



YUN-NUNG (VIVIAN) CHEN  
國立臺灣大學  
National Taiwan University

ASLI CELIKYILMAZ  
Microsoft Research

DILEK HAKKANI-TÜR  
Google

# Part I

## Introduction & Background

# Brief History of Dialogue Systems

3

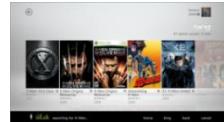
## Multi-modal systems

e.g., Microsoft MiPad, Pocket PC



## TV Voice Search

e.g., Bing on Xbox



## Task-specific argument extraction

(e.g., Nuance, SpeechWorks)

User: "I want to fly from Boston to New York next week."



Early 1990s

## Keyword Spotting

(e.g., AT&T)

System: "Please say collect, calling card, person, third number, or operator"

Early 2000s

## Intent Determination

(Nuance's Emily™, AT&T HMIHY)

User: "Uh...we want to move...we want to change our phone line from this house to another house"



DARPA  
CALO Project

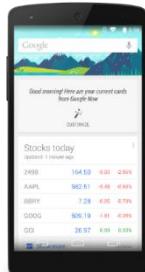
2017

## Virtual Personal Assistants



# Language Empowering Intelligent Assistant

4



Apple Siri (2011)

Google Now (2012)  
Google Assistant (2016)

Microsoft Cortana (2014)



Amazon Alexa/Echo (2014)



Facebook M & Bot (2015)



Google Home (2016)

# Challenges

5

- Variability in natural language
- Robustness
- Recall/Precision Trade-off
- Meaning Representation
- Common Sense, World Knowledge
- Ability to learn
- Transparency

# Dialogue Systems

6

## Task-Oriented

- Personal assistant, helps users achieve a certain task
- Combination of rules and statistical components
- Examples:
  - POMDP for spoken dialog systems (Williams and Young, 2007)
  - End-to-end trainable task-oriented dialogue system (Wen et al., 2016)
  - End-to-end reinforcement learning dialogue system (Zhao and Eskenazi, 2016)

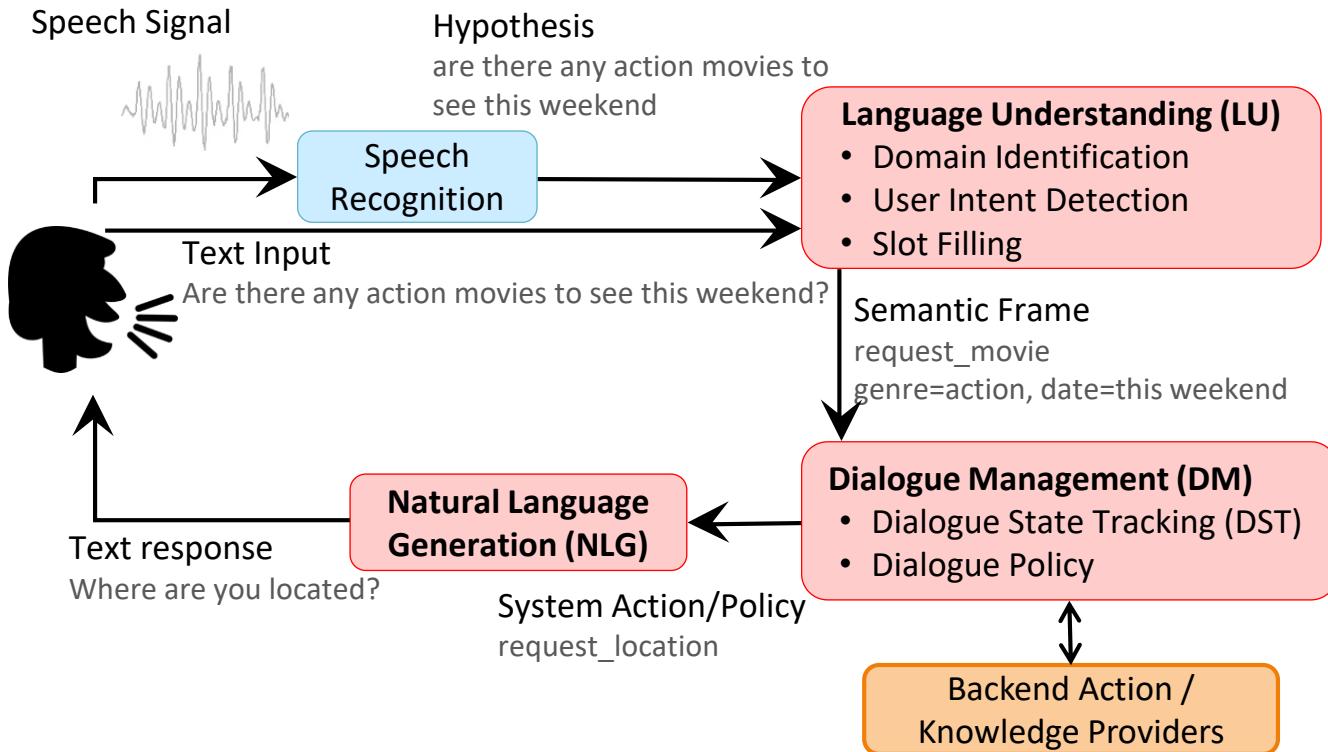
## Chit-Chat

- No specific goal, focus on natural responses
- Using variants of seq2seq model
- Examples:
  - A neural conversation model (Vinyals and Le, 2015)
  - Reinforcement learning for dialogue generation (Li et al., 2016)
  - Conversational contextual cues for response ranking (AI-Rfou et al., 2016)

# Task-Oriented Dialogue System (Young, 2000)

7

<http://rsta.royalsocietypublishing.org/content/358/1769/1389.short>



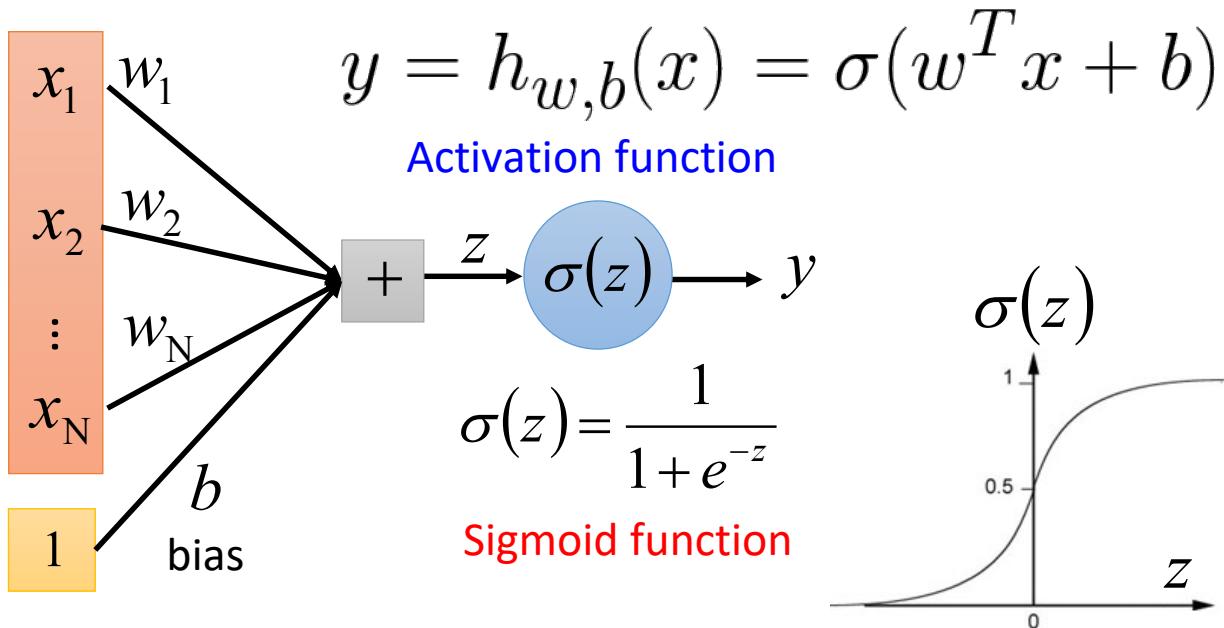
# Outline

8

- Introduction & Background
  - **Neural Networks**
  - Reinforcement Learning
- Deep Learning Based Dialogue System
  - Spoken/Natural Language Understanding (SLU/NLU)
  - Dialogue State Tracking (DST)
  - Dialogue Policy
  - Natural Language Generation (NLG)
  - End-to-End Learning for Dialogue Systems
- Evaluation
- Recent Trends on Learning Dialogues
- Challenges
- Conclusion

# A Single Neuron

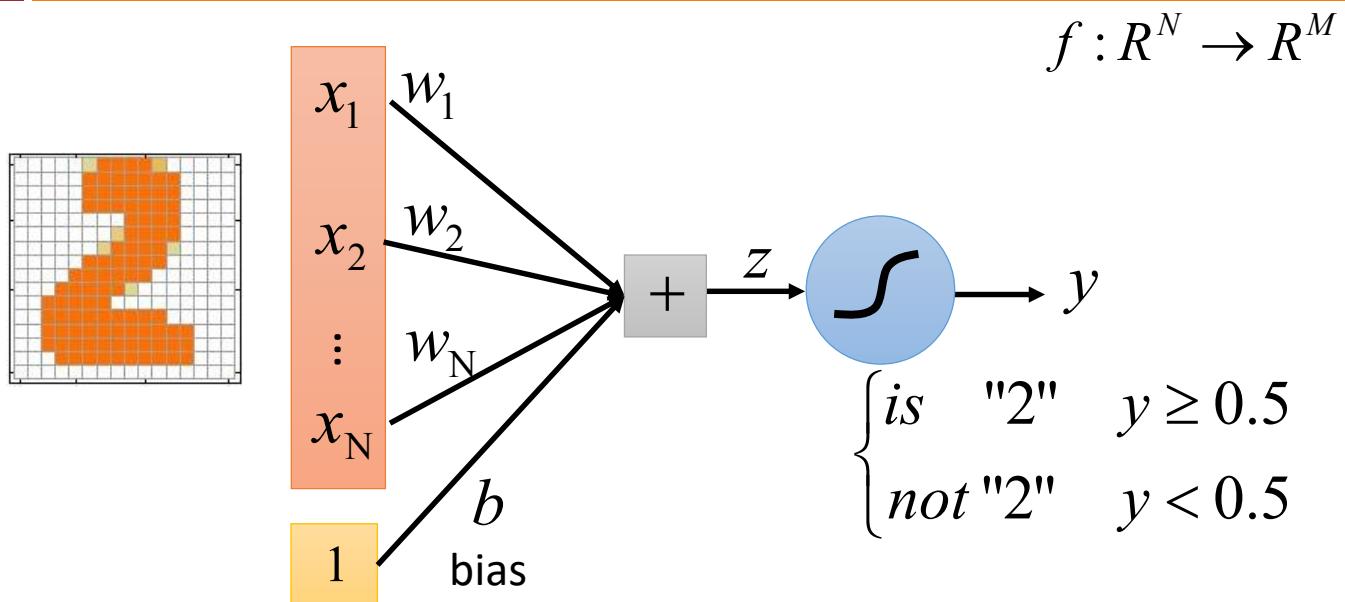
9



$w, b$  are the parameters of this neuron

# A Single Neuron

10

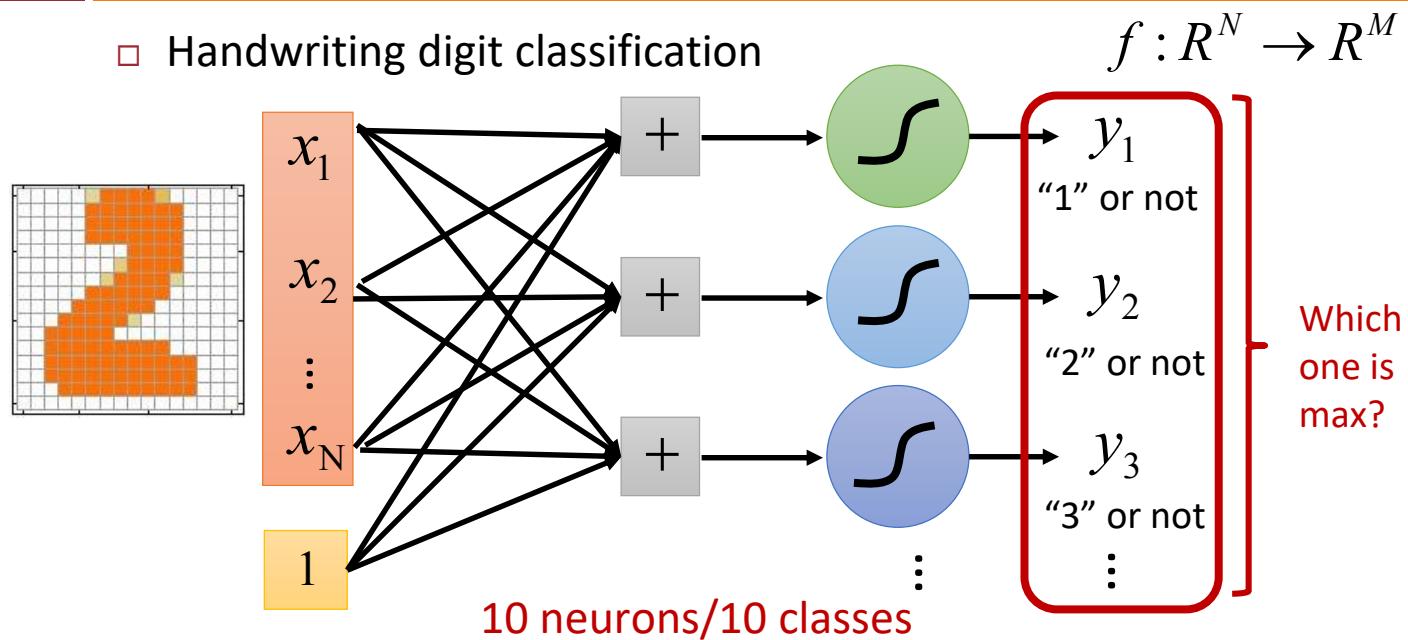


A single neuron can only handle binary classification

# A Layer of Neurons

11

- Handwriting digit classification



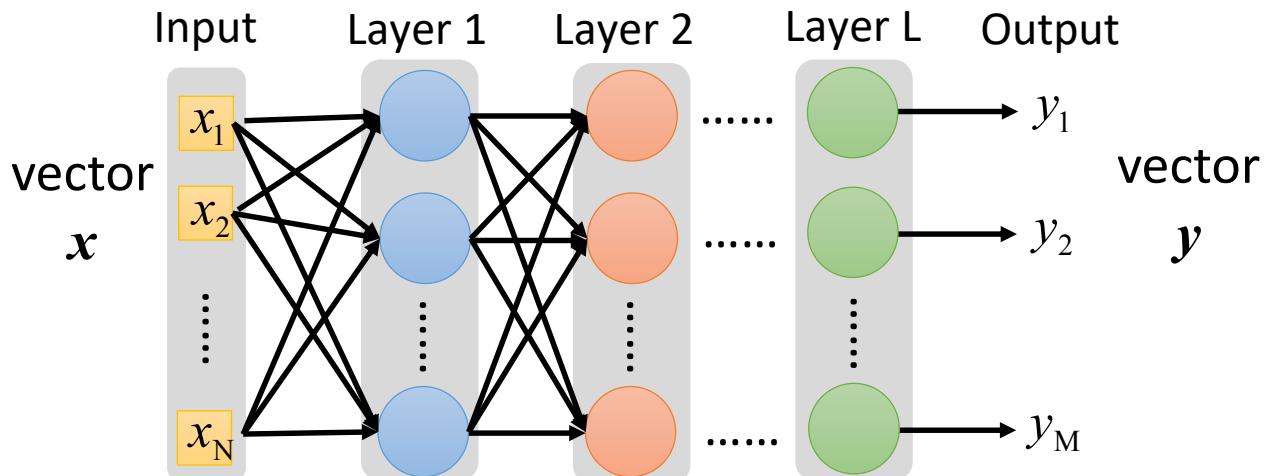
A layer of neurons can handle multiple possible output, and the result depends on the max one

# Deep Neural Networks (DNN)

12

- Fully connected feedforward network

$$f : R^N \rightarrow R^M$$



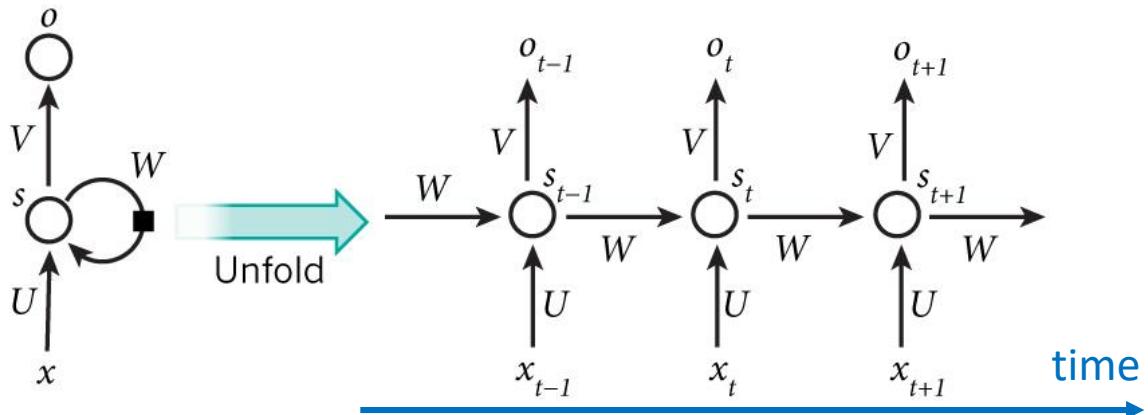
Deep NN: multiple hidden layers

# Recurrent Neural Network (RNN)

13

$$s_t = \sigma(W s_{t-1} + U x_t) \quad \sigma(\cdot): \text{tanh, ReLU}$$

$$o_t = \text{softmax}(V s_t)$$

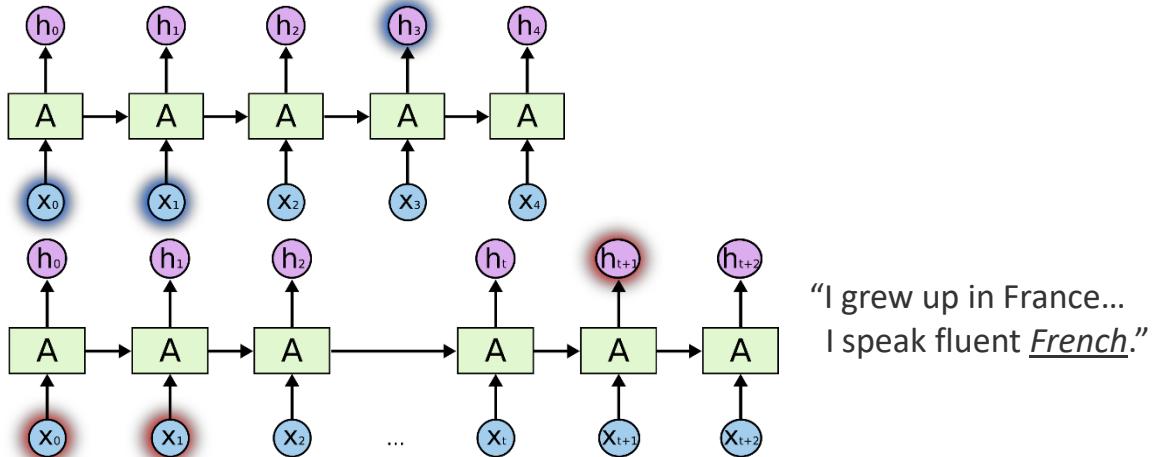


RNN can learn accumulated sequential information (time-series)

# Vanishing Gradient: Gating Mechanism

14

- RNN: keeps temporal sequence information



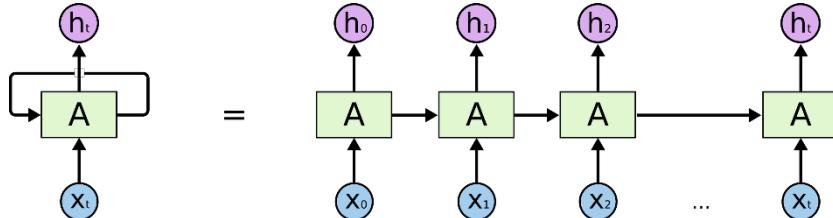
Issue: in theory, RNNs can handle “long-term” info , but cannot in practice

→ use gates to directly encode the long-distance information

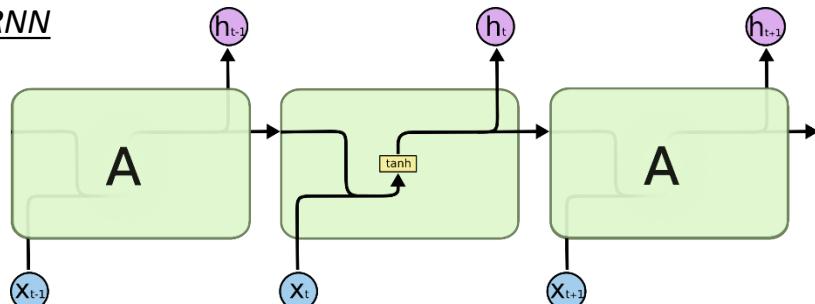
# Long Short-Term Memory (LSTM)

15

- LSTMs are explicitly designed to avoid the long-term dependency problem

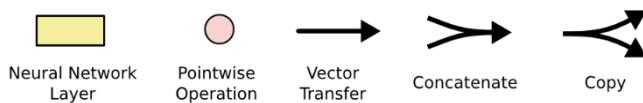


Vanilla RNN

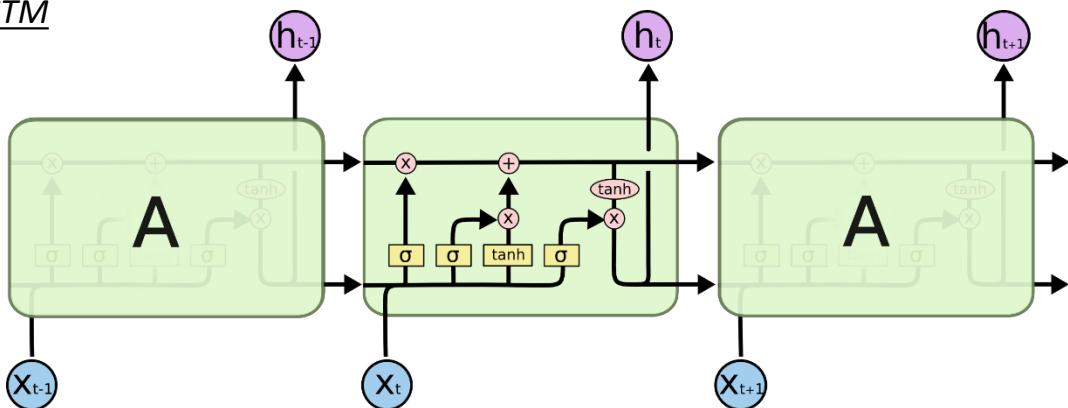


# Long Short-Term Memory (LSTM)

16

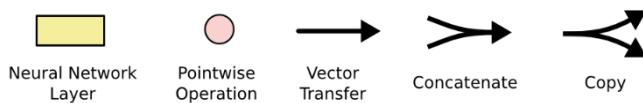


LSTM

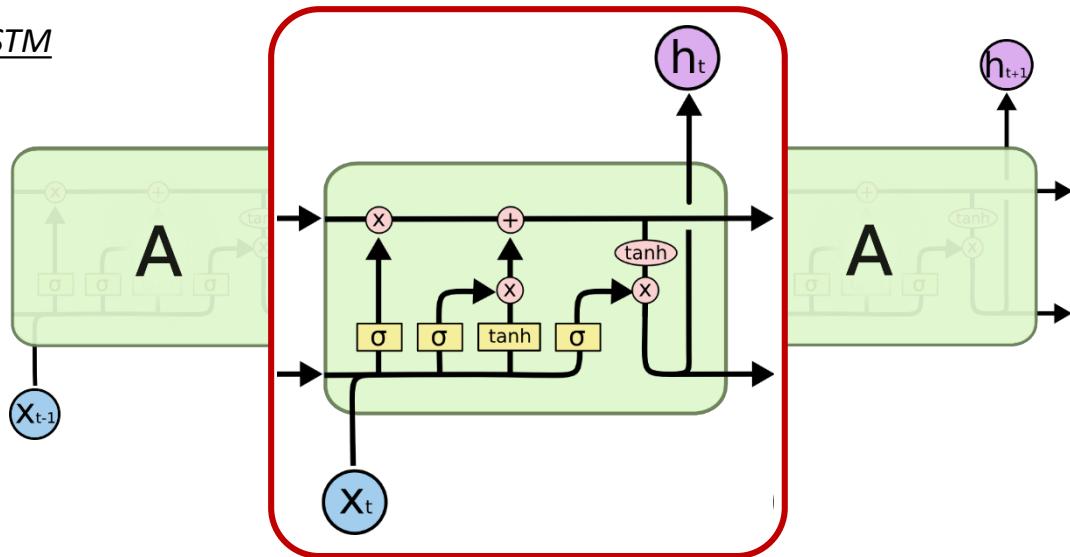


# Long Short-Term Memory (LSTM)

17

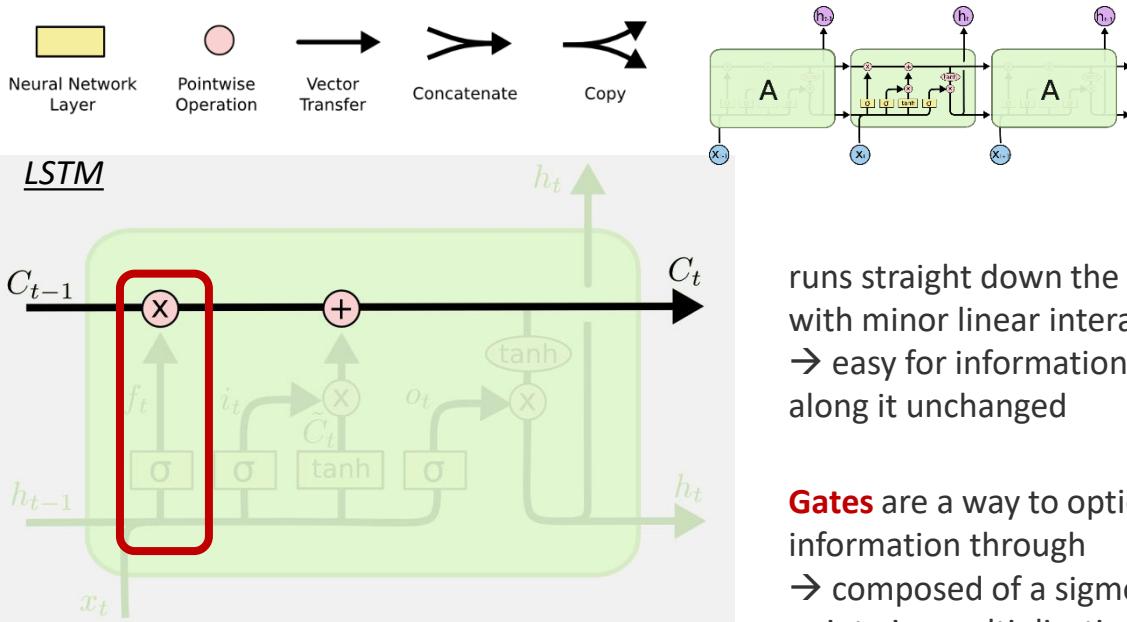


LSTM



# Long Short-Term Memory (LSTM)

18

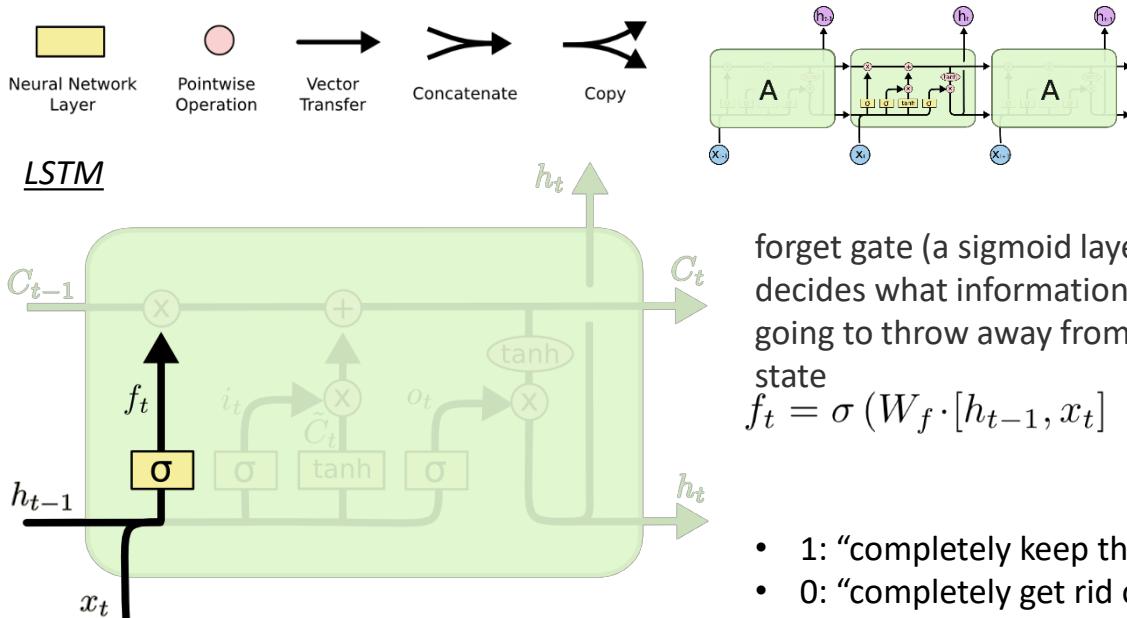


runs straight down the chain  
with minor linear interactions  
→ easy for information to flow  
along it unchanged

**Gates** are a way to optionally let information through  
→ composed of a sigmoid and a pointwise multiplication operation

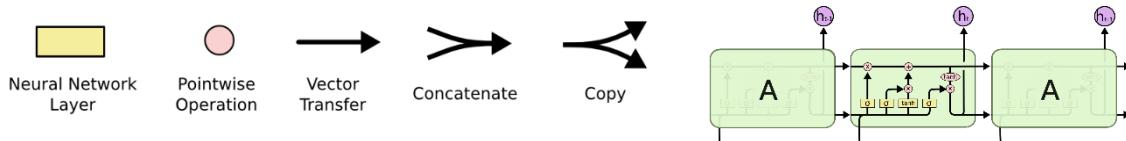
# Long Short-Term Memory (LSTM)

19

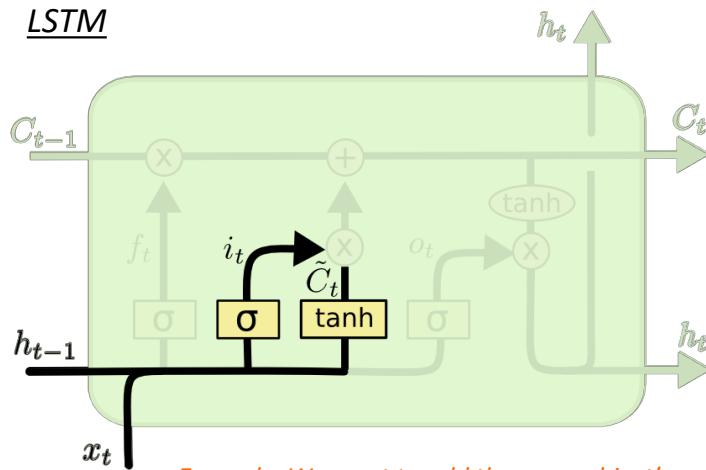


# Long Short-Term Memory (LSTM)

20



LSTM



input gate (a sigmoid layer): decides what new information we're going to store in the cell state

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

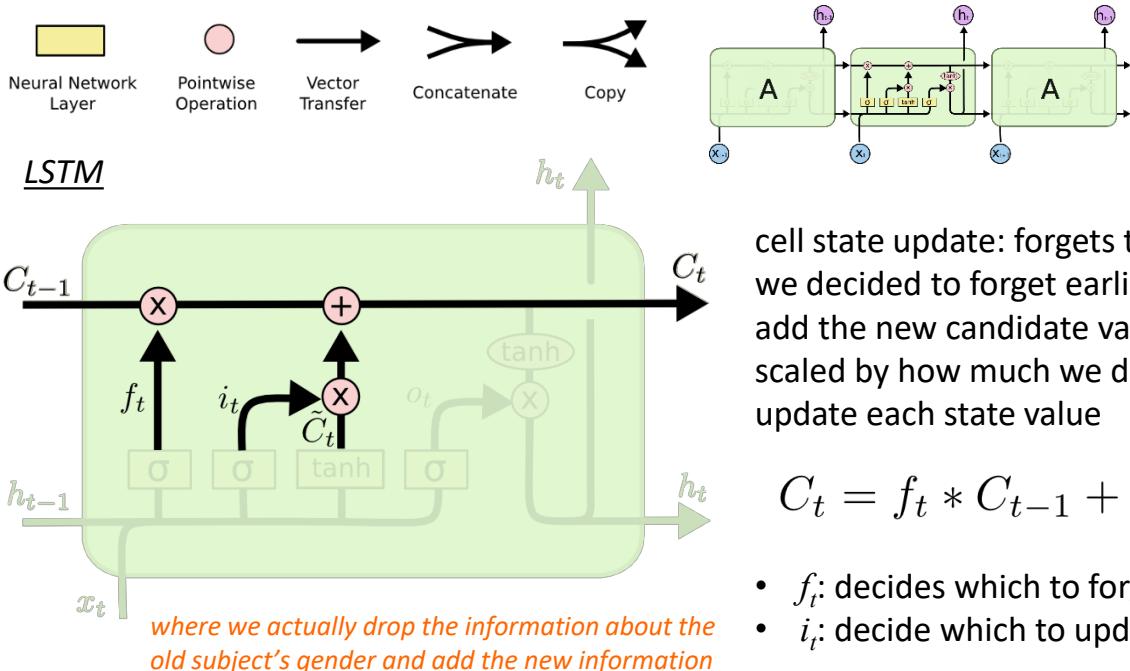
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

Vanilla RNN

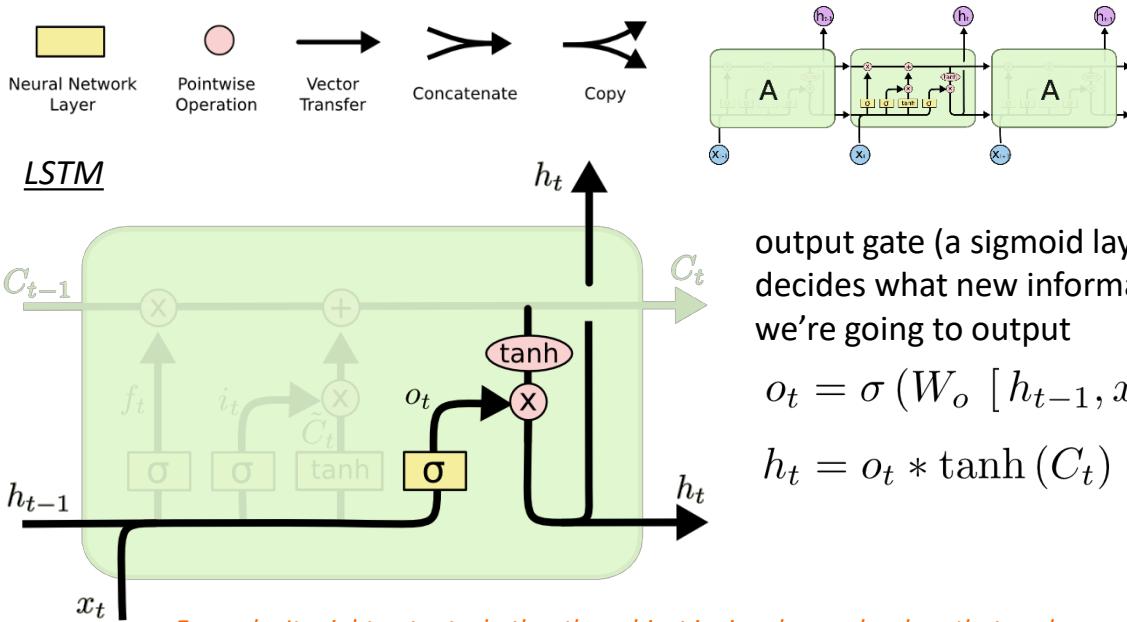
Example: We want to add the new subject's gender to the cell state for replacing the old one.

# Long Short-Term Memory (LSTM)

21



# Long Short-Term Memory (LSTM)



output gate (a sigmoid layer):  
decides what new information  
we're going to output

$$o_t = \sigma(W_o [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

*Example: It might output whether the subject is singular or plural, so that we know what form a verb should be conjugated into if that's what follows next.*

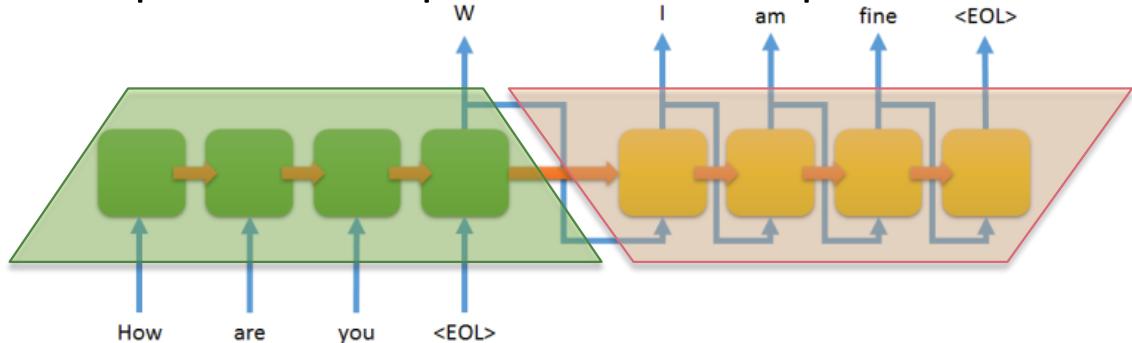
addressing gradient vanishing issues in RNN

# Seq2Seq Model (Sutskever et al., 2014)

23

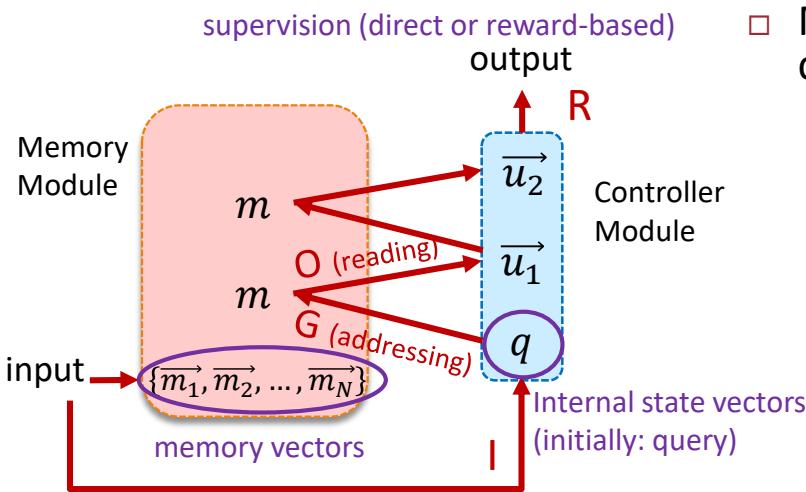
<http://papers.nips.cc/paper/5346-information-based-learning-by-agents-in-unbounded-state-spaces.pdf>

- Encode source into a fixed length vector, use it as initial recurrent state for target decoder model
- Cascade two RNNs, “encoder-decoder model”
  - Input: word sequences in the question
  - Output: word sequences in the response



The input and output should be model in a sequential way

# Memory Networks (Weston et al., 2014)



- Memory networks have 4 components:
  - **I:** (input feature map) convert incoming data to the internal feature representation
  - **G:** (generalization) update memories given new input
  - **O:** produce new output (in feature representation space) given the memories
  - **R:** (response) convert output O into a response seen by the outside world

Story (16: basic induction)	Support	Hop 1	Hop 2	Hop 3
Brian is a frog.	yes	0.00	0.98	0.00
Lily is gray.		0.07	0.00	0.00
Brian is yellow.	yes	0.07	0.00	1.00
Julius is green.		0.06	0.00	0.00
Greg is a frog.	yes	0.76	0.02	0.00
What color is Greg? Answer: yellow		Prediction: yellow		

Memory module stores the history to make the model find the supporting facts

# Outline

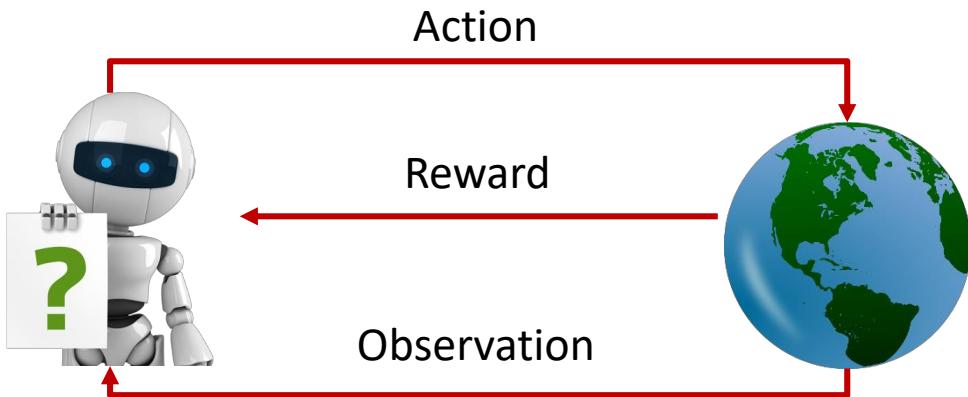
25

- Introduction & Background
  - Neural Networks
  - **Reinforcement Learning**
- Deep Learning Based Dialogue System
  - Spoken/Natural Language Understanding (SLU/NLU)
  - Dialogue State Tracking (DST)
  - Dialogue Policy
  - Natural Language Generation (NLG)
  - End-to-End Learning for Dialogue Systems
- Evaluation
- Recent Trends on Learning Dialogues
- Challenges
- Conclusion

# Reinforcement Learning

26

- RL is a general purpose framework for **decision making**
  - RL is for an *agent* with the capacity to *act*
  - Each *action* influences the agent's future *state*
  - Success is measured by a scalar *reward* signal
  - Goal: *select actions to maximize future reward*



# Reinforcing Learning

27

- Markov decision process (MDP)
  - $S$ : State set
  - $A$ : Action set
  - $R: S \rightarrow \mathbb{R}$  (Reward)
  - $P_{sa}$ : transition probabilities ( $p(s,a,s') \in R$ )
  - $\gamma$ : discount factor
- $MDP = (S, A, R, P_{sa}, \gamma)$ 
  - AlphaGo improves by self-playing
  - Car autonomously learns driving up!



# Reinforcing Learning

28

- Start from state  $s_0$
- Choose action  $a_0$
- Transit to  $s_1 \sim P(s_0, a_0)$

- Continue...

$$s_0 \xrightarrow{a_0} s_1 \xrightarrow{a_1} s_2 \xrightarrow{a_2} s_3 \xrightarrow{a_3} \dots$$

- Total reward:  $R(s_0) + \gamma R(s_1) + \gamma^2 R(s_2) + \dots$

**Goal:** select actions that maximize the expected total reward

$$\mathbb{E}[R(s_0) + \gamma R(s_1) + \gamma^2 R(s_2) + \dots]$$

# Reinforcement Learning Approach

29

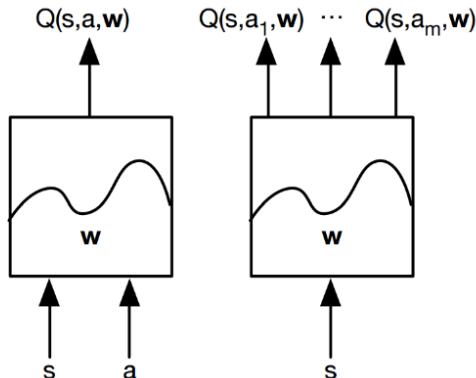
- Policy-based RL
  - Search directly for optimal policy  $\pi^*$ 
    - $\pi^*$  is the policy achieving maximum future reward
- Value-based RL
  - Estimate the optimal value function  $Q^*(s, a)$ 
    - $Q^*(s, a)$  is maximum value achievable under any policy
- Model-based RL
  - Build a model of the environment
  - Plan (e.g. by lookahead) using model

# Q-Networks (Sutton et al., 1998)

- Q-networks represent value functions with weights  $w$

$$Q(s, a, w) \approx Q^*(s, a)$$

- generalize from seen states to unseen states (#states is large)
- update parameter  $W$  for function approximation



# Q-Learning

31

- Goal: estimate optimal Q-values
  - Optimal Q-values obey a Bellman equation

$$Q^*(s, a) = \mathbb{E}_{s'}[r + \gamma \max_{a'} Q^*(s', a') \mid s, a]$$

learning target

- *Value iteration* algorithms solve the Bellman equation

$$Q_{i+1}(s, a) = \mathbb{E}_{s'}[r + \gamma \max_{a'} Q_i(s', a') \mid s, a]$$

# Deep Q-Networks (DQN) (Minh et al., 2013)

- Represent value function by deep Q-network with weights  $w$

$$Q(s, a, w) \approx Q^*(s, a)$$

- Objective is to minimize MSE loss by SGD

$$L(w) = \mathbb{E} \left[ \left( r + \gamma \max_{a'} Q(s', a', w) - Q(s, a, w) \right)^2 \right]$$

- Leading to the following Q-learning gradient

$$\frac{\partial L(w)}{\partial w} = \mathbb{E} \left[ \left( r + \gamma \max_{a'} Q(s', a', w) - Q(s, a, w) \right) \frac{\partial Q(s, a, w)}{\partial w} \right]$$

Issue: naïve Q-learning oscillates or diverges using NN due to:  
1) correlations between samples 2) non-stationary targets

# Stability by DQN

33

- Naive Q-learning **oscillates** or **diverges** with neural nets
  - 1) Sequential data: correlated, non-independent and identically distributed → use **experience replay**
  - 2) Policy oscillation: changes rapidly with slight changes to Q-values → freeze **target Q-network**
  - 3) Unknown scale of rewards and Q-values → **clip** rewards or **normalize** network adaptively to sensible range, **double Q-learning**

## Part II

Deep Learning Based Dialogue System

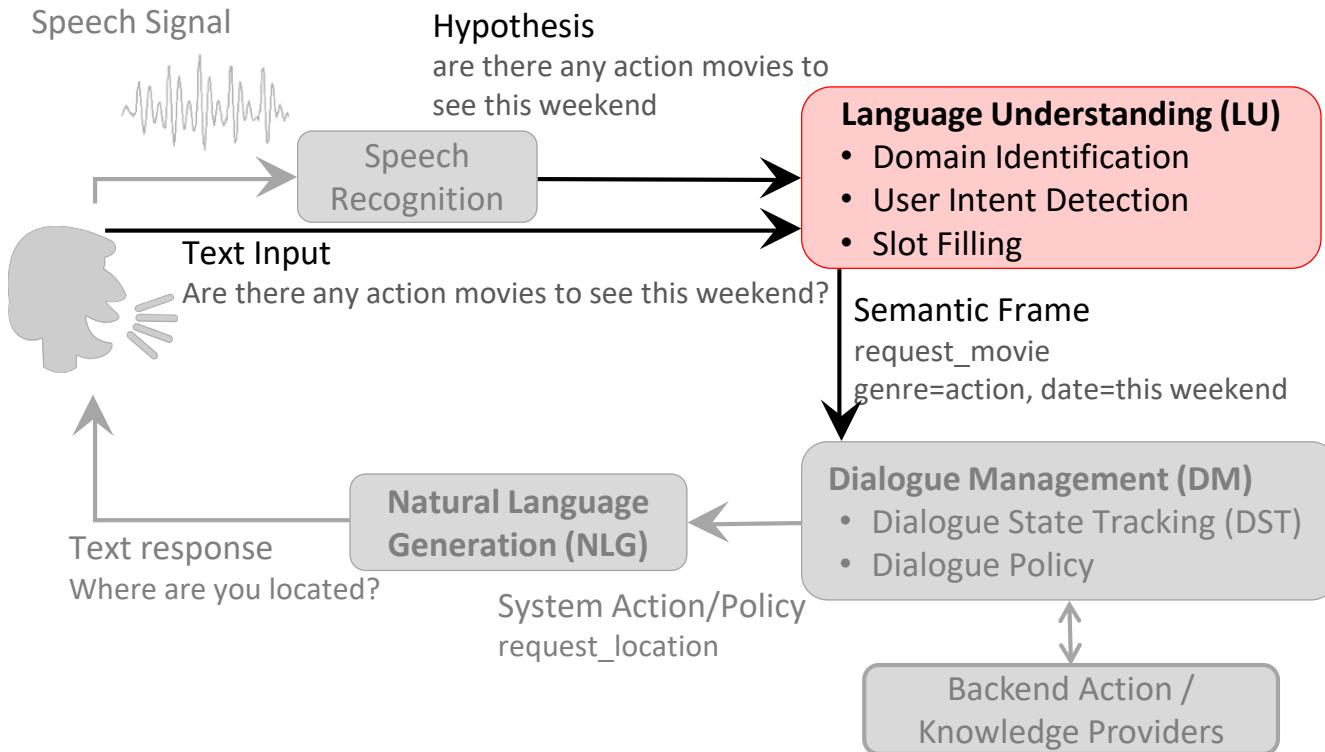
# Outline

35

- Introduction and Background
  - Neural Networks
  - Reinforcement Learning
- Deep Learning Based Dialogue System
  - **Spoken/Natural Language Understanding (SLU/NLU)**
  - Dialogue State Tracking (DST)
  - Dialogue Policy
  - Natural Language Generation (NLG)
  - End-to-End Learning for Dialogue Systems
- Evaluation
- Recent Trends on Learning Dialogues
- Challenges
- Conclusion

# Task-Oriented Dialogue System (Young, 2000)

36



# Semantic Frame Representation

37

- Requires a domain ontology: early connection to **backend**
- Contains **core content (intent, a set of slots with fillers)**

**Restaurant  
Domain**



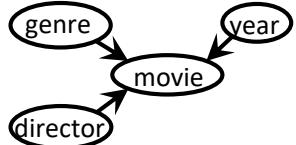
find me a cheap taiwanese restaurant in oakland



find\_restaurant (price="cheap",  
type="taiwanese", location="oakland")

**Movie  
Domain**

show me action movies directed by james cameron

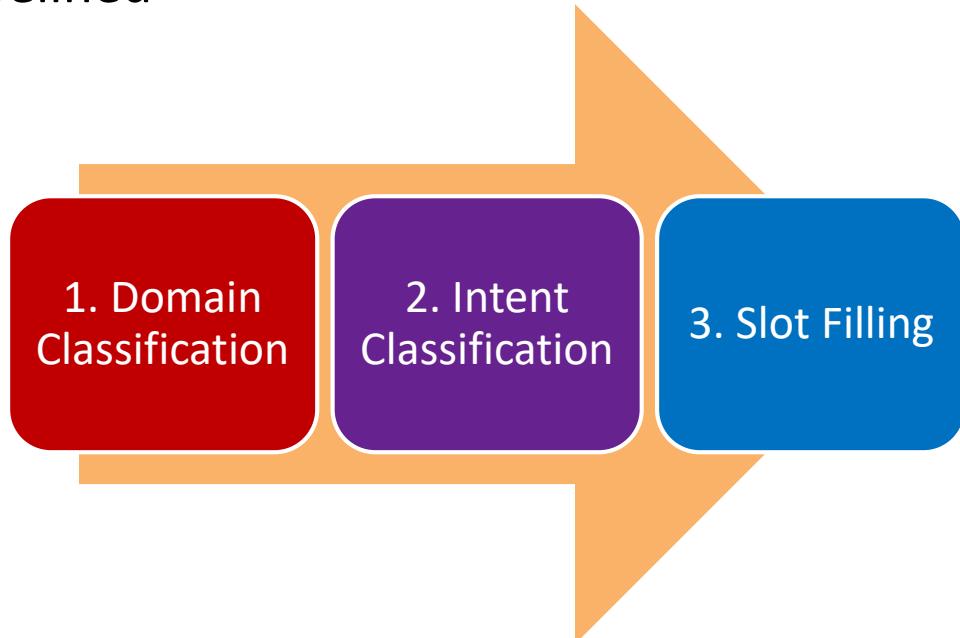


find\_movie (genre="action",  
director="james cameron")

# Language Understanding (LU)

38

## □ Pipelined



# LU – Domain/Intent Classification

39

As an **utterance classification task**

- Given a collection of utterances  $u_i$  with labels  $c_i$ ,  
 $D = \{(u_1, c_1), \dots, (u_n, c_n)\}$  where  $c_i \in C$ , train a model to estimate labels for new utterances  $u_k$ .

find me a cheap taiwanese restaurant in oakland

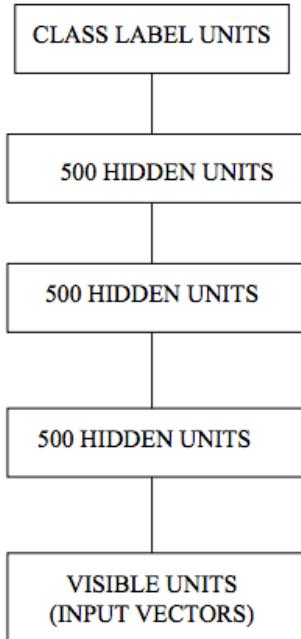
Domain	Intent
Movies	find_movie, buy_tickets
Restaurants	find_restaurant, find_price, book_table
Music	find_lyrics, find_singer
Sports	...
...	

# Deep Neural Networks for Domain/Intent Classification – I (Sarikaya et al, 2011)

40

<http://ieeexplore.ieee.org/abstract/document/5947649>

- Deep belief nets (DBN)
  - Unsupervised training of weights
  - Fine-tuning by back-propagation
  - Compared to MaxEnt, SVM, and boosting

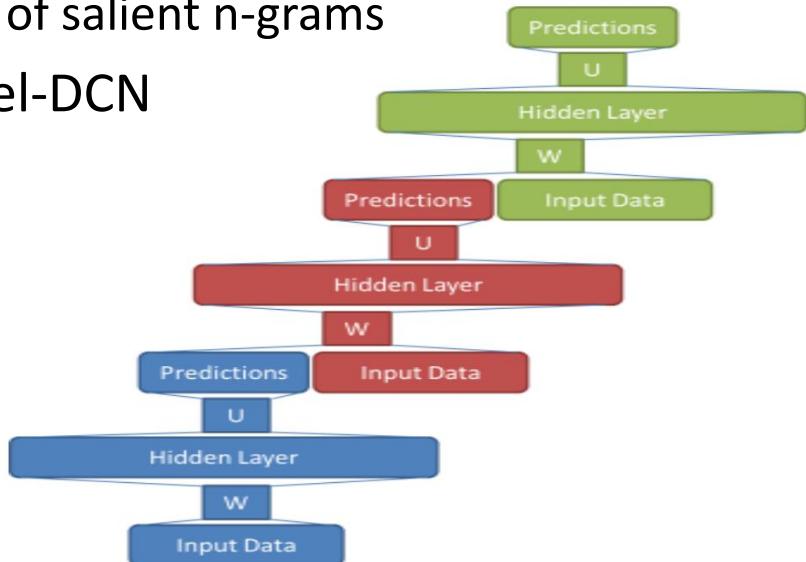


# Deep Neural Networks for Domain/Intent Classification – II (Tur et al., 2012; Deng et al., 2012)

41

<http://ieeexplore.ieee.org/abstract/document/6289054/>; <http://ieeexplore.ieee.org/abstract/document/6424224>

- Deep convex networks (DCN)
  - Simple classifiers are stacked to learn complex functions
  - Feature selection of salient n-grams
- Extension to kernel-DCN

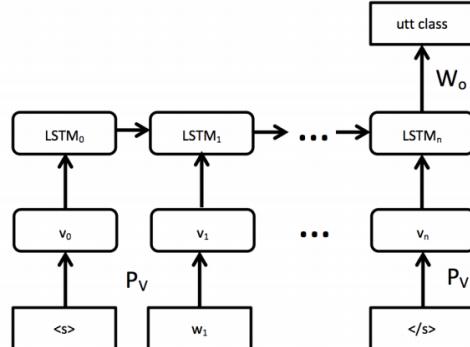
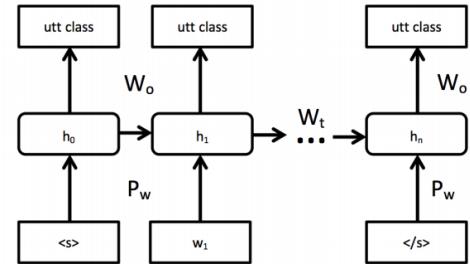


# Deep Neural Networks for Domain/Intent Classification – III (Ravuri and Stolcke, 2015)

42

[https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/RNNLM\\_addressee.pdf](https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/RNNLM_addressee.pdf)

- RNN and LSTMs for utterance classification
- Word hashing to deal with large number of singletons
  - Kat: #Ka, Kat, at#
  - Each character n-gram is associated with a bit in the input encoding



# LU – Slot Filling

43

As a sequence  
tagging task

- Given a collection tagged word sequences,  
 $S = \{(w_{1,1}, w_{1,2}, \dots, w_{1,n_1}), (t_{1,1}, t_{1,2}, \dots, t_{1,n_1}), (w_{2,1}, w_{2,2}, \dots, w_{2,n_2}), (t_{2,1}, t_{2,2}, \dots, t_{2,n_2})\} \dots\}$   
where  $t_i \in M$ , the goal is to estimate tags for a new word sequence.

flights from Boston to New York today

	flights	from	Boston	to	New	York	today
Entity Tag	O	O	B-city	O	B-city	I-city	O
Slot Tag	O	O	B-dept	O	B-arrival	I-arrival	B-date

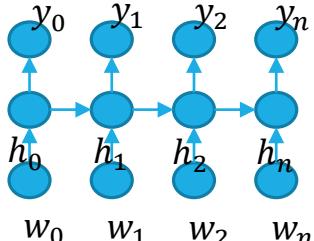
# Recurrent Neural Nets for Slot Tagging – I

(Yao et al, 2013; Mesnil et al, 2015)

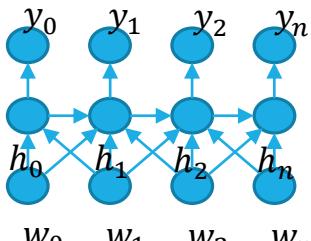
44

<http://131.107.65.14/en-us/um/people/gzweig/Pubs/Interspeech2013RNNLU.pdf>; <http://dl.acm.org/citation.cfm?id=2876380>

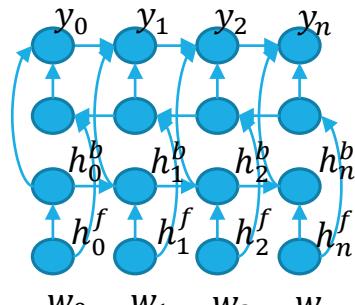
- Baseline: conditional random fields on ATIS corpus
- Variations:
  - a. RNNs with LSTM cells
  - b. Input, sliding window of n-grams
  - c. Bi-directional LSTMs



(a) LSTM



(b) LSTM-LA



(c) bLSTM

# Recurrent Neural Nets for Slot Tagging – II

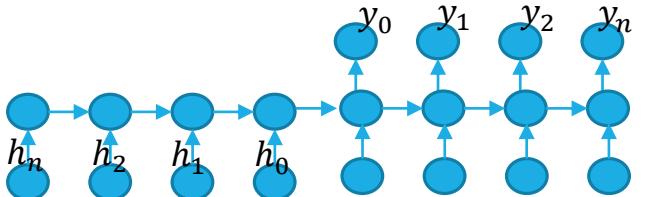
(Kurata et al., 2016; Simonnet et al., 2015)

45

<http://www.aclweb.org/anthology/D16-1223>

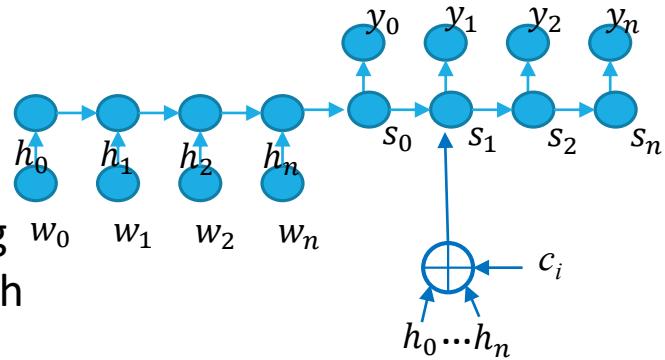
- Encoder-decoder networks

- Leverages sentence level information



- Attention-based encoder-decoder

- Use of attention (as in MT) in the encoder-decoder network



- Attention is estimated using a feed-forward network with input:  $h_t$  and  $s_t$  at time  $t$

# Recurrent Neural Nets for Slot Tagging – III

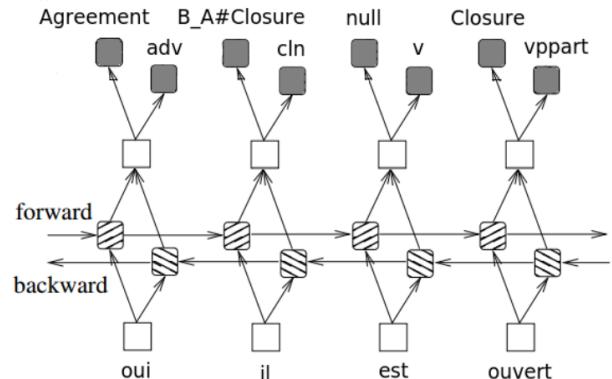
(Jaech et al., 2016; Tafforeau et al., 2016)

46

<https://arxiv.org/abs/1604.00117>; [http://www.sensei-conversation.eu/wp-content/uploads/2016/11/favre\\_is2016b.pdf](http://www.sensei-conversation.eu/wp-content/uploads/2016/11/favre_is2016b.pdf)

## □ Multi-task learning

- Goal: exploit data from domains/tasks with a lot of data to improve ones with less data
- Lower layers are shared across domains/tasks
- Output layer is specific to task



# Joint Semantic Frame Parsing

47

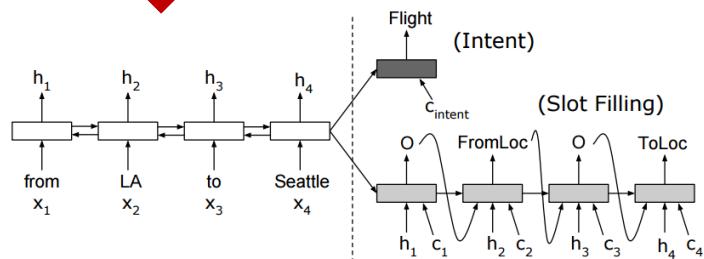
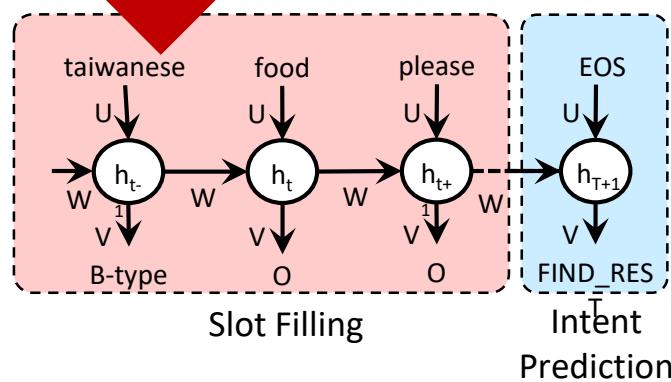
[https://www.microsoft.com/en-us/research/wp-content/uploads/2016/06/IS16\\_MultiJoint.pdf](https://www.microsoft.com/en-us/research/wp-content/uploads/2016/06/IS16_MultiJoint.pdf); <https://arxiv.org/abs/1609.01454>

Sequence-based  
(Hakkani-Tur et al., 2016)

- Slot filling and intent prediction in the same output sequence

Parallel  
(Liu and Lane, 2016)

- Intent prediction and slot filling are performed in two branches



# Contextual LU



Domain Identification → Intent Prediction → Slot Filling

*D* communication

*I* send\_email

*U* just sent email to bob about fishing this weekend  
*S* O O O O O I-subject I-subject I-subject  
 → send\_email(contact\_name="bob", subject="fishing this weekend")

*Single Turn*

*U<sub>1</sub>* send email to bob

*S<sub>1</sub>* B-contact\_name  
 → send\_email(contact\_name="bob")

*U<sub>2</sub>* are we going to fish this weekend  
*S<sub>2</sub>* B-message I-message I-message I-message I-message I-message  
 → send\_email(message="are we going to fish this weekend")

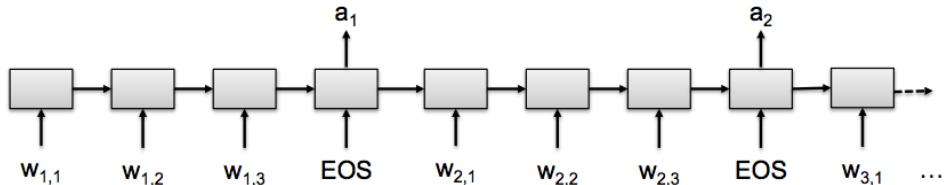
*Multi-Turn*

# Contextual LU (Bhargava et al., 2013; Hori et al, 2015)

49

<https://www.merl.com/publications/docs/TR2015-134.pdf>

- Leveraging contexts
  - Used for individual tasks
- Seq2Seq model
  - Words are input one at a time, tags are output at the end of each utterance

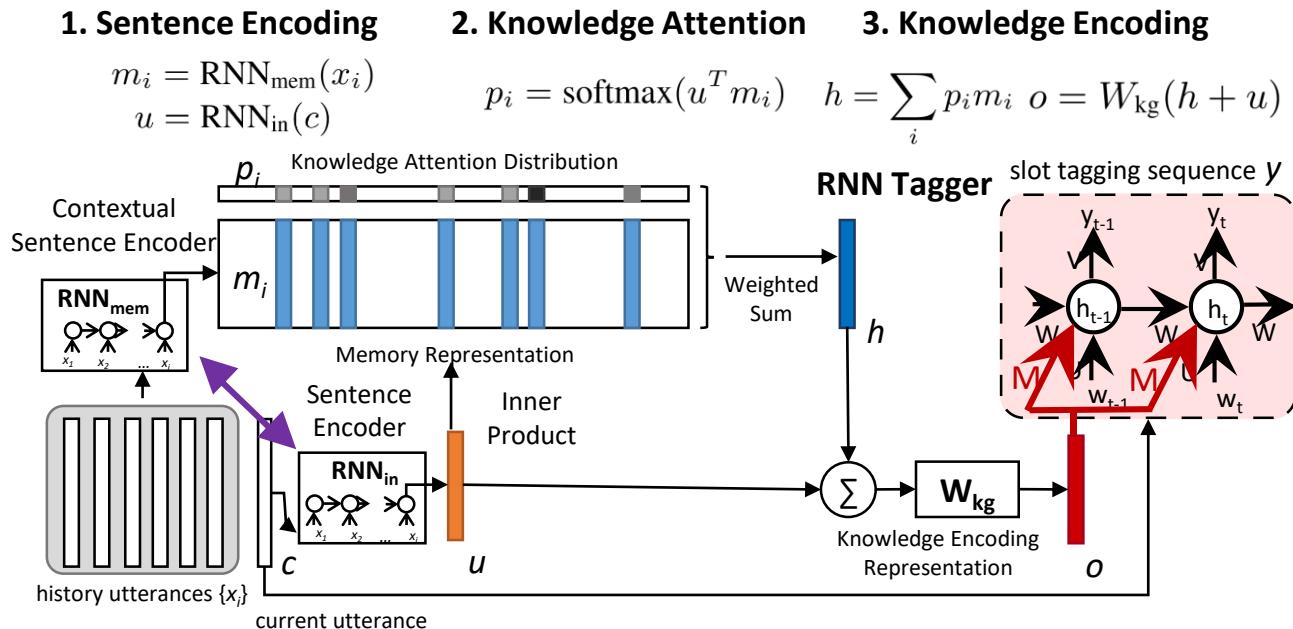


- Extension: LSTM with speaker role dependent layers

# E2E MemNN for Contextual LU (Chen et al., 2016)

50

[https://www.microsoft.com/en-us/research/wp-content/uploads/2016/06/IS16\\_ContextualSLU.pdf](https://www.microsoft.com/en-us/research/wp-content/uploads/2016/06/IS16_ContextualSLU.pdf)



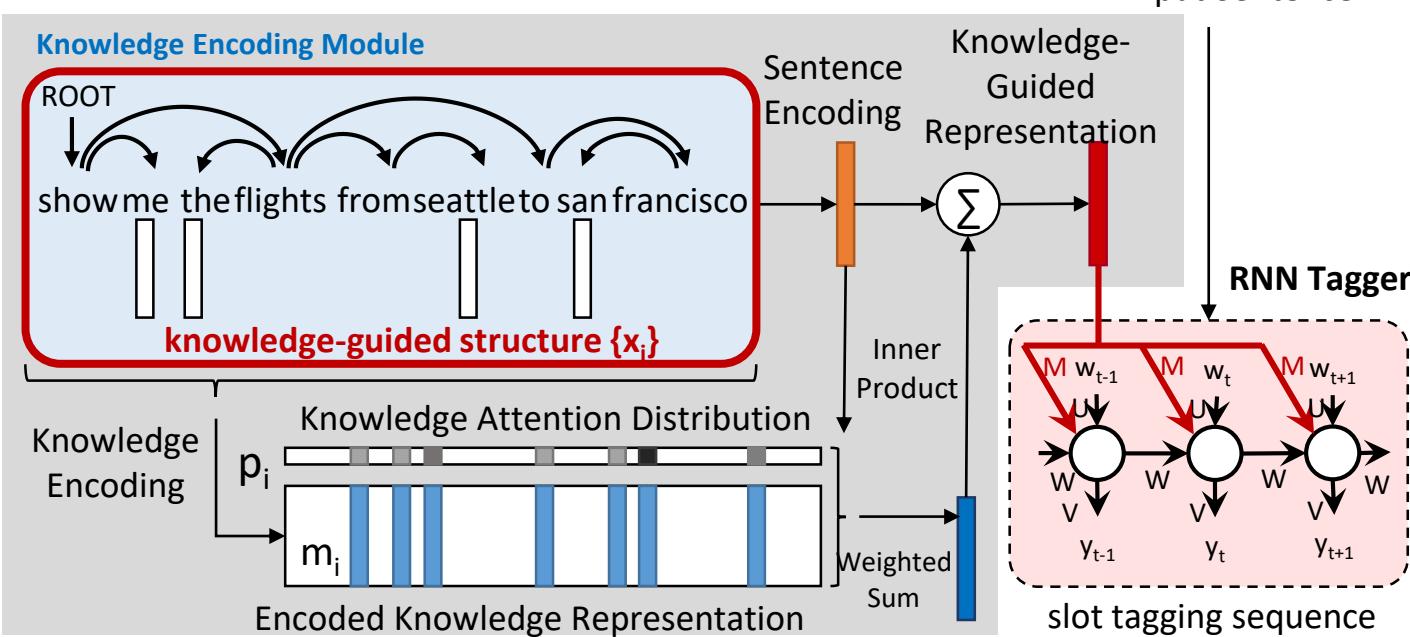
Idea: additionally incorporating contextual knowledge during slot tagging  
→ track dialogue states in a latent way

# Structural LU (Chen et al., 2016)

51

<http://arxiv.org/abs/1609.03286>

## □ Prior knowledge as a teacher



# Structural LU (Chen et al., 2016)

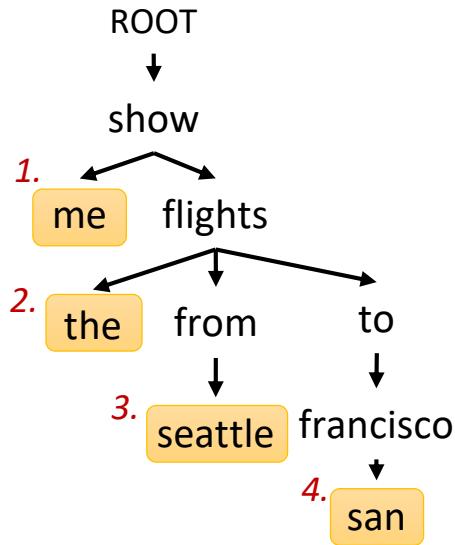
52

<http://arxiv.org/abs/1609.03286>

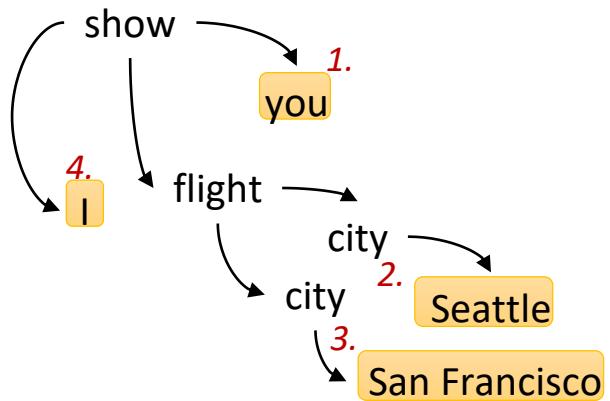
- Sentence structural knowledge stored as memory

**Sentence  $s$**  show me the flights from seattle to san francisco

Syntax (Dependency Tree)



Semantics (AMR Graph)



# LU Evaluation

53

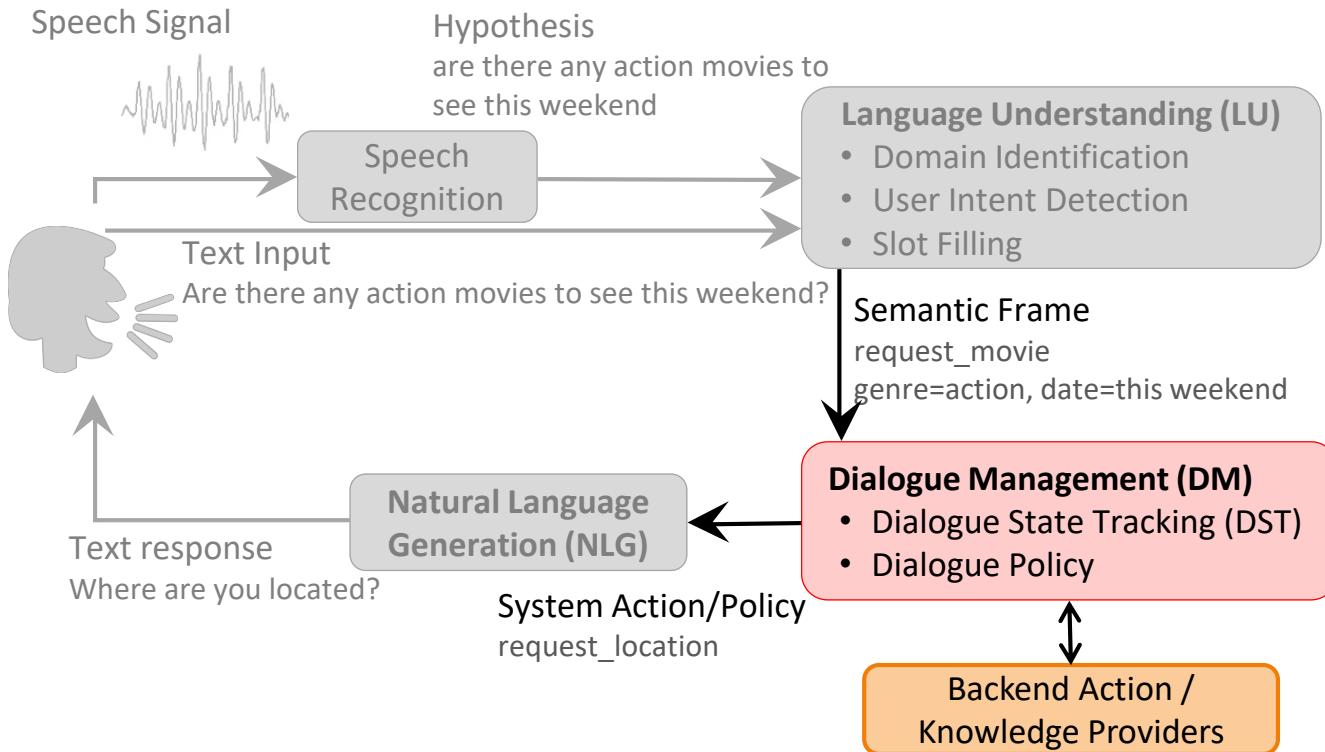
- Metrics
  - Sub-sentence-level: intent accuracy, slot F1
  - Sentence-level: whole frame accuracy

# Outline

- Introduction and Background
  - Neural Networks
  - Reinforcement Learning
- Deep Learning Based Dialogue System
  - Spoken/Natural Language Understanding (SLU/NLU)
  - **Dialogue State Tracking (DST)**
  - Dialogue Policy
  - Natural Language Generation (NLG)
  - End-to-End Learning for Dialogue Systems
- Evaluation
- Recent Trends on Learning Dialogues
- Challenges
- Conclusion

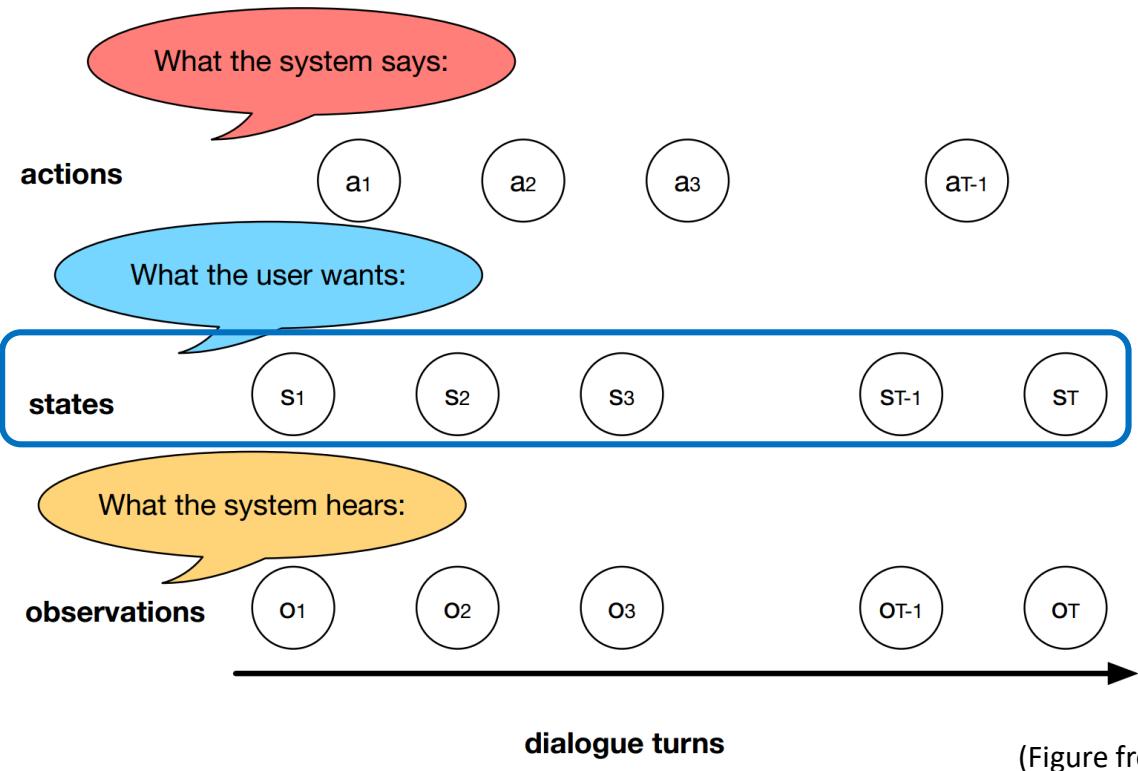
# Task-Oriented Dialogue System (Young, 2000)

55



# Elements of Dialogue Management

56



# Dialogue State Tracking (DST)

57

- Dialogue state: *a representation of the system's belief of the user's goal(s) at any time during the dialogue*
- Inputs
  - Current user utterance
  - Preceding system response
  - Results from previous turns
- For
  - Looking up knowledge or making API call(s)
  - Generating the next system action/response

# Dialogue State Tracking (DST)

58

## sample problem

S: where would you like to fly from?

U: [Boston/0.45]; [Austin/0.30]

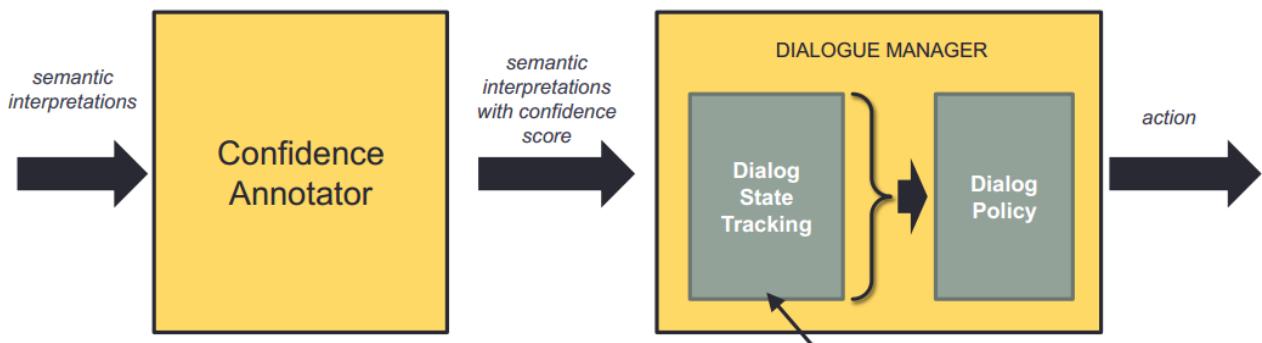
S: sorry, did you say you wanted to fly from Boston?

U: [No/0.37] + [Aspen / 0.7]

---

Updated belief = ?

[Boston/?; Austin/?; Aspen/?]



# Dialogue State Tracking (DST)

59

- Maintain a probabilistic distribution instead of a 1-best prediction for better robustness to recognition errors

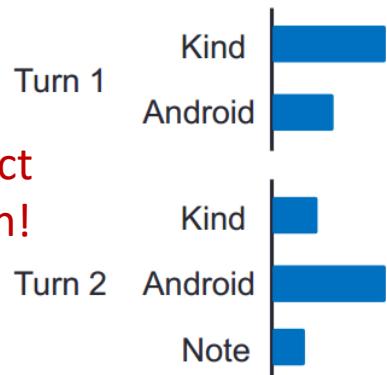
Turn 1
Kind
Android

Turn 1	
Kind	0.5
Android	0.3

Turn 2
Note
Android

Turn 2	
Note	0.4
Android	0.3

Incorrect  
for both!



# Dialogue State Tracking (DST)

60

- ❑ Maintain a probabilistic distribution instead of a 1-best prediction for better robustness to SLU errors or ambiguous input

Slot	Value
# people	5 (0.5)
time	5 (0.5)

Slot	Value
# people	3 (0.8)
time	5 (0.8)



# Dialog State Tracking Challenge (DSTC)

(Williams et al. 2013, Henderson et al. 2014, Henderson et al. 2014, Kim et al. 2016, Kim et al. 2016)

61

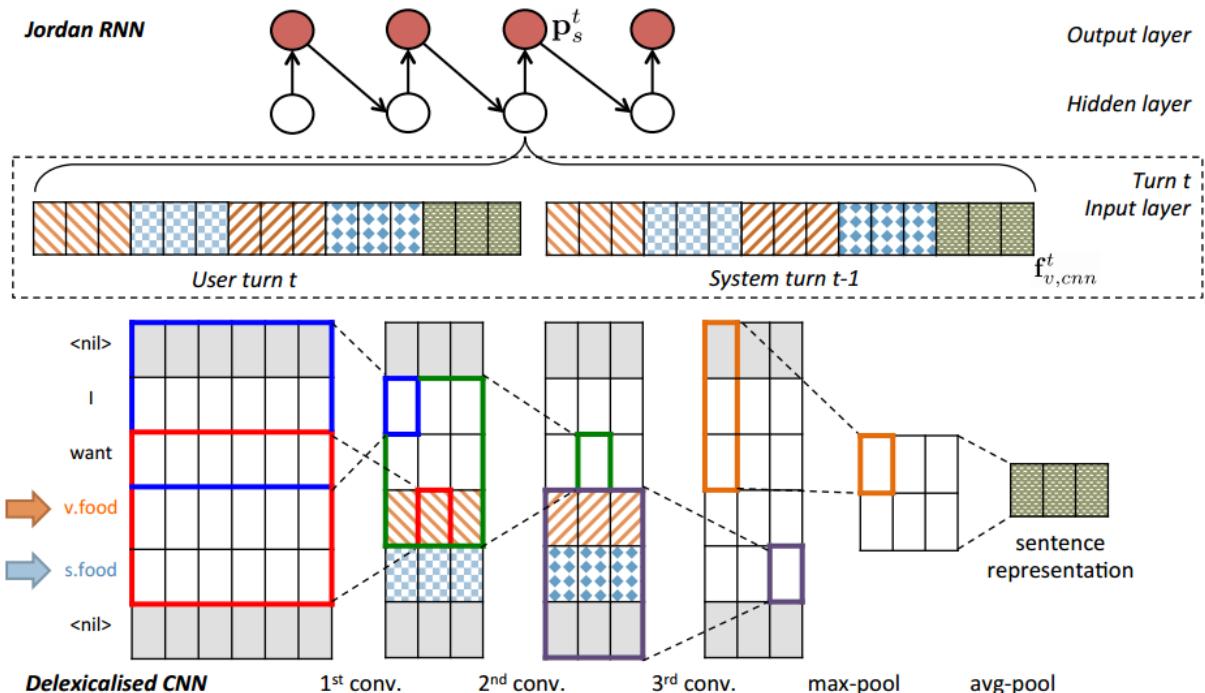
Challenge	Type	Domain	Data Provider	Main Theme
<a href="#"><u>DSTC1</u></a>	Human-Machine	Bus Route	CMU	Evaluation Metrics
<a href="#"><u>DSTC2</u></a>	Human-Machine	Restaurant	U. Cambridge	User Goal Changes
<a href="#"><u>DSTC3</u></a>	Human-Machine	Tourist Information	U. Cambridge	Domain Adaptation
<a href="#"><u>DSTC4</u></a>	Human-Human	Tourist Information	I2R	Human Conversation
<a href="#"><u>DSTC5</u></a>	Human-Human	Tourist Information	I2R	Language Adaptation

# Neural Belief Tracker

(Henderson et al., 2013;  
Henderson et al., 2014; Mrkšić et al., 2015)

62

<http://www.anthology.aclweb.org/W/W13/W13-4073.pdf>; <https://arxiv.org/abs/1506.07190>

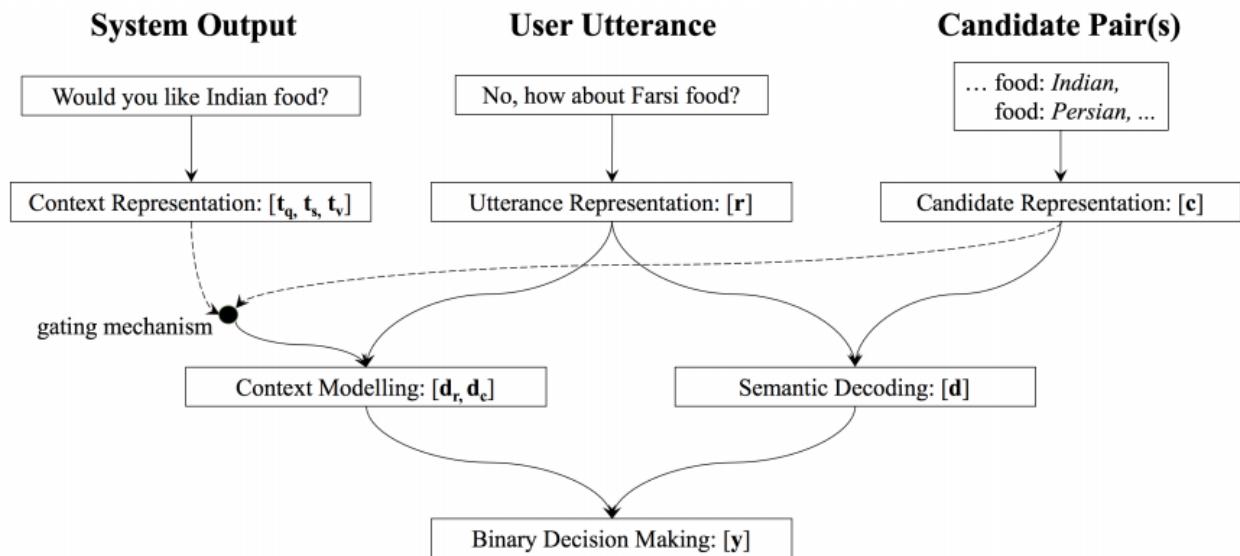


(Figure from Wen et al, 2016)

# Neural Belief Tracker (Mrkšić et al., 2016)

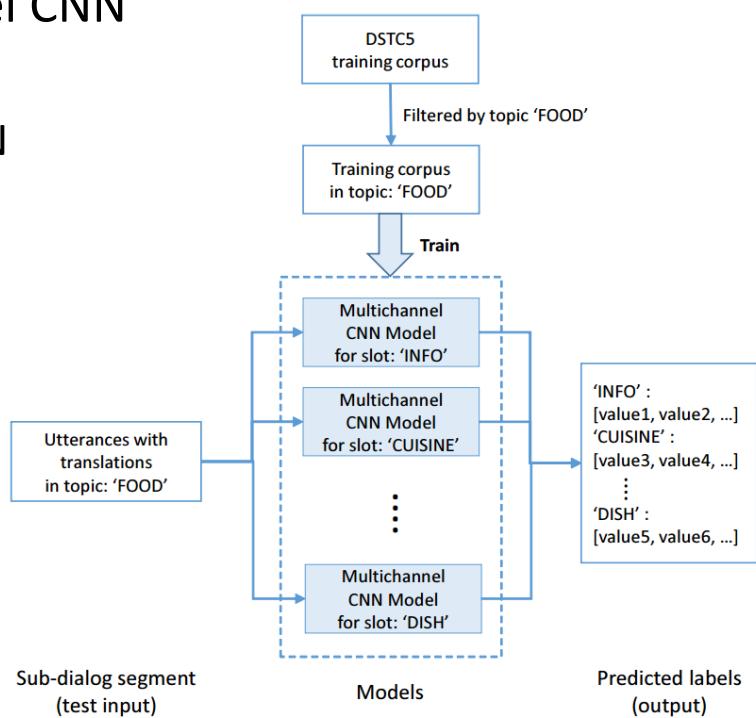
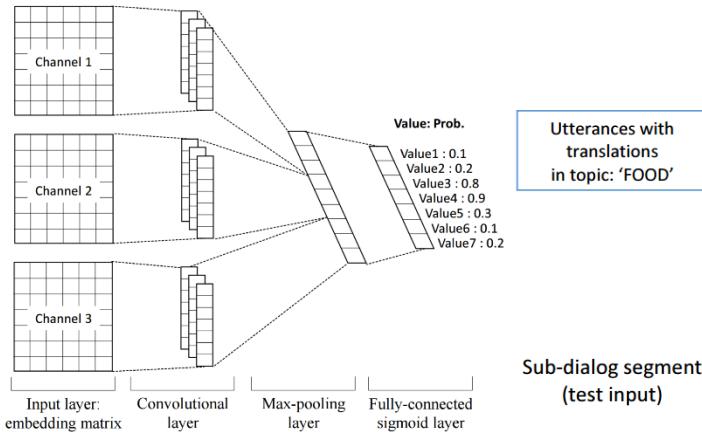
63

<https://arxiv.org/abs/1606.03777>



# Multichannel Tracker (Shi et al., 2016)

- Training a multichannel CNN for each slot
  - Chinese character CNN
  - Chinese word CNN
  - English word CNN



# DST Evaluation

65

- Dialogue State Tracking Challenges
  - DSTC2-3, human-machine
  - DSTC4-5, human-human
- Metric
  - Tracked state accuracy with respect to user goal
  - Recall/Precision/F-measure individual slots

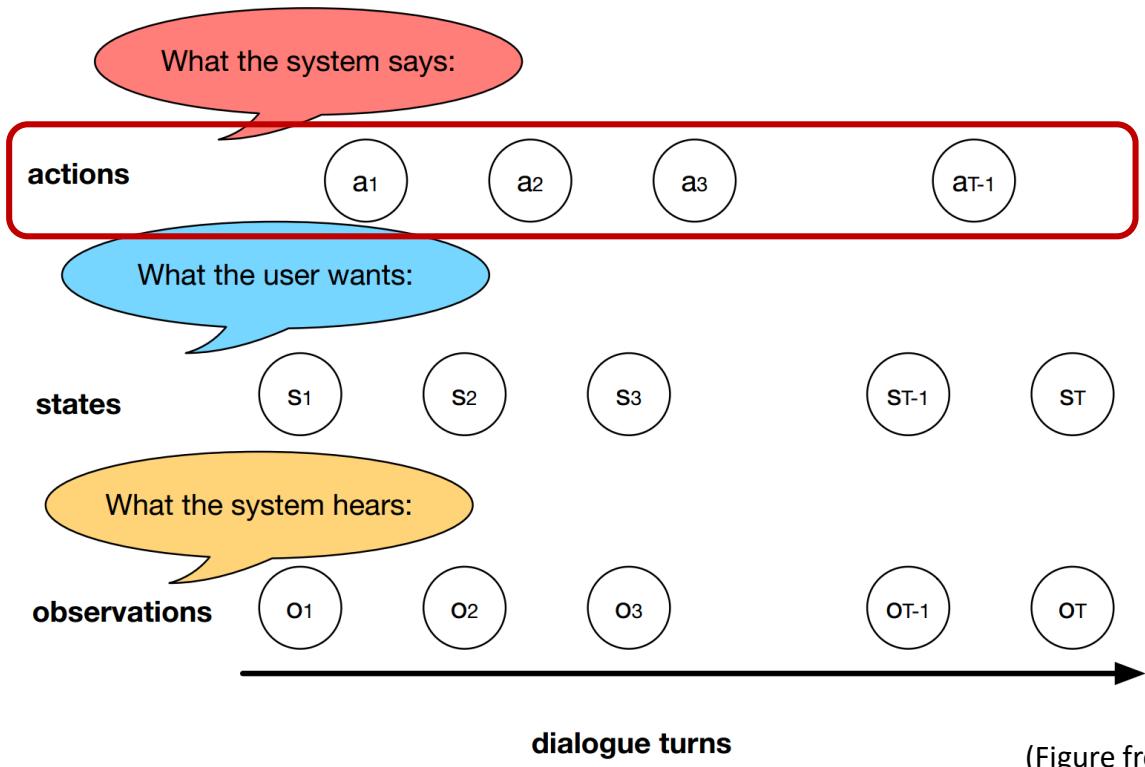
# Outline

66

- Introduction and Background
  - Neural Networks
  - Reinforcement Learning
- Deep Learning Based Dialogue System
  - Spoken/Natural Language Understanding (SLU/NLU)
  - Dialogue State Tracking (DST)
  - **Dialogue Policy**
  - Natural Language Generation (NLG)
  - End-to-End Learning for Dialogue Systems
- Evaluation
- Recent Trends on Learning Dialogues
- Challenges
- Conclusion

# Elements of Dialogue Management

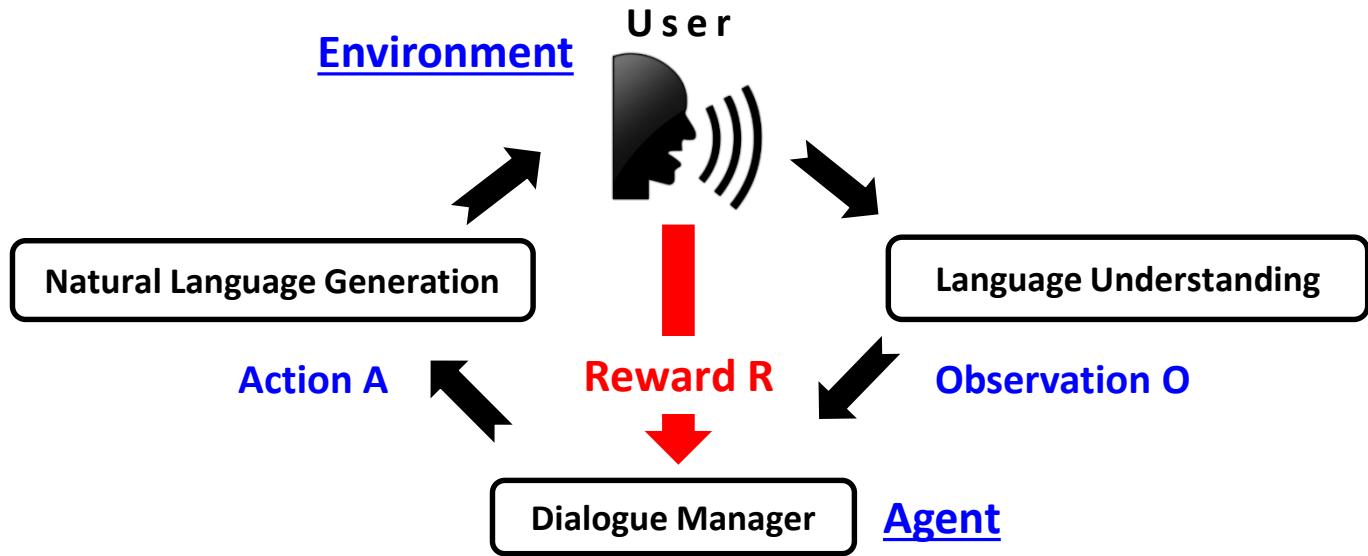
67



# Dialogue Policy Optimization

68

- Dialogue management in a RL framework



The optimized dialogue policy selects the best action that maximizes the future reward.  
Correct rewards are a crucial factor in dialogue policy training

# Reward for RL $\cong$ Evaluation for System

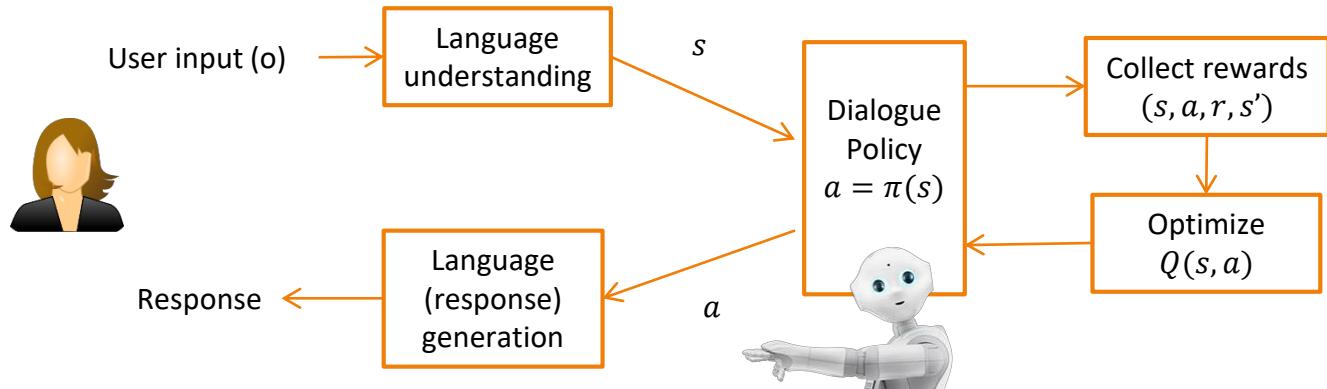
69

- Dialogue is a special RL task
  - Human involves in interaction and rating (evaluation) of a dialogue
  - Fully human-in-the-loop framework
- Rating: correctness, appropriateness, and adequacy

- Expert rating	high quality, <b>high</b> cost
- User rating	unreliable quality, <b>medium</b> cost
- Objective rating	Check desired aspects, <b>low</b> cost

# Reinforcement Learning for Dialogue Policy Optimization

70



Type of Bots	State	Action	Reward
Social ChatBots	Chat history	System Response	# of turns maximized; Intrinsically motivated reward
InfoBots (interactive Q/A)	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

Goal: develop a generic deep RL algorithm to learn dialogue policy for all bot categories

# Dialogue Reinforcement Learning Signal

71

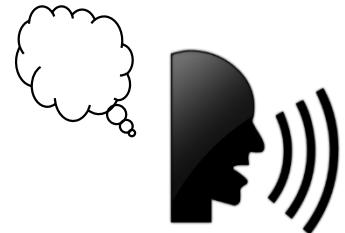
Typical reward function

- -1 for per turn penalty
- Large reward at completion if **successful**

Typically requires **domain knowledge**

- ✓ Simulated user
- ✗ Paid users (Amazon Mechanical Turk)
- ✗ Real users

The user simulator is usually required for dialogue system training before deployment

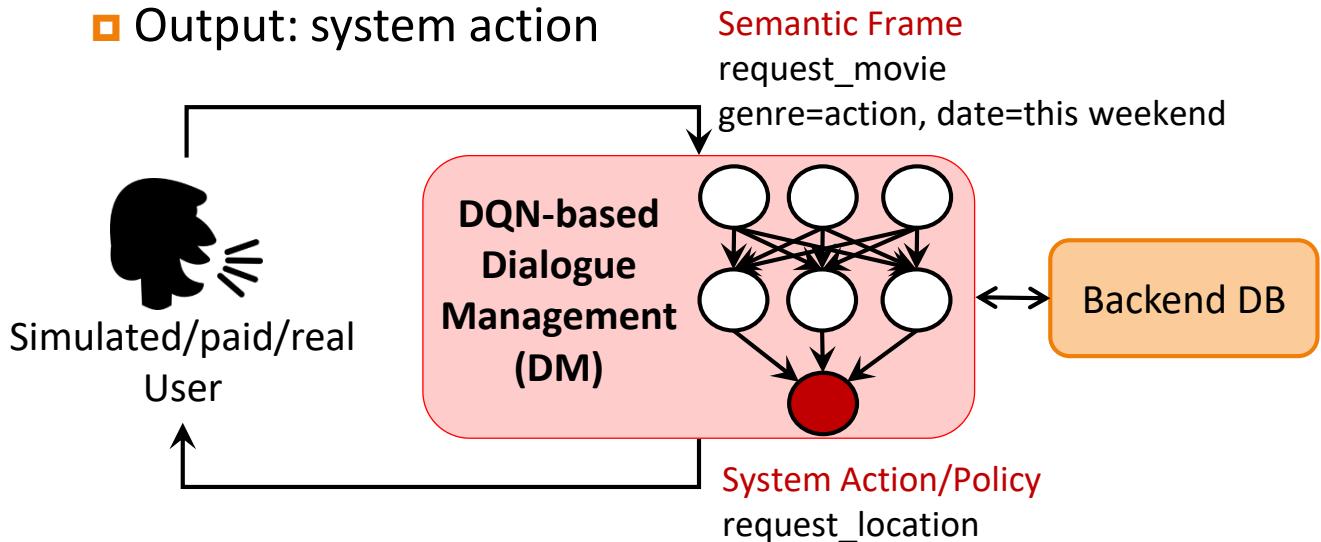


# DQN for Dialogue Management (Li et al., 2017)

72

<https://arxiv.org/abs/1703.01008>

- Deep RL for training DM
  - Input: current semantic frame observation, database returned results
  - Output: system action



# Online Training (Su et al., 2015; Su et al., 2016)

- Policy learning from real users
  - Infer reward directly from dialogues (Su et al., 2015)
  - User rating (Su et al., 2016)
- Reward modeling on user binary success rating

Hi, How may I help you?

I want some cheap Chinese food.

Where in the city would you like?

Somewhere in the west, please.

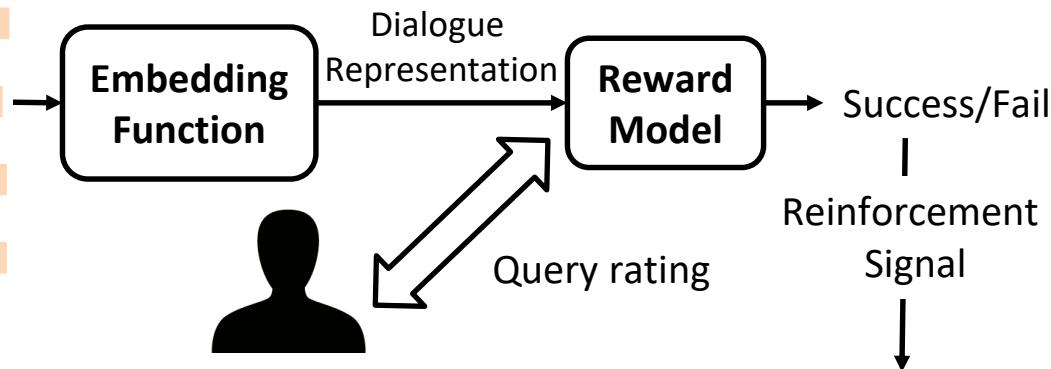
Yim Wah is a nice Chinese place.

Great, can you give me its address?

It is at 2-4 Lensfield Road.

Ok, thank you, bye!

Thanks, goodbye.



# Dialogue Management Evaluation

74

- Metrics
  - Turn-level evaluation: system action accuracy
  - Dialogue-level evaluation: task success rate, reward

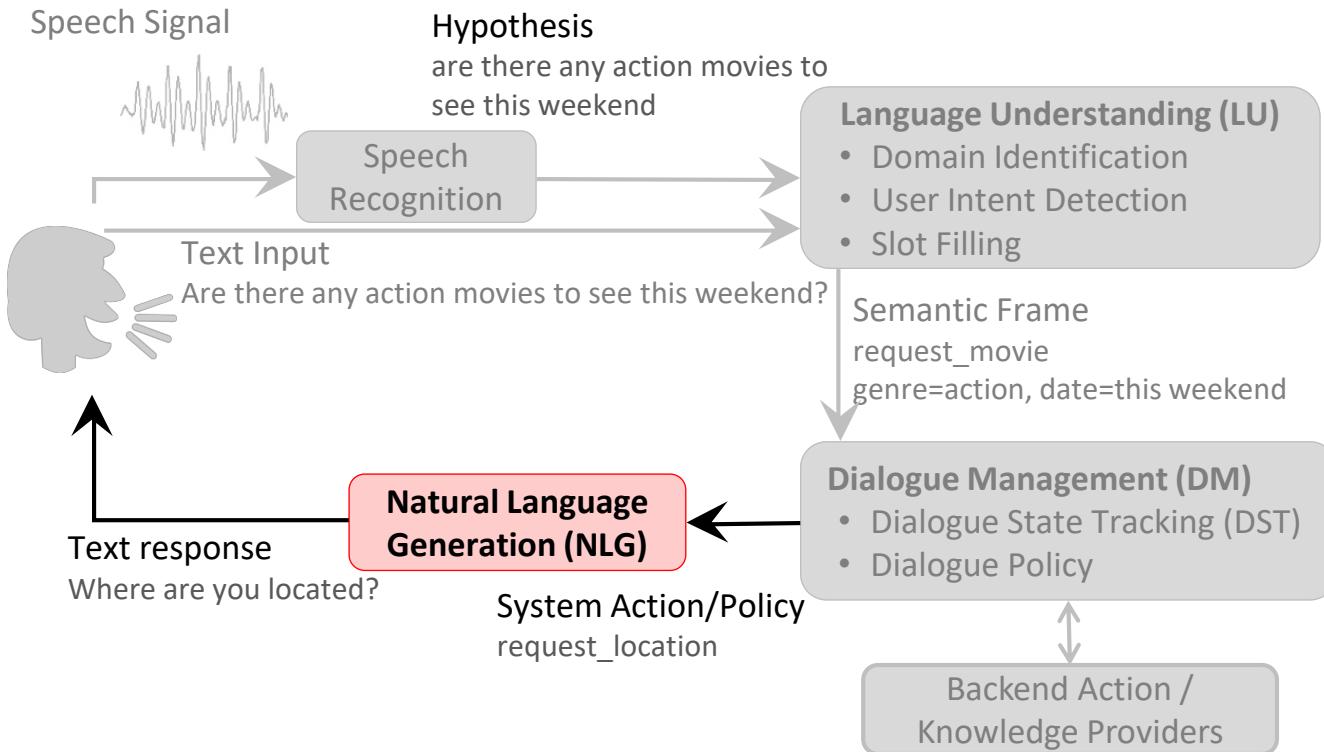
# Outline

75

- Introduction and Background
  - Neural Networks
  - Reinforcement Learning
- Deep Learning Based Dialogue System
  - Spoken/Natural Language Understanding (SLU/NLU)
  - Dialogue State Tracking (DST)
  - Dialogue Policy
  - **Natural Language Generation (NLG)**
  - End-to-End Learning for Dialogue Systems
- Evaluation
- Recent Trends on Learning Dialogues
- Challenges
- Conclusion

# Task-Oriented Dialogue System (Young, 2000)

76



# Natural Language Generation (NLG)

77

- Mapping dialogue acts into natural language

inform(name=Seven\_Days, foodtype=Chinese)



Seven Days is a nice Chinese restaurant

# Template-Based NLG

78

- Define a set of rules to map frames to NL

Semantic Frame	Natural Language
confirm()	“Please tell me more about the product your are looking for.”
confirm(area=\$V)	“Do you want somewhere in the \$V?”
confirm(food=\$V)	“Do you want a \$V restaurant?”
confirm(food=\$V,area=\$W)	“Do you want a \$V restaurant in the \$W.”

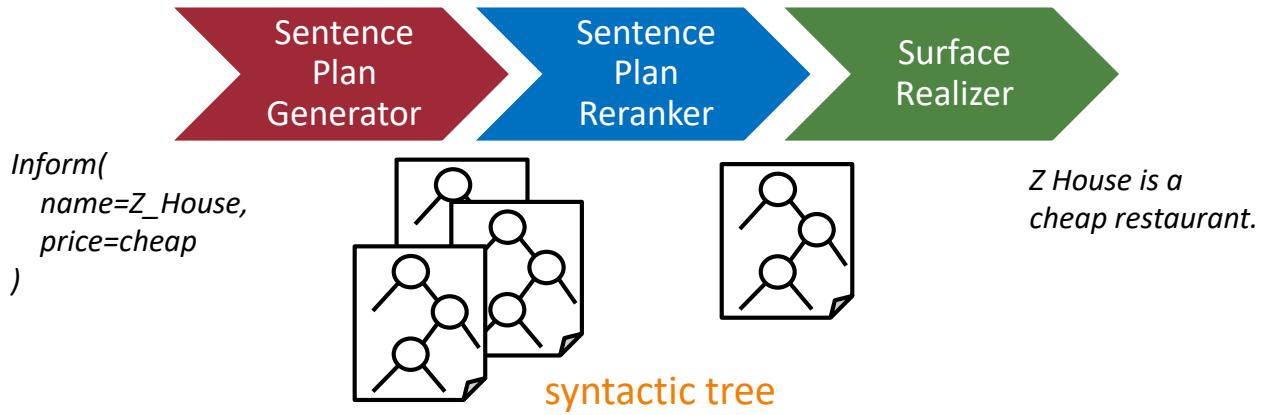
**Pros:** simple, error-free, easy to control

**Cons:** time-consuming, rigid, poor scalability

# Plan-Based NLG (Walker et al., 2002)

79

- Divide the problem into pipeline



- Statistical sentence plan generator (Stent et al., 2009)
- Statistical surface realizer (Dethlefs et al., 2013; Cuayáhuitl et al., 2014; ...)

**Pros:** can model complex linguistic structures

**Cons:** heavily engineered, require domain knowledge

# Class-Based LM NLG (Oh and Rudnicky, 2000)

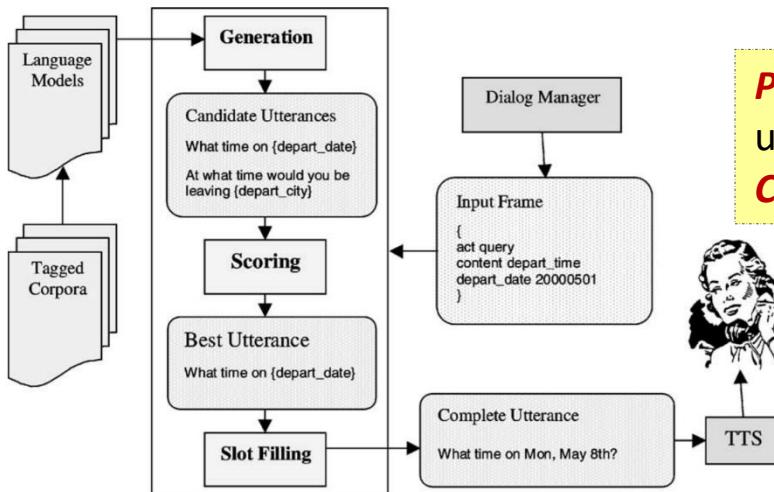
- Class-based language modeling

$$P(X \mid c) = \sum_t \log p(x_t \mid x_0, x_1, \dots, x_{t-1}, c)$$

- NLG by decoding  $X^* = \arg \max_X P(X \mid c)$

Classes:

- inform\_area
- inform\_address
- ...
- request\_area
- request\_postcode



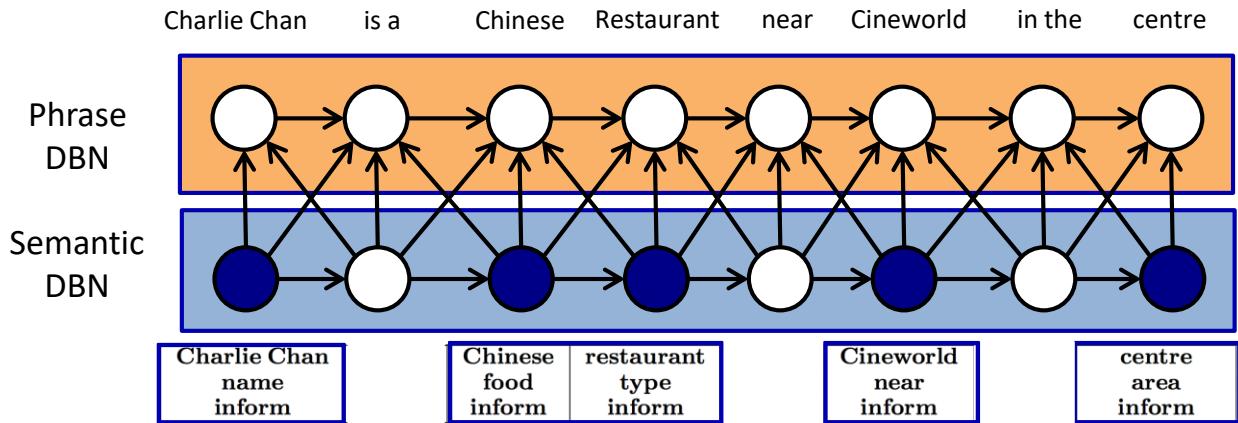
**Pros:** easy to implement/  
understand, simple rules

**Cons:** computationally inefficient

# Phrase-Based NLG (Mairesse et al, 2010)

81

<http://dl.acm.org/citation.cfm?id=1858838>



Inform(name=Charlie Chan, food=Chinese, type= restaurant, near=Cineworld, area=centre)  
realization phrase semantic stack

$r_t$	$s_t$	$h_t$	$l_t$
< s >	START	START	START
The Rice Boat	inform(name(X))	X	inform(name)
is a	inform	inform	EMPTY
restaurant	inform(type(restaurant))	restaurant	inform(type)
in the	inform(area)	area	inform
riverside	inform(area(riverside))	riverside	inform(area)
area	inform(area)	area	inform
that	inform	inform	EMPTY
serves	inform(food)	food	inform
French	inform(food(French))	French	inform(food)
food	inform(food)	food	inform
< / s >	END	END	END

**Pros:** efficient, good performance

**Cons:** require semantic alignments

# RNN-Based LM NLG (Wen et al., 2015)

Input

*Inform(name=Din Tai Fung, food=Taiwanese)*

{ 0, 0, 1, 0, 0, ..., 1, 0, 0, ..., 1, 0, 0, 0, 0, 0, 0 ... }

**dialogue act 1-hot representation**

conditioned on  
the dialogue act

SLOT\_NAME

serves

SLOT\_FOOD

.  
<EOS>

Output

<BOS>

SLOT\_NAME

serves

SLOT\_FOOD

.

*delexicalisation*

Slot weight tying

# Handling Semantic Repetition

83

- Issue: semantic repetition
  - Din Tai Fung is a great Taiwanese restaurant that serves Taiwanese.
  - Din Tai Fung is a child friendly restaurant, and also allows kids.
- Deficiency in either model or decoding (or both)
- Mitigation
  - Post-processing rules (Oh & Rudnicky, 2000)
  - Gating mechanism (Wen et al., 2015)
  - Attention (Mei et al., 2016; Wen et al., 2015)

# Semantic Conditioned LSTM (Wen et al., 2015)

## □ Original LSTM cell

$$\mathbf{i}_t = \sigma(\mathbf{W}_{wi}\mathbf{x}_t + \mathbf{W}_{hi}\mathbf{h}_{t-1})$$

$$\mathbf{f}_t = \sigma(\mathbf{W}_{wf}\mathbf{x}_t + \mathbf{W}_{hf}\mathbf{h}_{t-1})$$

$$\mathbf{o}_t = \sigma(\mathbf{W}_{wo}\mathbf{x}_t + \mathbf{W}_{ho}\mathbf{h}_{t-1})$$

$$\hat{\mathbf{c}}_t = \tanh(\mathbf{W}_{wc}\mathbf{x}_t + \mathbf{W}_{hc}\mathbf{h}_{t-1})$$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \hat{\mathbf{c}}_t$$

$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t)$$

## □ Dialogue act (DA) cell

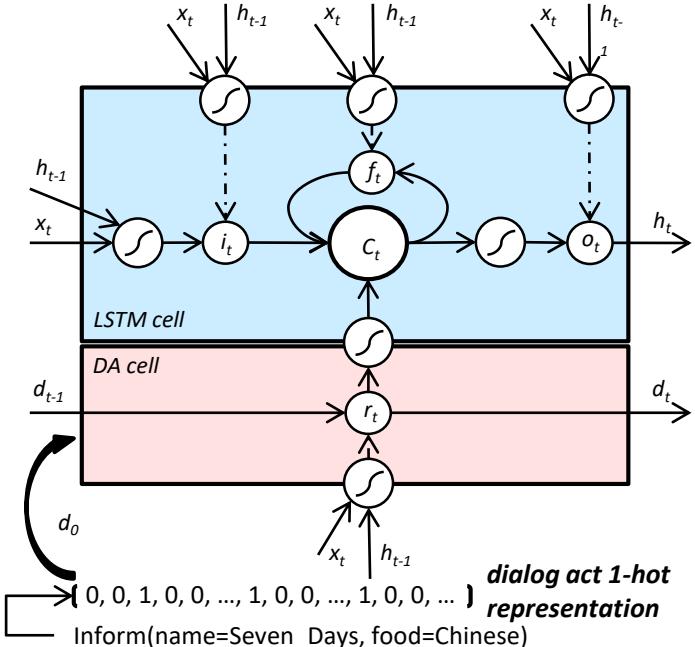
$$\mathbf{r}_t = \sigma(\mathbf{W}_{wr}\mathbf{x}_t + \mathbf{W}_{hr}\mathbf{h}_{t-1})$$

$$\mathbf{d}_t = \mathbf{r}_t \odot \mathbf{d}_{t-1}$$

## □ Modify $\mathbf{C}_t$

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \hat{\mathbf{c}}_t + \tanh(\mathbf{W}_{dc}\mathbf{d}_t)$$

<http://www.aclweb.org/anthology/D/D15/D15-1199.pdf>



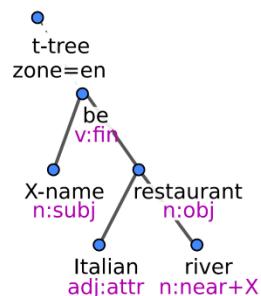
Idea: using gate mechanism to control the generated semantics (dialogue act/slots)

# Structural NLG (Dušek and Jurčíček, 2016)

- Goal: NLG based on the syntax tree

- Encode trees as sequences
- Seq2Seq model for generation

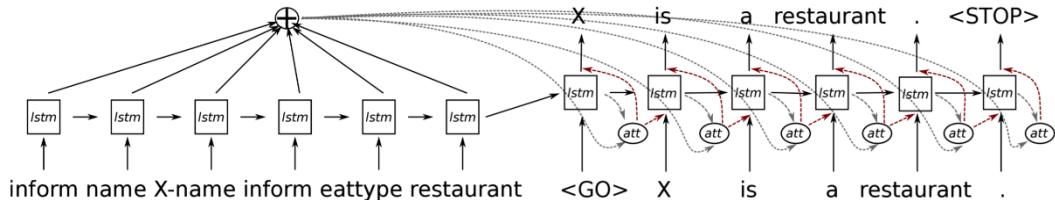
inform(name=X-name,type=placetoeat,eattype=restaurant,  
area=riverside,food=Italian)



( <root> <root> ( ( X-name n:subj ) be v:fin ( ( Italian adj:attr ) restaurant n:obj ( river n:near+X ) ) )  
X-name n:subj be v:fin Italian adj:attr restaurant n:obj river n:near+X



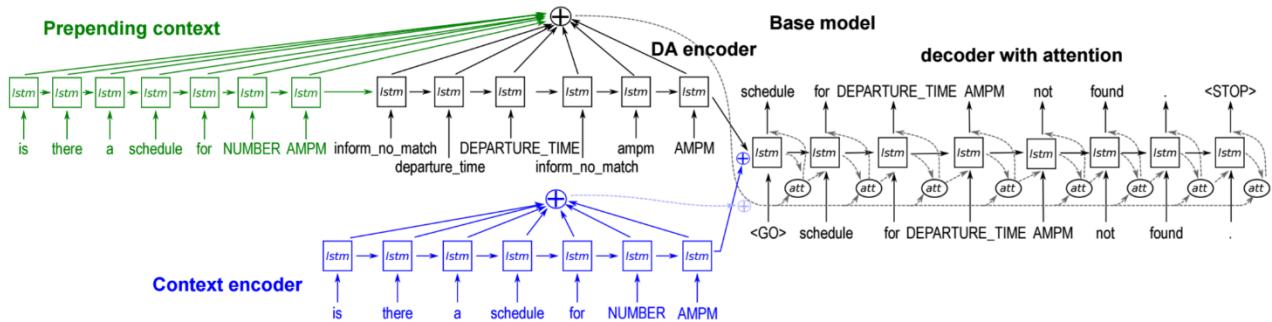
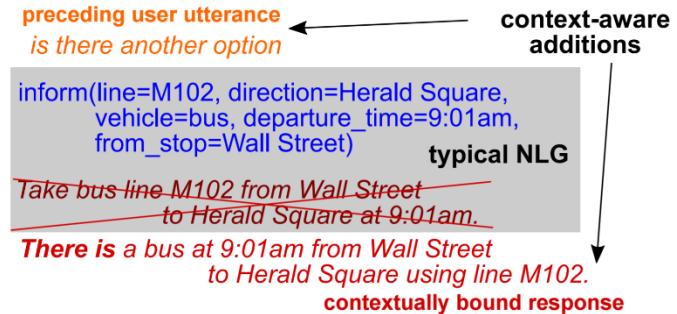
*X is an Italian restaurant near the river.*



# Contextual NLG (Dušek and Jurčíček, 2016)

<https://www.aclweb.org/anthology/W/W16/W16-36.pdf#page=203>

- Goal: adapting users' way of speaking, providing context-aware responses
  - Context encoder
  - Seq2Seq model



# NLG Evaluation

87

## □ Metrics

### □ Subjective: human judgement (Stent et al., 2005)

- Adequacy: correct meaning
- Fluency: linguistic fluency
- Readability: fluency in the dialogue context
- Variation: multiple realizations for the same concept

### □ Objective: automatic metrics

- Word overlap: BLEU (Papineni et al, 2002), METEOR, ROUGE
- Word embedding based: vector extrema, greedy matching, embedding average

There is a gap between human perception and automatic metrics

# Outline

88

- Introduction and Background
  - Neural Networks
  - Reinforcement Learning
- Deep Learning Based Dialogue System
  - Spoken/Natural Language Understanding (SLU/NLU)
  - Dialogue State Tracking (DST)
  - Dialogue Policy
  - Natural Language Generation (NLG)
  - **End-to-End Learning for Dialogue Systems**
- Evaluation
- Recent Trends on Learning Dialogues
- Challenges
- Conclusion

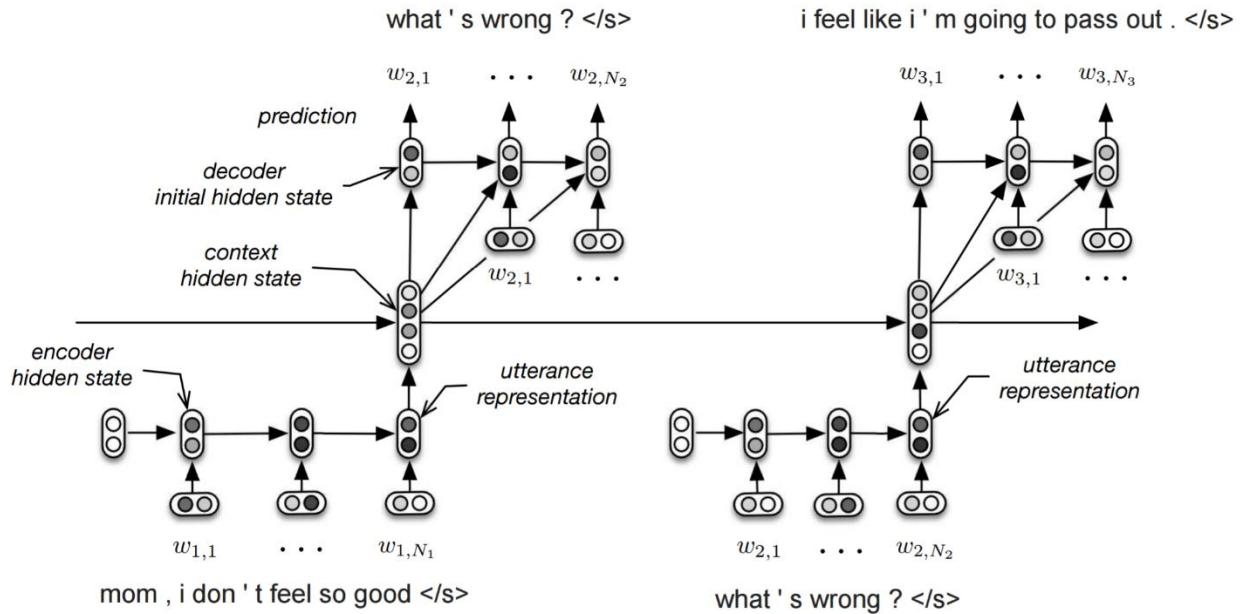
# ChitChat Hierarchical Seq2Seq

(Serban et.al., 2016)

89

<http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/11957>

- A hierarchical seq2seq model for generating dialogues



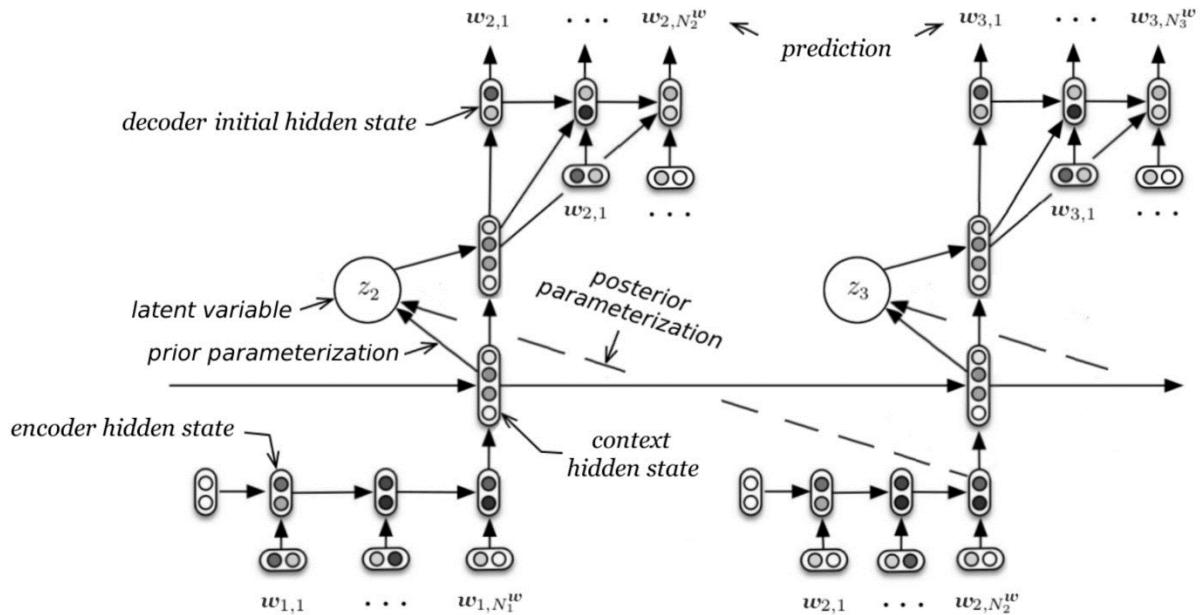
# ChitChat Hierarchical Seq2Seq

(Serban et.al., 2017)

90

<https://arxiv.org/abs/1605.06069>

- A hierarchical seq2seq model with **Gaussian latent variable** for generating dialogues



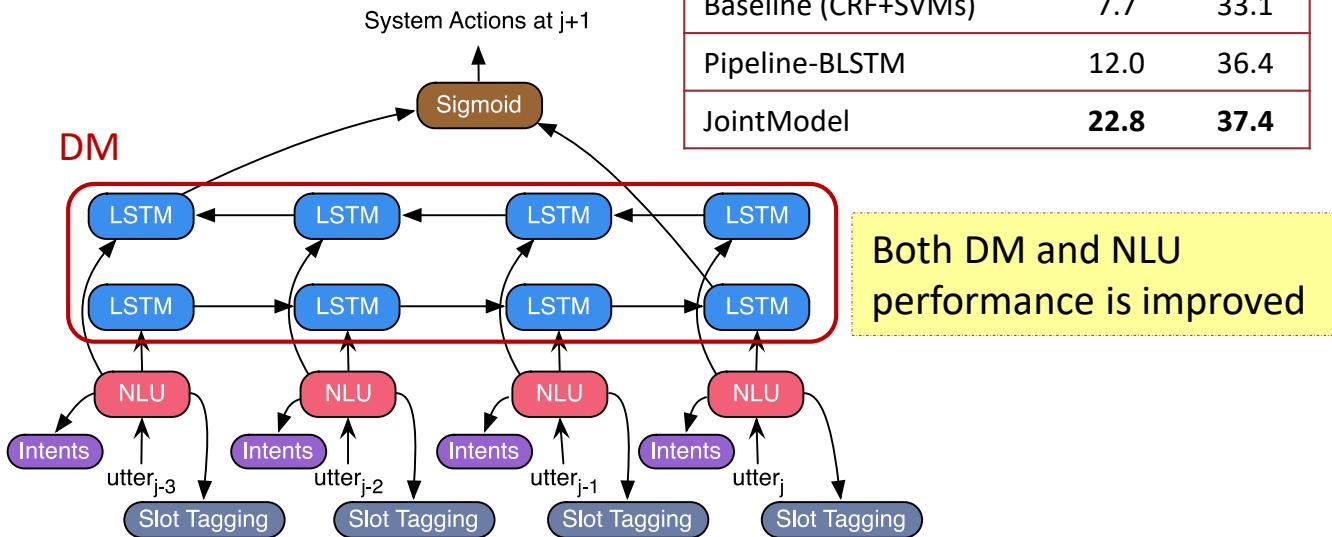
# E2E Joint NLU and DM (Yang et al., 2017)

91

<https://arxiv.org/abs/1612.00913>

- Idea: errors from DM can be propagated to NLU for better robustness

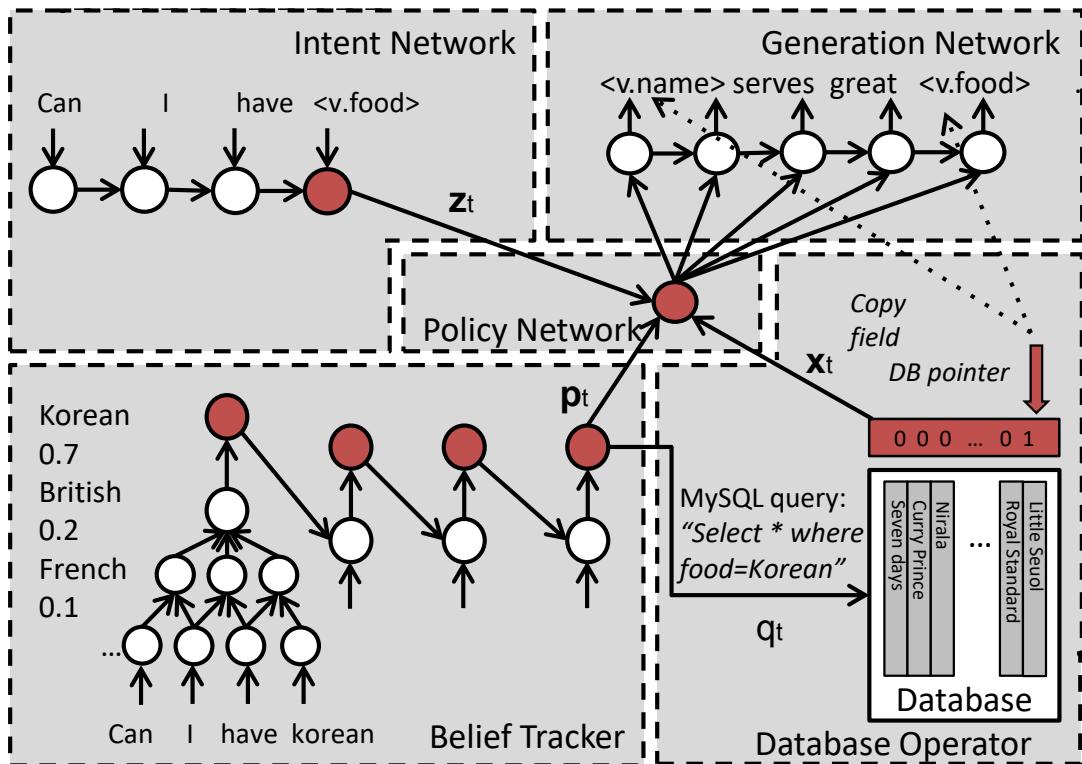
Model	DM	NLU
Baseline (CRF+SVMs)	7.7	33.1
Pipeline-BLSTM	12.0	36.4
JointModel	<b>22.8</b>	<b>37.4</b>



# E2E Supervised Dialogue System (Wen et al., 2016)

92

<https://arxiv.org/abs/1604.04562>

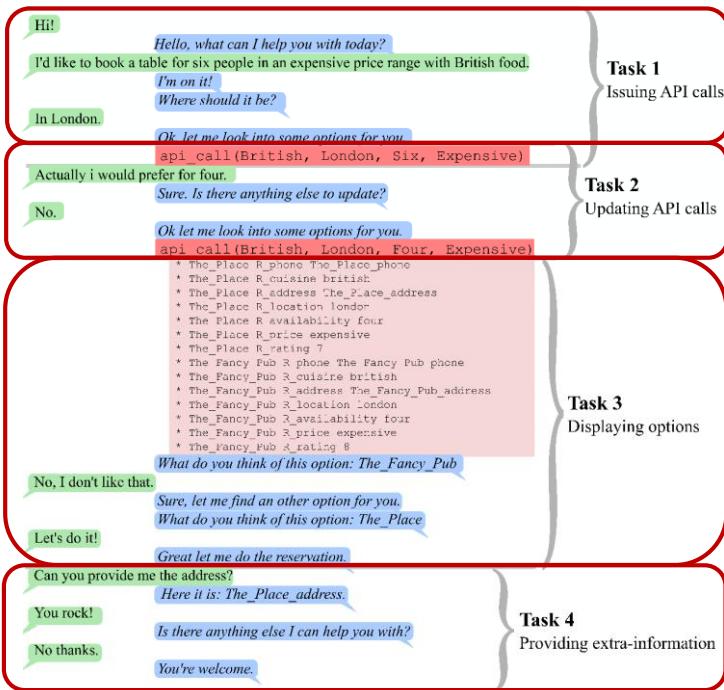


# E2E MemNN for Dialogues (Bordes et al., 2016)

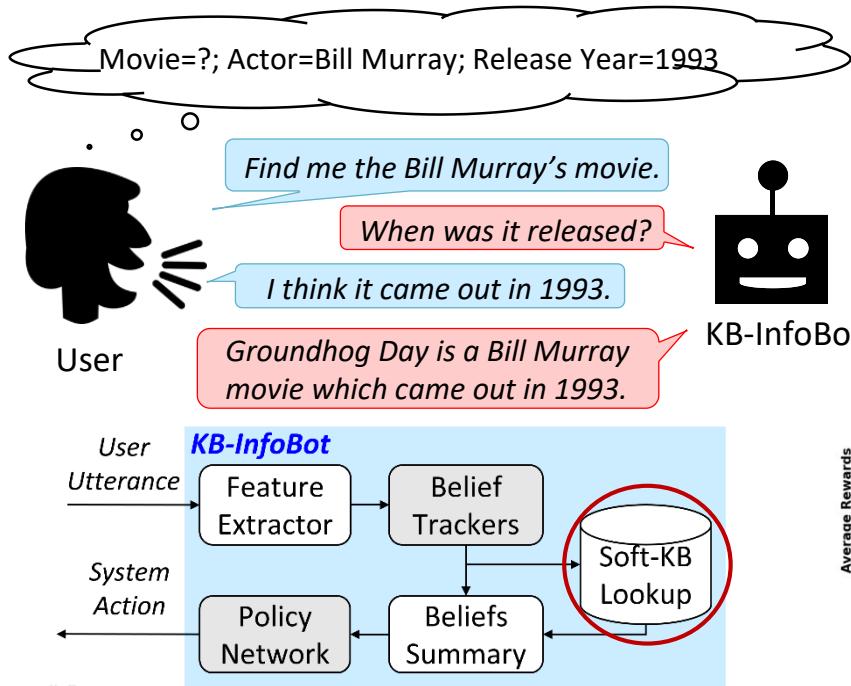
<https://arxiv.org/abs/1605.07683>

- Split dialogue system actions into subtasks
  - API issuing
  - API updating
  - Option displaying
  - Information informing

Task	Memory Networks	
	no match type	+ match type
T1: Issuing API calls	<b>99.9</b> (99.6)	<b>100</b> (100)
T2: Updating API calls	<b>100</b> (100)	98.3 (83.9)
T3: Displaying options	<b>74.9</b> (2.0)	<b>74.9</b> (0)
T4: Providing information	59.5 (3.0)	<b>100</b> (100)
T5: Full dialogs	<b>96.1</b> (49.4)	93.4 (19.7)
T1(OOV): Issuing API calls	72.3 (0)	<b>96.5</b> (82.7)
T2(OOV): Updating API calls	78.9 (0)	<b>94.5</b> (48.4)
T3(OOV): Displaying options	74.4 (0)	<b>75.2</b> (0)
T4(OOV): Providing inform.	57.6 (0)	<b>100</b> (100)
T5(OOV): Full dialogs	65.5 (0)	<b>77.7</b> (0)
T6: Dialog state tracking 2	<b>41.1</b> (0)	<b>41.0</b> (0)

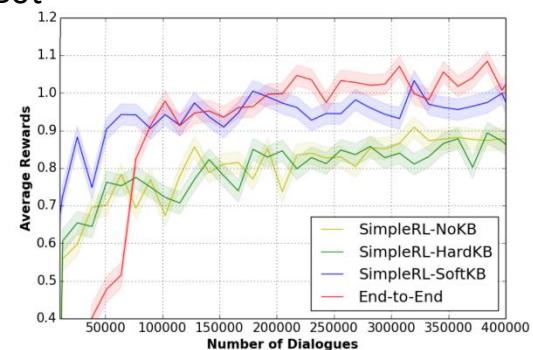


# E2E RL-Based Info-Bot (Dhingra et al., 2016)



**Knowledge Base (*head, relation, tail*)**

(*Groundhog Day, actor, Bill Murray*)  
 (*Groundhog Day, release year, 1993*)  
 (*Australia, actor, Nicole Kidman*)  
 (*Mad Max: Fury Road, release year, 2015*)



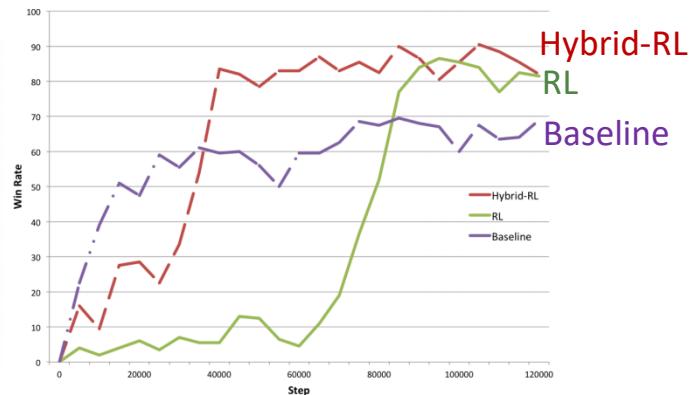
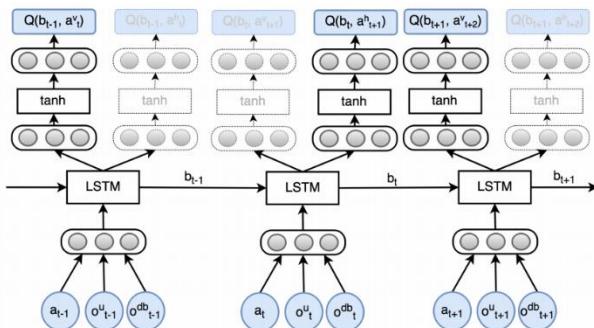
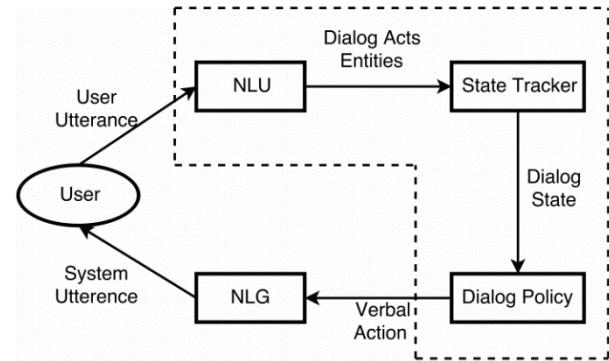
Idea: differentiable database for propagating the gradients

# E2E RL-Based System (Zhao and Eskenazi, 2016)

95

<http://www.aclweb.org/anthology/W/W16/W16-36.pdf#page=19>

- Joint learning
  - NLU, DST, Dialogue Policy
- Deep RL for training
  - Deep Q-network
  - Deep recurrent network



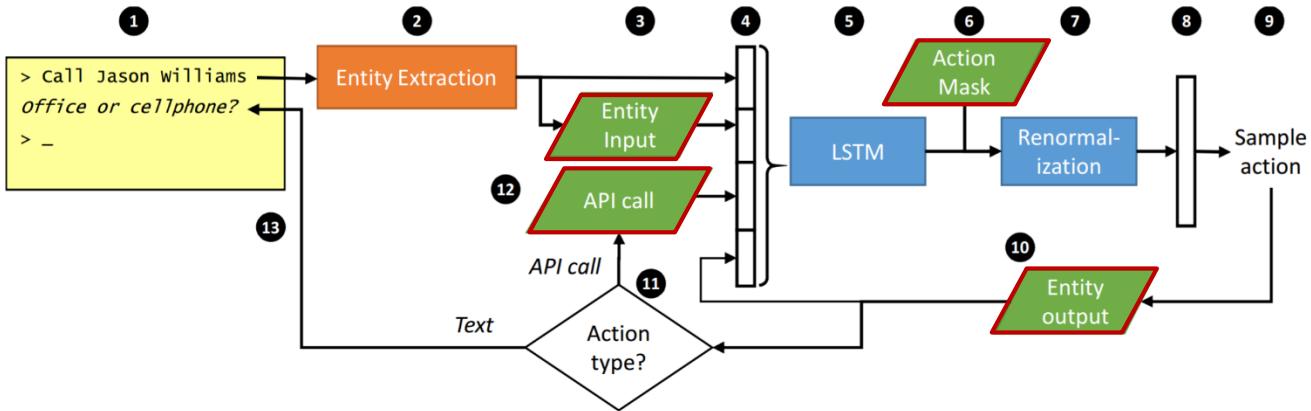
# E2E LSTM-Based Dialogue Control

(Williams and Zweig, 2016)

96

<https://arxiv.org/abs/1606.01269>

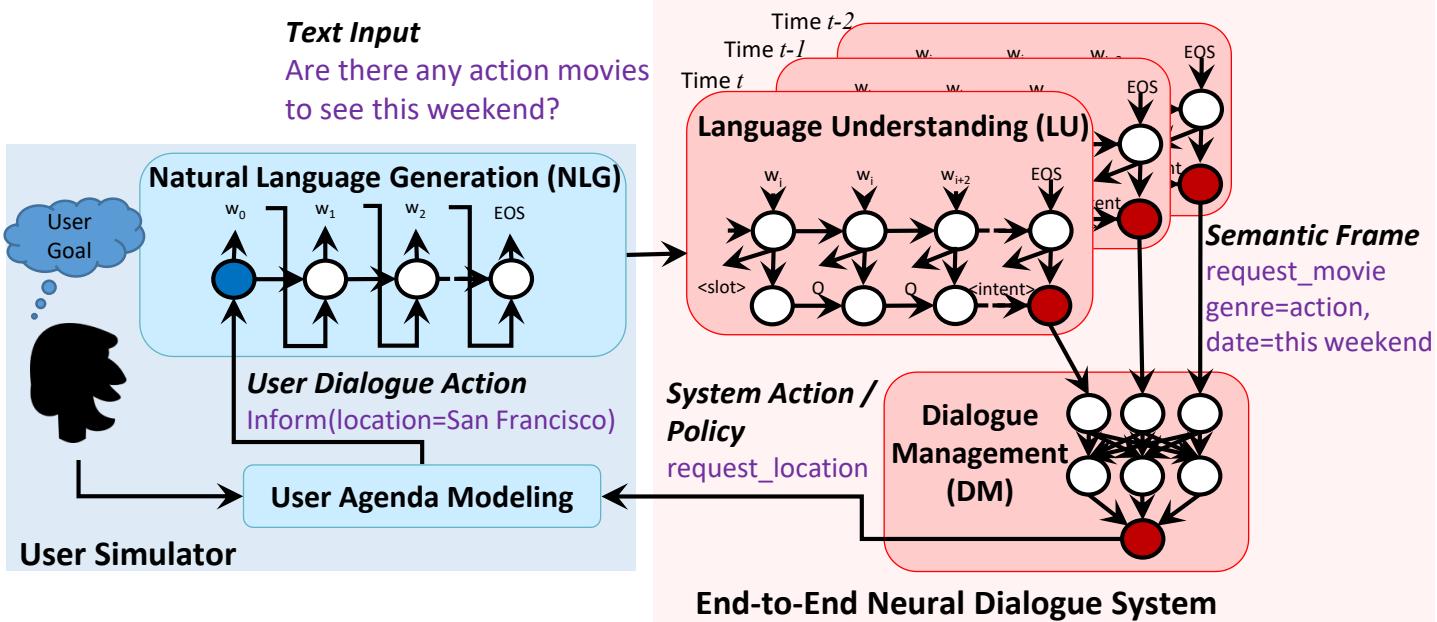
- Idea: an LSTM maps from raw dialogue history directly to a distribution over system actions
  - Developers can provide software including business rules & programmatic APIs  
→ LSTM can take actions in the real world on behalf of the user
  - The LSTM can be optimized using SL or RL



# E2E Task-Completion Bot (TC-Bot) (Li et al., 2017)

97

<https://arxiv.org/abs/1703.01008>



Idea: supervised learning for each component and reinforcement learning for end-to-end training the neural dialogue system

# E2E Task-Completion Bot (TC-Bot) (Li et al., 2017)

98

<https://arxiv.org/abs/1703.01008>

- User goal: ***Two tickets for “the witch” tomorrow 9:30 PM at regal meridian 16, Seattle.***

## RULE BASED SYSTEM

User: Can I get 2 tickets for the witch?

Agent: Which theater do you want?

User: Regal meridian 16

Agent: What time would you like to see it?

User: 9:30 pm

Agent: What date would you like to see it?

User: Tomorrow!

Agent: How many people are you?

User: Just two adults.

Agent: What is your location?

User: Seattle!

Agent: What kind of video format do you like?

User: Oh, any kind is okay.

Agent: Do you have any price constraint?

User: No, I don't care.

Agent: Great - I was able to purchase 2 tickets for you to see the witch tomorrow at regal meridian 16 theater in Seattle at 9:30 pm.

User: Thank you.

Agent: Thank you!

## REINFORCEMENT LEARNING SYSTEM

User: Can I get 2 tickets for the witch?

Agent: What date would you like to see it?

User: Tomorrow

Agent: Which theater do you prefer?

User: Regal meridian 16

Agent: What is your location?

User: Seattle

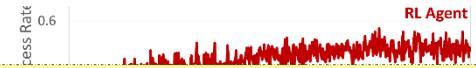
Agent: What time would you like to see it?

User: 9:30 pm.

Agent: Great - I was able to purchase 2 tickets for you to see the witch tomorrow at regal meridian 16 theater in Seattle at 9:30 pm.

User: Thanks.

Agent: Thanks!



The system can learn how to efficiently interact with users for task completion

0 100 200 300 400 500 600  
Simulation Epoch

## Part III

### Evaluation

# Outline

100

- Introduction and Background
  - Neural Networks
  - Reinforcement Learning
- Deep Learning Based Dialogue System
  - Spoken/Natural Language Understanding (SLU/NLU)
  - Dialogue State Tracking (DST)
  - Dialogue Policy
  - Natural Language Generation (NLG)
  - End-to-End Learning for Dialogue Systems
- **Evaluation**
- Recent Trends on Learning Dialogues
- Challenges
- Conclusion

# Dialogue System Evaluation

101

- Dialogue model evaluation
  - Crowd sourcing
  - User simulator
- Response generator evaluation
  - Word overlap metrics
  - Embedding based metrics

# Crowd Sourcing for Dialog System Evaluation (Yang, et.al. 2012)

102

[http://www-scf.usc.edu/~zhaojuny/docs/SDSchapter\\_final.pdf](http://www-scf.usc.edu/~zhaojuny/docs/SDSchapter_final.pdf)

## Q1 Do you think you understand from the dialog what the user wanted?

- Opt 1) No clue 2) A little bit 3) Somewhat  
4) Mostly 5) Entirely

Aim elicit the Worker's confidence in his/her ratings.

## Q2 Do you think the system is successful in providing the information that the user wanted?

- Opt 1) Entirely unsuccessful 2) Mostly unsuccessful  
3) Half successful/unsuccessful  
4) Mostly successful 5) Entirely successful

Aim elicit the Worker's perception of whether the dialog has fulfilled the informational goal of the user.

## Q3 Does the system work the way you expect it?

- Opt 1) Not at all 2) Barely 3) Somewhat  
4) Almost 5) Completely

Aim elicit the Worker's impression of whether the dialog flow suits general expectations.

## Q4 Overall, do you think that this is a good system?

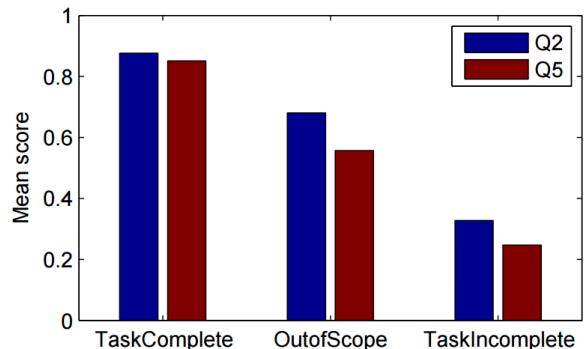
- Opt 1) Very poor 2) Poor 3) Fair 4) Good 5) Very good

Aim elicit the Worker's overall impression of the SDS.

## Q5 What category do you think the dialog belongs to?

- Opt 1) Task is incomplete 2) Out of scope  
3) Task is complete

Aim elicit the Worker's impression of whether the dialog reflects task completion.

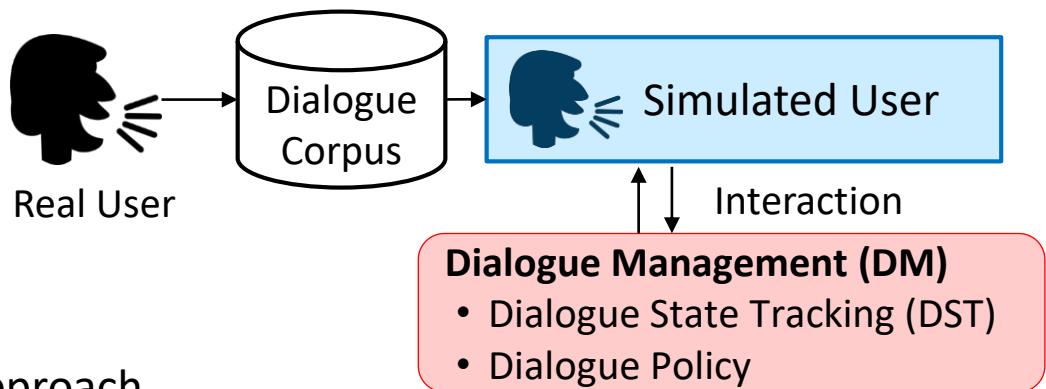


The normalized mean scores of Q2 and Q5 for approved ratings in each category. A higher score maps to a higher level of task success

# User Simulation

103

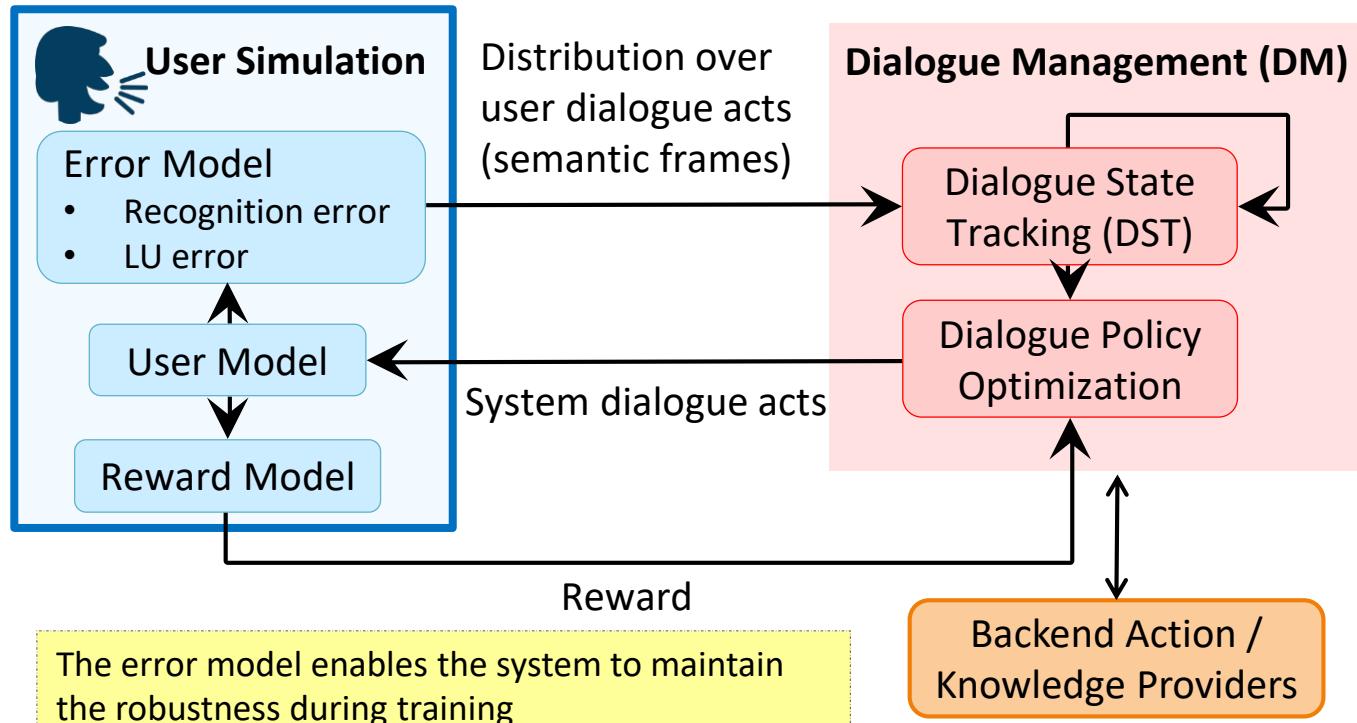
- Goal: generate natural and reasonable conversations to enable reinforcement learning for exploring the policy space



- Approach
  - Rule-based crafted by experts (Li et al., 2016)
  - Learning-based (Schatzmann et al., 2006; El Asri et al., 2016)

# Elements of User Simulation

104



# Rule-Based Simulator for RL Based System

(Li et.al., 2016)

105

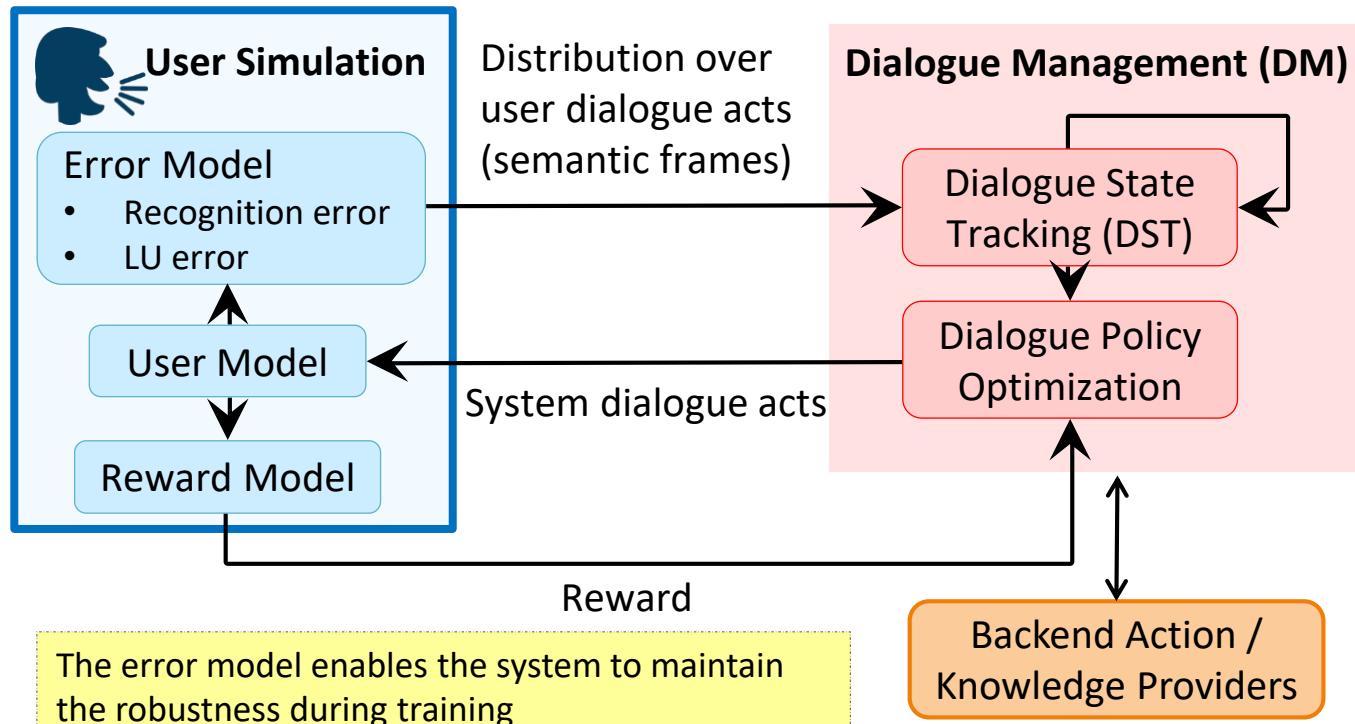
<http://arxiv.org/abs/1612.05688>

- rule-based simulator + collected data
- starts with sets of goals, actions, KB, slot types
- publicly available simulation framework
- movie-booking domain: ticket booking and movie seeking
- provide procedures to add and test own agent

```
1 class AgentDQN(Agent):  
2     def run_policy(self, representation):  
3         """ epsilon-greedy policy """  
4  
5         if random.random() < self.epsilon:  
6             return random.randint(0, self.num_actions - 1)  
7         else:  
8             if self.warm_start == 1:  
9                 if len(self.experience_replay_pool) > self.experience_replay_pool_size:  
10                     self.warm_start = 2  
11                     return self.rule_policy()  
12                 else:  
13                     return self.dqn.predict(representation, {}, predict_model=True)  
14  
15     def train(self, batch_size=1, num_batches=100):  
16         """ Train DQN with experience replay """  
17  
18         for iter_batch in range(num_batches):  
19             self.cur_bellman_err = 0  
20             for iter in range(len(self.experience_replay_pool)/(batch_size)):  
21                 batch = [random.choice(self.experience_replay_pool) for i in xrange(batch_size)]  
22                 batch_struct = self.dqn.singleBatch(batch, {'gamma': self.gamma}, self.clone_dqn)
```

# Elements of User Simulation

106



# Rule-Based Simulator for RL Based System

(Li et.al., 2016)

107

<http://arxiv.org/abs/1612.05688>

- Rule-based simulator + collected data
- Starts with sets of goals, actions, KB, slot types.
- Presents publicly available simulation framework, for the movie-booking domain: movie ticket booking and movie seeking.
- provide procedures to add and test own agent in their proposed framework

# Data-Driven Simulator for Automated Evaluation (Jung et.al., 2009)

108

- Three step process
  - 1) User intention simulator

request+search\_loc

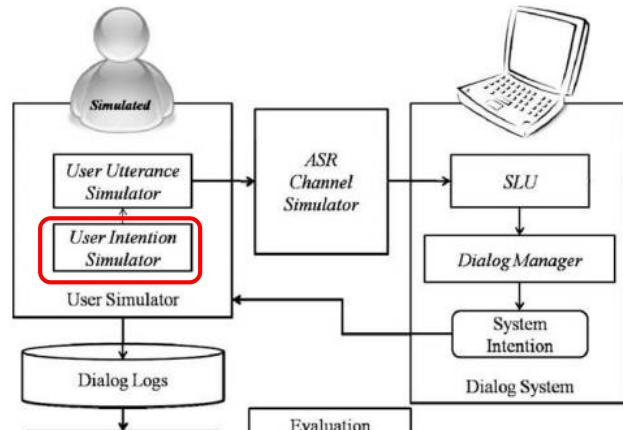
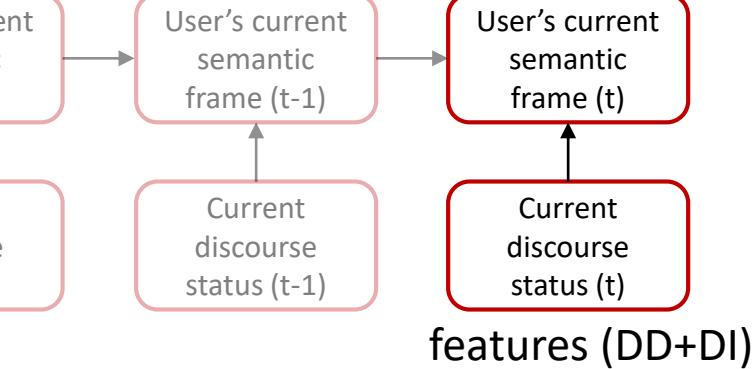


Fig. 1. Overall architecture of dialog simulation.

(\* ) compute all possible semantic frame given previous turn info  
(\*) randomly select one possible semantic frame

# Data-Driven Simulator for Automated Evaluation (Jung et.al., 2009)

109

- Three step process
  - 1) User intention simulator
  - 2) User utterance simulator

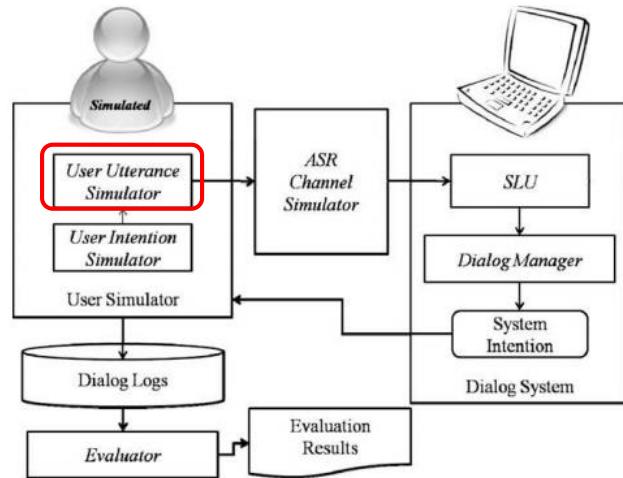


Fig. 1. Overall architecture of dialog simulation.

request+search\_loc

I want to go to the city hall

PRP VB TO VB TO [loc\_name]

Given a list of POS tags associated with the semantic frame, using LM+Rules they generate the user utterance.

# Data-Driven Simulator for Automated Evaluation (Jung et.al., 2009)

110

- Three step process:
  - 1) User intention simulator
  - 2) User utterance simulator
  - 3) ASR channel simulator
- Evaluate the generated sentences using BLUE-like measures against the reference utterances collected from humans (with the same goal)

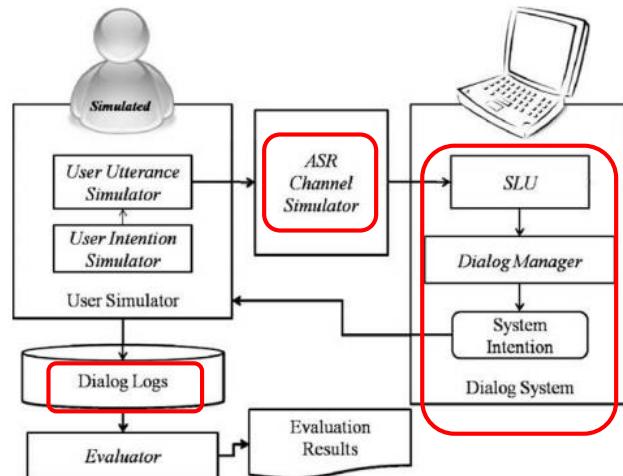


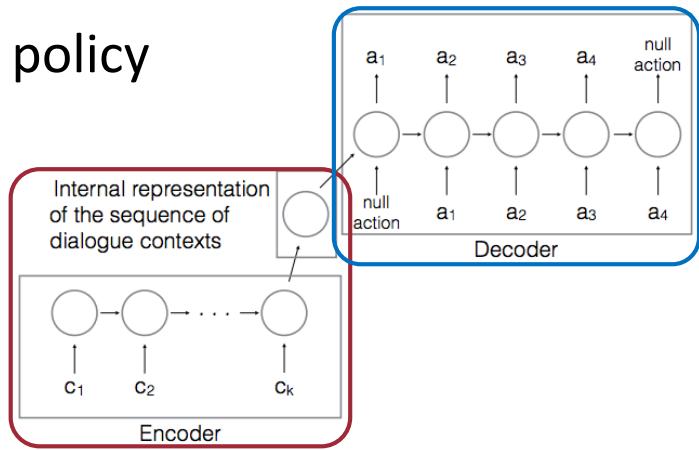
Fig. 1. Overall architecture of dialog simulation.

# Seq2Seq User Simulation (El Asri et al., 2016)

111

<https://arxiv.org/abs/1607.00070>

- Seq2Seq trained from dialogue data
  - Input:  $c_i$  encodes contextual features, such as the previous system action, consistency between user goal and machine provided values
  - Output: a dialogue act sequence form the user
- Extrinsic evaluation for policy



# User Simulator for Dialogue Evaluation Measures

112

## Understanding Ability

- whether constrained values specified by users can be understood by the system
- agreement percentage of system/user understandings over the entire dialog (averaging all turns)

## Efficiency

- Number of dialogue turns
- Ratio between the dialogue turns (larger is better)

## Action Appropriateness

- an explicit confirmation for an uncertain user utterance is an appropriate system action
- providing information based on misunderstood user requirements

# How not to evaluate your dialog system

(Liu et.al., 2017)

113

<https://arxiv.org/pdf/1603.08023.pdf>

- How to evaluate the quality of the generated response ?
  - Specifically investigated for chat-bots
  - Crucial for task-oriented tasks as well
- Metrics:
  - Word overlap metrics, e.g., BLEU, METEOR, ROUGE, etc.
  - Embeddings based metrics, e.g., contextual/meaning representation between target and candidate



# Dialog Response Evaluation (Lowe et al., 2017)

114

- Problems of existing automatic evaluation
  - can be biased
  - correlate poorly with human judgements of response quality
  - using word overlap may be misleading
- Solution
  - collect a **dataset of accurate human scores** for variety of dialogue responses (e.g., coherent/un-coherent, relevant/irrelevant, etc.)
  - use this dataset to train an **automatic dialogue evaluation model** – learn to compare **the reference to candidate responses!**
  - Use RNN to predict scores by comparing against human scores!

## Context of Conversation

**Speaker A:** Hey, what do you want to do tonight?

**Speaker B:** Why don't we go see a movie?

## Model Response

*Nah, let's do something active.*

## Reference Response

*Yeah, the film about Turing looks great!*

## Part IV

### Recent Trends on Learning Dialogues

# Outline

116

- Introduction and Background
  - Neural Networks
  - Reinforcement Learning
- Deep Learning Based Dialogue System
  - Spoken/Natural Language Understanding (SLU/NLU)
  - Dialogue State Tracking (DST)
  - Dialogue Policy
  - Natural Language Generation (NLG)
  - End-to-End Learning for Dialogue Systems
- Evaluation
- **Recent Trends on Learning Dialogues**
- Challenges
- Conclusion

# Dialog State Tracking Challenge (DSTC)

(Williams et al. 2013, Henderson et al. 2014, Henderson et al. 2014, Kim et al. 2016, Kim et al. 2016)

117

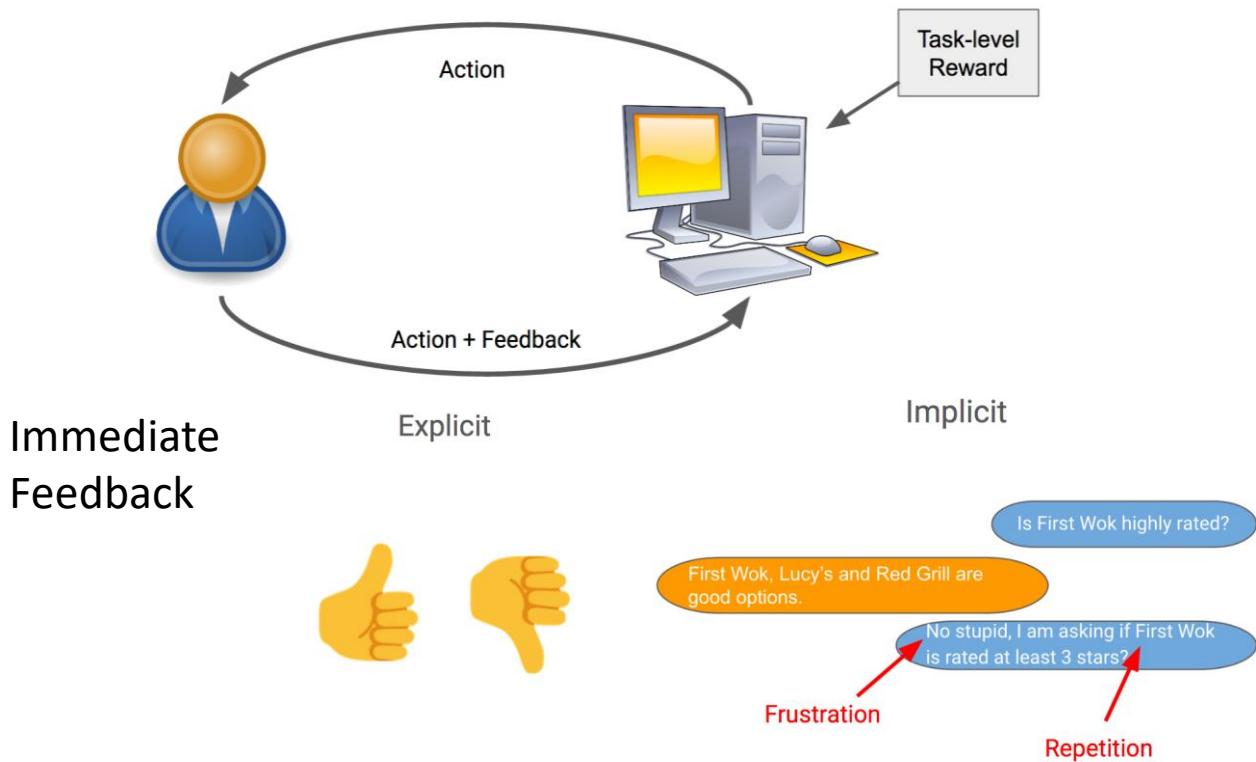
Challenge	Type	Domain	Data Provider	Main Theme
DSTC1	Human-Machine	Bus Route	CMU	Evaluation Metrics
DSTC2	Human-Machine	Restaurant	U. Cambridge	User Goal Changes
DSTC3	Human-Machine	Tourist Information	U. Cambridge	Domain Adaptation
DSTC4	Human-Human	Tourist Information	I2R	Human Conversation
DSTC5	Human-Human	Tourist Information	I2R	Language Adaptation
DSTC6				

DSTC renames as **Dialog System Technology Challenges**

# Interactive RL for DM (Shah et al., 2016)

118

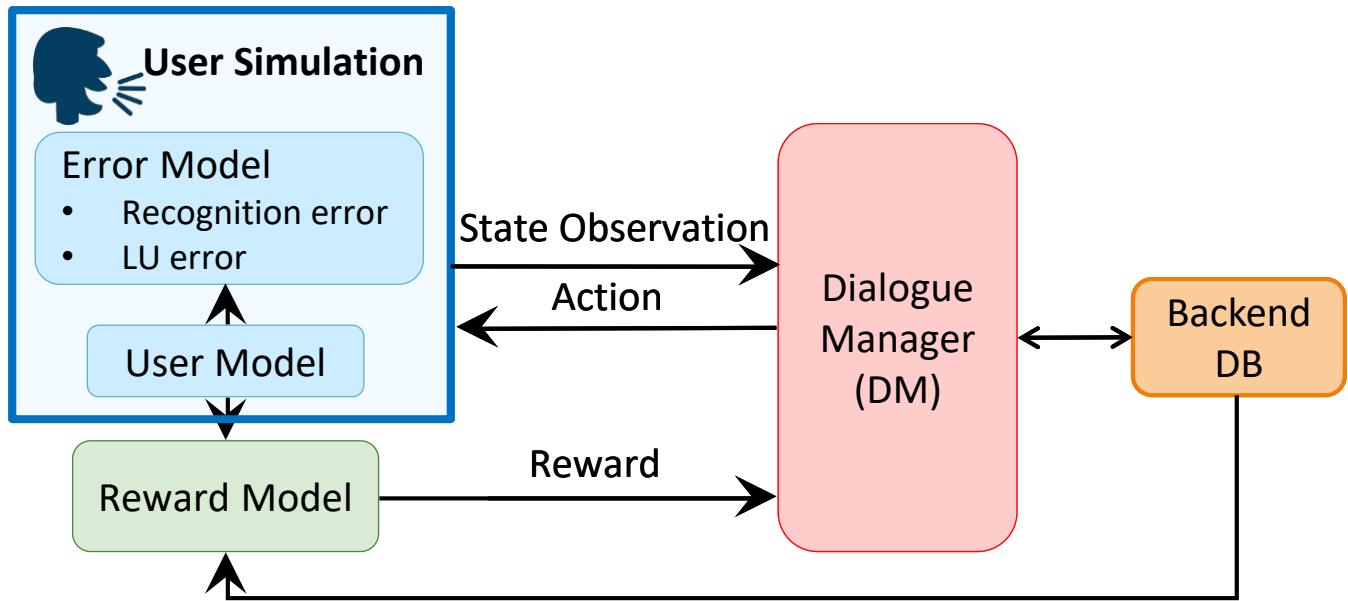
<https://research.google.com/pubs/pub45734.html>



# Interactive RL for DM (Shah et al., 2016)

119

<https://research.google.com/pubs/pub45734.html>



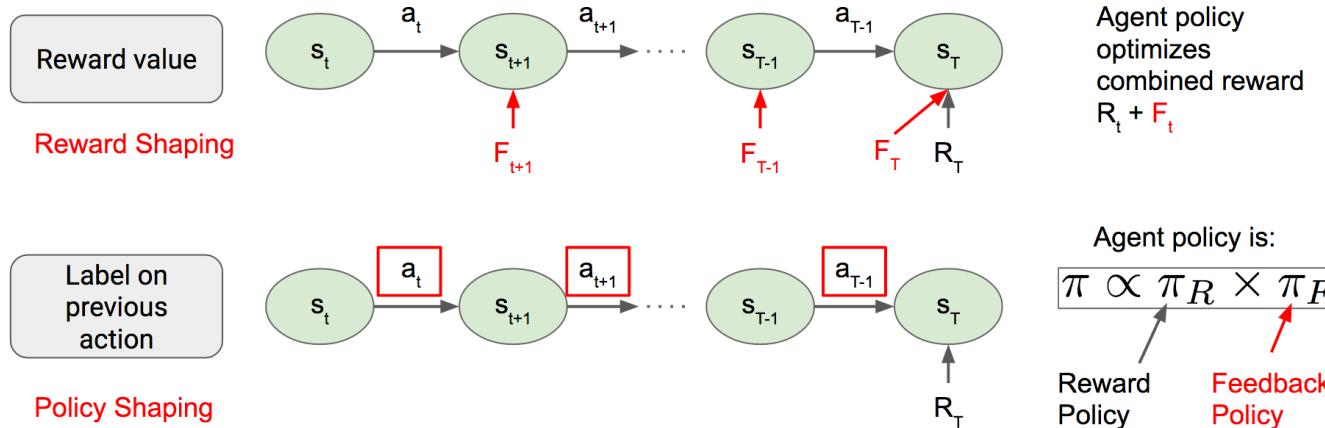
Use a third agent for providing interactive feedback to the DM

# Interpreting Interactive Feedback

(Shah et al., 2016)

120

<https://research.google.com/pubs/pub45734.html>



# Policy Shaping for RL (Shah et al., 2016)

121

<https://research.google.com/pubs/pub45734.html>

Feedback label

$$f_{(s,a),t} \in \{-1, +1\}$$

Feedback delta for  
action a in state s

$$\delta_{s,a} = \sum_t f_{(s,a),t}$$

Estimate of feedback  
consistency

Estimate of optimality  
of action a in state s

$$P_F(a|s) = \frac{C^{\delta_{s,a}}}{C^{\delta_{s,a}} + (1 - C)^{\delta_{s,a}}}$$

C=0.95

Feedback policy

$$\pi_F(s, a) = P_F(a|s)$$

Overall policy

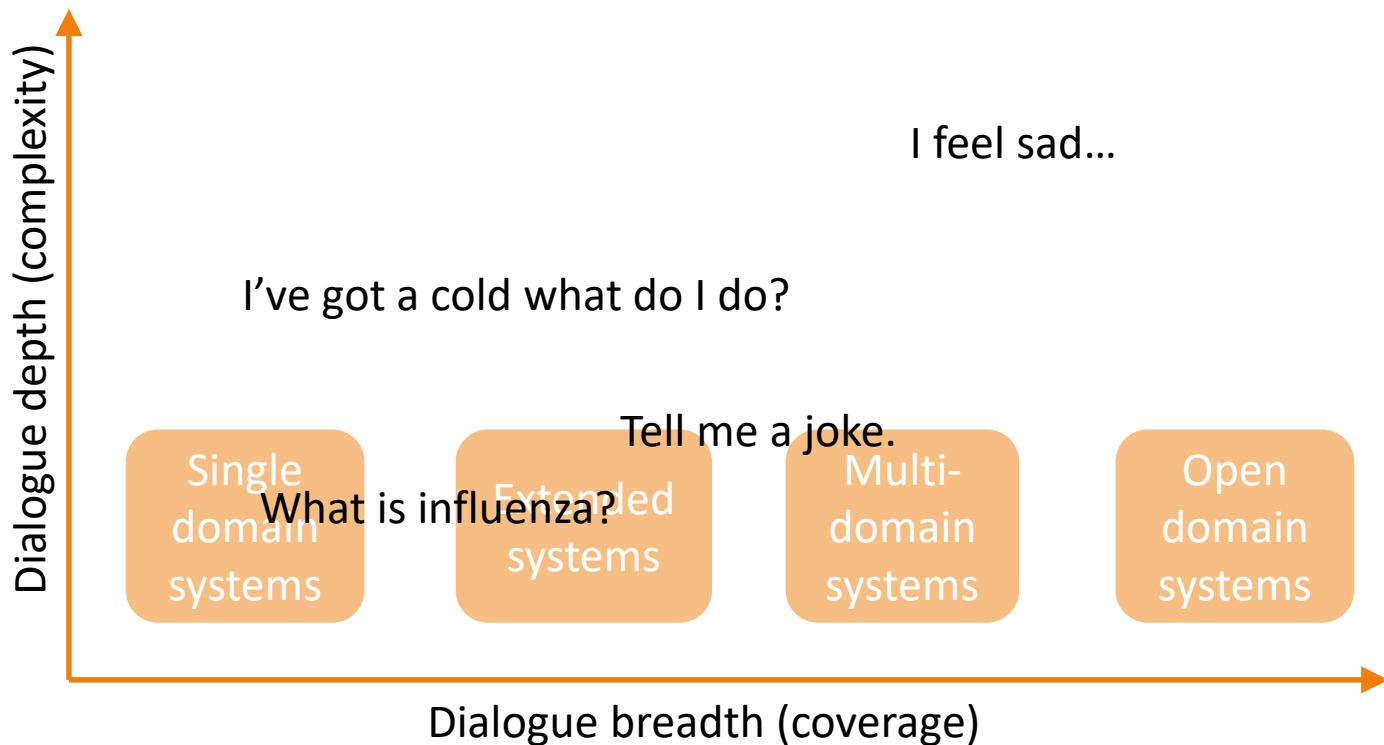
$$\pi \propto \pi_R \times \pi_F$$

$\delta_{s,a}$	$P_F(a s)$
-3	0.000145
0	0.5
+3	0.999854

Griffith, S., Subramanian, K., Scholz, J., Isbell, C., and Thomaz, A. L. (2013). Policy shaping: Integrating human feedback with reinforcement learning. In Advances in Neural Information Processing Systems, pages 2625–2633.

# Evolution Roadmap

122

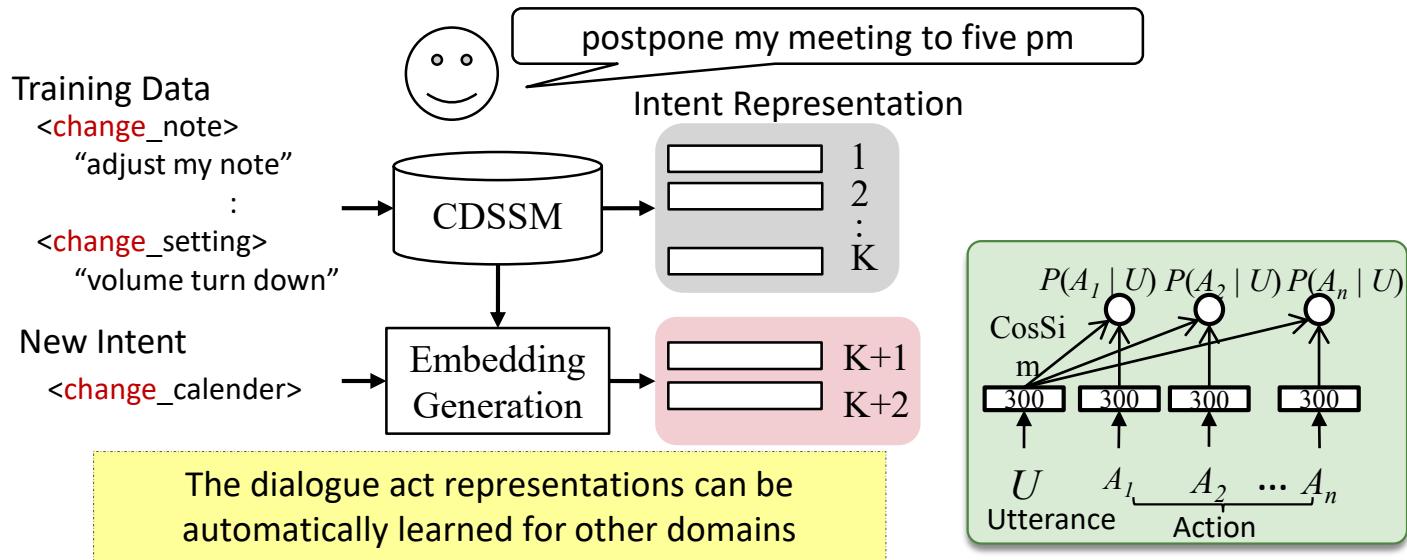


# Intent Expansion (Chen et al., 2016)

123

<http://ieeexplore.ieee.org/abstract/document/7472838>

- Transfer dialogue acts across domains
  - Dialogue acts are similar for multiple domains
  - Learning new intents by information from other domains

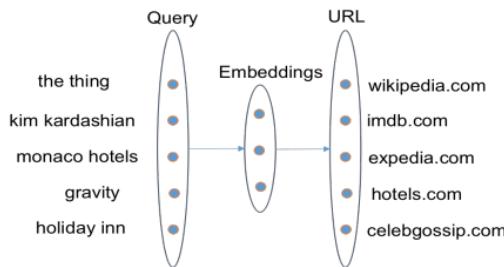


# Zero-Shot Learning (Daupin et al., 2016)

124

<https://arxiv.org/abs/1401.0509>

- Semantic utterance classification
  - ▣ Use query click logs to define a task that makes the networks learn the meaning or intent behind the queries



$$\mathcal{L}(X, Y) = -\log P(Y|X) + \lambda H(P(C|X)).$$

Depiction of the deep network from queries to URLs.

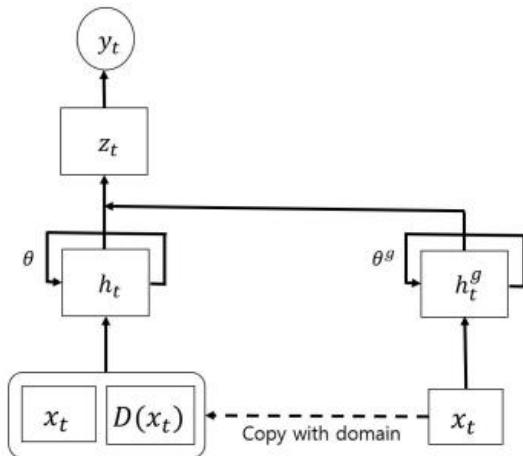
- The semantic features are the last hidden layer of the DNN
- Use Zero-Shot Discriminative embedding model combines  $H$  with the minimization of entropy of a zero-shot classifier

# Domain Adaptation for SLU (Kim et al., 2016)

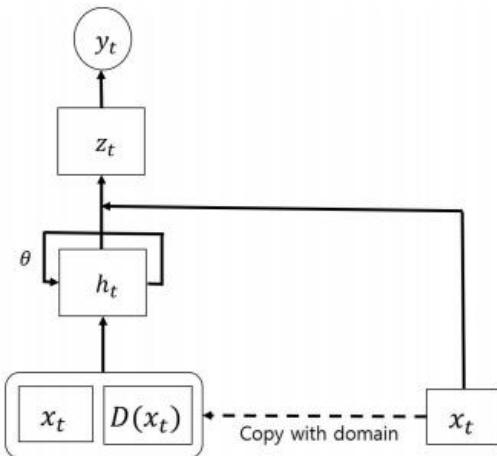
125

<http://www.aclweb.org/anthology/C/C16/C16-1038.pdf>

- Frustratingly easy domain adaptation
- Novel neural approaches to domain adaptation
- Improve slot tagging on several domains



(a) 1 domain specific LSTM + generic LSTM



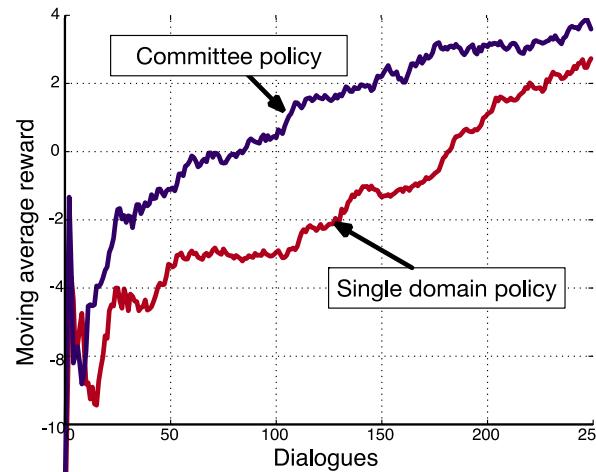
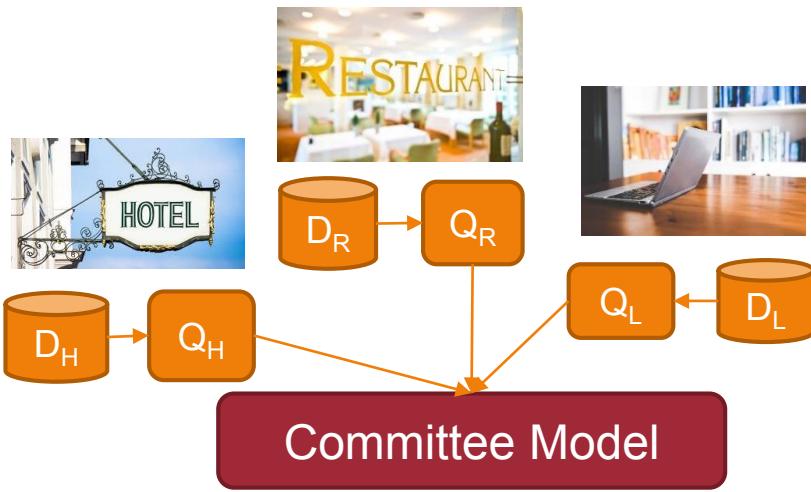
(b) 1 domain specific LSTM + generic embedding

# Policy for Domain Adaptation (Gašić et al., 2015)

126

<http://ieeexplore.ieee.org/abstract/document/7404871>

- Bayesian committee machine (BCM) enables estimated Q-function to share knowledge across domains



The policy from a new domain can be boosted by the committee policy

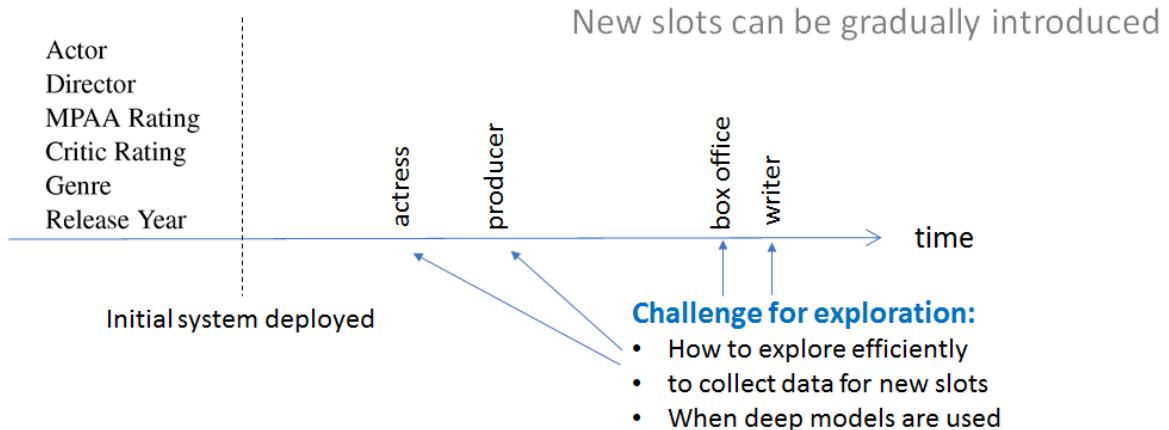
# Efficient Exploration for Domain Expansion

(Lipton et al., 2016)

127

<http://arxiv.org/abs/1608.05081>

- Goal : dialogue domain extension
- Most goal-oriented dialogues require a closed and well-defined domain
- Hard to include all domain-specific information up-front



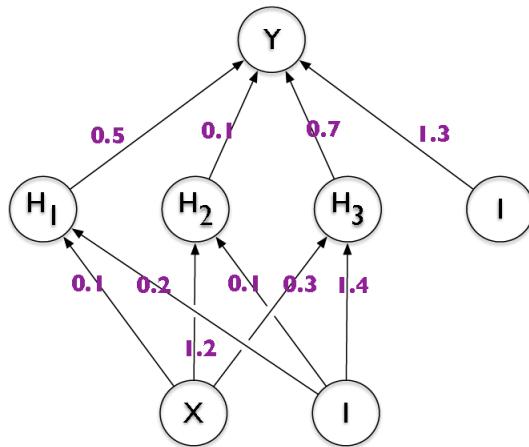
# Efficient Exploration for Domain Expansion

(Lipton et al., 2016)

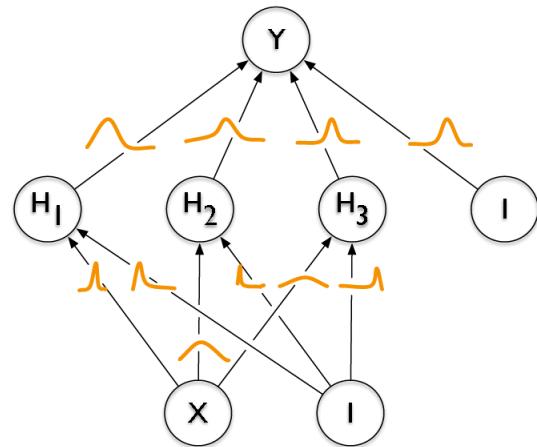
128

<http://arxiv.org/abs/1608.05081>

- Bayesian by back-propagation



Maintain point-  
estimates of weights



Maintain posterior  
distribution of weights

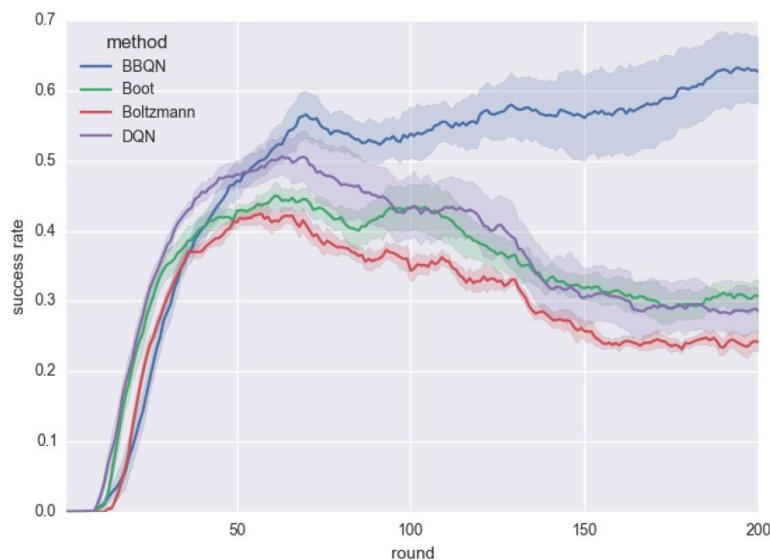
# Efficient Exploration for Domain Expansion

(Lipton et al., 2016)

129

<http://arxiv.org/abs/1608.05081>

## □ Bayes by Backprop Q-Network (BBQ)



Weight posteriors are maintained

- Combine RL and Bayes-by-BP
- Use variational inference to scale up

Thompson sampling for exploration  
[a.k.a. “posterior sampling”]

Efficient exploration accelerates policy optimization

# Evolution Roadmap

130

Dialogue depth (complexity)

Empathetic systems

I feel sad...

I've got a cold what do I do?

Common sense system

Tell me a joke.

What is influenza?

Knowledge based system

Dialogue breadth (coverage)

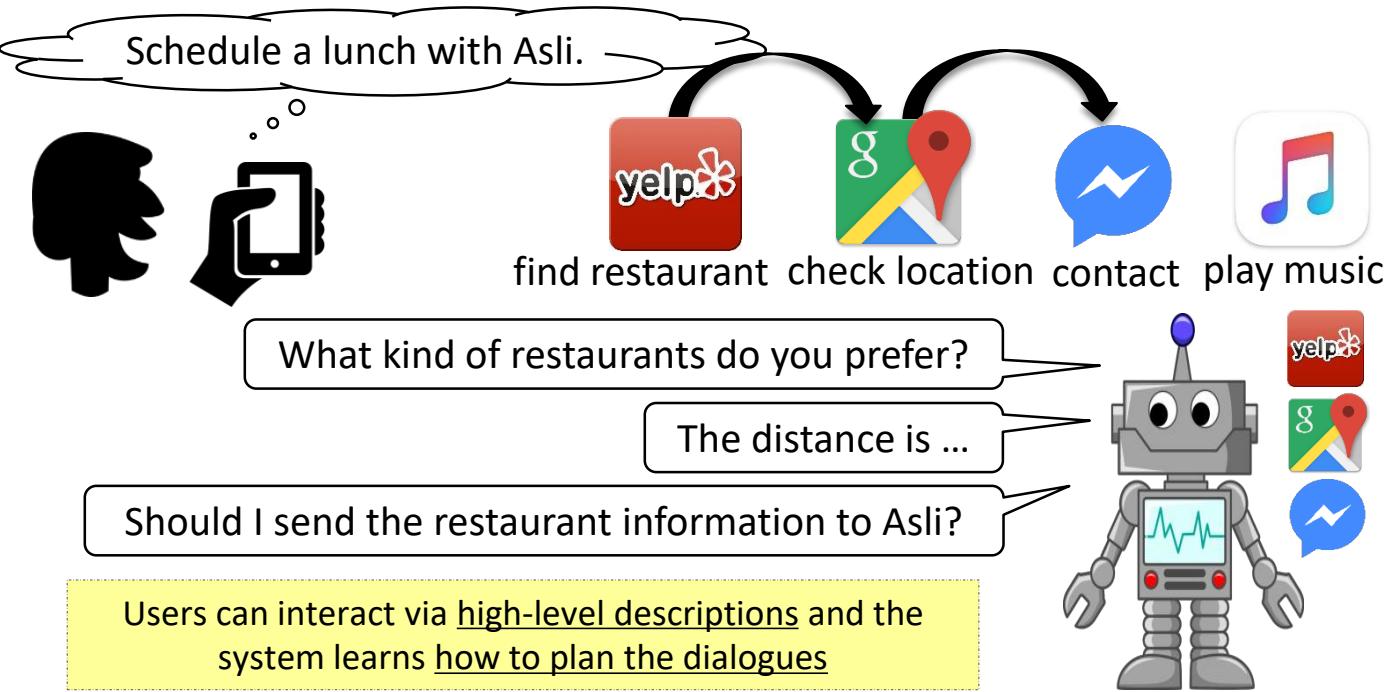
# High-Level Intention for Dialogue Planning

(Sun et al., 2016; Sun et al., 2016)

131

<http://dl.acm.org/citation.cfm?id=2856818>; [http://www.irec-conf.org/proceedings/irec2016/pdf/75\\_Paper.pdf](http://www.irec-conf.org/proceedings/irec2016/pdf/75_Paper.pdf)

- High-level intention may span several domains



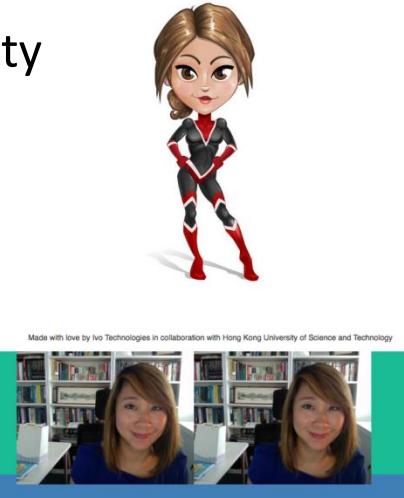
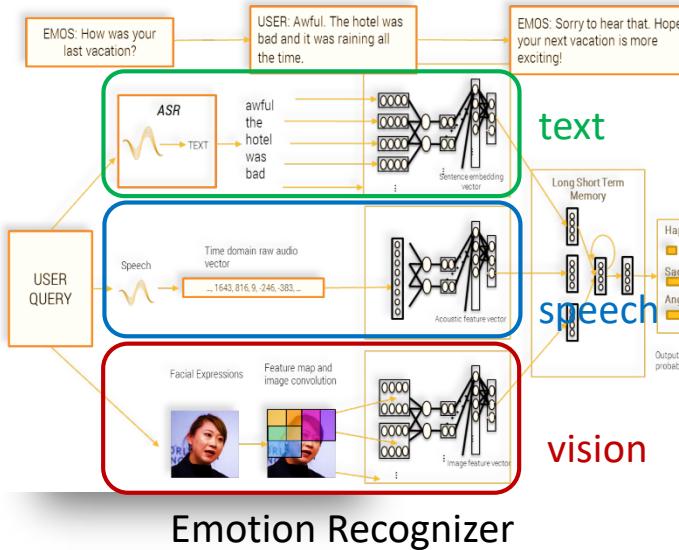
# Empathy in Dialogue System (Fung et al., 2016)

132

<https://arxiv.org/abs/1605.04072>

Zara - The Empathetic Supergirl

- Embed an empathy module
  - Recognize emotion using multimodality
  - Generate emotion-aware responses



**Face recognition output**

```
{  
  "recognition": "Race: Asian Confidence: 65.42750000000001 Smiling:  
  3.95896 Gender: Female Confidence: 88.9369",  
  "race": "Asian",  
  "race_confidence": "65.42750000000001",  
  "smiling": "3.95896",  
  "gender": "Female",  
  "gender_confidence": "88.9369"  
}
```

(index):1728  
(index):1729

# Visual Object Discovery through Dialogues

(Vries et al., 2017)

133

<https://arxiv.org/pdf/1611.08481.pdf>

- Recognize objects using “Guess What?” game
- Includes “spatial”, “visual”, “object taxonomy” and “interaction”



Is it a person?

**No**

Is it an item being worn or held?

**Yes**

Is it a snowboard?

**Yes**

Is it the red one?

**No**

Is it the one being held by the person in blue?

**Yes**



Is it a cow?

**Yes**

Is it the big cow in the middle?

**No**

Is the cow on the left?

**No**

On the right ?

**Yes**

First cow near us?

**Yes**

## Part V

### Challenges

# Outline

135

- Introduction and Background
  - Neural Networks
  - Reinforcement Learning
- Deep Learning Based Dialogue System
  - Spoken/Natural Language Understanding (SLU/NLU)
  - Dialogue State Tracking (DST)
  - Dialogue Policy
  - Natural Language Generation (NLG)
  - End-to-End Learning for Dialogue Systems
- Evaluation
- Recent Trends on Learning Dialogues
- **Challenges**
- Conclusion

# Challenges in Dialogue Modeling - I

136

- **Semantic schema induction** (Chen et al., 2013; Athanasopoulou, et al., 2014)
  - No predefined semantic schema
  - How to learn from data?
- **Tractability, and dimensionality reduction methods**
  - Learning with large state action spaces
- **End-to-end learning methods**
  - Learning when the user input is complex NL utterance
  - Learning with humans or KBs ?
  - Learning under domain shifts

# Challenges in Dialogue Modeling - II

137

- **Multiple-State hypothesis**
  - Tracking a distribution over multiple dialog states can improve dialog accuracy
  - How does current dialog systems deal with this?
- **Proactive v.s. reactive approaches to dialog modeling**
  - How to build DM models when the agent is proactive (i.e., does not wait for the user but sends messages and drives the conversation)
- **Localization, personalization, etc.**
  - How to deal with issue pertaining to place, temporal and personal context. Mostly dealt on speech side. How about DM side for when learning the policy?
- **Hierarchical RL approach to policy learning actually works?**
  - When are they useful?
  - How about for open domain systems (like chit-chat) - Are they powerful?

# Challenges in Dialogue Modeling - III

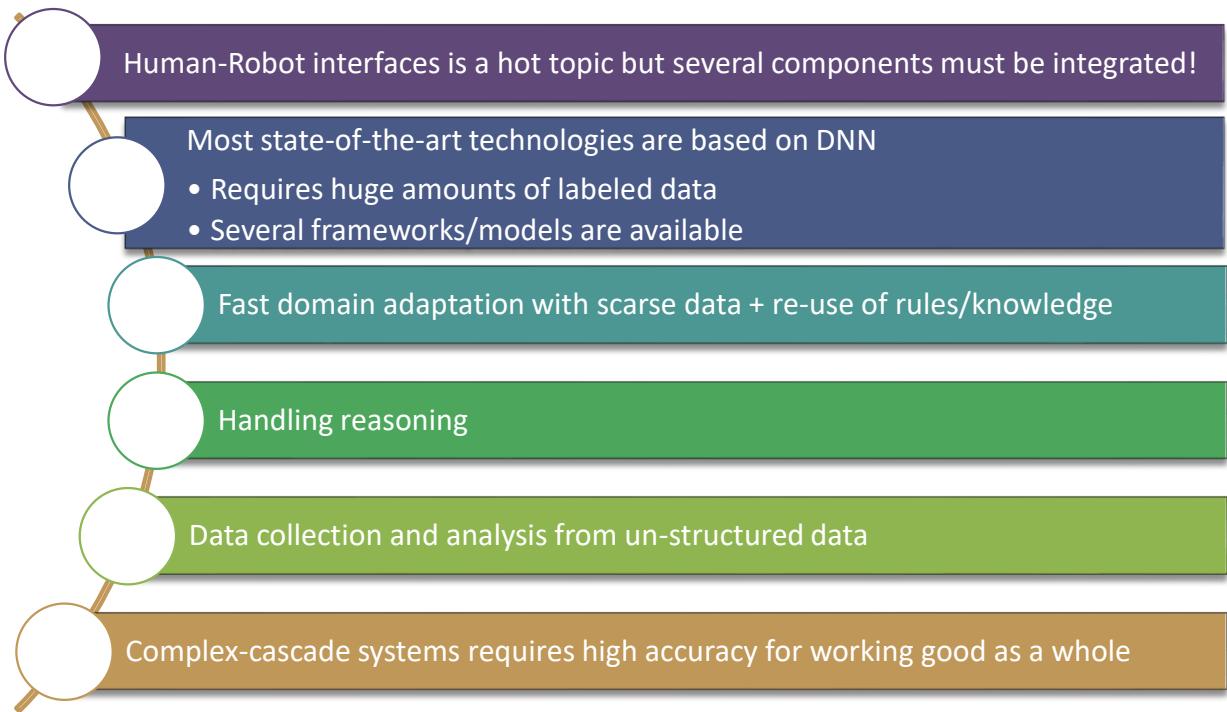
138

## □ Chat-Bot challenges

- Consistency: Keep similar answers in spite of different wordings
  - Human: *what is your job?*
  - Machine: *I am lawyer*
  - Human: *what do you do ?*
  - Machine: *I am a doctor*
- Quick domain-dependent adaptation: specially from unstructured data (Yan et.al, 2016)
- Personalization: handling profiles, interaction levels, and keep relevant context history (Li et al., 2016)
- Long sentence generation: most sentence are short or common phrases

# Challenge Summary

139



## Part VI

### Conclusion

# Briefly...

141

- We introduced recent deep learning approaches that are used in building dialogue models
- We highlighted the main components of dialogue systems and new deep learning architectures used for these components
- We talked about the challenges and new avenues for future research
- We provide all the material online!

<http://deepdialogue.miulab.tw>

# References

142

- The full list of references can be found in:  
<http://deepdialogue.miulab.tw>

# Acknowledgement

143

- We thanks Tsung-Hsien Wen, Pei-Hao Su, Li Deng, Sungjin Lee, Milica Gašić, Lihong Li for sharing their slides

# Thanks for Your Attendance!

