Zero-Shot Deployment for Cross-Lingual Dialogue System

Lu Xiang[1,2], Yang Zhao[1,2], Junnan Zhu[1,2], Yu Zhou[1,2,3(✉)], and Chengqing Zong[1,2]

[1] National Laboratory of Pattern Recognition, Institute of Automation, CAS, Beijing, China
{lu.xiang,yang.zhao,junnan.zhu,cqzong,yzhou}@nlpr.ia.ac.cn
[2] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[3] Fanyu AI Laboratory, Zhongke Fanyu Technology Co., Ltd., Beijing, China

Abstract. The dialogue system is widely used in many application scenarios, while the construction of the dialogue system always faces the difficulty of zero-resource training data. To alleviate that, we propose a knowledge transfer framework to build a dialogue system based on existing machine translators and training data in data-rich language. Specifically, we first generate various kinds of pseudo data with cyclic translation procedure and different data combinations. Then we propose a noise injection method and a multi-task training method for the pipeline system and end-to-end system, respectively. The noise injection method optimizes each module by incorporating machine translation noises into the pipeline process to handle the error propagation problem, thus improving the whole system's robustness. The multi-task training method combines cross-lingual dialogue, monolingual dialogue, and machine translation into the end-to-end dialogue system's training process, thus reducing the impact of noises in pseudo data. The extensive experiments on a real-world e-commerce dataset demonstrate that our methods can achieve remarkable improvements over strong baselines.

Keywords: Cross-lingual dialogue system · Noise injection · Multi-task

1 Introduction

Dialogue systems have stimulated great interest from both academia and industry [16,22,25]. However, most existing dialogue systems are developed based on monolingual training data, making the dialogue service only available in the corresponding language. Along with globalization, there is an increasing need for commercial dialogue systems to handle different languages. However, collecting high-quality dialogue data for a new language is quite expensive, leading to the development of a dialogue system face the challenge of few-shot or even

© Springer Nature Switzerland AG 2021
L. Wang et al. (Eds.): NLPCC 2021, LNAI 13029, pp. 193–205, 2021.
https://doi.org/10.1007/978-3-030-88483-3_15

zero-resource training data. Therefore, in this paper, we focus on building a cross-lingual dialogue system based on the existing monolingual dialogue system.

Despite the attractive progress in cross-lingual dialogue systems [3,9,14,15], researchers mainly focus on the sub-modules in a dialogue system. To the best of our knowledge, there is none work trying to build a complete cross-lingual dialogue system under the zero-shot setting, which is the focus of this paper.

Benefiting from the excellent performance of machine translation (MT), we adopt MT systems as the language bridge, and two basic methods can be adopted to deploy a dialogue system for the zero-resource language:

i) **MT-based pipeline dialogue system.** It consists of three steps: translation step, dialogue step, and back-translation step. A machine translator translates a user's utterance into a language consistent with the dialogue system. Then the dialogue system generates a response based on the translated utterance. Finally, the machine translator translates the response back into the user's language. This method is easy to implement. However, this method's core challenge is the amplification of translation errors.
ii) **End-to-end dialogue system.** The machine translator is used to translate the dialogue training data in data-rich language into zero-resource language. Then an end-to-end dialogue system can be directly trained from the translated data. However, there are still many noises and errors in the translated data, which will seriously affect the dialogue system's performance.

In this paper, we propose two possible workarounds to deploy a dialogue system for the zero-resource language without any dialogue data in that language under the guidance of the MT systems and the dialogue dataset in data-rich language. Specifically, we first generate various pseudo data that contain the dialogue knowledge of the data-rich language and translation knowledge between data-rich language and zero-resource language through cyclic translation procedure. Based on generated pseudo data, we propose two methods to enhance the performance of the MT-based pipeline system and end-to-end system, respectively. For the MT-based pipeline system, a **noise injection** method is proposed to optimize each module in the pipeline paradigm. This method injects noises into both the MT systems and the dialogue system with generated pseudo data, making the MT systems more relevant to the dialogue and the dialogue system more robust to the noise input. For the end-to-end model, a **multi-task training** method is designed to augment the performance by combining the training process of three tasks: cross-lingual dialogue system, monolingual dialogue system, and machine translation task. This kind of synchronous learning can optimize the encoder and reduce the impact of noises in the pseudo data. The main contributions of this paper are as follows:

(1) To the best of our knowledge, we are the first to make a full investigation about how to deploy a dialogue system to a zero-resource language, which only uses the dialogue data in data-rich language and machine translators.
(2) Noise injection method and multi-task training method are proposed to boost the performance of the pipeline model and end-to-end model, respectively.

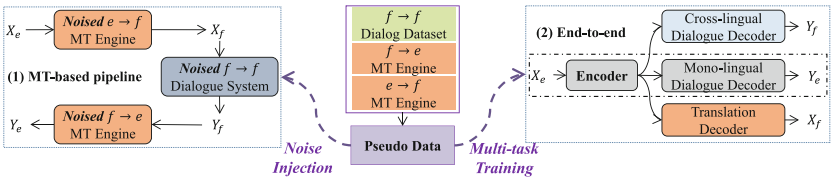

Fig. 1. The framework of our proposed methods.

(3) The proposed methods have been evaluated on the transfer of three language pairs. The results have shown the effectiveness of our methods.

2 Problem Definition and Background

Our goal is to construct a dialogue system for the new language e, which takes the contextual utterance X_e as input, and generates response Y_e. Due to the zero-resource training data, we make full use of the following two resources to transfer the dialogue system from data-rich language f to language e:

1) **Dialogue dataset in f**: $\mathcal{D}_{X_f \Rightarrow Y_f} = \{(X_f, Y_f)\}$, where X_f denotes the dialogue context and Y_f denotes the response.
2) **MT engines**, which can translate sentence from e to f (denoted as $MT_{e \Rightarrow f}$) and back-translate from f to e (denoted as $MT_{f \Rightarrow e}$). Hence, the dialogue system for e is formalized as follows:

$$Y_e = g(X_e | \mathcal{D}_{X_f \Rightarrow Y_f}, MT_{e \Rightarrow f}, MT_{f \Rightarrow e}) \tag{1}$$

We briefly describe the conversational model used in this paper. Considering the excellent text generation performance of the Transformer encoder-decoder network [21], we implement our neural conversational model entirely based on this framework. Given a dialogue data set $\mathcal{D} = \{(X, Y)\}$, where Y is a response of a dialogue context X. The encoder and decoder are trained jointly to maximize the conditional probability of response sequence given an input sequence:

$$L(\mathcal{D}; \theta) = \sum_{(X,Y) \in \mathcal{D}} \log p(Y | X; \theta) \tag{2}$$

3 Approach

To deploy the dialogue system for e, we first use the MT engines and dialogue dataset in f to construct various pseudo data. Then, we put forward a noise injection method for the pipeline system to alleviate the error propagation problem and a multi-task training method for the end-to-end model to reduce the influence of errors and noises in the pseudo data, as illustrated in Fig. 1.

3.1 Pseudo Data Construction

Given an input-response pair (X_f, Y_f), we first translate (X_f, Y_f) into (X_e, Y_e), and then translate back into language f, denoted as $(X_f^{'}, Y_f^{'})$. Thus, through the cyclic translation procedure we construct the following four pseudo datasets.

1) $\mathcal{D}_{X_e \Rightarrow Y_e} = \{(X_e, Y_e)\}$. It is a pseudo monolingual dialog dataset in language e consisting of the input-response pair (X_e, Y_e).
2) $\mathcal{D}_{X_e \Rightarrow Y_f} = \{(X_e, Y_f)\}$. It is a pseudo cross-lingual dialog dataset consisting of input-response pair (X_e, Y_f).
3) $\mathcal{D}_{X_f^{'} \Rightarrow Y_f^{'}} = \{(X_f^{'}, Y_f^{'})\}$. It is a pseudo monolingual dialog dataset in language f and contains input-response pair $(X_f^{'}, Y_f^{'})$.
4) $\mathcal{D}_{f \Rightarrow e} = \{(X_f, X_e) \cup (Y_f, Y_e)\}$. It is a pseudo parallel corpus consisting of each message including input and response and its translated message.

These four pseudo datasets contain dialogue knowledge from data-rich language and translation knowledge from MT engines. Then we use the datasets to optimize the MT-based pipeline system and the end-to-end system.

3.2 Noise Injection Method

For the MT-based pipeline system, the domain of the online MT engine is different from the dialogue scenario, which will introduce many errors and noises. Besides, the original dialogue system is trained on the clean dataset, making it impossible to work properly when given the translated utterances. The constructed pseudo datasets contain much knowledge from both the MT engine and dialogue. Therefore, we consider using the pseudo datasets to optimize each module in the pipeline system, as shown in Fig. 1. Since the modules are learning from noise data, they can better handle noise input. We name this method the **noise injection** method. It can be divided into two steps:

Noised NMT System. We use the generated pseudo dataset $\mathcal{D}_{f \Rightarrow e}$ to train Transformer-based neural machine translation (NMT) systems from both directions ($f \Rightarrow e$ and $e \Rightarrow f$). These two NMT systems are denoted as *Noised NMT Systems*. The two systems are more relevant to the dialogue task than the online MT engine since the pseudo parallel dataset is constructed from the original clean dialogue dataset.

The Transformer-based NMT also consists of an encoder and decoder. Given a parallel dataset $\mathcal{P} = \{(F, E)\}$, the loss function can be calculated as:

$$L(\mathcal{P}; \theta) = \sum_{(F,E) \in \mathcal{P}} \log p(E|F; \theta) \tag{3}$$

Hence, given the pseudo parallel dataset $\mathcal{D}_{f \Rightarrow e}$, the noised NMT systems can be trained by optimizing the loss function in Eq. 3.

Noised Dialogue System. To make the dialogue system better handle the noise input, we need to update the original dialogue system and let it experience

more noise data. To achieve this, we merge the dataset $\mathcal{D}_{X_f' \Rightarrow Y_f'}$ with the original clean dataset $\mathcal{D}_{X_f \Rightarrow Y_f}$ and retrain the dialogue system. This system is denoted as *Noised Dialogue System*. Given the pseudo data $\mathcal{C} = \{(X', Y')\}$ and the original clean training data $\mathcal{D} = \{(X, Y)\}$, the loss function is calculated as follows:

$$L(\mathcal{D}, \mathcal{C}; \theta) = \sum_{n=1}^{|\mathcal{D}|+|\mathcal{C}|} \left\{ \underbrace{\log p(Y_{\mathcal{D}}^n | X_{\mathcal{D}}^n; \theta)}_{\text{Lossfromcleandata}} + \underbrace{\log p(Y_{\mathcal{C}}^n | X_{\mathcal{C}}^n; \theta)}_{\text{Lossfrompseudodata}} \right\} \tag{4}$$

3.3 Multi-task Training and Adaptation

In the noise injection method, the NMT systems and the dialogue system are optimized separately, indicating that the error propagation still exists. Thus, we would like to know whether we can directly train an end-to-end dialogue system for language e using the generated pseudo dataset. However, due to the translation errors and noises in the dataset, it is not enough to use $\mathcal{D}_{X_e \Rightarrow Y_e}$ to train the end-to-end dialogue system. Notice that the clean data can be used to enhance the noise data, we consider using multi-task learning to integrate different tasks to improve the end-to-end dialogue system in language e.

We employ the one-to-many scheme [10,26] to incorporate the training process of several tasks. As shown in Fig. 1, the scheme involves one shared encoder and multiple task-specific decoders for three language generation tasks: cross-lingual dialogue system, monolingual dialogue system, and MT. Here, cross-lingual dialogue system refers to the system, of which the input and response are in different languages.

Three pseudo datasets are used for the training procedure, including $\mathcal{D}_{X_e \Rightarrow Y_f}$, $\mathcal{D}_{X_e \Rightarrow Y_e}$ and $\mathcal{D}_{f \Rightarrow e}$. These datasets contain both clean data and pseudo data, and the clean data can help to improve the response generation for language e. Furthermore, the multi-task training procedure can enhance the encoder, thus minimizing the impact of the noise data. The loss function is as follows:

$$\begin{aligned} L(\theta_e, \theta_d^{ml}, \theta_d^{cl}, \theta_d^{mt}) = & \underbrace{\log p(Y_f | X_e; \theta_e, \theta_d^{cl})}_{\text{cross-lingual dialogue task}} + \underbrace{\log p(Y_e | X_e; \theta_e, \theta_d^{ml})}_{\text{monolingualdialoguetask}} \\ & + \underbrace{\log p(X_f | X_e; \theta_e, \theta_d^{mt})}_{\text{MTtask}} \end{aligned} \tag{5}$$

where θ_e denotes the shared encoder. θ_d^{ml}, θ_d^{cl}, and θ_d^{mt} are the decoder for monolingual dialogue task, cross-lingual dialogue task, and MT task, respectively.

4 Experiments

4.1 Experimental Settings

Dataset. We adopt a publicly available Chinese e-commerce dialogue corpus[1] [24] collected from Taobao[2] to conduct experiments. Chinese is the

[1] https://github.com/cooelf/DeepUtteranceAggregation.
[2] https://www.taobao.com.

high-resource language. We transfer the Chinese e-commerce dialogue service into English, Spanish and Korean under zero-shot setting. To verify our method, we manually translate the Chinese test set into the other three languages. More details about the dataset are given in Table 1.

Table 1. Dialog dataset statistics.

	Number of input-response pairs	Average of words	
		Input	Response
Train	517,525	31.28	11.77
Valid	4,402	31.64	11.65
Chinese-Test	5,204	32.63	11.73
English-Test	5,204	36.19	13.70
Spanish-Test	5,204	31.72	12.06
Korean-Test	5,204	21.40	7.93

Evaluation Metrics. We conduct evaluation with both automatic metrics and human evaluation. For automatic evaluation, We adopt several widely used metrics [6,7,11,19] to measure the performance of our proposed method, including word overlap metrics (BLEU-4, METEOR, ROUGE-L), distinct metrics (Dist-1/2), and normalized average sequence length (NASL). We also carry out a human evaluation for a more realistic comparison of our proposed methods to the baselines. We focus on evaluating the generated responses from three aspects: (1) **Relevance**: if the response is relevant to the given history; (2) **Informative**: if the response contains informative and interesting content; and (3) **Fluency**: whether the response is fluent without grammatical error. The details of human evaluation will be described in the corresponding part.

Implementation Details. We use Byte-Pair Encoding (BPE) with 30K merge operations to segment Chinese, English, Spanish, and Korean into subword granularities. For the Transformer-based dialogue system, the vocabulary size of the source and target words are both 30K. We train our models using configuration *transformer_base* [21], which contains a 6-layer encoder and a 6-layer decoder with 512 dimension hidden representations. During training, we apply Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.998$, and $\epsilon = 10^{-9}$. In the cyclic translation procedure, we adopt the Google translator[3] to generate the pseudo dataset. For the self-trained NMT systems in the noise injection method, we also use configuration *transformer_base* to train the NMT systems.

4.2 Experimental Results and Analysis

Results of Noise Injection Method. Table 2 shows the experimental results of the noise injection method. We can reach the following conclusions:

[3] https://translate.google.com/.

Table 2. Experimental results of the *noise injection method*. In the noise injection method, modules in the parentheses mean which modules use the noised models while the rest use the original. *Dial* denotes dialogue.

#	System	BLEU-4	METEOR	ROUGE-L	Dist-1/2	NASL
Upper bound (Input = ZH)						
1	Dial	**9.54**	**11.34**	**18.38**	**0.0410/0.2082**	1.0589
Baseline (Input = EN; MT_1 = MT_2 = Google Translator)						
2	MT_1+Dial	3.93	7.56	12.53	0.0384/0.1908	1.1377
3	MT_1+Dial+MT_2	2.55	5.26	9.31	0.0323/0.1987	1.1345
4	End-to-end	2.90	6.06	10.50	0.0307/0.1830	1.0539
Noise Injection method (Input = EN)						
5	*Noise*(MT_1+Dial+MT_2)	**3.57**	**6.56**	**11.13**	0.0346/0.2107	0.9799
6	*Noise*(MT_1+Dial)	3.35	6.48	11.00	0.0362/0.2154	0.9687
7	*Noise*(Dial+MT_2)	2.90	5.76	10.08	0.0349/0.2162	0.9873
8	*Noise*(MT_1+MT_2)	3.41	6.28	10.72	0.0340/0.1975	0.9725
9	*Noise*(MT_1)	3.24	6.29	10.69	**0.0366**/0.2088	0.9716
10	*Noise*(Dial)	2.76	5.80	10.03	0.0362/**0.2194**	0.9890

Table 3. Experimental results of *multi-task training method*. *MonoDial* denotes the monolingual dialogue system and *CrossDial* denotes the cross-lingual dialogue system.

#	System	BLEU-4	METEOR	ROUGE-L	Dist-1/2	NASL
1	Dial	**9.54**	**11.34**	**18.38**	**0.0410/0.2082**	1.0589
2	End-to-end	2.90	6.06	10.50	0.0307/0.1830	1.0539
3	*Noise*(MT_1+Dial+MT_2)	3.57	6.56	11.13	**0.0346/0.2107**	0.9799
Multi-task training method						
4	MonoDial+CrossDial+MT	**3.81**	**6.59**	**11.51**	0.0269/0.1447	0.9783
5	MonoDial+CrossDial	3.32	6.15	10.87	0.0335/0.1981	0.9832
6	MonoDial+MT	3.57	6.28	11.14	0.0280/0.1465	0.9371

i) The MT-based pipeline dialogue system suffers heavily from error propagation. Compared to the dialogue model given the clean Chinese test data (line 1), the system's performance degrades drastically if the input is a noise input translated from English (line 2). After Google translator translates the Chinese response back into English, the performance continues to decline (line 3). The performance of the end-to-end model (line 4) is better than the pipeline system (line 3), which proves the end-to-end model can avoid the problem of error propagation to a certain extent. However, the performance of the end-to-end model is still seriously harmed by the noises and errors in the translated pseudo data.

ii) The proposed noise injection method can boost the performance of the pipeline system (line 5 and line 3). After using the noised MT (including MT_1 and MT_2) and noised dialogue, the performance has gained impressively.

iii) We also investigate the effect of each noised model in the pipeline system (line 6 to line 10). We can see that each of the noised models can improve performance. Meanwhile, *noising* the first two models is more critical for improvement (line 6). This is because retraining the first two models can

Table 4. Experimental results of language transfer to other languages.

#	System	BLEU-4	METEOR	ROUGE-L	Dist-1/2	NASL
Chinese ⇒ Spanish						
1	MT_1+Dial+MT_2	2.37	4.20	6.07	**0.0620/0.2863**	1.0040
2	End-to-end	4.34	5.58	8.64	0.0493/0.2483	1.1579
3	*Noise*(MT_1+Dial+MT_2)	6.58	**6.88**	10.31	0.0539/0.2616	1.0934
4	Multi-task training	**6.66**	6.74	**10.49**	0.0434/0.2027	1.0594
Chinese ⇒ Korean						
5	MT_1+Dial+MT_2	1.42	7.46	2.68	**0.1411/0.4394**	1.0389
6	End-to-end	3.39	9.70	4.86	0.1247/0.3877	1.2519
7	*Noise*(MT_1+Dial+MT_2)	**5.46**	10.34	6.38	0.1362/0.4141	1.0943
8	Multi-task training	5.31	**10.64**	**6.73**	0.1119/0.3323	1.2291

make the dialogue system act more appropriately when given the translated utterances.

Besides, the performance of the Dist-1/2 is different from the word overlap metrics. The end-to-end model (line 4) achieves higher word overlap metrics while the MT-based pipeline dialogue system (line 3) obtains higher Dist-1/2. Our proposed method can improve both the word overlap metrics and diversity of the responses, demonstrating our proposed noise injection method's effectiveness compared with the MT-based pipeline dialogue system.

Results of Multi-task Training Method. The experimental results are shown in Table 3. Compared to the end-to-end model (line 2), training monolingual dialogue system and cross-lingual dialogue system simultaneously improves both the word overlap metrics and diversity of the generated responses (line 5). MT task is helpful for the word overlap metrics but harmful for the diversity. When combining the monolingual dialogue task and MT task (line 6), the word overlap metrics are higher than those of the end-to-end model, but the diversity is lower. *MonoDial+CrossDial+MT* (line 4) outperforms the other two (line 5–6) in word overlap metrics since it uses both the history and response in the Chinese dialogue dataset. However, the diversity reaches the lowest. This illustrates that the Chinese dialogue dataset can boost the performance of the end-to-end English dialogue system when only pseudo data is available.

Results of Transfer to Other Languages. We further conduct experiments on the transfer from Chinese to Spanish and Korean. The settings are the same as Chinese to English transfer. The experimental results are shown in Table 4. Our two methods outperform the baseline systems by a big margin from the word overlap perspective, demonstrating that our proposed two methods effectively transfer the dialogue system to a new language by only using knowledge in data-rich language and MT. The word overlap metrics except for METEOR on Korean are much lower than those in English and Spanish. This may be because the Chinese to Korean translation performance is not that ideal, and the generated Korean pseudo data may contain much more noise than that of English and Spanish. Besides, the diversity score of the multi-task training method is lower than the noise injection method both in Spanish and Korean.

Table 5. The effect of machine translation performance.

#	System	BLEU-4	METEOR	ROUGE-L	Dist-1/2	NASL
The size of MT training corpus: 500K						
1	MT_1+Dial+MT_2	0.25	2.71	5.82	**0.0512/0.3027**	0.8034
2	End-to-end	0.46	3.17	6.79	0.0409/0.2379	0.8414
3	*Noise*(MT_1+Dial+MT_2)	**0.63**	**3.85**	**7.38**	0.0435/0.2536	0.9140
4	Multi-task training	0.57	3.40	7.15	0.0340/0.1858	0.7946
The size of MT training corpus: 1M						
5	MT_1+Dial+MT_2	0.26	2.64	5.84	**0.0508/0.3181**	0.7656
6	End-to-end	0.51	3.38	6.76	0.0406/0.2369	0.8661
7	*Noise*(MT_1+Dial+MT_2)	**0.75**	**3.93**	**7.65**	0.0432/0.2590	0.8964
8	Multi-task training	0.63	3.75	7.26	0.0332/0.1873	0.8447
The size of MT training corpus: 2M						
9	MT_1+Dial+MT_2	0.33	2.77	5.97	**0.0450/0.2708**	0.7620
10	End-to-end	0.70	3.57	7.34	0.0387/0.2303	0.7985
11	*Noise*(MT_1+Dial+MT_2)	**1.03**	**4.35**	**8.37**	0.0401/0.2457	0.8790
12	Multi-task training	0.80	3.88	8.12	0.0337/0.1894	0.7571
MT_1=MT_2 = Google Translator						
13	MT_1+Dial+MT_2	2.55	5.26	9.31	0.0323/0.1987	1.1345
14	End-to-end	2.90	6.06	10.50	0.0307/0.1830	1.0539
15	*Noise*(MT_1+Dial+MT_2)	3.57	6.56	11.13	**0.0346/0.2107**	0.9799
16	Multi-task training	**3.81**	**6.59**	**11.51**	0.0269/0.1447	0.9783

Table 6. Human evaluation results.

System	Relevance	Informative	Fluency
MT_1+Dial+MT_2	2.26	2.90	3.36
End-to-end	2.71	2.91	3.48
Noise(MT_1+Dial+MT_2)	2.95	3.11	3.43
Multi-task training	**3.17**	**3.22**	**3.66**

The Effect of MT Performance. As introduced before, the MT engines play an essential role, since our proposed two methods are based on the pseudo data generated by the MT engines. The above experiments show the effectiveness of the two methods when adopting Google translator. However, MT performance is not always satisfied. *How will our methods help if we do not have a good MT?*

To investigate the effect of MT, we trained another three ZH → EN (EN → ZH) translation systems with *transformer_base* configuration using 500K, 1M, and 2M sentence pairs extracted from the English-Chinese Machine Translation track corpus[4]. We use these three translation systems to replace Google translator and simulate low resources and poor translation performance situations. The results are presented in Table 5.

Although the overall performance drops sharply when using the pseudo data generated by self-trained MT systems, our proposed two methods can help boost the performance to a certain extent. From Table 5, we can find an interesting

[4] https://challenger.ai/datasets/translation.

Input	C: What do you suggest to buy? S: Personally i like tea needles which still depends on hobbies C: I don't understand, just a brick tea, I don't know how to open it and what tools are needed to open it ?
Gold Response	Just choose a *tea knife* or a *tea needle*
Pipeline	Yes dear, what you shot was a three-piece set
End-to-End	I' m really sorry because the big warehouse in the order before the holiday has already been packed overtime. Oh usually it will be shipped to you within 48 hours and you wait patiently here.
Noise Injection	The effect is similar to the kisses this person sees that they like me personally they like *tea knives* most of them can be considered.
Multi-task	I suggest you use a large *tea knife*.

Fig. 2. Response examples with various systems.

phenomenon: When using Google translator to generate pseudo data, the *multi-task training method* achieves the best performance, while the *noise injection method* acquires the best when using self-trained MT systems. The reasons are two-fold: (1) When using Google translator, the quality of pseudo data is relatively high, and the multi-task training procedure can enhance the encoder and learn better representations for the noise data. (2) Due to the training corpus size, the self-trained MT's performance is much more unsatisfactory, resulting in more noises in the pseudo data. In this situation, the noise injection method is more helpful since it optimizes each module and injects the same noise into the MT systems and dialogue system, making the system more robust. All in all, our proposed two methods can be beneficial, even if only a weak MT system is available. Furthermore, the experimental results also indicate that the *multi-task training method* will be more useful when the quality of MT is good. Otherwise, the *noise injection method* will be more helpful.

Human Evaluation. We conduct the human evaluation on 150 random samples from the English test set, and these responses are based on distinct dialogue history. We compare responses generated by our methods with the responses generated by baselines. Three graduate students are asked to judge the quality of the responses according to relevance, informative, and fluency with a score from 1 (worst) to 5 (best). The student is presented with a dialogue history and four outputs with the name anonymized in each judgment. The average scores are presented in Table 6.

Compared to the baseline pipeline system, the end-to-end model generates better responses. Moreover, the fluency score is even a bit higher than that of the noise injection method. Our noise injection method significantly improves all three scores compared with the baseline pipeline system. This is mainly because the noised MT system is more relevant to the dialogue task. More importantly, the noised dialogue system has experienced more noise data from MT and better handles noise utterance. The multi-task training method outperforms the end-to-end by an impressive margin. Overall, the results suggest that our proposed methods can effectively improve dialogue systems' ability to generate more appropriate responses when transferring the dialogue system to a new language.

Case Study. The above results show that our proposed two methods can deploy and enhance a dialogue system for the new language. To further verify our methods, we show an example of response generation with various systems in Fig. 2. We can see that the responses generated by the noise injection method and multi-task training are better than the two baseline systems. The two responses generated by the two baseline systems are irrelevant to the dialogue context. The response generated by the noise injection method mentions *tea knives*, while the response generated by multi-task training can be regarded as a proper response. We can also find that some of the generated responses are not fluent. Nevertheless, it does not hinder the real application since people can understand as long as the system expresses critical information.

5 Related Work

The study of cross-lingual dialogue systems has gained much attention, and it studies how to adapt a dialogue system into the target language. The current work can be divided into three categories: cross-lingual NLU [1,8,9,13,15], cross-lingual DST [3,9,13] and cross-lingual response selection [14]. [2] proposed a multi-task learning architecture with share-private memory for multilingual open-domain dialogue generation, which is different from ours since they aimed at learning the common features among languages to boost dialogue systems.

Existing Cross-lingual transfer learning methods can be divided into two categories: transfer through cross-lingual representations [4,12,20] and transfer through MT [5,15,27]. In this paper, we focus on using MT to bridge the language gap between data-rich and zero-resource languages.

The differences between our work and the above work are two-fold: (1) There is no work in cross-lingual dialogue systems focusing on building a complete dialogue system for a new language under the zero-resource setting, which is the focus of this paper. (2) To the best of our knowledge, none of the work has explored how to use MT to transfer a generative dialogue system to a new language. This paper will study how to deploy a dialogue system for a new language by transferring knowledge from data-rich language and MT.

6 Conclusion

In this paper, we present cross-lingual transfer for dialogue systems under the zero-resource scenario. To alleviate this problem, we propose two methods to boost the pipeline system and the end-to-end system with the help of existing MT engines and training data in data-rich language. We first use MT and dialogue training data to generate various pseudo data. Then, the noise injection method is proposed to improve the pipeline system by injecting MT noises into the pipeline process, and the multi-task training method is proposed to enhance the end-to-end system. Experimental results have shown that our proposed methods can improve the dialogue system's performance for the new language. Furthermore, extended experiments demonstrate that our proposed methods are still useful even if only MT systems with poor performance are available.

Acknowledgments. This work was supported by the National Key R&D Program of China under Grant No. 2020AAA0108600.

References

1. Bai, H., Zhou, Y., Zhang, J., Zhao, L., Hwang, M.-Y., Zong, C.: Source critical reinforcement learning for transferring spoken language understanding to a new language. In: Proceedings of COLING (2018)
2. Chen, C., Qiu, L., Fu, Z., Liu, J., Yan, R.: Multilingual dialogue generation with shared-private memory. In: Proceedings of NLPCC (2019)
3. Chen, W., Chen, J., Su, Y., Wang, X., Yu, D., Yan, X., et al.: Xl-nbt: a cross-lingual neural belief tracking framework. In: Proceedings of EMNLP (2018)
4. Conneau, A., Lample, G.: Cross-lingual language model pretraining. In: Proceedings of NeurIPS (2019)
5. Jain, A., Paranjape, B., Lipton, Z. C.: Entity projection via machine translation for cross-lingual ner. In: Proceedings of EMNLP-IJCNLP (2019)
6. Liu, C. W., Lowe, R., Serban, I., Noseworthy, M., Charlin, L., Pineau, J.: How not to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In: Proceedings of EMNLP (2016)
7. Li, J., Galley, M., Brockett, C., Gao, J., Dolan, B.: A diversity-promoting objective function for neural conversation models. In: Proceedings of NAACL-HLT (2016)
8. Liu, Z., Shin, J., Xu, Y., Winata, G. I., Xu, P., Madotto, A., Fung P.: Zero-shot cross-lingual dialogue systems with transferable latent variables. In: Proceedings of EMNLP- IJCNLP (2019)
9. Liu, Z., Winata, G. I., Lin, Z., Xu, P., Fung, P.: Attention-informed mixed-language training for zero-shot cross-lingual task-oriented dialogue systems. In: Proceedings of AAAI (2020)
10. Luong, M., Le, Q.V., Sutskever, I., Vinyals, O., Kaiser, L.: Multi-task sequence to sequence learning. In: Proceedings of ICLR (2016)
11. Olabiyi, O., Salimov, A.O., Khazane, A., Mueller, E.: Multi-turn dialogue response generation in an adversarial learning framework. In: Proceedings of the First Workshop on NLP for Conversational AI (2019)
12. Pires, T., Schlinger, E., Garrette, D.: How multilingual is multilingual bert? In: Proceedings of ACL (2019)
13. Qin, L., Ni, M., Zhang, Y., Che, W.: Cosda-ml: multi-lingual code-switching data augmentation for zero-shot cross-lingual nlp. In: Proceedings of IJCAI (2020)
14. Sato, M., Ouch, H., Tsuboi, Y.: Addressee and response selection for multilingual conversation. In: Proceedings of COLING (2018)
15. Schuster, S., Gupta, S., Shah, R., Lewis, M.: Cross-lingual transfer learning for multilingual task oriented dialog. In: Proceedings of NAACL (2019)
16. Serban, I., Sordoni, A., Bengio, Y., Courville, A., Pineau, J.: Building end-to-end dialogue systems using generative hierarchical neural network models. In: Proceedings of AAAI (2016)
17. Shang, L., Lu, Z., Li, H.: Neural responding machine for short-text conversation. In: Proceedings of ACL-IJCNLP (2015)
18. Shao, Y., Gouws, S., Britz, D., Goldie, A., Strope, B., Kurzweil, R.: Generating high-quality and informative conversation responses with sequence-to-sequence models. In: Proceedings of EMNLP (2017)

19. Sharma, S., El Asri, L., Schulz, H., Zumer, J.: Relevance of unsupervised metrics in task-oriented dialogue for evaluating natural language generation. arXiv preprint arXiv:1706.09799 (2017)
20. Sun, J., Zhou, Y., Zong, C.: Dual attention network for cross-lingual entity alignment. In: Proceedings of COLING (2020)
21. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., et al.: Attention is all you need. In: Proceedings of NeurIPS (2017)
22. Vinyals, O., Le, Q.: A neural conversational model. arXiv preprint arXiv:1506.05869 (2015)
23. Wu, Y., Wu, W., Yang, D., Xu, C., Li, Z: Neural response generation with dynamic vocabularies. In: Proceedings of AAAI (2018)
24. Zhang, Z., Li, J., Zhu, P., Zhao, H., Liu, G. Modeling multi-turn conversation with deep utterance aggregation. In: Proceedings of COLING (2018)
25. Zhao, T., Lee, K., Eskenazi, M.: Unsupervised discrete sentence representation learning for interpretable neural dialog generation. In: Proceedings of ACL (2018)
26. Zhu, J., et al.: Ncls: neural cross-lingual summarization. In: Proceedings of EMNLP- IJCNLP (2019)
27. Zhu, J., Zhou, Y., Zhang, J., Zong, C.: Attend, translate and summarize: an efficient method for neural cross-lingual summarization. In: Proceedings of ACL (2020)