*INTERSPEECH 2016*
September 8–12, 2016, San Francisco, USA

# Improving The Probabilistic Framework for Representing Dialogue Systems with User Response Model

*Miao Li, Zhipeng Chen, Ji Wu*

Department of Electronic Engineering, Tsinghua University, Beijing, China
{miao-li10, zp-chen11}@mails.tsinghua.edu.cn, wuji_ee@mail.tsinghua.edu.cn

## Abstract

A probabilistic framework for goal-driven spoken dialogue systems (SDSs) has been proposed by us in a previous work. In the framework, a target distribution, instead of the frame structure, is used to represent the dialogue state at each turn. The target-based state tracking algorithm enables the system to handle uncertainties in the dialogue. By summarizing the target-based state, information from the back-end database can be exploited to develop efficient dialogue strategies. To extend our probabilistic framework and adapt our approach to real application scenarios, a user response model is investigated and integrated into the probabilistic framework to enhance the dialogue policy in this paper. Experiments in both ideal setting and real user test setting are conducted to test the enhanced dialogue policy. The results show that despite an unavoidable mismatch between the user response model based on prior knowledge and real users' behaviors in the experiment, the enhanced dialogue policy works robustly and efficiently. The results further demonstrate that the probabilistic framework is quite flexible and amenable to the integration of additional factors and models of real-world dialogue problems.

**Index Terms**: spoken dialogue system, probabilistic dialogue representation, dialogue management, entropy minimization, user response model

## 1. Introduction

Dialogue Management (DM) is the most important module in a spoken dialogue system. Nowadays researchers have divided the module into two important tasks, namely dialogue state tracking task and dialogue control task. By maintaining a distribution over multiple hypotheses of the true dialogue state, dialogue state tracking aims to handle the uncertainty introduced by the automatic speech recognition (ASR) and spoken language understanding (SLU). Following the framework of Partially Observable Markov Decision Process (POMDP), generative methods are first investigated [1, 2, 3]. Recently, discriminative methods which directly model the posterior distribution are proposed and yield better performance. These methods include Maximum Entropy Model [4, 5], Conditional Random Field [6, 7] and Recurrent Neural Network [8, 9]. Besides these statistical approaches, some robust domain-independent rule-based approaches [10] are also attractive due to their efficiency and portability.

The work was supported in part by the National Natural Science Funds of China under Grant 61571266, and in part by the Electronic Information Industry Development Fund of China under project The R&D and Industrialization on Information Retrieval System Based on Man-Machine Interaction with Natural Speech.

Dialogue control task aims to find a policy to choose a proper system action to interact with the user. In the framework of POMDP, how to get a policy can be cast as a learning problem. The dialogue system can learn an optimal policy by interacts with users through Reinforcement Learning (RL) [11, 12, 13]. However the POMDP methods cannot make full use of the information from the database to generate system responses. Other researchers try to exploit the information in the back-end database and generate specific queries to users in a cooperative manner [14, 15]. The DM module scans the database in every turn of the dialogue and it filters the database based on current understanding of user's intentions and preferences. Then a system initiative query is generated based on the remaining items in the database. It tends to ask users about the attributes with highest uncertainty so as to reduce the search space as much as possible. One problem of this database summary dialogue management (DSDM) approach is that it assumes that all attributes are uniformly distributed which is not always true in real situations. Besides, it can not effectively handle uncertainties or errors caused by ASR and SLU modules.

To leverage the database information as well as handle the uncertainties in the dialogue, a probabilistic framework is introduced to represent spoken dialogue system [16, 17]. In this framework, the dialogue state is represented by a distribution over database entries (which we call targets in the rest of this paper). A target-based state tracker is used to track the target distribution according to N-best SLU hypotheses at each turn, which provides a way to handle uncertainties. By summarizing the target-based state, an entropy minimization dialogue management (EMDM) strategy is used to choose a best system action. In this paper, we will improve the probabilistic framework by integrating a user response model, which mentioned as future work in [16]. A reformative EMDM strategy is presented to deal with the user response model. It will be shown that the probabilistic framework is flexible and robust to extend to real world problems. The rest of this paper is organized as follows. Section 2 gives a review of the probabilistic framework. Section 3 explains the importance of user response model for policy optimization in the framework and then introduces a reformative EMDM strategy. Experiments are presented in Section 4 and we conclude this paper in Section 5.

## 2. Probabilistic Framework for spoken dialogue system

In this section, we will briefly review the probabilistic framework, which is originally proposed in [16].



http://dx.doi.org/10.21437/Interspeech.2016-810

### 2.1. Probabilistic Dialogue Representation

In general, there is a structural back-end database $\mathbf{D} = \{d_i | i = 1, 2, ..., I\}$ for each of the goal-driven information access system, where each entry $d_i$ of the database represent a potential dialogue target wanted by a user. Each entry is often associated with a set of attributes $\mathbf{A} = \{a_k | k = 1, 2, ..., K\}$. To begin a dialogue, the system is in a initial state $S^{(0)}$, a system action $q^{(1)}$ is presented to a user and let $r^{(1)}$ denotes the user response. The pair $(q^{(1)}, r^{(1)})$ forms the first turn of the dialogue. After the first turn of the dialogue, the system state evolves to a new state $S^{(1)}$ and generate a new system action $q^{(2)}$. Following this process, we can have a dialogue. Let $\mathcal{S}_0^T = \{S^{(t)} | t = 0, 1, ..., T\}$ denote the state sequence and $\mathcal{H}_1^T = \{(q^{(t)}, r^{(t)}) | t = 1, 2, ..., T\}$ denote the dialogue interaction history, where $T$ is the number of dialogue turns. The probability of the dialogue process up to turn $t$ is represented by:

$$\begin{aligned} &P(\mathcal{S}_0^t, \mathcal{H}_1^t, D) = P(q^{(t)}, r^{(t)}, S^{(t)} | \mathcal{S}_0^{t-1}, \mathcal{H}_1^{t-1}, D) \\ &\cdot P(\mathcal{S}_0^{t-1}, \mathcal{H}_1^{t-1}, D) \end{aligned} \tag{1}$$

The Equation (1) is a recursion formula, where the first item in the RHS of Equation (1) expresses the probability of the current dialogue situation conditioned on the past dialogue process. It can be further factored into three parts:

$$\begin{aligned} &P(q^{(t)}, r^{(t)}, S^{(t)} | \mathcal{S}_0^{t-1}, \mathcal{H}_1^{t-1}, D) = P(q^{(t)} | S^{(t-1)}, D) \\ &\cdot P(r^{(t)} | q^{(t)}, \mathcal{H}_1^{t-1}, D) \cdot P(S^{(t)} | S^{(t-1)}, \mathcal{H}_1^t, D) \end{aligned} \tag{2}$$

The first part in the RHS of Equation (2) denotes the policy of a DM model, the last part in the RHS of Equation (2) is the state evolution model. These two models characterize the two most important tasks in a DM model, i.e. the dialogue control task and the dialogue state tracking task. In addition, the second part in the RHS of Equation (2), which corresponds to a user response model, is also concerned in this probabilistic framework.

### 2.2. Target-Based State Representation and Tracking

In most of the goal-driven spoken dialogue systems, the dialogue states are represented by a frame structure, which consists of a combination of attributes and possible values. When the system collects enough information from the dialogue, it offers a target which fits all the information to the user. To handle the uncertainties in the dialogue, the system needs to maintain a distribution over possible frame states. Instead of maintaining such a belief state, we use a distribution over all possible targets to represent the dialogue state in our framework. The initial state $S^{(0)}$ is the prior target distribution. After each dialogue turn, the posterior distribution over all targets is updated to form a new dialogue state.

Similar to the general dialogue state tracking task, the n-best list of SLU hypotheses of each turn is used to track the target-based state. Each hypothesis consists of a dialogue action type such as "inform" or "deny" and a set of constrains about the attributes and values. There is also a corresponding probability indicator for the confidence level for each hypothesis. Each hypothesis will be converted to a support evidence for a particular target subset based on the constrains and dialogue action type. Then the probability of each SLU hypothesis is re-allocated to the targets in the subset. After processing all SLU hypotheses in the n-best list, we can construct a support distribution over the all targets. Thus, at the $t^{th}$ turn, let $P_i^{t-1}$ denotes the probability of a target $d_i$ in previous dialogue state and $C_i^t$ denotes the probability of $d_i$ in support distribution converted from $(q^t, r^t)$. The target distribution at turn $t$ can be updated as follows:

$$\begin{aligned} &P_i^t = \eta(1 - (1 - P_i^{t-1})(1 - C_i^t)), \forall i, C_i^t > 0; \\ &P_i^t = \eta(1 - \sum_k C_k^t) P_i^{t-1}, \forall i, C_i^t = 0; \end{aligned} \tag{3}$$

where $\eta$ is a normalization constant. Based on Equation (3), the posterior probability of a target $d_i$ will increase when the target is strongly supported by the SLU hypotheses. While the posterior probability of a target $d_i$ will decrease when the target $d_i$ is not supported by any SLU hypothesis. The update approach can handle the SLU hypothesis with joint constrains and don't have the update order problem in [10].

### 2.3. The EMDM strategy

The target-based state can provide us a clearer picture of the dialogue process. The target distribution is flat in the beginning of a dialogue if without any prior knowledge. The entropy of the target distribution is large because the system doesn't know which target is wanted by the user. As the system collects more and more information from the user, the target distribution keeps going sharper and sharper and the entropy keeps decreasing. When the target wanted by the user is correctly and successfully reached, the entropy, which describe the uncertainty of user goal, will decrease to a certain level. So if a dialogue strategy can minimize the entropy of the target distribution at each dialogue turn, we can have a more efficient dialogue. This is the intuition of our entropy minimization dialogue management strategy.

## 3. User Response Model and the Reformative EMDM Strategy

In our probabilistic framework, finding dialogue policy can be cast as a dynamic optimization problem. At turn $t$ of a dialogue, the objective of our EMDM strategy is to find an optimal action $q^{(t)}$ that could expectably minimize the entropy $H(S^{(t)})$ of the target-based state $S^{(t)}$. Based on Equation (2), we know the target distribution $S^{(t)}$ depends on the system action $q^{(t)}$, the user response $r^{(t)}$, and the previous target distribution $S^{(t-1)}$. So the objective of the EMDM strategy can be expressed as follows:

$$\underset{q^{(t)}}{\operatorname{argmin}} \{E_{r^{(t)}}[H(S^{(t)} | q^{(t)}, r^{(t)}, S^{(t-1)})]\} \tag{4}$$

Given a system action $q^{(t)}$, a different user response $r^{(t)}$ will lead to a different posterior target distribution $S^{(t)}$. So the expectation operation in Equation (4) is based on the distribution $P(r^{(t)} | q^{(t)}, \mathcal{H}_1^{t-1}, D)$, which is called the user response model in our framework. The action $q^{(t)}$ actually plays a role of leading users to present the most needed information to the system based on the summarization of the previous state $S^{(t-1)}$. The user response $r^{(t)}$ directly determines the entropy of the posterior target distribution $S^{(t)}$. Thus the dialogue strategy strongly relies on the user response model.

Minimize the entropy is equal to maximize the entropy reduction, so we get an equivalent objective of Equation (4):

$$\underset{q^{(t)}}{\operatorname{argmax}}\{E_{r^{(t)}}[H(S^{(t-1)}) - H(S^{(t)}|q^{(t)}, r^{(t)}, S^{(t-1)})]\} \tag{5}$$

If we restrict the system actions to query one of the attributes to the user at one turn. It has been proven that the solution of Equation (5), which is to find a system action that can maximize the entropy reduction, is equal to querying the attribute $a_k$ with maximal entropy if the user is always cooperative and knowledgable [16]. Following the simple assumption of the user response model, the resulting EMDM strategy is as below:

$$\underset{a_k}{\operatorname{argmax}}\{H(a_k|S^{t-1})\} \tag{6}$$

the marginal distribution of attribute $a_k$ can be easily calculated from the target distribution $S^{t-1}$. This is the EMDM strategy proposed in [16].

While in real world applications, we can't expect that the user is always knowledgable enough. For example, in a Song-On-Demand (SoD) task, the user may only know some attributes of the song he wanted. In general, the "singer" or the "language" of a song has more probability to be known by a user while the "lyricist" or the "album" may not be known by a user with high probability. So if we happen to ask a user about an attribute which he doesn't know, we can't reduce the entropy as we expected. The same situation happens when we meet a "don't care". In this paper, we improve the framework by introducing a user response model to handle such "don't know" or "don't care" situations. When considering the user response model, the objective of our reformative entropy minimization dialogue management (REMDM) strategy is:

$$\underset{a_k}{\operatorname{argmax}}\{q(a_k)H(a_k|S)\} \tag{7}$$

where $q(a_k)$ represents the probability of that users can answer their preferences about attribute $a_k$, so $q(a_k)H(a_k|S)$ is the expected entropy reduction for querying $a_k$. The experimental results shows that the REMDM strategy can bring a noteworthy improvement.

## 4. Experiments and Result Analysis

We conduct experiments in a SoD task. The database consists of 38,117 songs in total and each song is associated with a set of 12 attributes listed in Table 1. The number of possible values for each slot is shown in the rightmost column of Table 1.

A user response model is acquired by questionnaires from 52 college students. The user response model describes how likely the system can get a response from a user when querying an attribute to the user. The collected user response model is shown below. It is found that querying attributes such as "singer", "language" and "emotion" are more likely to get correct answers from users. While querying attributes such as "album", "company" and "Time" can hardly get answers form users.

To test the proposed REMDM strategy, we set up two experiment. The first one is tested with a simulated user in an ideal setting and the second one is tested with real users.

Table 1: *The 12 slots of a song in the database.*

| ID | Slot | Description | Size |
|---|---|---|---|
| 1 | Singer | The name of the singer | 3010 |
| 2 | Gender | The gender of the singer | 2 |
| 3 | Region | The region of the singer | 19 |
| 4 | Album | The album of the song | 10024 |
| 5 | Company | The publisher of the song | 1184 |
| 6 | Language | The language of the song | 10 |
| 7 | Lyricist | The lyricist of the song | 5633 |
| 8 | Composer | The composer of the song | 5582 |
| 9 | Live | Live version or not | 2 |
| 10 | Time | Release year of the song | 50 |
| 11 | Style | The style of the song | 15 |
| 12 | Emotion | The emotion of the song | 37 |

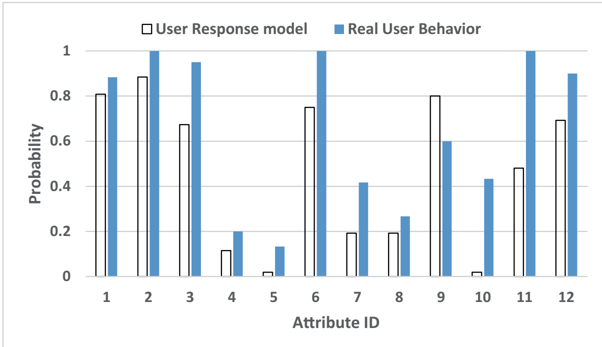


Figure 1: The user response model and real user behavior in experiments.

### 4.1. Dialogue Management Experiments in Ideal Settings

A simulated user is used in this experiment. For each dialogue, the simulated user first choose a song from the database as its goal, and the dialogue system has no idea of the chosen song. When in the dialogue process, the system chooses an attribute to ask the simulated user based on a particular strategy, then the simulated user tries to answer the question based on the collected user response model. In this ideal setting, the utterance generated by the simulated user can be fully understood by the dialogue system. Then the system updates the dialogue state and chooses another attribute to ask. This process repeated until one of the following three conditions are met: 1) there are no more than one song remaining in the candidate set, 2) the entropies of all 12 attributes dropped to 0, or 3) all attributes have been asked by the system.

Five strategies are compared in this experiment, namely sequential, random, DSDM [14] and the original EMDM strategy [16] and the REMDM strategy. The sequential strategy chooses attributes in a prefixed order while the random strategy chooses attributes in a random order. The DSDM strategy is an approximate entropy-based method, it chooses the attribute with maximum distinct values in current candidate set [14]. The songs used in this experiment are chosen in two settings. The first setting is called "uniform", which chooses each song once and only once. The second setting is called "sampling", which samples 500,000 songs from the database according to a prior distribution of these songs. The result of this experiment is shown in Table 2. We can find that the EMDM strategy and the REMDM

Table 2: *Average dialogue turns of each strategy.*

| Strategy | Uniform | Sampling |
|---|---|---|
| Sequential | 11.475 | 11.016 |
| Random | 11.020 | 10.040 |
| DSDM | 6.972 | 6.962 |
| EMDM | 6.772 | 6.368 |
| REMDM | **6.542** | **5.899** |

strategy outperform all the other three non-EMDM strategies. Because of using entropy as objective, the EMDM strategy and the REMDM strategy perform even better in the sampling setting. When integrating the user response model, the REMDM strategy leads to a significant improvement than all other strategies.

Table 3: *Comparison between the two EMDM strategies for dialogue turns in ideal setting.*

| Setting | #RE<#E | #RE=#E | #RE>#E | Total |
|---|---|---|---|---|
| Uniform | 42.64% | 32.05% | 25.31% | 38117 |
| Sampling | 50.12% | 25.26% | 24.62% | 500000 |

#RE=REMDM strategy, #E=EMDM strategy.

We further compare the REMDM strategy and the EMDM strategy in each test song, the results are shown in Table 3. With the "uniform" setting, the chance of REMDM performing shorter dialogue turns is 42.64% while the chance of EMDM to be shorter is 25.31%. With the "sampling" setting, the performance of REMDM is even better, the chance for REMDM (50.12%) to be shorter is more than 2 times than the EMDM (24.62%).

### 4.2. Dialogue Management Experiments with Real Users

We also setup experiments with real users on a full end-to-end spoken dialogue system. A large vocabulary continuous speech recognizer [18, 19] is used to transcribe the input speech. Based on the multiple-candidates recognition results, a rule-based SLU module [20] is used to generate the n-best SLU results[1]. Top-1 and Top-5 accuracies are used as the performance metrics for both ASR and SLU module. The top-1 accuracy of our ASR module is 86% and the top-5 accuracy of our ASR module is 92%. The top-1 accuracy of our SLU module is 84% and the top-5 accuracy of our SLU is 89%. This real users experiment aims to test the ability of handling errors of our framework as well as the effectiveness of the EMDM strategies. Six real users are involved in the experiment, each user is given 10 songs as his or her targets to test with the dialogue systems. The users response to the system based on his or her own knowledge. This yields 60 test cases for each strategy. The real users' behaviors in the experiments are shown in Figure 1. It can be found that there is a mismatch between the user response model and the real users' behavior during the experiments. For sequential and random strategies, we use frame structure to represent dialogue state, only top-1 SLU hypothesis is used to update the dialogue state, the update approach follows [10]. The termination conditions for these two strategies are same with the ideal test. For DSDM and EMDM strategies, the target-based state is used to represent the dialogue state and n-best SLU hypotheses are used to track the state. In this setting, we can hardly find a target song with a total certainty. So we change the dialogue termination conditions as follows: 1) the probability of a candidate song is larger than a threshold, for example 80% (the threshold is determined empirically); or 2) all attributes have been asked by the system. When a dialogue is finished, the top-1 accuracy (whether the offered song is the right song), top-5 accuracy (whether the right song is included in top-5 candidate songs) and average dialogue turns are evaluated as the metrics. The results of this experiment are shown in Table 4. It can be

Table 4: *Performances of the dialogue systems for real test*

| Strategy | Top-1 Accu(%) | Top-5 Accu(%) | Average Turns |
|---|---|---|---|
| Sequential | 65.0 | 71.7 | 10.133 |
| Random | 66.7 | 73.3 | 8.833 |
| DSDM | 66.7 | 86.7 | 8.800 |
| EMDM | 66.7 | 90.0 | 8.083 |
| REMDM | **73.3** | **91.7** | **7.833** |

found that using the top-5 SLU hypotheses and the target-based state tracking algorithm, the DSDM and the two EMDM systems achieve higher average dialogue successful rate than the two baseline system. Especially for the metric top-5 accuracy, the DSDM and the two EMDM systems outperform the baseline more than 10%. The EMDM systems also outperform all other non-EMDM systems in average dialogue turns, which shows the efficiency of the EMDM strategies. By integrating the user response model, the REMDM system reaches a best successful rate as well as the minimum dialogue turns. A detailed comparison between the REMDM and the EMDM strategy are shown in Table 5. For most of the test cases (80%), the REMDM strategy and the EMDM strategy get same results (both success or both failure). For test cases that the REMDM strategy and the EMDM strategy get different results (one get success but one get failure), the chance of REMDM strategy get success (13.33%) is double than the chance of EMDM strategy (6.67%). Moreover, the REMDM has more efficient dialogues (41.67%) than the EMDM strategy (26.66%). The proposed REMDM strategy is robust to the unavoidable mismatch between the real users' behaviors and the collected user response model.

Table 5: *Comparison between the two EMDM strategies in real user test setting.*

| Metric | #RE<#E | #RE=#E | #RE>#E | Total |
|---|---|---|---|---|
| Top-1 Accu | 6.67% | 80% | 13.33% | 60 |
| Average Turns | 41.67% | 31.67% | 26.66% | 60 |

# 5. Conclusion

In this paper, to extend our probabilistic framework and make it closer to real application scenarios, a user response model is proposed and integrated into the probabilistic framework to optimize the dialogue policy. The experimental results indicate the effectiveness of our probabilistic framework. By integrating the user response model, we can find a more efficient and robust dialogue strategy. Moreover, the framework is amenable to the integration of additional factors that may be relevant in real-world dialogue problems.

[1] n=5 in the entire experiments.

# 6. References


[1] S. Young, J. Schatzmann, K. Weilhammer, and H. Ye, "The hidden information state approach to dialog management," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, vol. 4. IEEE, 2007, pp. IV–149.

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

[3] J. D. Williams, "Incremental partition recombination for efficient tracking of multiple dialog states," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*. IEEE, 2010, pp. 5382–5385.

[4] A. Metallinou, D. Bohus, and J. Williams, "Discriminative state tracking for spoken dialog systems." in *ACL (1)*, 2013, pp. 466–475.

[5] S. Lee and M. Eskenazi, "Recipe for building robust spoken dialog state trackers: Dialog state tracking challenge system description," in *Proceedings of the SIGDIAL 2013 Conference*, 2013, pp. 414–422.

[6] S. Lee, "Structured discriminative model for dialog state tracking," in *Proceedings of the SIGDIAL 2013 Conference*, 2013, pp. 442–451.

[7] H. Ren, W. Xu, Y. Zhang, and Y. Yan, "Dialog state tracking using conditional random fields," in *Proceedings of the SIGDIAL 2013 Conference, Metz, France, August*, 2013.

[8] M. Henderson, B. Thomson, and S. Young, "Word-based dialog state tracking with recurrent neural networks," in *15th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 2014, p. 292.

[9] ——, "Robust dialog state tracking using delexicalised recurrent neural networks and unsupervised adaptation," in *Spoken Language Technology Workshop (SLT), 2014 IEEE*. IEEE, 2014, pp. 360–365.

[10] Z. Wang and O. Lemon, "A simple and generic belief tracking mechanism for the dialog state tracking challenge: On the believability of observed information," in *Proceedings of the SIGDIAL 2013 Conference*, 2013, pp. 423–432.

[11] J. D. Williams and S. Young, "Scaling up POMDPs for Dialog Management: The"Summary POMDP"Method," in *Automatic Speech Recognition and Understanding, 2005 IEEE Workshop on*. IEEE, 2005, pp. 177–182.

[12] ——, "Scaling POMDPs for spoken dialog management," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 7, pp. 2116–2129, 2007.

[13] S. Young, M. Gašić, S. Keizer, F. Mairesse, J. Schatzmann, B. Thomson, and K. Yu, "The hidden information state model: A practical framework for POMDP-based spoken dialogue management," *Computer Speech & Language*, vol. 24, no. 2, pp. 150–174, 2010.

[14] J. Polifroni and M. Walker, "Learning database content for spoken dialogue system design," in *5th International Conference on Language Resources and Evaluation (LREC)*, 2006.

[15] J. Polifroni and M. A. Walker, "Intensional summaries as cooperative responses in dialogue: Automation and evaluation." in *ACL*, 2008, pp. 479–487.

[16] J. Wu, M. Li, and C.-H. Lee, "A probabilistic framework for representing dialog systems and entropy-based dialog management through dynamic stochastic state evolution," *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, vol. 23, no. 11, pp. 2026–2035, Nov 2015.

[17] ——, "An entropy minimization framework for goal-driven dialogue management," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[18] X. Zhang, J. Trmal, D. Povey, and S. Khudanpur, "Improving deep neural network acoustic models using generalized maxout networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 215–219.

[19] Z. Chen, T. Zhang, and J. Wu, "Subword scheme for keyword search," in *Spoken Language Technology Workshop (SLT), 2014 IEEE*. IEEE, 2014, pp. 483–488.

[20] W. Ward and S. Issar, "Recent improvements in the CMU spoken language understanding system," in *Proceedings of the workshop on Human Language Technology*. Association for Computational Linguistics, 1994, pp. 213–216.