Review: Learning Goal-Oriented Spoken Dialogue

Ruolin Su

January 1, 2020

1 Introduction

Spoken Dialogue System (SDS) is a crucial component for human communicating with information systems using speech as medium. And learning from dialog is a major task for NLP. Motivated by the need for data-driven framework that reduces the cost of handcrafting complex dialog managers and that provides robustness against the errors created by ASR operating in noisy environment, a recent line of works develop statistical/parametric dialogue models for dialogue systems.

In common sense, SDS involves two main components:

- Dialogue Model $\mathcal{M}$ for tracking dialogue states;
- Policy Model $\mathcal{P}$ for the next system act via a decision rule.

The former learns the dialogue states from semantic representations and updates states based on the dialogue history. The latter component is mostly handling with taking a response to the current state. There are also some works considering both dialogue state tracking and decision making as end-to-end models [2, 11, 20] and can also combined with information extraction [1].

Outline The remainder of this article is organized as follows. Section 2 reviews works associated with Dialogue State Tracking (DST). Recent trend is from statistical models to sequential models and transfer learning for DST.

in Section 3, models and applications for conversational dialogue systems are introduced. Section 4 reviews a widely used statistical dialogue system, partially observable Markov decision processes (POMDPs). And Section 5 analyses dataset for SDS.

2 Dialogue State Tracking

In this section, we first define the problem of DST for spoken dialogue, and introduce the trend of works on DST algorithms.

- *Dialogue State.* Dialog state s_t is a data structure drawn from a set S that capturing dialog history up to time t to a level that provides sufficient information for choosing the next system action [23]. It can be considered as encoding user's goal in the conversation context (e.g. in bus timetable domain, which bus stop the user wants to leave from, where they are going to, and whether the system has already offered a bus on that route).

- *Dialog State Tracker.* Dialogue state tracker takes observable elements up to time t in a dialog as input, and outputs its estimate of the current state of the dialog (e.g. a distribution over multiple possible dialog states $b(s)$) to determine the true current state $s^\star$ [23]. Input is including all of the results from the ASR (e.g. N-Best list of sentences, a word confusion network or lattice) and SLU (e.g. N-Best list of semantic interpretations) components, all system actions taken, and external knowledge sources such as databases and models of past dialog.

Generally Speaking, there are three families of dialog state tracking algorithms: hand-crafted rules, generative models, discrimintive models [15].

- Hand-crafted Rules

  A line of works such as MIT JUPITER weather information system and the Information State Update approach maps the existing state s and the 1-best SLU result to a new state s', which is unable to make use of the entire ASR or SLU N-Best list by tracking a single dialog state. Other hand-crafted rule-based trackers compute scores for all dialog states suggested by the whole ASR/SLU N-best lists.

Hand-crafted rules are easy for bootstrapping with no need for data and easy for incorporating domain knowledge. Limitations are obvious that formula parameters are not derived directly from real dialog data, so they require careful tuning and do not learn from dialog datan(thus motivate data-driven methods).

- Generative Models

Generative approaches are based on the assumptions that dialog can be modeled as a Bayesian network, relating the dialog state s to the system action a, the (true, unobserved) user action u, and ASR or SLU result $\tilde{u}$ [23]. Distribution over possible dialog state is drawn from Bayesian inference.

$$b'(s') = \eta \sum_{u'} P(\tilde{u}'|u')P(u'|s', a) \sum_{s} P(s'|s, a)b(s)$$

- where $b'(s')$ is the updated distribution over dialogue states;

- η is a normalizing constant $\in [0, 1]$;

- $P(\tilde{u}'|u')$ is probability of the ASR/SLU producing the observed output $\tilde{u}'$ given the true & unobserved user action u';

- $P(u'|s', a)$ is probability of the user taking action u' given the true dialog state s' and system action a;

- $P(s'|s, a)$ is probability of the dialog state changing to s' given it is currently s and the system takes action a.

Variants are applications on different factorization of the hidden state (e.g. contexts, whether or not confirmed) [4, 16, 17, 19]. Weakness of generative models are that size of parameters is quadratic to num of states when running real time, all dependencies between features must be explicitly modeled seems impractical (e.g. independent assumptions like uniform errors violate Markov assumption). Approximation approaches includes 'beam' parameters, or further factorization.

- Discriminative Models

  Discriminative models update dialog state to s' as $b'(s') = P(s'|f')$ taking no prior assumptions on $P(s'|f')$, where f' represent features extracted from ASR/SLU/dialogue_history.

  Variants on discriminate models includes *a. encode dialog history in the features* to train a classifier for dialog states, *b. discriminative Markov Model, c. conditional random field (CRF), d. recurrent neural networks (b,c,d are sequential models)*

What is the role of discriminative models in spoken dialogue system?

Thanks to the work in use of a simulated user [7] and learned dialog policy by implementing reinforcement learning [8] implies the distribution of dialog states encountered at test time was different than that encountered in training, and the fact that the discriminative tracker performed well suggests not overfitting the training data.

Analysing the state-of-the-art models cast light on favorable ideas for better DST performance, such as discriminative models, use of ASR features, sequential modeling, capturing feature interactions and exploiting joint posteriors [19].

3 Conversational Dialogue Systems

There is always a need to design conversational dialogue system such as in chatbot design or voice assistant. This kind of dialogue systems are required to take response to the dialog history. Two types of dialogue systems have been developed: generation-based models and retrieval-based models [23].

For example, the Twitter Conversation Triple Dataset is typically used for evaluating generation-based dialogue systems, containing 3-turn Twitter conversation instances. Ritter et al. [10] employed the phrase-based statistical machine translation (SMT) framework to "translate" the message to its appropriate response. Sordoni et al. [12] reranked the best responses produced by machine translation with a context-sensitive RNN encoder-decoder framework, observing substantial gains. Li et al. [9] reported results on replacing the traditional maximum log likelihood training objective with the maximum mutual information training objective, in an effort to produce interesting and diverse responses, both of which are tested on a LSTM encoder-decoder framework.

On the other hand, the response retrieval task is defined as selecting the best response from a repository of candidate responses. The Ubuntu dialogue dataset was constructed by scraping multi-turn Ubuntu trouble-shooting dialogues from an online chatroom. Researchers used LSTMs to encode the message and response, and then inner product of the two sentence embeddings is used to rank candidates.

Other work includes exploiting the multi-turn nature of human conversation by employing the LSTM encoder on top of sentence-level CNN embeddings, and designing a memory network, where the past conversation was treated as memory and the latest utterance was considered as a "question" to be responded to using simple neural bag-of-word embedding.

4 POMDP-based Statistical Spoken Dialogue System

Partially observable Markov decision processes (POMDPs) is a framework including an explicit Bayesian model of uncertainty and optimizing the policy via a reward-driven process. In this section, we review algorithms for developing POMDP-based spoken dialogue system.

4.1 PMODP Model

POMDP assumes that dialog progress as a Markov process, at each turn, the system regards the output of the SLU as a noisy observation u of the user input with probability $P(u|s)$.

$$b'(s') = \eta P(u'|s', a) \sum_{s} P(s'|s, a) b(s)$$

$$\eta = 1/P(u'|b, a) \text{ normalization constant}$$

POMDP-based model of dialog combines two key ideas: belief state tracking and reinforcement learning [21]. Several advantages have been validated using POMDP-based dialogue model.

- update posterior of belief state via Bayesian inference (confusion network, N-best-lists), evidence is integrated across turns such that a single error has significantly reduced impact, and in contrast to conventional systems, user persistence is rewarded.
- be embedded in a simple homogeneous mapping from belief state to action.
- leads to optimal decision policies, avoids the cost of expensive manual tuning and refinement procedures, and enables more complex planning to be implemented than would be feasible using only manual handcrafted designs.

4.2 Policy Model

The system action is determined by a policy π, represented in generally two ways.

- a deterministic mapping from belief states b to actions a: $\pi(b) \in A$.
- stochastically via a distribution over actions $\pi(a|b) \in [0, 1]$ where $\pi(a|b)$ is the probability of taking action a in belief state b, and $sum(pi(a|b)) = 1 \ \forall \ b$.

4.3 Value Function (deterministic, stochastic)

Value functions can be categorized as deterministic functions as

$$V^{\pi}(b) = r(b, \pi(b)) + \gamma \sum_{u'} P(u'|b, \pi(b)V^{\pi}(b')$$

and stochastic functions as

$$V^{\pi}(b) = \sum_{a} \pi(a|b) \left\{ r(b, a) + \gamma \sum_{a'} P(u'|b, a)V^{\pi}(b') \right\}$$

4.4 Bellman Optimality Equation

The optimal policy is one that maximizes $V^{p}i$ to yield $V*$.

$$V^{*}(b) = max_a[r(b, a) + \gamma \sum_{u'} P((u'|b, a)V * (b')]$$

4.5 Factorization and Approximation

Belief update can be factorized as:

- The observation model represents the probability of an observation o given the user's actual utterance u. (speech understanding error effects). as (1).
- The user model represents the likelihood that the user would use utter u given the previous system output and new system state.(user behavior). as (2).
- The goal transition model represents the likelihood that the user goal has changed. as (3).
- The history model represents the system's memory of the dialog to the current state. as (4).

$$
\begin{aligned}
b'(g', u', h') =& \eta P(\tilde{u}'|u') & (1)\\
& \cdot P(u'|g', a) & (2)\\
& \cdot \sum_g P(g'|g, a) & (3)\\
& \sum_h P(h'|g', u', h, a) & (4)\\
& \cdot b(g, h) & (5)
\end{aligned}
$$

Approximation: N-best approach including pruning and recombination strategies [22, 6, 18, 5] e.g. HIS model (constant user goal and partition based)

$$
\begin{aligned}
b'(p', u', h') =& \eta P(\tilde{u}'|u') P(u'|p', a) & (6)\\
& \cdot \sum_h P(h'|p', u', h, a) & (7)\\
& \cdot P(p'|p) b(h) & (8)
\end{aligned}
$$

Approximation: Factored Bayesian network approach [3, 13, 14] factor the user goal into concepts that can be spoken about by the system.

- N-best approach, model all dependencies but with an incomplete distribution, and
- the slot-level factoring approach, can handle only a limited number of dependencies but can model the complete distribution.

4.5.1 Representation and estimation of the policy model

Policy model $\mathcal{P}$ maps from the belief state b to the appropriate system action a. A non-parametric policy should first encode a partitioning of belief space that maps to the same action, and second, encode the optimal action to take for each partition.

Summary space is a compressed feature space due to small part of belief space actually be visited and restricted range of plausible policy during normal dialog.

Summary-space mapping requires two components: a mechanism to select candidate actions in master space, and functions to extract features from the belief state and candidate actions.

Given a specific summary space, a policy is represented as deterministic mapping $\pi(\hat{b}) \to \hat{a}$ or as conditional probability distribution $\pi(\hat{b}, \hat{a}) = p(\hat{a}|\hat{b})$.

Deterministic mapping is to find an optimal Q-function:

$$\pi^*(\hat{b}) = argmax_{\hat{a}}\{Q^*(\hat{b}, \hat{a})\}$$

Methods such as planning under uncertainty, value iteration, Monte Carlo optimization, least squares policy iteration (LSPI), and natural actor–critic, are popular for policy optimization.

4.5.2 User Simulation

Learning directly from corpora is problematic since the state space that prevailed during the collection of the data may differ from that used in policy optimization and also it precludes the use of online interactive learning algorithms.

- The model for $p(u|\dots)$ should match the statistics of user responses in available corpora and the error model, match the characteristics of the speech recognition and understanding system.
- User Simulation Models: N grams, dynamic Bayesian networks, HMM, CRF.
- Error Simulation Models.

4.5.3 Parameter Optimization

Expectation Maximization EM operates by using a fixed set of parameters to estimate marginal distributions of the hidden parameters in its first step. It then uses these marginals to re-estimate the parameters of the model and repeats.

Expectation Propagation operates directly on a factored Bayesian network and simply extends the loopy belief propagation algorithm to handle continuous state spaces. This allows the parameters to be updated during the propagation step and means that all the conditional independence assumptions are used in simplifying the update. (requires no annotations of either the true semantics or the dialog state)

Reinforcement Learning changes from an inference problem to a reinforcement learning task.

4.6 Discussions on PMODP

POMDPs provide an explicit Bayesian model of uncertainty and by providing a reward-driven process for policy optimization [21], which not only reduces the cost of handcrafted dialog managers and provides robustness in noisy environments against the errors by speech recognizers. On the other hand, the complexity of optimizing the model and heavy dependence on the formula of reward are challenging for further exploration.

5 Dataset

Here we investigate the Dialog System Technology Challenges (DSTCs) [15] dataset. The input are the dialog state consists of a frame of slots.

- *Informable slots:* slots provided by the user that describe their goal, such as the bus route and origin bus stop
- *Requested slots:* slots the user wants to retrieve, e.g. the phone number, or price range (of a restaurant)
- *Search method:* if the user wanted to query by providing constraints, providing the name of a restaurant, navigating a results list, etc.

Output and Evaluation. Each outputs a probability distribution over the set of possible dialog states. The goal is to assign probability 1.0 to the correct state, and 0.0 to other states. In each dialog state hypothesis output, every slot is scored, so to be correct, the hypothesis must have perfect precision and recall.

- **Accuracy** measures the percent of turns where the top-ranked hypothesis is correct. This indicates thecorrectness of the item with the maximum score.
- **L2** measures the L2 distance between the vector of scores, and a vector of zeros with1in the position of the correct hypothesis. This indicates the quality of all scores, when the scores are viewed as probabilities.
- **AvgP** measures the mean score of the first correct hypothesis. This indicates the quality of the score assigned to the correct hypothesis, ignoring the distribution of scores to incorrect hypotheses.
- **MRR** measures the mean reciprocal rank of the first correct hypothesis. This indicates the quality of the ordering of the scores (without necessarily treating the scores as probabilities).
- **Receiver-operating characteristic (ROC) curve** measure the discrimination of the score for the highest-ranked state hypothesis.
- **Neglogpis** the mean negative logarithm of the score given to the correct hypothesis,$log\pi$. Also called the negative log likelihood, this is a standard score in machine learning tasks. Two metrics, **Update precision** and **Update accuracy** measure the accuracy and precision of updates to the top scoring hypothesis from one turn to the next.
- **Schedules** measure which turns to include when computing each metric. **schedule1** includes every turn. **schedule2** include turns where the target slot is either present on the SLU n-best list,or where the target slot is included in a system confirmation action – i.e., where there is some observable new information about the target slot. **schedule3** includes only the last turn of a dialog.

Baselines are provided for comparing SDS performance.

- Common simple baseline 'team0 entry0', maintains a single hypothesis for each slot. (mimics standard (non-statistical) approaches commonly used in spoken dialog systems, 1-best)
- Focus baseline 'team0 entry1', uses rule to change goal constraints at each turns.

- HWU Baseline 'team0 entry2', uses a selection of domain independent rules to update the beliefs, and noise adjustment rule, to adjust the SLU scores.
- Oracle 'team0 entry3', reports the correct label with score 1 for each component of the dialog state, but iff. suggested in the dialog so far by the SLU (gives an upper-bound for the performance of a tracker which uses only the SLU and its suggested hypotheses).

How much opportunity for improvement remains?

By calculating the percentage of turns where the tracker was correct and the baseline was not, and the percentage of turns where the baseline was correct and the tracker was not. Implying that there is additional scope for improvement, perhaps through combining multiple trackers using ensemble methods.

References

[1] B. AbuShawar and E. Atwell. Automatic extraction of chatbot training data from natural dialogue corpora. In *RE-WOCHAT: Workshop on Collecting and Generating Resources for Chatbots and Conversational Agents-Development and Evaluation Workshop Programme (May 28 th, 2016)*, page 29, 2016.

[2] A. Bordes, Y.-L. Boureau, and J. Weston. Learning end-to-end goal-oriented dialog. *arXiv preprint arXiv:1605.07683*, 2016.

[3] T. H. Bui, M. Poel, A. Nijholt, and J. Zwiers. A tractable hybrid ddn–pomdp approach to affective dialogue modeling for probabilistic frame-based dialogue systems. *Natural Language Engineering*, 15(2):273–307, 2009.

[4] D. DeVault and M. Stone. Managing ambiguities across utterances in dialogue. In *Proceedings of the 11th Workshop on the Semantics and Pragmatics of Dialogue (Decalog 2007)*, pages 49–56, 2007.

[5] M. Gašić and S. Young. Effective handling of dialogue state in the hidden information state pomdp-based dialogue manager. *ACM Transactions on Speech and Language Processing (TSLP)*, 7(3):4, 2011.

[6] K. Kim, C. Lee, S. Jung, and G. G. Lee. A frame-based probabilistic framework for spoken dialog management using dialog examples. In *Proceedings of the 9th SIGdial Workshop on Discourse and Dialogue*, pages 120–127. Association for Computational Linguistics, 2008.

[7] S. Kim and R. E. Banchs. Sequential labeling for tracking dynamic dialog states. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 332–336, 2014.

[8] S. Lee. Extrinsic evaluation of dialog state tracking and predictive metrics for dialog policy optimization. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 310–317, 2014.

[9] J. Li, M. Galley, C. Brockett, J. Gao, and B. Dolan. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*, 2015.

[10] A. Ritter, C. Cherry, and W. B. Dolan. Data-driven response generation in social media. In *Proceedings of the conference on empirical methods in natural language processing*, pages 583–593. Association for Computational Linguistics, 2011.

[11] N. Roy, J. Pineau, and S. Thrun. Spoken dialogue management using probabilistic reasoning. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, pages 93–100. Association for Computational Linguistics, 2000.

[12] A. Sordoni, M. Galley, M. Auli, C. Brockett, Y. Ji, M. Mitchell, J.-Y. Nie, J. Gao, and B. Dolan. A neural network approach to context-sensitive generation of conversational responses. *arXiv preprint arXiv:1506.06714*, 2015.

[13] B. Thomson, J. Schatzmann, and S. Young. Bayesian update of dialogue state for robust dialogue systems. In *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4937–4940. IEEE, 2008.

[14] B. Thomson and S. Young. Bayesian update of dialogue state: A pomdp framework for spoken dialogue systems. *Computer Speech & Language*, 24(4):562–588, 2010.

[15] J. Williams, A. Raux, and M. Henderson. The dialog state tracking challenge series: A review. *Dialogue & Discourse*, 7(3):4–33, 2016.

[16] J. D. Williams. Using particle filters to track dialogue state. In *2007 IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU)*, pages 502–507. IEEE, 2007.

[17] J. D. Williams. Exploiting the asr n-best by tracking multiple dialog state hypotheses. In *Ninth Annual Conference of the International Speech Communication Association*, 2008.

[18] J. D. Williams. Incremental partition recombination for efficient tracking of multiple dialog states. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5382–5385. IEEE, 2010.

[19] J. D. Williams. Challenges and opportunities for state tracking in statistical spoken dialog systems: Results from two public deployments. *IEEE Journal of Selected Topics in Signal Processing*, 6(8):959–970, 2012.

[20] J. D. Williams and S. Young. Partially observable markov decision processes for spoken dialog systems. *Computer Speech & Language*, 21(2):393–422, 2007.

[21] S. Young, M. Gašić, B. Thomson, and J. D. Williams. Pomdp-based statistical spoken dialog systems: A review. *Proceedings of the IEEE*, 101(5):1160–1179, 2013.

[22] S. Young, J. Schatzmann, K. Weilhammer, and H. Ye. The hidden information state approach to dialog management. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing-ICASSP'07*, volume 4, pages IV–149. IEEE, 2007.

[23] T. Young, D. Hazarika, S. Poria, and E. Cambria. Recent trends in deep learning based natural language processing. *ieee Computational intelligenCe magazine*, 13(3):55–75, 2018.