Data-Efficient Goal-Oriented Conversation with Dialogue Knowledge Transfer Networks

Igor Shalyminov[†], **Sungjin Lee**[‡], **Arash Eshghi**[†], **and Oliver Lemon**[†]
[†]Heriot-Watt University, UK
[‡]Amazon Alexa AI, USA
[†]{is33, a.eshghi, o.lemon}@hw.ac.uk, [‡]sungjinl@amazon.com

Abstract

Goal-oriented dialogue systems are now being widely adopted in industry where it is of key importance to maintain a rapid prototyping cycle for new products and domains. Data-driven dialogue system development has to be adapted to meet this requirement — therefore, reducing the amount of data and annotations necessary for training such systems is a central research problem.

In this paper, we present the Dialogue Knowledge Transfer Network (DiKTNet), a state-of-the-art approach to goal-oriented dialogue generation which only uses a few example dialogues (i.e. few-shot learning), none of which has to be annotated. We achieve this by performing a 2-stage training. Firstly, we perform *unsupervised dialogue representation pre-training* on a large source of goal-oriented dialogues in multiple domains, the MetaLWOz corpus. Secondly, at the *transfer stage*, we train DiKTNet using this representation together with 2 other textual knowledge sources with different levels of generality: ELMo encoder and the main dataset's source domains.

Our main dataset is the Stanford Multi-Domain dialogue corpus. We evaluate our model on it in terms of BLEU and Entity F1 scores, and show that our approach significantly and consistently improves upon a series of baseline models as well as over the previous state-of-the-art dialogue generation model, ZSDG. The improvement upon the latter — up to **10%** in Entity F1 and the average of **3%** in BLEU score — is achieved using only **10%** equivalent of ZSDG's in-domain training data.

1 Introduction

Machine learning-based dialogue systems, while still being a relatively new research direction, are experiencing increasingly wide adoption in industry. Large-scale dialogue assistant platforms such as *Google Assistant, Amazon Alexa, and Apple Siri* provide a unified conversational user interface (CUI) for third-party applications and services. Furthermore, products like *Google Dialogflow, Wit.ai, Microsoft LUIS, and Rasa* offer means for rapid development of a dialogue system's core modules. In addition, with the recently adopted technique of training dialogue systems end-to-end *data-efficiency* of such systems becomes the key question in their adoption in practical applications. Currently, while being extremely flexible and requiring little to no programming of in-domain business logic (see e.g. Ultes et al. (2018); Wen et al. (2017); Rojas-Barahona et al. (2017)), such systems have too high data consumption — including both collection and annotation effort — in order for them to be used in rapidly paced industrial product cycles. Therefore, approaches to training such systems with extremely limited data (i.e. zero-, one- and few-shot training) are a priority research direction in the dialogue systems area.

In this paper, we present the *Dialogue Knowledge Transfer Network* (or DiKTNet), a generative goal-oriented dialogue model designed for few-shot learning, i.e. training only using a small number of complete in-domain dialogues. The key underlying concept of this model is transfer learning: DiKTNet makes use of the latent text representation learned from several sources ranging from large-scale general-purpose textual corpora to similar dialogues in the domains different to the target one. We use the evaluation framework of Zhao and Eskénazi (2018) and the same dataset, and mainly compare our approach to theirs. While their method doesn't require complete in-domain dialogues and uses annotated utterances instead (and is therefore described as "zero-shot"), we show that our model achieves superior performance with roughly the same amount of data (with

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 1741–1751,
Hong Kong, China, November 3–7, 2019.

respect to in-domain utterances) while requiring no annotations whatsoever.

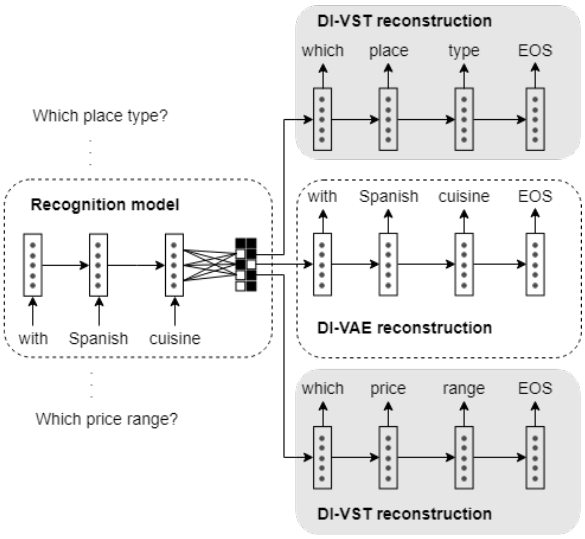

Figure 1: DI-VAE and DI-VST (DiKTNet Stage 1)

2 Related Work

The problem of data efficiency of dialogue systems has been extensively researched in the past. Starting with domain adaptation of a dialogue state tracker (Henderson et al., 2014) approached using Bayesian Processes (Gasic et al., 2017) and Recurrent Neural Networks (Mrksic et al., 2015), there has been significant work on training different dialogue system components using as little data as possible. As such, Williams et al. (2017) introduced a dialogue management model designed for bootstrapping from limited training data and further fine-tuning. A recent paper by Vlasov et al. (2018) introduced a dialogue management model which uses a unified embedding space for user and system turns allowing efficient cross-domain knowledge transfer.

There also exist approaches to end-to-end dialogue generation. Eshghi et al. (2017) proposed a linguistically informed model based on an incremental semantic parser (Eshghi et al., 2011) combined with a reinforcement learning-based agent. The parser was used for both maintaining the agent's state and pruning the agent's incremental, word-level generation actions (only the actions leading to syntactically correct word sequences were allowed for the agent to take). While outperforming end-to-end dialogue models on bAbI Dialog Tasks in a zero-shot setup (Shalyminov et al., 2017) due to its prior linguistic knowledge in the form of a dialogue grammar, this method inherited the limitations of it as well. Specifically, it's limited to a single domain until a wide-coverage grammar is available.

Meta-learning has also gained a lot of attention as a way to train models for maximally efficient adaptation to new data. As such, Qian and Yu (2019) presented such approach for fast adaptation of a dialogue model to a new domain. While highly promising, its main result was achieved on a synthetic dataset and would ideally need more testing on real data.

Finally, the method we directly compare our approach to is that of Zhao and Eskénazi (2018) who introduced the Zero-Shot Dialogue Generation (ZSDG) task and the corresponding model. In their work, they use a unified latent space for user utterances, system turns, and *domain descriptions* in the form of utterance-annotation pairs. Since they only used such utterances and no full dialogues for the target domain, they presented this approach as "zero-shot" learning. In our approach, we do use complete in-domain dialogues, but with significantly less data with respect to the number of in-domain utterances. Moreover, our method requires no annotation whatsoever.

Recent research in Natural Language Processing has shown that the transfer of text representation learned on larger data sources benefits target models' performance, just as was the case with ImageNet-based computer vision models (Deng et al., 2009).

For text, the main means for transfer was Word2Vec and GloVe embeddings (Mikolov et al., 2013; Pennington et al., 2014) recently extended with context-aware models like ELMo (Peters et al., 2018) and BERT (Devlin et al., 2018). Trained on large and diverse textual corpora, they were shown to improve target models' performance on a number of Natural Language Processing tasks. Although highly beneficial, those models' use may not be sufficient for the case of dialogue as response generation for goal-oriented dialogue from extremely limited data requires specialized tools. General-purpose embeddings lack specificity for close dialogue domains since they have been learned from very heterogeneously distributed data: in dialogue, the distribution of word sequences is highly specific to a given domain or task, i.e. word sequences in dialogue can take on an astonishingly wide variety of meanings in dif-

ferent contexts.

In this paper, we will work with *autoencoders*, a class of unsupervised text representation models working via reconstructing the input — specifically, a Variational Autoencoder (VAE) was considered the main means to learn robust text representations (Bowman et al., 2016). Although the model itself was challenging to train and was mainly used with plenty of workarounds, and recently there started to appear variants of this model with improved stability. One such model we will use in this paper is that of Zhao et al. (2018) (see Section 4.1 for more detail).

3 Few-Shot Dialogue Generation

We first describe the task we are addressing in this paper, and the corresponding base model. Specifically, we have a set of dialogues in source domains and just a few seed dialogues in the target domain. And the model's task is, having been trained on all the available *source data*, to fine-tune on the *target data* to be further evaluated on the full set of target-domain dialogues.

We are basing our model for this task on a Hierarchical Encoder-Decoder (HRED) architecture with attention-based copying (Merity et al., 2017). The base optimization objective is as follows:

$$\mathcal{L}_{\text{HRED}} = \log p_{\mathcal{F}^d}(\boldsymbol{x}_{\text{sys}} \mid \mathcal{F}^e(\boldsymbol{c}, \boldsymbol{x}_{\text{usr}})) \quad (1)$$

where $\boldsymbol{x}_{\text{usr}}$ is user's query, $\boldsymbol{x}_{\text{sys}}$ is the system's response, $\boldsymbol{c}$ is the dialogue context, and $\mathcal{F}^e$ and $\mathcal{F}^d$ are respectively hierarchical encoder and decoder.

We work with goal-oriented dialogues, so it's natural in our setting to take into account an underlying Knowledge Base (or API) providing results on the user's queries. Given that such KB information may contain unseen token sequences for the most part, especially in the target domain, we use a copy mechanism in order to be able to use this information in the system's responses. More specifically, we represent KB info as token sequences and concatenate it to the dialogue context similarly to CopyNet setup of Eric et al. (2017). Our copy mechanism's implementation is the Pointer-Sentinel Mixture Model (Merity et al., 2017; Zhao and Eskénazi, 2018):

$$p(w_t \mid s_t) = g p_{\text{vocab}}(w_t \mid s_t) + (1 - g) p_{\text{ptr}}(w_t \mid s_t) \quad (2)$$

In the formula above, w_t and s_t are respectively the output word and the decoder state at step t; p_{ptr} is the probability of attention-based copying of the word w_t, and g is the mixture weight:

$$p_{\text{ptr}}(w_t \mid s_t) = \sum_{k_j \in I(w, \boldsymbol{x})} \alpha_{k_j, t} \quad (3)$$

$$g = \text{Softmax}(u^T \tanh(W_\alpha s_i)) \quad (4)$$

where $\alpha_{k_j,t}$ is the attention weight for kth token in flattened dialogue context at the decoding step t and u is the sentinel vector — for more detail, see (Zhao and Eskénazi, 2018).

4 Dialogue Knowledge Transfer Network

Transfer learning is considered the key means for efficient training with minimal data, and our DiKTNet model essentially introduces several knowledge-transfer augmentations to the base HRED model described above. DiKTNet training is performed in two stages described below.

4.1 Stage 1. Dialogue representation pre-training

Dialogue structure — e.g. word sequences — is highly specific to a given domain or task, and the meaning of conversational utterances is highly contextual, i.e. similar utterances may have different meanings depending on the context. Nevertheless, there is a lot of similarity in dialogue structure — i.e. sequences of dialogue actions — across domains, e.g. a conversation normally starts with a mutual greeting and a question is very often followed by an answer. Here, we propose to exploit this phenomenon in the form of learning a latent dialogue action representation in order to better capture the dialogue structure by abstracting away from surface forms. Crucially, we learn such representation from MetaLWOz (Lee et al., 2019), a dataset specifically created for the purposes of meta-learning and transfer learning and consisting of human-human conversations in 51 unique domains (for more detail, see Section 6).

For this stage of training we use unsupervised, variational autoencoder-based (VAE) representation learning following the *Latent Action Encoder-Decoder (LAED)* approach of Zhao et al. (2018). LAED's underlying model is called *Discrete Information VAE (DI-VAE)*, a variant of a VAE with

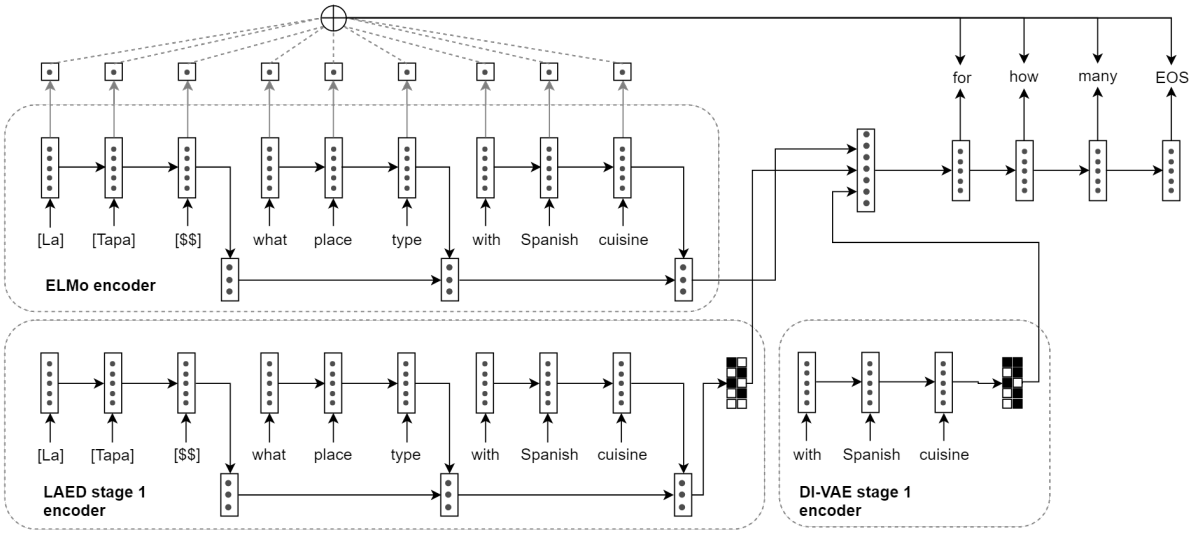

Figure 2: DiKTNet Stage 2 (tokens in brackets are KB data)

two modifications. Firstly, its optimization objective accounts for the mutual information I between the input and the latent variable which is implicitly discouraged in the original VAE objective (see Eqs. 5 and 6).

$$\begin{aligned}\mathcal{L}_{VAE} &= \mathbb{E}_x[\mathbb{E}_{q_\mathcal{R}(\boldsymbol{z}|\boldsymbol{x})}[\log p_\mathcal{G}(\boldsymbol{x} \mid \boldsymbol{z})] \\ &\quad - KL(q(\boldsymbol{z})\|p(\boldsymbol{z}))] = \\ &\quad \mathbb{E}_{q(\boldsymbol{z}|x)p(x)}[\log p_\mathcal{G}(\boldsymbol{x} \mid \boldsymbol{z})] \\ &\quad - I(Z, X) - KL(q(\boldsymbol{z})\|p(\boldsymbol{z})),\end{aligned} \tag{5}$$

$$\begin{aligned}\mathcal{L}_{DI\text{-}VAE} &= \mathcal{L}_{VAE} + I(Z, X) \\ &= \mathbb{E}_{q_\mathcal{R}(\boldsymbol{z}|\boldsymbol{x})p(\boldsymbol{x})}[\log p_\mathcal{G}(\boldsymbol{x} \mid \boldsymbol{z})] \\ &\quad - KL(q(\boldsymbol{z})\|p(\boldsymbol{z}))\end{aligned} \tag{6}$$

where $\boldsymbol{x}$ is the input utterance, $\boldsymbol{z}$ is the latent variable (X and Z corresponding to their batch-wise vectors), $\mathcal{R}$ and $\mathcal{G}$ are the recognition and generation models (implemented as RNNs) respectively, and $q(\boldsymbol{z}) = \mathbb{E}_x[q_\mathcal{R}(\boldsymbol{z} \mid x)]$.

Secondly, the latent variable $\boldsymbol{z}$ in DI-VAE is *discrete* as opposed to the continuous one in a vanilla VAE. The discrete latent code lends itself well to interpretation and can be viewed as a form of unsupervised dialogue act tagging. The discrete nature also makes the calculation of the KL-term more tractable via the *Batch Prior Regularization* technique (Zhao et al., 2018):

$$KL(q'(\boldsymbol{z})\|p(\boldsymbol{z})) = \sum_{k=1}^{K} q'(\boldsymbol{z}=k) \log \frac{q'(\boldsymbol{z}=k)}{p(\boldsymbol{z}=k)} \tag{7}$$

where K is the number of $\boldsymbol{z}$'s possible values and $q'(\boldsymbol{z})$ is the approximation to $q(\boldsymbol{z})$ over N data points:

$$q'(\boldsymbol{z}) = \frac{1}{N}\sum_{n=1}^{N} q_\mathcal{R}(\boldsymbol{z} \mid x_n) \tag{8}$$

In addition, we employ ***DI-VST***, DI-VAE's counterpart working in a Variational Skip-Thought manner (Hill et al., 2016) and reconstructing the input $\boldsymbol{x}$'s previous ($\boldsymbol{x_p}$) and next ($\boldsymbol{x_n}$) context utterances instead:

$$\begin{aligned}\mathcal{L}_{DI\text{-}VST} &= \\ &\mathbb{E}_{q_\mathcal{R}(\boldsymbol{z}|\boldsymbol{x})p(\boldsymbol{x})}[\log p^n_\mathcal{G}(\boldsymbol{x_n} \mid \boldsymbol{z}) p^p_\mathcal{G}(\boldsymbol{x_p} \mid \boldsymbol{z})] \\ &- KL(q(\boldsymbol{z})\|p(\boldsymbol{z}))\end{aligned} \tag{9}$$

DI-VAE and DI-VST models are visualized in Figure 1.

In the downstream DiKTNet model, we use DI-VAE autoencoder in order to obtain the representation of the user's query: $\boldsymbol{z}_{\text{usr}} = \text{DI-VAE}(\boldsymbol{x}_{\text{usr}})$.

In turn, DI-VST is used to obtain a prediction of the system's action $\boldsymbol{z}_{\text{sys}}$ in the discretized latent form given the user's input $\boldsymbol{x}_{\text{usr}}$ as well as the full dialogue context $\boldsymbol{c}$. For that, DI-VST autoencoder is used as part of a hierarchical, context-aware encoder-decoder response generation model (we refer to it as ***LAED*** itself). Its optimization

objective is as follows:

$$\mathcal{L}_{LAED}(\theta_{\mathcal{F}}, \theta_{\pi}) = \mathbb{E}_{q_{\mathcal{R}}(\boldsymbol{z}_{\text{sys}}|\boldsymbol{x}_{\text{sys}})p(\boldsymbol{x}_{\text{sys}},\boldsymbol{c})}[\log p_{\pi}(\boldsymbol{z}_{\text{sys}} \mid \boldsymbol{c}) + \log p_{\mathcal{F}}(\boldsymbol{x}_{\text{sys}} \mid \boldsymbol{z}_{\text{sys}}, \boldsymbol{c})] \quad (10)$$

where $\theta_{\mathcal{F}}$ is the set of parameters of the context-aware encoder and decoder, θ_{π} is the set of parameters of the policy θ_{π}. θ_{π} is the component trained to directly predict $\boldsymbol{z}_{\text{sys}}$ from the context $\boldsymbol{c}$.

We use different models for different aspects of the dialogue: DI-VAE for user's utterance representation, and DI-VST-based LAED — for the system's action prediction. In that, we follow the intuition of Zhao and Eskénazi (2018) who said that DI-VAE is better at capturing specific words of an utterance, while DI-VST represents the overall dialogue action better.

We train these two models on MetaLWOz in an unsupervised way with the objectives as described above, and use their discretized latent codes $\boldsymbol{z}_{\text{usr}}$ and $\boldsymbol{z}_{\text{sys}}$ respectively in the downstream model at the next stage of training.

4.2 Stage 2. Transfer

At this stage, we train directly for our target task, few-shot dialogue generation, and thus go back to the model described in Section 3. While the training procedure of this model naturally assumes *domain transfer*, we will provide it with more sources of textual and dialogue knowledge of varying generality described below.

As opposed to direct domain transfer, we incorporate domain-general dialogue understanding from the LAED representation trained on MetaLWOz at the previous stage. LAED captures the background top-down dialogue structure: sequences of dialogue acts in a cooperative conversation, latent dialog act-induced clustering of utterances, and the overall phrase structure of spoken utterances. We incorporate this information into the model by conditioning HRED's decoder on the combined latent codes from Stage 1 and refer to this model as ***HRED+LAED***.

$$\mathcal{L}_{\text{HRED+LAED}} = \mathbb{E}_{p(\boldsymbol{x}_{\text{usr}},\boldsymbol{c})p(\boldsymbol{z}_{\text{usr}}|\boldsymbol{x}_{\text{usr}})p_{\pi}(\boldsymbol{z}_{\text{sys}}|\boldsymbol{x}_{\text{usr}},\boldsymbol{c})}[\log p_{\mathcal{F}^d}(\boldsymbol{x}_{\text{sys}} \mid \{\mathcal{F}^e(\boldsymbol{x}_{\text{usr}}, \boldsymbol{c}), \boldsymbol{z}_{\text{usr}}, \boldsymbol{z}_{\text{sys}}\})]. \quad (11)$$

where $\boldsymbol{z}_{\text{usr}}$ and $\boldsymbol{z}_{\text{sys}}$ are respectively samples obtained from the DI-VAE user utterance model and LAED/DI-VST system action model, and $\{\}$ is the concatenation operator.

The last, most general source of knowledge we use is a pre-trained ELMo model (Peters et al., 2018). Apart from using an underlying bidirectional RNN encoder, ELMo captures both token-level and character-level information which is especially crucial in understanding unseen tokens and KB items in the underrepsesented target domain. HRED model with ELMo as the utterance-level encoder is referred to as ***HRED+ELMo***.

Finally, ***DiKTNet*** is HRED augmented with both ELMo encoder and LAED representation.

DiKTNet is visualized in Figure 2. The model (as well as its variants listed above) is implemented in PyTorch (Paszke et al., 2017), and the code is openly available[1].

5 Baselines

We perform an exhaustive ablation study of DiKTNet by comparing it to all its variations mentioned above: HRED, HRED+ELMo, and HRED+LAED. In addition to that, we have the ***HRED+VAE*** —a version of HRED+LAED for which we use a regular, continuous VAE behind DI-VAE and DI-VST in order to see the impact of discretized latent codes (see Eq 5 for the corresponding objective function).

Furthermore, we compare DiKTNet to the previous state-of-the-art approach, Zero-Shot Dialogue Generation (Zhao and Eskénazi, 2018). This model didn't use any complete in-domain dialogues but instead it relied on annotated utterances in all of the domains. We use it as-is (***ZSDG***) as well its variation as follows.

We make use of its central idea of 'domain descriptions' bridging dialogue understanding across domains, but instead of using manually annotated utterances, we employ automatic Natural Language Understanding markup. Our NLU annotations include:

- Named Entity Recognition — Stanford NER model ensemble of case-sensitive and caseless models (Finkel et al., 2005),
- date/time markup — Stanford SUTime (Chang and Manning, 2012),
- Wikidata entity linking — Yahoo FEL (Blanco et al., 2015; Pappu et al., 2017).

[1] https://bit.ly/fsdg_emnlp2019

Model \ Domain	Navigation		Weather		Schedule	
	BLEU, %	Entity F1, %	BLEU, %	Entity F1, %	BLEU, %	Entity F1, %
ZSDG	5.9	14.0	8.1	31	7.9	36.9
NLU_ZSDG	6.1 ± 2.2	12.7 ± 3.3	5.0 ± 1.6	16.8 ± 6.7	6.0 ± 1.7	26.5 ± 5.4
NLU_ZSDG+LAED	7.9 ± 1	12.3 ± 2.9	8.7 ± 0.6	21.5 ± 6.2	8.3 ± 1	20.7 ± 4.8
HRED@1%	6.0 ± 1.8	9.8 ± 4.8	6.9 ± 1.1	22.2 ± 10.7	5.5 ± 0.8	25.6 ± 8.2
HRED@3%	7.9 ± 0.7	11.8 ± 4.4	9.6 ± 1.8	39.8 ± 7	8.2 ± 1.1	34.8 ± 4.4
HRED@5%	8.3 ± 1.3	15.3 ± 6.3	11.5 ± 1.6	38.0 ± 10.5	9.7 ± 1.4	37.6 ± 8.0
HRED@10%	9.8 ± 0.8	19.2 ± 3.2	12.9 ± 2.4	40.4 ± 11.0	12.0 ± 1.0	38.2 ± 4.2
HRED+VAE@1%	3.6 ± 2.6	9.3 ± 4.1	6.8 ± 1.3	23.2 ± 10.1	4.6 ± 1.6	28.9 ± 7.3
HRED+VAE@3%	6.9 ± 1.9	15.6 ± 5.8	9.5 ± 2.6	32.2 ± 11.8	6.6 ± 1.7	34.8 ± 7.7
HRED+VAE@5%	7.8 ± 1.9	12.7 ± 4.2	10.1 ± 2.1	40.3 ± 10.4	8.2 ± 1.7	34.2 ± 8.7
HRED+VAE@10%	9.0 ± 2.0	18.0 ± 5.8	12.9 ± 2.2	40.1 ± 7.6	11.6 ± 1.5	39.9 ± 6.9
HRED+LAED@1%	7.1 ± 0.8	10.1 ± 4.5	10.6 ± 2.1	31.4 ± 8.1	7.4 ± 1.2	29.1 ± 6.6
HRED+LAED@3%	9.2 ± 0.8	14.5 ± 4.8	13.1 ± 1.7	40.8 ± 6.1	9.2 ± 1.2	32.7 ± 6.1
HRED+LAED@5%	10.3 ± 1.2	15.6 ± 4.5	14.5 ± 2.2	40.9 ± 8.6	11.8 ± 1.9	37.6 ± 6.1
HRED+LAED@10%	12.3 ± 0.9	17.3 ± 4.5	17.6 ± 1.9	47.5 ± 6.0	15.2 ± 1.6	38.7 ± 8.4
HRED+ELMo@1%	5.8 ± 1.9	$18.2 \pm 3.8^{\star}$	7.3 ± 2.6	38.5 ± 11.1	6.3 ± 2.6	36.3 ± 9.2
HRED+ELMo@3%	8.0 ± 1.3	17.2 ± 4.2	10.6 ± 1.1	42.0 ± 11.0	9.5 ± 2.0	39.6 ± 9.2
HRED+ELMo@5%	9.4 ± 0.8	21.5 ± 7.3	12.1 ± 2.0	39.0 ± 12.8	11.3 ± 2.1	40.0 ± 5.6
HRED+ELMo@10%	9.9 ± 1.1	24.3 ± 5.7	14.9 ± 2.7	41.4 ± 12.0	14.5 ± 1.4	43.4 ± 3.9
DiKTNet@1%	$\mathbf{8.4 \pm 0.7^{\star}}$	$\mathbf{15.2 \pm 4.0}$	$\mathbf{11.5 \pm 1.7^{\star}}$	$\mathbf{43.0 \pm 10.5^{\star}}$	$\mathbf{8.1 \pm 0.8^{\star}}$	$\mathbf{40.5 \pm 6.3^{\star}}$
DiKTNet@3%	10.4 ± 1.2	19.2 ± 4.8	15.7 ± 2.1	44.0 ± 11.7	11.1 ± 1.3	38.2 ± 5.8
DiKTNet@5%	11.5 ± 1.1	23.9 ± 2.9	15.5 ± 2.1	39.5 ± 14.8	13.7 ± 2.0	41.1 ± 3.8
DiKTNet@10%	12.9 ± 1.0	26.8 ± 4.2	20.4 ± 1.2	48.0 ± 5.6	17.5 ± 1.3	42.8 ± 2.6

Table 1: Evaluation results. Marked with asterisks are individual results higher than ZSDG's performance which are achieved with the minimum amount of training data. In bold is the model consistently outperforming ZSDG in all domains and metrics with minimum data.

We serialize annotations from these sources into token sequences and make domain description tuples out of all the utterances in the source and target domains. This way, most of our domain descriptions share the structure and content of the original ones.

For example, for the phrase *'Will it be cloudy in Los Angeles on Thursday?'*, the original ZSDG annotation is of the form `"request #goal cloudy #location Los Angeles #date Thursday"`. Our NLU annotation for this phrase is `"LOCATION Los Angeles DATE Thursday"`.

We have two models in this setup, with (***NLU_ZSDG+LAED***) and without the use of LAED representation (***NLU_ZSDG***) respectively.

6 Datasets

Number of Domains:	51
Number of Dialogues:	40,388
Mean dialogue length:	11.91

Table 2: MetaLWOz dataset statistics

We use the Stanford Multi-Domain (SMD) dialogue dataset (Eric et al., 2017) containing human-human goal-oriented dialogues in three domains: appointment scheduling, city navigation, and weather information. Each dialogue has to do with a single task queried by the user and thus comes with additional knowledge base information coming from implicit querying of the underlying domain-specific API. Although sharing some common features (the setting of an intelligent in-car assistant and the use of the underlying KB), the dialogues differs significantly across domains which makes doamin transfer sufficiently challenging.

Statistic \ Domain	Navigation	Weather	Schedule
Dialogues	800	797	828
Utterances	5248	4314	3170
Avg. dialogue length	6.56	5.41	3.83

Table 3: Stanford multi-domain dataset statistics (trainset)

For the latent representation learning, we use MetaLWOz, a goal-oriented dialogue dataset containing human-human dialogues in diverse domains and several tasks in each of those. The dialogues are collected in a Wizard-of-Oz method where human participants were given a problem domain and a specific task in it, and were asked to complete the task via dialogue. No domain-specific APIs or knowledge bases were available

for the participants, and in the actual dialogues they were free to use fictional names and entities in a consistent way. The dataset's statistics are shown in Table 2. All the domains available in the MetaLWOz dataset are listed in the Table 6 of the Appendix A.

7 Experimental setup and evaluation

Our few-shot setup is as follows. Given the target domain, we first train LAED model(s) on the MetaLWOz data — here we exclude from training every domain that might overlap with the target one. Specifically, for the *Navigation* domain in SMD, it's *Store Details*, for *Weather* it's *Weather Check*, and for *Schedule* it's *Update Calendar* and *Appointment Reminder*.

In our final setup, at Stage 1 we used a DI-VST-based LAED and a DI-VAE, both of the size 10×5.

Next, having trained and frozen Stage 1 models, we train DiKTNet on all the source domains from the SMD dataset. We use a random sample of the target domain utterances together with their contexts and KB info, varying the amount of those from 1% to 10% of all available target data.

For the NLU_ZSDG setup, we annotated all available SMD data and randomly selected a subset of 1000 utterances from each source domain, and 200 utterances from the target domain. For source domains, this number amounts to roughly a quarter of all available training data — we chose it in order to make use of as much annotated data as possible while keeping the domain description task secondary. For the target domain, we made sure to keep under roughly the same in-domain data requirements as the ZSDG baseline.

For evaluation, we follow the approach of Zhao and Eskénazi (2018) and report BLEU and Entity F1 scores. Given the non-deterministic nature of our training setup, we report means and variances of our results over 10 runs with different random seeds.

We also perform an additional evaluation of DiKTNet's performance with extended amounts of target data and compare it to the original Key-Value Retrieval Network (***KVRet***) by Eric et al. (2017) which was originally trained with all the available data. In this case we average BLEU scores across all 3 SMD domains in order to be consistent with the form the corresponding results are presented in the original paper.

We train our models with the Adam optimizer (Kingma and Ba, 2014) with learning rate 0.001. Our hierarchical models' utterance encoder is an LSTM cell (Hochreiter and Schmidhuber, 1997) of size 256, and the dialog-level encoder is a GRU (Cho et al., 2014) of size 512.

8 Results and discussion

Our results are shown in Table 1 — our objective here is maximum accuracy with minimum training data required.

8.1 Results for the few-shot setup

It can be seen that few-shot models with LAED representation are the best performing models for this objective. While improvements upon ZSDG can already be seen with simple HRED in a few-shot setup, the use of the LAED representation and domain-general ELMo encoding helps significantly reduce the amount of in-domain training data needed: at 1% of in-domain dialogues, we see that DiKTNet consistently and significantly improves upon ZSDG in every domain. In SMD, with its average dialogue length of 5.25 turns, 1% of training dialogues amounts to approximately 40 in-domain training utterances. In contrast, the ZSDG setup used approximately 150 training utterance-annotation pairs for each domain, including the target one, totalling about *450 annotated* utterances.

Although in our few-shot approach we use full in-domain dialogues, we end up having significantly less in-domain training data, with the crucial difference that none of those has to be annotated for our approach. Therefore, the method we introduced attains state-of-the-art in both accuracy and data-efficiency.

In turn, the results of the ZSDG_NLU setup demonstrate that single utterance annotations, if not domain-specific and produced by human experts, don't provide as much signal as full dialogues, even without annotations at all. Even the significant number of such annotated utterances per domain didn't make a difference in this case.

We would also like to point out that, as can be seen in the table, our results have quite high variance — the main source of it is the nature of our training/evaluation setup where we average over 10 runs with 10 different sets of seed dialogues. However, in the majority of cases with comparable means, DiKTNet has a lower variance than

Domain		Context	Gold response	Predicted response
schedule	<usr>	Remind me to take my pills	Ok setting your medicine appointment for 7pm	Okay, setting *a reminder to take your pills at 7 pm.*
	<sys>	What time do you need to take your pills?		
	<usr>	I need to take my pills at 7 pm.		
navigate	<usr>	Find the address to a hospital	Have a good day	*No problem.*
	<sys>	Stanford Express Care is at 214 El Camino Real.		
	<usr>	Thank you.		
weather	<usr>	What is the weather forecast for the weekend?	For what city would you like to know that?	For what city would you like *the weekend forecast for?*

Table 4: DiKTNet's selected responses

Where can I go shopping?
Where does my friend live?
Where can I get Chinese food?
Where can I go to eat?
Can you please take me to a coffee house?
I'd like to set a reminder for my meeting at 2pm later this month please.
What is the time and agenda for my meeting, and who is attending?
Schedule a lab appointment with my aunt for the 7th at 1pm.
Schedule a calendar reminder for yoga with Jeff at 6pm on the 5th.
Car I'm desiring to do some shopping: which one is it the nearest shopping ...
... center? Anything within 4 miles?
Get the address to my friend's house that i could get to the fastest
Car I need to get to my friends house, it should be within 4 miles from here

Table 5: Selected clusters of utterances sharing the same LAED codes

the alternative models at the same percentage of seed data. And in the extreme case with 1% target data, DiKTNet improves on all the other models in terms of both means and variances.

8.2 Discussion of the latent representations

The comparison of the setups with different latent representations also gives us some insight: while the VAE-powered HRED model improves on the baseline in multiple cases, it lacks generalization potential compared to the LAED setup. The reason for that might be inherently more stable training of LAED due to its modified objective function which in turn results in a more informative representation providing better generalization. In order to have a glimpse into the LAED-produced clustering, in Table 5 we present a snippet of the utterance clusters sharing the same, most frequent latent codes throughout the dataset (the clustering is obtained with LAED model trained on every domain but 'Store details', i.e. the one for the evaluation on 'Navigate' SMD domain). From this snippet, it can be seen that those clusters work well for domain separation, as well as capturing dialogue intents.

8.3 Results with extended data

We performed an additional experiment with extended target data (see Figure 3 of Appendix A). It showed that DiKTNet, when trained with as little as 5% of target data, can outperform a KVRet trained using the entire dataset. Furthermore, with 50% of the target data, DiKTNet becomes more than twice as good as KVRet in terms of overall language generation.

However, goal-oriented metrics such as Entity F1 are more challenging to bootstrap. As such, DiKTNet outperforms KVRet on 'Weather' domain starting at 10% of the target data, but only has a trend on narrowing down the performance gap with KVRet on 'Navigate', and certainly needs more training data in 'Schedule' domain.

The explanation to that might be that most of the dialogue entities come from the KB snippets which are the least represented resource in our setup. They aren't available in MetaLWOz, and in SMD, KB snippets share little in common across domains. Therefore, in order to increase Entity F1, KB information should be directly copied to the output more efficiently — and increasing the robustness of the copy-augmented decoder is one of our future research directions.

8.4 Discussion of the evaluation metrics

We use BLEU as one of the main evaluation metrics in this paper — we do it in order to fully conform with the setup of Zhao and Eskénazi (2018) which we base our work on. But while being widely adopted as a general-purpose language generation metric, BLEU might not be sufficient in the dialogue settings (see Novikova et al. (2017) for a review). Specifically, we have observed several cases where the model would produce an overall grammatical response with the correct dialogue intent (e.g. "You are welcome! Anything else?"), but BLEU would output a lower score for it due to word mismatch (e.g. "You're welcome!"; see more examples in Table 4). This is a general issue in dialogue model evaluation since the variability of possible responses equivalent in meaning is very high in dialogue. In future work, we will put more emphasis on the meaning of utterances, for example by incorporating external dialogue act tagging resources in the evaluation setup which, together with general language generation metrics like perplexity, can make for more robust evaluation criteria than word overlap.

9 Conclusion and future work

In this paper, we have introduced DiKTNet, a model achieving state-of-the-art dialogue generation performance in a few-shot setup, without using any annotated data. By transferring latent dialogue knowledge from multiple sources of varying generality, we obtained a model with superior generalization to an underrepresented domain.

Specifically, we showed that our few-shot approach achieves state-of-the art results on the Stanford Multi-Domain dataset while being more data-efficient than the previous best model, by requiring significantly less data none of which has to be annotated.

While being state-of-the-art, the accuracy scores themselves still suggest that our technique is not ready for immediate adoption for real-world production purposes, and the task of few-shot generalization to a completely new dialogue domain remains an area of active research. In our own future work, we will try and find ways to improve the unsupervised representation (Shi et al., 2019) in order to increase the transfer potential. We will also explore ways to enable more efficient copying from the input which is crucial for correctly handling entities and therefore attaining high goal-oriented performance of the system.

Apart from that, we will consider alternative evaluation criteria to account for rich surface variability of natural speech.

References

Roi Blanco, Giuseppe Ottaviano, and Edgar Meij. 2015. Fast and space-efficient entity linking in queries. In *Proceedings of the Eight ACM International Conference on Web Search and Data Mining*, WSDM 15, New York, NY, USA. ACM.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, Rafal Józefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, August 11-12, 2016*, pages 10–21.

Angel X. Chang and Christopher D. Manning. 2012. Sutime: A library for recognizing and normalizing time expressions. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012, Istanbul, Turkey, May 23-25, 2012*, pages 3735–3740.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1724–1734.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Mihail Eric, Lakshmi Krishnan, Francois Charette, and Christopher D. Manning. 2017. Key-value retrieval networks for task-oriented dialogue. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue, Saarbrücken, Germany, August 15-17, 2017*, pages 37–49.

A. Eshghi, M. Purver, and Julian Hough. 2011. Dylan: Parser for dynamic syntax. Technical report, Queen Mary University of London.

Arash Eshghi, Igor Shalyminov, and Oliver Lemon. 2017. Interactional Dynamics and the Emergence of Language Games. In *Proceedings of the ESSLLI 2017 workshop on Formal approaches to the Dynamics of Linguistic Interaction*, Barcelona.

Jenny Rose Finkel, Trond Grenager, and Christopher D. Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 25-30 June 2005, University of Michigan, USA*, pages 363–370.

Milica Gasic, Nikola Mrksic, Lina Maria Rojas-Barahona, Pei-Hao Su, Stefan Ultes, David Vandyke, Tsung-Hsien Wen, and Steve J. Young. 2017. Dialogue manager domain adaptation using gaussian process reinforcement learning. *Computer Speech & Language*, 45:552–569.

Matthew Henderson, Blaise Thomson, and Jason D. Williams. 2014. The third dialog state tracking challenge. In *2014 IEEE Spoken Language Technology Workshop, SLT 2014, South Lake Tahoe, NV, USA, December 7-10, 2014*, pages 324–329.

Felix Hill, Kyunghyun Cho, and Anna Korhonen. 2016. Learning distributed representations of sentences from unlabelled data. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 1367–1377.

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

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Sungjin Lee, Hannes Schulz, Adam Atkinson, Jianfeng Gao, Kaheer Suleman, Layla El Asri, Mahmoud Adada, Minlie Huang, Shikhar Sharma, Wendy Tay, and Xiujun Li. 2019. Multi-domain task-completion dialog challenge.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2017. Pointer sentinel mixture models. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 3111–3119.

Nikola Mrksic, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gasic, Pei-hao Su, David Vandyke, Tsung-Hsien Wen, and Steve J. Young. 2015. Multi-domain dialog state tracking using recurrent neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 2: Short Papers*, pages 794–799.

Jekaterina Novikova, Ondrej Dusek, Amanda Cercas-Curry, and Verena Rieser. 2017. Why we need new evaluation metrics for nlg. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Aasish Pappu, Roi Blanco, Yashar Mehdad, Amanda Stent, and Kapil Thadani. 2017. Lightweight multilingual entity extraction and linking. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, WSDM 17, New York, NY, USA. ACM.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch. In *NIPS-W*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 2227–2237.

Kun Qian and Zhou Yu. 2019. Domain adaptive dialog generation via meta learning. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 2639–2649.

Lina Maria Rojas-Barahona, Milica Gasic, Nikola Mrksic, Pei-Hao Su, Stefan Ultes, Tsung-Hsien Wen, Steve J. Young, and David Vandyke. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 1: Long Papers*, pages 438–449.

Igor Shalyminov, Arash Eshghi, and Oliver Lemon. 2017. Challenging neural dialogue models with natural data: Memory networks fail on incremental phenomena. In *Proceedings of the 21st Workshop on the Semantics and Pragmatics of Dialogue (SemDial 2017 - SaarDial)*, Barcelona.

Weiyan Shi, Tiancheng Zhao, and Zhou Yu. 2019. Unsupervised dialog structure learning. *Proceedings of NAACL*.

Stefan Ultes, Pawel Budzianowski, Iñigo Casanueva, Lina Maria Rojas-Barahona, Bo-Hsiang Tseng, Yen-Chen Wu, Steve J. Young, and Milica Gasic. 2018. Addressing objects and their relations: The conversational entity dialogue model. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue, Melbourne, Australia, July 12-14, 2018*, pages 273–283.

Vladimir Vlasov, Akela Drissner-Schmid, and Alan Nichol. 2018. Few-shot generalization across dialogue tasks. *CoRR*, abs/1811.11707.

Tsung-Hsien Wen, Yishu Miao, Phil Blunsom, and Steve J. Young. 2017. Latent intention dialogue models. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 3732–3741.

Jason D. Williams, Kavosh Asadi, and Geoffrey Zweig. 2017. Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 665–677.

Tiancheng Zhao and Maxine Eskénazi. 2018. Zero-shot dialog generation with cross-domain latent actions. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue, Melbourne, Australia, July 12-14, 2018*, pages 1–10.

Tiancheng Zhao, Kyusong Lee, and Maxine Eskénazi. 2018. Unsupervised discrete sentence representation learning for interpretable neural dialog generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 1098–1107.