Affordable On-line Dialogue Policy Learning
— Hybrid-Intelligent Approaches

SJTU SpeechLab
Annual Academic Meeting
05/11/2018

Our Team

Prof. Kai Yu（俞凯教授）

Lu Chen (陈露)

Ph.D. Candidate

Cheng Chang(常成)

Master

Zihao Ye（叶子豪）

Undergrad

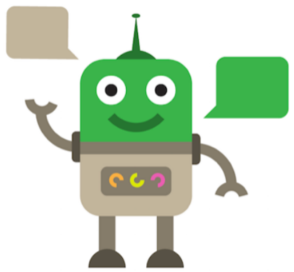
Xiang Zhou（周翔）

Undergrad

Runzhe Yang (杨闰哲)

Undergrad

Overview



Affordable Online Dialogue Policy Learning

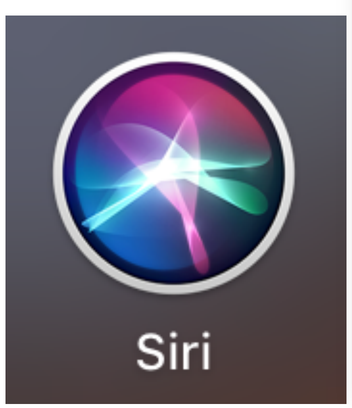
Hybrid-Intelligent Task-Oriented SDSs

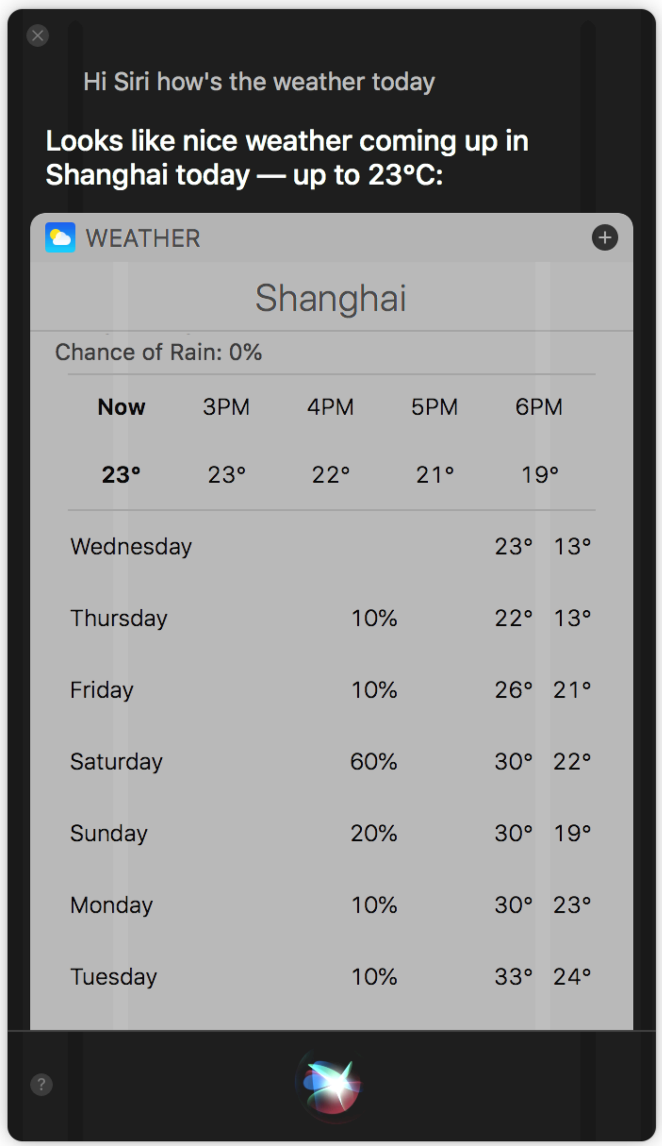
2 papers at EMNLP 2017 and 1 short paper at EACL 2017

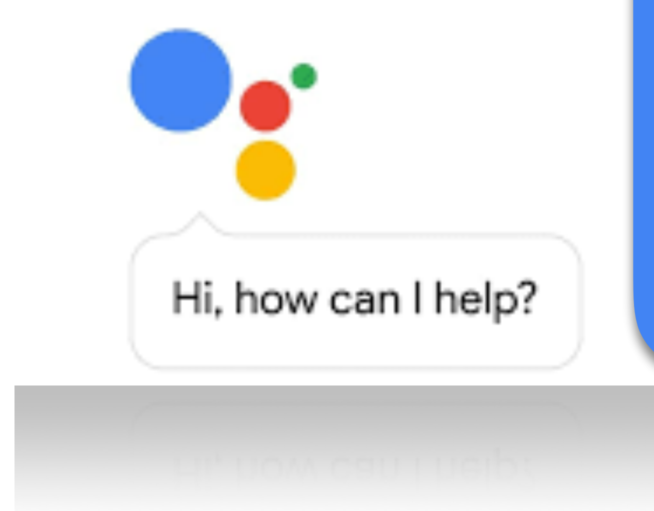
- What's a Task-Oriented Spoken Dialogue System (SDS)?
  - 1. Task-Oriented SDSs
  - 2. Dialogue Policies
  - 3. Reinforcement Learning
- The Cold Start Problem
  - 1. A Human-in-the-Loop Solution
  - 2. A Complete Companion Teaching Framework
  - 3. Replacing Human Teachers with Rule-Based Systems
- Summary

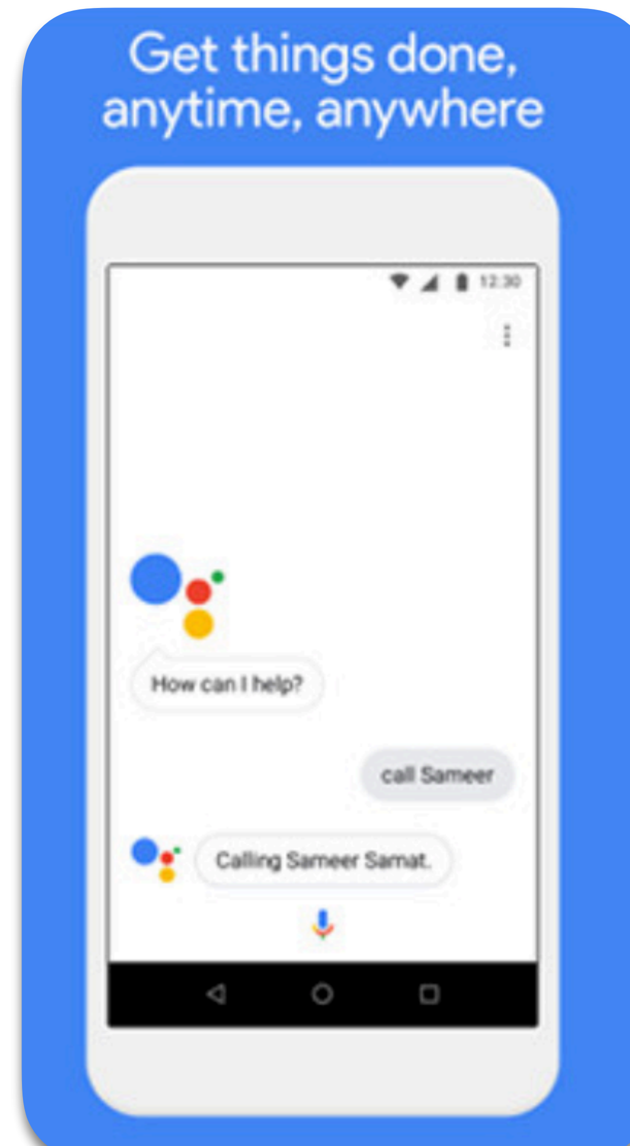
Introduction

What's a Task-Oriented Spoken Dialogue System?



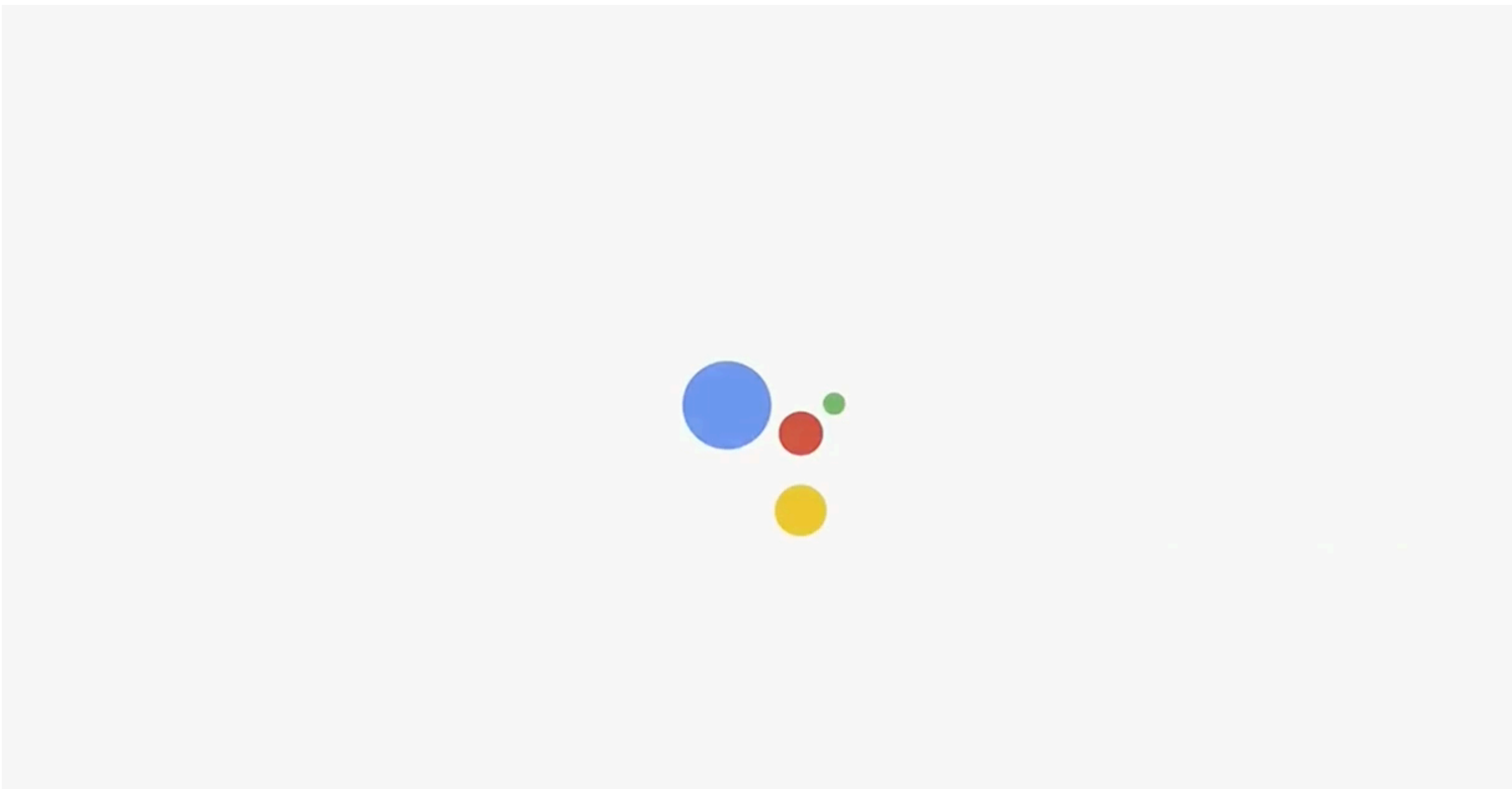






Introduction

What's a Task-Oriented Spoken Dialogue System?



Introduction

What's a Task-Oriented Spoken Dialogue System?

Task-Oriented SDS is a killer app for AI.

Introduction

What's a Task-Oriented Spoken Dialogue System?

Task-Oriented SDS is a killer app for AI.

- Required to **satisfy user goals**
  - e.g., restaurant reservation, weather information query

Introduction

What's a Task-Oriented Spoken Dialogue System?

Task-Oriented SDS is a killer app for AI.

- Required to **satisfy user goals**
    - e.g., restaurant reservation, weather information query
- Required to make **multi-round interaction**
    - to maintain the context and the user intention

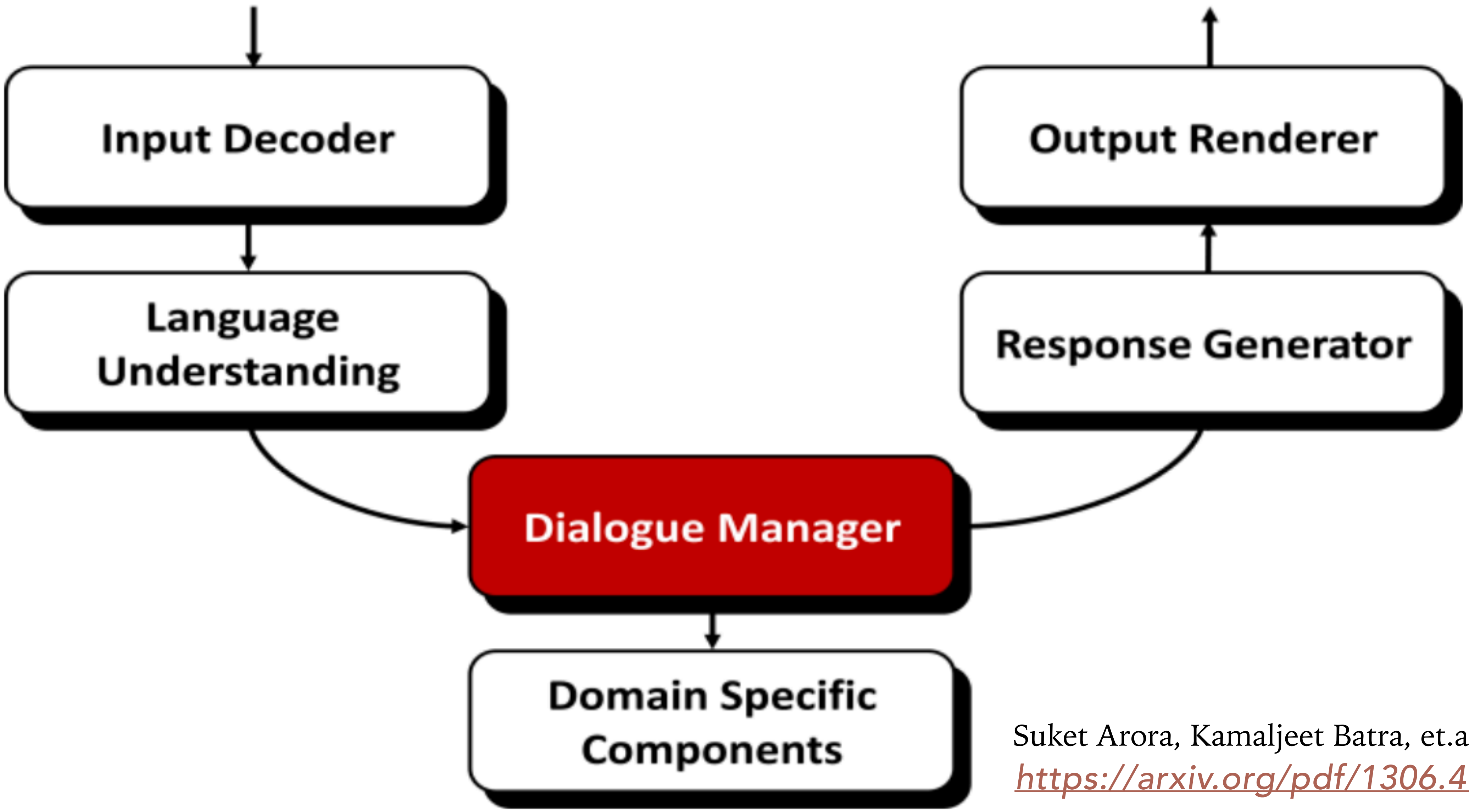
Introduction

What's a Task-Oriented Spoken Dialogue System?

Task-Oriented SDS is a killer app for AI.

- Required to **satisfy user goals**
  - e.g., restaurant reservation, weather information query
- Required to make **multi-round interaction**
  - to maintain the context and the user intention
- Required to deal with **uncertainty**
  - errors from both recognition and understanding

Task-Oriented Spoken Dialogue Systems

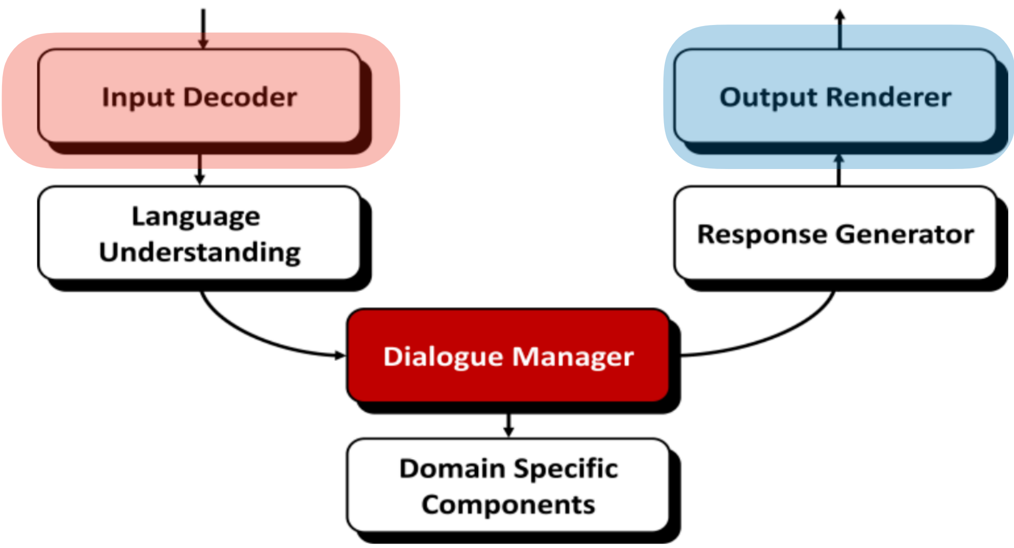


Suket Arora, Kamaljeet Batra, et.al., 2013
https://arxiv.org/pdf/1306.4134.pdf

Task-Oriented Spoken Dialogue Systems

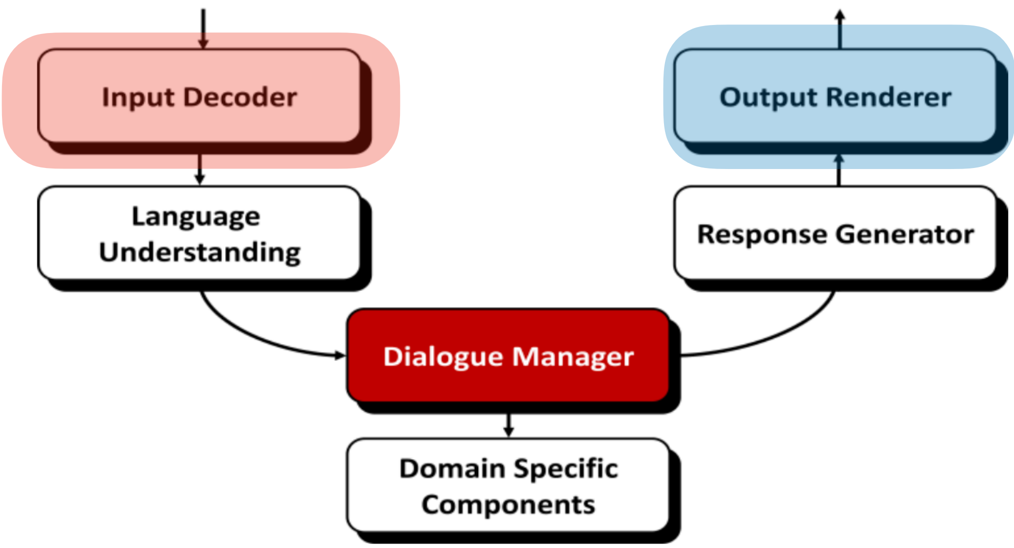
System:	East Pittsburg Bus Schedules. Say a bus route, like 28X, or say I'm not sure.	
	hello(), request(route), example(route=28x), example(route=dont_know)	
User:	61A	
SLU:	0.77	inform(route=61a)
	0.12	inform(route=61)
	0.01	inform(route=61d)
System:	Okay, 61A. To change, say go back. Where are you leaving from?	
	impl-conf(route=61a), example(act=goback), request(from)	
User:	Downtown	
SLU:	0.59	inform(from.desc=downtown)
	0.10	inform(from.desc=from downtown)
System:	Okay, downtown. You can always say go back. And where are you going to?	
	impl-conf(from.desc=downtown), example(act=goback), request(to)	
User:	East Pittsburgh East Pittsburgh	
SLU:	0.25	inform(to.desc=pittsburgh)
	0.20	inform(to.desc=east pittsburgh)

Task-Oriented Spoken Dialogue Systems



User:	Downtown	
SLU:	0.59	inform(from.desc=downtown)
	0.10	inform(from.desc=from downtown)
System:	Okay, downtown. You can always say go back. And where are you going to?	
	impl-conf(from.desc=downtown), example(act=goback), request(to)	
User:	East Pittsburgh East Pittsburgh	
SLU:	0.25	inform(to.desc=pittsburgh)
	0.20	inform(to.desc=east pittsburgh)

Task-Oriented Spoken Dialogue Systems

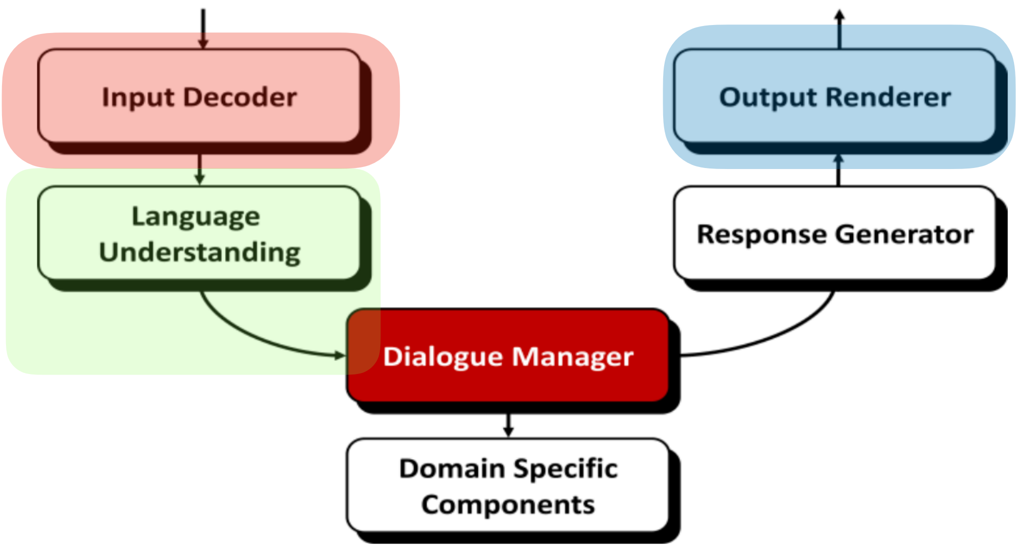


Intent-level interaction

Dialogue Act: acttype-slot-value, e.g. inform(route=61a)

User: Downtown
SLU: 0.59 inform(from.desc=downtown)
0.10 inform(from.desc=from downtown)
System: Okay, downtown. You can always say go back. And where are you going to?
impl-conf(from.desc=downtown), example(act=goback), request(to)
User: East Pittsburgh East Pittsburgh
SLU: 0.25 inform(to.desc=pittsburgh)
0.20 inform(to.desc=east pittsburgh)

Task-Oriented Spoken Dialogue Systems



Intent-level interaction

Dialogue Act: acttype-slot-value, e.g. inform(route=61a)

User: Downtown

SLU: 0.59 inform(from.desc=downtown)

0.10 inform(from.desc=from downtown)

Dialogue Acts (probability distribution)

System: Okay, downtown. You can always say go back. And where are you going to?

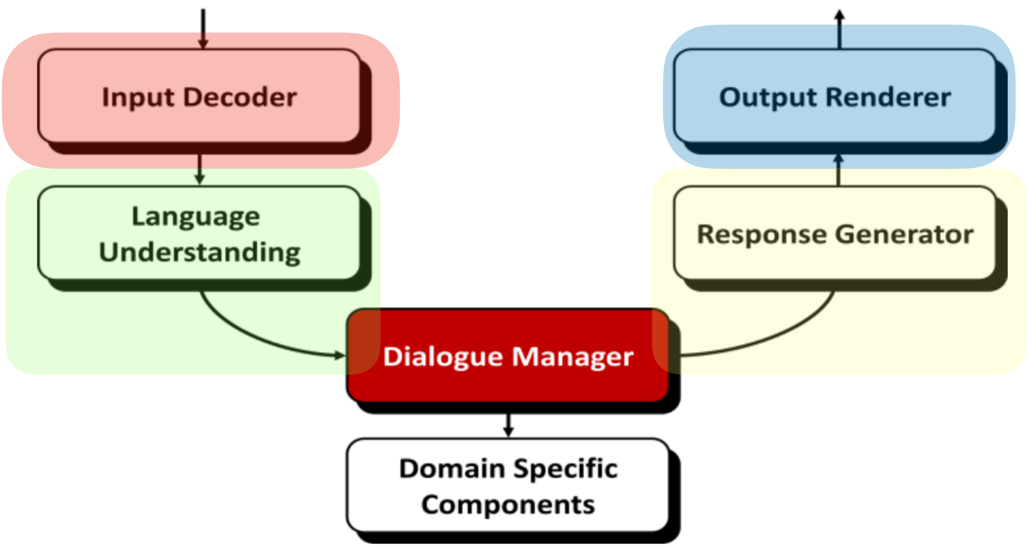
impl-conf(from.desc=downtown), example(act=goback), request(to)

User: East Pittsburgh East Pittsburgh

SLU: 0.25 inform(to.desc=pittsburgh)

0.20 inform(to.desc=east pittsburgh)

Task-Oriented Spoken Dialogue Systems



Intent-level interaction

Dialogue Act: acttype-slot-value, e.g. inform(route=61a)

User: Downtown

SLU: 0.59 inform(from.desc=downtown)

0.10 inform(from.desc=from downtown)

Dialogue Acts (probability distribution)

System: Okay, downtown. You can always say go back. And where are you going to?

impl-conf(from.desc=downtown), example(act=goback), request(to)

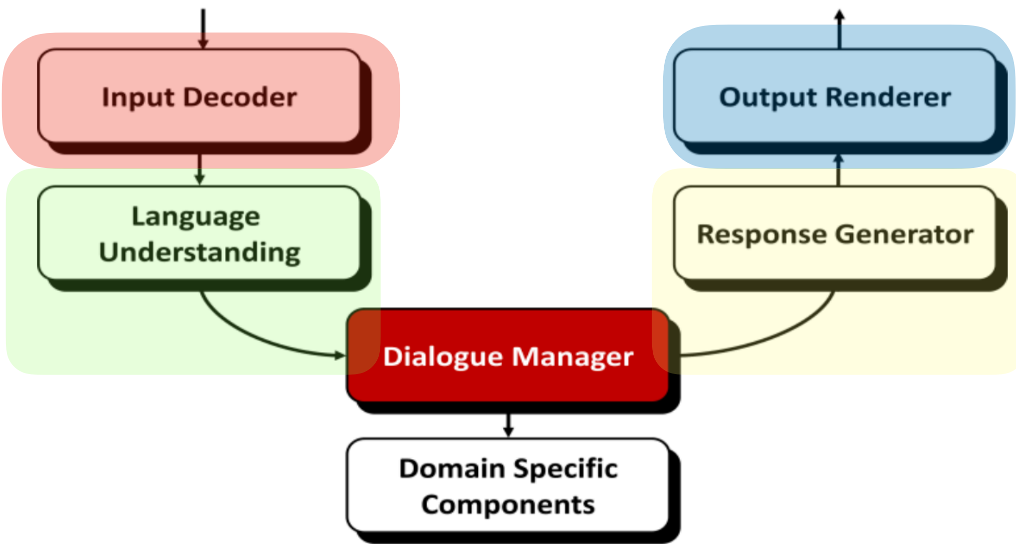
User: East Pittsburgh East Pittsburgh

Dialogue Acts

SLU: 0.25 inform(to.desc=pittsburgh)

0.20 inform(to.desc=east pittsburgh)

Task-Oriented Spoken Dialogue Systems



Intent-level interaction

Dialogue Act: acttype-slot-value, e.g. inform(route=61a)

$$\texttt{Dialogue\_Manager} : \Delta(\texttt{ACT}_{user}) \rightarrow \texttt{ACT}_{sys}$$

User: Downtown

SLU: 0.59 inform(from.desc=downtown)
0.10 inform(from.desc=from downtown)

Dialogue Acts (probability distribution)

System: Okay, downtown. You can always say go back. And where are you going to?
impl-conf(from.desc=downtown), example(act=goback), request(to)

Dialogue Acts

User: East Pittsburgh East Pittsburgh

SLU: 0.25 inform(to.desc=pittsburgh)
0.20 inform(to.desc=east pittsburgh)

Dialogue Manager

The “brain” of SDS?

$$\texttt{Dialogue\_Manager} : \Delta(\texttt{ACT}_{user}) \rightarrow \texttt{ACT}_{sys}$$

- Required to **satisfy user goals**
- Required to make **multi-round interaction**
- Required to deal with **uncertainty**

Dialogue Manager

The "brain" of SDS?

✖ $\texttt{Dialogue\_Manager} : \Delta(\texttt{ACT}_{user}) \rightarrow \texttt{ACT}_{sys}$

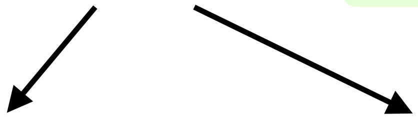
dialogue acts do not encode the user goal & context

- Required to **satisfy user goals**
- Required to make **multi-round interaction**
- Required to deal with **uncertainty**

Dialogue Manager

The "brain" of SDS?

✖ $\texttt{Dialogue\_Manager} : \Delta(\texttt{ACT}_{user}) \rightarrow \texttt{ACT}_{sys}$



$\texttt{Dialgue\_State\_Tracker}$ $\texttt{Dialgue\_Policy\_Maker}$

Dialogue Manager

The "brain" of SDS?

✖ $\texttt{Dialogue\_Manager} : \Delta(\texttt{ACT}_{user}) \rightarrow \texttt{ACT}_{sys}$

$\texttt{Dialgue\_State\_Tracker}$ $\texttt{Dialgue\_Policy\_Maker}$

Dialogue State
(Probability Distribution)

= **Goal** x **Current Semantics** x **History**

e.g. (from.desc = ?, to.desc = ?, route =?)

Dialogue Manager

User Dialogue Acts
(probability distribution)

The "brain" of SDS?

✖ $\texttt{Dialogue\_Manager} : \Delta(\texttt{ACT}_{user}) \rightarrow \texttt{ACT}_{sys}$

$\texttt{Dialgue\_State\_Tracker}$ $\texttt{Dialgue\_Policy\_Maker}$

Dialogue State
(Probability Distribution)

= Goal x Current Semantics x History

e.g. (from.desc = ?, to.desc = ?, route =?)

Dialogue Act

= acttype-slot-value

Dialogue Manager - State Tracker

The "brain" of SDS?

User Dialogue Acts
(probability distribution)

✖ Dialogue_Manager : $\Delta(\mathtt{ACT}_{user}) \rightarrow \mathtt{ACT}_{sys}$

Dialgue_State_Tracker

Dialogue State
(Probability Distribution)

= Goal x Current Semantics x History

e.g. (from.desc = ?, to.desc = ?, route =?)

Dialgue_Policy_Maker

Dialogue Act

= acttype-slot-value

Dialogue Manager - Policy Maker

The "brain" of SDS?

User Dialogue Acts
(probability distribution)

✖ $\texttt{Dialogue\_Manager}: \Delta(\texttt{ACT}_{user}) \rightarrow \texttt{ACT}_{sys}$

$\texttt{Dialgue\_State\_Tracker}$

Dialogue State
(Probability Distribution)

= Goal x Current Semantics x History

e.g. (from.desc = ?, to.desc = ?, route =?)

$\texttt{Dialgue\_Policy\_Maker}$

Dialogue Act

= acttype-slot-value

Dialogue Manager - Policy Maker

The "brain" of SDS?

User Dialogue Acts
(probability distribution)

✖ $\texttt{Dialogue\_Manager} : \Delta(\texttt{ACT}_{user}) \rightarrow \texttt{ACT}_{sys}$

$\texttt{Dialgue\_State\_Tracker}$

Dialogue State
(Probability Distribution)

= Goal x Current Semantics x History

e.g. (from.desc = ?, to.desc = ?, route =?)

$\texttt{Dialgue\_Policy\_Maker}$

Dialogue Act

= acttype-slot-value

How?

$\Delta(\texttt{STATE}) \rightarrow \texttt{ACT}_{sys}$

- to **satisfy user goals**

Dialogue Manager - Policy Maker

How do we build the “brain”? (esp. to find good policy?)

Dialgue_Policy_Maker $\Delta(\texttt{STATE}) \rightarrow \texttt{ACT}_{sys}$

Dialogue Manager - Policy Maker

How do we build the “brain”? (esp. to find good policy?)

$$\texttt{Dialgue\_Policy\_Maker} \qquad \Delta(\texttt{STATE}) \rightarrow \texttt{ACT}_{sys}$$

Rule-Based Methods

- hand-craft rules, “safe” but not “flexible”.

Dialogue Manager - Policy Maker

How do we build the "brain"? (esp. to find good policy?)

$$\texttt{Dialgue\_Policy\_Maker} \quad \Delta(\texttt{STATE}) \rightarrow \texttt{ACT}_{sys}$$

Rule-Based Methods

- hand-craft rules, "safe" but not "flexible".

Data-Driven Methods

- learn from interactions, dialogue manager is evolvable.

Dialogue Manager - Policy Maker

How do we build the "brain"? (esp. to find good policy?)

$$\texttt{Dialgue\_Policy\_Maker} \quad \Delta(\texttt{STATE}) \rightarrow \texttt{ACT}_{sys}$$

Rule-Based Methods

- hand-craft rules, "safe" but not "flexible".

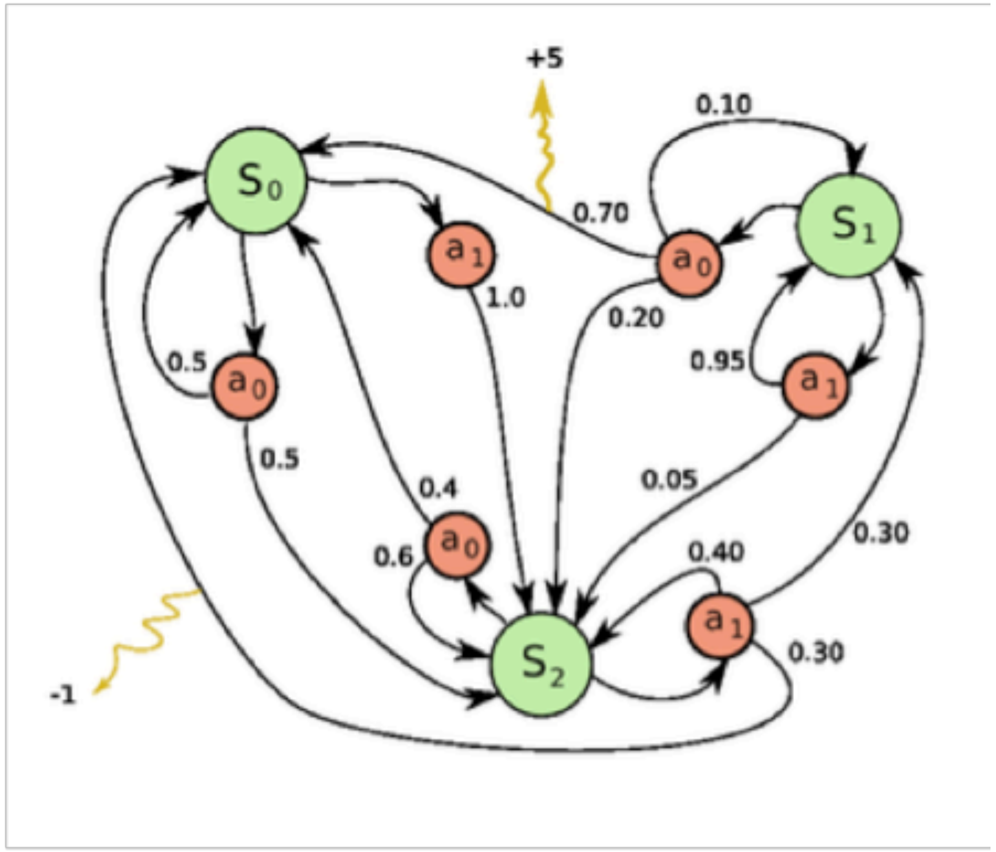
Data-Driven Methods

- learn from interactions, dialogue manager is evolvable.
- convert to **sequential decision make problems.**

Markov Decision Processes (MDPs)

Data-Driven Methods

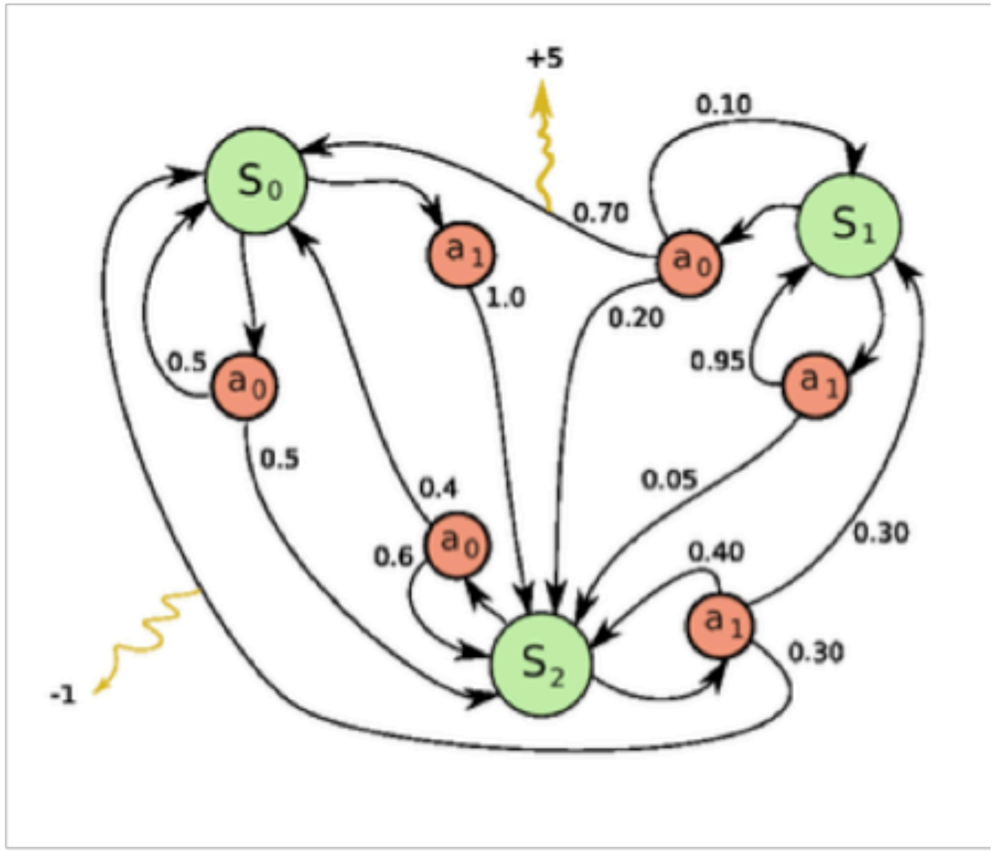
- convert to **sequential decision make problems.**



Markov Decision Processes (MDPs)

Data-Driven Methods

- convert to **sequential decision make problems.**



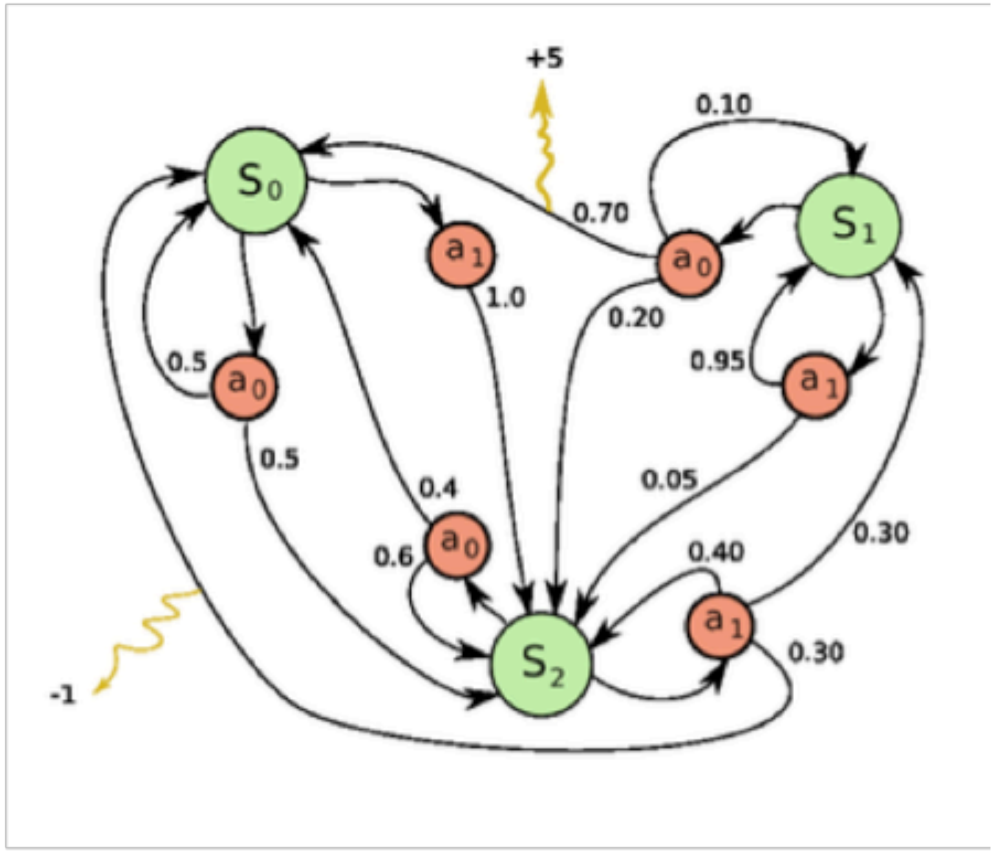
$$\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma \rangle$$

State Space

Markov Decision Processes (MDPs)

Data-Driven Methods

- convert to **sequential decision make problems.**



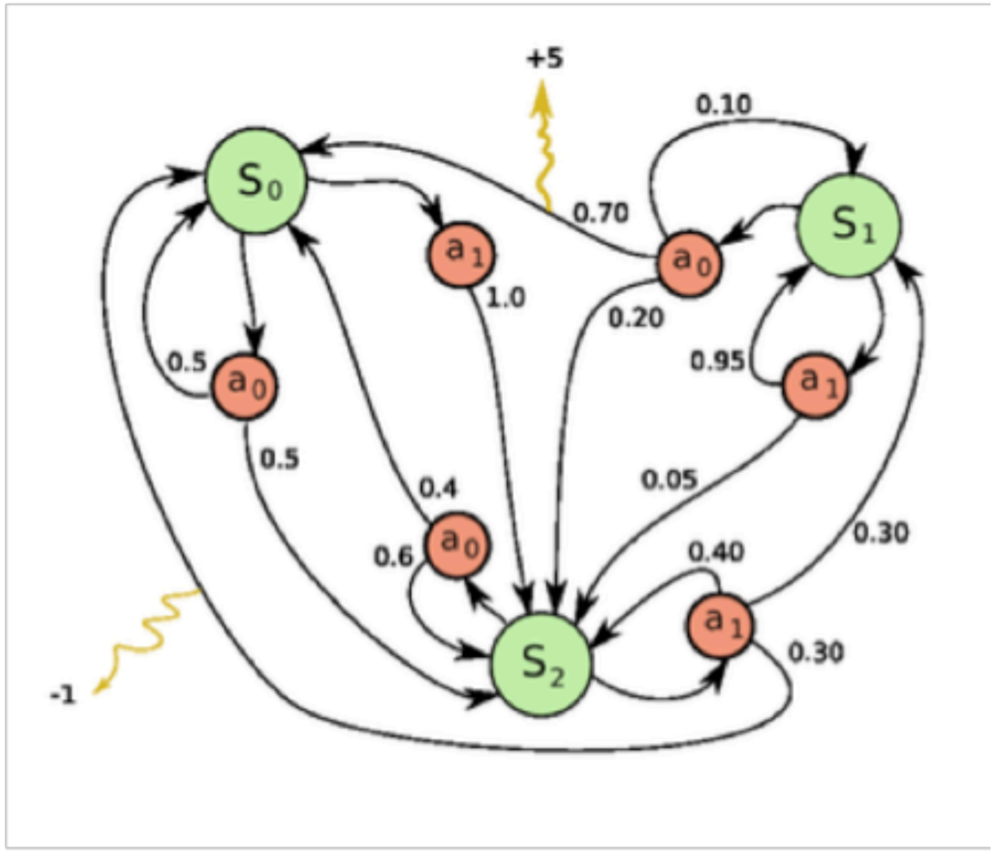
$$\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma \rangle$$

State Space *Action Space*

Markov Decision Processes (MDPs)

Data-Driven Methods

- convert to **sequential decision make problems.**



$$\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma \rangle$$

State Space *Action Space*

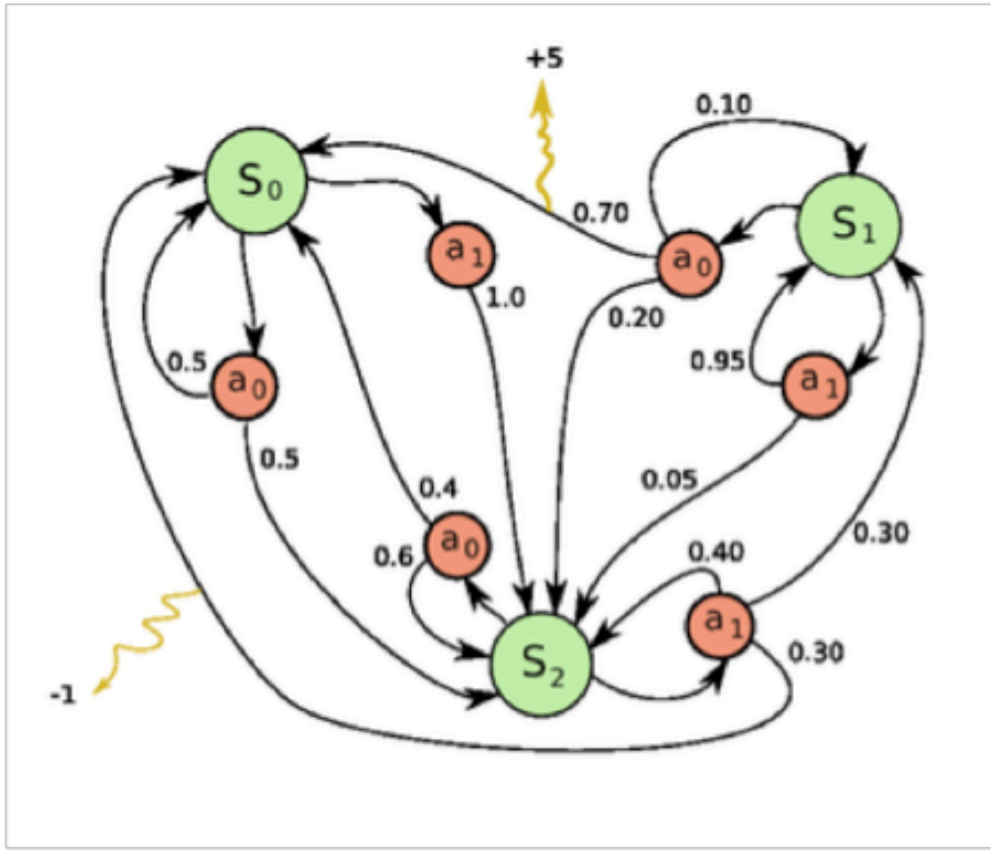
Stochastic Transition Kernel $\mathcal{P}(s'|s, a)$

e.g. $\mathcal{P}(S_0|S_1, a_0) = 0.7$

Markov Decision Processes (MDPs)

Data-Driven Methods

- convert to **sequential decision make problems.**



$\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma \rangle$

State Space *Action Space*

Stochastic Transition Kernel $\mathcal{P}(s'|s, a)$

e.g. $\mathcal{P}(S_0|S_1, a_0) = 0.7$

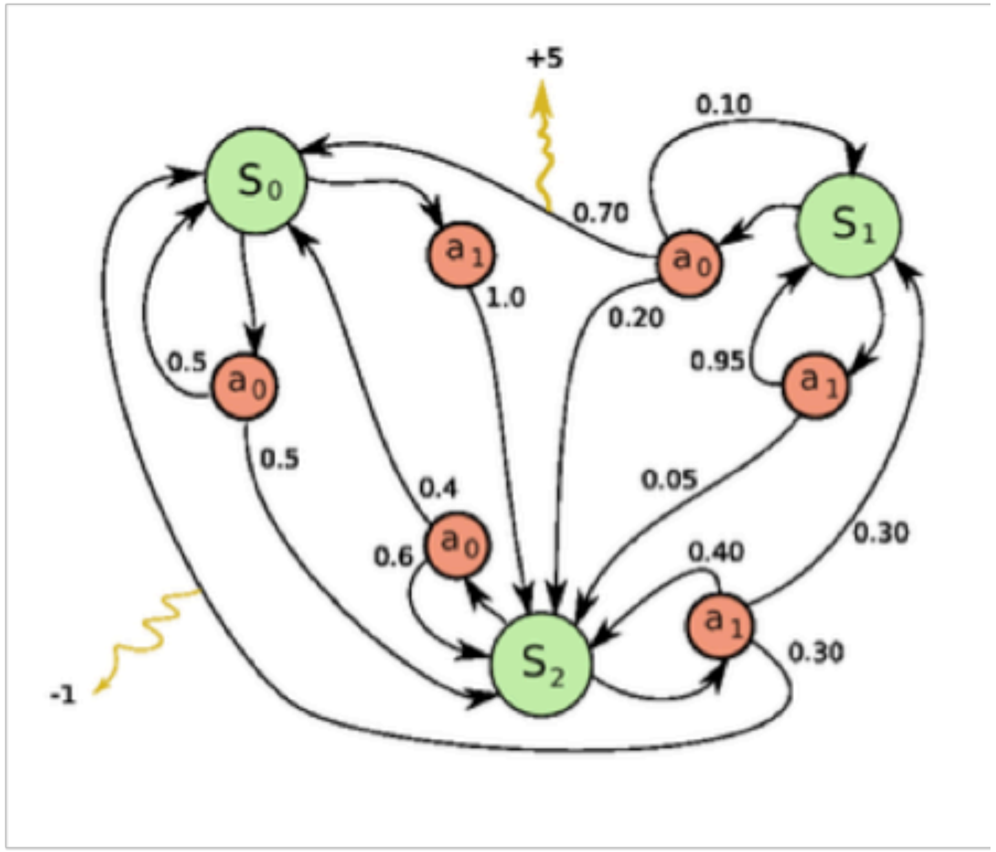
Reward Function $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$

e.g. $r(S_1, a_0) = 3.5$

Markov Decision Processes (MDPs)

Data-Driven Methods

- convert to **sequential decision make problems.**



$$\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma \rangle$$

State Space *Action Space*

Stochastic Transition Kernel $\mathcal{P}(s'|s,a)$

e.g. $\mathcal{P}(S_0|S_1,a_0) = 0.7$

Reward Function $r : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$

e.g. $r(S_1, a_0) = 3.5$

$\gamma \in [0,1)$ is a *discount factor*

Markov Decision Processes (MDPs)

Data-Driven Methods

- convert to **sequential decision make problems.**

State Space

$\Delta(\texttt{STATE})$ **Dialogue State (Probability Distribution)**

Action Space

$\texttt{ACT}_{sys}$ **Dialogue Act**

acttype-slot-value
e.g. inform(route=61a)

Reward Function:

$$r_t = r_t^{turn} + r_t^{\texttt{succ}}$$

$$\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma \rangle$$

State Space *Action Space*

Stochastic Transition Kernel $\mathcal{P}(s'|s, a)$

e.g. $\mathcal{P}(S_0|S_1, a_0) = 0.7$

Reward Function $r : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$

e.g. $r(S_1, a_0) = 3.5$

$\gamma \in [0, 1)$ is a *discount factor*

Markov Decision Processes (MDPs)

Goal: find optimal policy π such that

$$v^{\pi}(s) := \mathbb{E}_{\tau \sim (P,\pi) | s_0 = s} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right] \quad \text{is maximized.}$$

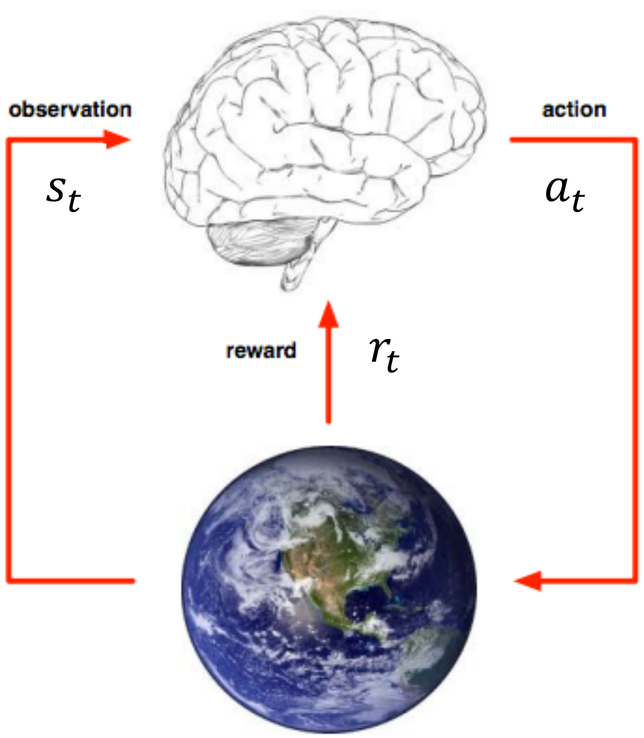
Solve by Value-Based Reinforcement Learning

Markov Decision Processes (MDPs)

Goal: find optimal policy π such that

$$v^{\pi}(s) := \mathbb{E}_{\tau \sim (P,\pi)|s_0=s}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right] \quad \textbf{is maximized.}$$

Solve by Value-Based Reinforcement Learning



- $Q(s_t, a_t)$ represents the expected total reward after take the action a_t at the state s_t

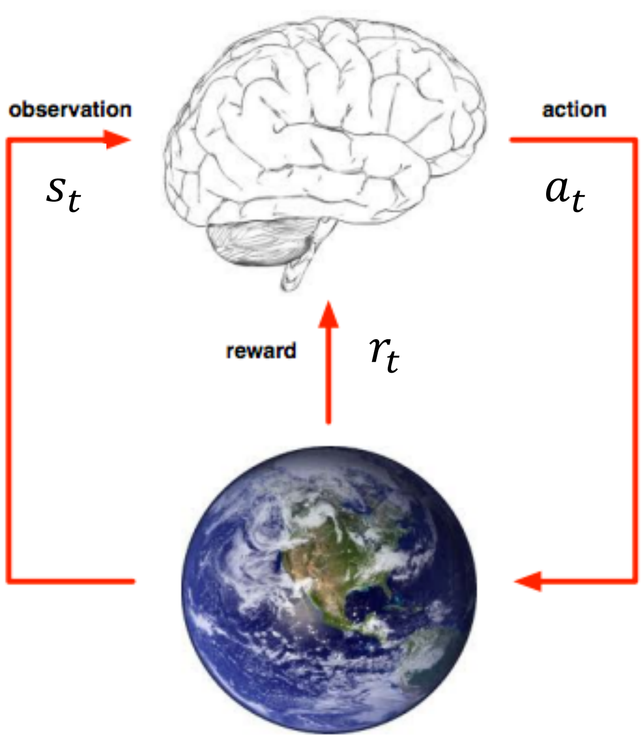
$$Q(s_t, a_t) = r_t + \gamma \max_{a'} Q(s_{t+1}, a')$$

Markov Decision Processes (MDPs)

Goal: find optimal policy π such that

$$v^{\pi}(s) := \mathbb{E}_{\tau \sim (P,\pi)|s_0=s}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right]$$ is maximized.

Solve by Value-Based Reinforcement Learning



- $Q(s_t, a_t)$ represents the expected total reward after take the action a_t at the state s_t

$$Q(s_t, a_t) = r_t + \gamma \max_{a'} Q(s_{t+1}, a')$$

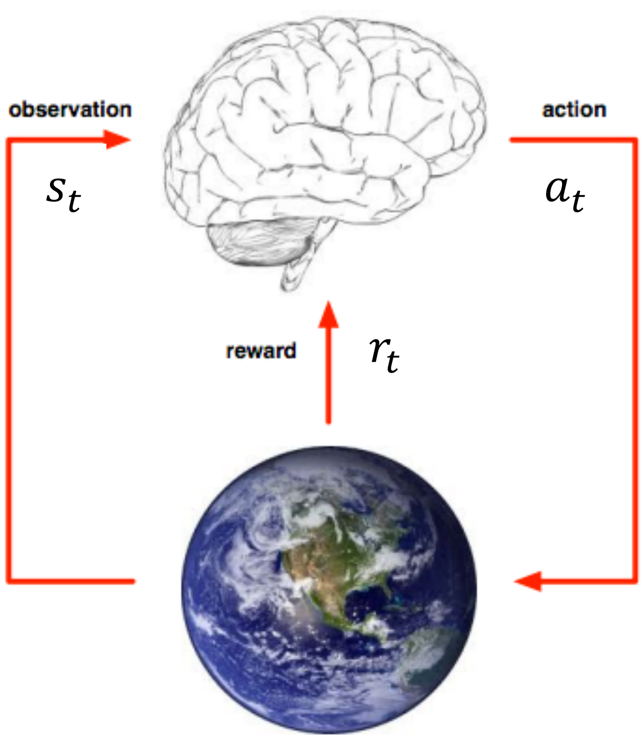
- Decision: $a_t = \max_{a_t} Q(s_t, a_t)$

Markov Decision Processes (MDPs)

Goal: find optimal policy π such that

$$v^{\pi}(s) := \mathbb{E}_{\tau \sim (P,\pi)|s_0=s}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right]$$ is maximized.

Solve by Value-Based Reinforcement Learning



- $Q(s_t, a_t)$ represents the expected total reward after take the action a_t at the state s_t

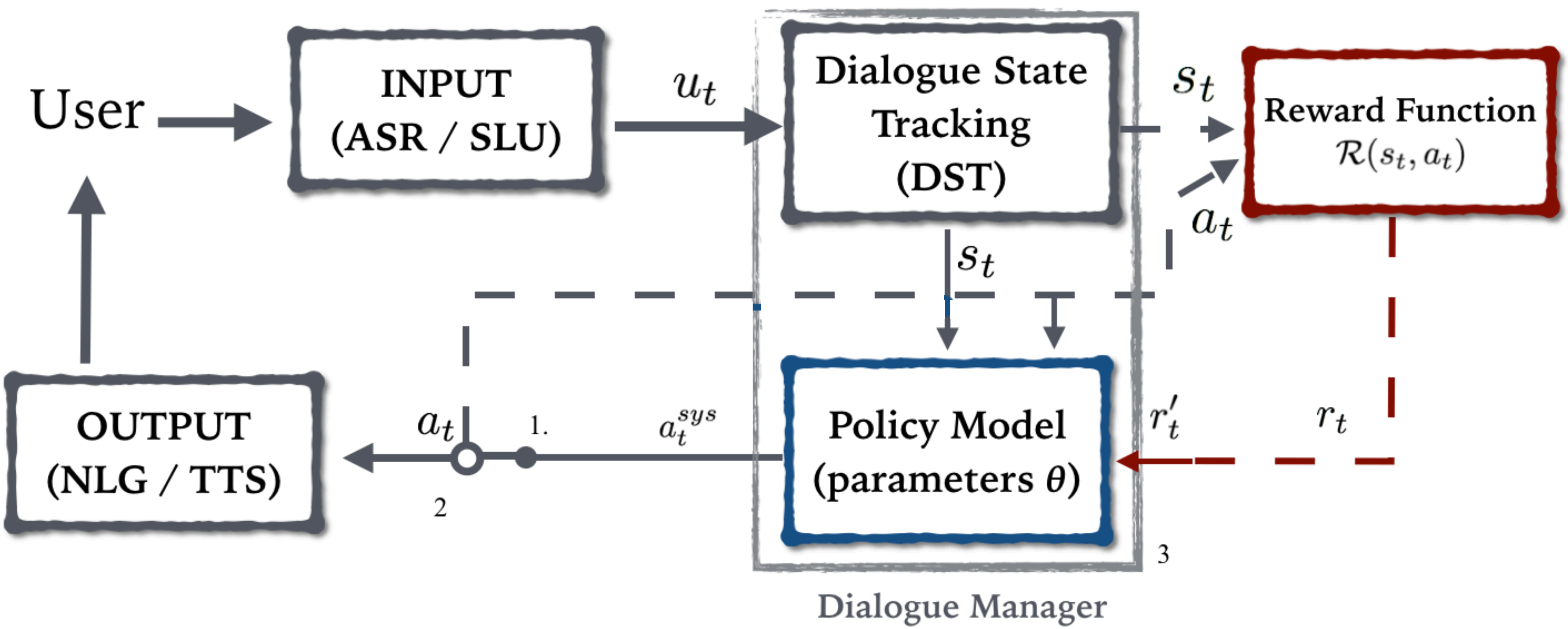
$$Q(s_t, a_t) = r_t + \gamma \max_{a'} Q(s_{t+1}, a')$$

- Decision: $a_t = \max_{a_t} Q(s_t, a_t)$

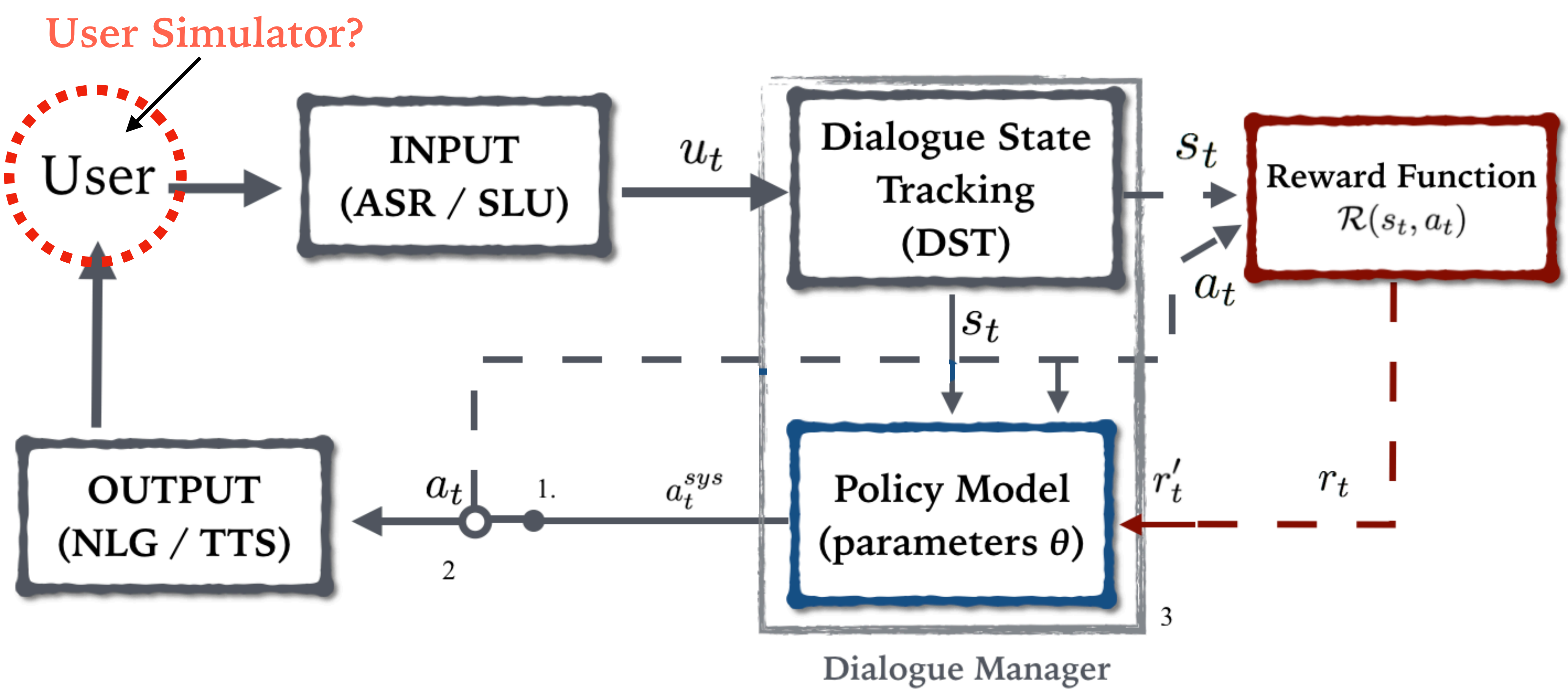
- Training: $Q(s_t, a_t, \theta)$ is approximated by NN

$$l(\theta) = \mathbb{E}_{s,a\sim\pi_\theta}[(Q_{target} - Q(s_t, a_t, \theta))^2]$$

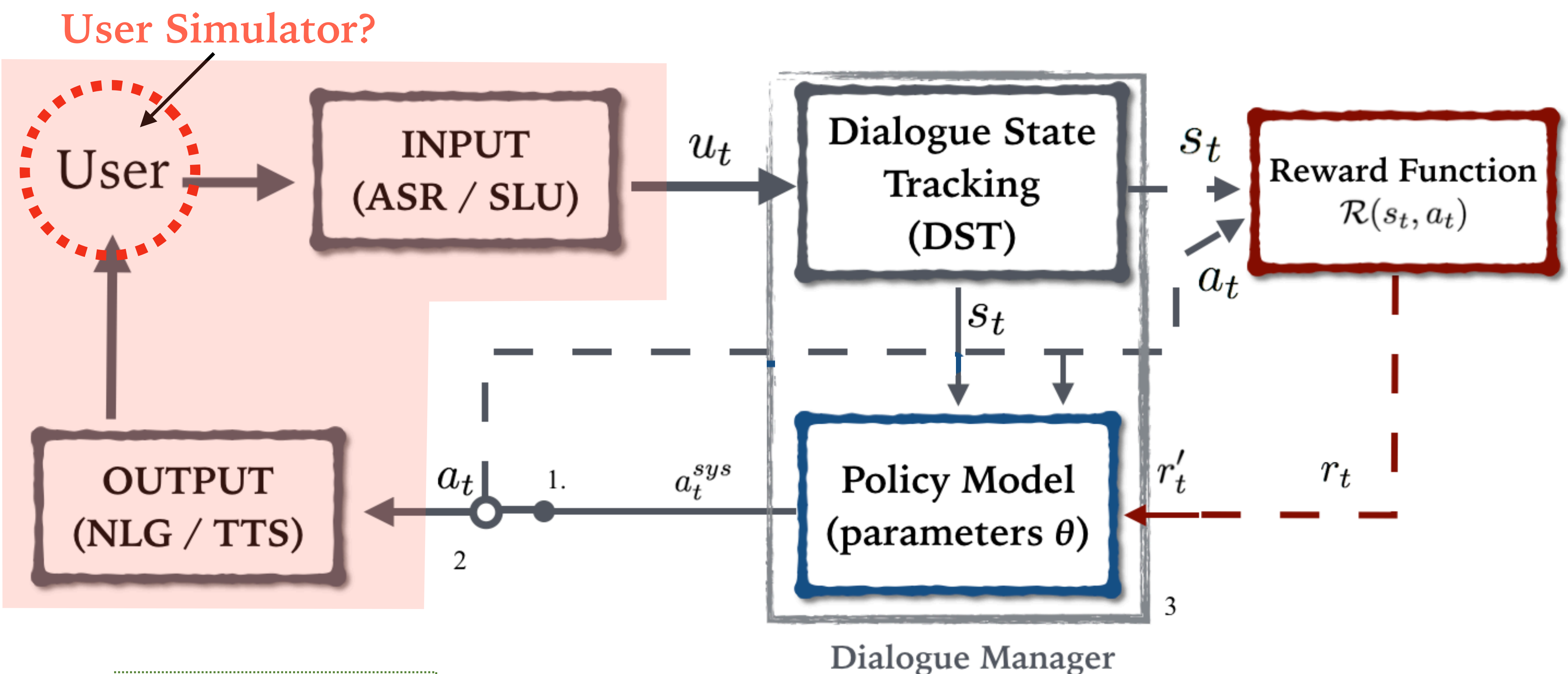
RL-Based Framework

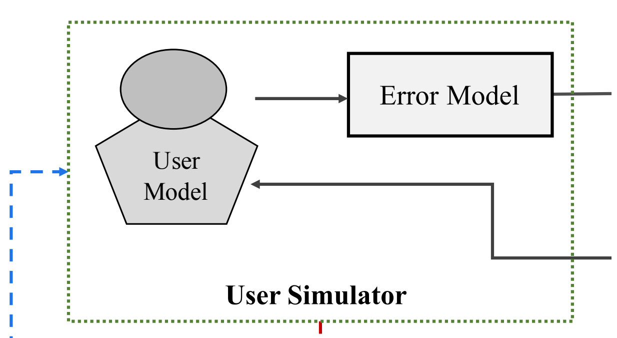


RL-Based Framework



RL-Based Framework



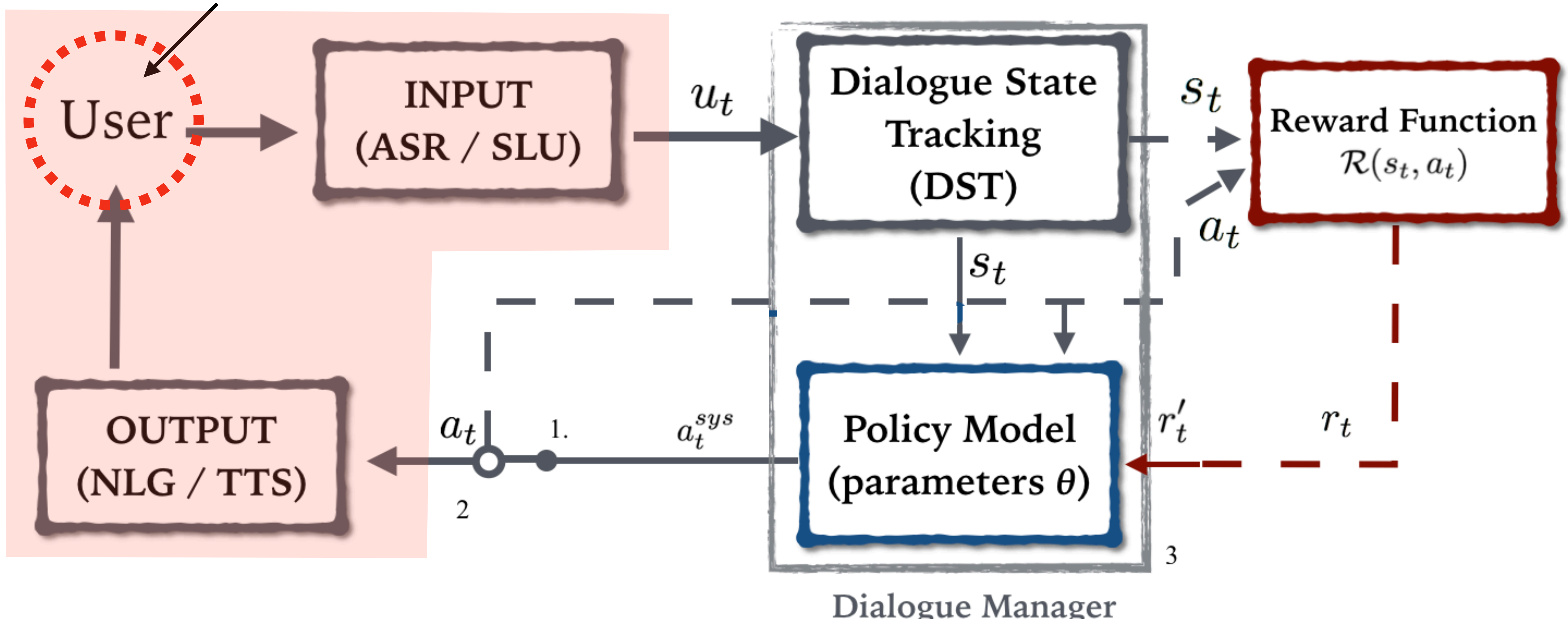


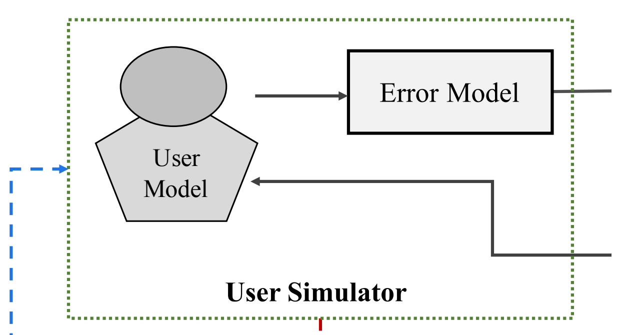
User Model: Simulate User Reactions

Error Model: Simulate the ASR and SLU errors

RL-Based Framework

User Simulator? Pro: Cheap, Convenient Con: Biases

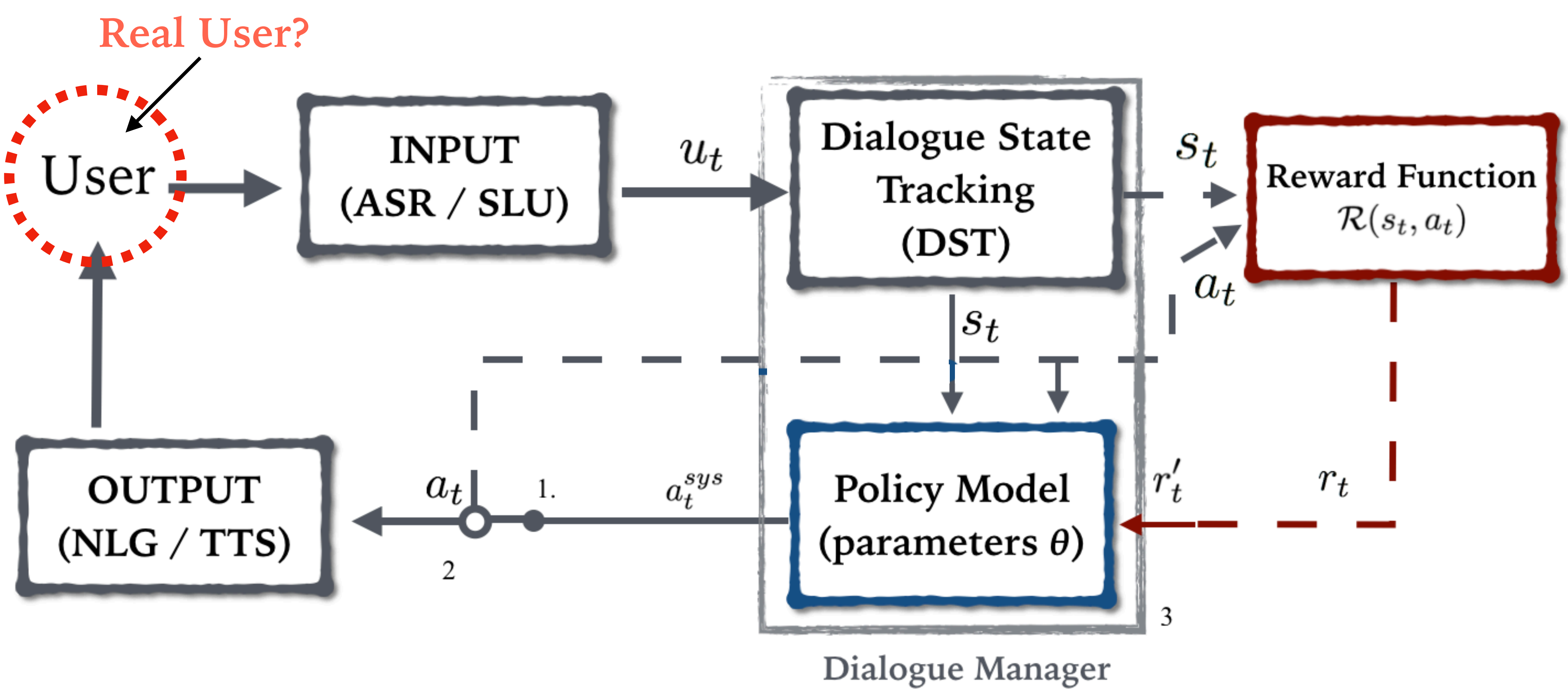




User Model: Simulate User Reactions

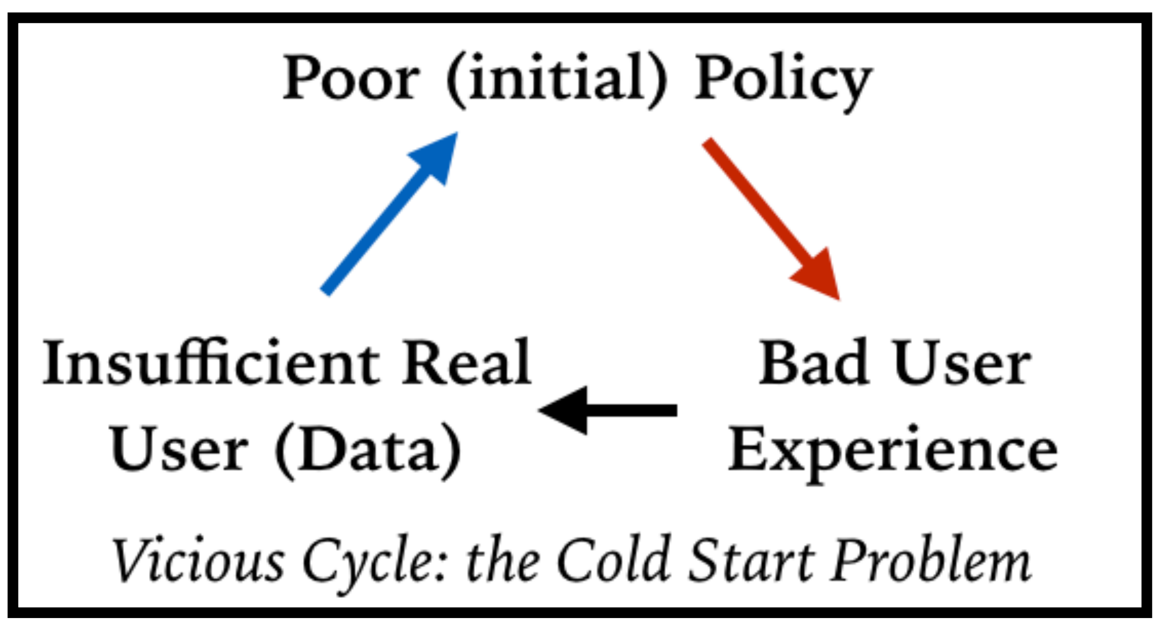
Error Model: Simulate the ASR and SLU errors

RL-Based Framework



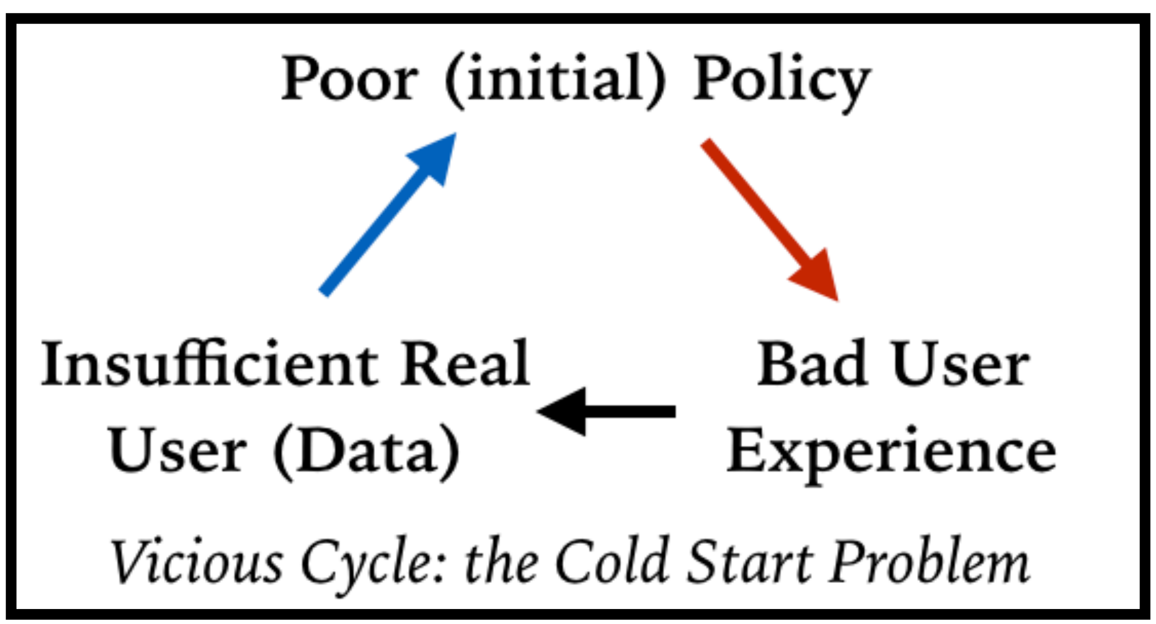
The Cold Start Problem

Rule-Based Methods → Data-Driven Methods



The Cold Start Problem

Rule-Based Methods → **Data-Driven Methods**



- → Inefficient Learning Process ✓
- → Unsafe System Behavior ✓
- → Individual Rationality ✗

Therefore, an ideal on-line policy learning framework should be measured with following two criteria:

→ Inefficient Learning Process (Solvable) ✓

→ Unsafe Policy Behavior (Solvable) ✓

Therefore, an ideal on-line policy learning framework should be measured with following two criteria:

→ Inefficient Learning Process (Solvable) ✓

Efficiency reflects how long it takes for the on-line policy learning algorithm to reach a satisfactory performance level.

→ Unsafe Policy Behavior (Solvable) ✓

Therefore, an ideal on-line policy learning framework should be measured with following two criteria:

→ Inefficient Learning Process (Solvable) ✓

Efficiency reflects how long it takes for the on-line policy learning algorithm to reach a satisfactory performance level.

→ Unsafe Policy Behavior (Solvable) ✓

Safety* reflects whether the initial policy can satisfy the quality-of-service requirement in real-world scenarios during on-line policy learning period.

Therefore, an ideal on-line policy learning framework should be measured with following two criteria:

→ Inefficient Learning Process (Solvable) ✓

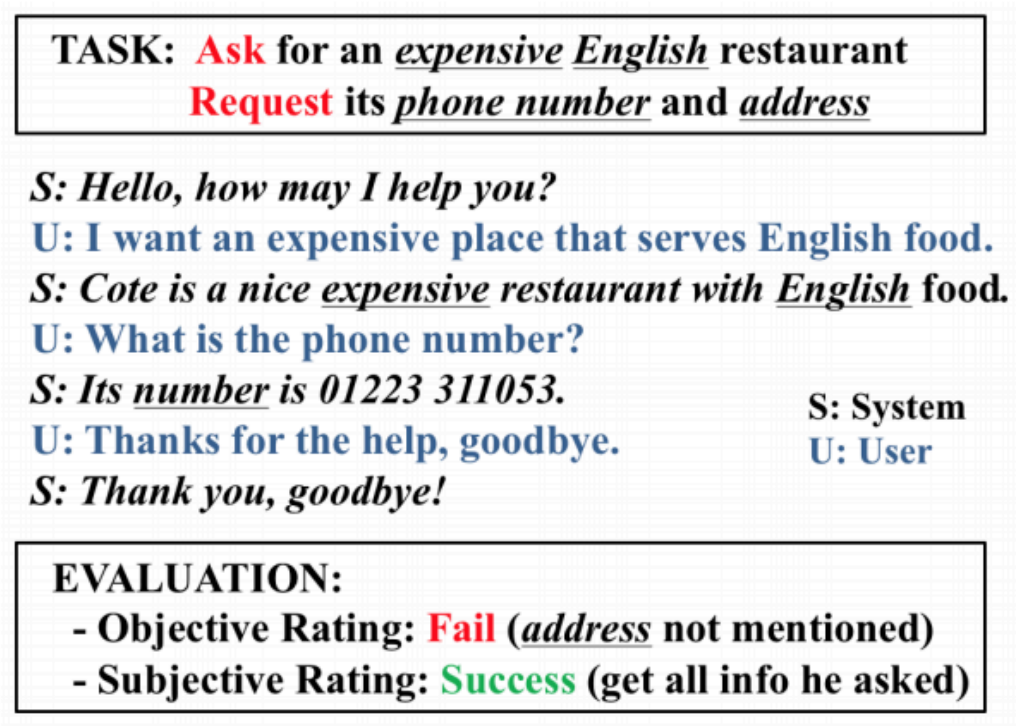
→ Unsafe Policy Behavior (Solvable) ✓

* Most previous studies of on-line policy learning have been focused on the ***efficiency*** **issue**, such as

- **Gaussian process reinforcement learning (GPRL) (Gasic et al., 2010)**,
- Deep reinforcement learning (DRL) (Fatemi et al., 2016; Williams and Zweig, 2016; Su et al., 2016), etc.

Therefore, an ideal on-line policy learning framework should be measured with following two criteria:

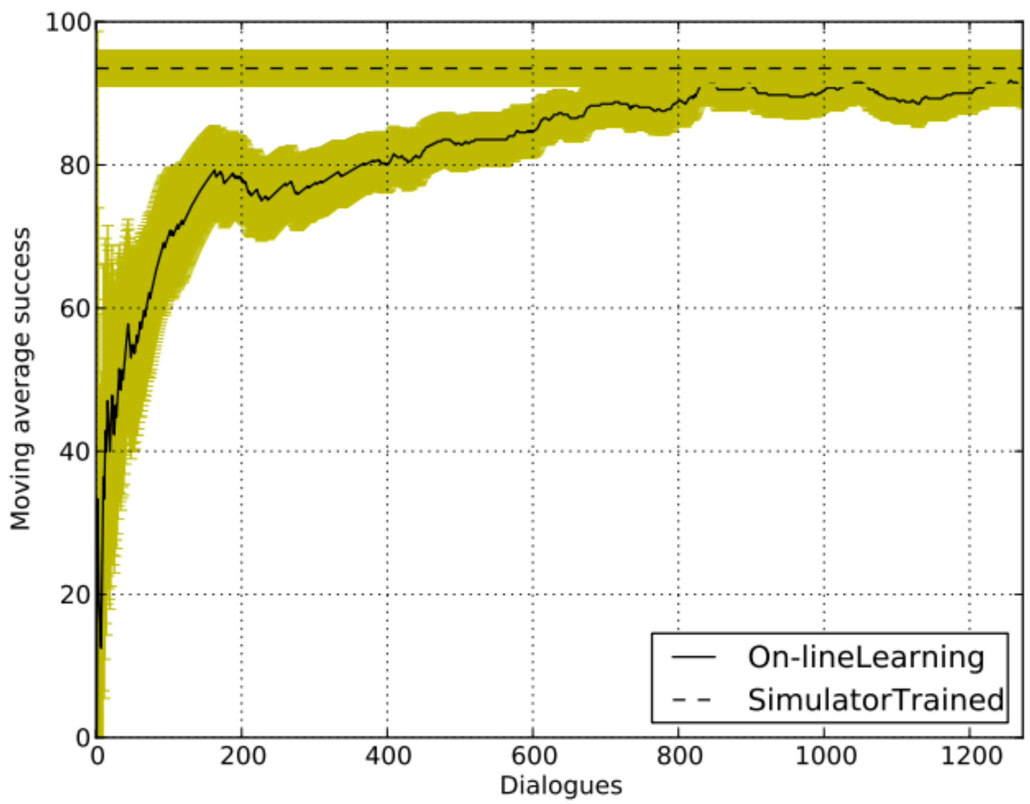
- Inefficient Learning Process (Solvable) ✓
- Unsafe Policy Behavior (Solvable) ✓


TASK: Ask for an *expensive English* restaurant
Request its *phone number* and *address*

S: Hello, how may I help you?
U: I want an expensive place that serves English food.
S: Cote is a nice expensive restaurant with English food.
U: What is the phone number?
S: Its number is 01223 311053.
U: Thanks for the help, goodbye.
S: Thank you, goodbye!

S: System
U: User

EVALUATION:
- Objective Rating: Fail (*address* not mentioned)
- Subjective Rating: Success (get all info he asked)



Therefore, an ideal on-line policy learning framework should be measured with following two criteria:

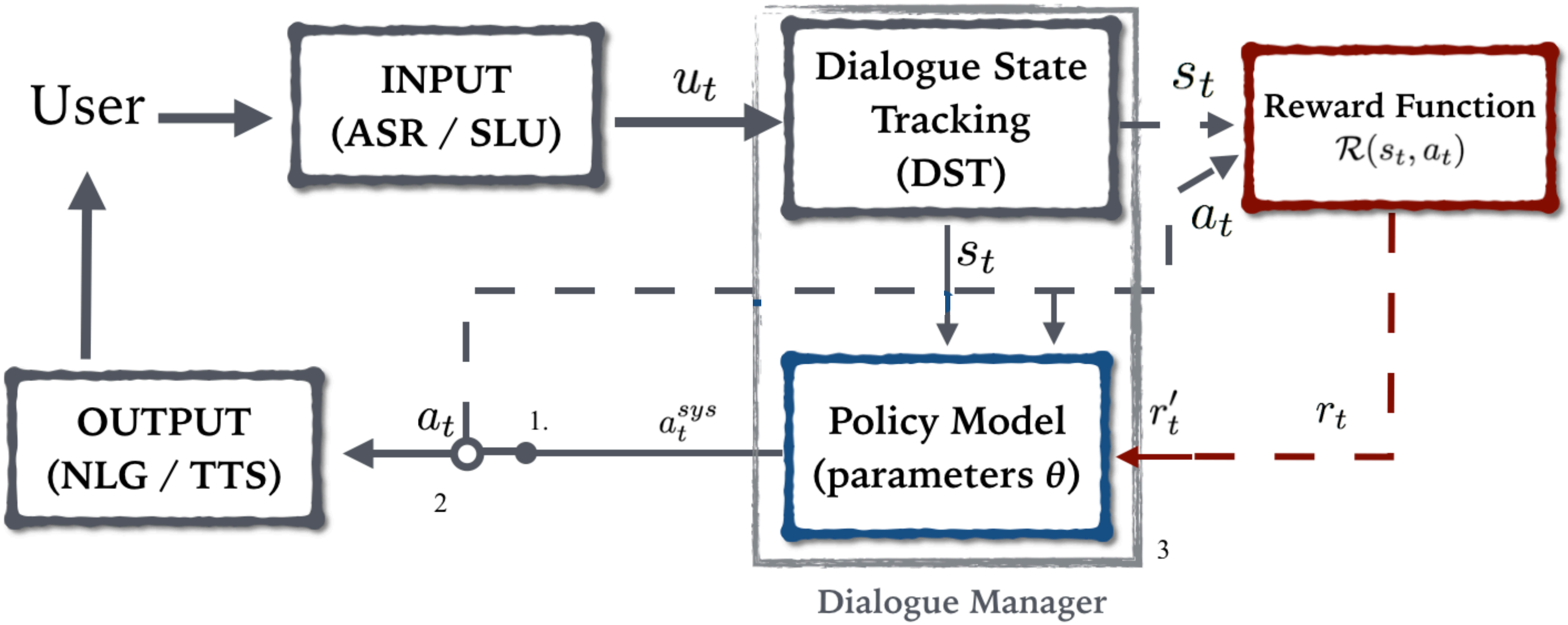
→ Inefficient Learning Process (Solvable) ✓

→ Unsafe Policy Behavior (Solvable) ✓

* However, ***safety*** **is a prerequisite for the efficiency to be achieved**.
  - **Reason**: an unsafe on-line learned policy can consequently fail to attract sufficient real users to continuously improve the policy, no matter how efficient the algorithm is.
  - **Urgency**: on the ***safety*** **issue** which little work has been done.

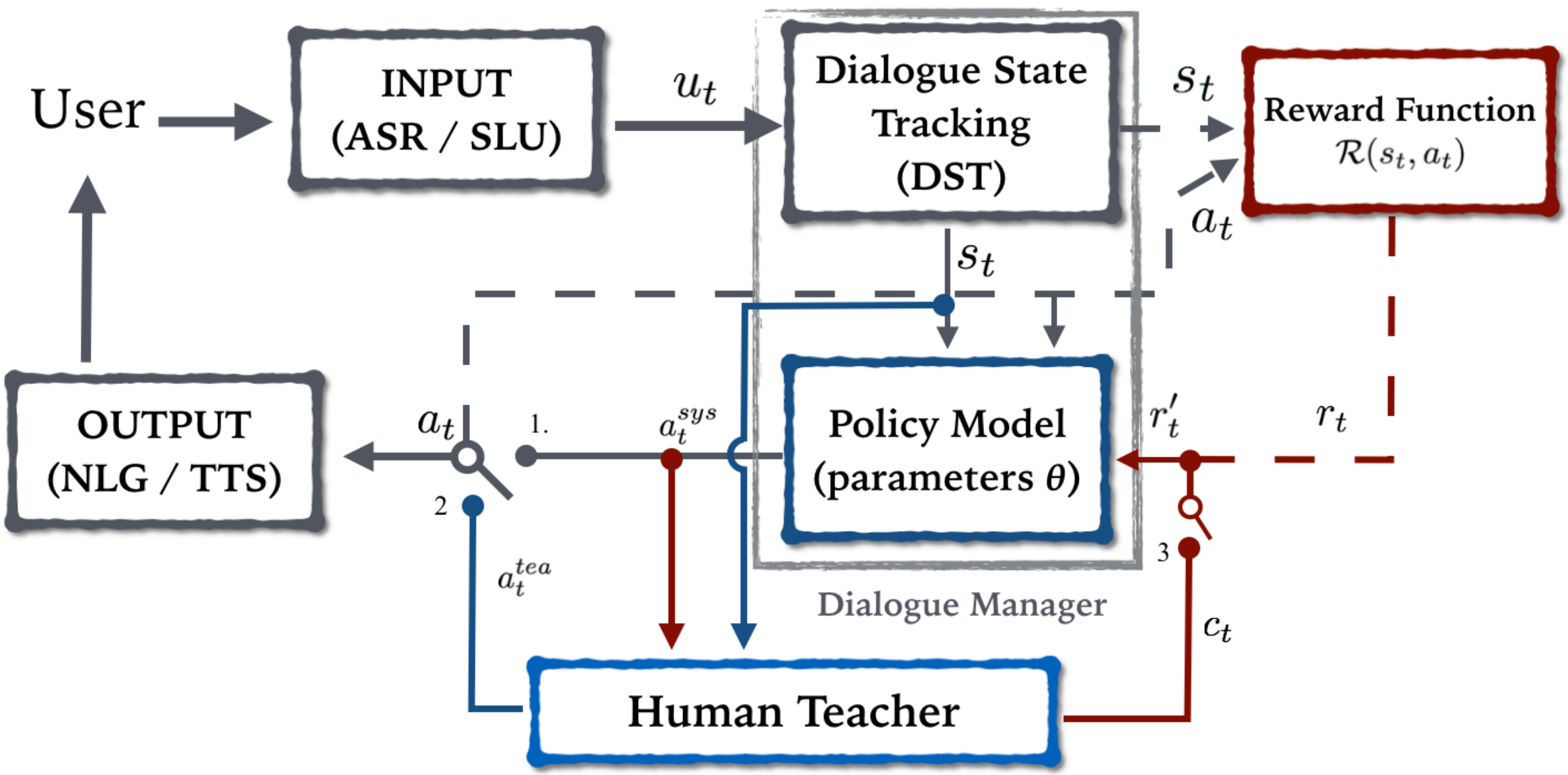
1. A Human-in-the-Loop Solution

Traditional RL Framework

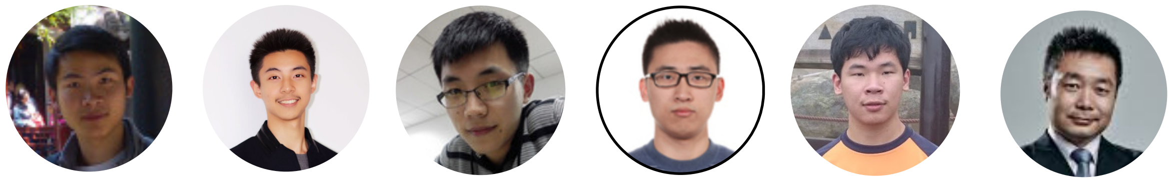


1. A Human-in-the-Loop Solution

Companion Teaching Framework



On-line Dialogue Policy Learning with Companion Teaching
Lu Chen, Runzhe Yang, et.al., **EACL 2017**
http://aclweb.org/anthology/E/E17/E17-2032.pdf

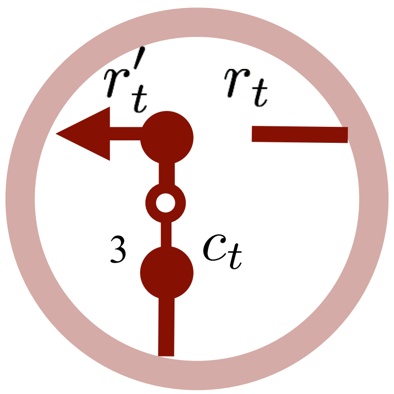


Lu Chen | Runzhe Yang | Cheng Chang | Zihao Ye | Xiang Zhou | Prof. Kai Yu

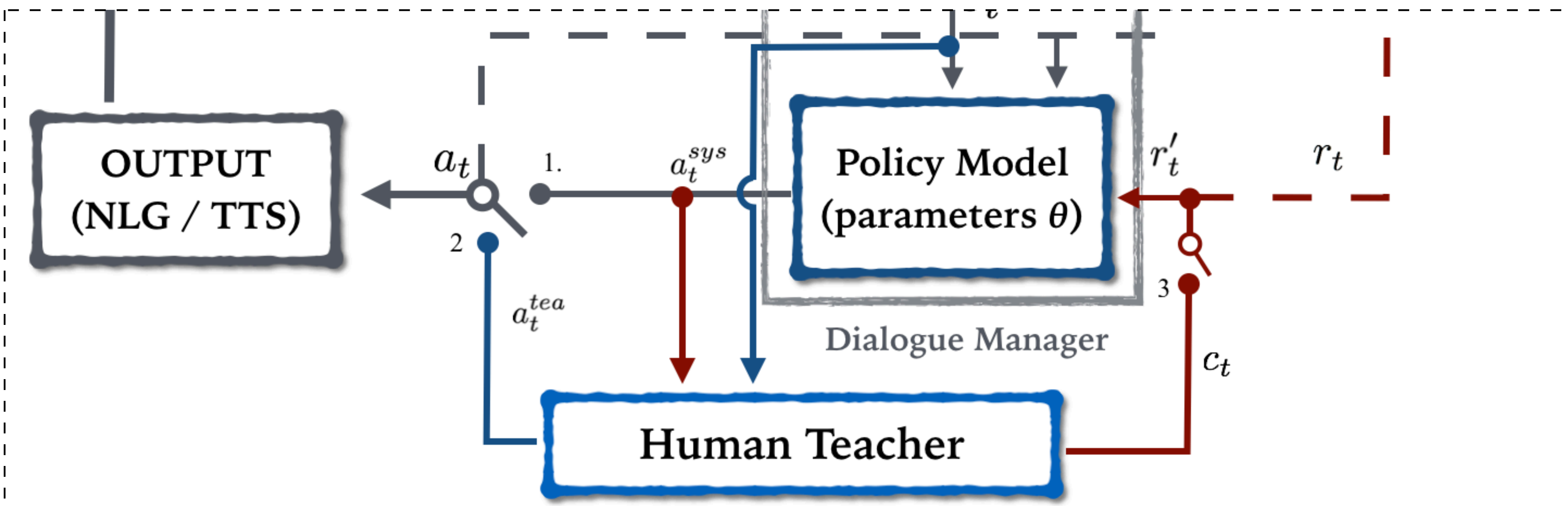
1. A Human-in-the-Loop Solution

SJTU SPEECH LAB
上海交通大學智能語音實驗室

Teaching Strategies

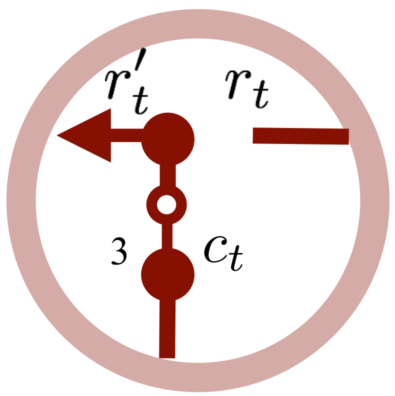


Teaching via
Critic Advice (CA)

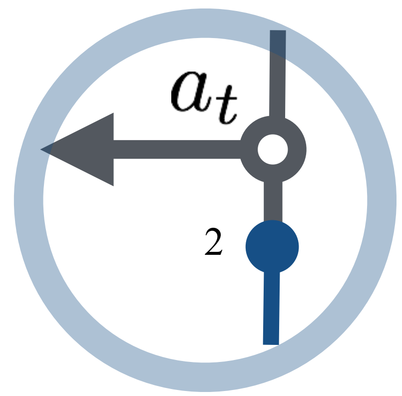


1. A Human-in-the-Loop Solution

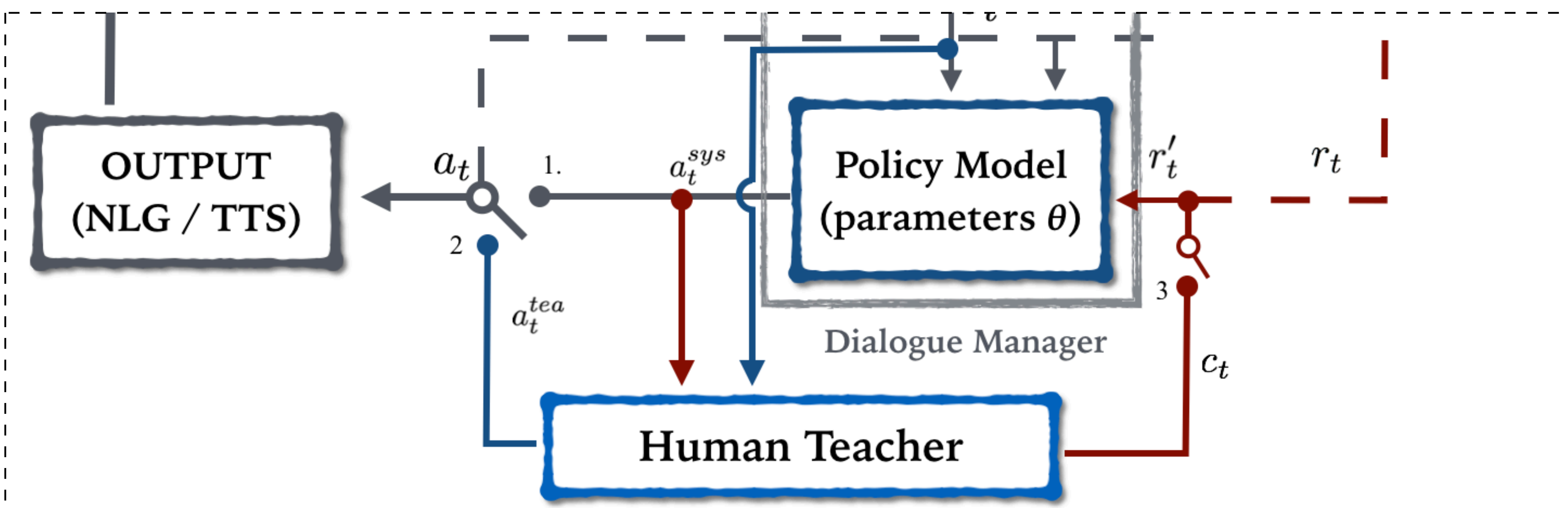
Teaching Strategies



Teaching via
Critic Advice (CA)

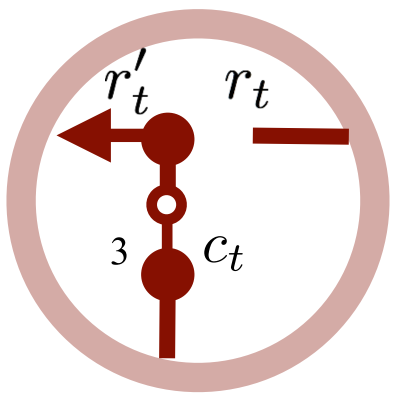


Teaching via
Example Action (EA)

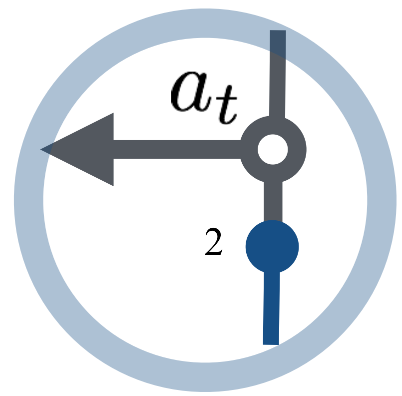


1. A Human-in-the-Loop Solution

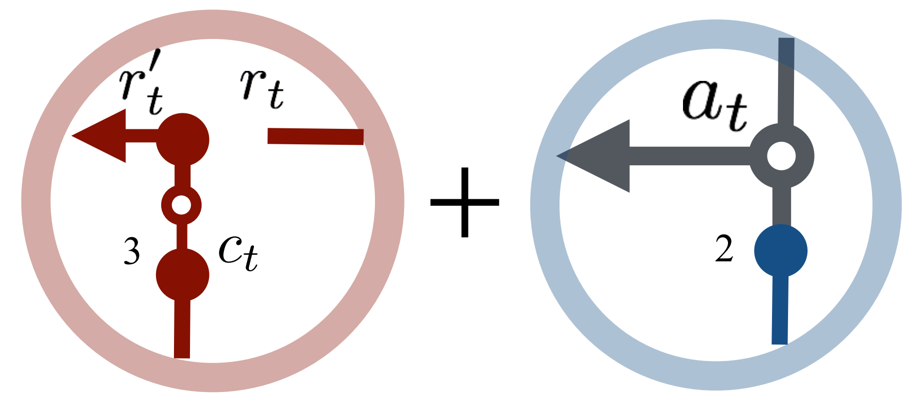
Teaching Strategies



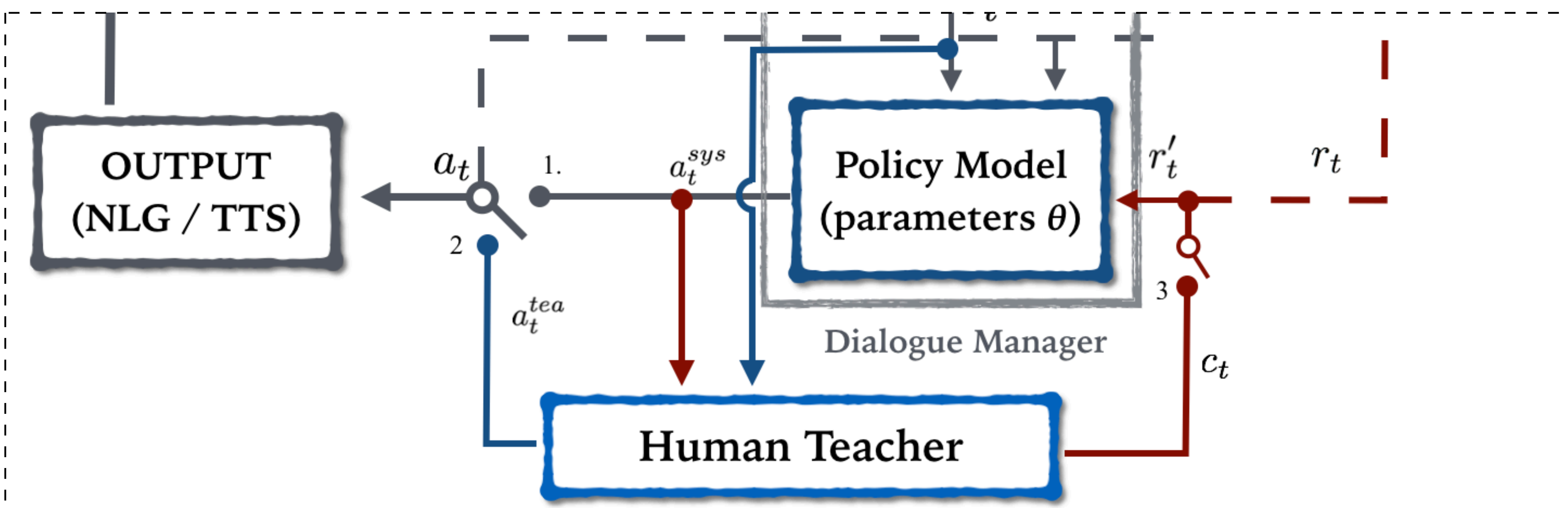
Teaching via
Critic Advice (CA)



Teaching via
Example Action (EA)

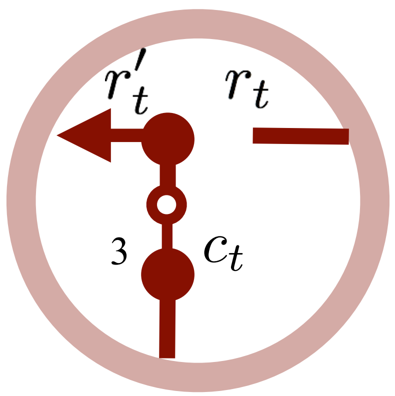


Teaching via Example Action
with Predicted Critique (EAPC)

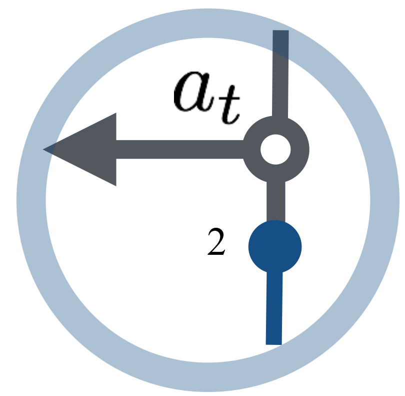


1. A Human-in-the-Loop Solution

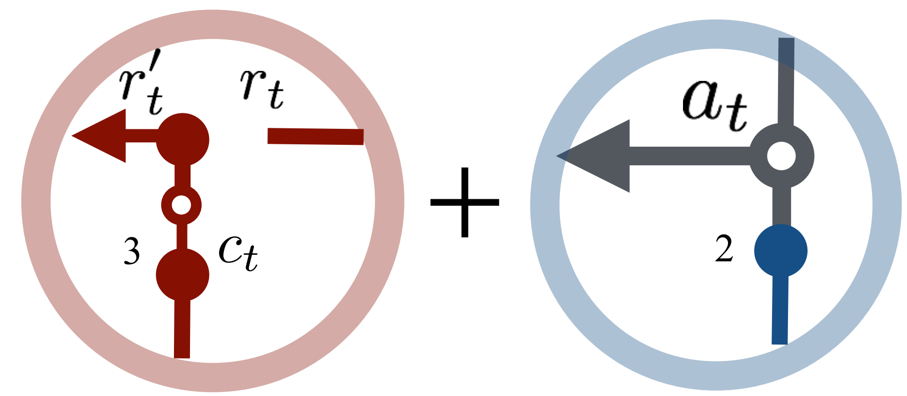
Training with a Replay Buffer



Teaching via
Critic Advice (CA)



Teaching via
Example Action (EA)



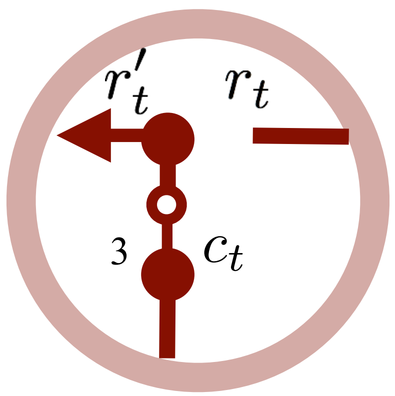
Teaching via Example Action
with Predicted Critique (EAPC)

$$(s_t, a_t, s_{t+1}, r) \sim \mathcal{D}_{replay}$$

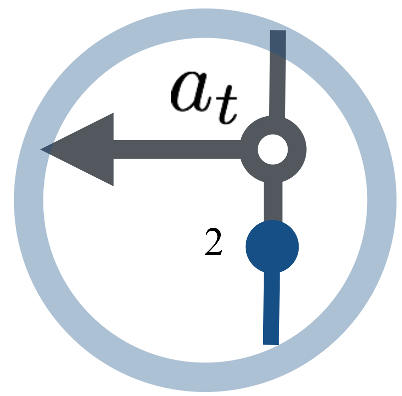
1. A Human-in-the-Loop Solution

SJTU SPEECH LAB
上海交通大學智能語音實驗室

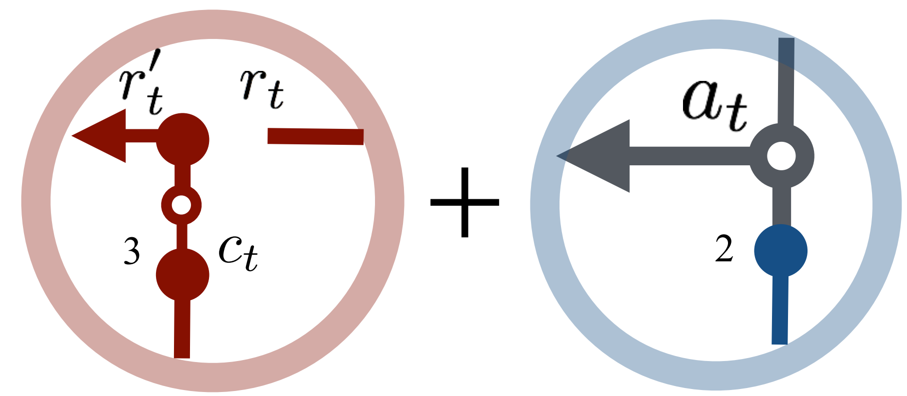
Training with a Replay Buffer



Teaching via
Critic Advice (CA)



Teaching via
Example Action (EA)



Teaching via Example Action
with Predicted Critique (EAPC)

$$(s_t, a_t, s_{t+1}, r) \sim \mathcal{D}_{replay}$$

$$l(\theta) = \mathbb{E}_{s,a\sim\pi_\theta}[(Q_{target} - Q(s_t, a_t, \theta))^2]$$

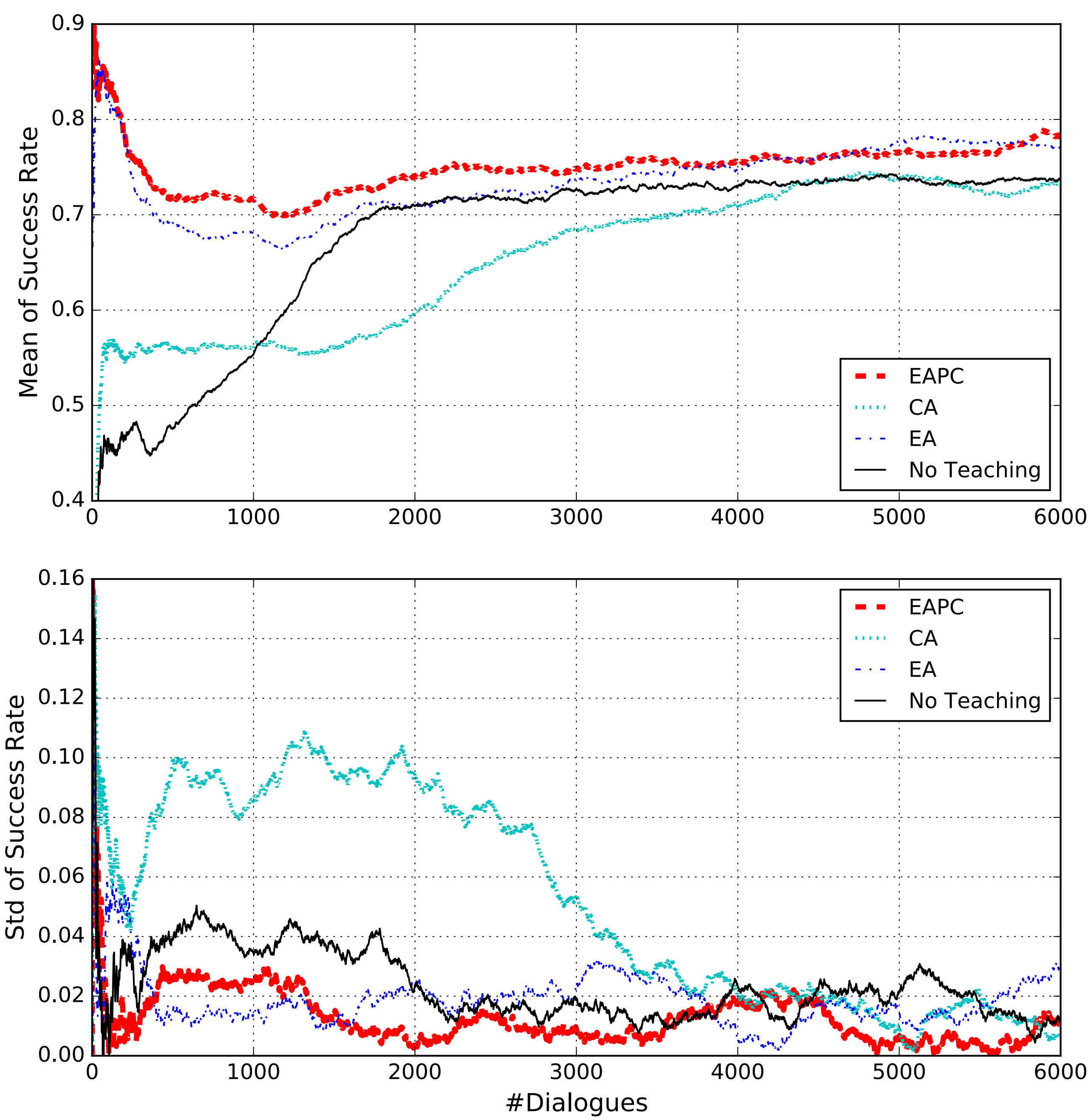
$$Q_{target} = r + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}, \theta)$$

1. A Human-in-the-Loop Solution

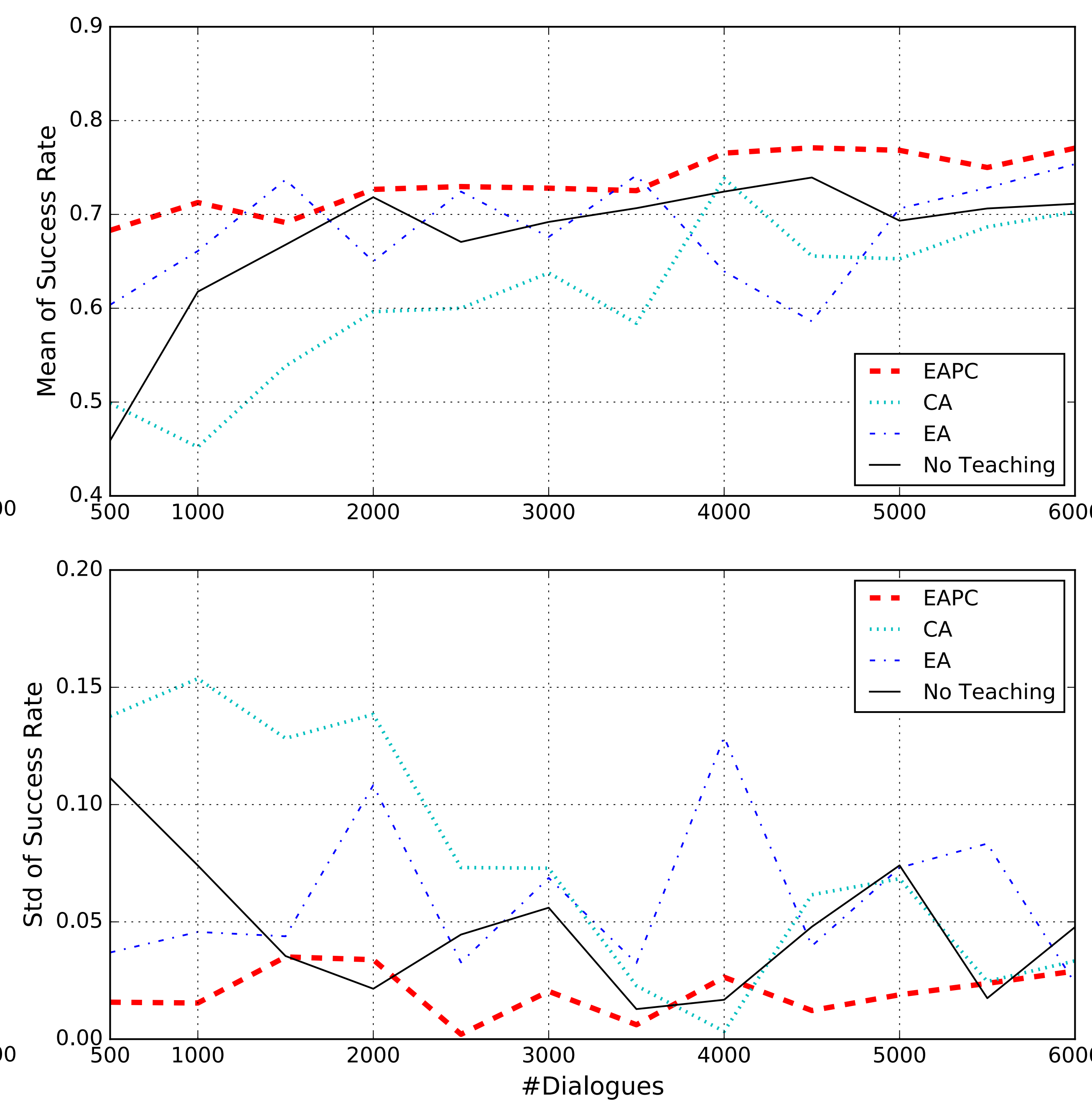
SJTU Speech Lab 上海交通大學智能語音實驗室

- ***Dataset:*** DSTC-2 , ***Teaching Budget:*** 1500 turns
- ***Simulated Teacher:*** a **well-trained policy model** with success rate 0.7

Safety Evaluation



Efficiency Evaluation

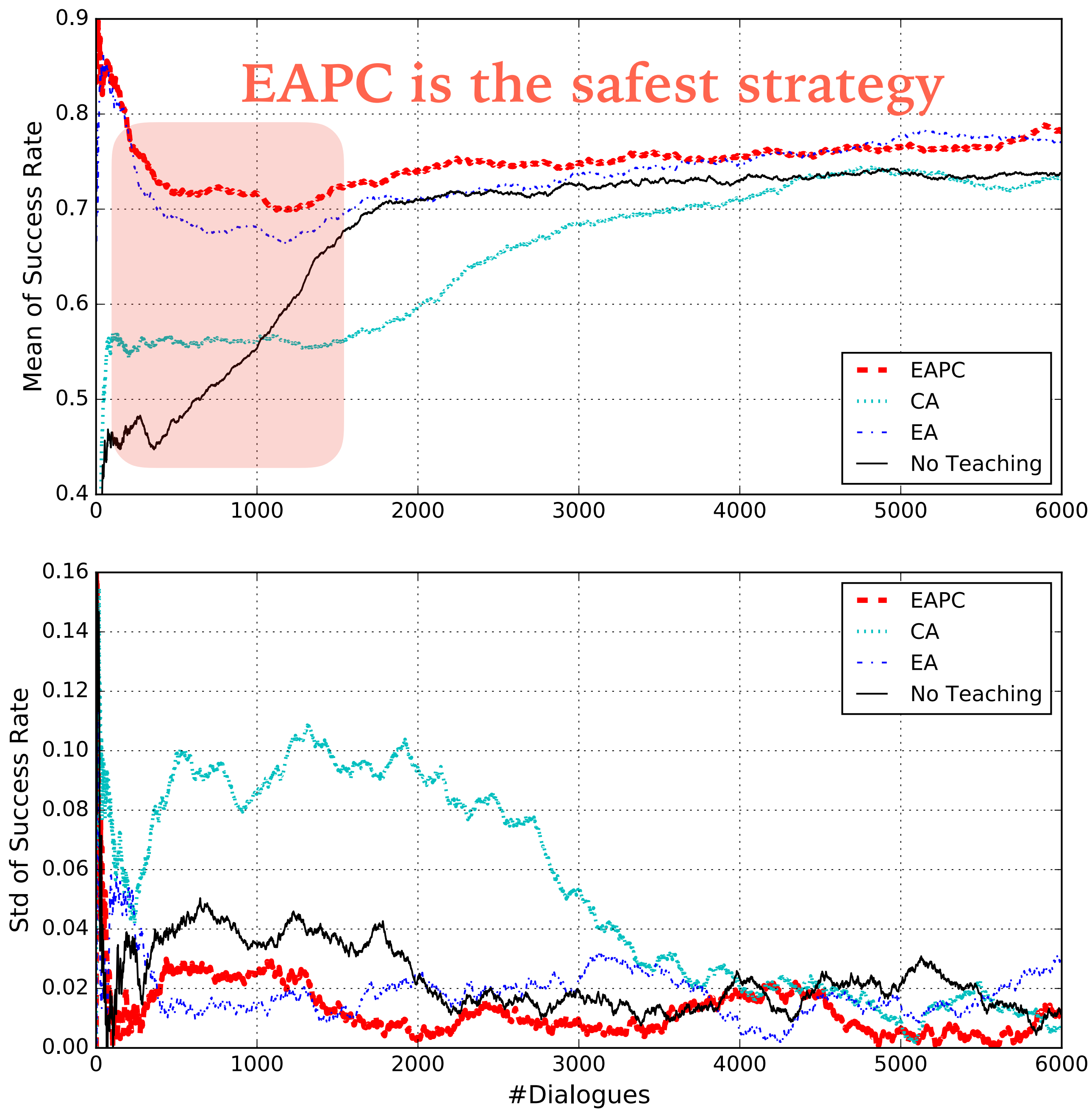


1. A Human-in-the-Loop Solution

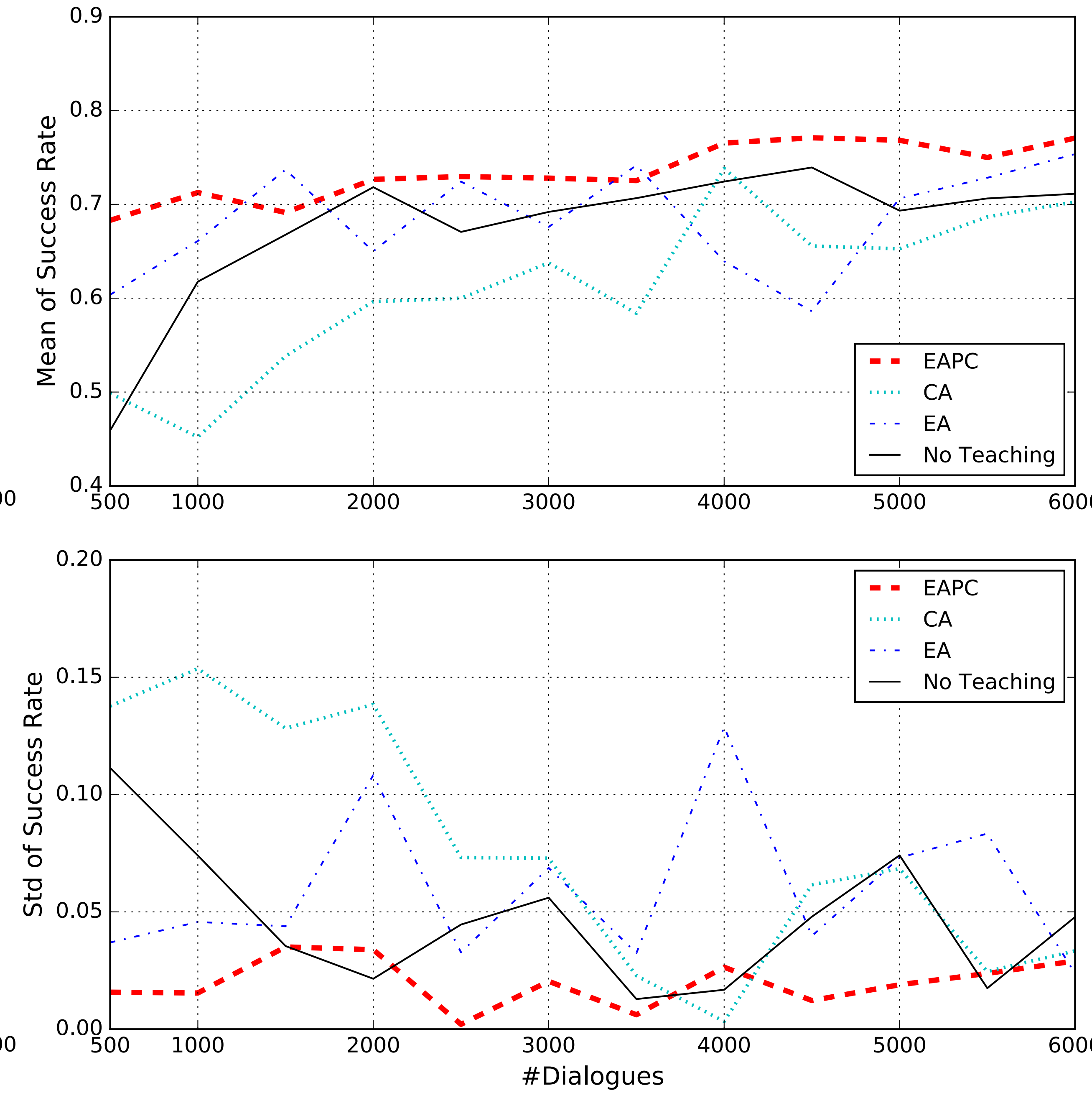
SJTU SPEECH LAB
上海交通大學智能語音實驗室

- ***Dataset:*** DSTC-2 , ***Teaching Budget:*** 1500 turns
- ***Simulated Teacher:*** a **well-trained policy model** with success rate 0.7

Safety Evaluation



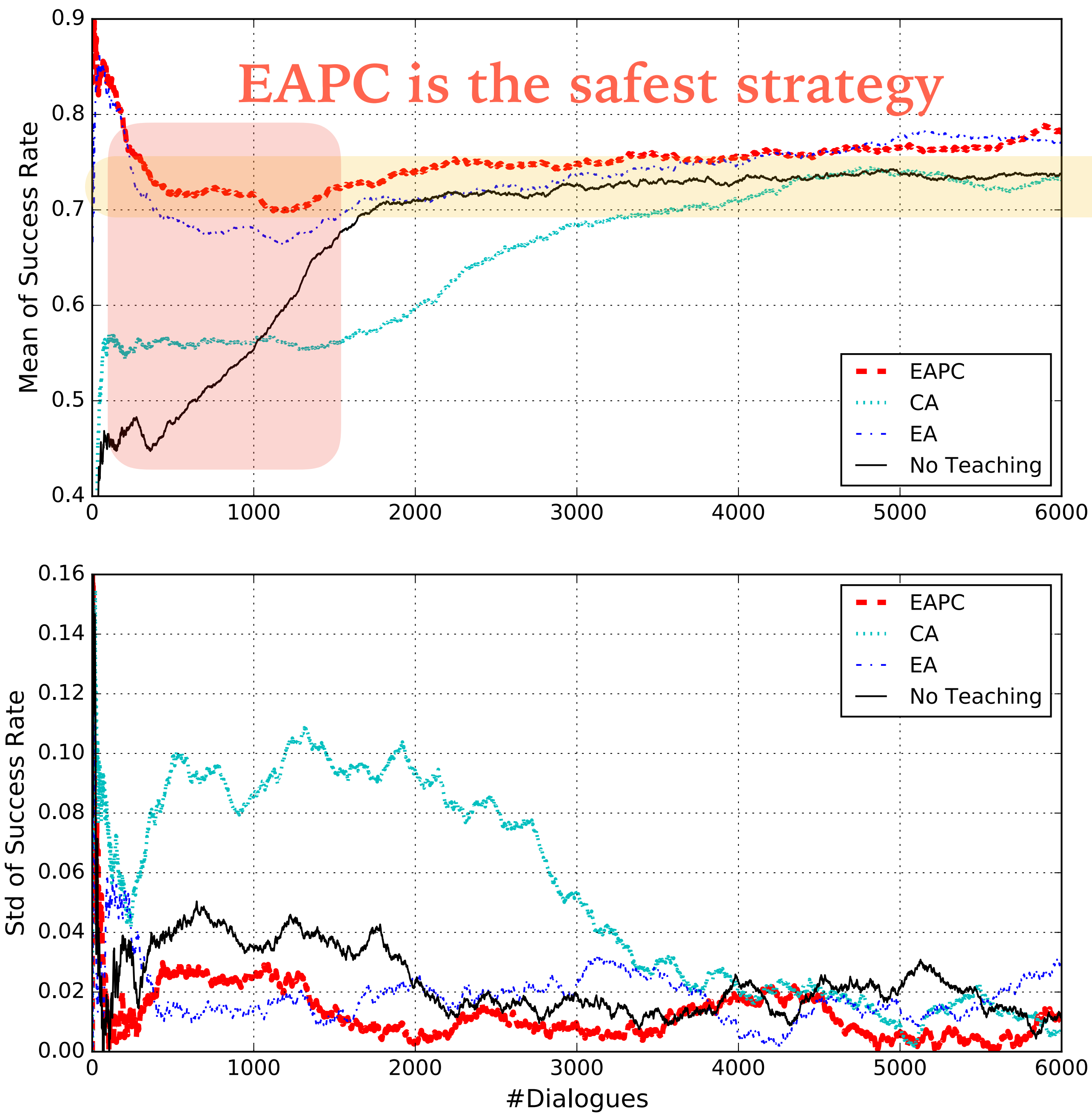
Efficiency Evaluation



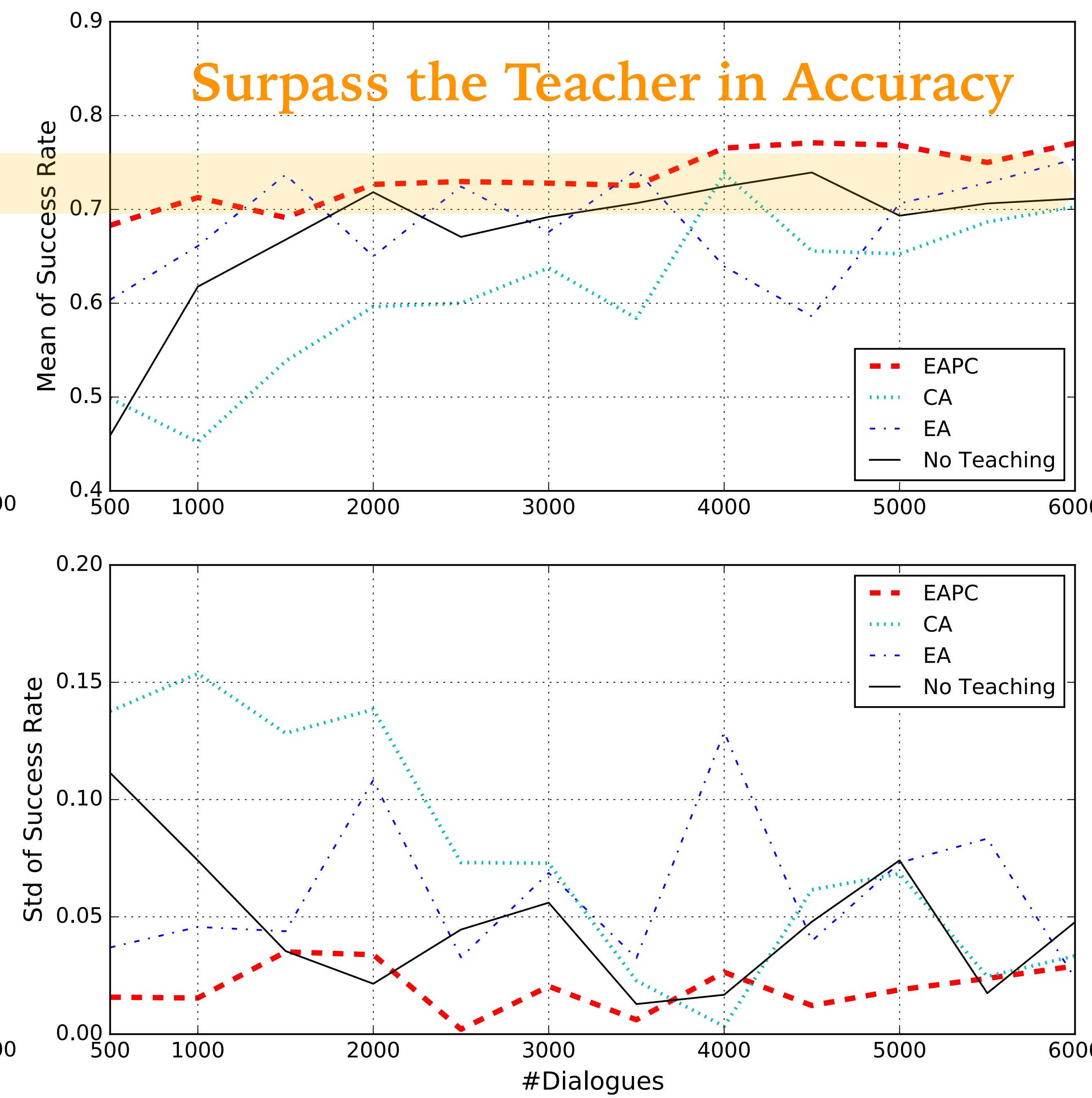
1. A Human-in-the-Loop Solution

- ***Dataset:*** DSTC-2 , ***Teaching Budget:*** 1500 turns
- ***Simulated Teacher:*** a **well-trained policy model** with success rate 0.7

Safety Evaluation

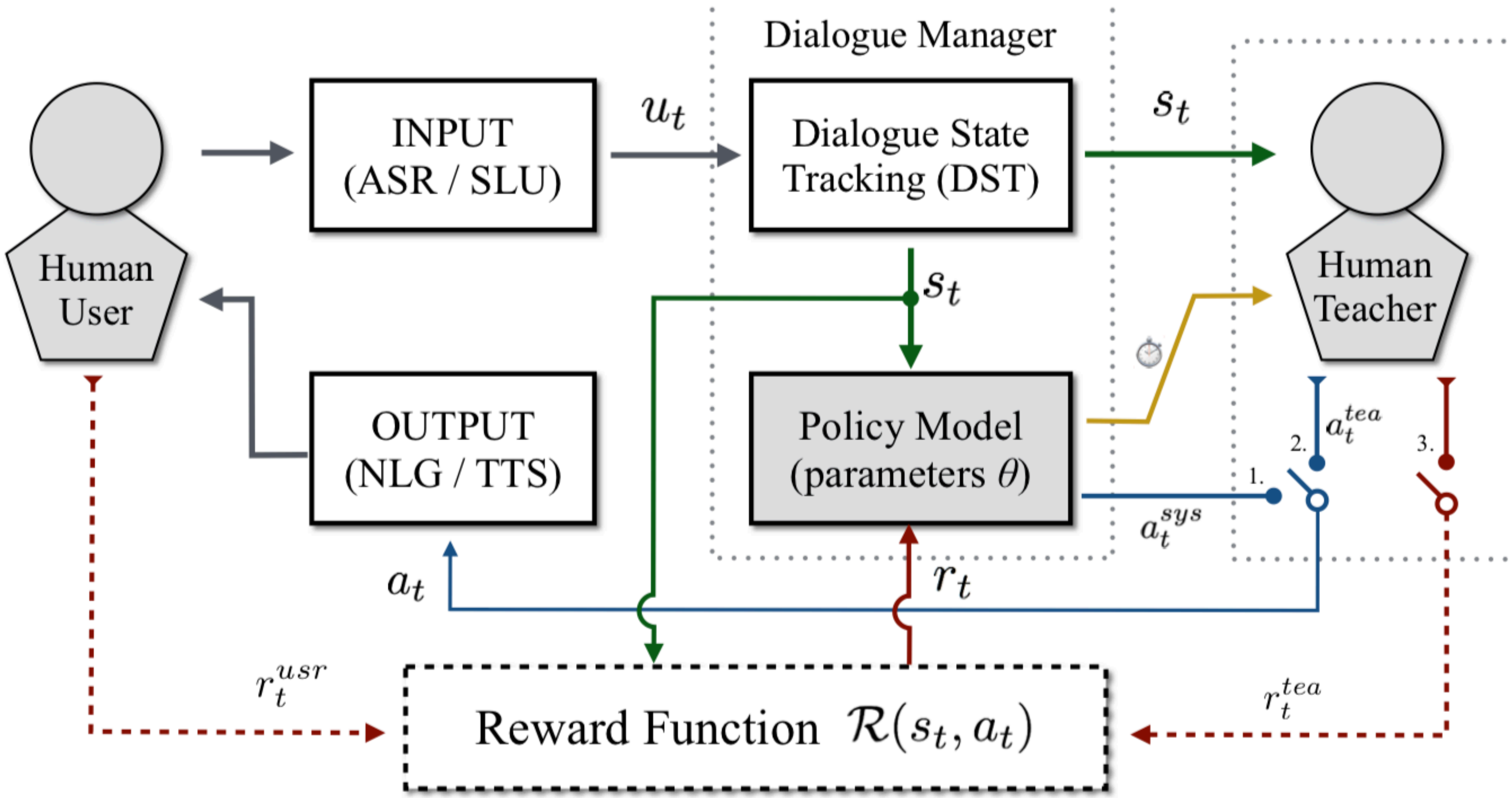


Efficiency Evaluation



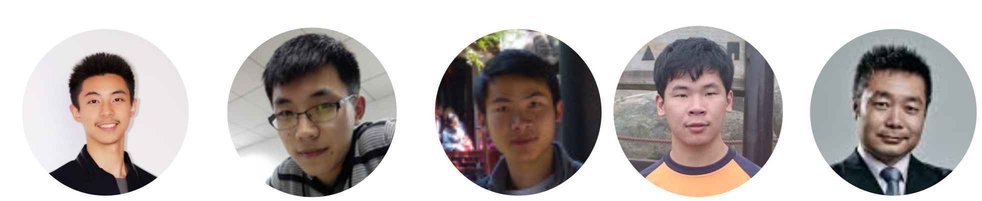
2. A Complete Companion Teaching Framework

When to teach? (Economically Utilize Teaching Budget)



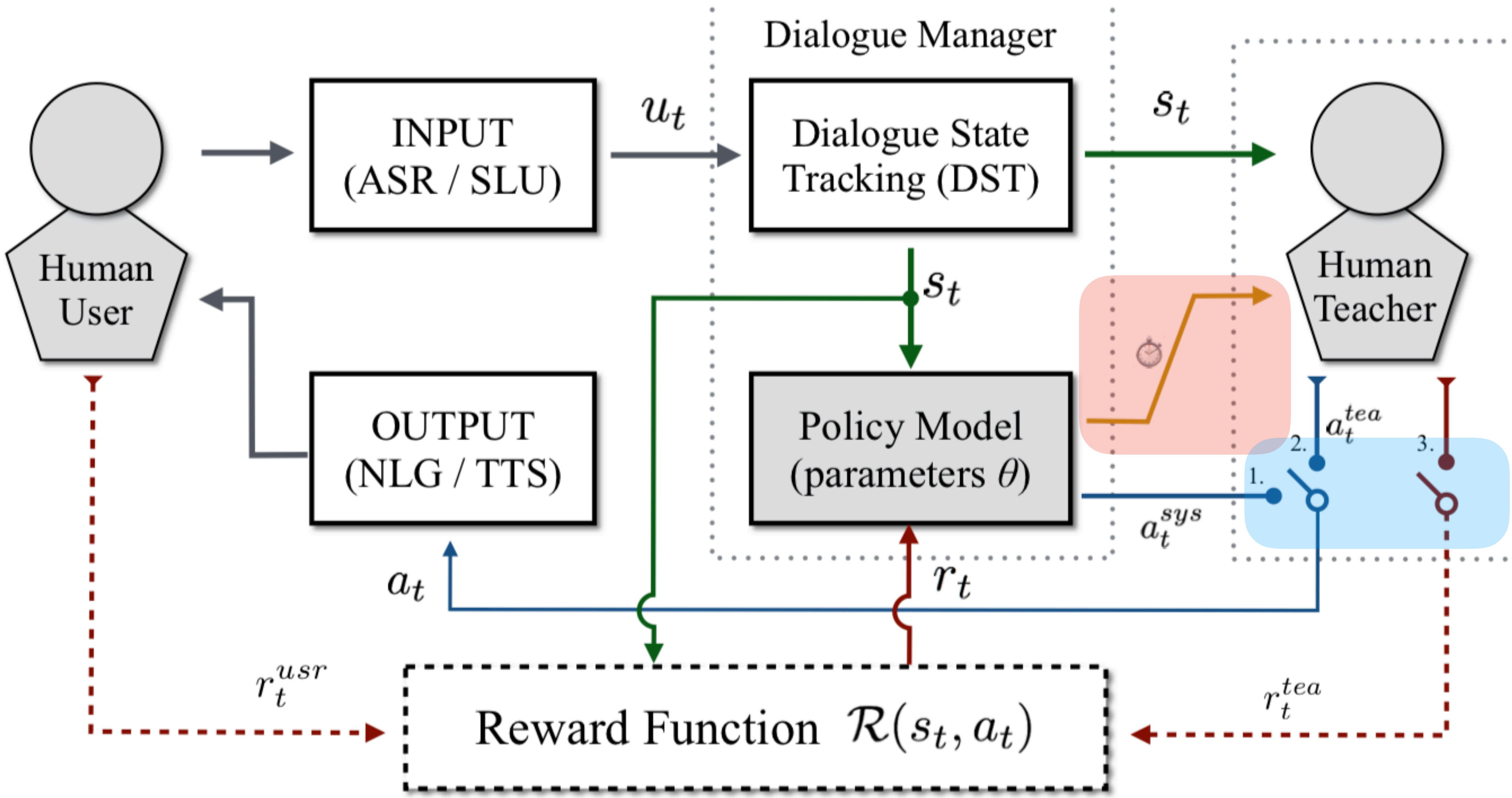
Teaching Scheme = Teaching Heuristic + Teaching Strategy

Affordable On-line Dialogue Policy Learning
Cheng Chang*, Runzhe Yang*, et.al., **EMNLP 2017**
http://www.aclweb.org/anthology/D/D17/D17-1234.pdf



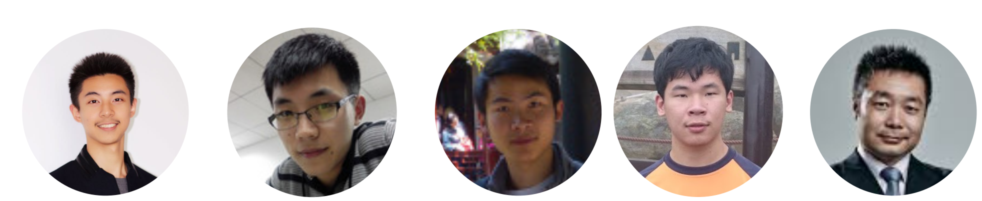
2. A Complete Companion Teaching Framework

When to teach? (Economically Utilize Teaching Budget)



Teaching Scheme = Teaching Heuristic + Teaching Strategy

Affordable On-line Dialogue Policy Learning
Cheng Chang*, Runzhe Yang*, et.al., **EMNLP 2017**
http://www.aclweb.org/anthology/D/D17/D17-1234.pdf



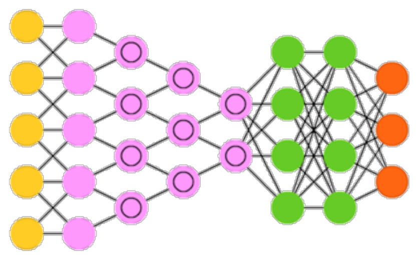
Runzhe Yang* Cheng Chang* Lu Chen Xiang Zhou Prof. Kai Yu

2. A Complete Companion Teaching Framework

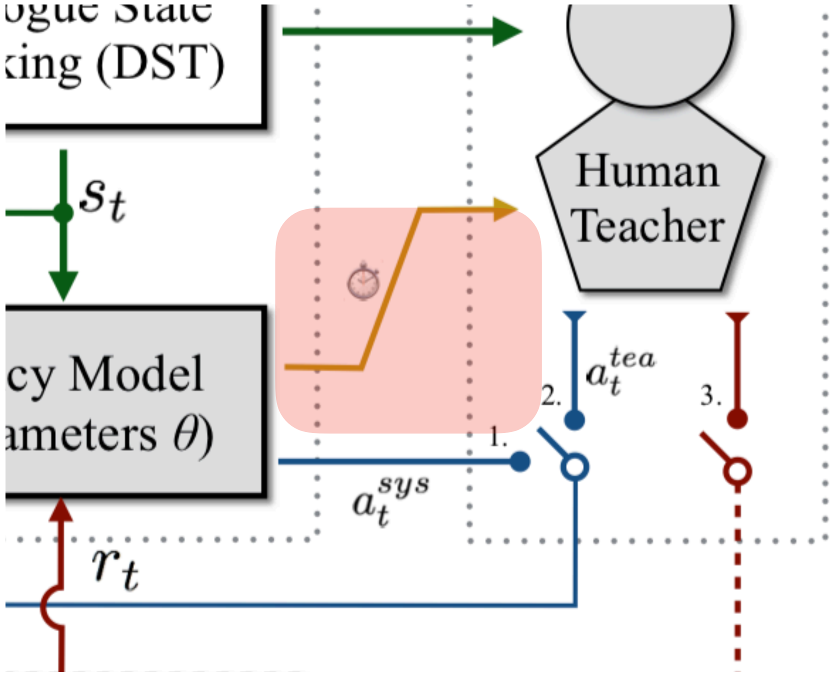
When to teach? (Economically Utilize Teaching Budget)

State Importance

Torrey and Taylor (2013):



$$I(s) = max_a Q_{(s,a)} - min_a Q_{(s,a)}$$



Teach when the current state is IMPORTANT:

$$I(s) > t_{si}$$

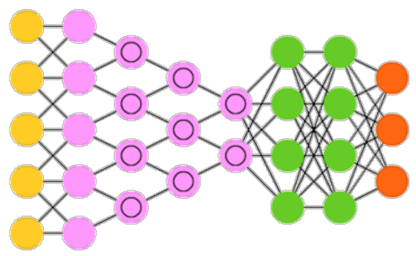
Teaching Scheme = Teaching Heuristic + Teaching Strategy

2. A Complete Companion Teaching Framework

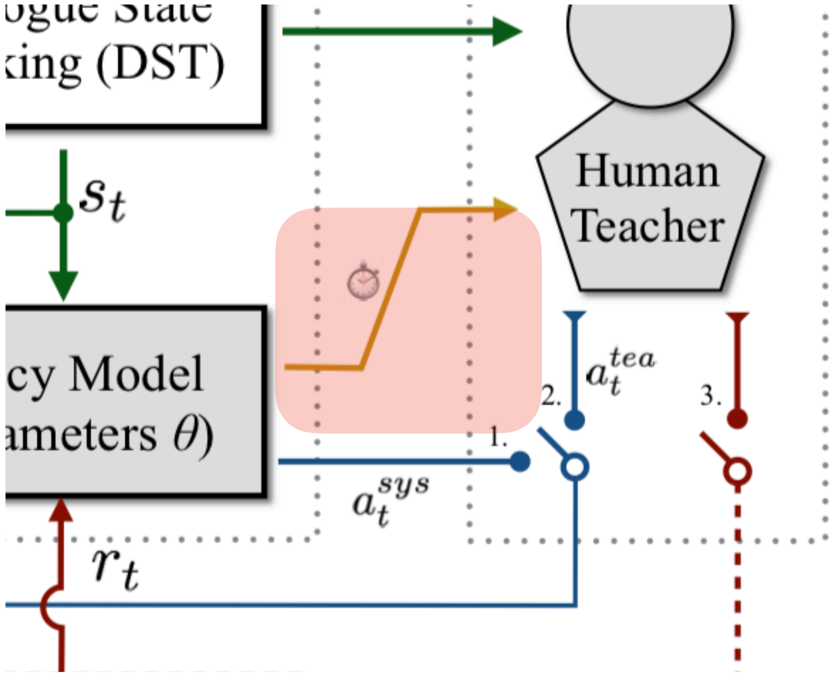
When to teach? (Economically Utilize Teaching Budget)

State Importance

Torrey and Taylor (2013):



$$I(s) = max_a Q_{(s,a)} - min_a Q_{(s,a)}$$



Teach when the student is UNCERTAIN:

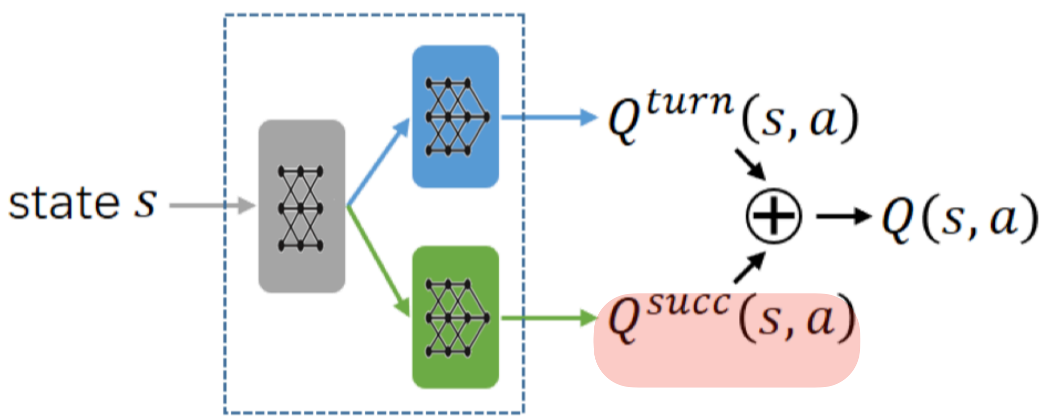
$$I(s) < t_{su}$$

Teaching Scheme = Teaching Heuristic + Teaching Strategy

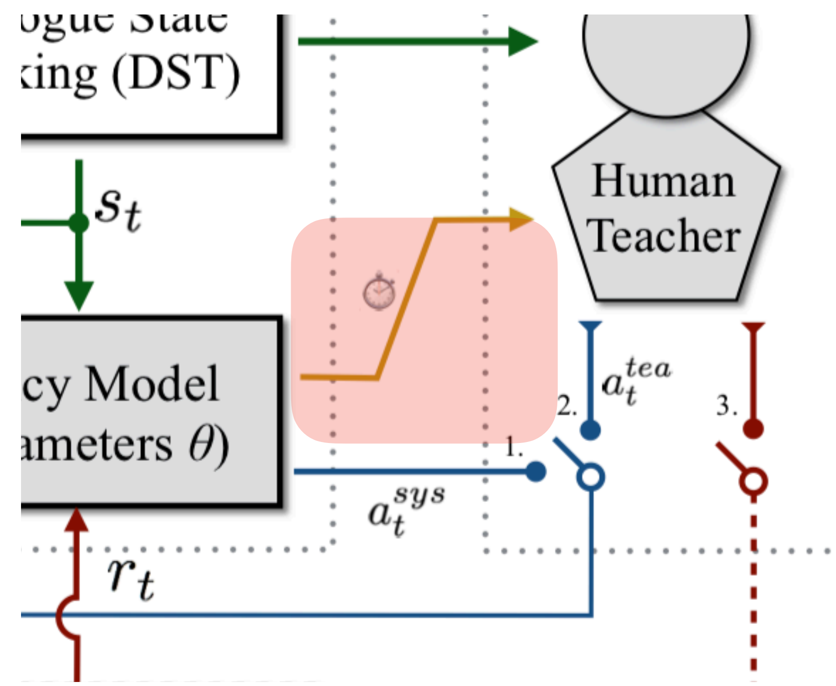
2. A Complete Companion Teaching Framework

When to teach? (Economically Utilize Teaching Budget)

Failure Prognosis based Teaching heuristic (FTP)



MultiTask-DQN Structure



Teach when the dialogue is likely to fail:

$$Q^{\text{succ}}(s_t, a_t) < \alpha \frac{1}{w} \sum_{j=t-w}^{t-1} Q^{\text{succ}}(s_j, a_j)$$

Teaching Scheme = Teaching Heuristic + Teaching Strategy

2. A Complete Companion Teaching Framework

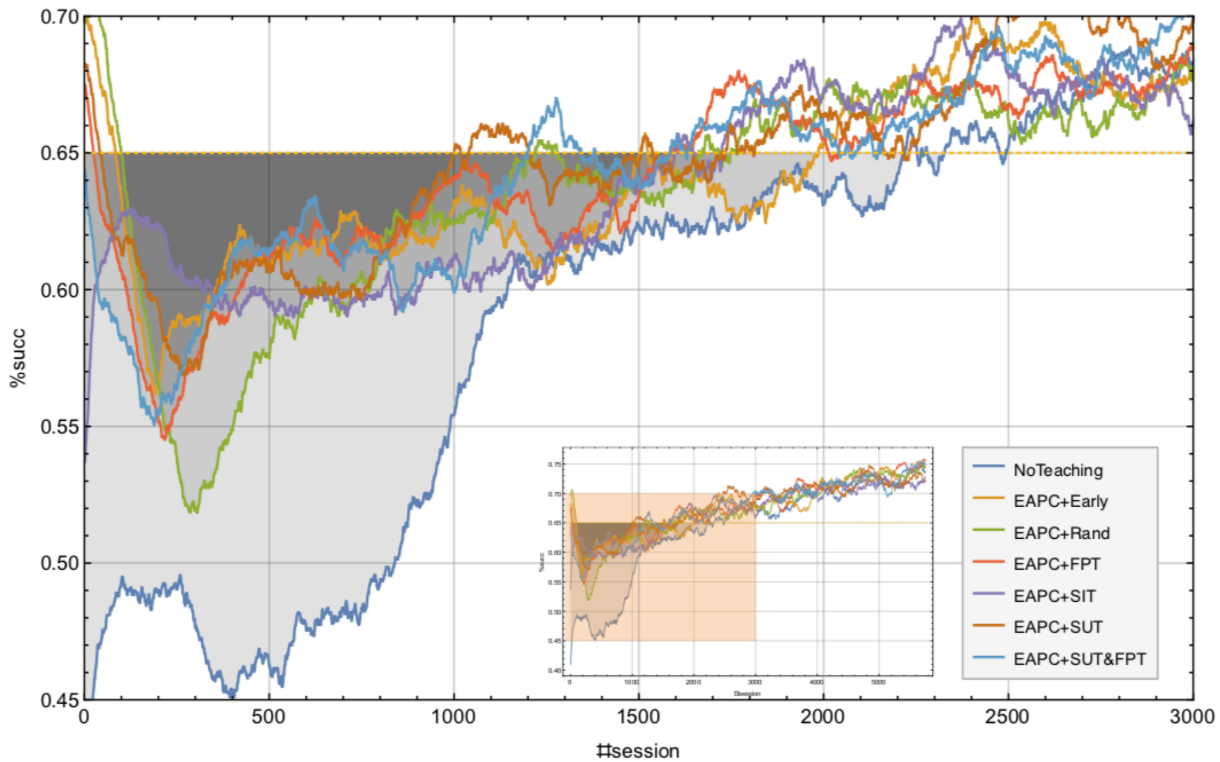
When to teach? (Economically Utilize Teaching Budget)

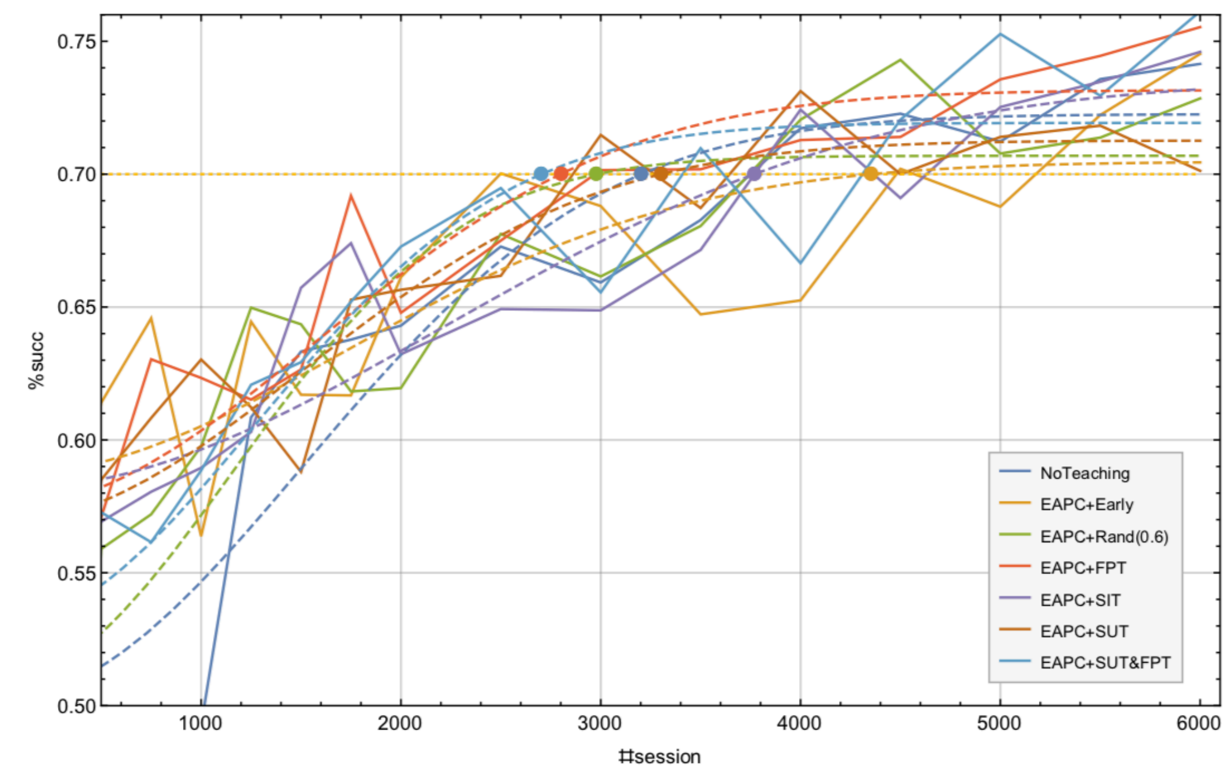
TASK: ask for moderate chinese restaurant & request its phone number

	Dialogue Turn	Conf. Score	|Q^turn |	Q^succ	Teaching?
System	Hello, how may I help you?				
User	[Top ASR] I would like it to be moderate.	0.68	6.05	**0.911**	FALSE
System	Could you repeat it?				
User	[Top ASR] I would like it to be moderate.	0.81	5.35	**0.879**	FALSE
System	Could you repeat it?				
User	[Top ASR] Moderate.	0.57	3.31	**0.681**	**TRUE**
Teacher	**Do you want a moderate restaurant?**				
User	[Top ASR] Yes, a moderate chinese restaurant	0.95	3.19	**0.914**	FALSE
System	Seven Day is a nice chinese restaurant in the north part.				
User	[Top ASR] The phone number.	0.92	1.23	**0.942**	FALSE
System	1223314954				

2. A Complete Companion Teaching Framework

How to evaluate the safety and the efficiency?





Risk Index: $\texttt{RI} = \int_{t=0}^{T} \texttt{dis}(t)\delta_{\texttt{risk}}(t)dt,$

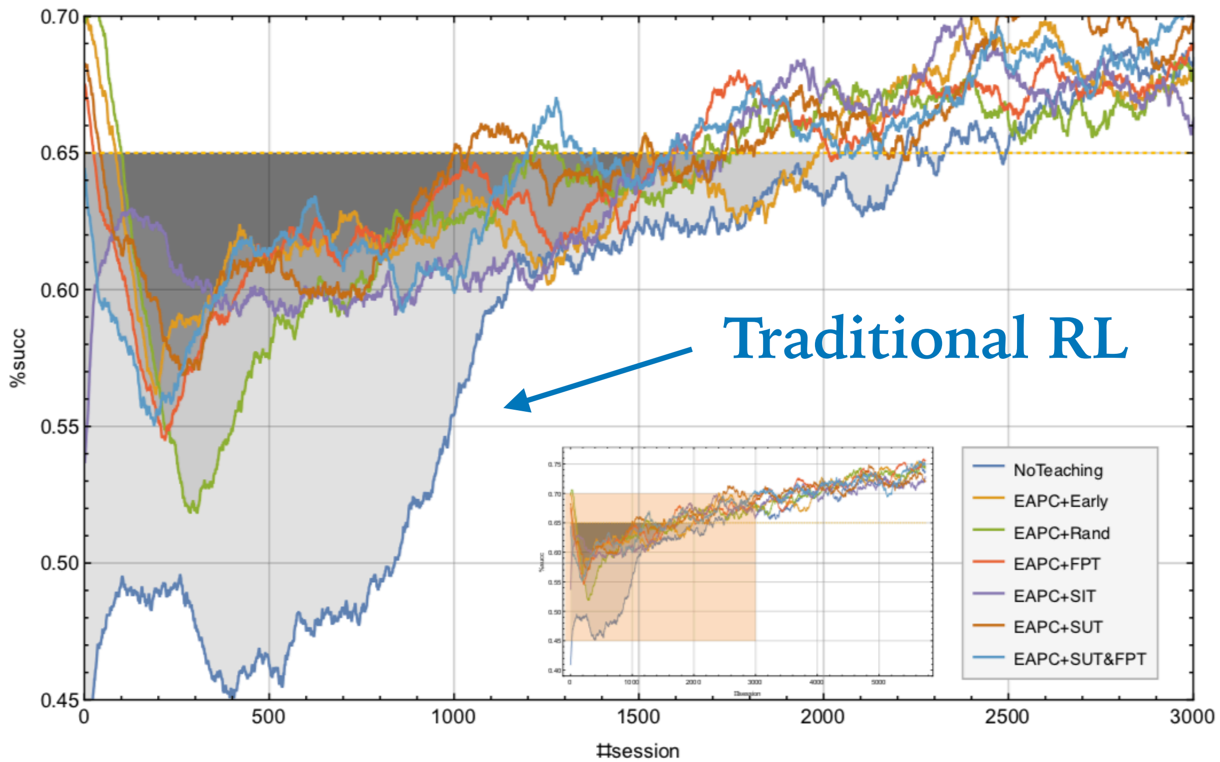
Hitting Time: $\texttt{HT} = c\sqrt{\ln\left(\frac{b}{a-\tau}\right)}.$

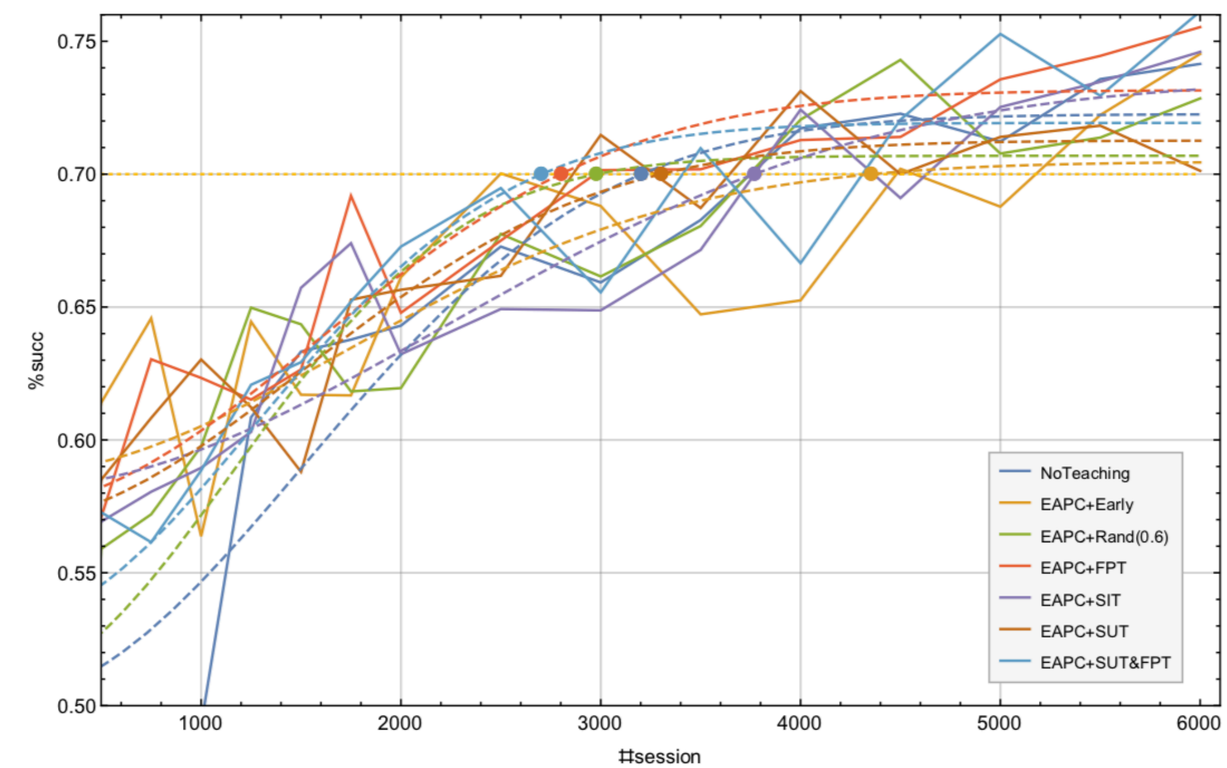
	CA	EA	EAPC
Early	**98.5**	110.6	56.1
Rand	193.4	102.4	65.5
FPT	154.4	**86.2**	**53.6**
SIT	230.8	121.7	66.0
SUT	183.5	**95.8**	**44.5***
SUT&FPT	**131.6**	101.8	54.6
NoTeaching	202.9		

	CA	EA	EAPC
Early	3390.9	3479.4	4354.7
Rand	3669.0	3518.5	2979.2
FPT	**3089.4**	**2921.1**	**2798.4**
SIT	3576.4	4339.7	3768.7
SUT	3230.4	**2954.5**	3300.2
SUT&FPT	**2890.7**	3393.0	**2702.2***
NoTeaching	3204.1		

2. A Complete Companion Teaching Framework

How to evaluate the safety and the efficiency?





Risk Index: $\texttt{RI} = \int_{t=0}^{T} \texttt{dis}(t)\delta_{\texttt{risk}}(t)dt,$

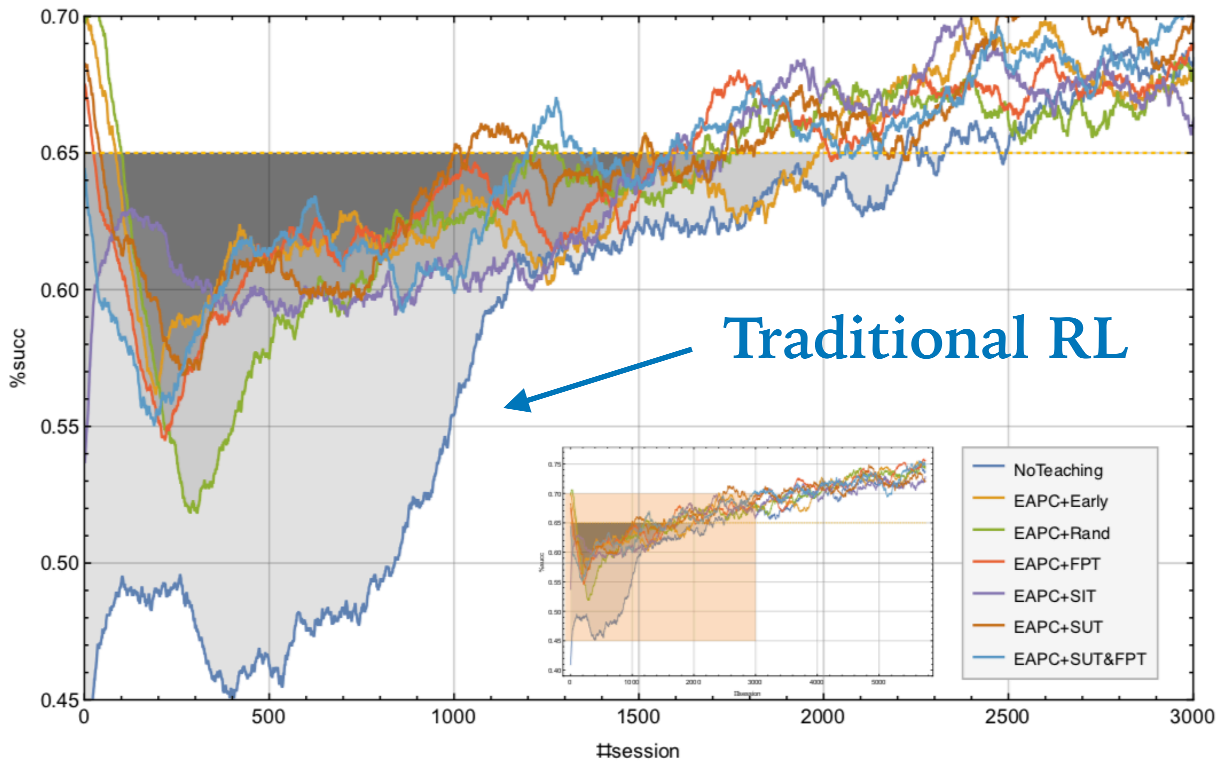
Hitting Time: $\texttt{HT} = c\sqrt{\ln\left(\frac{b}{a-\tau}\right)}.$

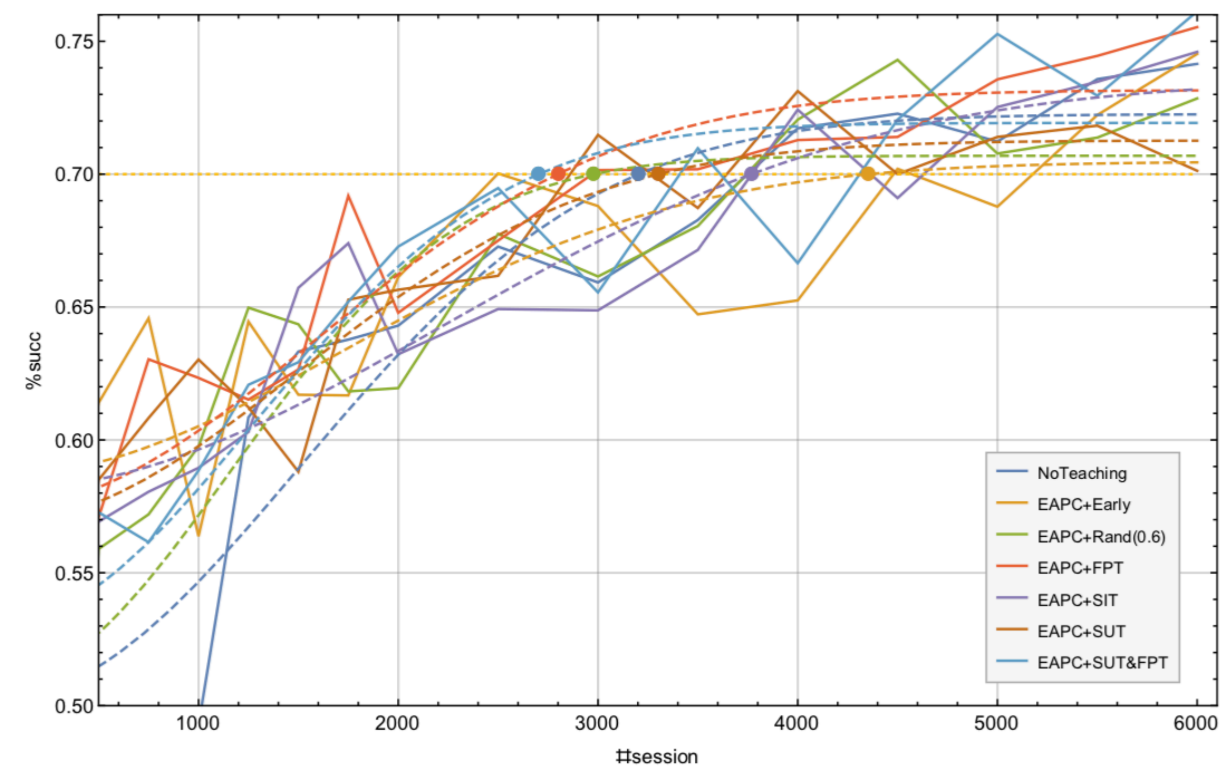
	CA	EA	EAPC
Early	**98.5**	110.6	56.1
Rand	193.4	102.4	65.5
FPT	154.4	**86.2**	**53.6**
SIT	230.8	121.7	66.0
SUT	183.5	**95.8**	**44.5***
SUT&FPT	**131.6**	101.8	54.6
NoTeaching	202.9		

	CA	EA	EAPC
Early	3390.9	3479.4	4354.7
Rand	3669.0	3518.5	2979.2
FPT	**3089.4**	**2921.1**	**2798.4**
SIT	3576.4	4339.7	3768.7
SUT	3230.4	**2954.5**	3300.2
SUT&FPT	**2890.7**	3393.0	**2702.2***
NoTeaching	3204.1		

2. A Complete Companion Teaching Framework

How to evaluate the safety and the efficiency?





Risk Index: $\mathtt{RI} = \int_{t=0}^{T} \mathtt{dis}(t)\delta_{\mathtt{risk}}(t)dt,$

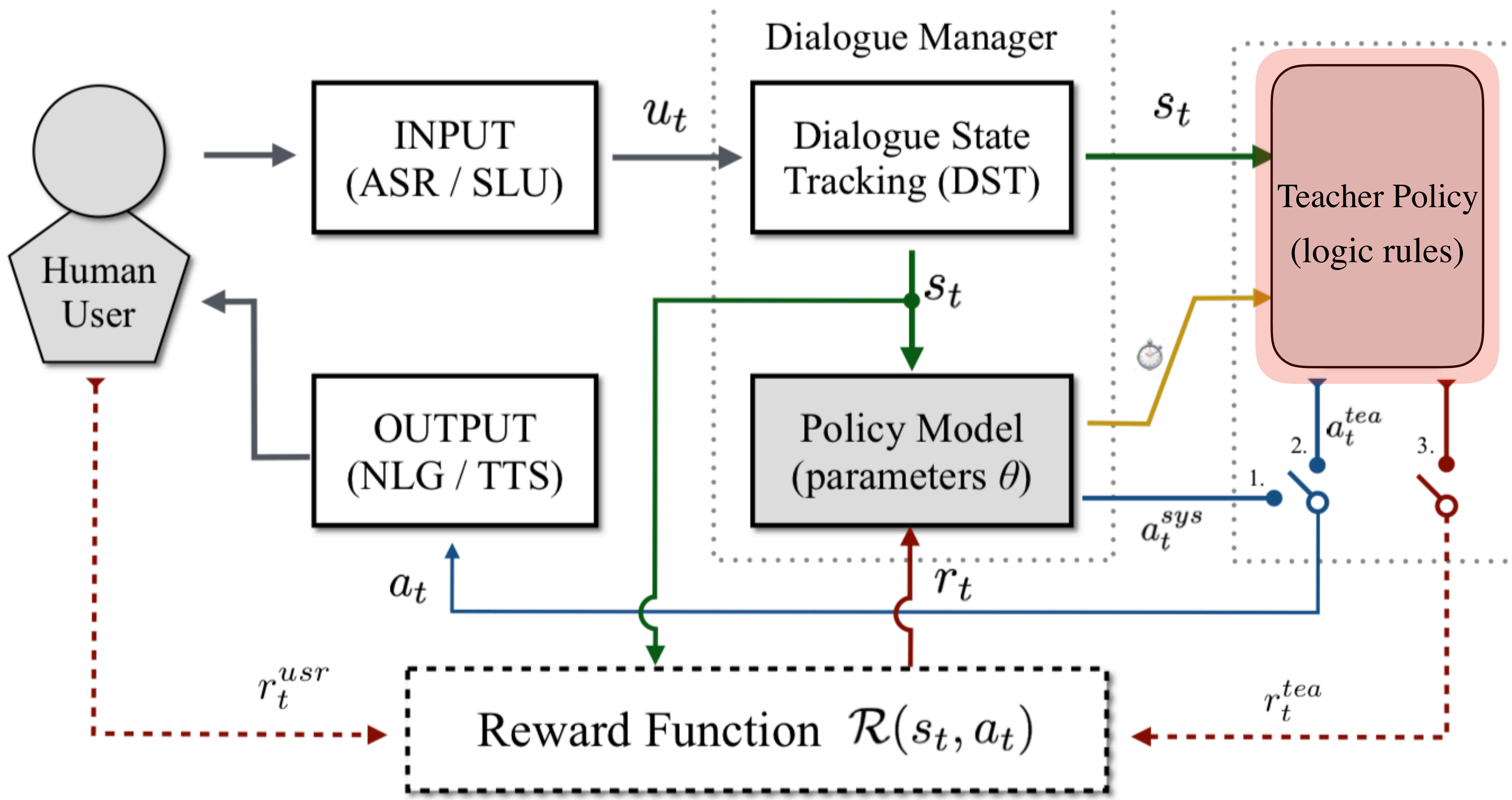
Hitting Time: $\mathtt{HT} = c\sqrt{\ln\left(\frac{b}{a-\tau}\right)}.$

	CA	EA	EAPC
Early	<u>**98.5**</u>	110.6	56.1
Rand	193.4	102.4	65.5
FPT	<u>154.4</u>	<u>**86.2**</u>	**53.6**
SIT	230.8	121.7	66.0
SUT	183.5	**95.8**	<u>**44.5**</u>*
SUT&FPT	**131.6**	<u>101.8</u>	<u>54.6</u>
NoTeaching	202.9		

	CA	EA	EAPC
Early	3390.9	3479.4	4354.7
Rand	3669.0	3518.5	2979.2
FPT	**3089.4**	<u>**2921.1**</u>	**2798.4**
SIT	3576.4	4339.7	3768.7
SUT	3230.4	**2954.5**	3300.2
SUT&FPT	<u>**2890.7**</u>	3393.0	<u>**2702.2**</u>*
NoTeaching	3204.1		

3. Replacing Human with Rule-Based Systems

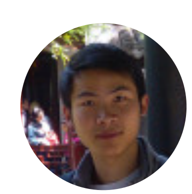
Replace human with rule-based systems



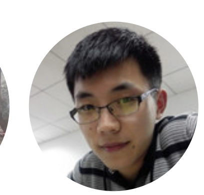
Agent-Aware Dropout DQN for Safe and Efficient On-line Dialogue Policy Learning

Lu Chen, Xiang Zhou, Cheng Chang, Runzhe Yang, Kai Yu. **EMNLP 2017**

http://www.aclweb.org/anthology/D/D17/D17-1260.pdf


Lu Chen


Xiang Zhou


Cheng Chang


Runzhe Yang


Prof. Kai Yu

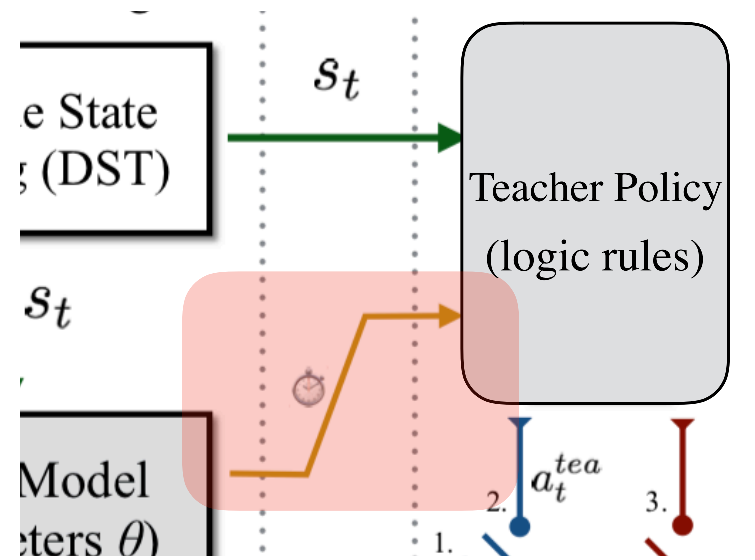
3. Replacing Human with Rule-Based Systems

Agent-Aware Dropout DQN

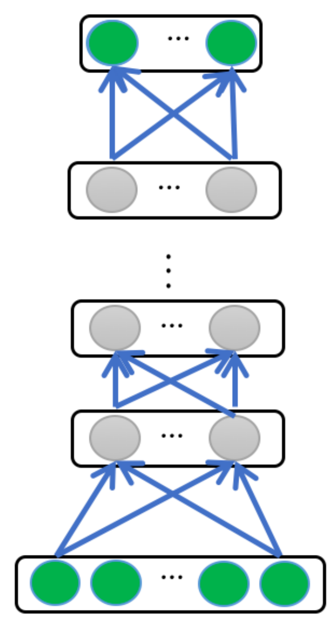
N stochastic forward passes

for $i = 1, N$ **do**
    $\mathbf{q}_i \leftarrow \text{DropoutQNetwork}(\mathbf{b}_t)$
    $a_{ti} \leftarrow \arg\max_j q_{ij}$
    $\mathbf{p}[a_{ti}] \leftarrow \mathbf{p}[a_{ti}] + 1/N$
end for
$c_t \leftarrow \max_j p_j$
$a_t^{stu} \leftarrow \arg\max_j p_j$

c_t uncertainty



$P_{tea}(\Delta C_e)$ where $\Delta C_e = \max(0, C_{th} - \overline{C}_e)$



$\{b_t, b_t, b_t, b_t, b_t, b_t, b_t, b_t\}$

$\{2, 1, 3, 2, 4, 1, 2, 2, 3\}$ $\begin{cases} a_t = 2 \\ c_t = \frac{4}{8} \end{cases}$

3. Replacing Human with Rule-Based Systems

Agent-Aware Dropout DQN

N stochastic forward passes

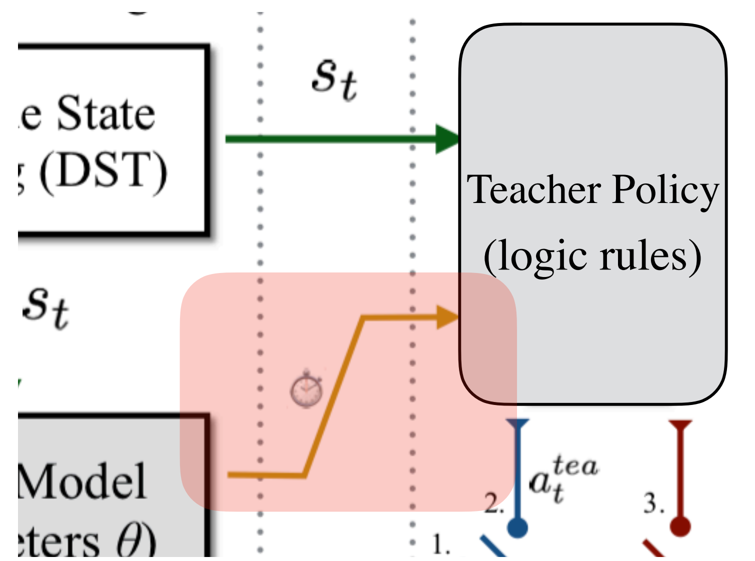
for $i = 1, N$ **do**
  $\mathbf{q}_i \leftarrow \text{DropoutQNetwork}(\mathbf{b}_t)$
  $a_{ti} \leftarrow \arg\max_j q_{ij}$
  $\mathbf{p}[a_{ti}] \leftarrow \mathbf{p}[a_{ti}] + 1/N$
end for
$c_t \leftarrow \max_j p_j$
$a_t^{stu} \leftarrow \arg\max_j p_j$

c_t uncertainty

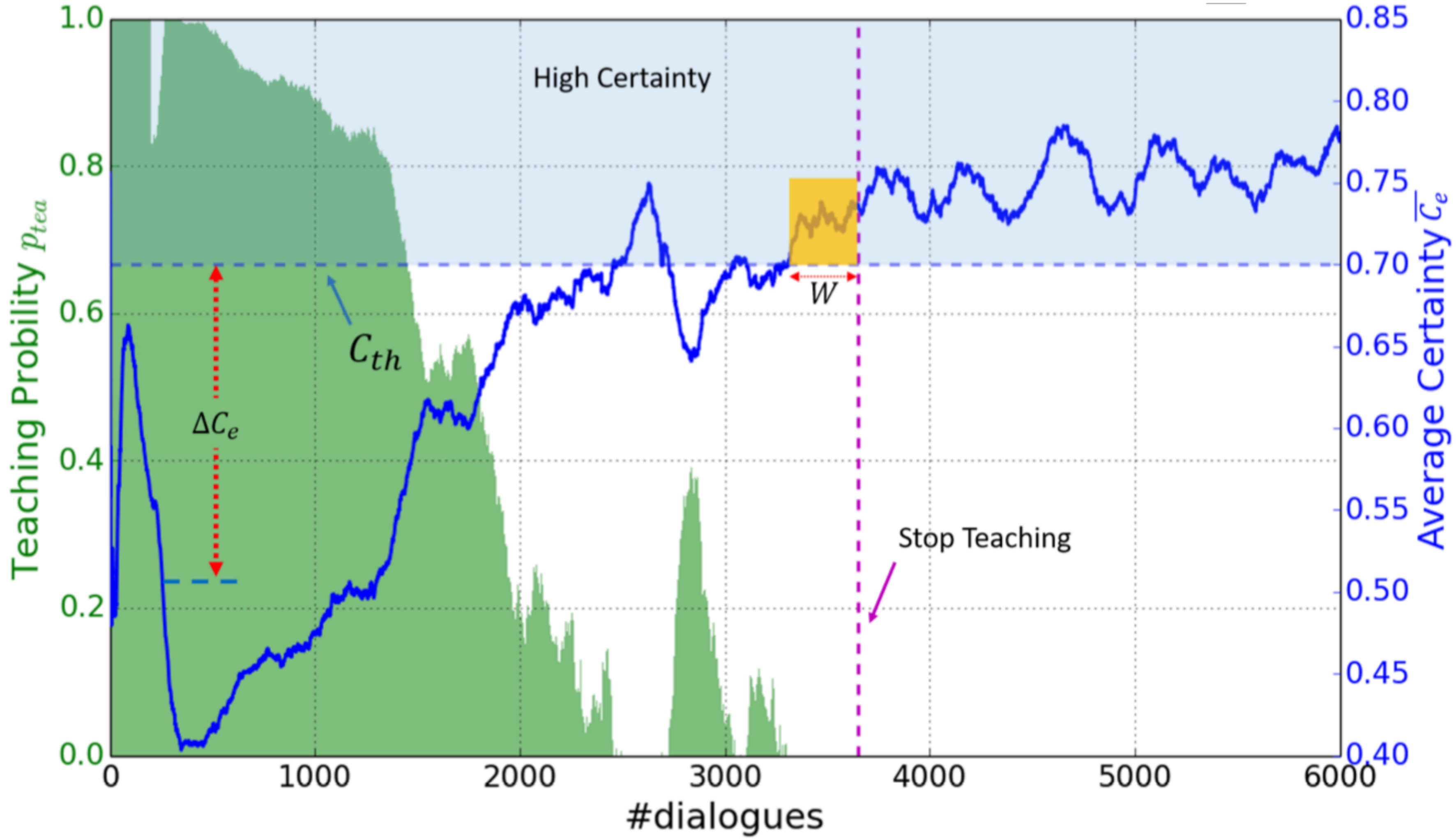
$$\overline{C}_e = \frac{1}{W}\sum_{i=e-W}^{e-1} C_i$$

average uncertainty

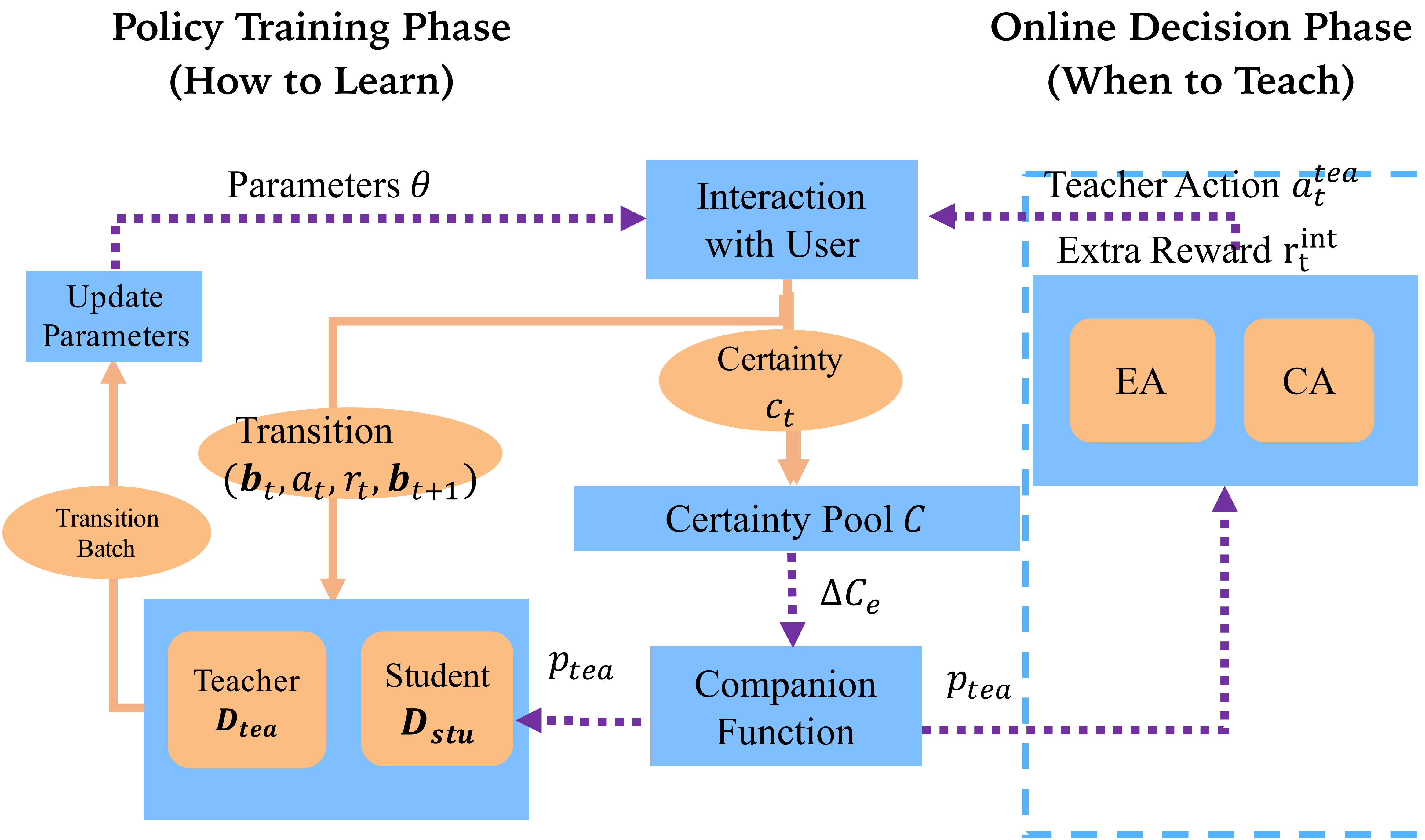
Teach when uncertain



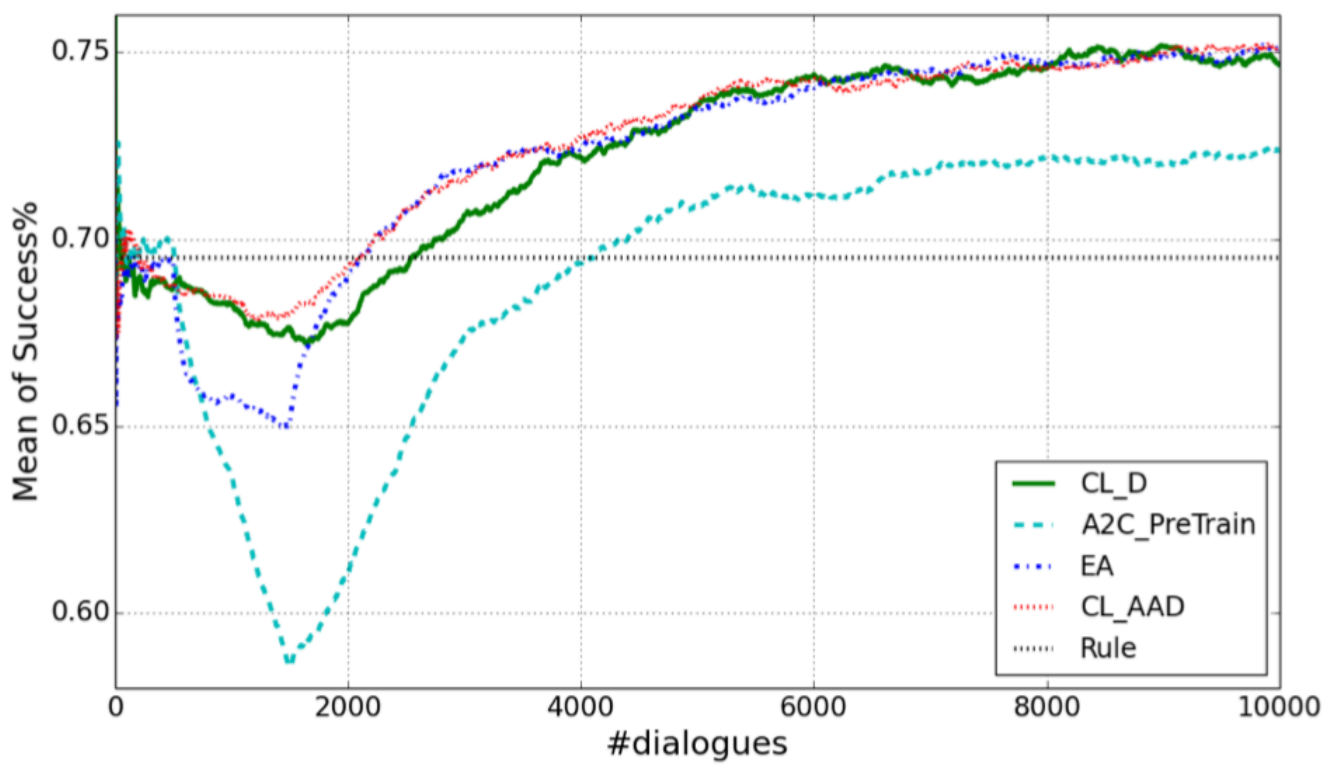
$P_{tea}(\Delta C_e)$ where $\Delta C_e = \max(0, C_{th} - \overline{C}_e)$

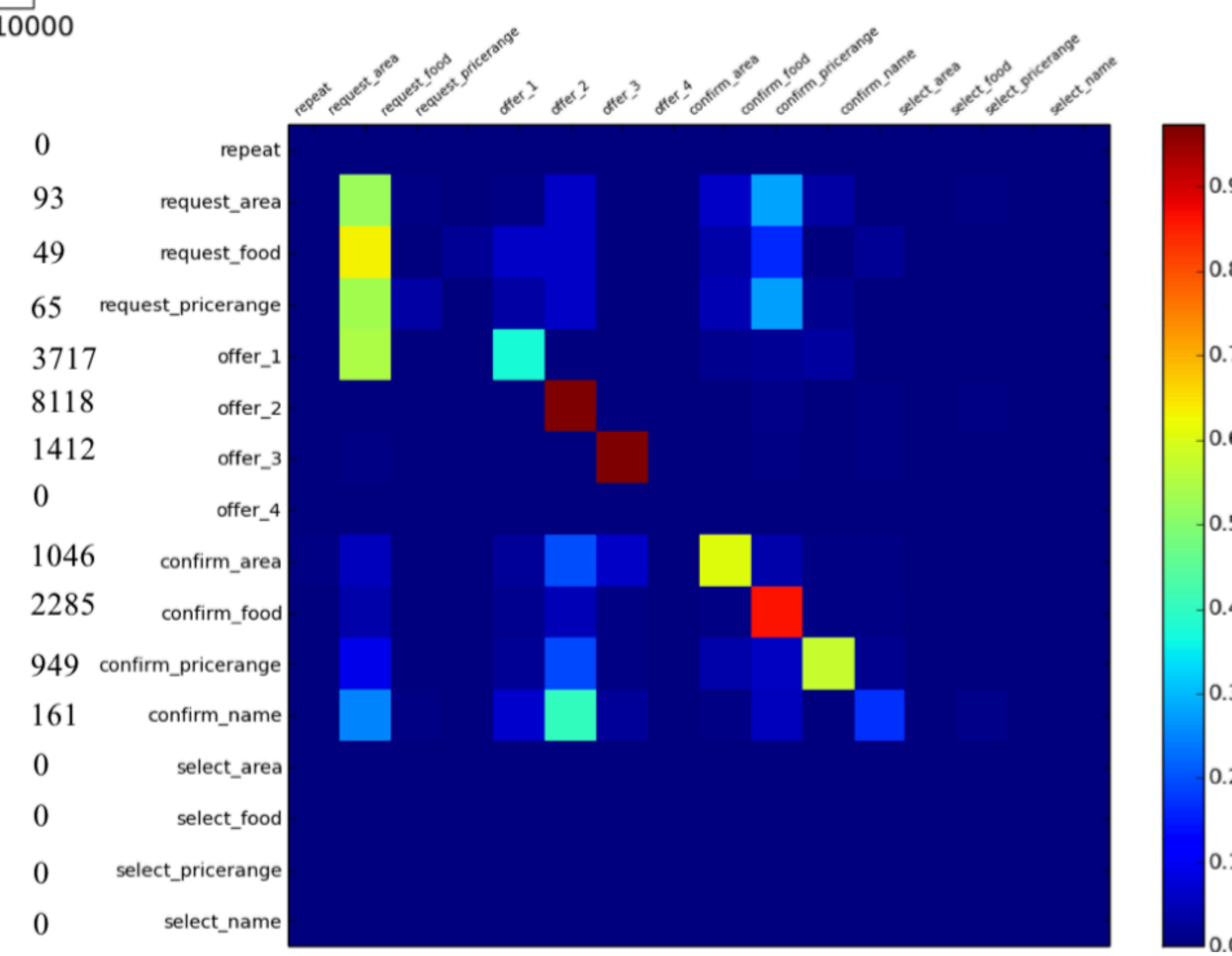


3. Replacing Human with Rule-Based Systems

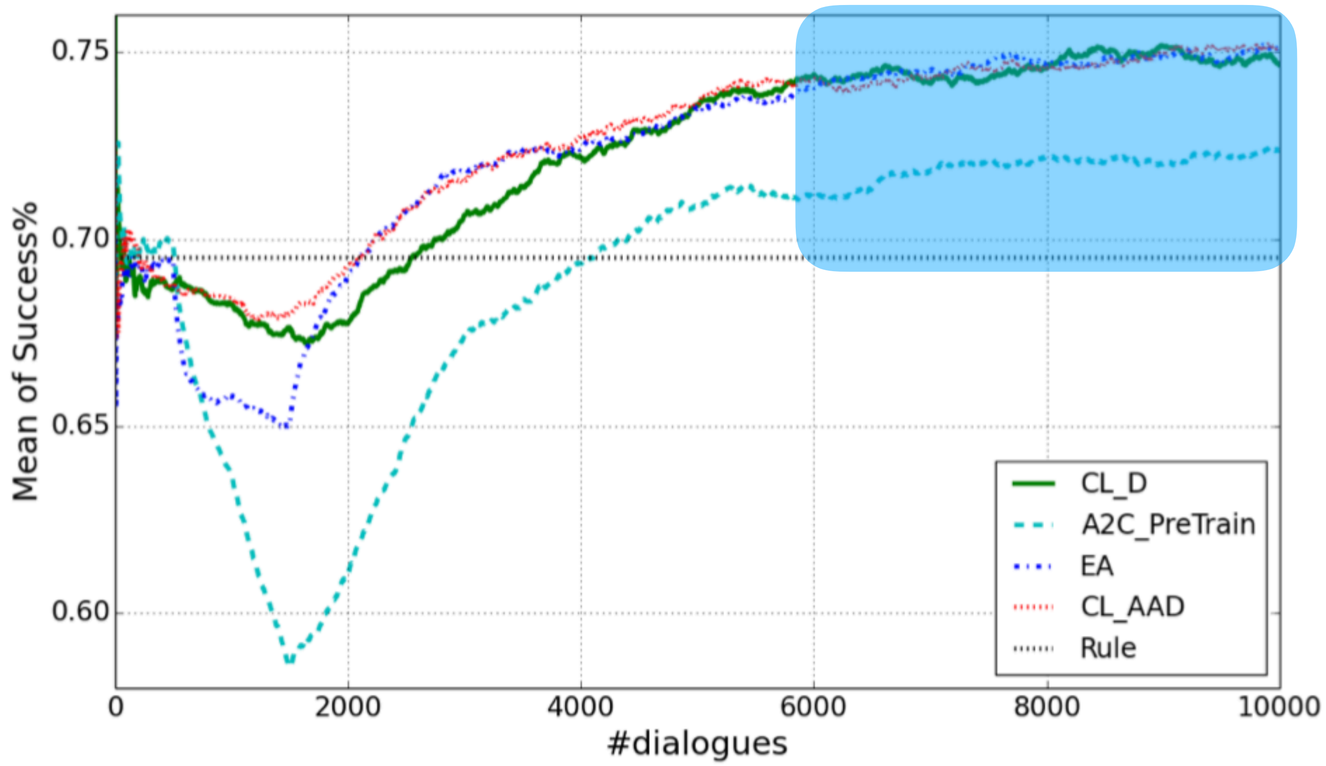


3. Replacing Human with Rule-Based Systems

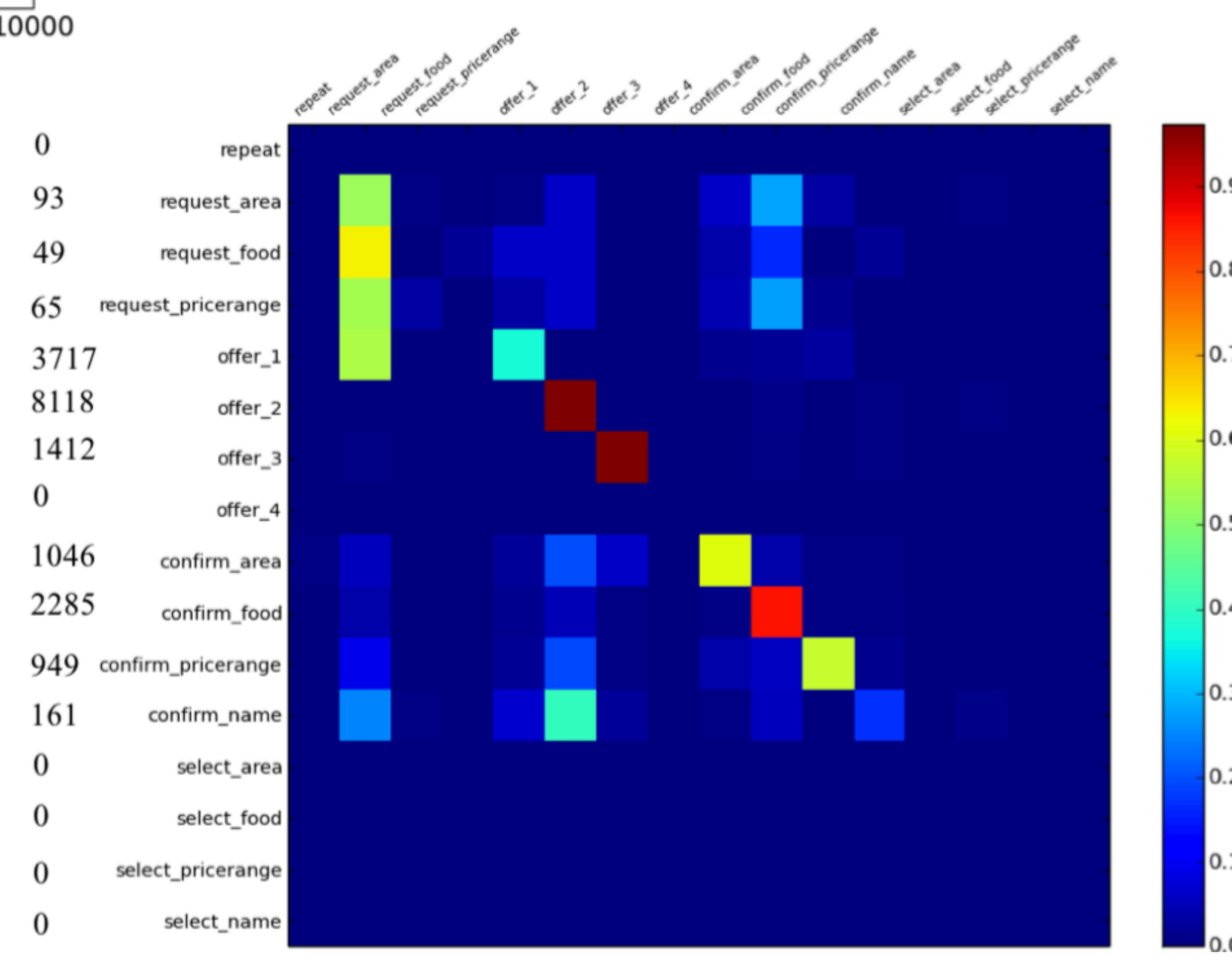




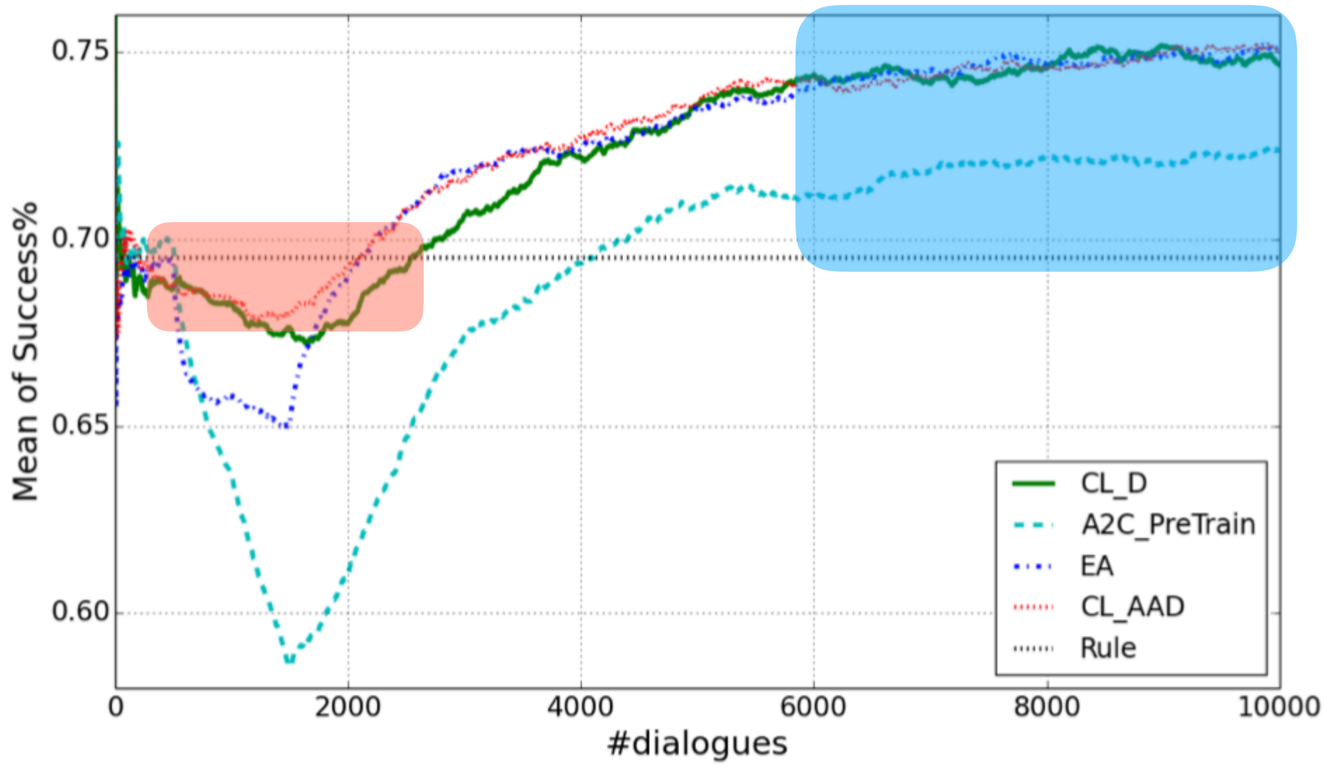
3. Replacing Human with Rule-Based Systems



Surpass Rule Policy in Accuracy

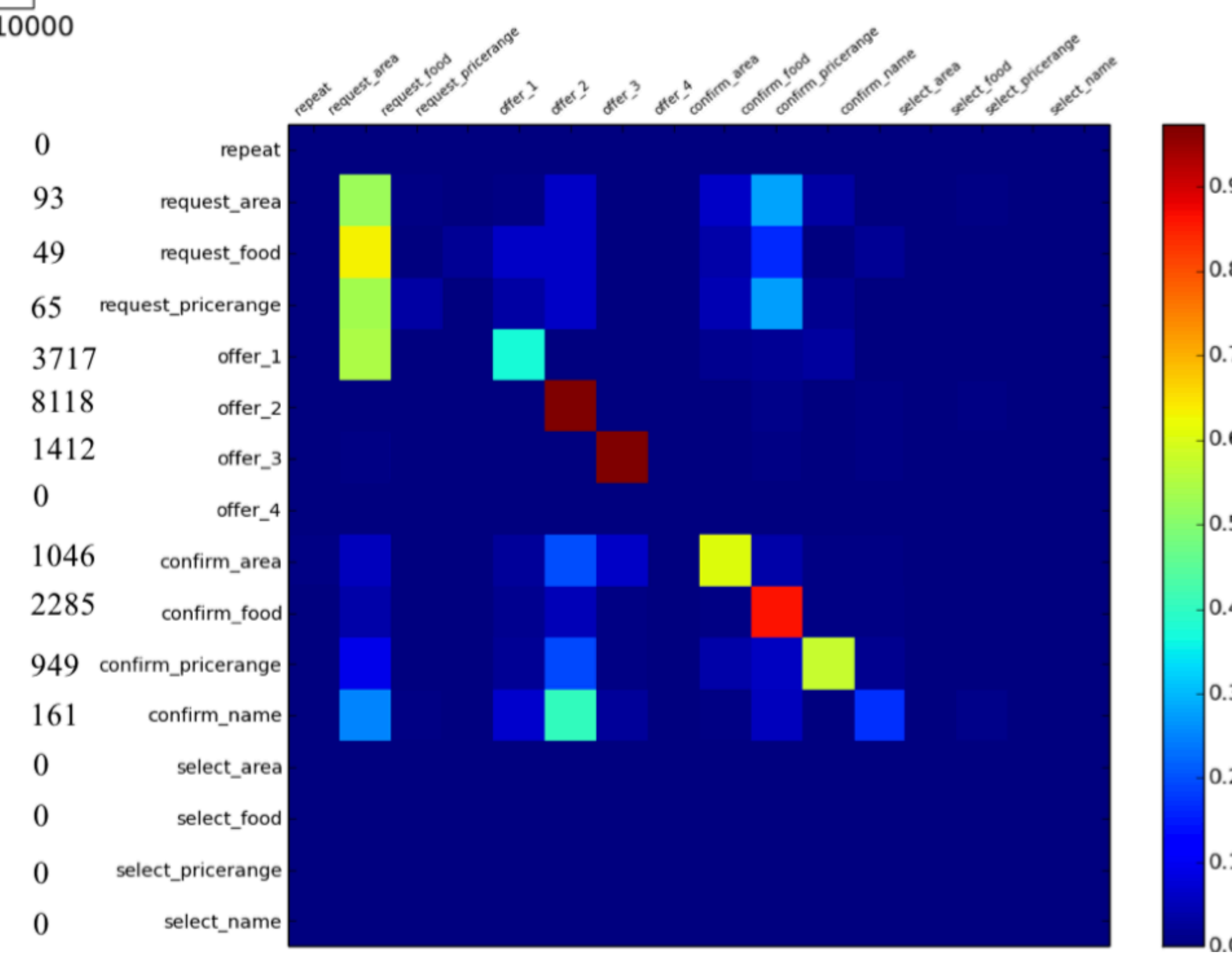


3. Replacing Human with Rule-Based Systems

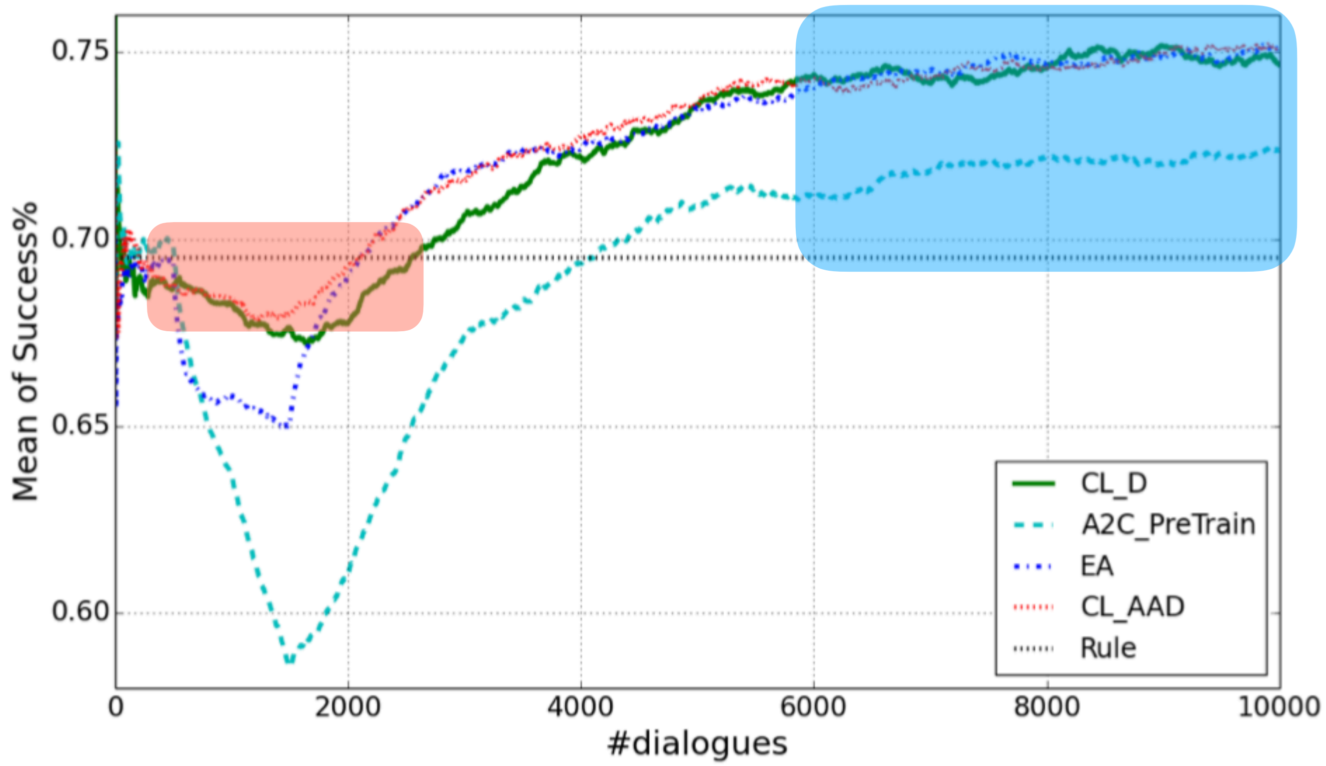


Surpass Rule Policy in Accuracy

AAD-DQN with uncertainty based heuristic provides the safer learning process.



3. Replacing Human with Rule-Based Systems

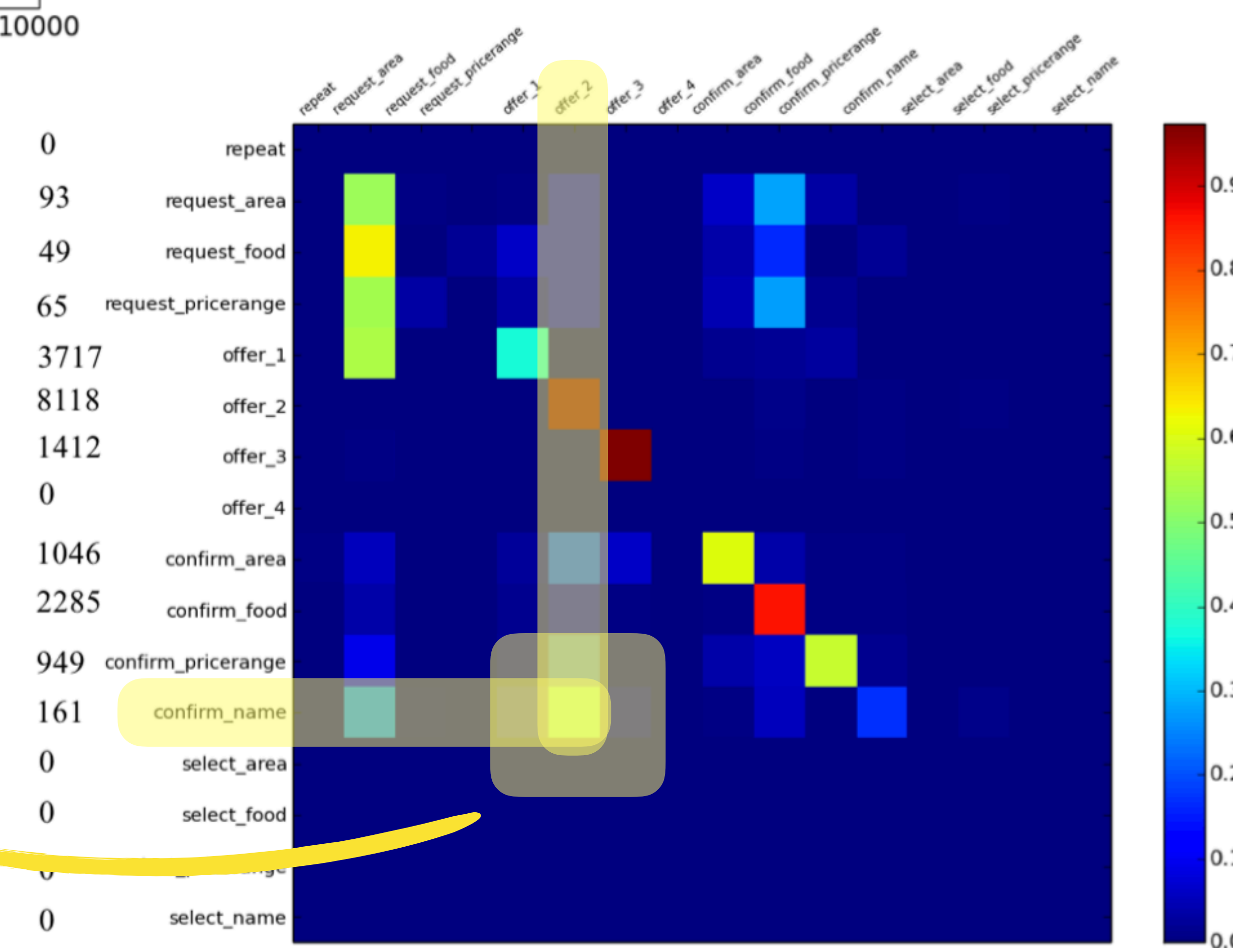


Surpass Rule Policy in Accuracy

AAD-DQN with uncertainty based heuristic provides the safer learning process.

Better policies are found by AAD-DQN:

New policy can offer the information (nn policy: offer_2) **while the rule based policy needs to confirm.** (rule: confirm_name / confirm_area)



Summary

	User Simulator	Real (Recruited) User	Real User + Human Teacher	Real User + Human Rules
Pros:	Low cost, easy to tune			
Cons:	Training env. may be different with the real env.			

Summary

	User Simulator	Real (Recruited) User	Real User + Human Teacher	Real User + Human Rules
Pros:	Low cost, easy to tune	Training env. is close to the real application scenario		
Cons:	Training env. may be different with the real env.	**Cold Start Problem**		

Summary

	User Simulator	Real (Recruited) User	Real User + Human Teacher (Companion Teaching)	Real User + Human Rules
Pros:	Low cost, easy to tune	Training env. is close to the real application scenario	Safety, efficiency	
Cons:	Training env. may be different with the real env.	**Cold Start Problem**	Expensive, teachers are not 24-7 available	

Summary

	User Simulator	Real (Recruited) User	Real User + Human Teacher	Real User + Human Rules (Companion Learning)
Pros:	Low cost, easy to tune	Training env. is close to the real application scenario	Safety, efficiency	Safety, efficiency, economic
Cons:	Training env. may be different with the real env.	**Cold Start Problem**	Expensive, teachers are not 24-7 available	Cost of hand-crafting rules

Thank you!