The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Conversational Model Adaptation via KL Divergence Regularization

Juncen Li,[1] Ping Luo,[2,3] Fen Lin,[1] Bo Chen[1]
[1]WeChat Search Application Department, Tencent, China. {juncenli}@tencent.com, {felicialin,jennychen}@tencent.com
[2]Key Lab of Intelligent Information Processing of Chinese Academy of Sciences (CAS),
Institute of Computing Technology, CAS, Beijing 100190, China. {luop}@ics.ict.ac.cn
[3]University of Chinese Academy of Sciences, Beijing 100049, China.

Abstract

In this study we formulate the problem of conversational model adaptation, where we aim to build a generative conversational model for a target domain based on a limited amount of dialogue data from this target domain and some existing dialogue models from related source domains. This model facilitates the fast building of a chatbot platform, where a new vertical chatbot with only a small number of conversation data can be supported by other related mature chatbots. Previous studies on model adaptation and transfer learning mostly focus on classification and recommendation problems, however, how these models work for conversation generation are still un-explored.

To this end, we leverage a KL divergence (KLD) regularization to adapt the existing conversational models. Specifically, it employs the KLD to measure the distance between source and target domain. Adding KLD as a regularization to the objective function allows the proposed method to utilize the information from source domains effectively. We also evaluate the performance of this adaptation model for the online chatbots in Wechat platform of public accounts using both the BLEU metric and human judgement. The experiments empirically show that the proposed method visibly improves these evaluation metrics.

Introduction

Recently, end-to-end neural systems have made great progress in various natural language tasks, such as machine translation (Cho et al. 2014b; Sutskever, Vinyals, and Le 2014), question answering (Yin et al. 2016), and dialog systems (Gu et al. 2016; Serban et al. 2016; Sordoni et al. 2015). In general, most of these systems consist of multi-layer RNN networks with a large number of parameters, thus, they need a large amount of text data for training. In previous studies for conversation modeling, plenty of dialogue data is usually collected from social platforms (such as Weibo (Wang et al. 2013)) or transformed from some public data (Banchs and Li 2012), which usually covers various vertical domains.

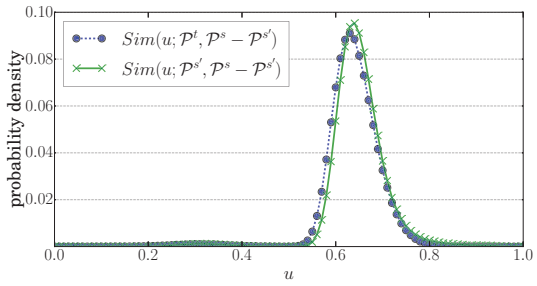
The background of this study is about the building of a chatbot platform, which contains various chatbots for diverse vertical domains, such as entertainment, sports, religion, etc. To build a vertical chatbot we usually need enough dialogue text from the corresponding vertical domain for end-to-end training. However, when a new chatbot is online for only a short time the data accumulated for training are very limited. Thus, we need a solution to address this cold start problem for new chatbots in a chatbot platform.

To address this data sparsity issue, we formulate the problem of conversational model adaptation. Specifically, for the building of a new chatbot, it leverages not only a limited amount of training data from this target domain, but also some existing conversational models. Here, the existing models we consider are trained by the data from open social platforms (Wang et al. 2013) (considered as the source domain), thus they may contain enough language patterns for the free-chat support.

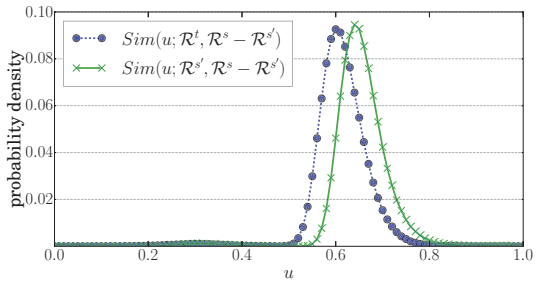
However, even though the source domain contains enough data for training a conversational model, its data distribution might be dissimilar from that in the target domain, as shown in our quantitative study on the data distributions of source and target domains (detailed later). Thus, adding the source domain data directly into the target domain may lead to a huge drift of data distribution, and result in a conversational model, which loses the domain-dependent characteristics for the target domain.

In this paper, we propose a KL divergence (KLD) regularization method for conversational model adaptation. Specifically, we first build a model based on the huge amount of dialog data from a social platform. Then, the small amount of target domain data is used to adapt the pre-trained model to the target domain via KLD regularization. This joint regularization framework prevents the dialog system from overfitting to the target domain, and simultaneously makes use of the information from the source domain. To evaluate the effectiveness of our method, we use both objective and subjective measures. Results of experiments show that our method visibly improves the performance of the existing models without model adaptation. In summation, our contributions are three-folds:

- To the best of our knowledge, we are the first to propose the problem of conversational model adaptation, where the building of a new vertical domain dialogue system is supported by both the small amount of target domain data and large number of source domain data.
- We develop a KLD regularization method to adapt conver-



(a) Input post comparison



(b) Response comparison

Figure 1: Comparison between the source and target domains

sational models, and the careful evaluation demonstrates its effectiveness.

- We also stress that the proposed adaptation framework is agnostic to the methods for conversational modeling. In other word, it can easily accommodate any end-to-end dialogue systems, such as memory networks (Sukhbaatar et al. 2015), neural encoder-decoder model (Cho et al. 2014b) and so on.

Quantitative Study of Similarity between Source and Target Domains

In this problem we are given two corpora, source domain corpus $\mathcal{D}^s$ and target domain corpus $\mathcal{D}^t$. Each corpus is the set of *post-response* pairs. Namely, $\mathcal{D} = \{(x, y)|y \text{ is the response of post } x\}$. Specifically, $\mathcal{D}^s$ is collected from Tencent Weibo, including 1,903,512 pairs, and $\mathcal{D}^t$ is given by a third-party company, including 15,315 pairs.

To study the similarity between the data distributions of $\mathcal{D}^s$ and $\mathcal{D}^t$, we first define the following measure of maximum cosine similarity between a sentence s and a corpus $\mathcal{D}$, as follows:

$$MCS(s, \mathcal{D}) = \max_{s' \in \mathcal{D}}(\phi(s, s')) \quad (1)$$

where ϕ calculates the similarity between the two sentences. To compute this similarity, we represent each sentence as a paragraph vector (Le and Mikolov 2014), and apply the cosine similarity on these two vectors. s is closer to the corpus $\mathcal{D}$ when the value of maximum cosine similarity is higher.

We further define the similarity between two corpora, $\mathcal{S}^1$ and $\mathcal{S}^2$ (the sets of sentences), as a distribution of u:

$$Sim(u; \mathcal{S}^1, \mathcal{S}^2) = \frac{1}{|\mathcal{S}^1|} \sum_{s_i \in \mathcal{S}^1} \delta(u = MCS(s_i, \mathcal{S}^2)) \quad (2)$$

where u is a user-specified value in $[-1, 1]$, δ is the Kronecker delta function. In other words, the similarity between $\mathcal{S}^1$ and $\mathcal{S}^2$ can be represented by the empirical distribution of $Sim(u; \mathcal{S}^1, \mathcal{S}^2)$.

With the definition of $Sim(u; \mathcal{S}^1, \mathcal{S}^2)$, we design the following process to show the similarity between data distributions of $\mathcal{D}^s$ and $\mathcal{D}^t$. Specifically, we consider the post sentences and response sentences, respectively. Namely, $\mathcal{P}^s$ and $\mathcal{P}^t$ include all the input posts in source and target domain, respectively. Next, we randomly sample a subset $\mathcal{P}^{s'}$ of $\mathcal{P}^s$, such that $|\mathcal{P}^{s'}| = |\mathcal{P}^t|$. Then, we calculate the following two distributions,

$$\begin{aligned} &Sim(u; \mathcal{P}^{s'}, \mathcal{P}^s - \mathcal{P}^{s'}) \\ &Sim(u; \mathcal{P}^t, \mathcal{P}^s - \mathcal{P}^{s'}) \end{aligned} \quad (3)$$

Here, $Sim(u; \mathcal{P}^{s'}, \mathcal{P}^s - \mathcal{P}^{s'})$ actually calculates the similarities between the posts in the sampled source domain $\mathcal{P}^{s'}$ and the remaining data set $\mathcal{P}^s - \mathcal{P}^{s'}$. $Sim(u; \mathcal{P}^t, \mathcal{P}^s - \mathcal{P}^{s'})$ measures the similarities between the posts in the target domain $\mathcal{P}^t$ and the source domain $\mathcal{P}^s - \mathcal{P}^{s'}$. Finally, we can compare these two distributions to compare the data distributions in $\mathcal{P}^s$ and $\mathcal{P}^t$. As shown in Fig. 1(a) these two distributions are very similar.

Similarly, we can apply the same process to the response sentences in $\mathcal{D}^s$ and $\mathcal{D}^t$. Then, we calculate the following two distributions,

$$\begin{aligned} &Sim(u; \mathcal{R}^{s'}, \mathcal{R}^s - \mathcal{R}^{s'}) \\ &Sim(u; \mathcal{R}^t, \mathcal{R}^s - \mathcal{R}^{s'}) \end{aligned} \quad (4)$$

where the set $\mathcal{R}$ includes all the response sentences in the corresponding domain. As shown in Fig. 1(b), these two distributions are quite different. Thus, responses in the target domain are dissimilar to responses in the source domain.

To conclude, we observe that the input posts are similar between the source and target domain. In other words, chatbot users may ask similar questions to the two chatbots from related vertical domains. However, their responses might be different based on their individual expertise. Therefore, the dissimilarity between target domain and source domain is mainly reflected on their responses.

Background and Preliminaries

While this paper mainly introduces the adaptation method applied to RNN-based encoder-decoder for simplicity, our adaptation method is suitable for most conversational models.

RNN-based Encoder-Decoder

RNN-based encoder-decoder can be expressed as a model maximizing the likelihood of the output sequence given

an input sequence. Supposed that we have a corpus $\mathcal{D} = \{(x^i, y^i)|y^i \text{ is the response of post } x^i\}$, where x^i and y^i are two sequences of tokens, a RNN-based encoder-decoder is typically trained to maximize the likelihood:

$$L = \frac{1}{N}\sum_{i=1}^{N} \log p(y^i|x^i) \tag{5}$$

where N is the number of samples in the corpus. A RNN-based encoder-decoder mainly includes two parts: encoder and decoder which are implemented as RNNs. The encoder converts input sequence $x^i = (x_1^i, \ldots, x_{T_i}^i)$ to a fixed length context vector c^i, i.e.

$$h_t^i = f(x_t^i, h_{t-1}^i), c^i = \psi((h_1^i, \ldots, h_{T_i}^i)) \tag{6}$$

where T_i is the length of input sequence x_i, h_t^i is the hidden state at time t of encoder sequence, f is a non-linear function and ψ summarizes the hidden states.

The context vector c^i is utilized by the decoder to generate the output sequence $y^i = (y_1^i, \ldots, y_{T_i'}^i)$. There are different methods to unfold c^i. (Sutskever, Vinyals, and Le 2014) used c^i as the initial hidden state s_0^i of the decoder and the function to calculate hidden states of the decoder is:

$$s_t^i = f(y_{t-1}^i, s_{t-1}^i) \tag{7}$$

where s_t^i is the hidden state at time t of decoder sequence. While (Cho et al. 2014b) argued that adding c^i to the input of every step helps decoder RNN make use of context information and improve performance:

$$s_t^i = f(y_{t-1}^i, s_{t-1}^i, c^i) \tag{8}$$

With the hidden state s_t^i, the target symbol y_t^i at time t can be predicted by:

$$p(y_t^i|y_{<t}^i, x^i) = g(y_{t-1}^i, s_t^i, c^i) \tag{9}$$

where $y_{<t}^i$ represents the history $\{y_0^i, \ldots, y_{t-1}^i\}$, g is a non-linear function.

In the rest of this paper, RNN refers to the RNN family including RNN with different structures. LSTM (Hochreiter and Schmidhuber 1997) and GRU (Cho et al. 2014a) are two examples and often perform better than vanilla ones.

KL Divergence Regularized Adaptation Method

Our goal is to train a satisfactory target domain conversational model using target domain data with the assistance of source domain data. Combining the target domain data and the source domain data directly is the most straight-forward approach. However, target domain data will be overwhelmed by the source domain data. Training directly with the combined data may lead to a conversation model failing to respond to target domain posts or responding to posts in the style of the source domain instead of the target domain. Another approach is to pre-train the system with the source domain data and fine-tune using the target domain data. Although this approach makes use of the target domain data, it may destroy the information extracted from the source domain and overfit to the target domain.

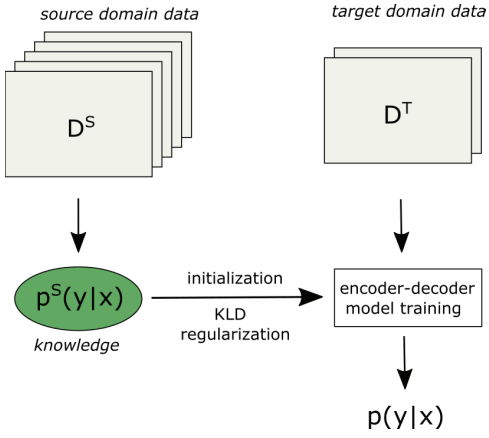


Figure 2: Illustration of KL divergence regularized adaptation method.

To address these problems, We propose a KL divergence regularized adaptation method as depicted in Fig. 2. Our method pre-trains the conversation model with source domain data and adapts the model to the target domain with KLD regularization. Using KLD to measure the deviation of distributions $p(y|x)$ estimated from the source and target domain data, we can limit the distance between them to prevent the target domain distribution from deviating too far from the source domain distribution. This method helps conversational models make use of source domain information and avoid overfitting effectively. To materialize the above proposal, we need to maximize the likelihood of outputs in the target domain and minimize the KLD between two distributions simultaneously. Adding the divergence as a regularization term, the objective function of our proposed model is defined as follows:

$$L = (1-\alpha)\frac{1}{N}\sum_{i=1}^{N} \log p(y^i|x^i) - \alpha D_{\mathrm{KL}}(p^S||p) \tag{10}$$

where p^S is the distribution of the source domain data, α is regularization weight, D_{KL} is the KLD function:

$$\begin{aligned} D_{\mathrm{KL}}(p^S||p) =& \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} p^S(y^j|x^i)\log p^S(y^j|x^i) \\ &- \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} p^S(y^j|x^i)\log p(y^j|x^i) \end{aligned} \tag{11}$$

where M is the number of output samples. Since the first term of Eq. 11 is a constant and unrelated to the model parameters, we can remove this term and get the regularized optimization criterion:

$$\begin{aligned} L =& (1-\alpha)\frac{1}{N}\sum_{i=1}^{N} \log p(y^i|x^i) \\ &+ \alpha\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} p^S(y^j|x^i)\log p(y^j|x^i) \end{aligned} \tag{12}$$

The framework of our method is shown in Algorithm 1.

Algorithm 1 KLD REGULARIZED ADAPTATION$(\mathcal{D}^s, \mathcal{D}^t)$

Require:
Source domain data, $\mathcal{D}^s$
Target domain data, $\mathcal{D}^t$
Ensure: The parameters of our model, Θ
1: Initialize Θ randomly;
2: Pre-train Θ with $\mathcal{D}^s$ by maximizing Eq. 5
3: Calculate the conditional probability $p^S(y|x)$ of all pairs in $\mathcal{D}^t$ using the pre-trained model.
4: Fine-tune Θ with $\mathcal{D}^t$ by maximizing Eq. 12
5: **return** Θ;

p^S is analogous to the dark knowledge distilled from the source domain (Hinton, Vinyals, and Dean 2015). The dark knowledge contains information from source domain. Thus, we can use it as the soft target. With this supervision, our model can absorb the knowledge and imitate the distribution of source domain. The degree of imitation can be controlled by regularization weight.

The regularization weight α ranges from 0 to 1. When $\alpha = 0$, the source domain data has no influence on the method and only the target domain data is used. On the contrary, when $\alpha = 1$, our adaptation method completely trusts the source domain distribution and neglects the target domain information. The regularization weight α is related to the size of the target domain data. With a larger target domain data size, we can use a smaller value of α.

Experiments

We use RNN-based encoder-decoder with GRU (Cho et al. 2014b) as the basic conversational model and compare our proposed method with three benchmarks described in the following subsections. For convenience, we use "adaptation method" to refer to the KL divergence regularized adaptation method.

Dataset

The dataset includes two parts, source domain data and target domain data. We collect source domain data from Tencent Weibo using a similar method described in (Wang et al. 2013). The source domain data includes 1,903,512 pairs and 26,814 words. For the target domain data, we get 15,315 pairs related to Buddhism from a third-party company. We randomly divide this data into training, validation and test set with no overlap posts. In other words, we ensure that there are no posts appearing in two different sets. The target domain data has 4,130 words. Table 1 shows the details of our data.

Benchmarks

We use three methods adopted from the model proposed by (Cho et al. 2014b) as our benchmarks. In the following sections, we use the acronym in the brackets to refer them.

- RNN-based encoder-decoder model trained with target domain data (t-RED).

Table 1: Data statistics

Source domain	posts	556,639
Source domain	responses	557,169
Source domain	pairs	803,716
Training	posts	8,921
Training	responses	9,467
Training	pairs	12,322
Validation	posts	1,000
Validation	responses	1,286
Validation	pairs	1,449
Test	posts	1000

- RNN-based encoder-decoder model trained with combined data consisting of source domain data and target domain data (c-RED).
- Pre-training RNN-based encoder-decoder model with source domain data and fine-tuning the model with target domain data (ft-RED).

Experimental Details

We merge dictionaries of two domains and get a combined dictionary consisting of 27,037 words. We randomly select 300 pairs from the target domain as the test set and the rest are used for training.

For the RNN-based encoder-decoder model, we use 1-layer GRU with 512 cells for both the encoder and the decoder. Word embeddings are treated separately for the encoder and the decoder as suggested in (Shang, Lu, and Li 2015). Embedding dimensions are set to 128. All parameters are initialized with the uniform distribution between -0.1 and 0.1. The activation function we use is maxout which can effectively avoid overfitting (Goodfellow et al. 2013). We use Adadelta (Zeiler 2012) in training and a minibatch size of 128. All RNN-based encoder-decoder models referred in the experiments use the above settings.

For the adaptation method , we set regularization weight α to 0.5. According to Eq. 12, we need to traverse all the responses for every post which greatly prolongs the training process. In practice, we focus on the response corresponding to the post in our training set. Thus, we can sample responses by posts and simplify our objective function as:

$$\begin{aligned} L =& (1-\alpha)\frac{1}{N}\sum_{i=1}^{N}\log p(y^i|x^i) \\ &+ \alpha\frac{1}{N}\sum_{i=1}^{N} p^S(y^i|x^i)\log p(y^i|x^i) \end{aligned} \tag{13}$$

Results

We use two evaluation methods to compare the performances of our method against those of the benchmarks: BLEU score (Papineni et al. 2002) and manual evaluation method (Shang, Lu, and Li 2015). They are commonly employed metrics to evaluate performances of conversational models.

Table 2: BLEU Score

Methods	BLEU Score
t-RED	4.36
c-RED	1.35
ft-RED	7.71
Adaptation Method	**10.53**

Table 3: Manual evaluation examples

No	Posts	Responses	Score
1	你快去问你师父呀 You should go to ask your teacher quickly	不告诉你。 Do not tell you.	0
2	那什么时候会累 When do you feel tired	为人民服务该。 Serve the people should.	0
3	答不上来了吧？ Can't you answer it?	我答不上 I cannot answer it.	1
4	你父母呢？ Where are your parents?	我父母不在家 My parents are not at home.	1
5	你父母呢？ Where are your parents?	机器人怎么会有父母。 Robots do not have parents.	2

BLEU Metric

Table 2 shows BLEU scores of the proposed method and three benchmarks. We find that our proposed adaptation method is superior to all three benchmarks. Compared to t-RED, ft-RED not only makes use of the target domain data, but also utilizes some information from the source domain. Thus, it performs better. However, the ft-RED overfits to the target domain data, so it has lower BLEU score than our adaptation method.

Additionally, We discover that c-RED performs worse than t-RED on BLEU score. Because the size of source domain data is much larger than that of the target domain, the target domain data is overwhelmed by the source domain data when combining. In addition, reference responses used to calculate BLEU scores are all from the target domain, which are dissimilar to source domain responses. Thus, the c-RED has a lower BLEU score than the t-RED.

Manual Evaluation

Referring to (Shang, Lu, and Li 2015), we use manual evaluation method to evaluate the models. To prevent human annotation bias, we mix generated results of all models up and let three judges score the same result set. The score ranges from 0 to 2 indicating bad, normal and good respectively.

- Bad(0): The generated response is not related to the post or there are some grammatical mistakes in the response.
- Normal(1): The generated response has no grammatical mistakes and is suitable for the post in some scenarios. But it may be not the perfect response or may have some minor deviations from the target domain.
- Good(2): The generated response is free of mistakes and in line with the target domain data. Additionally, it is a very satisfying response to the post.

Table 3 shows the manual evaluation examples. The first example is scored 0 because the response is unrelated to the post. Because of a grammatical mistake in the second example, it is scored 0. The third and fourth examples are both scored 1 for different reasons. The third example's response suits to the post, but it is too general to be a perfect response. The response of the fourth one has some minor deviations from target domain, because in target domain the responder is a robot and has no parents. Thus, it is better to respond "Robots do not have parents" as given in the fifth example.

The manual evaluation results and several examples of responses from different models are shown in Table 4 and Table 5 respectively. The agreement (Fleiss and others 1971) is a statistical measure of inter-rater consistency. In Table 4, agreements of all the models range from 0.2 to 0.4. It indicates that our manual evaluation is "Fair agreement". From Table 4, we discover that our method is superior to the benchmarks. Our conjectured explanation of the improved performance is as follows, clarified by the examples in Table 5.

Firstly, source domain data assists our proposed method in understanding the posts. From Table 5, we can find that the t-RED generates high-frequency and unrelated responses to the posts in the first two examples. This case indicates that these two posts are not well understood by this method. On the contrary, c-RED and our adaptation method generate related and suitable responses which means that the posts can be better understood with the help of source domain data. Though the ft-RED makes use of the source domain data, it overfits to the target domain data and deteriorates information extracted from source domain. Therefore, the ft-RED is not able to respond to posts in the first two examples, but responds properly to the post in the third example.

Secondly, our proposed method makes use of the target domain information more effectively. From Table 5 we can find that most responses generated by c-RED do not match the style of responses in the target domain. Instead, they lean toward source domain high frequency responses in the forms of "It is not XXX at all" and "It is rather you who is XXX". On the contrary, our adaptation method generates responses more aligned with target domain data. This observation shows that when the size of target domain is far smaller than that of the source domain, combining the data of two domains directly leads to source domain data overwhelming target domain data. On the other hand, our proposed method is a better way to use the source domain data to assist the training of conversational models.

There is an inconsistency between BLEU metric and manual evaluation results on c-RED and t-RED. The main reason is that most suitable responses generated by c-RED are not in target domain and have little overlapping n-grams with the reference responses. The fourth example in Table 5 is a typical case. "Pursue him/her if you like him/her." is a suitable response to "Xian Er, what should I do when I fall in love with someone?". But there is no overlapping n-grams with reference responses in the target domain. The t-RED's number of suitable responses is lower, but there are more overlapping n-grams with reference responses. Thus, although the c-RED performs better than t-RED according to manual evaluation, its BLEU score is lower than t-RED.

Table 4: Manual evaluation results

Methods	Mean Score	Good(2)	Normal(1)	Bad(0)	Agreement
t-RED	0.318	3.12%	25.5%	71.3%	0.281
c-RED	0.429	5.9%	31.1%	63.0%	0.295
ft-RED	0.484	6.9%	34.6%	58.5%	0.298
Adaptation Method	0.530	7.7%	37.6%	54.7%	0.289

Table 5: Responses examples

No	Post	t-RED	c-RED	ft-RED	Adaptation Method
1	好奇怪 It is so strange.	阿弥陀佛 Amitabha	一点都不奇怪 It is not strange at all.	阿弥陀佛 Amitabha	有什么奇怪的? What is strange about it?
2	牛 Awesome	阿弥陀佛 Amitabha	你才牛呢 It is rather you who are awesome	阿弥陀佛 Amitabha	谢谢鼓励。 Thank you for your encouragement.
3	什么是串习 What is habitual tendency	学习。 Study.	你才串呢 It is rather you who are string	串习就是不断的重复。 Habitual tendency is continuously repeating.	串习就是不断的重复。 Habitual tendency is continuously repeating.
4	贤儿我喜欢别人怎么办 Xian Er, what should I do when I fall in love with someone?	随缘。 Leave it to destiny.	喜欢就去追。 Pursue him/her if you like him/her.	随缘。 Leave it to destiny.	随缘。 Leave it to destiny.
5	你父母呢? Where are your parents?	阿弥陀佛 Amitabha	我父母在家呢 My parents are at home	我是机器人 I'm a robot.	机器人怎么会有父母。 Robots do not have parents.

Related Work

Our work is mainly related to conversational models and adaptation and transfer learning methods. Because the basic frameworks of conversational models come from statistical machine translation, we include it in this section.

Statistical Machine Translation

RNN-based encoder-decoder is a basic model for end-to-end systems. It was first proposed in statistical machine translation (SMT). Sutskever et al. (Sutskever, Vinyals, and Le 2014) used a four-layer LSTM to convert the input sequence into a fixed length context vector and another multilayered LSTM to decode the target sequence. In this work, the context vector was only inputted into the first step of the decoder. Cho et al. (Cho et al. 2014b) proposed another model with GRU and feed the fixed length context vector into all decoding steps. Bahdanau et al. (Bahdanau, Cho, and Bengio 2015) argues that the fixed-length context vector can be the performance bottleneck of RNN-based encoder-decoder models. They proposed an attention mechanism to obtain different context vectors for different steps of the decoder and achieved better results.

Conversational Models

Inspired by the SMT methods, researchers put forward several improved models for conversational models. Shang et al. (Shang, Lu, and Li 2015) introduced three types of encoding schemes as extensions of attention method. They found that hybrid scheme performs better than the other two schemes in generating responses. Instead of focusing on one-round dialog, Sordoni et al. (Sordoni et al. 2015) built Dynamic-Context Generative Model(DCGM) considering contexts of dialogs. Serban et al. (Serban et al. 2016) devised a hierarchical neural network to encode a sequence of words into an utterance vector while keeping track of the utterance vector to utilize context information. Serban et al. (Serban et al. 2017) extended the hierarchical neural network by adding a parallel RNN encoder, which encodes the high-level coarse tokens, into the previous framework. Another method to improve conversational models is to use active learning to solve data sparsity issue (Asghar et al. 2017).

Adaptation and Transfer Learning Methods

Adaptation methods are widely used in speech recognition. (Abrash et al. 1995; Neto et al. 1995; Albesano, Gemello, and Mana 2000) linearly transformed input features to do the adaptation. Yao et al. (Yao et al. 2012) performed linear transformation on the softmax layer. Conservative training (Abrash et al. 1995) is another category of adaptation. It adds a regularization to the objective function. L2 (Li 2007) and KLD (Yu et al. 2013) are two possible regularization terms. Additionally, adaptation is a special case of transfer learning methods which are extensively surveyed in (Pan and Yang 2010).

Conclusion

We come up with a problem of building conversational model for a target domain with scarce training data assisted by some existing conversational models from source domain. Then we propose a KL divergence regularized adaptation method which pre-trains the system with the source domain data and adapts the system to the target domain with KL divergence regularization. Our method takes advantage of information in both domains and prevents the system from overfitting to the target domain. Experiment results show superior performance of the proposed method compared to existing conversational models. Additionally, our method is model agnostic and can be widely used.

For future work, we can incorporate transfer learning methods. In this paper, all parameters are fine-tuned in the

adaptation process. It is possible to fix some parameters of the system and fine-tune other parameters when adapting. How to incorporate transfer learning in the training of dialog systems is an interesting problem for further studies.

Acknowledgments

Ping Luo is Supported by the National Natural Science Foundation of China (No. 61473274). We thank Kaisheng Yao, Qiang Yang, Cheng Niu, Wilson Tam, Ganbin Zhou, Yijun Xiao for their constructive advices. We also thank the anonymous AAAI reviewers for their helpful feedback.

References

Abrash, V.; Franco, H.; Sankar, A.; and Cohen, M. 1995. Connectionist speaker normalization and adaptation. In *EUROSPEECH*. ISCA.

Albesano, D.; Gemello, R.; and Mana, F. 2000. Hybrid hmm–nn modeling of stationary–transitional units for continuous speech recognition. *Information Sciences* 123(1):3–11.

Asghar, N.; Poupart, P.; Jiang, X.; and Li, H. 2017. Deep active learning for dialogue generation. *In Proceedings of the 6th Joint Conference on Lexical and Computational Semantics*.

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

Banchs, R. E., and Li, H. 2012. Iris: a chat-oriented dialogue system based on the vector space model. In *ACL (System Demonstrations)*, 37–42. The Association for Computer Linguistics.

Cho, K.; van Merrienboer, B.; Bahdanau, D.; and Bengio, Y. 2014a. On the properties of neural machine translation: Encoder-decoder approaches. *ACL*.

Cho, K.; van Merrienboer, B.; Gülçehre, Ç.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014b. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *EMNLP*.

Fleiss, J., et al. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin*.

Goodfellow, I. J.; Warde-farley, D.; Mirza, M.; Courville, A.; and Bengio, Y. 2013. Maxout networks. In *ICML*.

Gu, J.; Lu, Z.; Li, H.; and Li, V. O. K. 2016. Incorporating copying mechanism in sequence-to-sequence learning. *ACL*.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural Comput.*

Le, Q. V., and Mikolov, T. 2014. Distributed representations of sentences and documents. *PMLR*.

Li, X. 2007. *Regularized Adaptation: Theory, Algorithms and Applications*. Ph.D. Dissertation, Seattle, WA, USA. AAI3265367.

Neto, J. P.; Almeida, L. B.; Hochberg, M.; Martins, C.; Nunes, L.; Renals, S.; and Robinson, T. 1995. Speaker-adaptation for hybrid hmm-ann continuous speech recognition system. In *EUROSPEECH*. ISCA.

Pan, S. J., and Yang, Q. 2010. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* 22(10):1345–1359.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *Computational Linguistics (ACL), Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), Philadelphia*, 311–318.

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

Serban, I. V.; Klinger, T.; Tesauro, G.; Talamadupula, K.; Zhou, B.; Bengio, Y.; and Courville, A. C. 2017. Multiresolution recurrent neural networks: An application to dialogue response generation. *AAAI*.

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

Sordoni, A.; Galley, M.; Auli, M.; Brockett, C.; Ji, Y.; Mitchell, M.; Nie, J.; Gao, J.; and Dolan, B. 2015. A neural network approach to context-sensitive generation of conversational responses. *ACL*.

Sukhbaatar, S.; Szlam, A.; Weston, J.; and Fergus, R. 2015. Weakly supervised memory networks. *CoRR*.

Sutskever, I.; Vinyals, O.; and Le, Q. V. 2014. Sequence to sequence learning with neural networks. *NIPS*.

Wang, H.; Lu, Z.; Li, H.; and Chen, E. 2013. A dataset for research on short-text conversations. In *EMNLP*.

Yao, K.; Yu, D.; Seide, F.; Su, H.; Deng, L.; and Gong, Y. 2012. Adaptation of context-dependent deep neural networks for automatic speech recognition. In *SLT*, 366–369. IEEE.

Yin, J.; Jiang, X.; Lu, Z.; Shang, L.; Li, H.; and Li, X. 2016. Neural generative question answering. *IJCAI*.

Yu, D.; Yao, K.; Su, H.; Li, G.; and Seide, F. 2013. Kl-divergence regularized deep neural network adaptation for improved large vocabulary speech recognition. In *ICASSP*, 7893–7897. IEEE.

Zeiler, M. D. 2012. ADADELTA: an adaptive learning rate method. *CoRR*.