chen et al. 2019): The model with the state-of-the-art performance on the STC task. We re-run the officially released code[12] to obtain the results on our dataset.

Main Results

Table 1 and 2 list the automatic evaluation results and the human evaluation results respectively. In terms of BLEU, the proposed model with beam search decoding, namely, OURS-`bm`, consistently achieve the best scores. Besides, OURS-`bm` outperforms all compared models on the keyword-overlapping-based HIT metrics, suggesting that our model, armed with Supervised Source Attention component (SSA) and Topic Inference (TI) component, is beneficial for the generation of informative topical words related to the query. Surprisingly, OURS-`bm` also obtains better DIST metrics than the baseline models. After replacing the beam search with top-*k* sampling, our model (OURS-`tk`) is further enhanced in diversity modeling, reaching 0.107 and 0.544 on DIST-1 and DIST-2 respectively.

Regarding the more reliable human evaluations, both of OURS-`bm` and OURS-`tk` are the top-ranked models. Specifically, despite its unsatisfactory results on the automatic BLEU and HIT metrics, OURS-`tk` performs the best on the manually annotated *Relevance* metric with 5% improvement over the current state-of-the-art MMPMS model. Instead, OURS-`bm`, the best model on the automatic relevance metrics, still yields competitive results on the *Relevance*. It is reasonable because some words not appearing in the query/references, especially those not being frequently used, are still related to the discussed topic in the conversations. At the same time, such inconsistency between automatic and human evaluations demonstrates the effectiveness of top-*k* sampling, a ***randomization-over-maximization*** decoding strategy, in discovering infrequent but meaningful patterns for the STC task.

We now turn to discuss the performance of the other compared methods. Inheriting the powerful modeling capability of Transformer, TFM-S2S obtains the best automatic relevance scores as well as the second best *Relevance* among the baselines. TFM-LM, another Transformer-based baseline following the language model formulation in our paper, performs not as good as TFM-S2S on all of the metrics except *Fluency*, verifying the postulation that the ***explanation away*** issue of language model has the tendency to produce fluent but topically irrelevant responses. Despite of this, the TFM-LM outperforms LSTM-LM and LSTM-S2S, proving the superiority of Transformer to LSTM in response generation. Owing to the re-ranking mechanism,

[9] https://www.weibo.com/

[10] https://github.com/fxsjy/jieba

[11] https://github.com/snakeztc/NeuralDialog-CVAE

[12] https://github.com/PaddlePaddle/models

Model	Relevance					Diversity	
	BLEU-2	BLEU-3	BLEU-4	HIT-Q	HIT-R	DIST-1	DIST-2
LSTM-LM	3.8	0.9	0.3	0.084	0.066	0.028	0.094
LSTM-S2S	5.6	2.8	1.8	0.293	0.145	0.039	0.137
TFM-LM	6.9	3.2	2.1	0.295	0.144	0.058	0.259
TFM-S2S	7.3	3.5	2.3	0.369	0.172	0.078	0.290
MMI	7.9	2.5	1.0	0.197	0.145	0.093	0.349
CVAE	5.8	1.5	0.4	0.211	0.135	0.060	0.211
MMPMS	6.7	3.0	1.8	0.151	0.102	0.057	0.220
OURS-`tk` w/o SSA & TI	4.9	1.0	0.3	0.119	0.076	0.086	0.441
OURS-`tk` w/o SSA	5.5	2.1	1.5	0.150	0.146	0.102	0.521
OURS-`tk` w/o TI	5.1	2.1	1.4	0.171	0.132	0.090	0.445
OURS-`bm`	**10.3**	**5.3**	**3.4**	**0.510**	**0.193**	0.102	0.398
OURS-`tk`	6.0	3.6	2.5	0.191	0.152	**0.107**	**0.544**

Table 1: Experimental results on the automatic metrics. The best results are in **bold**.

Model	Evaluation Metrics		
	Relevance	*Fluency*	*Acceptance*
LSTM-LM	1.206	1.297	0.26
LSTM-S2S	1.386	1.285	0.37
TFM-LM	1.412	1.328	0.39
TFM-S2S	1.475	1.306	0.43
MMI	1.432	1.301	0.34
CVAE	1.316	1.274	0.33
MMPMS	1.528	1.396	0.42
OURS-`tk` w/o SSA & TI	1.273	1.368	0.28
OURS-`tk` w/o SSA	1.485	**1.407**	0.39
OURS-`tk` w/o TI	1.503	1.303	0.36
OURS-`bm`	1.515	1.359	0.38
OURS-`tk`	**1.606**	1.346	**0.44**

Table 2: Human evaluation results with the best ones in **bold**.

the MMI model is the strongest baseline on diversity modeling but OURS-`bm`/OURS-`tk` still achieves approximately 14%/55% improvement on DIST-2.

Ablation Study

In order to track the source of the performance gains, we also conduct the ablation study on the OURS-`tk`. The corresponding automatic and human evaluation results are shown in the second group of Table 1 and Table 2. As expected, the model without relevance-promoting design, i.e., OURS-`tk` w/o SSA & TI, is the worst one on the relevance metrics. OURS-`k` w/o SSA and OURS-`tk` w/o TI, the variants incorporating either TI or SSA for relevance modeling, boost the *Relevance* score by $\sim$17% and $\sim$18% respectively. Although they are comparable on the relevance metrics but the former achieves higher diversity scores (DIST-2: 0.521 v.s. 0.441). We attribute this phenomenon to the TI component, which exploits the usage of more related topical words mentioned in the multiple references. With the help of both SSA component and TI component, OURS-`tk` becomes the best model on *Relevance* and DIST metrics, demonstrating the necessity of the relevance modeling for the transformer language model. Another interesting finding is that the SSA component decreases the *Fluency* score (see the results of OURS-`tk` w/o TI), which indicates that fighting against ***explanation-away*** issue by incorporating additional query context may be coupled with corrupting the language model.

Case Study

Figure 3 shows example responses generated by our model and the most competitive baseline models. OURS-`tk`, which explicitly incorporates the query context and exploits the tokens potentially related to the query, always produces meaningful and informative responses. Taking the Query #1 & #2 as examples, the generated responses accurately respond to the query because they mention "flower ladder"/"matcha" and "cream", which are exactly the topics discussed in the conversations. The response for the Query #3 can easily engage user in the conversation and thus it is also a meaningful prediction. The outputs of TFM-LM are generally fluent. However, due to the ***explanation away*** issue, TFM-LM tends to generate the irrelevant response (Case #1) or response with phrase repetition (Case #2). Under the sequence-to-sequence formulation, TFM-S2S obtains the responses moderately related to the corresponding queries although the third output, directly copying part of the source text (i.e., query), is still unsatisfactory. MMPMS and MMI, the models aiming for promoting diversity, have chances to yield irrelevant responses.

Further Discussions on Top-*k* Sampling

We further investigate the impact of top-k sampling on the STC models. Firstly, we conduct additional automatic and human evaluations on the baseline models with results shown in Table 3. As can be seen, the top-*k* sampling consistently improves the DIST-2 score by a large margin on all models but the *Relevance* scores of LSTM-S2S, TFM-LM and TFM-S2S decrease after top-*k* sampling is applied. The variation trends of *Fluency* across the evaluated models are also inconsistent. These observations suggest that top-*k* sampling is simple yet effective to achieve diverse response generation but it should be carefully utilized in the model because of its uncertainty on relevance and fluency.

As discussed in Case Study, the transformer-based models adopting beam search have the tendency to generate the responses with repetition and those directly copying the query. We here investigate whether top-*k* sampling can help solve these issues. Figure 4 depicts the ratios of responses in the test set falling into the phrase repetition and query copy. The

TFM-LM	MMI	TFM-S2S	MMPMS	OURS-tk
Query #1: 有了这样的花梯，回家的感觉真好 (With this flower ladder, it feels so good going back home.)				
好漂亮的花，好喜欢 The flower is beautiful! Like it!	我也有这样的感觉 I have the same feeling!	我很喜欢！ I really like it!	我擦，这也太有才了吧 Oh my god! It is really funny	美得令人窒息的花梯 What a beautiful flower ladder!!
Query #2: 抹茶与奶油的结合，精致得不忍下口 (This is the combination of matcha and cream. It is quite delicate such that I couldn't bear to eat.)				
不忍下口，不忍下口。 Couldn't bear to eat, couldn't bear to eat.	好想吃啊！ I really want to have a try.	我能接受。 The food is acceptable for me.	我喜欢抹茶味的抹茶。 I like the matcha of matcha flavor.	我喜欢抹茶和奶油 I love matcha and cream!
Query #3: 移动的别墅。介个太强大了 (This moving cottage is awesome!)				
我也想要这样的房子 I wanna a house like this, too.	有创意，有创意啊 It is quite creative.	介个太强大了，房间装不下 It is too strong to fit in the room.	移动和联通 Chinese Mobile and Chinese Unicom	这是哪儿啊，怎么拍的，好厉害好酷哦 Where is it? How do you shoot this video. It look cool!

Figure 3: Examples of response generation. We translate Chinese samples to English.

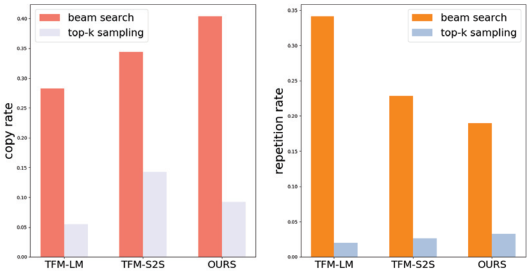

Figure 4: Comparison results on beam search and top-*k* sampling. Specifically, if the length of the longest common substring between response and query is larger than 4, then the response is regarded as a "copy" of query. If a response contains the word/phrase loop over 3 times, it is regarded as a response with repetition.

top-*k* sampling greatly reduces the query copy rate (about 72% on average) and almost eliminates the phrase repetition phenomenon in the Transformer-based models. However, note that Table 3 shows both TFM-LM and TFM-S2S perform worse on *Relevance* after using top-*k* sampling. We consider these results are consistent with human perception because enriching the morphology via sampling-based decoding strategy will inevitably introduce irrelevant information, leading to the degradation of relevance score. It is noticeable that the proposed model (i.e., OURS) is not affected on relevance modeling due to its capability of filtering some topically irrelevant candidates for the sampling process.

Models	*Relevance* (Δ)	*Fluency* (Δ)	DIST-2 (Δ)
LSTM-LM-tk	1.111 (-0.09)	1.270 (-0.03)	0.383 (+0.29)
LSTM-S2S-tk	1.439 (+0.05)	1.265 (-0.20)	0.490 (+0.35)
TFM-LM-tk	1.273 (-0.14)	**1.368** (+0.04)	0.441 (+0.18)
TFM-S2S-tk	1.270 (-0.15)	1.321 (+0.15)	0.507 (+0.22)
OURS-tk	**1.606** (+0.10)	1.346 (-0.13)	**0.544** (+0.20)

Table 3: Experimental results on the models adopting top-*k* sampling. Δ refers to the improvement over the original model adopting beam search. The best results are in **bold**.

Related Work

Short Text Conversation Short Text Conversation (STC) is usually formulated as a conditional text generation task (Shang, Lu, and Li 2015; Serban et al. 2016). The sequence-to-sequence (SEQ2SEQ) encoder-decoder framework (Cho et al. 2014; Sutskever, Vinyals, and Le 2014; Bahdanau, Cho, and Bengio 2015) and its variants have been studied extensively for solving this task. Li et al. 2016a introduce diversity-promoting decoding strategies into the SEQ2SEQ model. Some (Mou et al. 2016; Xing et al. 2017; Yao et al. 2017; Zhou et al. 2017; Gao et al. 2019) attempt to guide the SEQ2SEQ model to generate keyword/topic-aware responses while others (Wu et al. 2019; Cai et al. 2019) try to control the response generation with additional retrieved data. The advanced techniques such as RL, GAN and VAE are also considered for improving conversational experience (Li et al. 2016b; Xu et al. 2017; Du et al. 2018).

Transformer-based Language Model Deep transformer-based architecture (Vaswani et al. 2017) has led to significant performance gains on the language modeling task (Al-Rfou et al. 2019; Dai et al. 2019; Radford et al. 2019), compared to the existing CNN/RNN-based architectures (Dauphin et al. 2017; Merity, Keskar, and Socher 2018; Melis, Dyer, and Blunsom 2018). Meanwhile, GPT-2 (Radford et al. 2019) and UNILM (Dong et al. 2019) are the pioneer works adapting the transformer language model for the conditional text generation tasks.

Conclusion

In this paper, we present a language model based solution instead of traditional SEQ2SEQ paradigm for handling Short-Text Conversation (STC). We firstly tailor-make a training strategy to adapt the language model for the STC task. Then, we propose a relevance-promoting transformer language model to distill the relevance clues from the query as well as the topics inferred from the references, and incorporate them into the generation. Moreover, we explore the usage of top-*k* sampling for the STC task to further improve the response diversity. Experimental results on a large-scale STC dataset validate that our model is superior to the compared models on both relevance and diversity from automatic and human evaluations.

References

Al-Rfou, R.; Choe, D.; Constant, N.; Guo, M.; and Jones, L. 2019. Character-level language modeling with deeper self-attention. In *AAAI*, 3159–3166.

Bahdanau, D.; Cho, K.; and Bengio, Y. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Bengio, Y.; Ducharme, R.; Vincent, P.; and Jauvin, C. 2003. A neural probabilistic language model. *JMLR* 3(Feb):1137–1155.

Cai, D.; Wang, Y.; Bi, W.; Tu, Z.; Liu, X.; Lam, W.; and Shi, S. 2019. Skeleton-to-response: Dialogue generation guided by retrieval memory. In *NAACL*, 1219–1228.

Chen, C.; Peng, J.; Wang, F.; Xu, J.; and Wu, H. 2019. Generating multiple diverse responses with multi-mapping and posterior mapping selection. *arXiv preprint arXiv:1906.01781*.

Cheng, J.; Dong, L.; and Lapata, M. 2016. Long short-term memory-networks for machine reading. In *EMNLP*, 551–561.

Cho, K.; van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *EMNLP*, 1724–1734.

Dai, Z.; Yang, Z.; Yang, Y.; Carbonell, J.; Le, Q.; and Salakhutdinov, R. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *ACL*.

Dauphin, Y. N.; Fan, A.; Auli, M.; and Grangier, D. 2017. Language modeling with gated convolutional networks. In *ICML*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 4171–4186.

Dong, L.; Yang, N.; Wang, W.; Wei, F.; Liu, X.; Wang, Y.; Gao, J.; Zhou, M.; and Hon, H.-W. 2019. Unified language model pre-training for natural language understanding and generation. *arXiv preprint arXiv:1905.03197*.

Du, J.; Li, W.; He, Y.; Xu, R.; Bing, L.; and Wang, X. 2018. Variational autoregressive decoder for neural response generation. In *EMNLP*, 3154–3163.

Fan, A.; Lewis, M.; and Dauphin, Y. 2018. Hierarchical neural story generation. In *ACL*, 889–898.

Gao, J.; Bi, W.; Liu, X.; Li, J.; and Shi, S. 2019. Generating multiple diverse responses for short-text conversation. In *AAAI*.

Holtzman, A.; Buys, J.; Forbes, M.; and Choi, Y. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

Ippolito, D.; Kriz, R.; Sedoc, J.; Kustikova, M.; and Callison-Burch, C. 2019. Comparison of diverse decoding methods from conditional language models. In *ACL*, 3752–3762.

Khandelwal, U.; He, H.; Qi, P.; and Jurafsky, D. 2018. Sharp nearby, fuzzy far away: How neural language models use context. In *ACL*, 284–294.

Li, J.; Galley, M.; Brockett, C.; Gao, J.; and Dolan, B. 2016a. A diversity-promoting objective function for neural conversation models. In *NAACL*, 110–119.

Li, J.; Monroe, W.; Ritter, A.; Jurafsky, D.; Galley, M.; and Gao, J. 2016b. Deep reinforcement learning for dialogue generation. In *EMNLP*, 1192–1202.

Lin, Z.; Feng, M.; Santos, C. N. d.; Yu, M.; Xiang, B.; Zhou, B.; and Bengio, Y. 2017. A structured self-attentive sentence embedding. In *ICLR*.

Liu, L.; Utiyama, M.; Finch, A.; and Sumita, E. 2016. Neural machine translation with supervised attention. In *COLING*.

Liu, Y.; Bi, W.; Gao, J.; Liu, X.; Yao, J.; and Shi, S. 2018. Towards less generic responses in neural conversation models: A statistical re-weighting method. In *EMNLP*, 2769–2774.

Mei, H.; Bansal, M.; and Walter, M. R. 2017. Coherent dialogue with attention-based language models. In *AAAI*.

Melis, G.; Dyer, C.; and Blunsom, P. 2018. On the state of the art of evaluation in neural language models. In *ICLR*.

Merity, S.; Keskar, N. S.; and Socher, R. 2018. Regularizing and optimizing lstm language models. In *ICLR*.

Mi, H.; Wang, Z.; and Ittycheriah, A. 2016. Supervised attentions for neural machine translation. In *EMNLP*, 2283–2288.

Mou, L.; Song, Y.; Yan, R.; Li, G.; Zhang, L.; and Jin, Z. 2016. Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation. In *COLING*.

Nallapati, R.; Zhou, B.; dos Santos, C.; Gulçehre, Ç.; and Xiang, B. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *CoNLL*, 280–290.

Olabiyi, O., and Mueller, E. T. 2019. Dlgnet: A transformer-based model for dialogue response generation. *arXiv preprint arXiv:1908.01841*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Peters, M.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; and Zettlemoyer, L. 2018. Deep contextualized word representations. In *NAACL*, 2227–2237.

Radford, A.; Narasimhan, K.; Salimans, T.; and Sutskever, I. 2018. Improving language understanding by generative pre-training.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2019. Language models are unsupervised multitask learners. *OpenAI Blog* 1(8).

Ritter, A.; Cherry, C.; and Dolan, W. B. 2011. Data-driven response generation in social media. In *EMNLP*, 583–593.

Serban, I. V.; Sordoni, A.; Bengio, Y.; Courville, A.; and Pineau, J. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *AAAI*.

Shang, L.; Lu, Z.; and Li, H. 2015. Neural responding machine for short-text conversation. In *ACL*, 1577–1586.

Sutskever, I.; Vinyals, O.; and Le, Q. V. 2014. Sequence to sequence learning with neural networks. In *NeurIPS*, 3104–3112.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *NeurIPS*, 5998–6008.

Wu, Y.; Wei, F.; Huang, S.; Wang, Y.; Li, Z.; and Zhou, M. 2019. Response generation by context-aware prototype editing. In *AAAI*.

Xing, C.; Wu, W.; Wu, Y.; Liu, J.; Huang, Y.; Zhou, M.; and Ma, W.-Y. 2017. Topic aware neural response generation. In *AAAI*.

Xu, Z.; Liu, B.; Wang, B.; SUN, C.; Wang, X.; Wang, Z.; and Qi, C. 2017. Neural response generation via gan with an approximate embedding layer. In *EMNLP*, 617–626.

Yao, L.; Zhang, Y.; Feng, Y.; Zhao, D.; and Yan, R. 2017. Towards implicit content-introducing for generative short-text conversation systems. In *EMNLP*, 2190–2199.

Yu, L.; Blunsom, P.; Dyer, C.; Grefenstette, E.; and Kocisky, T. 2017. The neural noisy channel. In *ICLR*.

Zhang, Y.; Sun, S.; Galley, M.; Chen, Y.-C.; Brockett, C.; Gao, X.; Gao, J.; Liu, J.; and Dolan, B. 2019. Dialogpt: Large-scale generative pre-training for conversational response generation. *arXiv preprint arXiv:1911.00536*.

Zhao, T.; Zhao, R.; and Eskenazi, M. 2017. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *ACL*, 654–664.

Zhou, G.; Luo, P.; Cao, R.; Lin, F.; Chen, B.; and He, Q. 2017. Mechanism-aware neural machine for dialogue response generation. In *AAAI*.