Applied Deep Learning

RL for Dialogues

May 12th, 2020 http://adl.miulab.tw

國立臺灣大學
National Taiwan University

2

RL for Dialogue Systems

- In NLP, RL is mostly used in dialogues regarding its interactive characteristic

Type of Bots	State	Action	Reward
Social Chatbots	Chat history	System Response	# of turns maximized; Intrinsic reward
InfoBots (interactive QA)	User current question + Context	Answers to current question	Relevance of answer; # of turns minimized
Task-Completion Bots	User current input + Context	System dialogue act w/ slot value (or API calls)	Task success rate; # of turns minimized

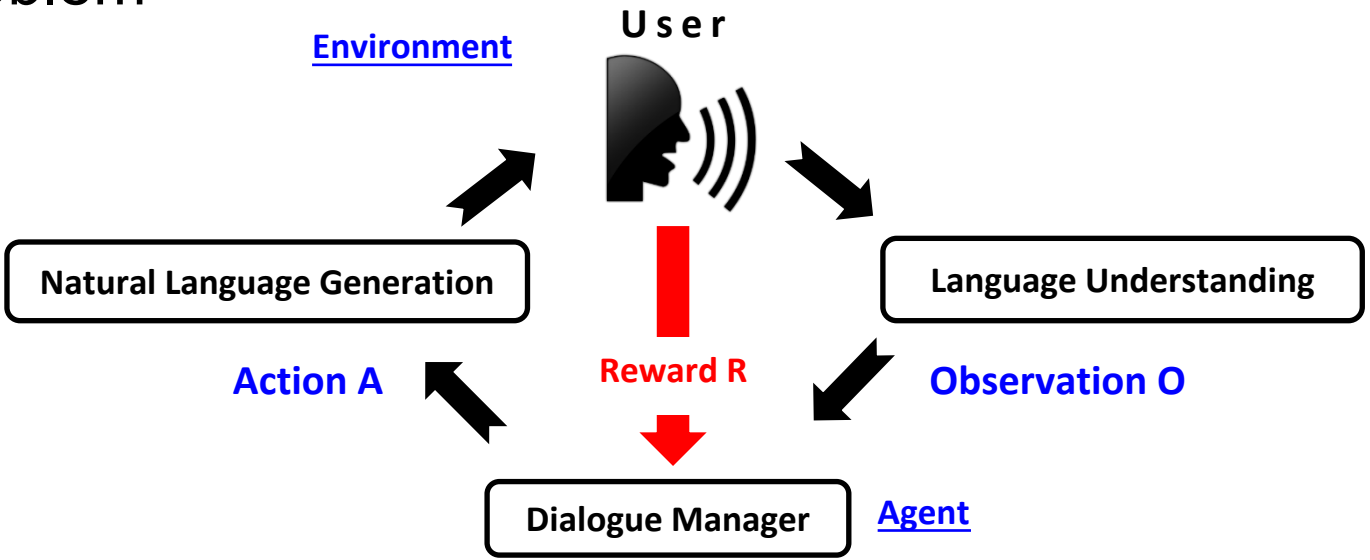
3

Dialogue Policy Optimization

Dialogue manager: the core of determining next action

Dialogue Policy Optimization

- Often formulated as a Reinforcement Learning (RL) problem



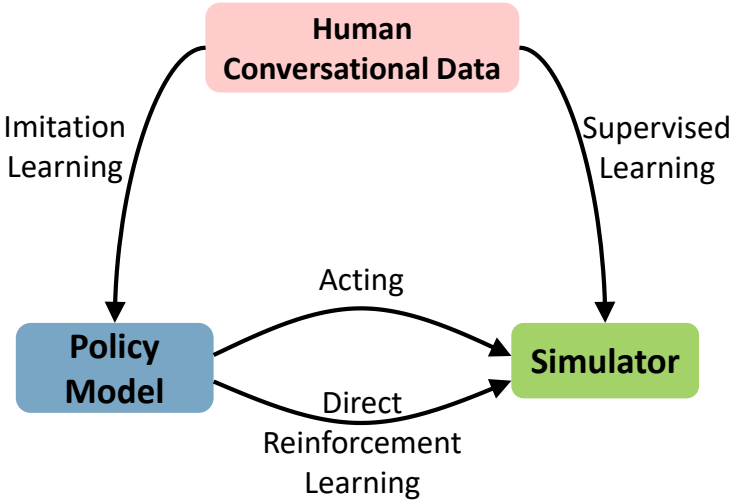
Select the best action that **maximizes the future reward**

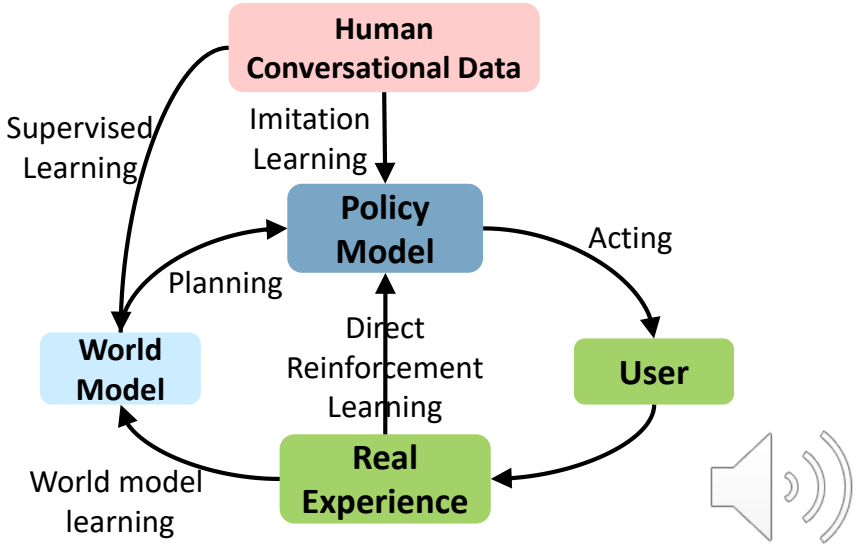
Issues of Dialogue Policy Learning

- Sample inefficient, hard to design reward function, local optima, unstable…
- Real users are expensive, so we usually conduct rule-based user simulators
- Discrepancy between real users and simulators

Deep Dyna-Q

- discrepancy between real users and simulators → learn a user by real data





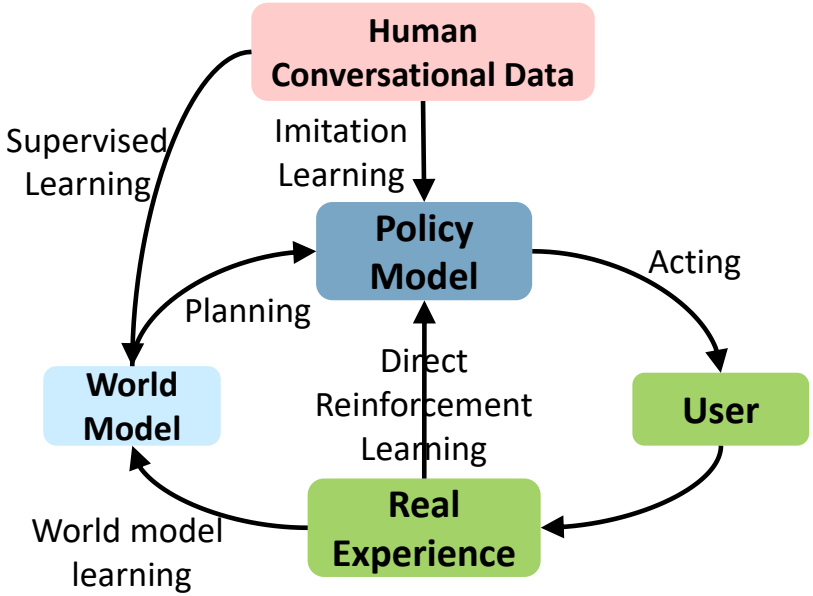
World Model

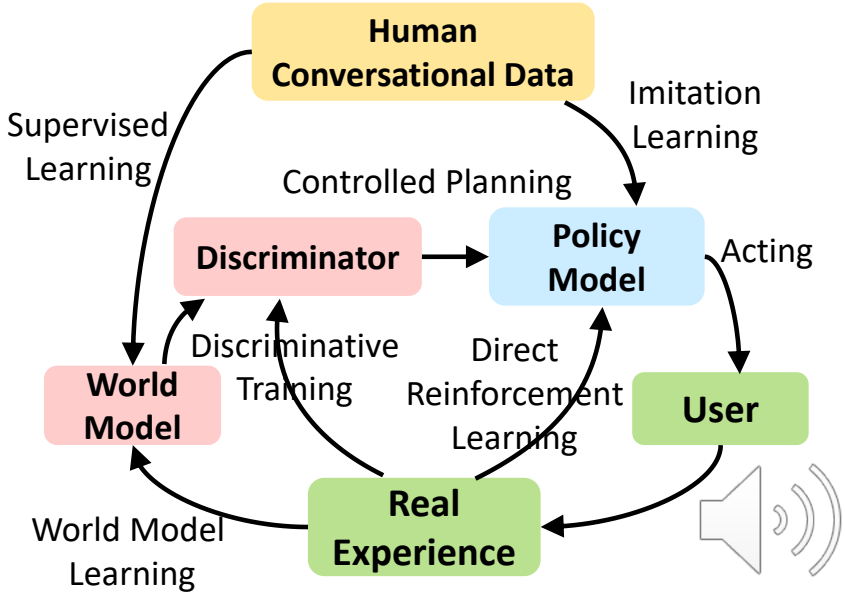
- A trainable model using multi-task DNN to generate simulated experiences for planning
- Input: current dialogue state and the last system action
- Output: user response, reward, a binary signal for dialogue termination
- Regression and classification tasks

$$h = \tanh(W_h(s, a) + b_h) \quad t = \texttt{sigmoid}(W_t h + b_t)$$
$$r = W_r h + b_r \quad o = \texttt{softmax}(W_a h + b_a)$$

Discriminative Deep Dyna-Q

- low-quality fake experience would potentially harm learning
- learn a discriminator (classifier) to judge





Controlled Planning with Discriminator

- Learning effectiveness depends on the quality of simulated experiences used in the planning stage
- A discriminator is to differentiate between real and fake experiences and further pick the “realistic” simulated experiences
- Same objective function of discriminator in GAN:

$$\mathbb{E}_{real}[\log D(x)] + \mathbb{E}_{simu}[\log(1 - D(G(.)))]$$

Iterative Policy Learning

- jointly optimizing the dialog agent and the user simulator by simulating dialogs
- let the agent and the user simulator to positively collaborate to achieve the goal.

$$J(\theta_a, \theta_u) = \mathbb{E}[R] = \sum \pi_{\theta_a}(a_a|s_a)\pi_{\theta_u}(a_u|s_u)R$$

$$\nabla_{\theta_a} J(\theta_a, \theta_u) = \mathbb{E}[\nabla_{\theta_a} \log \pi_{\theta_u}(a_u|s_u)R]$$

Hierarchical Policy Learning

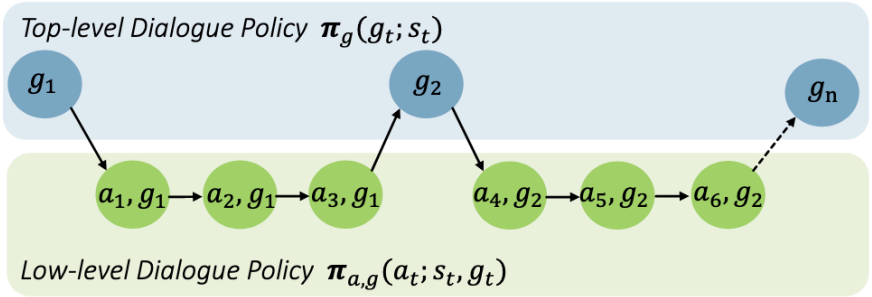
- Consider an important type of complex tasks, termed composite task, which consists of a set of subtasks that need to be fulfilled collectively.
- For example, in order to make a travel plan, we need to book air tickets, reserve a hotel, rent a car, etc.
- The composite task is fulfilled after all its subtasks are completed collectively.

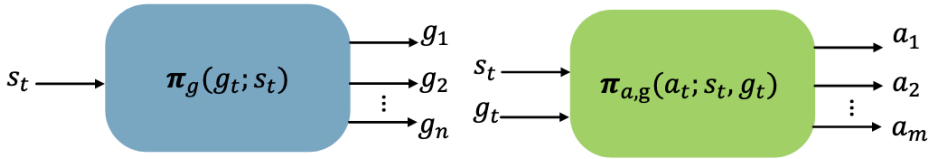
Options over MDPs

- Two-level hierarchical process
- an *option* consists of three components: a set of states where the option can be initiated, an intra-option policy that selects primitive actions while the option is in control, and a termination condition that specifies when the option is completed.

Hierarchical Policy Learning

- The intra-option is a conventional policy over primitive actions, we can consider an inter-option policy over sequences of options in much the same way as we consider the intra-option policy over sequences of actions





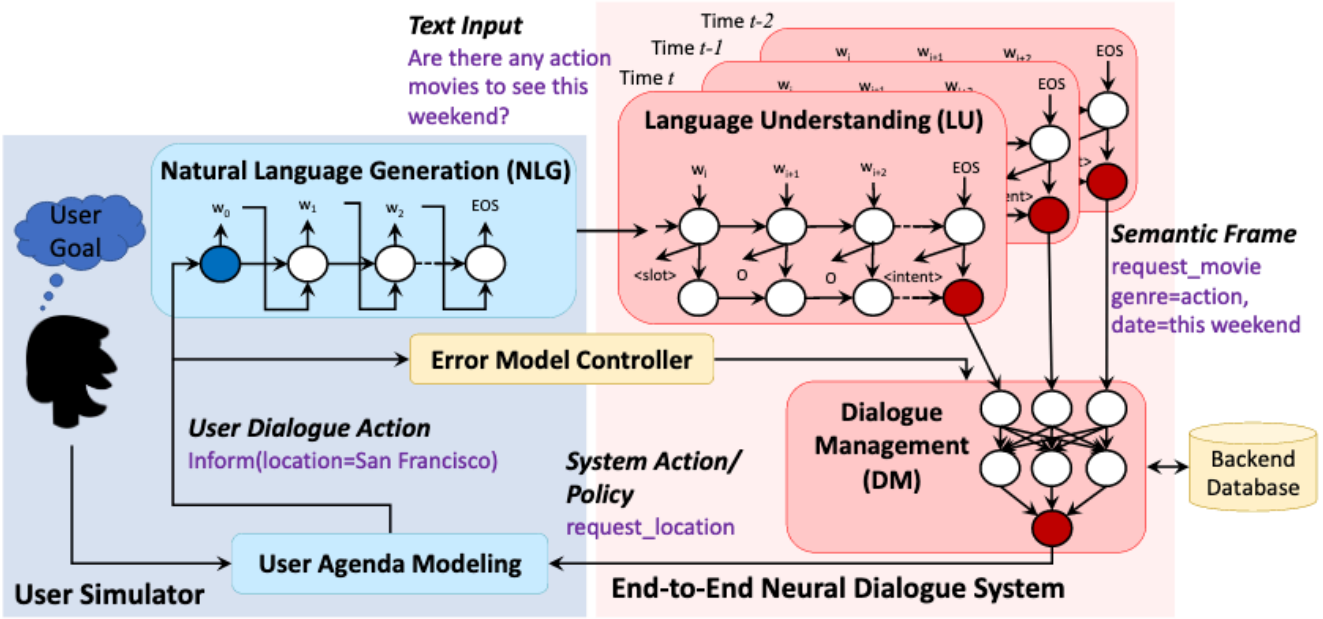
14

End-to-End Neural Dialogue System

Train whole dialogue systems in an end-to-end manner by RL

End-to-End Task-Completion Neural Dialogue Systems

- Use the final reward to train the whole neural dialogue system by RL

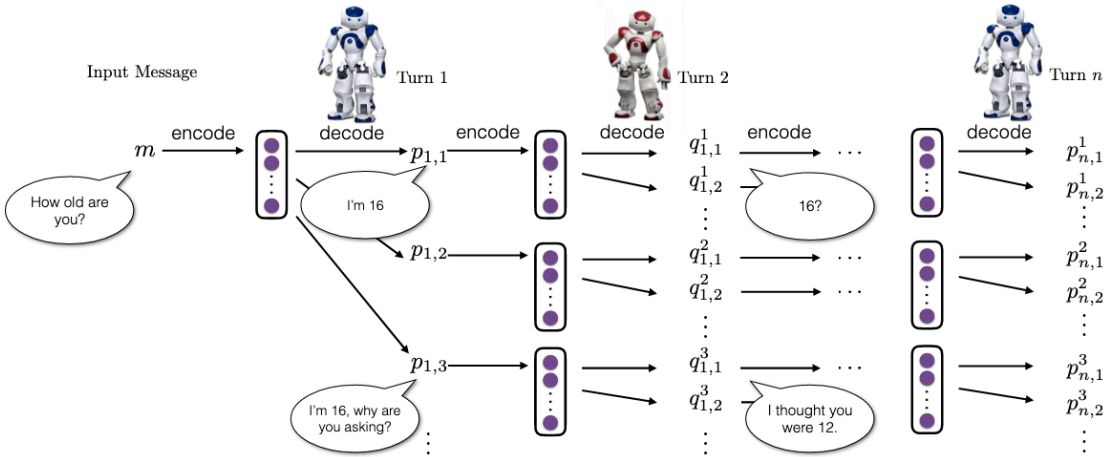


Open-Domain Dialogue Generation

- Unlike RL for task-oriented dialogue, a main challenge that E2E systems are facing is the lack of well-defined metrics for success
- the goal of chatbot is to provide interesting, diverse, and informative feedback that keeps users engaged

Open-Domain Dialogue Generation

- Let the agent model converse with a user model
- The objective is to maximize the expected total reward over the dialogues generated by the user simulator and the agent to be learned



Open-Domain Dialogue Generation

- Formally, the objective is

$$J(\theta) = \mathbb{E}[R(T_1, T_2, ..., T_N)]$$
$$= \sum p(T_1, T_2, ..., T_N) R(T_1, T_2, ..., T_N)$$

- What could be included in the reward function?

Reward

- **Ease of answering**: a turn generated by a machine should be easy to respond to.
- measure the ease of answering a generated turn by using the negative log likelihood of responding to that utterance with a dull response.
- constructed a list of dull responses consisting 8 turns such as “ I have no idea”.

$$r_1 = -\frac{1}{N_{\mathbb{S}}} \sum_{s \in \mathbb{S}} \frac{1}{N_s} \log p_{\text{seq2seq}}(s|a)$$

Reward

- **Information Flow**: we want each agent to contribute new information at each turn to keep the dialogue moving and avoid repetitive sequences.
- Penalizing semantic similarity between consecutive turns from the same agent

$$r_2 = -\log\cos(h_{p_i}, h_{p_{i+1}}) = -\log\cos\frac{h_{p_i} \cdot h_{p_{i+1}}}{\|h_{p_i}\|\|h_{p_{i+1}}\|}$$

Reward

- **Semantic Coherence**: we also need to measure the adequacy of responses to avoid situations in which the generated replies are highly rewarded but are ungrammatical or not coherent.
- Consider the mutual information between the action and previous turns

$$r_3 = \frac{1}{N_a} \log p_{\text{seq2seq}}(a|q_i, p_i) + \frac{1}{N_{q_i}} \log p^{\text{backward}}_{\text{seq2seq}}(q_i|a)$$

Reward

- The final reward for action is a weighted sum of the rewards discussed above:

$$r(a, [p_i, q_i]) = \lambda_1 r_1 + \lambda_2 r_2 + \lambda_3 r_3$$

23

Reward Function Learning

Model-based reward function

Reward Function?

- In practice, designing an appropriate reward function is not always obvious, and substantial domain knowledge is needed.
- Q: How to rate a robot trying backflips?
- **Q: If achieving success in a goal-oriented dialogue, how much reward should we provide? 10? 30?**

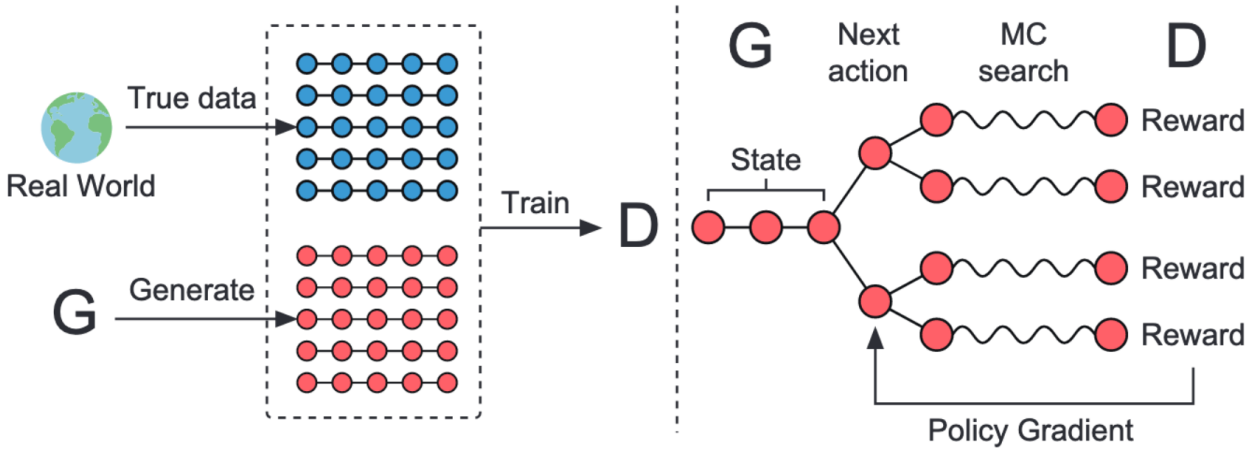
Discriminator as Reward Provider

- A discriminator is to differentiate between real and fake experiences and further pick the “realistic” simulated experiences
- Same objective function of discriminator in GAN:

$$\mathbb{E}_{real}[\log D(x)] + \mathbb{E}_{simu}[\log(1 - D(G(.)))]$$

SeqGAN

- Monte Carlo (MC) search to approximate the state-action value



References

- https://arxiv.org/abs/1703.01008
- https://arxiv.org/abs/1801.06176
- https://arxiv.org/abs/1808.09442
- https://arxiv.org/abs/1709.06136
- https://arxiv.org/abs/1704.03084
- https://arxiv.org/abs/1606.01541
- https://arxiv.org/abs/1805.11762
- https://arxiv.org/abs/1809.08267
- https://arxiv.org/abs/1609.05473