Dialogue Generation on Infrequent Sentence Functions via Structured Meta-Learning

Yifan Gao[1], Piji Li[2], Wei Bi[2], Xiaojiang Liu[2], Michael R. Lyu[1], and Irwin King[1]

[1] Department of Computer Science and Engineering,
The Chinese University of Hong Kong, Shatin, N.T., Hong Kong
[2] Tencent AI Lab, Shenzhen, China
[1]{yfgao,king,lyu}@cse.cuhk.edu.hk [2]{pijili,victoriabi,kieranliu}@tencent.com

Abstract

Sentence function is an important linguistic feature indicating the communicative purpose in uttering a sentence. Incorporating sentence functions into conversations has shown improvements in the quality of generated responses. However, the number of utterances for different types of fine-grained sentence functions is extremely imbalanced. Besides a small number of high-resource sentence functions, a large portion of sentence functions is infrequent. Consequently, dialogue generation conditioned on these infrequent sentence functions suffers from data deficiency. In this paper, we investigate a structured meta-learning (SML) approach for dialogue generation on infrequent sentence functions. We treat dialogue generation conditioned on different sentence functions as separate tasks, and apply model-agnostic meta-learning to high-resource sentence functions data. Furthermore, SML enhances meta-learning effectiveness by promoting knowledge customization among different sentence functions but simultaneously preserving knowledge generalization for similar sentence functions. Experimental results demonstrate that SML not only improves the informativeness and relevance of generated responses, but also can generate responses consistent with the target sentence functions.

1 Introduction

Humans express intentions in conversations through sentence functions, such as interrogation for acquiring further information, declaration for making statements, and imperative for making requests and instructions. For machines to interact with humans, it is therefore essential to enable them to make use of sentence functions for dialogue generation. Sentence function is an important linguistic feature indicating the communicative purpose of a sentence in a conversation. There are four

Query	今天心情不错
	I am in a good mood today
Sentence Function	Positive Declarative
Response 1	什么让你这么开心？
	What makes you so happy?
Sentence Function 1	Wh-style Interrogative
Response 2	我今天心情很不好
	I feel bad today
Sentence Function 2	Negative Declarative

Figure 1: Query-response pairs with fine-grained sentence functions. Responses under different sentence functions are completely different in global structures.

major sentence functions: *Declarative*, *Interrogative*, *Exclamatory* and *Imperative* (Rozakis, 2003). Each major sentence function can be further decomposed into fine-grained ones according to different purposes indicated in conversations. For example, *Interrogative* is divided into *Wh-style Interrogative*, *Yes-no Interrogative* and other types. These fine-grained sentence functions have great influences on the structures of utterances in conversations including word orders, syntactic patterns, and other aspects (Akmajian, 1984; Yule, 2016). Figure 1 presents how sentence functions influence the responses. Given the same query expressed in *Positive Declarative*, the responses expressed in *Wh-style Interrogative* and in *Negative Declarative* are completely different.

Although the use of sentence functions improves the overall quality of generated responses (Ke et al., 2018), it suffers from the data imbalance issue. For example, in the recently released response generation dataset with manually annotated sentence functions STC-SeFun (Bi et al., 2019), more than 40% of utterances are *Positive Declarative* while utterances annotated with *Declarative with Interrogative words* account for less than 1% of the entire dataset. Therefore, dialogue generation mod-

els suffer from data deficiency for these infrequent sentence functions.

Recently, model-agnostic meta-learning (MAML) (Finn et al., 2017) has shown promising results on several low-resource natural language generation (NLG) tasks, including neural machine translation (Gu et al., 2018), personalized response generation (Madotto et al., 2019) and domain-adaptive dialogue generation (Qian and Yu, 2019). They treat languages of translation, personas of dialog and dialog domains as separate tasks in MAML respectively. In the same spirit of previous works, we first treat dialogue generation conditioned on different sentence functions as separate tasks, and meta-train a dialogue generation model using high-resource sentence functions. Moreover, we observe that sentence functions have hierarchical structures: four major sentence functions can be further divided into twenty fine-grained types. Some fine-grained sentence functions may share some similarities while some others are disparate. For example, utterances belong to *Wh-style Interrogative* and *Yes-no Interrogative* may share some transferable word patterns while utterances in *Wh-style Interrogative* and in *Exclamatory with interjections* totally differ from each other. Motivated by this observation, we explore a structured meta-learning (SML) considering inherent structures among fine-grained sentence functions. Inspired from recent advances on learning several initializations with a set of meta-learners (Yao et al., 2019; Vuorio et al., 2019), we develop our own approach to utilize the underlying structure of sentence functions. More specifically, our proposed SML explicitly tailors transferable knowledge among different sentence functions. It utilizes the learned representations of fine-grained sentence functions as parameter gates to influence the globally shared parameter initialization. Therefore, conversation models for similar sentence functions can share similar parameter initializations and vice versa. As a result, SML enhances meta-learning effectiveness by promoting knowledge customization among different sentence functions but simultaneously preserving knowledge generalization for similar sentence functions.

The experimental results on STC-SeFun dataset (Bi et al., 2019) show that responses generated from our proposed structured meta-learning algorithm are of better quality over several baselines in both human and automatic evaluations. Moreover, our proposed model can generate responses consistent with the target sentence functions while baseline models may ignore the target sentence functions or generate some generic responses. We further conduct a detailed analysis on our proposed model and show that it indeed can learn word orders and syntactic patterns for different fine-grained sentence functions.

2 Background

Dialogue Generation with Sentence Function. Open domain dialogue generation has been widely studied with sequence-to-sequence learning (Seq2Seq) (Sutskever et al., 2014). To alleviate the generate generic and dull responses issue of Seq2Seq (Li et al., 2016), some efforts provide additional controlling signals in dialogue generation, such as emotion (Zhou et al., 2017), persona (Zhang et al., 2018), and topic (Xing et al., 2017). Different from these local controlling factors, sentence function can influence the global structure of the entire response such as changing word orders and word patterns. Zhao et al. (2017) utilize dialogue acts as prior linguistic knowledge and integrate it with conditional variational autoencoders to achieve the discourse-level diversity of generated responses. Ke et al. (2018) adopt a conditional variational autoencoder to capture various word patterns and introduce a type controller to control the sentence function. Xu et al. (2019) generalize the concept of dialogue act into meta words and use meta words for open domain dialogue generation.

Meta-Learning for Low-Resource NLG. Humans can learn quickly with a few examples while data-driven models are mostly compute-intensive. In meta-learning, the goal of the trained model is to quickly learn a new task from a small amount of data. Therefore, the model should be able to learn transferable knowledge on a large number of different tasks. Recently, model-agnostic meta-learning (MAML) (Finn et al., 2017) has shown promising results on several few-shot classification tasks. MAML directly optimizes the gradient towards a good parameter initialization for easy fine-tuning on low-resource scenarios. Because of the model-agnostic nature of MAML, it can be directly applied to low-resource NLG tasks with modifications on corresponding training strategies. Gu et al. (2018) frame machine translation between two language pairs as a single task in meta-learning,

and learn to adapt to low-resource languages based on multilingual high-resource language tasks. In a similar spirit, recent works apply MAML to personalized response generation (Madotto et al., 2019) and task-oriented dialogue agents (Mi et al., 2019; Qian and Yu, 2019). In this paper, we not only investigate how MAML helps for open domain dialogue generation on infrequent sentence functions, but also develop a structured approach to fit the hierarchical structure of sentence functions.

3 Problem Formulation

We define response generation conditioned on every query-response sentence function pair (d_X, d_Y) as a single task. As the number of utterances for different sentence functions is extremely imbalanced, some tasks have abundant utterances while some others are low-resource. We take K high-resource tasks as training data, denoted as:

$$D_{train}^k = \{(X_n^k, Y_n^k, d_X^k, d_Y^k), n = 1...N\}, k = 1...K \quad (1)$$

Then, we take T tasks with infrequent sentence functions as target tasks, denoted as:

$$D_{target}^t = \{(X_n^t, Y_n^t, d_X^t, d_Y^t), n = 1...N'\}, t = 1...T \quad (2)$$

where $N' \ll N$.

During training, taking the query X, its sentence function d_X and the target response sentence function d_Y as inputs, a dialog model f parameterized by θ learns the mapping between inputs and the corresponding response Y using training data D_{train} of K tasks,

$$f_\theta : X^k \times (d_X^k, d_Y^k) \rightarrow Y^k, k = 1...K \quad (3)$$

The initialization parameters of model f learned from the training process, denoted by θ_0, are used as the initialization parameters in the adaptation process. The adaptation process on each target task D_{target}^t can be formulated as follows:

$$f_{\theta^*} = \arg\max_\theta \log p(f_\theta | D_{target}^t, f_{\theta_0}) \quad (4)$$

where f_{θ^*} is the fine-tuned model that could perform well on the target task D_{target}^t.

4 Proposed Approach

In this section, we first introduce the conditional sequence-to-sequence (C-Seq2Seq) model for open domain dialogue generation with fine-grained sentence function. Then we describe how to meta-train

C-Seq2Seq under the algorithm of model-agnostic meta-learning. Finally, we explore the structure of fine-grained sentence functions and propose the structured meta-learning (SML) algorithm.

4.1 C-Seq2Seq

Conditional sequence-to-sequence (C-Seq2Seq) (Ficler and Goldberg, 2017) is the best generative model on STC-SeFun dataset (Bi et al., 2019). We use it to test the effectiveness of our proposed structured meta-training approach. C-Seq2Seq follows the widely used encoder-decoder framework (Sutskever et al., 2014; Vinyals et al., 2015). The encoder transforms the query X into contextualized representation $(\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_n)$ through bidirectional LSTMs (Hochreiter and Schmidhuber, 1997). For the decoder part, we learn an additional sentence function embedding $[\mathbf{s}_1, \mathbf{s}_2, ..., \mathbf{s}_K]$ for each query-response sentence function pair, which plays a major role in our structured modeling (Sec. 4.3). Then we takes the concatenation of word embedding $\mathbf{w}_t$ and the sentence function embedding $\mathbf{s}_k$ as input at each timestep, and updates its hidden state as follows,

$$\mathbf{u}_t = \text{LSTM}(\mathbf{u}_{t-1}, [\mathbf{w}_t; \mathbf{s}_k]). \quad (5)$$

The decoder utilizes soft attention mechanism (Luong et al., 2015) to derive the context vector $\mathbf{c}_t$,

$$a_{t,i} = \frac{\exp(\mathbf{u}_t^\top \mathbf{W}_a \mathbf{h}_i)}{\sum_j \exp(\mathbf{u}_t^\top \mathbf{W}_a \mathbf{h}_j)}, \ \mathbf{c}_t = \sum_i a_{t,i} \mathbf{h}_i. \quad (6)$$

Finally, the predicted probability distribution over the vocabulary V is computed as:

$$\tilde{\mathbf{h}}_t = \tanh(\mathbf{W}_h [\mathbf{u}_t; \mathbf{c}_t]), \quad (7)$$

$$\mathbf{P}_V = \text{softmax}(\mathbf{W}_V \tilde{\mathbf{h}}_t + \mathbf{b}_V), \quad (8)$$

where $\mathbf{W}_a$, $\mathbf{W}_h$, $\mathbf{W}_V$ and $\mathbf{b}_V$ are trainable parameters.

4.2 Meta-Learning for C-Seq2Seq

The fundamental idea behind meta-learning is based on a simple machine learning principle: test and train conditions must match. In the context of meta-learning, it becomes that the conditions between task adaptation (fine-tuning) stage (Eqn. 4) and meta-training stage (Eqn. 3) must match. To mimic the task adaptation stage, Model-Agnostic Meta-Learning (MAML) (Finn et al., 2017) explicitly train the parameters of the model such that a small number of gradient steps with a small amount
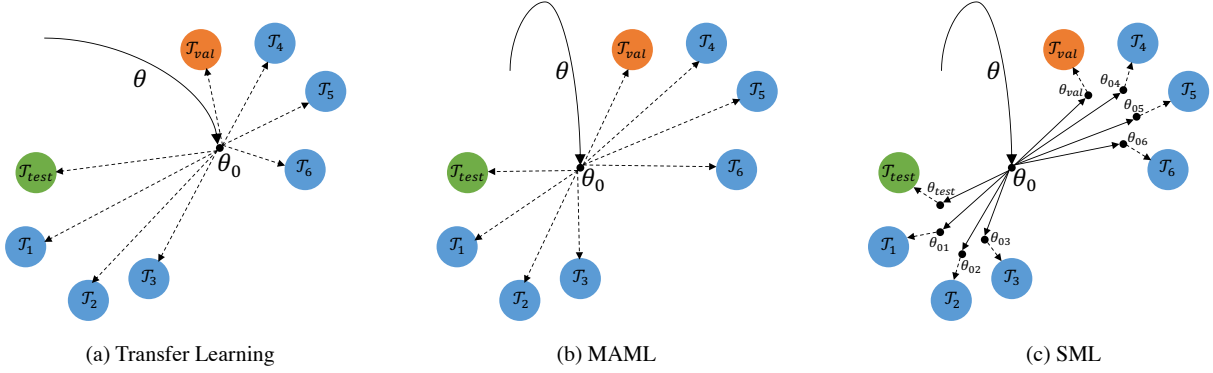
(a) Transfer Learning   (b) MAML   (c) SML

Figure 2: An intuitive comparison among (a) Transfer Learning, (b) Model-Agnostic Meta-Learning (MAML) and (c) Structured Meta-Learning (SML). Tasks of training, validation and testing are colored in blue, orange and green respectively. Solid lines represent the learning of initialization θ_0 while dashed lines show the path of fine-tuning. Our structured modeling can learn the structure in different sentence functions so that similar tasks will be initialized from closer starting points than others ($T_{1,2,3}$ and $T_{4,5,6}$ in (c)). In the testing (adaptation) stage, a new sentence function such as Negative Declarative will benefit from this learned structure by initializing from a point that is close to other fine-grained sentence functions in the same category of Declarative (T_{test} in (c)).

of training data will make rapid progress on new tasks. The intuition behind MAML is that there exist some transferable internal representations across tasks. MAML aims to find the most sensitive model parameters such that small changes in model parameters will produce large improvements on each task.

Here is how we apply MAML to response generation on infrequent sentence functions. We uniformly sample one source task $\mathcal{T}_k$ at random. Then we independently sample two subsets of data $(D_{\mathcal{T}_k}, D'_{\mathcal{T}_k})$ from task $\mathcal{T}_k$. $D_{\mathcal{T}_k}$ is used to simulate the process when f_θ adapts to the target low-resource tasks while $D'_{\mathcal{T}_k}$ is used to evaluate the outcome of the adapted model.

In the simulation of adaptation stage, the model f parameterized by θ adapts to this new task $\mathcal{T}_k$ using one or more gradient descent updates,

$$\theta'_k = \theta - \alpha \nabla_\theta \mathcal{L}^{D_{\mathcal{T}_k}}(f_\theta), \qquad (9)$$

where α is a hyperparameter for task-specific learning rate. Then the model evaluates the updated parameters θ'_k towards $D'_{\mathcal{T}_k}$. The loss can be formulated as,

$$\mathcal{L}^{D'_{\mathcal{T}_k}}(f_{\theta'_k}) = \mathcal{L}^{D'_{\mathcal{T}_k}}(f_{\theta - \alpha \nabla_\theta \mathcal{L}^{D_{\mathcal{T}_k}}(f_\theta)}) \qquad (10)$$

Afterward, the model is trained by optimizing the performance of $\mathcal{L}(f_{\theta'_k})$ with respect to θ across randomly sampled tasks. To learn the internal representation shared across tasks, it is possible to aggregate gradients $\nabla_\theta \mathcal{L}(f_{\theta'_k})$ sampled from several tasks in the meta-update,

$$\theta \leftarrow \theta - \beta \sum_k \nabla_\theta \mathcal{L}(f_{\theta'_k}), \qquad (11)$$

where β is the meta learning rate across tasks and k is the sampled tasks for gradient aggregation[1]. Different from common gradient-based approaches, Eqn. 11 update the model not from θ'_k but from θ because MAML aims to learn the most sensitive parameters to facilitate fast adaptation. As a result, the meta-learned model is not necessarily a good model on its own, but it adapts fast on any new task with a few gradient update steps.

4.3 Exploring Structure Modeling

MAML learns some transferable knowledge in different training tasks (a task in our paper is defined as response generation conditioned on a given sentence function). In effect, the meta-learned model can adapt fast for the low-resource testing tasks (sentence functions). However, MAML assumes all tasks in training and adaptation stages distributed uniformly, which is not the case for our conditioned response generation – some tasks may share some similarities while some are exclusive to each other. For example, utterances belong to *Wh-style Interrogative* and *Yes-no Interrogative* may share some transferable word patterns while word patterns in *Wh-style Interrogative* and *Exclamatory with interjections* are totally different. Therefore, we propose

---

[1]In our implementation, we follow Finn et al. (2017) to adopt a first-order approximation for the meta-gradient update to reduce the computational complexity.

434

to represent sentence functions explicitly through learned embeddings $\mathbf{s}_1; ...; \mathbf{s}_K$ (Sec. 4.1). Then sentence function embeddings are used to interact with each other via a gated self-attention mechanism, which can be viewed as a clustering process to make similar sentence function embeddings close to each other. Finally, the self-attended representations of these sentence functions are used as parameter gates to tailor the transferable knowledge of the meta-learned prior parameters. Recently, the task-aware modulation problem in meta-learning is also investigated in the machine learning community (Yao et al., 2019; Vuorio et al., 2019). In their approaches, they first learn the task mode from input data as a vectorized representation and use the identified mode to modulate the meta-learned prior parameters. The key difference between our structured modeling and their approaches is our model does not learn the task-aware representation through input data because the sentence functions are predefined before generating the responses. Moreover, we propose the gated self-attentive approach to learn the underlying structure which has never been used before.

Task Representation Learning. To ensure similar tasks share similar representations, we design a gated self-attention module for sentence function embeddings. For any fine-grained sentence function k, we first match the sentence function embedding matrix $\mathbf{S} = [\mathbf{s}_1; ...; \mathbf{s}_K]$ with itself $\mathbf{s}_k$ to compute the self-matching representation $\mathbf{m}_k$, and then combine it with the original representation $\mathbf{s}_k$:

$$\mathbf{a}_k = \mathrm{softmax}(\mathbf{S}^\top \mathbf{s}_k), \ \mathbf{m}_k = \mathbf{S}\mathbf{a}_k \quad (12)$$
$$\mathbf{f}_k = \tanh(\mathbf{W}_f[\mathbf{s}_k; \mathbf{m}_k]), \quad (13)$$

The self-matching operation matches the sentence form embedding to other sentence form embeddings, which can be viewed as a clustering process so that embeddings from similar sentence forms will be close to each other.

The final representation $\tilde{\mathbf{s}}_k$ is derived via a gated summation through a learnable gate vector $\mathbf{g}_k$,

$$\mathbf{g}_k = \mathrm{sigmoid}(\mathbf{W}_g[\mathbf{s}_k; \mathbf{m}_k]) \quad (14)$$
$$\tilde{\mathbf{s}}_k = \mathbf{g}_k \odot \mathbf{f}_k + (1 - \mathbf{g}_k) \odot \mathbf{s}_k \quad (15)$$

where $\mathbf{W}_f, \mathbf{W}_g$ are learnable weights, $\odot$ is the element-wise multiplication. For sentence function embeddings $\mathbf{s}_{new}$ in the adaptation stage, we use the already well-learned sentence function embeddings $\mathbf{S} = [\mathbf{s}_1; ...; \mathbf{s}_K]$ in the meta-training

---

Algorithm 1 Meta-training of SML

Require: $\mathcal{E}$: distribution over tasks $\{\mathcal{T}_1, ..., \mathcal{T}_K\}$
Require: α, β: step size hyperparameters
1: Randomly initialize θ
2: **while** not done **do**
3:     Sample a batch of tasks $\mathcal{T}_k \sim \mathcal{E}$
4:     **for** all $\mathcal{T}_k$ **do**
5:         Sample $D_{\mathcal{T}_k}, D'_{\mathcal{T}_k}$ from $\mathcal{T}_k$
6:         Compute task representation $\tilde{\mathbf{s}}_k$ in Eqn. 15
7:         Compute θ_{0k} in Eqn. 16
8:         Evaluate $\nabla_{\theta_{0k}} \mathcal{L}(f_{\theta_{0k}})$ with respect to $D_{\mathcal{T}_k}$
9:         Update $\theta'_{0k} = \theta_{0k} - \alpha \nabla_{\theta_{0k}} \mathcal{L}(f_{\theta_{0k}})$ in Eqn. 9
10:     **end for**
11:     Update $\theta \leftarrow \theta - \beta \sum_k \nabla_{\theta'_{0k}} \mathcal{L}(f_{\theta'_{0k}})$ in Eqn. 11
        with respect to all $D'_{\mathcal{T}_k}$
12: **end while**

---

stage, concatenate $\mathbf{s}_{new}$ with learned embeddings as $\mathbf{S}' = [\mathbf{s}_1; ...; \mathbf{s}_K; \mathbf{s}_{new}]$ and apply Eqn. 12 $\sim$ 15 for task representation learning in the adaptation stage. Because sentence form embeddings seen in training are learned to build the underlined structure: similar sentence functions are clustered close to each other. In the adaptation phase, the unseen sentence function embedding can adapt fast by moving to the cluster it belongs to. For example, a new sentence form Yes-no Interrogative can learn some transferable knowledge from trained sentence forms under the Interrogative category.

Task-Specific Knowledge Adaptation. To adapt globally transferable knowledge θ_0 to each sentence function, we design a parameter gate $\mathbf{o}_k$ for θ_0,

$$\mathbf{o}_k = \mathrm{FC}^\sigma_{\mathbf{W}_p}(\tilde{\mathbf{s}}_k), \quad \theta_{0k} = \theta_0 \odot \mathbf{o}_k \quad (16)$$

where $\mathrm{FC}^\sigma_{\mathbf{W}_p}$ is a fully connected layer parameterized by $\mathbf{W}_p$ and activated by a sigmoid function σ, $\odot$ is the element-wise multiplication. Intuitively, sentence functions with similar representations will activate similar initial parameters while dissimilar sentence functions trigger different ones. One major problem for Eqn. 16 is that it introduces dozens of parameters compared to θ_0 to achieve the element-wise dot product with θ_0. Here we only tailor parameters in the decoder to reduce the total amount of learnable parameters.

SML Algorithm and Visualization. The whole algorithm of our proposed model is detailed in Algorithm 1. Figure 2 visually illustrates the difference between transfer learning, model-agnostic

meta-learning (MAML) and our proposed structured meta-learning (SML). All methods in Figure 2 use tasks $\{\mathcal{T}_1, ..., \mathcal{T}_6\}$ for training, $\mathcal{T}_{val}$ for validation and $\mathcal{T}_{test}$ for target task adaptation. Transfer learning in Figure 2(a) solves the training tasks $\{\mathcal{T}_1, ..., \mathcal{T}_6\}$ in the multi-task learning approach without knowing the adaptation task $\mathcal{T}_{test}$. It aims at solving tasks in training set and select models based on the validation task $\mathcal{T}_{val}$. MAML in Figure 2(b) tries to learn transferable representation by repeatedly simulating the learning process in low-resource task-specific learning. In effect, the learned model can adapt fast to any unseen task, such as $\mathcal{T}_{test}$. SML in Figure 2(c) additionally explores the structure across tasks so that similar tasks will be initialized from closer starting points than others. In the testing (adaptation) stage, a new sentence function such as Negative Declarative will benefit from this learned structure by initializing from a point that is close to other fine-grained sentence functions in the same category of Declarative.

5 Experimental Settings

Dataset. We conduct experiments on STC-SeFun dataset (Bi et al., 2019) which is a large-scale Chinese short text conversation dataset with manually labeled sentence functions. Utterances in STC-SeFun have two-level sentence function labels. The four major sentence function types include: *Declarative*, *Interrogative*, *Imperative* and *Exclamatory*. Each major sentence function is further divided into fine-grained sentence function labels like *Wh-style Interrogative* which in total is 20 categories. Considering all query and response sentence functions, we could have $20 \times 20 = 400$ meta tasks. However, some tasks are extremely low-resource with less than 100 samples. Incorporating these tasks as our adaptation tasks leads to a high variance of test performance. To establish concrete evaluation, we only consider tasks with more than 700 samples, in which 100 samples are used for validation in the adaptation stage and 500 samples are used for the final testing. Under this constraint, we receive 18 query-response fine-grained sentence function pairs as 18 tasks. We select 9 high-resource tasks for meta-training, 4 tasks for meta-validation and 5 tasks for testing (adaptation). The dataset statistics is shown in Table 1. Although 18 tasks are far smaller than thousands of tasks for few-shot image classification, it is still comparable to previous works such as low-resource machine translation

	Query SF	Response SF	# Samples		
Meta Train	Positive DE	Positive DE	27058		
	Wh-style IN	Positive DE	12854		
	Positive DE	Negative DE	5831		
	Negative DE	Positive DE	4006		
	Positive DE	Wh-style IN	3935		
	A-not-A IN	Positive DE	3508		
	Wh-style IN	Negative DE	3367		
	Yes-no IN	Positive DE	3267		
	Negative DE	Negative DE	2466		
Meta Val	Wh-style IN	DE w/ IN words	271	100	500
	Negative DE	Wh-style IN	161	100	500
	Positive DE	EX w/ interjections	134	100	500
	Positive DE	DE w/ IN words	120	100	500
Meta Test	Positive DE	Yes-no IN	1314	100	500
	Yes-no IN	Negative DE	893	100	500
	Positive DE	EX w/o tone words	846	100	500
	A-not-A IN	Negative DE	684	100	500
	Wh-style IN	Wh-style IN	488	100	500

Table 1: Dataset statistics for our experiments. SF: Sentence Function; DE: Declarative; IN: Interrogative; EX: Exclamatory. For meta validation and meta test tasks, samples are further split into train, validation (100 samples) for task-specific adaptation. The rest 500 samples are used to test the performance of adapted models.

(Gu et al., 2018). Moreover, the experimental results may generalize to more tasks if there is more data.

Baselines and Ablations. We train the model C-Seq2Seq described in Section 4.1 under the following learning settings:

• Multi-Task Learning (MTL): We train C-Seq2Seq under the multi-task learning approach with training and validation data. Then we directly apply the trained model on each target task without fine-tuning.
• Multi-Task Learning + Fine-tuning (MTL+FT): We train C-Seq2Seq under the same training paradigm as multi-task learning. Then we fine-tune the model on each target task. This setting corresponds to a transfer learning scenario.
• Model-Agnostic Meta-Learning (MAML): We meta-train the model under the methodology model-agnostic meta-learning in other low-resource NLG tasks (Gu et al., 2018; Qian and Yu, 2019). Then we fine-tune the meta-trained model on each target task. This is the ablation without structure modeling (Sec. 4.3).
• Structured Meta-Learning (SML): We meta-train the model using structured meta-learning described in Algorithm 1. Then we fine-tune the meta-trained model on each target task.

Model Settings. We take the most frequent 30k words as our vocabulary and use the pretrained em-

Query SF	Response SF	Model	Human Evaluation				Automatic Evaluation					
			Flue	Rele	Info	Accu	FT Step	PPL	B1	B2	Dist1	Dist2
Positive DE	Yes-no IN	MTL	59.40	50.40	36.53	0.33	n/a	177.44	4.59	1.49	0.11	0.16
		MTL+FT	62.47	53.93	39.87	34.00	304.00	91.65	10.47	3.38	0.17	0.36
		MAML	63.40	55.87	39.87	57.67	111.33	87.55	12.20	4.47	0.18	0.39
		SML	64.27	56.00	40.13	69.00	104.00	87.28	12.88	4.72	0.21	0.48
Yes-no IN	Negative DE	MTL	60.47	57.07	49.87	6.00	n/a	73.04	5.46	1.46	0.10	0.14
		MTL+FT	61.07	56.80	54.00	73.00	195.00	56.85	9.98	3.37	0.22	0.40
		MAML	62.00	59.53	53.67	91.00	111.00	58.79	11.12	3.81	0.16	0.26
		SML	64.93	57.80	55.93	91.00	102.33	58.65	11.91	3.76	0.13	0.20
Positive DE	EX without tone words	MTL	57.13	53.53	35.40	1.00	n/a	100.21	6.06	2.69	0.14	0.23
		MTL+FT	56.40	53.67	36.67	39.00	268.00	80.32	12.20	4.26	0.12	0.22
		MAML	65.33	56.20	39.27	71.00	89.33	68.08	14.09	4.61	0.19	0.40
		SML	65.80	57.13	40.93	68.00	91.33	71.71	14.94	4.94	0.17	0.33
A-not-A IN	Negative DE	MTL	60.33	54.93	49.73	4.33	n/a	73.99	5.89	1.77	0.10	0.17
		MTL+FT	62.13	55.13	51.47	53.67	158.00	59.53	9.79	3.41	0.15	0.27
		MAML	62.60	55.20	51.27	89.33	114.33	60.35	10.39	3.60	0.13	0.20
		SML	63.27	56.00	52.80	96.00	105.00	58.24	11.28	3.96	0.12	0.19
Wh-style IN	Wh-style IN	MTL	62.47	51.67	38.33	1.00	n/a	97.24	7.63	2.33	0.14	0.22
		MTL+FT	63.60	52.60	39.13	22.33	167.00	61.70	7.98	2.58	0.17	0.30
		MAML	64.07	53.13	43.33	85.00	88.00	44.02	7.84	3.31	0.19	0.46
		SML	64.13	53.80	45.20	88.00	83.33	43.06	8.04	3.96	0.19	0.43

Table 2: Human evaluation results (in percentage %) and automatic evaluation results in five testing tasks. The best/second-best results are bold/underlined except automatic metrics which are inconsistent with human perceptions (Liu et al., 2016). Note that MTL does not fine-tune on target tasks, so FT Step is not applicable to this setting.

Metrics	Flue	Rele	Info	Accu
Fleiss's Kappa κ	0.61	0.72	0.67	0.90

Table 3: Fleiss's Kappa score for evaluating the inter annotator agreement.

beddings (Song et al., 2018) for initialization. The sentence function embedding with dimension 20 is randomly initialized and learned through training. We use two-layer LSTMs in both encoder and decoder, and the LSTMs hidden unit size is set to 400. We use dropout (Srivastava et al., 2014) with the probability $p = 0.3$. All trainable parameters, except word embeddings, are randomly initialized with the uniform distribution in $(-0.1, 0.1)$. We adopt the teacher-forcing for the training. In the testing, we select the model with the lowest perplexity and beam search with size 5 is employed for generation. All hyper-parameters and models are selected on the validation dataset.

Learning Settings. We use SGD as the optimizer with a minibatch size of 64 and an initial learning rate of 1.0 for both meta-learning (line 9 and line 11 in Algorithm 1) and multi-task learning. For meta-learning, we sample 3 tasks for line 3 in Algorithm 1 and take a single gradient step for line 9 and line 11 in Algorithm 1. We meta-train the model for 8 epochs and start having the learning rate after the 3 epoch. All models are fine-tuned with a SGD optimizer with a minibatch size of 64 and learning rate of 0.1. We set the gradient norm

upper bound to 3 and 1 during the training and fine-tuning respectively. To avoid any random results, we report the average of five runs for all results.

Evaluation Metrics. Since automatic metrics for open-domain conversations may not be consistent with human perceptions (Liu et al., 2016), we hire 5 full-time human judges from a third-party data annotation company. We provide them a detailed annotation guideline (in Chinese) with good and bad response samples for each metric. They are first asked to annotate 100 responses for trial, and we select the top 3 judges according to the annotation quality. Finally, the 3 selected judges independently evaluate 2,000 responses generated from our model and three baselines for all five adaptation tasks. For each query, four responses generated from the proposed model, and three baselines are randomly shuffled to reduce the priming effect.

The annotators evaluate responses on four metrics: (1) "Fluency" (Flue) measures the grammatical correctness of responses; (2) "Relevance" (Rele) measures whether the response is a relevant reply to the query; (3) "Informativeness" (Info) evaluates whether the response provides any meaningful information with regard to the query; (4) "Accuracy" (Accu) evaluates whether the response is coherent with the given response sentence function. "Fluency", "Relevance" and "Informativeness" are graded independently in a 1-5 scale where 5 is the best. "Accuracy" takes a binary value (1 or 0). We further normalize the average scores over

all rated samples into $[0, 1]$. Besides, we compare the adaption time of all models by calculating the "Fine-Tuning Step" (`FT Step`) till convergence on each test task. For completeness, we also show the following automatic evaluation metrics: "Perplexity" (`PPL`); "BLEU-1/2" (`B1`/`B2`) (Papineni et al., 2002); "Distinct-1/2" (`Dist1`/`Dist2`) (Li et al., 2016).

6 Results

Performance of Human Evaluation. Human evaluation on "Fluency" (`Flue`), "Relevance" (`Rele`), "Informativeness" (`Info`) and "Accuracy" (`Accu`) are shown in Table 2. The inter annotator agreement κ scores are shown in Table 3. We can make the following observations:

- MTL receives the worst performance on all human evaluation metrics. Recall that MTL is trained on a mixture of all training tasks. It can only learn some generic response patterns like "So do I". That is why MTL has the worst performance on "Informativeness" metric. Moreover, since MTL never sees responses in target sentence functions in training, the generated responses are not coherent with the given sentence function at all.
- MTL+FT achieves better performance than MTL because it further fine-tunes on each target dataset. However, the performance on the accuracy of target sentence function is still unsatisfactory. This reveals that fine-tuning may not solve the adaptation problem on low-resource tasks.
- SML and MAML achieve the best/second-best human evaluation results across most of the metrics. This indicates that by simulating the low-resource testing scenarios in meta-training, the learned model adapts well on the low-resource testing tasks. Moreover, there is a huge improvement on the accuracy of given sentence functions (Accu), which reveals that MAML/SML can find model parameters that are sensitive to changes in the new task, such that small changes in the parameters produce large improvement on the accuracy of sentence functions.
- SML outperforms MAML in most of the cases. This tells us exploring the structure of sentence functions can balance knowledge generation and knowledge customization. The task-specific initialized model can leverage the knowledge of
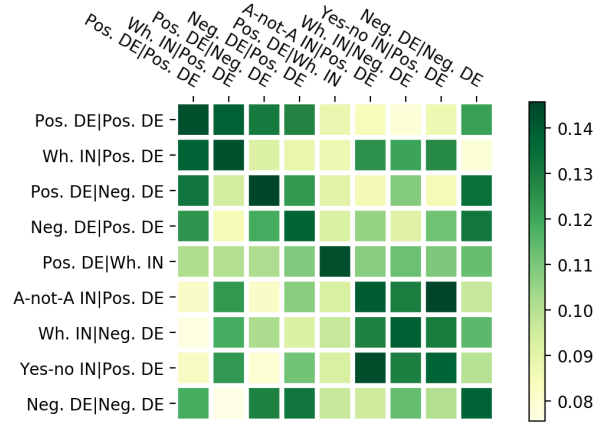

Figure 3: Heatmap of the self-attention weight matrix. Each row shows the attention distribution $\mathbf{a}_k$ in Eqn. 12 for a given query-response sentence function pair (denoted in "Query | Response" format).

similar tasks and thus adapts the target tasks better.

Performance of Automatic Evaluation. We also show the results of automatic evaluations in Table 2. Compared to transfer learning based model MTL+FT, our meta-learning based models adapt faster (lower fine-tune step) and better (lower perplexity). Although BLEU is not reliable enough to evaluate response generation, MAML and SML still achieve slightly better results than baselines. Presumably, they can capture frequent word pattern in low-resource tasks. Finally, MTL+FT, MAML and SML achieve comparable performance with regard to the unigram/bigram diversity (Dist1/2) of generated responses.

Effect of Structure Modeling. To get more insight into how our proposed SML balances the knowledge generalization and customization, in Figure 3, we visualize the heatmap of self-attention weight $\mathbf{a}_k$ in Eqn. 12 for all 9 training sentence function representations. Each row in Figure 3 demonstrates the similarity between sentence function $\mathbf{s}_k$ and all sentence functions $[\mathbf{s}_1; ...; \mathbf{s}_K]$. Take the first row in Figure 3 for example, it tells us how all nine query-response sentence function pairs contribute to the representation of current query-response sentence function pair "*Positive Declarative (DE) | Positive Declarative (DE)*". We see that sentence functions containing *Interrogative(IN)* have nearly zero contribution while sentence functions containing *Declarative(DE)* in both query and response have a certain amount of contribution. Therefore, similar sentence functions trig-

Query	科四没考过是不是很挫？
	(Am I stupid for not passing the 4th part of the driving-test?)
Query SF	A-not-A Interrogative
Target Response SF	Negative Declarative
MTL	我也是
	(So do I)
MTL+FT	我还没补考
	(I have not taken a make-up driving-test)
MAML	还没考
	(I have not taken the driving-test yet)
SML	我也没过
	(I did not pass too.)
Query	今天领证了
	(I register for a marriage today)
Query SF	Positive Declarative
Target Response SF	Yes-no Interrogative
MTL	恭喜恭喜
	(Congratulations)
MTL+FT	恭喜你
	(Congratulations to you)
MAML	还没办婚礼吗？
	(Haven't you held the wedding ceremony?)
SML	办婚礼了吗？
	(Do you hold the wedding ceremony?)

Figure 4: Responses of all models. Words in red are related to the target sentence function. SF: Sentence Function.

ger similar initializations and dissimilar sentence functions trigger different ones.

Case Study. We present two examples in Figure 4, each of which shows a test query with the target sentence function and responses generated by all models. We see that responses generated by MTL are generic and can be used to reply to a large number of queries. With fine-tuning on responses of the target sentence function, MTL+FT can capture the correct response pattern in some cases. However, it is inferior to our proposed models MAML and SML, which can not only generate words related to the target sentence function but also keep the coherence and informativeness of responses.

7 Conclusion

In this paper, we propose a structured meta-learning algorithm for open domain dialogue generation on infrequent sentence functions. To tackle the low-resource issue, our proposed model, based on the recently proposed model-agnostic meta-learning, can find both transferable internal representations and sensible parameters which can produce large improvement under a few adaptation steps. Moreover, we further explore the structure across fine-grained sentence functions and such that the model can balance knowledge generalization and knowledge customization. Extensive experiments show that our structured meta-learning (SML) algorithm outperforms existing approaches under the low-resource setting.

References

Adrian Akmajian. 1984. Sentence types and the form-function fit. *Natural Language & Linguistic Theory*, 2(1):1–23.

Wei Bi, Jun Gao, Xiaojiang Liu, and Shuming Shi. 2019. Fine-grained sentence functions for short-text conversation. In *ACL*.

Jessica Ficler and Yoav Goldberg. 2017. Controlling linguistic style aspects in neural language generation. In *Proceedings of the Workshop on Stylistic Variation*, pages 94–104, Copenhagen, Denmark. Association for Computational Linguistics.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*.

Jiatao Gu, Yong Wang, Yun Chen, Victor O. K. Li, and Kyunghyun Cho. 2018. Meta-learning for low-resource neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3622–3631, Brussels, Belgium. Association for Computational Linguistics.

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

Pei Ke, Jian Guan, Minlie Huang, and Xiaoyan Zhu. 2018. Generating informative responses with controlled sentence function. In *ACL*.

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*, pages 110–119, San Diego, California. 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, Austin, Texas. Association for Computational Linguistics.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *EMNLP*.

Andrea Madotto, Zhaojiang Lin, Chien-Sheng Wu, and Pascale Fung. 2019. Personalizing dialogue agents via meta-learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5459, Florence, Italy. Association for Computational Linguistics.

Fei Mi, Minlie Huang, Jiyong Zhang, and Boi Faltings. 2019. Meta-learning for low-resource natural language generation in task-oriented dialogue systems. In *International Joint Conference on Artificial Intelligence*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Kun Qian and Zhou Yu. 2019. Domain adaptive dialog generation via meta learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2639–2649, Florence, Italy. Association for Computational Linguistics.

Laurie Rozakis. 2003. *The complete idiot's guide to grammar and style*. Penguin.

Yan Song, Shuming Shi, Jing Li, and Haisong Zhang. 2018. Directional skip-gram: Explicitly distinguishing left and right context for word embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 175–180, New Orleans, Louisiana. Association for Computational Linguistics.

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

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. In *NIPS*.

Risto Vuorio, Shao-Hua Sun, Hexiang Hu, and Joseph J Lim. 2019. Multimodal model-agnostic meta-learning via task-aware modulation. In *Advances in Neural Information Processing Systems 32*, pages 1–12. Curran Associates, Inc.

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

Can Xu, Wei Wu, Chongyang Tao, Huang Hu, Matt Schuerman, and Ying Wang. 2019. Neural response generation with meta-words. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5416–5426, Florence, Italy. Association for Computational Linguistics.

Huaxiu Yao, Ying Wei, Junzhou Huang, and Zhenhui Li. 2019. Hierarchically structured meta-learning. In *International Conference on Machine Learning*.

George Yule. 2016. *The study of language*. Cambridge university press.

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? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2204–2213, Melbourne, Australia. Association for Computational Linguistics.

Tiancheng Zhao, Ran Zhao, and Maxine Eskénazi. 2017. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *ACL*.

Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2017. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *AAAI Conference on Artificial Intelligence Conference on Artificial Intelligence*.