# End-to-End Learning of Task-Oriented Dialogs

**Bing Liu, Ian Lane**
Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213, USA
liubing@cmu.edu, lane@cmu.edu

## Abstract

In this thesis proposal, we address the limitations of conventional pipeline design of task-oriented dialog systems and propose end-to-end learning solutions. We design neural network based dialog system that is able to robustly track dialog state, interface with knowledge bases, and incorporate structured query results into system responses to successfully complete task-oriented dialog. In learning such neural network based dialog systems, we propose hybrid offline training and online interactive learning methods. We introduce a multi-task learning method in pre-training the dialog agent in a supervised manner using task-oriented dialog corpora. The supervised training agent can further be improved via interacting with users and learning online from user demonstration and feedback with imitation and reinforcement learning. In addressing the sample efficiency issue with online policy learning, we further propose a method by combining the learning-from-user and learning-from-simulation approaches to improve the online interactive learning efficiency.

## 1 Introduction

Dialog systems, also known as conversational agents or chatbots, are playing an increasingly important role in today's business and social life. People communicate with a dialog system in natural language form, via either textual or auditory input, for entertainment and for completing daily tasks. Dialog systems can be generally divided into chit-chat systems and task-oriented dialog systems based on the nature of conversation. Comparing to chit-chat systems that are designed to engage users and provide mental support, task-oriented dialog systems are designed to assist user to complete a particular task by understanding requests from users and providing relevant information. Such systems usually involve retrieving information from external resources and reasoning over multiple dialog turns. This thesis work focuses on task-oriented dialog systems.

Conventional task-oriented dialog systems have a complex pipeline (Raux et al., 2005; Young et al., 2013) consisting of independently developed and modularly connected components for spoken language understanding (SLU) (Sarikaya et al., 2014; Mesnil et al., 2015; Chen et al., 2016), dialog state tracking (DST) (Henderson et al., 2014; Mrkšić et al., 2016; Lee and Stent, 2016), and dialog policy learning (Gasic and Young, 2014; Su et al., 2016). Such pipeline system design has a number of limitations. Firstly, credit assignment in such pipeline systems can be challenging, as errors made in upper stream modules may propagate and be amplified in downstream components. Moreover, each component in the pipeline is ideally re-trained as preceding components are updated, so that we have inputs similar to the training examples at run-time. This domino effect causes several issues in practice.

We address the limitations of pipeline dialog systems and propose end-to-end learning solutions. The proposed model is capable of robustly tracking dialog state, interfacing with knowledge bases, and incorporating structured query results into system responses to successfully complete task-oriented dialog. With each functioning unit being modeled by a neural network and connected via differentiable operations, the entire system can be optimized end-to-end.

In learning such neural network based dialog model, we propose hybrid offline training and online interactive learning methods. We first let the agent to learn from human-human conversations with offline supervised training. We then improve the agent further by letting it to interact with users and learn from user demonstrations and feedback with imitation and reinforcement learning. In ad-



*Proceedings of NAACL-HLT 2018: Student Research Workshop*, pages 67–73
New Orleans, Louisiana, June 2 - 4, 2018. 

dressing the sample efficiency issue with online policy learning via interacting with real users, we further propose a learning method by combining learning-from-user and learning-from-simulation approaches. We conduct empirical study with both automatic system evaluation and human user evaluation. Experimental results show that our proposed model can robustly track dialog state and produce reasonable system responses. Our proposed learning methods also lead to promising improvement on dialog task success rate and human user ratings.

## 2 Related Work

### 2.1 Task-Oriented Dialog Systems

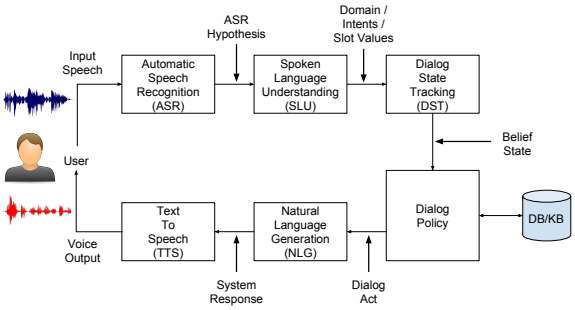


Figure 1: Pipeline architecture for spoken dialog systems

Figure 1 shows a typical pipeline architecture of task-oriented spoken dialog system. Transcriptions of user's speech are firstly passed to the SLU module, where the user's intention and other key information are extracted. This information is then formatted as the input to DST, which maintains the current state of the dialog. Outputs of DST are passed to the dialog policy module, which produces a dialog act based on the facts or entities retrieved from external resources (such as a database or a knowledge base). The dialog act emitted by the dialog policy module serves as the input to the NLG, through which a natural language format system response is generated. In this thesis work, we propose end-to-end solutions that focus on three core components of task-oriented dialog system: SLU, DST, and dialog policy.

### 2.2 End-to-End Dialog Models

Conventional task-oriented dialog systems use a pipeline design by connecting the above core system components together. Such pipeline system design makes it hard to track source of errors and align system optimization targets. To ameliorate these limitations, researchers have recently started exploring end-to-end solutions for task-oriented dialogs. Wen et al. (Wen et al., 2017) proposed an end-to-end trainable neural dialog model with modularly connected system components for SLU, DST, and dialog policy. Although these system components are end-to-end connected, they are trained separately. It is not clear whether common features and representations for different tasks can be effectively shared during the dialog model training. Moreover, the system is trained with supervised learning on fixed dialog corpora, and thus may not generalize well to unseen dialog states when interacting with users.

Bordes and Weston (Bordes and Weston, 2017) proposed a task-oriented dialog model from a machine reading and reasoning approach. They used an RNN to encode the dialog state and applied end-to-end memory networks to learn it. In the same line of research, people explored using query-regression networks (Seo et al., 2016), gated memory networks (Liu and Perez, 2017), and copy-augmented networks (Eric and Manning, 2017) to learn the dialog state RNN. Similar to (Wen et al., 2017), these systems are trained on fixed sets of simulated and human-machine dialog corpora, and thus are not capable to learn interactively from users. The knowledge base information is pulled offline based on existing dialog corpus. It is unknown whether the reasoning capability achieved in offline model training can generalize well to online user interactions.

Williams et al. (Williams et al., 2017) proposed a hybrid code network for task-oriented dialog that can be trained with supervised and reinforcement learning (RL). Li et al. (Li et al., 2017) and Dhingra et al. (Dhingra et al., 2017) also proposed end-to-end task-oriented dialog models that can be trained with hybrid supervised learning and RL. These systems apply RL directly on supervised pre-training models, without discussing the potential issue with dialog state distribution mismatch between supervised training and interactive learning. Moreover, current end-to-end dialog models are mostly trained and evaluated against user simulators. Ideally, RL based dialog learning should be performed with human users by collecting real user feedback. In interactive learning with human users, online learning efficiency becomes a critical factor. This sample efficiency issue with online policy learning is not addressed in these works.

## 3 End-to-End Dialog Learning

### 3.1 Proposed Dialog Learning Framework

Task-oriented dialog system assists human user to complete tasks by conducting multi-turn conversations. From a learning point of view, the dialog agent learns to act by interacting with users and trying to maximize long-term success or an expected reward. Ideally, the dialog agent should not only be able to passively receive signals from the environment (i.e. the user) and learn to act on it, but also to be able to understand the dynamics of the environment and predict the changes of the environment state. This is also how we human beings learn from the world. We design our dialog learning system following the same philosophy.

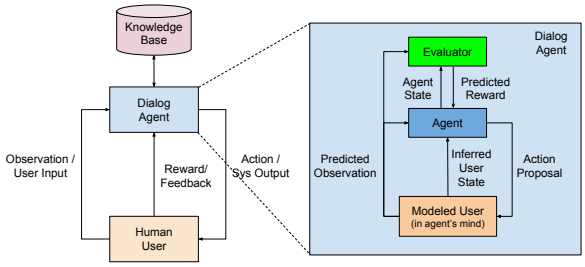


Figure 2: Proposed task-oriented dialog learning framework

Figure 2 shows our proposed learning framework for task-oriented dialog. The dialog agent interacts with user in natural language format and improves itself with the received user feedback. The dialog agent also learns to interface with external resources, such as a knowledge base or a database, so as to be able to provide responses to user that are based on the facts in the real world. Inside the dialog agent, the agent learns to model the user dynamics by predicting their behaviors in conversations. Such modeled user or simulated user in the agent's mind can help the agent to simulate dialogs that mimic the conversations between the agent and a real user. By "imagining" such conversations and learning from it, the agent can potentially learn more effectively and reduce the number of learning cycles with real users.

### 3.2 End-to-End System Architecture

Figure 3 shows the architecture of our proposed end-to-end task-oriented dialog system (Liu et al., 2017). We use a hierarchical LSTM to model a dialog with multiple turns. The lower level LSTM, which we refer to as the utterance-level LSTM, is used to encode the user utterance. The higher-level LSTM, which we refer to as the dialog-level LSTM, is used to model a dialog over a sequence of turns. User input to the system in natural language format is encoded in a continuous vector form via the utterance-level LSTM. The LSTM outputs can be fed to SLU decoders such as an intent classifier and a slot filler. We may use such SLU module outputs as input to the state tracker. Alternatively, we may directly use the continuous representation of user's utterance without passing it through a semantic decoder. The encoded user utterance, together with the encoding of the previous system turn, is connected to the dialog-level LSTM. State of this dialog-level LSTM maintains a continuous representation of the dialog state. Based on this state, the belief tracker generates a probability distribution over candidate values for each of the tracked goal slots. A query command can then be formulated with the belief tracking outputs and sent to a database to retrieve requested information. Finally, the system produces an action by combining information from the dialog state, the belief tracking outputs, and the encoding of the query results. This system dialog action, together with the belief tracking output and the query results, is used to generate the final natural language system response via a natural language generator. All system components are connected via differentiable operations, and the entire system (SLU, DST, and policy) can thus be optimized end-to-end.

## 4 Learning from Dialog Corpora

In this section, we describe our proposed corpus-based supervised training methods for task-oriented dialog. We first explain our supervised learning models for SLU, and then explain how these models are extended for dialog modeling.

### 4.1 SLU and Utterance Modeling

We first describe our proposed utterance representation learning method by jointly optimizing the two core SLU tasks, intent detection and slot filling. Intent detection and slot filling are usually handled separately by different models, without effectively utilizing features and representations that can be shared between the two tasks. We propose to jointly optimize the two SLU tasks with recurrent neural networks. A bidirectional LSTM reader is used to encode the user utterance. LSTM

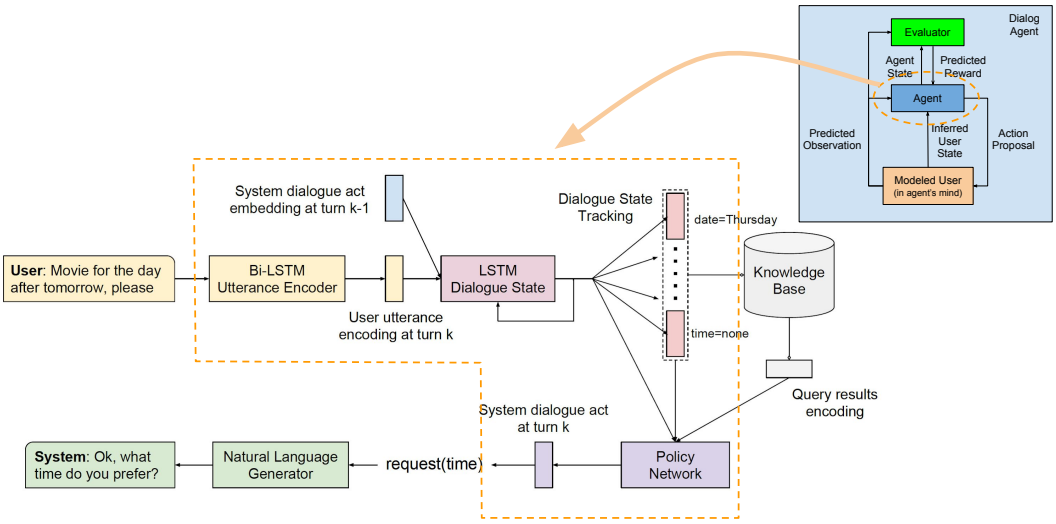


Figure 3: End-to-end task-oriented dialog system architecture

state output at each time step is used for slot label prediction for each word in the utterance. A weighted average of these LSTM state outputs is further used as the representation of the utterance for user intent prediction. The model objective function is a linear interpolation of the cross-entropy losses for intent and slot label predictions. Experiment results (Liu and Lane, 2016a,b) on ATIS SLU corpus show that the proposed joint training model achieves state-of-the-art intent detection accuracy and slot filling F1 scores. The joint training model also outperforms the independent training models on both tasks.

### 4.2 Dialog Modeling with Hierarchical LSTM

The above described SLU models operate on utterance or turn level. In dialog learning, we expect the system to be able to reason over the dialog context, which covers information over a sequence of past dialog turns. We extend the LSTM based SLU models by adding a higher level LSTM on top to model dialog context over multiple turns (Liu and Lane, 2017a). The lower level LSTM uses the same bidirectional LSTM design as in section 4.1 to encode natural language utterance. These encoded utterance representation at each turn serve as the input to the upper level LSTM that models dialog context. Based on the dialog state encoded in the dialog-level LSTM, the model produces a probability distribution over candidate values for each of the tracked goal slots. This serves the functionality of a dialog state tracker. Furthermore, the model predicts the system dialog act or a delexicalised system response based on the current dialog state. This can be seen as learning a supervised dialog policy by following the expert actions via behavior cloning.

In supervised model training, we optimize the parameter set $\theta$ to minimize the cross-entropy losses for dialog state tracking and system action prediction:

$$\min_{\theta} \sum_{k=1}^{K} -\Big[ \sum_{m=1}^{M} \lambda_{l^m} \log P(l_k^{m*} | \mathbf{U}_{\leq k}, \mathbf{A}_{<k}, \mathbf{E}_{<k}; \theta) + \lambda_a \log P(a_k^* | \mathbf{U}_{\leq k}, \mathbf{A}_{<k}, \mathbf{E}_{\leq k}; \theta) \Big] \quad (1)$$

where $\lambda$s are the linear interpolation weights for the cost of each system output. $l_k^{m*}$ and $a_k^*$ are the ground truth labels for goal slots and system action at the $k$th turn. In the evaluation (Liu and Lane, 2017b) on DSTC2 dialog datset, the proposed model achieves near state-of-the-art performance in dialog goal tracking accuracy. Moreover, the proposed model demonstrates promising results in producing appropriate system responses, outperforming prior end-to-end neural network models using per-response accuracy evaluation metric.

## 5 Learning from Human Demonstration

The supervised training dialog model described in section 4 performs well in offline evaluation setting on fixed dialog corpora. The same model performance may not generalize well to unseen dialog states when the system interacts with users. We propose interactive dialog learning methods with imitation learning to address this issue.

### 5.1 Imitation Learning with Human Teaching

Supervised learning succeeds when training and test data distributions match. During dialog interaction with users, any mistake made by the system or any deviation in the user's behavior may lead it to a different state distribution that the supervised training agent has seen in the training corpus. The supervised training agent thus may fail due to the compounding errors and dialog state distribution mismatch between offline training and user interaction. To address this issue, we propose a dialog imitation learning method and let the dialog agent to learn interactively from user teaching. After obtaining a supervised training model, we deploy the agent to let it interact with users using its learned dialog policy. The agent may make errors during user interactions. We then ask expert users to correct the agent's mistakes and demonstrate the right actions for the agent to take (Ross et al., 2011). In this manner, we collect additional dialog samples that are guided by the agent's own policy. Learning on these samples directly addresses the limitation of the currently learned model. With experiments (Liu et al., 2018) in a movie booking domain, we show that the agent can efficiently learn from the expert demonstrations and improve dialog task success rate with the proposed imitation dialog learning method.

### 5.2 Dialog Reward Learning with Adversarial Training

Supervised learning models that imitate expert behavior in conducting task-oriented dialogs usually require a large amount of training samples to succeed due to the compounding errors from covariate shift as discussed in 5.1. A potential resolution to this problem is to infer a dialog reward function from expert demonstrations and use it to guide dialog policy learning. Task-oriented dialog systems are mainly designed to maximize overall user satisfaction, which can be seen as a reward, in assisting users with tasks. As claimed by Ng et al. (Ng et al., 2000), reward function as opposed to policy can usually provide the most succinct and robust definition of a task.

We propose a generative adversarial training method in recovering the dialog reward function in the expert's mind. The generator is the learned dialog agent, who interacts with users to generate dialog samples. The discriminator is a neural network model whose job is to distinguish between the agent's behavior and an expert's behavior. Specifically, we present two dialog samples, one from the human agent and one from the machine agent, to the discriminator. We let the discriminator to maximize the likelihood of the sample from the human agent and minimize that from the machine agent. The likelihood of the sample generated by the machine agent can be used as the reward to the agent. Gradient of the discriminator in optimization can be written as:

$$\nabla_{\theta_D}\Big[\mathbb{E}_{d\sim\theta_{demo}}\left[log(D(d))\right]+\mathbb{E}_{d\sim\theta_G}\left[log(1-D(d))\right]\Big] \tag{2}$$

where $\theta_G$ is the learned policy of the machine agent and $\theta_{demo}$ is the human agent policy. $\theta_D$ is the parameters of the discriminator model.

## 6 Learning from Human Feedback

In this section, we describe our proposed methods in learning task-oriented dialog model interactively from human feedback with reinforcement learning (RL).

### 6.1 End-to-End Dialog Learning with RL

After the supervised and imitation training stage, we propose to further optimize the dialog model with RL by letting the agent to interact with users and collecting simple form of user feedback. The feedback is only collected at the end of a dialog. A positive reward is assigned for success tasks, and a zero reward is assigned for failure tasks. A small step penalty is applied to each dialog turn to encourage the agent to complete the task in fewer steps. We propose to use policy gradient based methods for dialog policy learning. With likelihood ratio gradient estimator, the gradient of the objective function can be derived as:

$$\begin{aligned}\nabla_\theta J_k(\theta) &= \nabla_\theta \mathbb{E}_\theta\left[R_k\right] \\ &= \mathbb{E}_{\theta_a}\left[\nabla_\theta \log \pi_\theta(a_k|s_k)R_k\right]\end{aligned} \tag{3}$$

This last expression above gives us an unbiased gradient estimator. We sample the agent action based on the currently learned policy at each dialog turn and compute the gradient. In the experiments (Liu et al., 2017) on a movie booking task domain, we show that the proposed RL based optimization leads to significant improvement on task success rate and reduction of dialog turn size comparing to supervised training model. RL after imitation learning with human teaching not only

improves dialog policy, but also improves the underlying system components (e.g. state tracking) in the end-to-end training framework.

### 6.2 Co-Training of Dialog Agent and Simulated User

We aim to design a dialog agent that can not only learn from user feedback, but also to understand the user dynamics and predict the change of user states. Thus, we need to build a user model, which can be used to simulate conversation between an agent and a user to help the agent to learn better policies. Similar to how a dialog agent acts in task-oriented dialogs, a simulated user picks actions based on the dialog state. In addition, the user policy also depends on the user's goal. In modeling the user (Liu and Lane, 2017c), we design a hierarchical LSTM model, similar to the design of dialog agent described in section 3.2, with additional user goal encoding as the model input. The simulated user is firstly trained in a supervised manner using task-oriented dialog corpora, similar to how we train the dialog agent as described in section 4.2. After bootstrapping a dialog agent and a simulated user with supervised training, we improve them further by simulating task-oriented dialogs between the two agents and iteratively optimizing their policies with deep RL. The reward for RL can either be obtained from the learned reward function described in section 5.2 or given by the human users. The intuition behind the co-training framework is that we model task-oriented dialog as a goal fulfilling process, in which we let the dialog agent and the modeled user to positively collaborate to achieve the goal. The modeled user is given a goal to complete, and it is expected to demonstrate coherent but diverse user behavior. The agent, on the other hand, attempts to estimate the user's goal and fulfill his request.

### 6.3 Learning from Simulation and Interaction with RL

Learning dialog model from user interaction by collecting user feedback (as in section 6.1) is effective but can be very sample inefficient. One might have to employ a large number of users to interact with the agent before the system can reach a satisfactory performance level. On the other hand, learning from dialog simulation internally to the dialog agent (as in section 6.2) is relatively cheap to conduct, but the performance is limited to the modeling capacity learned from the limited labeled dialog samples. In this section, we describe our proposed method in combining the learning-from-user approach and learning-from-simulation approach, with expectation to improve the online interactive dialog learning efficiency with real users.

We let the dialog agent to conduct dialog with real users using its learned policy and collect feedback (reward) from the user. The newly collected dialog sample and reward are then used to update the dialog agent with the RL algorithm described in section 6.1. Before letting the agent to start a new session of interactive learning with users, we perform a number of learning-from-simulation training cycles. We let the updated dialog agent to "imagine" its conversation with the modeled user, and fine-tune both of them with RL using the reward obtained from the learned reward function (section 5.2). The intuition behind this proposed integrated learning method is that we want to enforce the dialog agent to fully digest the knowledge learned from the interaction with real user by simulating similar dialogs internally. Such integrated learning method may effectively improve dialog learning efficiency and reduce the number of interactive learning attempts with real users.

## 7 Conclusions

In this thesis proposal, we design an end-to-end learning framework for task-oriented dialog system. We present our proposed neural network based end-to-end dialog model architecture and discuss the proposed learning methods using offline training with dialog corpora and interactive learning with users. The proposed end-to-end dialog learning framework addresses the limitations of the popular pipeline design of task-oriented dialog systems. We show that the proposed model is able to robustly track dialog state, retrieve information from external resources, and produce appropriate system responses to complete task-oriented dialogs. The proposed learning methods achieve promising dialog task success rate and user satisfaction scores. We will further study the effectiveness of the proposed hybrid learning method in improving sample efficiency in online RL policy learning. We believe the work proposed in this thesis will pioneer a new class of end-to-end learning systems for task-oriented dialog and make a significant step towards intelligent conversational human-computer interactions.

## References


Antoine Bordes and Jason Weston. 2017. Learning end-to-end goal-oriented dialog. In *ICLR*.

Yun-Nung Chen, Dilek Hakkani-Tür, Gökhan Tür, Jianfeng Gao, and Li Deng. 2016. End-to-end memory networks with knowledge carryover for multi-turn spoken language understanding. In *INTERSPEECH*, pages 3245–3249.

Bhuwan Dhingra, Lihong Li, Xiujun Li, Jianfeng Gao, Yun-Nung Chen, Faisal Ahmed, and Li Deng. 2017. Towards end-to-end reinforcement learning of dialogue agents for information access. In *ACL*.

Mihail Eric and Christopher D Manning. 2017. A copy-augmented sequence-to-sequence architecture gives good performance on task-oriented dialogue. In *EACL*.

Milica Gasic and Steve Young. 2014. Gaussian processes for pomdp-based dialogue manager optimization. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Matthew Henderson, Blaise Thomson, and Steve Young. 2014. Word-based dialog state tracking with recurrent neural networks. In *SIGDIAL*.

Sungjin Lee and Amanda Stent. 2016. Task lineages: Dialog state tracking for flexible interaction. In *SIGDIAL*.

Xuijun Li, Yun-Nung Chen, Lihong Li, and Jianfeng Gao. 2017. End-to-end task-completion neural dialogue systems. In *IJCNLP*.

Bing Liu and Ian Lane. 2016a. Attention-based recurrent neural network models for joint intent detection and slot filling. In *Interspeech*.

Bing Liu and Ian Lane. 2016b. Joint online spoken language understanding and language modeling with recurrent neural networks. In *SIGDIAL*.

Bing Liu and Ian Lane. 2017a. Dialog context language modeling with recurrent neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE.

Bing Liu and Ian Lane. 2017b. An end-to-end trainable neural network model with belief tracking for task-oriented dialog. In *Interspeech*.

Bing Liu and Ian Lane. 2017c. Iterative policy learning in end-to-end trainable task-oriented neural dialog models. In *IEEE ASRU*.

Bing Liu, Gokhan Tur, Dilek Hakkani-Tur, Pararth Shah, and Larry Heck. 2017. End-to-end optimization of task-oriented dialogue model with deep reinforcement learning. In *NIPS Workshop on Conversational AI*.

Bing Liu, Gokhan Tur, Dilek Hakkani-Tur, Pararth Shah, and Larry Heck. 2018. Dialogue learning with human teaching and feedback in end-to-end trainable task-oriented dialogue systems. In *NAACL*.

Fei Liu and Julien Perez. 2017. Gated end-to-end memory networks. In *EACL*.

Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tur, Xiaodong He, Larry Heck, Gokhan Tur, Dong Yu, et al. 2015. Using recurrent neural networks for slot filling in spoken language understanding. *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, 23(3):530–539.

Nikola Mrkšić, Diarmuid O Séaghdha, Tsung-Hsien Wen, Blaise Thomson, and Steve Young. 2016. Neural belief tracker: Data-driven dialogue state tracking. *arXiv preprint arXiv:1606.03777*.

Andrew Y Ng, Stuart J Russell, et al. 2000. Algorithms for inverse reinforcement learning. In *ICML*.

Antoine Raux, Brian Langner, Dan Bohus, Alan W Black, and Maxine Eskenazi. 2005. Lets go public! taking a spoken dialog system to the real world. In *in Proc. of Interspeech 2005*.

Stéphane Ross, Geoffrey J Gordon, and Drew Bagnell. 2011. A reduction of imitation learning and structured prediction to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics*, pages 627–635.

Ruhi Sarikaya, Geoffrey E Hinton, and Anoop Deoras. 2014. Application of deep belief networks for natural language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.

Minjoon Seo, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Query-regression networks for machine comprehension. *arXiv preprint arXiv:1606.04582*.

Pei-Hao Su, Milica Gasic, Nikola Mrksic, Lina Rojas-Barahona, Stefan Ultes, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2016. On-line active reward learning for policy optimisation in spoken dialogue systems. In *ACL*.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *EACL*.

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 *ACL*.

Steve Young, Milica Gašić, Blaise Thomson, and Jason D Williams. 2013. Pomdp-based statistical spoken dialog systems: A review. *Proceedings of the IEEE*.