Latent Intention Dialogue Models

Tsung-Hsien Wen [1 *] **Yishu Miao** [2 *] **Phil Blunsom** [2] **Steve Young** [1]

Abstract

Developing a dialogue agent that is capable of making autonomous decisions and communicating by natural language is one of the long-term goals of machine learning research. Traditional approaches either rely on hand-crafting a small state-action set for applying reinforcement learning that is not scalable or constructing deterministic models for learning dialogue sentences that fail to capture natural conversational variability. In this paper, we propose a Latent Intention Dialogue Model (LIDM) that employs a discrete latent variable to learn underlying dialogue intentions in the framework of neural variational inference. In a goal-oriented dialogue scenario, these latent intentions can be interpreted as actions guiding the generation of machine responses, which can be further refined autonomously by reinforcement learning. The experimental evaluation of LIDM shows that the model out-performs published benchmarks for both corpus-based and human evaluation, demonstrating the effectiveness of discrete latent variable models for learning goal-oriented dialogues.

1. Introduction

Recurrent neural networks (RNNs) have shown impressive results in modeling generation tasks that have a sequential structured output form, such as machine translation (Sutskever et al., 2014; Bahdanau et al., 2015), caption generation (Karpathy & Fei-Fei, 2015; Xu et al., 2015), and natural language generation (Wen et al., 2015; Kiddon et al., 2016). These discriminative models are trained to learn only a conditional output distribution over strings and despite the sophisticated architectures and conditioning mechanisms used to ensure salience, they are not able to model the underlying actions needed to generate natural dialogues. As a consequence, these sequence-to-sequence models are limited in their ability to exhibit the intrinsic variability and stochasticity of natural dialogue. For example both goal-oriented dialogue systems (Wen et al., 2017; Bordes & Weston, 2017) and sequence-to-sequence learning chatbots (Vinyals & Le, 2015; Shang et al., 2015; Serban et al., 2015) struggle to generate diverse yet causal responses (Li et al., 2016a; Serban et al., 2016). In addition, there is often insufficient training data for goal-oriented dialogues which results in over-fitting which prevents deterministic models from learning effective and scalable interactions. In this paper, we propose a latent variable model – Latent Intention Dialogue Model (LIDM) – for learning the complex distribution of communicative intentions in goal-oriented dialogues. Here, the latent variable representing dialogue intention can be considered as the autonomous decision-making center of a dialogue agent for composing appropriate machine responses.

Recent advances in neural variational inference (Kingma & Welling, 2014; Mnih & Gregor, 2014) have sparked a series of latent variable models applied to NLP (Bowman et al., 2015; Serban et al., 2016; Miao et al., 2016; Cao & Clark, 2017). For models with continuous latent variables, the reparameterisation trick (Kingma & Welling, 2014) is commonly used to build an unbiased and low-variance gradient estimator for updating the models. However, since a continuous latent space is hard to interpret, the major benefits of these models are the stochasticity and the regularisation brought by the latent variable. In contrast, models with discrete latent variables are able to not only produce interpretable latent distributions but also provide a principled framework for semi-supervised learning (Kingma et al., 2014). This is critical for NLP tasks, especially where additional supervision and external knowledge can be utilized for bootstrapping (Faruqui et al., 2015; Miao & Blunsom, 2016; Kočiský et al., 2016). However, variational inference with discrete latent variables is relatively difficult due to the problem of high variance during sampling. Hence we introduce baselines, as in the REINFORCE (Williams, 1992) algorithm, to mitigate the high variance problem, and carry out efficient neural variational inference (Mnih & Gregor, 2014) for the latent variable model.

[*]Equal contribution [1]Department of Engineering, University of Cambridge, Cambridge, United Kingdom [2]Department of Computer Science, University of Oxford, Oxford, United Kingdom. Correspondence to: Tsung-Hsien Wen <thw28@cam.ac.uk>, Yishu Miao <yishu.miao@cs.ox.ac.uk>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, 2017. JMLR: W&CP. Copyright 2017 by the author(s).

In the LIDM, the latent intention is inferred from user input utterances. Based on the dialogue context, the agent draws a sample as the intention which then guides the natural language response generation. Firstly, in the framework of neural variational inference (Mnih & Gregor, 2014), we construct an inference network to approximate the posterior distribution over the latent intention. Then, by sampling the intentions for each response, we are able to directly learn a basic intention distribution on a human-human dialogue corpus by optimising the variational lower bound. To further reduce the variance, we utilize a labeled subset of the corpus in which the labels of intentions are automatically generated by clustering. Then, the latent intention distribution can be learned in a semi-supervised fashion, where the learning signals are either from the direct supervision (labeled set) or the variational lower bound (unlabeled set).

From the perspective of reinforcement learning, the latent intention distribution can be interpreted as the intrinsic policy that reflects human decision-making under a particular conversational scenario. Based on the initial policy (latent intention distribution) learnt from the semi-supervised variational inference framework, the model can refine its strategy easily against alternative objectives using policy gradient-based reinforcement learning. This is somewhat analagous to the training process used in AlphaGo (Silver et al., 2016) for the game of Go. Based on LIDM, we show that different learning paradigms can be brought together under the same framework to bootstrap the development of a dialogue agent (Li et al., 2016c; Weston, 2016).

In summary, the contribution of this paper is two-fold: firstly, we show that the neural variational inference framework is able to discover discrete, interpretable intentions from data to form the decision-making basis of a dialogue agent; secondly, the agent is capable of revising its conversational strategy based on an external reward within the same framework. This is important because it provides a stepping stone towards building an autonomous dialogue agent that can continuously improve itself through interaction with users. The experimental results demonstrate the effectiveness of our latent intention model which achieves state-of-the-art performance on both automatic corpus-based evaluation and human evaluation.

2. Latent Intention Dialogue Model for Goal-oriented Dialogue

Goal-oriented dialogue[1] (Young et al., 2013) aims at building models that can help users to complete certain tasks via natural language interaction. Given a user input utterance u_t at turn t and a knowledge base (KB), the model needs to parse the input into actionable commands Q and access the KB to search for useful information in order to answer the query. Based on the search result, the model needs to summarise its findings and reply with an appropriate response m_t in natural language.

2.1. Model

The LIDM is based on the end-to-end system architecture described in (Wen et al., 2017). It comprises three components: (1) Representation Construction; (2) Policy Network; and (3) Generator, as shown in Figure 1. To capture the user's intent and match it against the system's knowledge, a dialogue state vector $\mathbf{s}_t = \mathbf{u}_t \oplus \mathbf{b}_t \oplus \mathbf{x}_t$ is derived from the user input u_t and the knowledge base KB: $\mathbf{u}_t$ is the distributed utterance representation, which is formed by encoding the user utterance[2] u_t with a bidirectional LSTM (Hochreiter & Schmidhuber, 1997) and concatenating the final stage hidden states together,

$$\mathbf{u}_t = \text{biLSTM}_\Theta(u_t). \tag{1}$$

The belief vector $\mathbf{b}_t$, which is a concatenation of a set of probability distributions over domain specific slot-value pairs, is extracted by a set of pre-trained RNN-CNN belief trackers (Wen et al., 2017; Henderson et al., 2014), in which u_t and m_{t-1} are processed by two different CNNs as shown in Figure 1,

$$\mathbf{b}_t = \text{RNN-CNN}(u_t, m_{t-1}, \mathbf{b}_{t-1}) \tag{2}$$

where m_{t-1} is the preceding machine response and $\mathbf{b}_{t-1}$ is the preceding belief vector. They are included to model the current turn of the discourse and the long-term dialogue context, respectively. Based on the belief vector, a query Q is formed by taking the union of the maximum values of each slot. Q is then used to search the internal KB and return a vector $\mathbf{x}_t$ representing the degree of matching in the KB. This is produced by counting all the matching venues and re-structuring it into a six-bin one-hot vector. Among the three vectors that comprise the dialogue state $\mathbf{s}_t$, $\mathbf{u}_t$ is completely trainable from data, $\mathbf{b}_t$ is pre-trained using a separate objective function, and $\mathbf{x}_t$ is produced by a discrete database accessing operation. For more details about the belief trackers and database operation refer to Wen et al (2016; 2017).

Conditioning on the state $\mathbf{s}_t$, the policy network parameterises the latent intention z_t by a single layer MLP,

$$\pi_\Theta(z_t|\mathbf{s}_t) = \text{softmax}(\mathbf{W}_2^\intercal \cdot \tanh(\mathbf{W}_1^\intercal \mathbf{s}_t + \mathbf{b}_1) + \mathbf{b}_2) \tag{3}$$

where $\mathbf{W}_1$, $\mathbf{b}_1$, $\mathbf{W}_2$, $\mathbf{b}_2$ are model parameters. Since $\pi_\Theta(z_t|\mathbf{s}_t)$ is a discrete conditional probability distribution

[1] Like most of the goal-oriented dialogue research, we focus on information seek type dialogues.

[2] All sentences are pre-processed by delexicalisation (Henderson et al., 2014) where slot-value specific words are replaced with their corresponding generic tokens based on an ontology.

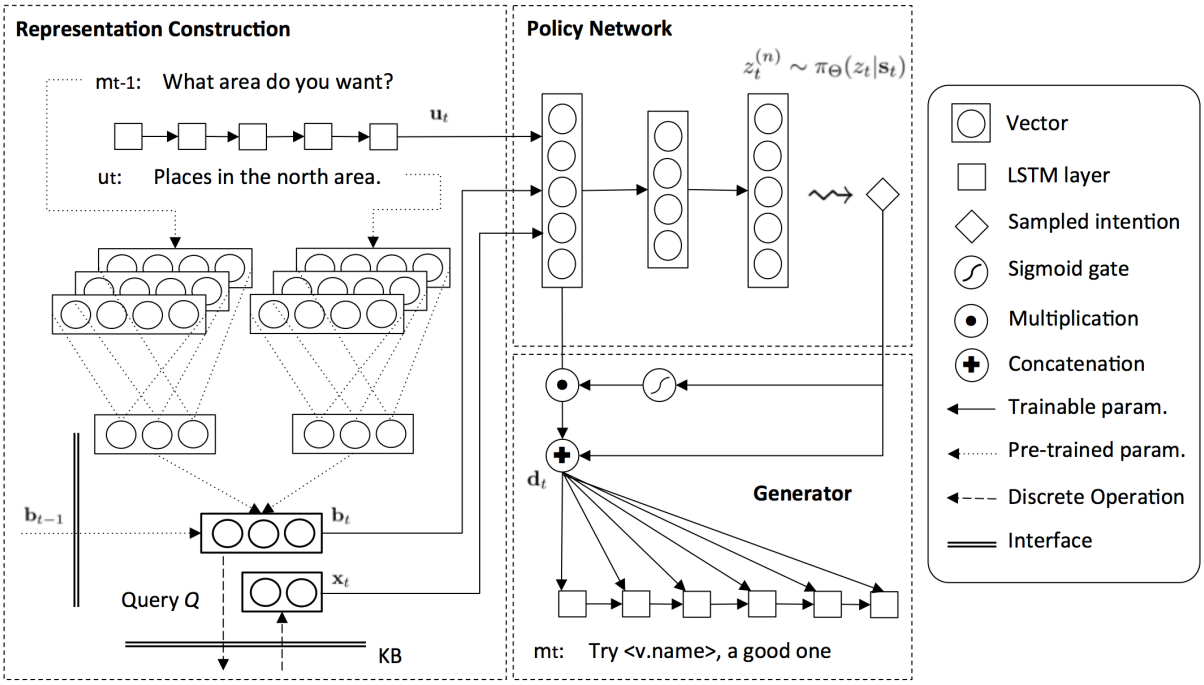

Figure 1. LIDM for Goal-oriented Dialogue Modeling

based on dialogue state, we can also interpret the policy network here as a latent dialogue management component in the traditional POMDP-based framework (Young et al., 2013; Gašić et al., 2013). A latent intention $z_t^{(n)}$ (or an action in the reinforcement learning literature) can then be sampled from the conditional distribution,

$$z_t^{(n)} \sim \pi_\Theta(z_t|\mathbf{s}_t). \tag{4}$$

This sampled intention (or action) $z_t^{(n)}$ and the state vector $\mathbf{s}_t$ can then be combined into a control vector $\mathbf{d}_t$, which is used to govern the generation of the system response based on a conditional LSTM language model,

$$\mathbf{d}_t = \mathbf{W}_4^\intercal \mathbf{z}_t \oplus [\text{sigmoid}(\mathbf{W}_3^\intercal \mathbf{z}_t + \mathbf{b}_3) \cdot \mathbf{W}_5^\intercal \mathbf{s}_t] \tag{5}$$

$$p_\Theta(m_t|\mathbf{s}_t, z_t) = \prod_j p(w_{j+1}^t|w_j^t, \mathbf{h}_{j-1}^t, \mathbf{d}_t) \tag{6}$$

where $\mathbf{b}_3$ and $\mathbf{W}_{3\sim5}$ are parameters, $\mathbf{z}_t$ is the 1-hot representation of $z_t^{(n)}$, w_j^t is the last output token (i.e. a word, a delexicalised[2] slot name or a delexicalised[2] slot value), and $\mathbf{h}_{j-1}^t$ is the decoder's last hidden state. Note in Equation 5 the degree of information flow from the state vector is controlled by a sigmoid gate whose input signal is the sampled intention $z_t^{(n)}$. This prevents the decoder from overfitting to the deterministic state information and forces it to take the sampled stochastic intention into account. The LIDM can then be formally written down in its parameterised form with parameter set Θ,

$$p_\Theta(m_t|\mathbf{s}_t) = \sum_{z_t} p_\Theta(m_t|z_t, \mathbf{s}_t)\pi_\Theta(z_t|\mathbf{s}_t). \tag{7}$$

2.2. Inference

To carry out inference for the LIDM, we introduce an inference network $q_\Phi(z_t|\mathbf{s}_t, m_t)$ to approximate the posterior distribution $p(z_t|\mathbf{s}_t, m_t)$ so that we can optimise the variational lower bound of the joint probability in a neural variational inference framework (Miao et al., 2016). We can then derive the variational lower bound as,

$$\begin{aligned}\mathcal{L} &= \mathbb{E}_{q_\Phi(z_t)}[\log p_\Theta(m_t|z_t, \mathbf{s}_t)] - \lambda D_{KL}(q_\Phi(z_t)||\pi_\Theta(z_t|\mathbf{s}_t)) \\ &\leq \log \sum_{z_t} p_\Theta(m_t|z_t, \mathbf{s}_t)\pi_\Theta(z_t|\mathbf{s}_t) \\ &= \log p_\Theta(m_t|\mathbf{s}_t)\end{aligned} \tag{8}$$

where $q_\Phi(z_t)$ is a shorthand for $q_\Phi(z_t|\mathbf{s}_t, m_t)$. Note that we use a modified version of the lower bound here by incorporating a trade-off factor λ (Higgins et al., 2017). The inference network $q_\Phi(z_t|\mathbf{s}_t, m_t)$ is then constructed by

$$q_\Phi(z_t|\mathbf{s}_t, m_t) = \text{Multi}(\mathbf{o}_t) = \text{softmax}(\mathbf{W}_6\mathbf{o}_t) \tag{9}$$

$$\mathbf{o}_t = \text{MLP}_\Phi(\mathbf{b}_t, \mathbf{x}_t, \mathbf{u}_t, \mathbf{m}_t) \tag{10}$$

$$\mathbf{u}_t = \text{biLSTM}_\Phi(u_t), \mathbf{m}_t = \text{biLSTM}_\Phi(m_t) \tag{11}$$

where $\mathbf{o}_t$ is the joint representation, and both $\mathbf{u}_t$ and $\mathbf{m}_t$ are modeled by a bidirectional LSTM network. Although both

$q_\Phi(z_t|\mathbf{s}_t, m_t)$ and $\pi_\Theta(z_t|\mathbf{s}_t)$ are modelled as parameterised multinomial distributions, the approximation $q_\Phi(z_t|\mathbf{s}_t, m_t)$ only functions during inference by producing samples to compute the stochastic gradients, while $\pi_\Theta(z_t|\mathbf{s}_t)$ is the generative distribution that generates the required samples for composing the machine response.

Based on the samples $z_t^{(n)} \sim q_\Phi(z_t|\mathbf{s}_t, m_t)$, we use different strategies to alternately optimise the parameters Θ and Φ against the variational lower bound (Equation 8). To do this, we further divide Θ into two sets $\Theta = \{\Theta_1, \Theta_2\}$. Parameters Θ_1 on the decoder side are directly updated by back-propagating the gradients,

$$\frac{\partial \mathcal{L}}{\partial \Theta_1} = \mathbb{E}_{q_\Phi(z_t|\mathbf{s}_t, m_t)}[\frac{\partial \log p_{\Theta_1}(m_t|z_t, \mathbf{s}_t)}{\partial \Theta_1}] \approx \frac{1}{N}\sum_n \frac{\partial \log p_{\Theta_1}(m_t|z_t^{(n)}, \mathbf{s}_t)}{\partial \Theta_1}. \quad (12)$$

Parameters Θ_2 in the generative network, however, are updated by minimising the KL divergence,

$$\frac{\partial \mathcal{L}}{\partial \Theta_2} = -\frac{\partial \lambda D_{KL}(q_\Phi(z_t|\mathbf{s}_t, m_t)||\pi_{\Theta_2}(z_t|\mathbf{s}_t))}{\partial \Theta_2} = -\lambda \sum_{z_t} q_\Phi(z_t|\mathbf{s}_t, m_t)\frac{\partial \log \pi_{\Theta_2}(z_t|\mathbf{s}_t)}{\partial \Theta_2} \quad (13)$$

where the entropy derivative $\partial H[q_\Phi(z_t|\mathbf{s}_t, m_t)]/\partial \Theta_2 = 0$ and therefore can be ignored. Finally, for the parameters Φ in the inference network, we firstly define the learning signal $r(m_t, z_t^{(n)}, \mathbf{s}_t)$,

$$r(m_t, z_t^{(n)}, \mathbf{s}_t) = \log p_{\Theta_1}(m_t|z_t^{(n)}, \mathbf{s}_t) - \lambda(\log q_\Phi(z_t^{(n)}|\mathbf{s}_t, m_t) - \log \pi_{\Theta_2}(z_t^{(n)}|\mathbf{s}_t)). \quad (14)$$

Then the parameters Φ are updated by,

$$\frac{\partial \mathcal{L}}{\partial \Phi} = \mathbb{E}_{q_\Phi(a_t|\mathbf{s}_t, m_t)}[r(m_t, a_t, \mathbf{s}_t)\frac{\partial \log q_\Phi(a_t|\mathbf{s}_t, m_t)}{\partial \Phi}] \approx \frac{1}{N}\sum_n r(m_t, z_t^{(n)}, \mathbf{s}_t)\frac{\partial \log q_\Phi(z_t^{(n)}|\mathbf{s}_t, m_t)}{\partial \Phi}. \quad (15)$$

However, this gradient estimator has a large variance because the learning signal $r(m_t, z_t^{(n)}, \mathbf{s}_t)$ relies on samples from the proposal distribution $q_\Phi(z_t|\mathbf{s}_t, m_t)$. To reduce the variance during inference, we follow the REINFORCE algorithm (Mnih et al., 2014; Mnih & Gregor, 2014) and introduce two baselines b and $b(\mathbf{s}_t)$, the centered learning signal and input dependent baseline respectively to help reduce the variance. b is a learnable constant and $b(\mathbf{s}_t) = \text{MLP}(\mathbf{s}_t)$. During training, the two baselines are updated by minimising the distance,

$$\mathcal{L}_b = \Big[r(m_t, z_t^{(n)}, \mathbf{s}_t) - b - b(\mathbf{s}_t)\Big]^2 \quad (16)$$

and the gradient w.r.t. Φ can be rewritten as

$$\frac{\partial \mathcal{L}}{\partial \Phi} \approx \frac{1}{N}\sum_n [r(m_t, z_t^{(n)}, \mathbf{s}_t) - b - b(\mathbf{s}_t)]\frac{\partial \log q_\Phi(z_t^{(n)}|\mathbf{s}_t, m_t)}{\partial \Phi}. \quad (17)$$

2.3. Semi-Supervision

Despite the steps described above for reducing the variance, there remain two major difficulties in learning latent intentions in a completely unsupervised manner: (1) the high variance of the inference network prevents it from generating sensible intention samples in the early stages of training, and (2) the overly strong discriminative power of the LSTM language model is prone to the *disconnection* phenomenon between the LSTM decoder and the rest of the components whereby the decoder learns to ignore the samples and focuses solely on optimising the language model. To ensure more stable training and prevent disconnection, a semi-supervised learning technique is introduced.

Inferring the latent intentions underlying utterances is similar to an unsupervised clustering task. Standard clustering algorithms can therefore be used to pre-process the corpus and generate automatic labels $\hat{z}_t$ for part of the training examples $(m_t, \mathbf{s}_t, \hat{z}_t) \in \mathbb{L}$. Then when the model is trained on the unlabeled examples $(m_t, \mathbf{s}_t) \in \mathbb{U}$, we optimise it against the modified variational lower bound given in Equation 8

$$\mathcal{L}_1 = \sum_{(m_t, \mathbf{s}_t) \in \mathbb{U}} \mathbb{E}_{q_\phi(z_t|\mathbf{s}_t, m_t)}\left[\log p_\theta(m_t|z_t, \mathbf{s}_t)\right] - \lambda D_{KL}(q_\phi(z_t|\mathbf{s}_t, m_t)||\pi_\theta(z_t|\mathbf{s}_t)) \quad (18)$$

However, when the model is updated based on examples from the labeled set $(m_t, \mathbf{s}_t, \hat{z}_t) \in \mathbb{L}$, we treat the labeled intention $\hat{z}_t$ as an observed variable and train the model by maximising the joint log-likelihood,

$$\mathcal{L}_2 = \sum_{(m_t, \hat{z}_t, \mathbf{s}_t) \in \mathbb{L}} \log\left[p_\Theta(m_t|\hat{z}_t, \mathbf{s}_t)\pi_\Theta(\hat{z}_t|\mathbf{s}_t)q_\Phi(\hat{z}_t|\mathbf{s}_t, m_t)\right] \quad (19)$$

The final joint objective function can then be written as $\mathcal{L}' = \alpha\mathcal{L}_1 + \mathcal{L}_2$, where α controls the trade-off between the supervised and unsupervised examples.

2.4. Reinforcement Learning

One of the main purposes of learning interpretable, discrete latent intention inside a dialogue system is to be able to control and refine the model's behaviour with operational experience. The learnt generative network $\pi_\Theta(z_t|\mathbf{s}_t)$ encodes the policy discovered from the underlying data distribution but this is not necessarily optimal for any specific task. Since $\pi_\Theta(z_t|\mathbf{s}_t)$ is a parameterised policy network itself, any policy gradient-based reinforcement learning algorithm (Williams, 1992; Konda & Tsitsiklis, 2003) can be

used to fine-tune the initial policy against other objective functions that we are more interested in.

Based on the initial policy $\pi_\Theta(z_t|\mathbf{s}_t)$, we revisit the training dialogues and update parameters based on the following strategy: when encountering unlabeled examples $\mathbb{U}$ at turn t the system samples an action from the learnt policy $z_t^{(n)} \sim \pi_\Theta(z_t|\mathbf{s}_t)$ and receives a reward $r_t^{(n)}$. Conditioning on these, we can directly fine-tune a subset of the model parameters Θ' by the policy gradient method,

$$\frac{\partial \mathcal{J}}{\partial \Theta'} \approx \frac{1}{N} \sum_n r_t^{(n)} \frac{\partial \log \pi_\Theta(z_t^{(n)}|\mathbf{s}_t)}{\partial \Theta'} \tag{20}$$

where $\Theta' = \{\mathbf{W}_1, \mathbf{b}_1, \mathbf{W}_2, \mathbf{b}_2\}$ is the MLP that parameterises the policy network (Equation 3). However, when a labeled example $\in \mathbb{L}$ is encountered we force the model to take the labeled action $z_t^{(n)} = \hat{z}_t$ and update the parameters by Equation 20 as well. Unlike Li et al (2016c) where the whole model is refined end-to-end using RL, updating only Θ' effectively allows us to refine only the *decision-making* of the system and avoid the problem of over-fitting.

3. Experiments

3.1. Dataset & Setup for Goal-oriented Dialogue

We explored the properties of the LIDM model[3] using the CamRest676 corpus[4] collected by Wen et al (2017), in which the task of the system is to assist users to find a restaurant in the Cambridge, UK area. The corpus was collected based on a modified Wizard of Oz (Kelley, 1984) online data collection. Workers were recruited on Amazon Mechanical Turk and asked to complete a task by carrying out a conversation, alternating roles between a user and a wizard. There are three informable slots (*food*, *pricerange*, *area*) that users can use to constrain the search and six requestable slots (*address*, *phone*, *postcode* plus the three informable slots) that the user can ask a value for once a restaurant has been offered. There are 676 dialogues in the dataset (including both finished and unfinished dialogues) and approximately 2750 conversational turns in total. The database contains 99 unique restaurants.

To make a direct comparison with prior work we follow the same experimental setup as in Wen et al (2016; 2017). The corpus was partitioned into training, validation, and test sets in the ratio 3:1:1. The LSTM hidden layer sizes were set to 50, and the vocabulary size is around 500 after pre-processing, to remove rare words and words that can be delexicalised[2]. All the system components were trained jointly by fixing the pre-trained belief trackers and the discrete database operator with the model's latent intention

[3]Will be available at https://github.com/shawnwun/NNDIAL

[4]https://www.repository.cam.ac.uk/handle/1810/260970

ID	#	content words
0	138	thank, goodbye
1	91	welcome, goodbye
3	42	phone, address, [v.phone],[v.address]
14	17	address, [v.address]
31	9	located, area, [v.area]
34	9	area, would, like
46	7	food, serving, restaurant, [v.food]
85	4	help, anything, else

Table 1. An example of the automatically labeled response seed set for semi-supervised learning during variational inference.

size I set to 50, 70, and 100, respectively. The trade-off constants λ and α were both set to 0.1. To produce self-labeled response clusters for semi-supervised learning of the intentions, we firstly removed function words from all the responses and clustered them according to their content words. We then assigned the responses in the i-th frequent cluster to the i-th latent dimension as its supervised set. This results in about 35% ($I = 50$) to 43% ($I = 100$) labeled responses across the whole dataset. An example of the resulting seed set is shown in Table 1. During inference we carried out stochastic estimation by taking one sample for estimating the stochastic gradients. The model is trained by Adam (Kingma & Ba, 2014) and tuned (early stopping, hyper-parameters) on the held-out validation set. We alternately optimised the generative model and the inference network by fixing the parameters of one while updating the parameters of the other.

During reinforcement fine-tuning, we generated a sentence m_t from the model to replace the ground truth $\hat{m}_t$ at each turn and define an immediate reward as whether m_t can improve the dialogue success (Su et al., 2015) relative to $\hat{m}_t$, plus the sentence BLEU score (Auli & Gao, 2014),

$$r_t = \eta \cdot \text{sBLEU}(m_t, \hat{m}_t) + \begin{cases} 1 & m_t \text{ improves} \\ -1 & m_t \text{ degrades} \\ 0 & \text{otherwise} \end{cases} \tag{21}$$

where the constant η was set to 0.5. We fine-tuned the model parameters using RL for only 3 epochs. During testing, we greedily selected the most probable intention and applied beam search with the beamwidth set to 10 when decoding the response. The decoding criterion was the average log-probability of tokens in the response. We then evaluated our model on task success rate (Su et al., 2015) and BLEU score (Papineni et al., 2002) as in Wen et al (2016; 2017) in which the model is used to predict each system response in the held-out test set.

Model	Success (%)	BLEU
Ground Truth		
Ground Truth	91.6	1.000
Published Models (Wen et al., 2016)		
NDM	76.1	0.212
NDM+Att	79.0	0.224
NDM+Att+SS	81.8	0.240
LIDM Models		
LIDM, $I = 50$	66.9	0.238
LIDM, $I = 70$	61.0	**0.246**
LIDM, $I = 100$	63.2	0.242
LIDM Models + RL		
LIDM, $I = 50$, +RL	82.4	0.231
LIDM, $I = 70$, +RL	81.6	0.230
LIDM, $I = 100$, +RL	**84.6**	0.240

Table 2. Corpus-based Evaluation.

3.2. Experiments on Goal-oriented Dialogue

Table 2 presents the results of the corpus-based evaluation. The *Ground Truth* block shows the two metrics when we compute them on the human-authored responses. This sets a gold standard for the task. In the *Published Models* block, the results for the three baseline models were borrowed from Wen et al (2016), they are: (1) the vanilla neural dialogue model (NDM), (2) NDM plus an attention mechanism on the belief trackers, and (3) the attentive NDM with self-supervised sub-task neurons. The results of the LIDM model with and without RL fine-tuning are shown in the *LIDM Models* and the *LIDM Models + RL* blocks, respectively. As can be seen, the initial policy learned by fitting the latent intention to the underlying data distribution yielded reasonably good results on BLEU but did not perform well on task success when compared to their deterministic counterparts (*block 2 v.s. 3*). This may be due to the fact that the variational lower bound of the dataset was optimised rather than task success during variational inference. However, once RL was applied to optimise the success rate as part of the reward function (Equation 21) during the fine-tuning phase, the resulting LIDM+RL models outperformed the three baselines in terms of task success without significantly sacrificing BLEU (*block 2 v.s. 4*)[5].

In order to assess the human perceived performance, we evaluated the three models (1) NDM, (2) LIDM, and (3) LIDM+RL by recruiting paid subjects on Amazon Mechanical Turk. Each judge was asked to follow a task and carried out a conversation with the machine. At the end of each conversation the judges were asked to rate and compare the model's performance. We assessed the subjective success rate, the perceived comprehension ability and the naturalness of responses on a scale of 1 to 5. For each model, we collected 200 dialogues and averaged the scores. During human evaluation, we sampled from the top-5 intentions of the LIDM models and decoded a response based on the sample. The result is shown in Table 3. One interesting fact to note is that although the LIDM did not perform well on the corpus-based task success metric, the human judges rated its subjective success almost indistinguishably from the others. This discrepancy between the two experiments arises mainly from a flaw in the corpus-based success metric in that it favors greedy policies because the user side behaviours are fixed rather than interactional[6]. Despite the fact that LIDMs are considered only marginally better than NDM on subjective success, the LIDMs do outperform NDM on both comprehension and naturalness scores. This is because the proposed LIDM models can better capture multiple modes in the communicative intention and thereby respond more naturally by sampling from the latent intention variable.

Metrics	NDM	LIDM	LIDM+RL
Success	91.5%	92.0%	93.0%
Comprehension	4.21	4.40*	4.40
Naturalness	4.08	4.29*	4.28*
# of Turns	4.45	4.54	4.29

* p <0.05

Table 3. Human evaluation. The significance test is based on a two-tailed student-t test, between NDM and LIDMs.

Three example conversations are shown between a human judge and a machine, one from LIDM in Table 4 and two from LIDM+RL in Table 5, respectively. The results are displayed one exchange per block. Each induced latent intention is shown by a tuple (index, probability) followed by a decoded response, and the sample dialogues were produced by following the responses highlighted in bold. As can be seen, the LIDM shown in Table 4 clearly has multiple modes in the distribution over the learned intention latent variable, and what it represents can be easily interpreted by the response generated. However, some intentions (such as *intent 0*) can result in very different responses under different dialogue states even though they were supervised by a small response set as shown in Table 1. This is mainly because of the variance introduced during variational inference. Finally, when comparing Table 4 and Table 5, we can observe the difference between the two dialogue strategies: the LIDM, by inferring its policy from the supervised dataset, reflects the diverse set of modes in the underlying distribution; whereas the LIDM+RL, which

[5]Note that both NDM+Att+SS and LIDM use self-supervised information

[6]The system tries to provide as much information as possible in the early turns, in case the fixed user side behaviours a few turns later do not fit the scenario the system originally planned.

Human :	Hi, I'm hungry for some Indonesian		
Machine:	**(0**	**0.71)**	**i am sorry , there are no indonesian restaurants in the area . would you like something else ?**
	(46	0.19)	there are no restaurants serving indonesian kind of food .
	(34	0.07)	what location would you like ?
Human :	Oh no. How about Indian?		
Machine:	(0	0.59)	there are 9 restaurants in the east side of town . would you like their address ?
	(30	0.15)	curry prince serves indian cuisine .
	(34	**0.10)**	**what area would you like ?**
Human :	East side please		
Machine:	**(0**	**0.62)**	**curry prince is a indian restaurant in the east side of town . would you like their telephone number ?**
	(84	0.19)	curry prince is an indian restaurant in the east of town and the number is 01223 566388 .
	(81	0.08)	curry prince serves indian type of cuisine and is located in the east part of town .
Human :	yes , the address and phone number please		
Machine:	(3	0.46)	the address is 451 newmarket road fen ditton and the phone is 01223 566388 .
	(11	**0.16)**	**the address for curry prince is 451 newmarket road fen ditton and the phone number is 01223 566388 .**
	(0	0.15)	curry prince is located at 451 newmarket road fen ditton and their phone is 01223 566388 . is there anything else i can help you with ?
Human :	Okay, thank you goodbye		
Machine:	(0	0.57)	thank you , goodbye .
	(2	0.13)	goodbye .
	(1	**0.11)**	**you are welcome . goodbye .**
	(6	0.10)	thank you for using the cambridge restaurant system . goodbye .

Table 4. A sample dialogue from the LIDM, I=100 model, one exchange per block. Each latent intention is shown by a tuple (index, probability), followed by a decoded response. The sample dialogue was produced by following the responses highlighted in bold.

refined its strategy using RL, exhibits a much greedier behavior in achieving task success (e.g. in *Table 5 in block 2 & 4* the LIDM+RL agent provides the address and phone number even before the user asks). This is also supported by the human evaluation in Table 3 where LIDM+RL has much shorter dialogues on average compared to the other two models.

4. Discussion

Learning an end-to-end dialogue system is appealing but challenging because of the credit assignment problem. Discrete latent variable dialogue models such as LIDM are attractive because the latent variable can serve as an interface for decomposing the learning of language and the internal dialogue decision-making. This decomposition can effectively help us resolve the credit assignment problem where different learning signals can be applied to different sub-modules to update the parameters. In variational inference for discrete latent variables, the latent distribution is basically updated by the reward from the variational lower bound. While in reinforcement learning, the latent distribution (i.e. policy network) is updated by the rewards from dialogue success and sentence BLEU score. Hence, the latent variable bridges the different learning paradigms such as Bayesian learning and reinforcement learning and brings them together under the same framework. This framework provides a more robust neural network-based approach than previous approaches because it does not depend solely on sequence-to-sequence learning but instead explicitly models the hidden dialogue intentions underlying the user's utterances and allows the agent to directly learn a dialogue policy through interaction.

5. Related work

Modeling chat-based dialogues (Serban et al., 2015; Shang et al., 2015) as a sequence-to-sequence learning (Sutskever et al., 2014) problem is a common theme in the deep learning community. Vinyals and Le (2015) has demonstrated a seq2seq-based model trained on a huge amount of conversation corpora which learns interesting replies conditioned on different user queries. However, due to an inability to model dialogue context, these models generally suffer from the generic response problem (Li et al., 2016a; Serban et al., 2016). Several approaches have been proposed to mitigate this issue, such as modeling the persona (Li et al., 2016b), reinforcement learning (Li et al., 2016c), and introducing continuous latent variables (Serban et al., 2016; Cao & Clark, 2017). While in our case, we not only make use of the latent variable to inject stochasticity for generating natural and diverse machine responses but also model the hidden dialogue intentions explicitly. This combines the merits of reinforcement learning and generative models.

At the other end of the spectrum, goal-oriented dialogue systems typically adopt the POMDP framework (Young et al., 2013) and break down the development of the dialogue systems into a pipeline of modules: natural language understanding (Henderson, 2015), dialogue manage-

Human :	Hi, I'm hungry for some Indonesian .		
Machine:	**(0**	**0.61)**	**i am sorry , there are no indonesian restaurants in the area . would you like something else ?**
	(46	0.17)	no restaurants in the part of town .
	(34	0.14)	what part of town would you like ?
Human :	Oh no. How about Indian?		
Machine:	**(93**	**0.51)**	**the gandhi is located at 72 regent street city centre and their telephone number is 01223 353942 .**
	(72	0.21)	the gandhi serves indian kind of food and is in the centre of town and is in the cheap price range .
	(12	0.12)	the gandhi is a indian restaurant in the centre location and is in the cheap price range .
Human :	Okay, thank you goodbye		
Machine:	**(6**	**0.28)**	**thank you for using the cambridge restaurant system . goodbye .**
	(1	0.24)	you are welcome . goodbye .
	(85	0.19)	is there anything else i can help you with ?
Human :	hey can i have some chinese cruisine in the centre ? and a cheap one please		
Machine:	(15	0.42)	golden house is located at 12 lensfield road city centre and their number is 01842 753771 .
	(5	0.38)	**golden house is located at 12 lensfield road city centre and the number is 01842 753771 . is there ...**
	(93	0.10)	golden house is located at 12 lensfield road city centre and their phone is 01842 753771 .
Human :	that is all		
Machine:	**(6**	**0.57)**	**thank you for using the cambridge restaurant system . goodbye .**
	(1	0.20)	you are welcome . goodbye .
	(17	0.11)	have a nice day !

Table 5. Two sample dialogues from the LIDM+RL, I=100 model, one exchange per block. Comparing to Table 4, the RL agent demonstrates a much greedier behavior toward task success. This can be seen in *block 2 & block 4* in which the agent provides the address and phone number even before the user asks.

ment (Gašić et al., 2013), and natural language generation (Wen et al., 2015). These system modules communicate through a dialogue act formalism (Traum, 1999), which in effect constitute a fixed set of handcrafted intentions. This limits the ability of such systems to scale to more complex tasks. In contrast, the LIDM directly infers all underlying dialogue intentions from data and can handle intention distributions with long tails by measuring similarities against the existing ones during variational inference. Modeling of end-to-end goal-oriented dialogue systems has also been studied recently (Wen et al., 2016; 2017; Bordes & Weston, 2017), however, these models are typically deterministic and rely on decoder supervision signals to finetune a large set of model parameters.

Much research has focused on combining different learning paradigms and signals to bootstrap performance. For example, semi-supervised learning (Kingma et al., 2014) has been applied in the sample-based neural variational inference framework as a way to reduce sample variance. In practice, this relies on a discrete latent variable (Miao & Blunsom, 2016; Kočiský et al., 2016) as the vehicle for the supervision labels. As in reinforcement learning, which has been a very common learning paradigm in dialogue systems (Gašić et al., 2013; Su et al., 2016; Li et al., 2017), the policy is also parameterised by a discrete set of actions. As a consequence, the LIDM, which parameterises the intention space via a discrete latent variable, can automatically enjoy the benefit of bootstrapping from signals coming from different learning paradigms. In addition, self-supervised learning (Snow et al., 2004) (or distant, weak supervision) as a simple way to generate automatic labels by heuristics is popular in many NLP tasks and has been applied to memory networks (Hill et al., 2016) and neural dialogue systems (Wen et al., 2016) recently. Since there is no additional effort required in labeling, it can also be viewed as a method for bootstrapping.

6. Conclusion

In this paper, we have proposed a framework for learning dialogue intentions via discrete latent variable models and introduced the Latent Intention Dialogue Model (LIDM) for goal-oriented dialogue modeling. We have shown that the LIDM can discover an effective initial policy from the underlying data distribution and is capable of revising its strategy based on an external reward using reinforcement learning. We believe this is a promising step forward for building autonomous dialogue agents since the learnt discrete latent variable interface enables the agent to perform learning using several differing paradigms. The experiments showed that the proposed LIDM is able to communicate with human subjects and outperforms previous published results.

Acknowledgements

Tsung-Hsien Wen is supported by Toshiba Research Europe Ltd, Cambridge Research Laboratory. The authors would like to thank the members of the Cambridge Dialogue Systems Group for their valuable comments.

References

Auli, Michael and Gao, Jianfeng. Decoder integration and expected bleu training for recurrent neural network language models. In *ACL*. Association for Computational Linguistics, 2014.

Bahdanau, Dzmitry, Cho, Kyunghyun, and Bengio, Yoshua. Neural machine translation by jointly learning to align and translate. *ICLR*, 2015.

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

Bowman, Samuel R., Vilnis, Luke, Vinyals, Oriol, Dai, Andrew M., Józefowicz, Rafal, and Bengio, Samy. Generating sentences from a continuous space. *arXiv preprint:*, 2015.

Cao, Kris and Clark, Stephen. Latent variable dialogue models and their diversity. In *EACL*, 2017.

Faruqui, Manaal, Dodge, Jesse, Jauhar, Sujay Kumar, Dyer, Chris, Hovy, Eduard, and Smith, Noah A. Retrofitting word vectors to semantic lexicons. In *NAACL-HLT*, pp. 1606–1615, Denver, Colorado, May–June 2015. Association for Computational Linguistics.

Gašić, Milica, Breslin, Catherine, Henderson, Matthew, Kim, Dongho, Szummer, Martin, Thomson, Blaise, Tsiakoulis, Pirros, and Young, Steve. On-line policy optimisation of bayesian spoken dialogue systems via human interaction. In *ICASSP*, 2013.

Henderson, Matthew. Machine learning for dialog state tracking: A review. In *Machine Learning in Spoken Language Processing Workshop*, 2015.

Henderson, Matthew, Thomson, Blaise, and Young, Steve. Word-based dialog state tracking with recurrent neural networks. In *SigDial*, pp. 292–299, Philadelphia, PA, U.S.A., June 2014. Association for Computational Linguistics.

Higgins, Irina, Matthey, Loic, Pal, Arka, Burgess, Christopher, Glorot, Xavier, Botvinick, Matthew, Mohamed, Shakir, and Lerchner, Alexander. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017.

Hill, Felix, Bordes, Antoine, Chopra, Sumit, and Weston, Jason. The goldilocks principle: Reading children's books with explicit memory representations. In *ICLR*, 2016.

Hochreiter, Sepp and Schmidhuber, Jürgen. Long short-term memory. *Neural Computation*, 1997.

Karpathy, Andrej and Fei-Fei, Li. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015.

Kelley, John F. An iterative design methodology for user-friendly natural language office information applications. *ACM Transaction on Information Systems*, 1984.

Kiddon, Chloé, Zettlemoyer, Luke, and Choi, Yejin. Globally coherent text generation with neural checklist models. In *EMNLP*, pp. 329–339, Austin, Texas, November 2016. Association for Computational Linguistics.

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

Kingma, Diederik P. and Welling, Max. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, 2014.

Kingma, Diederik P., Mohamed, Shakir, Rezende, Danilo J., and Welling, Max. Semi-supervised learning with deep generative models. In *NIPS*, pp. 3581–3589. Curran Associates, Inc., 2014.

Konda, Vijay R. and Tsitsiklis, John N. On actor-critic algorithms. *SIAM J. Control Optim.*, 42(4): 1143–1166, April 2003. ISSN 0363-0129. doi: 10.1137/S0363012901385691.

Kočiský, Tomáš, Melis, Gábor, Grefenstette, Edward, Dyer, Chris, Ling, Wang, Blunsom, Phil, and Hermann, Karl Moritz. Semantic parsing with semi-supervised sequential autoencoders. In *EMNLP*, pp. 1078–1087, Austin, Texas, November 2016. Association for Computational Linguistics.

Li, Jiwei, Galley, Michel, Brockett, Chris, Gao, Jianfeng, and Dolan, Bill. A diversity-promoting objective function for neural conversation models. In *NAACL-HLT*, pp. 110–119, San Diego, California, June 2016a. Association for Computational Linguistics.

Li, Jiwei, Galley, Michel, Brockett, Chris, Spithourakis, Georgios, Gao, Jianfeng, and Dolan, Bill. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 994–1003, Berlin, Germany, August 2016b. Association for Computational Linguistics.

Li, Jiwei, Monroe, Will, Ritter, Alan, Jurafsky, Dan, Galley, Michel, and Gao, Jianfeng. Deep reinforcement learning for dialogue generation. In *EMNLP*, pp. 1192–1202, Austin, Texas, November 2016c. Association for Computational Linguistics.

Li, Jiwei, Miller, Alexander H., Chopra, Sumit, Ranzato, Marc'Aurelio, and Weston, Jason. Dialogue learning with human-in-the-loop. In *ICLR*, 2017.

Miao, Yishu and Blunsom, Phil. Language as a latent variable: Discrete generative models for sentence compression. In *EMNLP*, pp. 319–328, Austin, Texas, November 2016. Association for Computational Linguistics.

Miao, Yishu, Yu, Lei, and Blunsom, Phil. Neural variational inference for text processing. In *ICML*, 2016.

Mnih, Andriy and Gregor, Karol. Neural variational inference and learning in belief networks. In *ICML*, 2014.

Mnih, Volodymyr, Heess, Nicolas, Graves, Alex, and kavukcuoglu, koray. Recurrent models of visual attention. In *NIPS*, 2014.

Papineni, Kishore, Roukos, Salim, Ward, Todd, and Zhu, Wei-Jing. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pp. 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics.

Serban, Iulian V., Sordoni, Alessandro, Lowe, Ryan, Charlin, Laurent, Pineau, Joelle, Courville, Aaron, and Bengio, Yoshua. A hierarchical latent variable encoder-decoder model for generating dialogues. *arXiv preprint: 1605.06069*, 2016.

Serban, Iulian Vlad, Sordoni, Alessandro, Bengio, Yoshua, Courville, Aaron C., and Pineau, Joelle. Hierarchical neural network generative models for movie dialogues. In *AAAI*, 2015.

Shang, Lifeng, Lu, Zhengdong, and Li, Hang. Neural responding machine for short-text conversation. In *ACL*, 2015.

Silver, David, Huang, Aja, Maddison, Chris J, Guez, Arthur, Sifre, Laurent, Van Den Driessche, George, Schrittwieser, Julian, Antonoglou, Ioannis, Panneershelvam, Veda, Lanctot, Marc, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

Snow, Rion, Jurafsky, Daniel, and Ng, Andrew Y. Learning syntactic patterns for automatic hypernym discovery. In *NIPS*, 2004.

Su, Pei-Hao, Vandyke, David, Gasic, Milica, Kim, Dongho, Mrksic, Nikola, Wen, Tsung-Hsien, and Young, Steve. Learning from real users: Rating dialogue success with neural networks for reinforcement learning in spoken dialogue systems. In *Interspeech*, 2015.

Su, Pei-Hao, Gasic, Milica, Mrkšić, Nikola, Rojas Barahona, Lina M., Ultes, Stefan, Vandyke, David, Wen, Tsung-Hsien, and Young, Steve. On-line active reward learning for policy optimisation in spoken dialogue systems. In *ACL*, pp. 2431–2441, Berlin, Germany, August 2016. Association for Computational Linguistics.

Sutskever, Ilya, Vinyals, Oriol, and Le, Quoc V. Sequence to sequence learning with neural networks. In *NIPS*, 2014.

Traum, David R. *Foundations of Rational Agency*, chapter Speech Acts for Dialogue Agents. Springer, 1999.

Vinyals, Oriol and Le, Quoc V. A neural conversational model. In *ICML Deep Learning Workshop*, 2015.

Wen, Tsung-Hsien, Gasic, Milica, Mrkšić, Nikola, Su, Pei-Hao, Vandyke, David, and Young, Steve. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. In *EMNLP*, pp. 1711–1721, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

Wen, Tsung-Hsien, Gasic, Milica, Mrkšić, Nikola, Rojas Barahona, Lina M., Su, Pei-Hao, Ultes, Stefan, Vandyke, David, and Young, Steve. Conditional generation and snapshot learning in neural dialogue systems. In *EMNLP*, pp. 2153–2162, Austin, Texas, November 2016. Association for Computational Linguistics.

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

Weston, Jason E. Dialog-based language learning. In *NIPS*, pp. 829–837. Curran Associates, Inc., 2016.

Williams, Ronald J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3-4):229–256, May 1992. ISSN 0885-6125. doi: 10.1007/BF00992696.

Xu, Kelvin, Ba, Jimmy, Kiros, Ryan, Cho, Kyunghyun, Courville, Aaron, Salakhutdinov, Ruslan, Zemel, Richard, and Bengio, Yoshua. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015.

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