Few-Shot Structured Policy Learning for Multi-Domain and Multi-Task Dialogues

Thibault Cordier[1,2] and **Tanguy Urvoy**[2] and **Fabrice Lefèvre**[1] and **Lina M. Rojas-Barahona**[2]
[1]LIA - Avignon University, France
[2]Orange Innovation, Lannion, France
thibault.cordier@alumni.univ-avignon.fr
fabrice.lefevre@univ-avignon.fr
{linamaria.rojasbarahona, tanguy.urvoy}@orange.com

Abstract

Reinforcement learning has been widely adopted to model *dialogue managers* in task-oriented dialogues. However, the user simulator provided by state-of-the-art dialogue frameworks are only rough approximations of human behaviour. The ability to learn from a small number of human interactions is hence crucial, especially on multi-domain and multi-task environments where the action space is large. We therefore propose to use *structured policies* to improve sample efficiency when learning on these kinds of environments. We also evaluate the impact of *learning from human vs simulated experts*. Among the different levels of structure that we tested, the graph neural networks (GNNs) show a remarkable superiority by reaching a success rate above 80% with only 50 dialogues, when learning from simulated experts. They also show superiority when learning from human experts, although a performance drop was observed, indicating a possible difficulty in capturing the variability of human strategies. We therefore suggest to concentrate future research efforts on bridging the gap between human data, simulators and automatic evaluators in dialogue frameworks.

1 Introduction

Multi-domain multi-task dialogue systems are designed to complete specific *tasks* in distinct *domains* such as finding and booking a hotel or a restaurant (Zhu et al., 2020). A domain is formally defined as a list of *slots* with their valid values. The most common task, the information-seeking task, is usually modelled as a slot-filling data-query problem in which the system requests constraints to the user and proposes items that fulfil those constraints.

The design of a *dialogue manager* (DMs) is costly: *hand-crafted* policies require a lot of engineering, pure *supervised learning* (or *behaviour cloning*) requires a lot of expert demonstrations, and pure *reinforcement learning* requires a lot of user interactions to converge. The simulators provided with frameworks, such as PyDial (Ultes et al., 2017) or ConvLab (Zhu et al., 2020), are only rough approximations of human behaviour and the ability to learn from a small number of human interactions remains crucial. This is especially true on multi-domain and multi-task environments where the action space is large (Gao et al., 2018).

A popular approach to reduce these costs is to wire some knowledge about the problem into the policy model, namely: *few shot learning* (Wang et al., 2020). In particular, structured policies like *graph neural networks* (GNNs) are known to be well suited to handle a variable number of slots and domains for the information-seeking task (Chen et al. 2018; Chen et al. 2020). In this paper, we explore structured policies based on GNN. A graph in a GNN is *fully connected* and *directed*. Each *node* represents a sub-policy associated with a slot, while a directed *edge* between two nodes represents a message passing.

For studying sample efficiency, we analyse the dialogue success rate of structured policies once trained in a supervised way from expert demonstrations. We consider two types of demonstrations: *human experts* extracted from the MultiWOZ dataset (Budzianowski et al., 2018), and *simulated experts* generated by letting the ConvLab's *hand-crafted* policy interact with a simulated user.

We perform large scale experiments. We study the impact of different levels of structure (see them in Figure 2) on policy success rate after a limited number of dialogue demonstrations. For each level of structure, we also compare two sources of demonstrations: simulated and human dialogues. We show a notable result: our structured policies are able to reach a success rate above 80% with only 50 when following a simulated expert in ConvLab. To the best of our knowledge there are not previous works that studied the impact of structure for dialogue policy in a few-shot setting.

Findings of the Association for Computational Linguistics: EACL 2023, pages 432–441
May 2-6, 2023 ©2023 Association for Computational Linguistics

Another important finding is that few-shot learning from human demonstrations is harder, producing a lower success rate. This can be explained first by the large variability of human strategies that is not covered by simulated users which stick to more repetitive – easy to learn – dialogue patterns. Another explanation could be an evaluation bias, simulated dialogues are more in line with artificial evaluators.

The remainder of this paper is structured as follows. We present the related work in Section 2. Section 3 presents the proposed GNNs from demonstrations. The experiments and evaluation are described in Sections 4 and 5 respectively. Finally, we conclude in Section 6.

2 Related Work

Few shot learning takes advantage of prior knowledge to avoid overloading the empirical risk minimiser when the number of available examples is small. In particular, prior knowledge can be used to constrain hypothesis space (i.e. model parameters) with parameter sharing or tying in order to reduce reliance on data acquisition and on data annotation (Wang et al., 2020).

Prior knowledge can be built into dialogue systems by imposing a structure in the neural network architecture. A first approach is to use *hierarchical reinforcement learning* that divides a main problem into several simpler sub-problems. We refer to Sutton et al. (1999) that introduces *semi-Markov decision process* using temporal abstraction and to Wen et al. (2020) that introduces *sub-Markov decision process* using state partition. In the scope of the paper, a *hierarchical policy* corresponds to a meta-controller that chooses to activate a domain and we have one sub-policy per domain (Budzianowski et al., 2017; Casanueva et al., 2018; Le et al., 2018).

In the same vein, *graph neural networks* (GNNs) have been explored in a wide range of domains because of their empirical success and their theoretical properties which explains its efficiency: the abilities of generalisation, stability and expressiveness (Garcia and Bruna, 2018). GNNs are suitable for applications where the data have a graph structure i.e where the graph outputs are supposed to be permutation-invariant or equivariant to the input features (Zhou et al., 2020; Wu et al., 2020).

In single-domain dialogue environments, this architecture has been adapted to model the DM in Chen et al. (2018) and Chen et al. (2020). They have shown that GNNs generalise between similar dialogue slots, manage a variable number of slots and transfer to different domains that perform similar tasks. We thus adopt in this work the *domain independent parametrisation* (DIP) (Wang et al., 2015), which standardises the slots representation into a common feature space.

In this work, as in Chen et al. (2018) and Chen et al. (2020), we propose to improve multi-domain covering by learning a generic policy based on GNN. But unlike them, (i) we use a multi-domain multi-task setting, in which several domains and tasks can be evoked in a dialogue; (ii) the *dialogue state tracker* (DST) output is not discarded when activating the domain; and (iii) we adapt the GNN structure to each domain by keeping the relevant nodes while sharing the edge's weights.

3 Structured Policies with Expert Demonstrations

In order to investigate the impact of structured policies with behaviour cloning in improving sample efficiency in multi-domain multi-task dialogue environments, we introduce the dialogue state and action spaces for structured policies and we present the different policies and the experts' nature.

3.1 Dialogue State / Action Representations

In multi-domain multi-task dialogues, the *domain* refers to the set of concepts and values speakers can talk about. Examples of domains are restaurants, attractions, hotels, trains, etc. A *dialogue act* is a predicate that refers to the performative actions of speakers in conversations (Austin, 1975). These actions are formalised as predicates like INFORM (*i.e.*, affirm) or REQUEST with slots or slot-values pairs as arguments. Examples of system actions are: REQUEST(food), or INFORM(address). These structured actions are used to frame a message to the user. We adopt here the multi-task setting as presented in CONVLAB (Zhu et al., 2020), in which a single dialogue can have the following tasks: (i) *find*, in which the system requests information in order to query a database and make an offer; (ii) *book*, in which the system requests information in order to book the item.

We adopt the DIP state and action representations, which are not reduced to a flat vector but to a set of sub-vectors: one corresponding to the domain parametrisation (or *slot-independent representation*), the others to the slots parametrisation

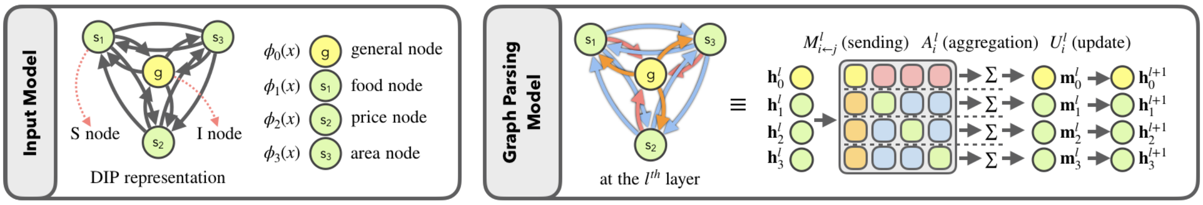

Input Model
s1
s3
g
s2
S node
I node
DIP representation
$\phi_0(x)$ general node
$\phi_1(x)$ food node
$\phi_2(x)$ price node
$\phi_3(x)$ area node
Graph Parsing Model
at the l^{th} layer
≡
$M^l_{i\leftarrow j}$ (sending) A^l_i (aggregation) U^l_i (update)
h^l_0 h^l_1 h^l_2 h^l_3
Σ
m^l_0 m^l_1 m^l_2 m^l_3
h^{l+1}_0 h^{l+1}_1 h^{l+1}_2 h^{l+1}_3

Figure 1: Structure of the input and graph parsing model in restaurant domain example. The input is a fully-connected graph with two kinds of nodes and three kinds of edges. The I-NODE are depicted in yellow; the S-NODE in green. The structured policy is described by successive graph convolutions composed of the shared weights $\mathbf{W}^l_{i,j}$.

(or *slot-dependent representations*). For any active domain, the input to the *slot-independent representation* is the concatenation of the previous *slot-independent* user and system actions (see examples of the output below, and a formal definition in Section 3.2), the number of entities fulfilling the user's constraints in the database, the booleans indicating if the dialogue is terminated and whether an offer has been found / booked. The output corresponds to action scores such as REQMORE, OFFER, BOOK, GREAT, etc. Regarding the *slot-dependent representation*, its input is composed of the previous *slot-dependent* user and system actions (see output below), the booleans indicating if a value is known and whether the slot is needed for the *find* / *book* tasks. Its output are actions scores such as INFORM, REQUEST and SELECT. The parameterisation used in CONVLAB does not depend on the probabilistic representation of the states, *i.e.* does not consider the uncertainty in the predictions made by the *natural language understanding* (NLU) module.

3.2 Graph Neural Network

Prior knowledge can be integrated in our models by constraining the layer structure imposing symmetries in the neural dialogue policies. Without prior knowledge, the standard structure used is the *feed-forward neural network* layer (FNN). This unconstrained structure does not assume any symmetry in the network.

Assuming that sub-policies associated with the slots are the same, a better alternative is to use the *graph neural network* layer (GNN). This structure assumes that the state and action representations have a graph structure that are identically parameterised by DIP. The GNN structure is a fully connected and directed graph, in which each *node* represents a sub-policy associated with a slot and a directed *edge* between two sub-policies represents a message passing. We identify two roles for sub-policies: the general node as I-NODE associated to the *slot-independent representation* and the slot nodes denoted as S-NODE associated to the *slot-dependent representations*. Both representations were introduced in Section 3.1. We also identify the relations: I2S for I-NODE to S-NODE, S2I and S2S respectively[1] (as presented in Figure 1).

We formally define the GNN structure as follows. Let n be the number of slots and L the number of layers. Let be x the dialogue state, $\mathbf{x}_0 = \phi_0(x)$, $\mathbf{h}^l_0$ $\forall l \in [0, L-1]$ and $\mathbf{y}_0$ be respectively the input, hidden and output I-NODE representations. Let the input, hidden and output S-NODES representations be respectively $\forall i \in [1, n]$, $\mathbf{x}_i = \phi_i(x)$, $\mathbf{h}^l_i$ $\forall l \in [0, L-1]$ and $\mathbf{y}_i$. First, the GNN transforms inputs:

$$\forall i \in [0, n], \quad \mathbf{h}^0_i = \sigma^0(\mathbf{W}^0_i \phi_i(\mathbf{x}) + \mathbf{b}^0_i) \qquad (1)$$

Then, at the l-th layer, it computes the hidden nodes representations by following message sending[2] (Eq. 2), message aggregation (Eq. 3) and representation update (Eq. 4). $\forall i, j \in [0, n]^2$:

$$\mathbf{m}^l_{i\leftarrow j} = M^l_{i\leftarrow j}(\mathbf{h}^{l-1}_j) = \mathbf{W}^l_{i,j}\mathbf{h}^{l-1}_j + \mathbf{b}^l_{i,j} \qquad (2)$$

$$\mathbf{m}^l_i = A^l_i(\mathbf{m}^l_{i\leftarrow *}) = \tfrac{1}{n}\textstyle\sum_{j=0}^n \mathbf{m}^l_{i\leftarrow j} \qquad (3)$$

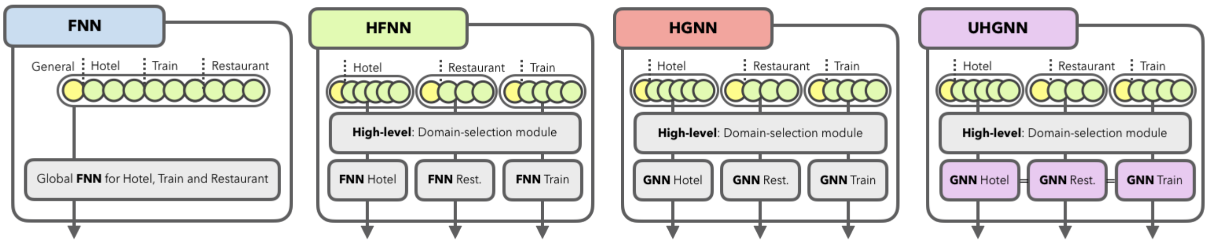
$$\mathbf{h}^l_i = U^l_i(\mathbf{m}^l_i) = \sigma^l(\mathbf{m}^l_i) \qquad (4)$$

The message sending function $M^l_{i\leftarrow j}$ is a linear transformation with bias. The message aggregation function A^l_i is the average pooling function. The representation update function U^l_i compute the new hidden representation with RELU activation function and dropout technique during learning stage. Finally, the GNN concatenates ($\oplus$ symbol) all final nodes representations and computes the policy function with the Softmax activation function.

$$\mathbf{y} = \sigma^L(\textstyle\bigoplus_{i=0}^n \mathbf{W}^L_i \mathbf{h}^{L-1}_i + \mathbf{b}^L_i) \qquad (5)$$

[1] We omit the I2I relation because there is only one I-node.

[2] The notation $i \leftarrow j$ denotes a message sending from slot j to slot i. It also corresponds to the directed relation between the slots j and i. The notation $i \leftarrow *$ denotes all messages sending to slot i.



(a) FNN. (b) HFNN. (c) HGNN. (d) UHGNN.

Figure 2: Policy and input data structures. Different levels of structure are presented from classical *feed-forward neural network* (FNN) to *graph neural network* (GNN). The prefix H- corresponds to a hierarchical policy and UH- corresponds to a unique sub-policy for all domains. For a FNN layer, the input data is the concatenation of all DIP slot representations. For a GNN layer, the input keeps its structure.

3.3 Structured Policies

We propose a wide range of dialogue policies to study the impact of the structure in sample efficiency. An ablation study progressively adds some notion of hierarchy to FNNs to approximate the structure of GNNs. Similarly, we analyse the advantage of sharing a generic GNN among several domains versus specialising a GNN to each domain. Therefore, we propose from the least to the most constrained:

- **Feed-forward Neural Network** (**FNN**) that is a classical feed-forward neural network with DIP parametrisation (Figure 2a).
- **Hierarchy of Feed-forward Neural Networks** (**HFNN**) that is a hierarchical policy with hand-crafted domain-selection and FNNs for each domain. Each domain has one corresponding FNN model (Figure 2b).
- **Hierarchy of Graph Neural Networks** (**HGNN**) that is a hierarchical policy with hand-crafted domain-selection and GNNs. Each domain has one corresponding GNN model (Figure 2c).
- **Hierarchy with Unique Graph Neural Network** (**UHGNN**) that is a HGNN with a unique GNN for all domains. Each domain shares the same GNN model (Figure 2d).

3.4 The Expert's Nature

Since our goal is to learn on observed demonstrations delivered by an expert, we propose to focus on policies that learn from both simulated and human experts. For this purpose, we use the dataset MULTIWOZ (Budzianowski et al., 2018) to follow human experts and the hand-crafted policy of CONVLAB (Zhu et al., 2020) as the simulated expert.

Human expert The MULTIWOZ dataset is a large annotated and open-sourced collection of human-human chats that covers multiple domains and tasks. Nearly 10k dialogues have been collected by a *Wizard-of-Oz* set-up at relatively low cost and with a small time effort. However, different versions of this dataset corrected and improved the annotations (Eric et al., 2020; Zang et al., 2020; Han et al., 2021; Ye et al., 2021). In this work, we use the MULTIWOZ dataset integrated in CONVLAB with extended user dialogue act annotations.

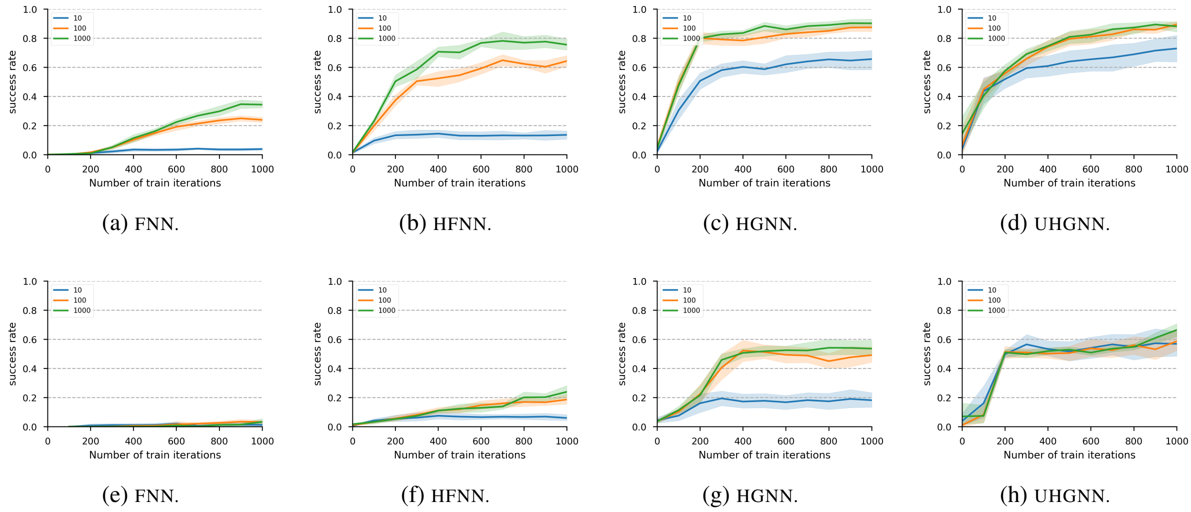
Simulated expert The CONVLAB framework has been proposed to automatically build, train and evaluate multi-domain multi-task oriented dialogue systems based on MULTIWOZ features. It implements both hand-crafted simulated user and policy. The latter has been shown to be nearly the optimal policy according to the CONVLAB evaluation setup of (Takanobu et al., 2020). Therefore we use it as the simulated expert.

4 Experiments

In this section we explain the experimental setup, the proposed models and the evaluation metrics.

4.1 Experiment Setup

We performed an ablation study by gradually adding different levels of structure from a baseline FNN to the proposed GNN (Subsection 4.2). On the one hand, we analyse the learning efficiency of our models in small training steps. On the other hand, we compare their generalisation ability in few shot learning.



(a) FNN. (b) HFNN. (c) HGNN. (d) UHGNN.

(e) FNN. (f) HFNN. (g) HGNN. (h) UHGNN.

Figure 3: Dialogue manager evaluation with simulated users. We present the success rate on 10 / 100 / 1 000 training dialogues as a function of the number of gradient descent steps in a short training scenario. Learning is based on simulated experts (Figures (a) up to (d)) or on human experts (Figures (e) up to (h)). The line plot represents the mean and the coloured area represents the 95% confidence interval over a sample of 10 runs.

To analyse the learning efficiency, we measure performance with respect to the number of gradient descent steps up to 1 000 iterations with a step size of 100 iterations. We compare learning curves based on randomly chosen 10, 100 and 1 000 training dialogues[3]. We also measure performance as a function of the number of training dialogues available (randomly chosen) namely 10, 50, 100, 500 and 1000 when each training is performed up to 10 000 gradient descent steps. All the experiments were run on CONVLAB, restarted 10 times with random initialisation and the results estimated on 500 new dialogues.

4.2 Models

The FNN models have two hidden layers, both with 128 neurons. The GNN models have one first hidden layer with 64 neurons for both nodes (S-NODE and I-NODE). Then the second hidden layer is composed of 64 neurons for each relation (S2S, S2I and I2S). For training stage, we use the ADAM optimiser with a learning rate $lr = 0.001$, a dropout rate $dr = 0.1$ and a batch size $bs = 64$.

4.3 Metrics

We evaluate the performance of the policies for all tasks as in CONVLAB. Precision, recall and F-score, namely the **inform rates**, are used for the *find* task. Inform recall evaluates whether all the requested information has been informed while inform precision evaluates whether only the requested information has been informed. For the *book* task, the accuracy, namely the **book rate**, is used. It assesses whether the offered entity meets all the constraints specified in the user goal. The dialogue is marked as **successful** if and only if both inform recall and book rate are equal to 1. The dialogue is considered **completed** if it is successful from the user's point of view[4].

5 Evaluation

First, we evaluate the dialogue manager performance when talking to a simulated user. Second, we evaluate the learned policies within the entire dialogue system both with simulated and with real users. The evaluations have been done within CONVLAB.

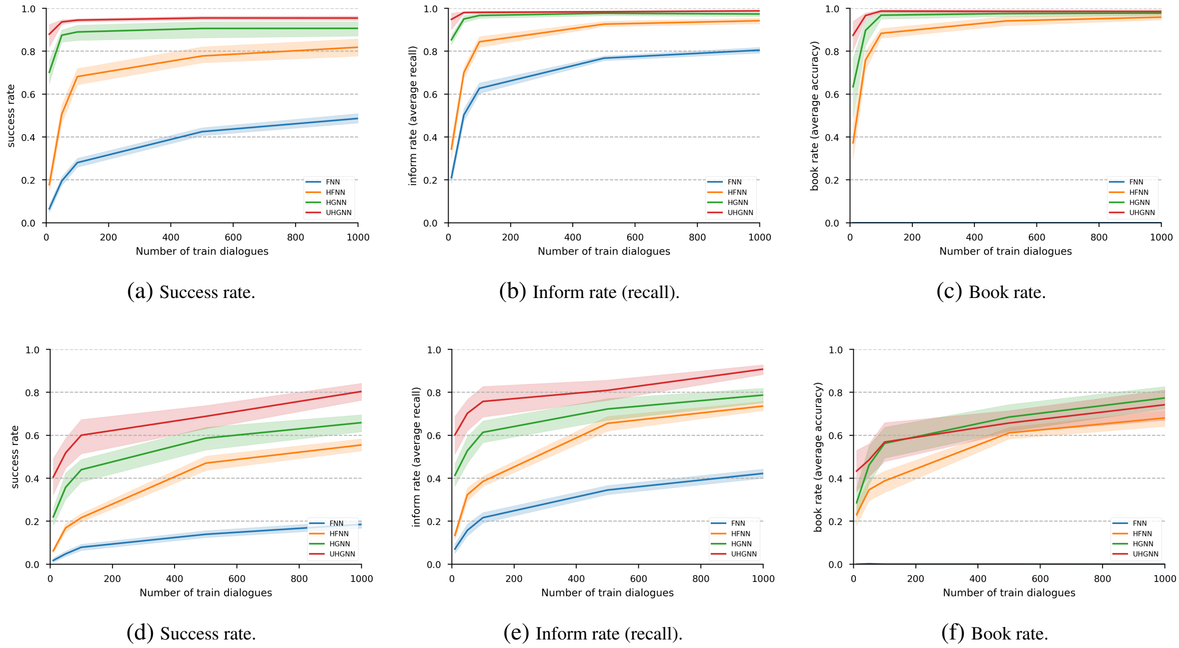
5.1 Dialogue Manager Evaluation

We analyse our models on the learning efficiency in small training steps and on the ability to generalise in a few-shot setting.

Efficiency We report in Figure 3 the results of the ablation study showing the ability of the models to succeed in a short training stage. First, when

[3] These values were chosen arbitrarily to give us an insight into the impact of the number of dialogues on the performance.

[4] A dialogue can be completed without being successful if the information provided is not the one objectively expected by the simulator.



(a) Success rate.

(b) Inform rate (recall).

(c) Book rate.

(d) Success rate.

(e) Inform rate (recall).

(f) Book rate.

Figure 4: Dialogue manager evaluation with simulated user presenting the success rate based on 10 000 training iterations as a function of the number of training dialogues in a long learning scenario. Learning is based on a simulated expert (Figures (a), (b) and (c)) or human experts (Figures (d), (e) and (f)). The line plot represents the mean and the coloured area represents the 95% confidence interval over a sample of 10 runs.

learning from simulated demonstrations we notice in Figure 3a that the baseline (FNN) needs a large number of training dialogues (more than 100) to achieve a moderate performance (less than 40%). We show then in Figure 3b that hierarchical networks (HFNN) do improve learning efficiency up to 60% with 100 dialogues, up to 80% with 1 000 dialogues. Finally we show that graph neural network (HGNN in Figure 3c) and generic policy (UHGNN in Figure 3d) drastically improve the efficiency with few dialogues, more than 60% with 10 dialogues, and achieve remarkable performance above 80% with only 100 dialogues in 1 000 training steps. These observations confirm that hierarchical and generic GNNs allow efficient learning and collaborative gradient update in a short training stage.

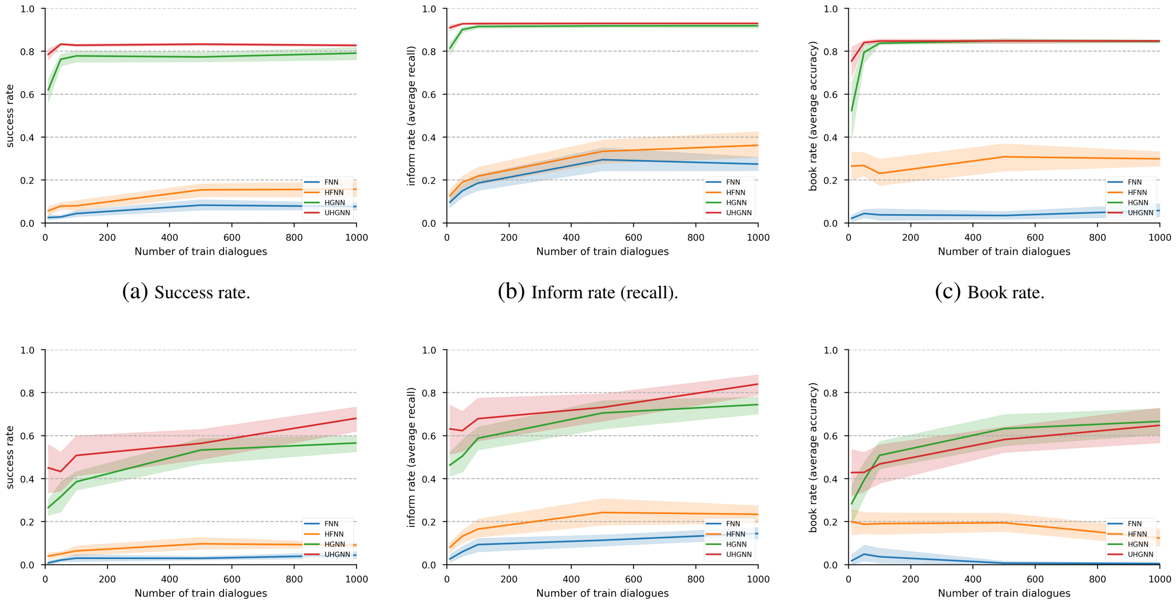
Although standard or hierarchical policies (FNN in Figure 3e and HFNN in Figure 3f) are less efficient when learning from human demonstrations, they are still above baselines. It is worth noting that structured or generic GNN policies HGNN in Figure 3g and UHGNN in Figure 3h are able to reach more than 50% success rate.

Few-Shot We extended the ablation study in a few-shot scenario focusing on the ability of the models to succeed on specific dialogue tasks as reported in Figure 4. In particular, we show the success rate in Figure 4a, the inform rate (recall) in Figure 4b and the book rate in Figure 4c when using simulated demonstrations and respectively in Figure 4d, Figure 4e and Figure 4f when using human demonstrations. The more structured the model, the greater the learning efficiency and the greater the data efficiency. Likewise, we notice that learning is more data-intensive when imitating human strategies. It appears that the booking task is more difficult to perform according to human demonstrations (when comparing Figure 4c and Figure 4f) or using a flat architecture (FNN gets null results). We therefore foresee that more high quality data is needed to learn on human dialogues.

5.2 Dialogue System Evaluation

We continue our analysis on the robustness of the studied models with the entire dialogue system facing both simulated and human users. The dialogue system utilises a BERT NLU (Devlin et al., 2019) and a hand-crafted NLG.

Simulated User Evaluation As in the previous subsection, we study the robustness of the models in a few-shot scenario as presented in Figure 5.



(a) Success rate.

(b) Inform rate (recall).

(c) Book rate.

(d) Success rate.

(e) Inform rate (recall).

(f) Book rate.

Figure 5: Dialogue system performance with simulated user based on 10 000 training iterations as a function of the number of training dialogues in a long training scenario. The supervised DM is based on simulated demonstrations (Figures (a),(b),(c)) or on human demonstrations (Figures (d),(e),(f)). The line plot represents the mean and the coloured area represents the 95% confidence interval over a sample of 10 runs.

We observe that FNN (in blue) and HFNN (in orange) learning is collapsing when using simulated dialogues (see Figures 5a, 5b and 5c). On the opposite, HGNN (in green) and UHGNN (in red) performance appears more stable in the entire dialogue system even when using real dialogues (see Figures 5d, 5e and 5f). Therefore, these results confirm that behaviour cloning is easier from simulated than human experts. As observed before in Subsection 5.2, this can be explained by an large variability of human strategies (hence the need for more data to improve performance). Another explanation is that simulated dialogues are more in line with the artificial evaluator provided in the ConvLab. In addition, it is important not to neglect the side effects of cascading errors due to successive NLU, DST, DM and NLG modules. In particular, the NLU BERT proposed by ConvLab was pre-trained and evaluated on 7 372 user utterances with 14% of errors (F1 86.4%, precision 85.1%, recall 87.8%). This problem can therefore be exacerbated by cascading human errors, as confirmed in the next paragraph.

Finally, we present a detailed comparison table with the best structured policies UHGNN trained on simulated dialogues of ConvLab noted MLE-UHGNN-HDC (HDC for *hand-crafted policy*) and trained on real dialogues of MultiWOZ noted MLE-UHGNN-MW and the baselines of ConvLab (see Table 1). In particular, the *maximum likelihood estimator* (MLE) proposed by ConvLab is an implementation of FNN model trained on MultiWOZ corpus in a very long training scenario (multiple passes on all $10k$ dialogues)[5]. Our models show competitive results against ConvLab's baselines, confirming that the structured with supervised learning in few-shot settings is adapted to address the difficulties in multi-task multi-domain dialogues.

Human Evaluation We organised preliminary evaluation sessions, in which volunteers were invited to chat on-line with three dialogue systems that were randomly assigned[6]. Subjects do not know which system they are evaluating. Each sys-

[5] Another difference is that our models returns one unique action per turn instead of a group of actions.

[6] Crowdsourcing was not used because of ethical concerns regarding the work conditions of collaborators. Volunteers from our research institution were invited to participate and they were aware of the scientific motivations behind the evaluation. In this sense, they were motivated to participate without any economic reward implying no pressure and without knowing the nature of the models they were evaluating, avoiding in this way any evaluation bias.

Configuration	Avg Turn (succ/all)	Inform rate (%) Prec. / Rec. / F1	Book Rate (%)	Complete Rate (%)		Success Rate (%)	
Dialogue Management							
HDC	10.6/10.6	87.2 / 98.6 / 90.9	98.6	**97.9**	-	**97.3**	-
MLE-UHGNN-HDC (ours)	12.8/13.0	95.3 / 98.8 / 96.4	98.5	97.3	(-0.6)	95.4	(-1.9)
MLE-UHGNN-MW (ours)	16.5/20.7	94.3 / 90.7 / 91.6	76.7	81.4	(-16.5)	81.0	(-6.3)
Dialogue System (BERT NLU + hand-crafted NLG)							
HDC	11.4/12.0	82.8 / 94.1 / 86.2	91.5	92.7	-	83.8	-
HDC†	11.6/12.3	79.7 / 92.6 / 83.5	91.1	90.5	(-2.2)	81.3	(-2.5)
MLE†	12.1/24.1	62.8 / 69.8 / 62.9	17.6	42.7	(-50.0)	35.9	(-47.9)
PG†	11.0/25.3	57.4 / 63.7 / 56.9	17.4	37.4	(-55.3)	31.7	(-52.1)
GDPL†	11.5/21.3	64.5 / 73.8 / 65.6	20.1	49.4	(-43.3)	38.4	(-45.4)
PPO†	13.1/17.8	69.4 / 85.8 / 74.1	86.6	75.5	(-17.2)	71.7	(-12.1)
MLE-UHGNN-HDC (ours)	14.0/15.4	89.3 / 93.0 / 90.2	84.8	**90.0**	(-2.7)	**82.7**	(-1.1)
MLE-UHGNN-MW (ours)	17.0/23.0	84.0 / 87.6 / 84.5	64.8	72.1	(-20.6)	68.1	(-15.7)

Table 1: Dialogue manager and system evaluations with simulated users. When evaluating the dialogue manager, the simulated user passes directly dialogue acts and vice-versa. Our tested configurations are evaluated and averaged on 10 run each with 250 dialogues. Configurations with † are taken from the GitHub of ConvLab.

Dialogue System (BERT NLU + Rule NLG)	Avg Turn	Satisfaction Rate (%)	Nb of Dial.
HDC	22.6	**92.6 ± 9.87**	27
MLE-UHGNN-HDC	25.6	50.0 ± 14.8	44
MLE-UHGNN-MW	17.3	36.7 ± 17.2	30

Table 2: Dialogue system evaluation with real users with a 95% confidence level for satisfaction rate.

tem has a different DM model: HDC (*hand-crafted policy*), MLE-UHGNN-HDC (based on simulated demonstrations with HDC policy) and MLE-UHGNN-MW (based on MultiWOZ demonstrations) combined with the BERT NLU and the hand-crafted NLG provided by ConvLab. At the end of the chat, evaluators were asked whether or not they reach the goal and were satisfied with the performance of the system. The **satisfaction rate** is then the proportion of dialogues in which the system solved the task at the end of the dialogue according to the human evaluator. We reported results on roughly 30 dialogues for each method. The results of this experimentation are presented in Table 2. Although test is small-sized and not highly statistically significant, these preliminary results are disconcerting with respect to the simulated ones. The HDC does very well whereas MLE-UHGNN-HDC gets by in half the cases, MLE-UHGNN-MW fails in most cases.

These results can be explained by the limitations of the NLU facing impatient evaluators, short and ambiguous sentences where the active domain is unclear (as in this example of the user saying "What is the name?") or typographical errors. Moreover, it is important to underline that ConvLab does not natively propose the management of uncertainties in the state representation which can strongly restrict the performance of the learning methods in noisy environments. Another limitation is that the HDC is more adapted to conventional dialogues whereas MLE-UHGNNs were trained only on winning dialogues. This implies that learning methods are more sensitive to dialogues that break out of the learned patterns. Similarly, the strategies of simulated and real users do not seem to be well aligned with each other and even more strongly with the expectations of human evaluators.

6 Conclusion

We investigated in this work the impact of policy structure and experts on success rate in few-shot learning for multi-domain multi-task dialogues. Promising results were obtained: hierarchical and generic GNN policies are able to achieve remarkable performance with few dialogues and few training iterations when following a simulated expert. This confirms the growing interest for these neural structures. We also present an important finding: the policy performance degrades in few-shot learning when using human demonstrations. This fact questions the alignment between dialogue evaluators and human strategies in state-of-the-art dialogue frameworks.

Limitations

The reduced performance when learning from human experts suggests that we shall concentrate the efforts in bridging the gap between automatic evaluators and high-quality human-human datasets. We also devise the use of *curriculum learning* (Bengio et al., 2009) strategies: starting from simple – simulated – dialogues then adding progressively more complex, human dialogues demonstrations.

It is also necessary to analyse the impact of GNN policies with neural NLU/NLG modules to study how to integrate such structures in end-to-end architectures.

We point out some limitations of CONVLAB. The detection of the active domain is sensitive to the output of the NLU and thus sensitive to ambiguous statements. Data representation restricts the DST to a deterministic view and must be adapted to a probabilistic representation to capture the uncertainties in the user's input. Similarly, it may be worthwhile to improve the action space by adding more possibilities for human users, for instance to CONFIRM or DENY in a more flexible way.

Finally, the human evaluation was performed on a small scale and on models trained in a context with few training iterations. A more in-depth or supervised study could shed more light on the raised issues.

Acknowledgements

This work was performed using HPC resources from GENCI-IDRIS (Grant 20XX-[AD011011407]).

References

John Langshaw Austin. 1975. *How to do things with words*. Oxford university press.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48.

Paweł Budzianowski, Stefan Ultes, Pei-Hao Su, Nikola Mrkšić, Tsung-Hsien Wen, Inigo Casanueva, Lina Rojas-Barahona, and Milica Gašić. 2017. Subdomain modelling for dialogue management with hierarchical reinforcement learning. *arXiv preprint arXiv:1706.06210*.

Pawel Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gasic. 2018. Multiwoz-a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. In *EMNLP*.

Iñigo Casanueva, Paweł Budzianowski, Pei-Hao Su, Stefan Ultes, Lina M Rojas Barahona, Bo-Hsiang Tseng, and Milica Gasic. 2018. Feudal reinforcement learning for dialogue management in large domains. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 714–719.

Lu Chen, Bowen Tan, Sishan Long, and Kai Yu. 2018. Structured dialogue policy with graph neural networks. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1257–1268.

Zhi Chen, Lu Chen, Xiaoyuan Liu, and Kai Yu. 2020. Distributed structured actor-critic reinforcement learning for universal dialogue management. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2400–2411.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyang Gao, and Dilek Hakkani-Tür. 2020. Multiwoz 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. In *Proceedings of the 12th Language Resources and Evaluation Conference*, page 422–428. Marseille, France. European Language Resources Association.

Jianfeng Gao, Michel Galley, and Lihong Li. 2018. Neural approaches to conversational ai. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 1371–1374.

Victor Garcia and Joan Bruna. 2018. Few-shot learning with graph neural networks. In *6th International Conference on Learning Representations, ICLR 2018*.

Ting Han, Ximing Liu, Ryuichi Takanabu, Yixin Lian, Chongxuan Huang, Dazhen Wan, Wei Peng, and Minlie Huang. 2021. Multiwoz 2.3: A multi-domain task-oriented dialogue dataset enhanced with annotation corrections and co-reference annotation. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 206–218. Springer.

Hoang Le, Nan Jiang, Alekh Agarwal, Miroslav Dudik, Yisong Yue, and Hal Daumé III. 2018. Hierarchical imitation and reinforcement learning. In *International conference on machine learning*, pages 2917–2926. PMLR.

Richard S Sutton, Doina Precup, and Satinder Singh. 1999. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2):181–211.

Ryuichi Takanobu, Qi Zhu, Jinchao Li, Baolin Peng, Jianfeng Gao, and Minlie Huang. 2020. Is your goal-oriented dialog model performing really well? empirical analysis of system-wise evaluation. In *Proceedings of the 21th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 297–310.

Stefan Ultes, Lina M Rojas Barahona, Pei-Hao Su, David Vandyke, Dongho Kim, Iñigo Casanueva, Paweł Budzianowski, Nikola Mrkšić, Tsung-Hsien Wen, and Milica Gasic. 2017. Pydial: A multi-domain statistical dialogue system toolkit. In *Proceedings of ACL 2017, System Demonstrations*, pages 73–78.

Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. 2020. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys (csur)*, 53(3):1–34.

Zhuoran Wang, Tsung-Hsien Wen, Pei-Hao Su, and Yannis Stylianou. 2015. Learning domain-independent dialogue policies via ontology parameterisation. In *Proceedings of the 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 412–416.

Zheng Wen, Doina Precup, Morteza Ibrahimi, Andre Barreto, Benjamin Van Roy, and Satinder Singh. 2020. On efficiency in hierarchical reinforcement learning. *Advances in Neural Information Processing Systems*, 33:6708–6718.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. 2020. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems*, 32(1):4–24.

Fanghua Ye, Jarana Manotumruksa, and Emine Yilmaz. 2021. Multiwoz 2.4: A multi-domain task-oriented dialogue dataset with essential annotation corrections to improve state tracking evaluation. *arXiv preprint arXiv:2104.00773*.

Xiaoxue Zang, Abhinav Rastogi, Srinivas Sunkara, Raghav Gupta, Jianguo Zhang, and Jindong Chen. 2020. MultiWOZ 2.2 : A dialogue dataset with additional annotation corrections and state tracking baselines. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 109–117, Online. Association for Computational Linguistics.

Jie Zhou, Ganqu Cui, Shengding Hu, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2020. Graph neural networks: A review of methods and applications. *AI Open*, 1:57–81.

Qi Zhu, Zheng Zhang, Yan Fang, Xiang Li, Ryuichi Takanobu, Jinchao Li, Baolin Peng, Jianfeng Gao, Xiaoyan Zhu, and Minlie Huang. 2020. Convlab-2: An open-source toolkit for building, evaluating, and diagnosing dialogue systems. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 142–149.