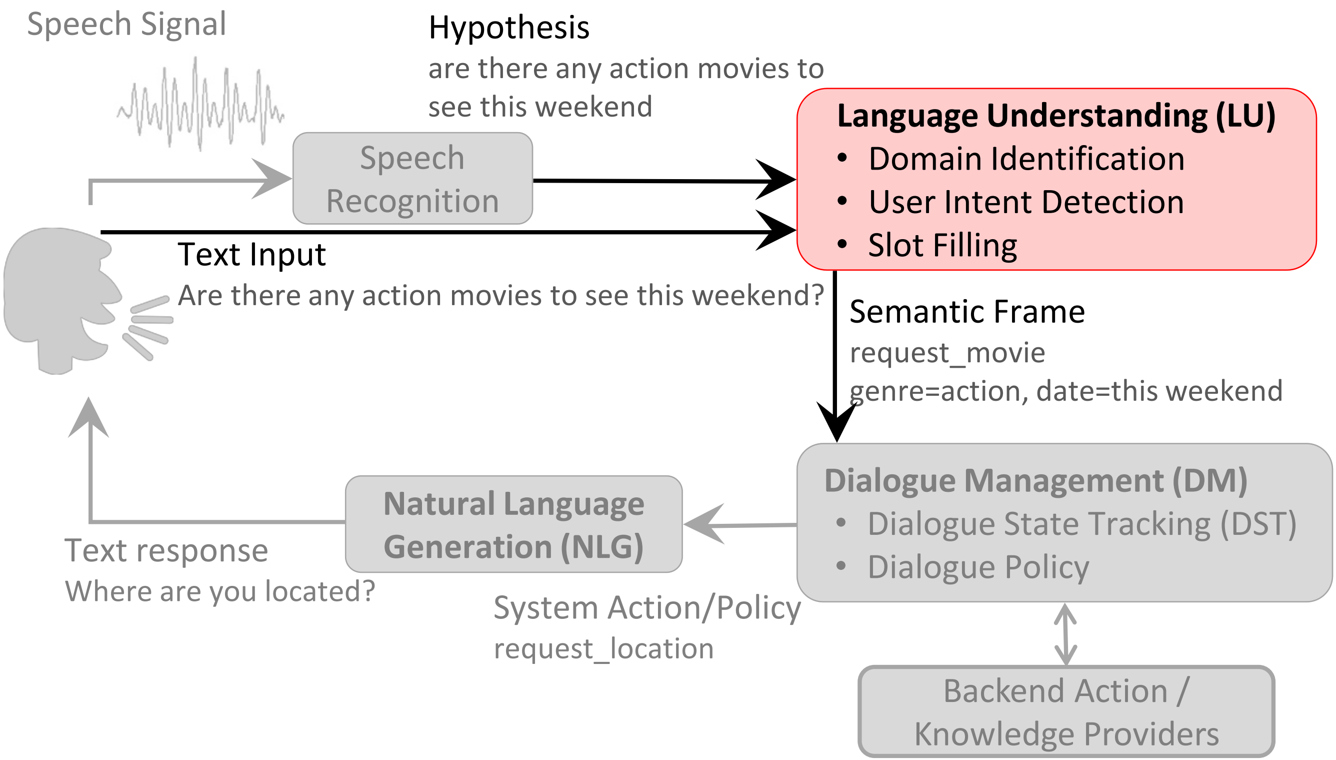
Outline

- 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)

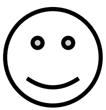


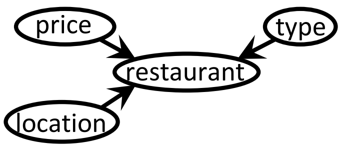
Semantic Frame Representation

- 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

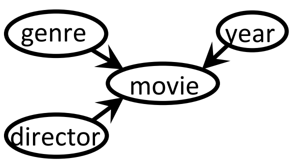




find_restaurant (price="cheap", type="taiwanese", location="oakland")

Movie Domain

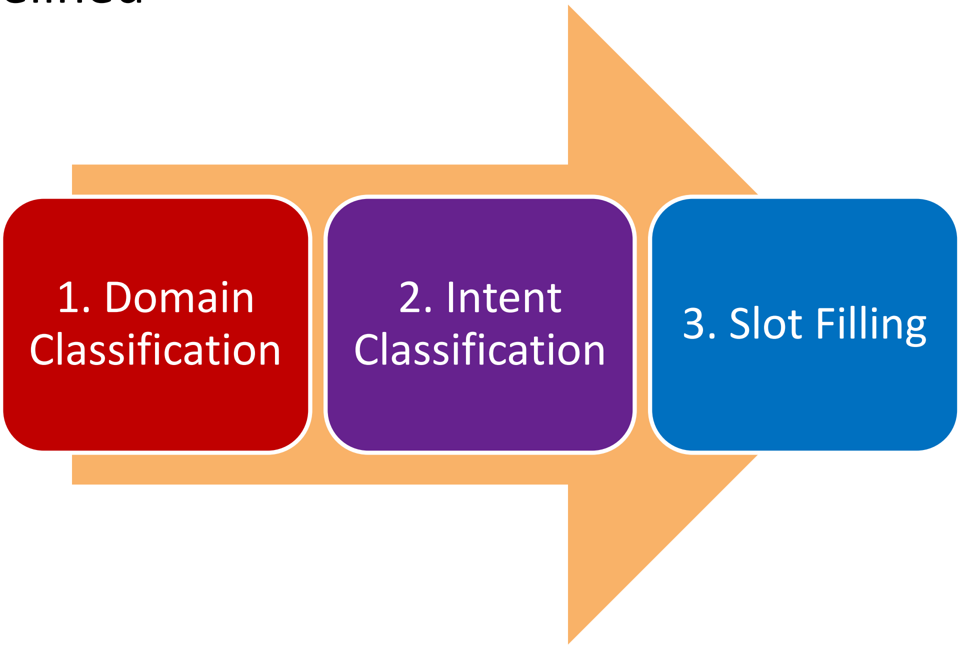
show me action movies directed by james cameron



find_movie (genre="action", director="james cameron")

Language Understanding (LU)

- Pipelined



LU – Domain/Intent Classification

As an utterance classification task

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

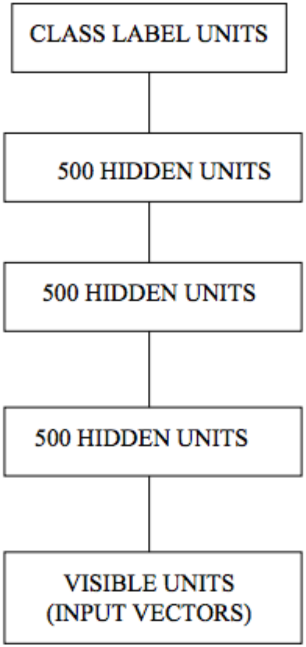
find me a cheap taiwanese restaurant in oakland

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)

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

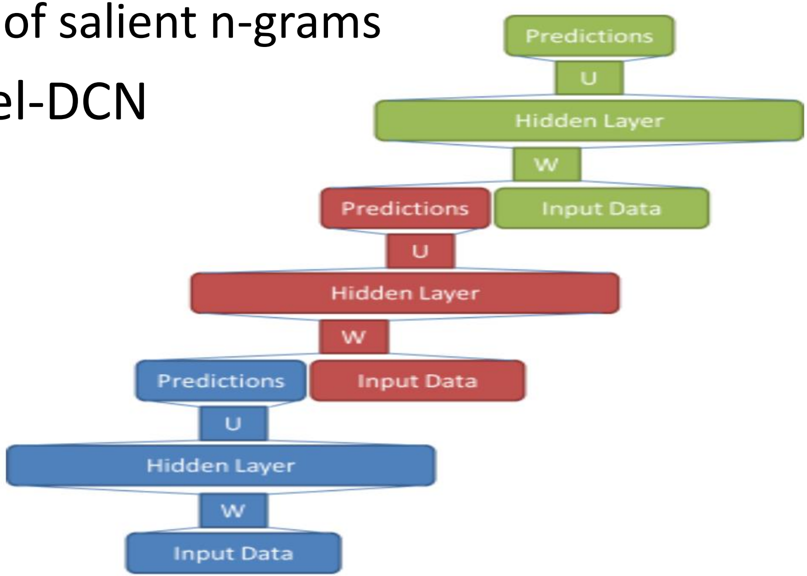
- Deep belief nets (DBN)
  - Unsupervised training of weights
  - Fine-tuning by back-propagation
  - Compared to MaxEnt, SVM, and boosting



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

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

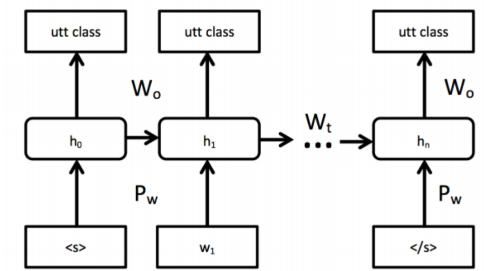
- Deep convex networks (DCN)
  - Simple classifiers are stacked to learn complex functions
  - Feature selection of salient n-grams
- Extension to kernel-DCN

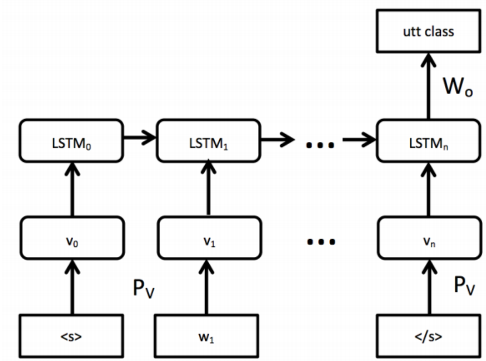


Deep Neural Networks for Domain/Intent Classification – III (Ravuri and Stolcke, 2015)

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

As a **sequence tagging task**

- Given a collection tagged word sequences, $S=\{((w_{1,1},w_{1,2},\ldots, w_{1,n1}), (t_{1,1},t_{1,2},\ldots,t_{1,n1})), ((w_{2,1},w_{2,2},\ldots,w_{2,n2}), (t_{2,1},t_{2,2},\ldots,t_{2,n2})) \ldots\}$ where $t_i \in M$, the goal is to estimate tags for a new word sequence.

flights from Boston to New York today

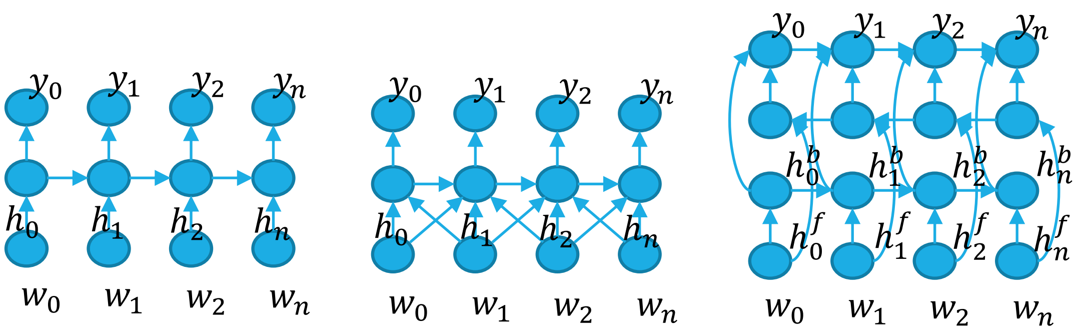
	flights	from	Boston	to	New	York	today
Entity Tag	O	O	B-city	O	B-city	I-city	O
Slot Tag	O	O	B-dept	O	B-arrival	I-arrival	B-date

Recurrent Neural Nets for Slot Tagging – I

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

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



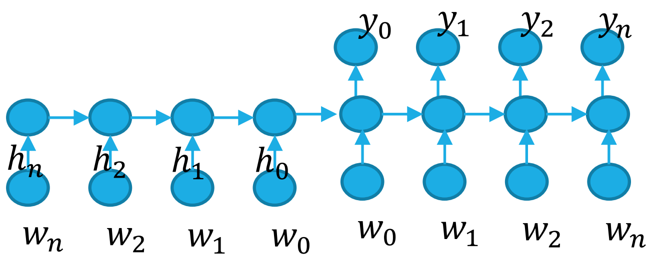
(a) LSTM (b) LSTM-LA (c) bLSTM

Recurrent Neural Nets for Slot Tagging – II

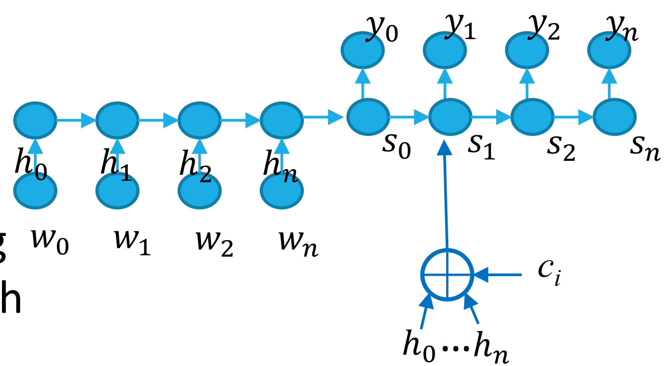
(Kurata et al., 2016; Simonnet et al., 2015)

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

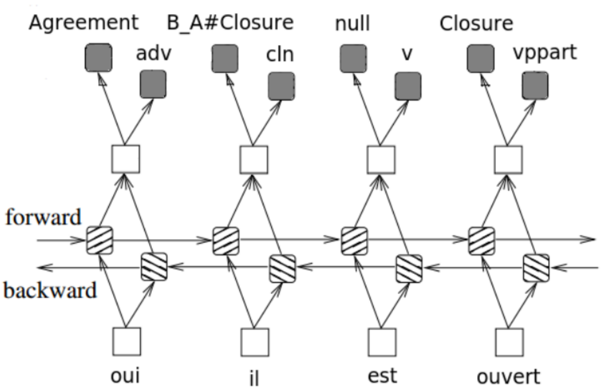


Recurrent Neural Nets for Slot Tagging – III

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

https://arxiv.org/abs/1604.00117; 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

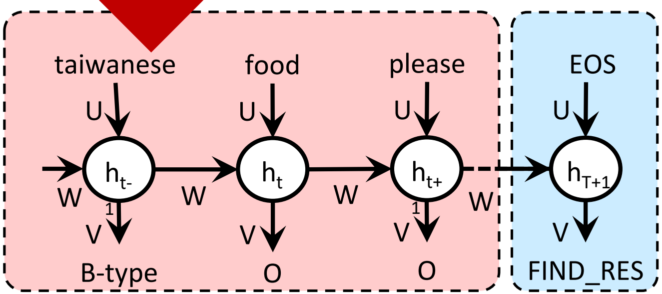
https://www.microsoft.com/en-us/research/wp-content/uploads/2016/06/IS16_MultiJoint.pdf; https://arxiv.org/abs/1609.01454

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

- Slot filling and intent prediction in the same output sequence

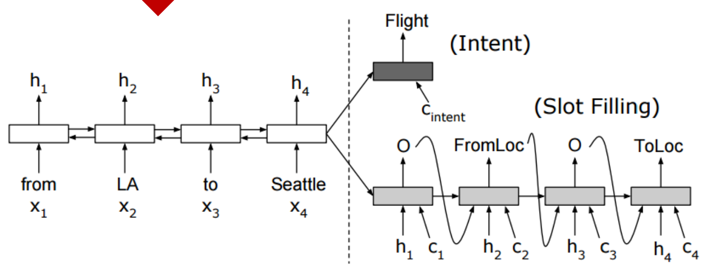
Parallel (Liu and Lane, 2016)

- Intent prediction and slot filling are performed in two branches



Slot Filling

Intent Prediction



Contextual LU

Domain Identification → Intent Prediction → Slot Filling

D communication *I* send_email

Single Turn

U just sent email to bob about fishing this weekend

S O O O O B-contact_name O B-subject I-subject I-subject

→ send_email(contact_name="bob", subject="fishing this weekend")

Multi-Turn

U_1 send email to bob

S_1 B-contact_name

→ send_email(contact_name="bob")

U_2 are we going to fish this weekend

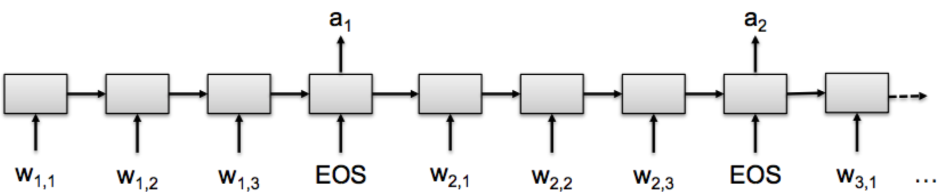
S_2 B-message I-message I-message I-message I-message I-message I-message

→ send_email(message="are we going to fish this weekend")

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

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)

https://www.microsoft.com/en-us/research/wp-content/uploads/2016/06/IS16_ContextualSLU.pdf

1. Sentence Encoding

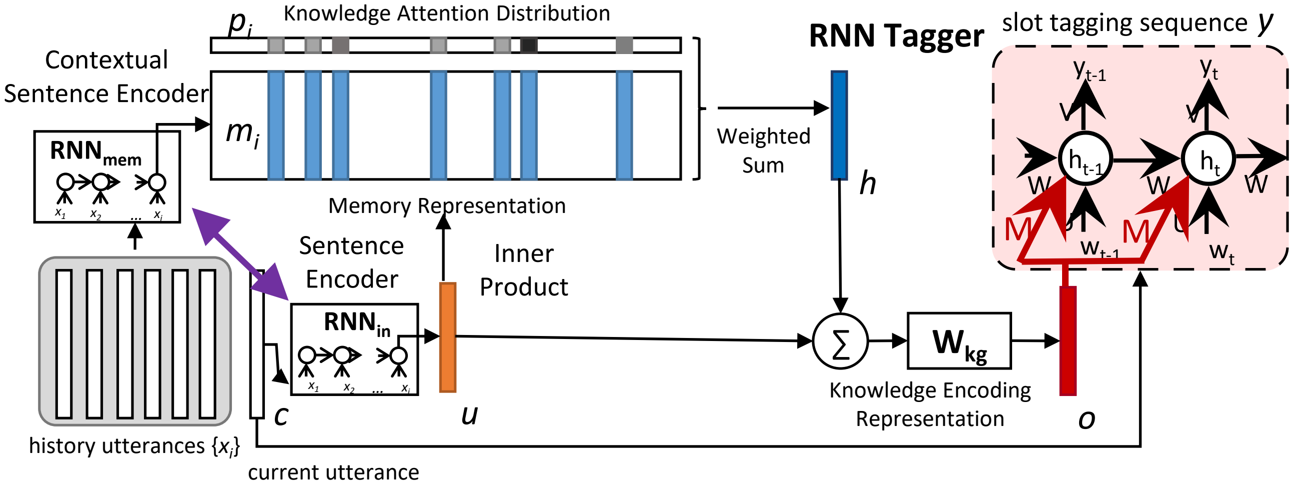
$$m_i = \mathrm{RNN}_{\mathrm{mem}}(x_i)$$
$$u = \mathrm{RNN}_{\mathrm{in}}(c)$$

2. Knowledge Attention

$$p_i = \mathrm{softmax}(u^T m_i)$$

3. Knowledge Encoding

$$h = \sum_i p_i m_i \quad o = W_{\mathrm{kg}}(h + u)$$

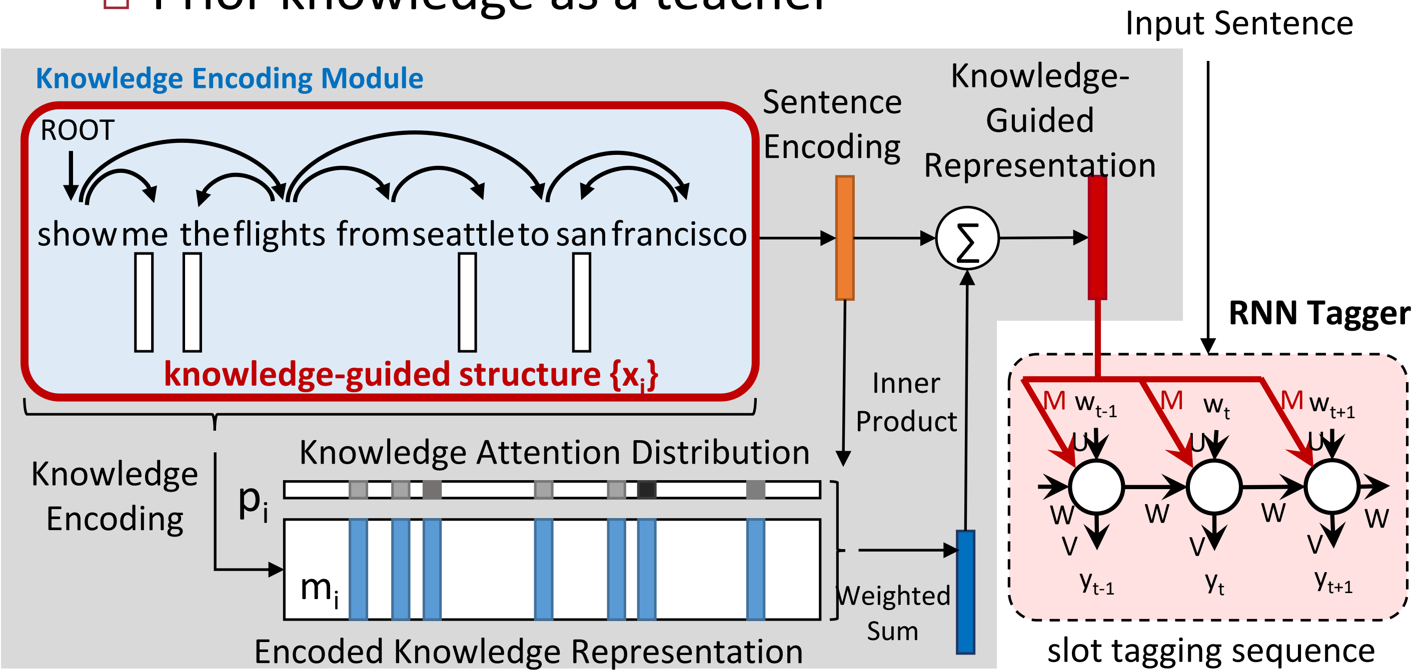


Idea: additionally incorporating contextual knowledge during slot tagging
→ track dialogue states in a latent way

Structural LU (Chen et al., 2016)

http://arxiv.org/abs/1609.03286

- Prior knowledge as a teacher



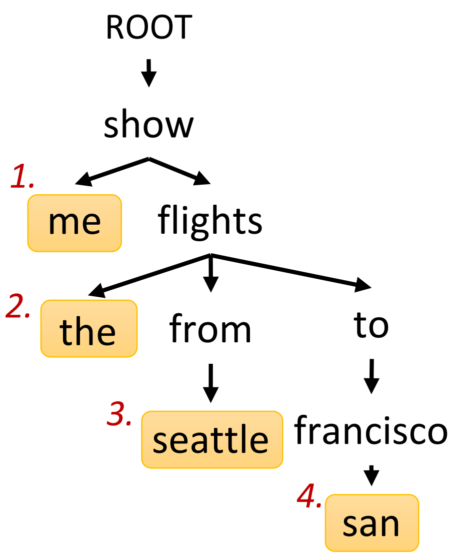
Structural LU (Chen et al., 2016)

http://arxiv.org/abs/1609.03286

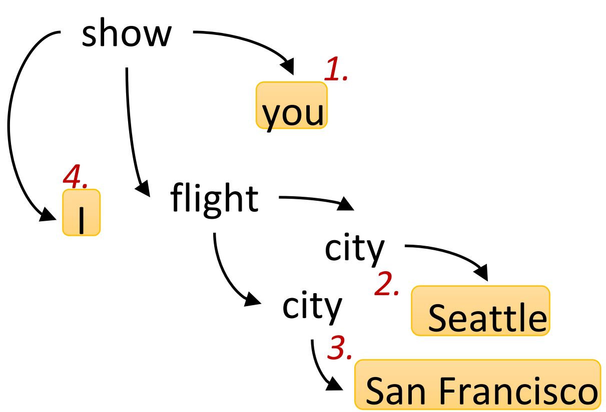
- Sentence structural knowledge stored as memory

Sentence s show me the flights from seattle to san francisco

Syntax (Dependency Tree)



Semantics (AMR Graph)



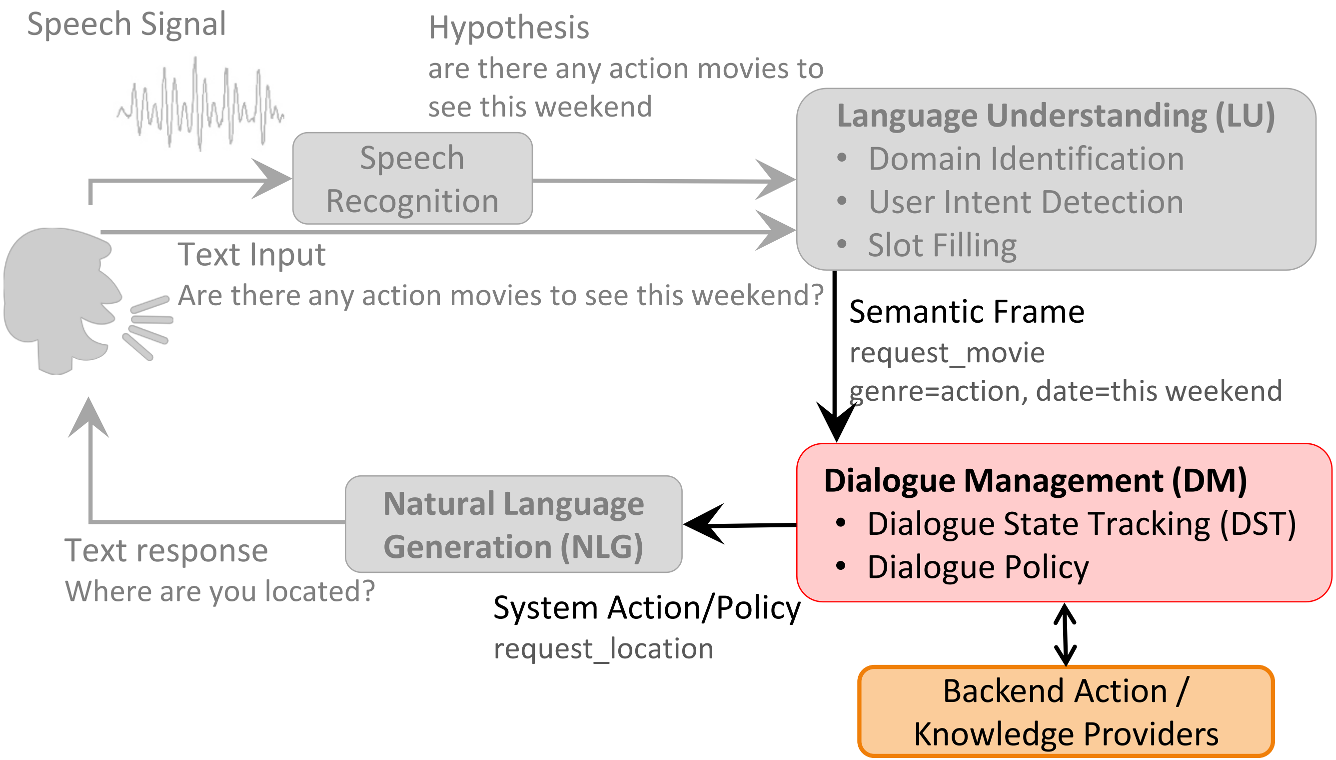
LU Evaluation

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

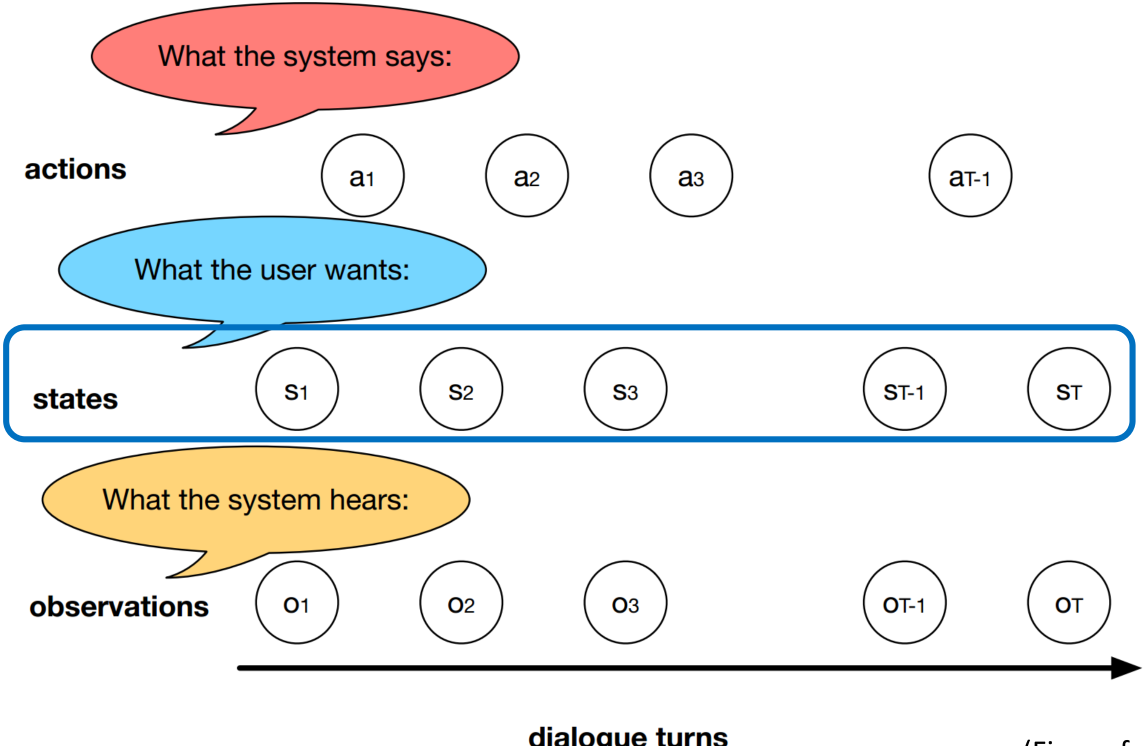
Outline

- 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)



Elements of Dialogue Management



(Figure from Gašić)

Dialogue State Tracking (DST)

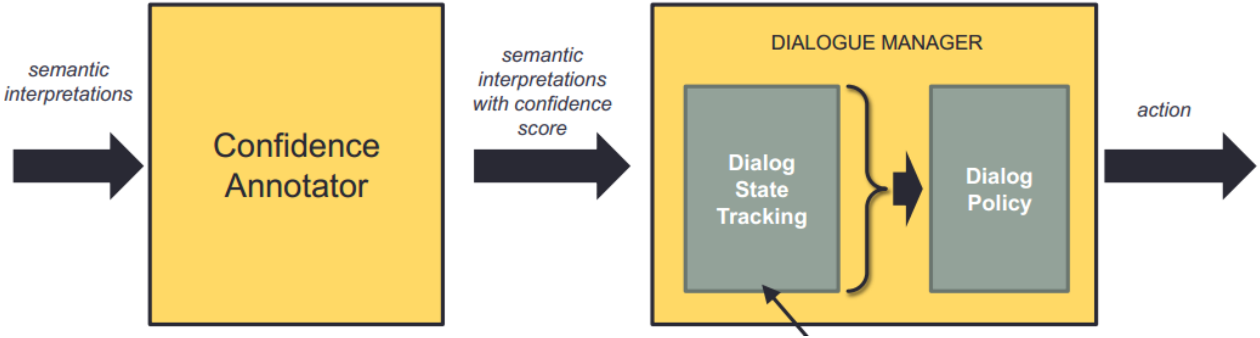
- Dialogue state: *a representation of the system's belief of the user's goal(s) at any time during the dialogue*
- Inputs
  - Current user utterance
  - Preceding system response
  - Results from previous turns
- For
  - Looking up knowledge or making API call(s)
  - Generating the next system action/response

Dialogue State Tracking (DST)

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)

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

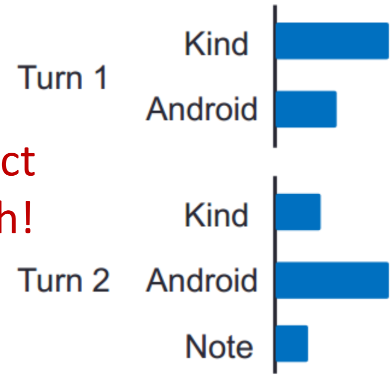
Turn 1
Kind
Android

Turn 2
Note
Android

Turn 1	
Kind	0.5
Android	0.3

Turn 2	
Note	0.4
Android	0.3

Incorrect for both!



Dialogue State Tracking (DST)

- 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)

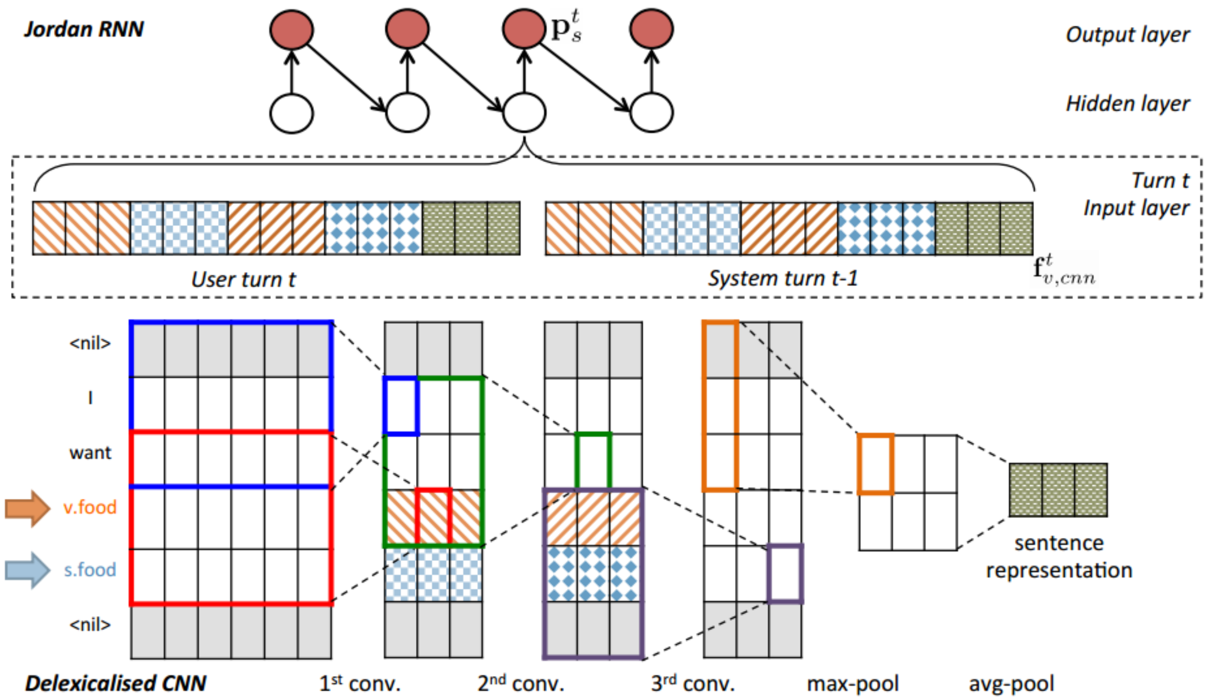
Dialog State Tracking Challenge (DSTC)

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

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

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

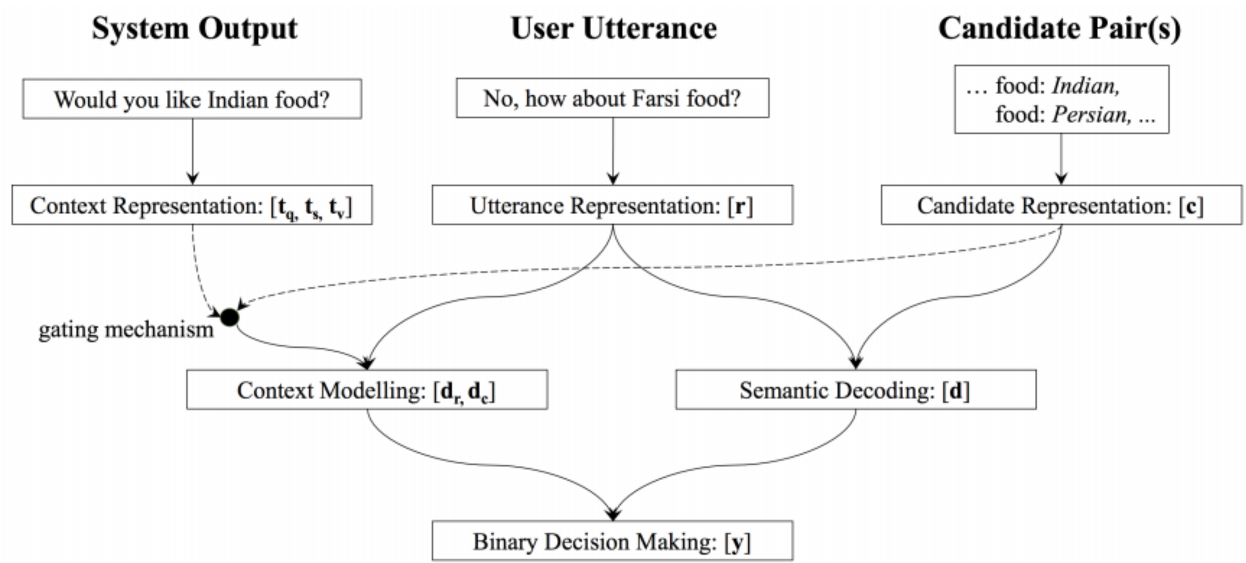
http://www.anthology.aclweb.org/W/W13/W13-4073.pdf; https://arxiv.org/abs/1506.07190



(Figure from Wen et al, 2016)

Neural Belief Tracker (Mrkšić et al., 2016)

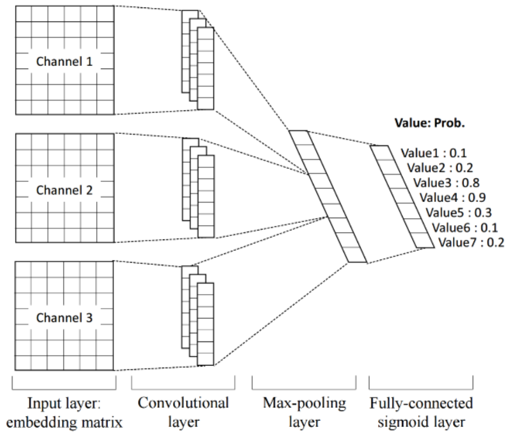
https://arxiv.org/abs/1606.03777

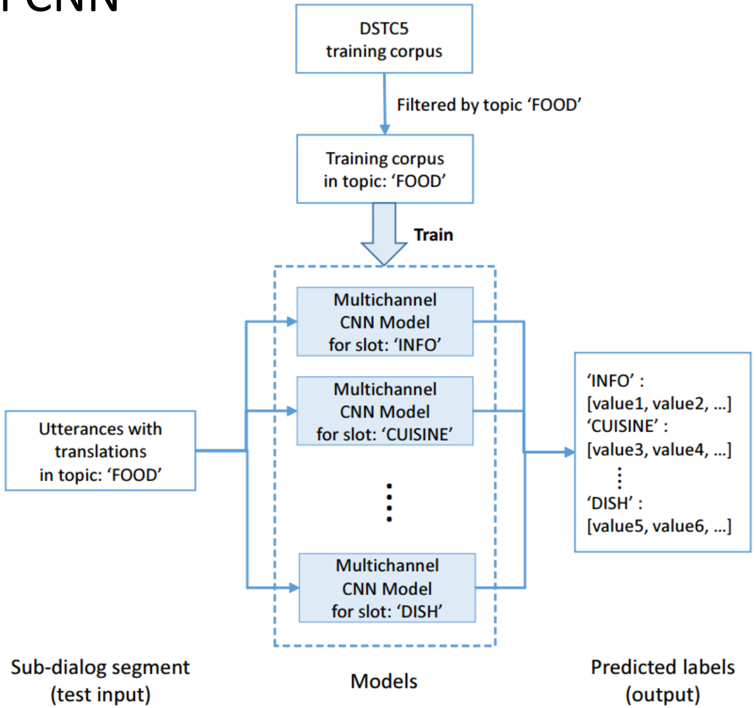


Multichannel Tracker (Shi et al., 2016)

https://arxiv.org/abs/1701.06247

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





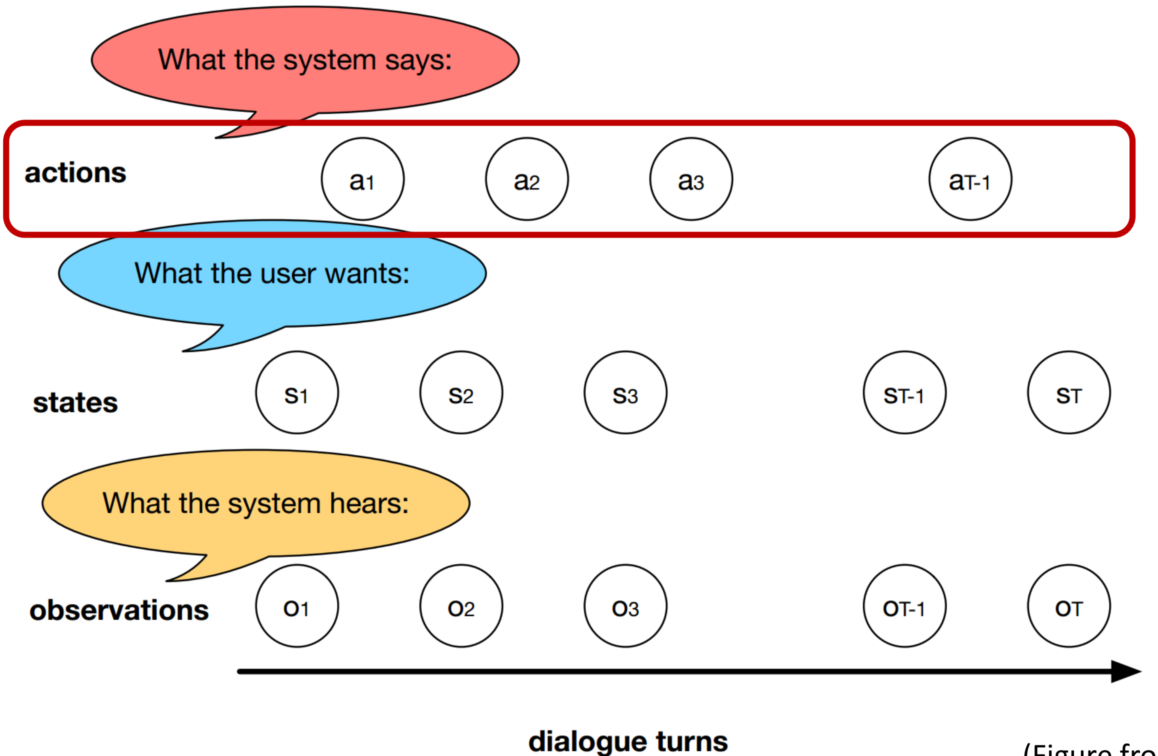
DST Evaluation

- 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

- 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

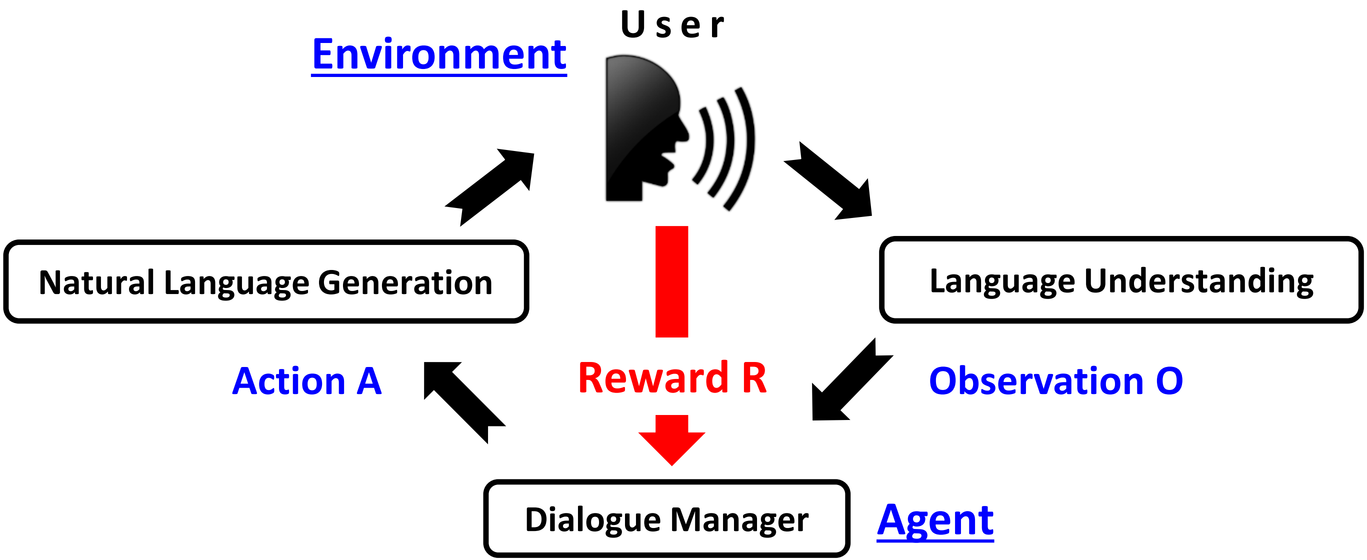
Elements of Dialogue Management



(Figure from Gašić)

Dialogue Policy Optimization

- Dialogue management in a RL framework



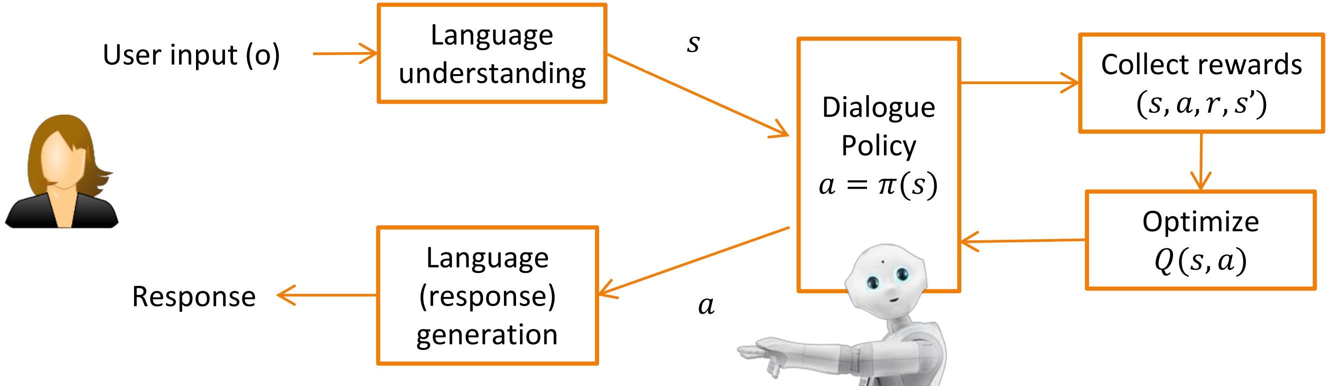
The optimized dialogue policy selects the best action that maximizes the future reward. Correct rewards are a crucial factor in dialogue policy training

Reward for RL $\cong$ Evaluation for System

- 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, high cost
- User rating	unreliable quality, medium cost
- Objective rating	Check desired aspects, low cost

Reinforcement Learning for Dialogue Policy Optimization



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

Typical reward function

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

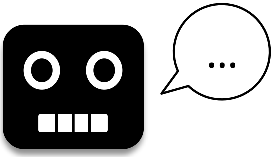
Typically requires domain knowledge

✓ Simulated user

✗ Paid users (Amazon Mechanical Turk)

✕ Real users

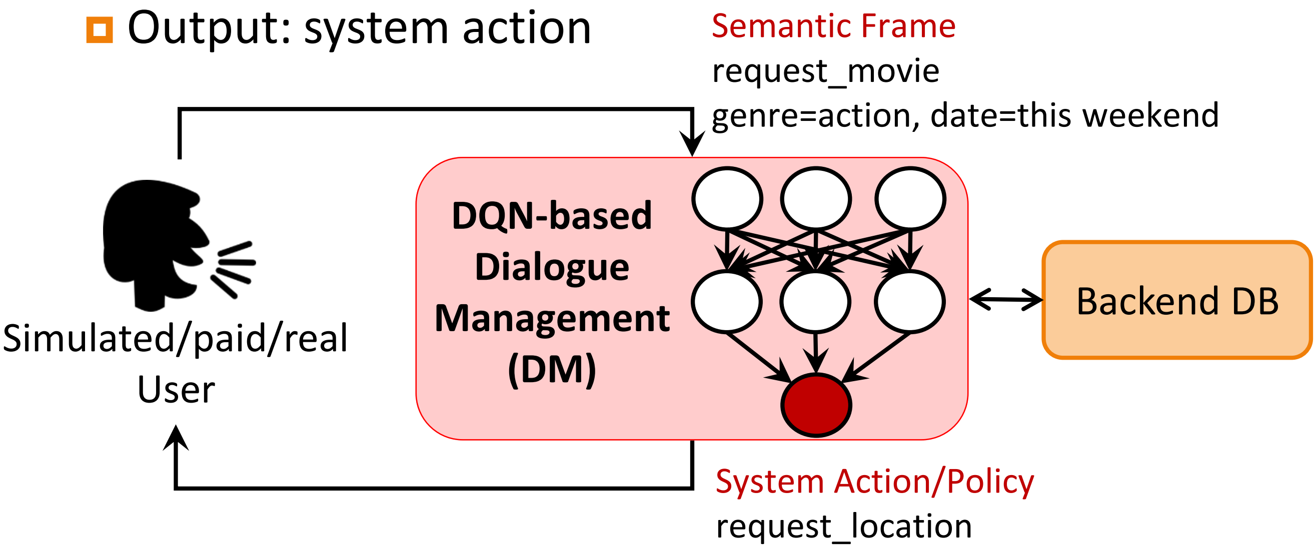
The user simulator is usually required for dialogue system training before deployment



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

https://arxiv.org/abs/1703.01008

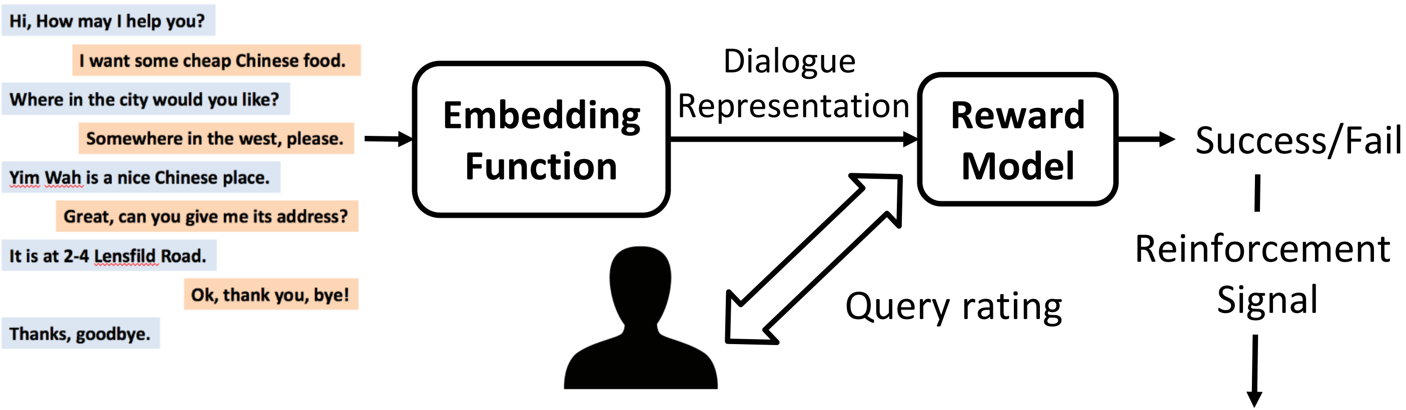
- Deep RL for training DM
  - Input: current semantic frame observation, database returned results
  - Output: system action



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

http://www.anthology.aclweb.org/W/W15/W15-46.pdf#page=437; https://www.aclweb.org/anthology/P/P16/P16-1230.pdf

- 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



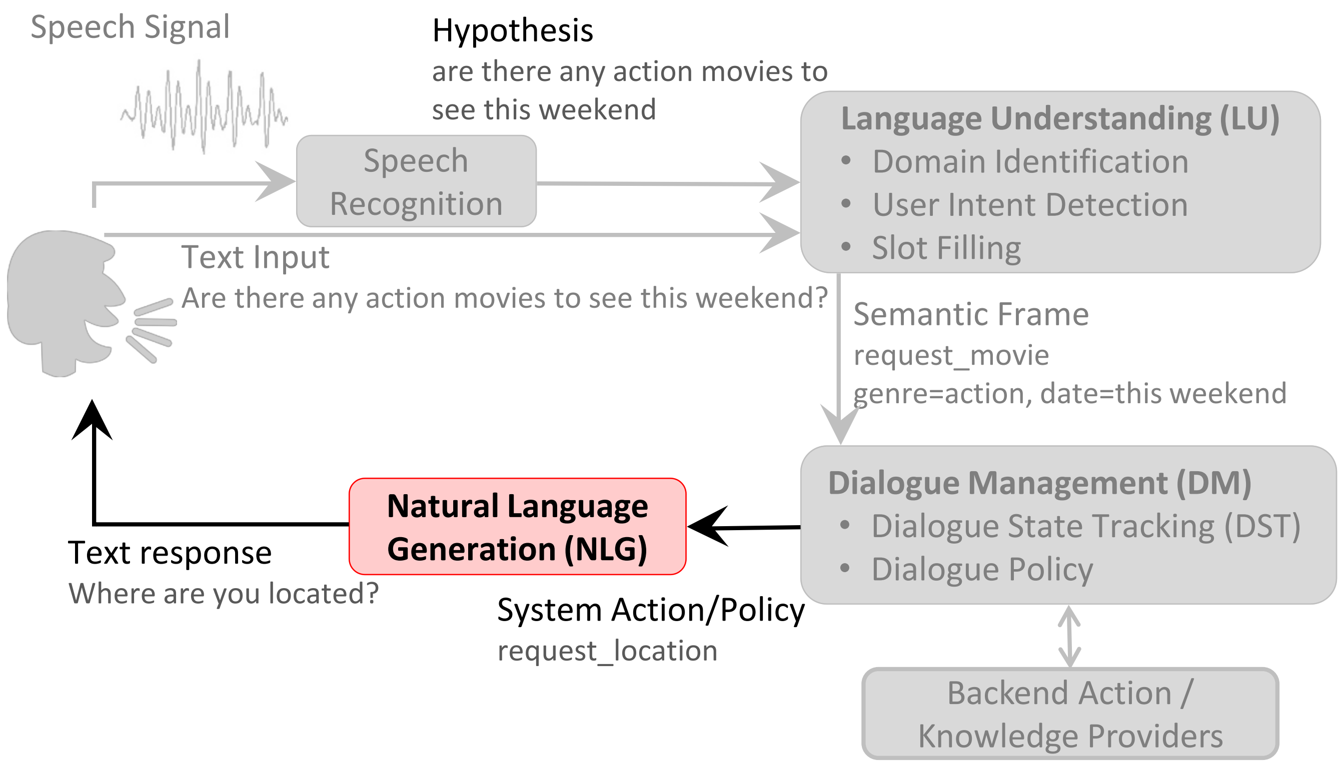
Dialogue Management Evaluation

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

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)



Natural Language Generation (NLG)

- Mapping dialogue acts into natural language

inform(name=Seven_Days, foodtype=Chinese)

Seven Days is a nice Chinese restaurant

Template-Based NLG

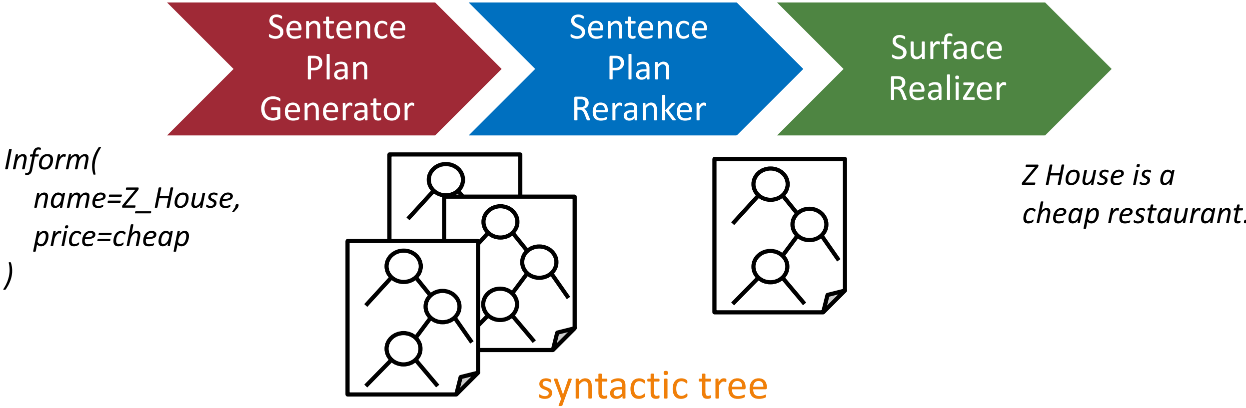
- Define a set of rules to map frames to NL

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)

- 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)

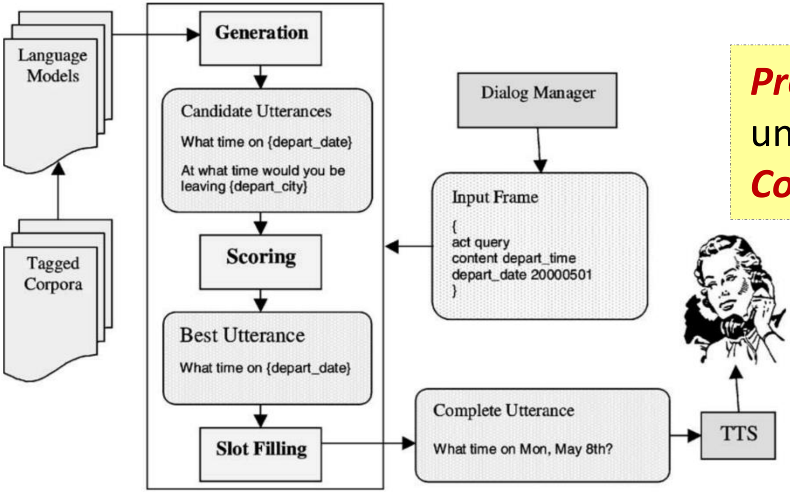
http://dl.acm.org/citation.cfm?id=1117568

- Class-based language modeling

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

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

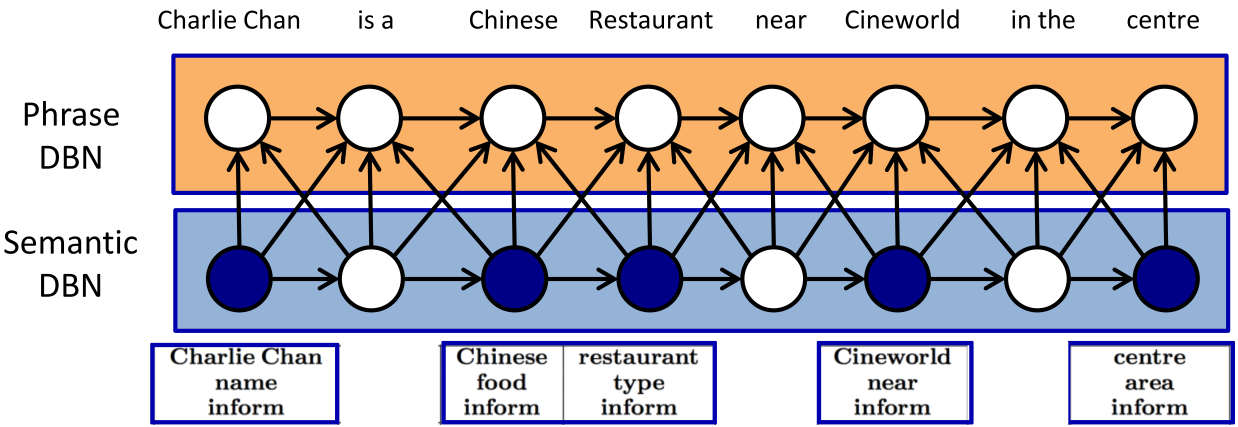
Classes:
inform_area
inform_address
…
request_area
request_postcode



Pros: easy to implement/ understand, simple rules
Cons: computationally inefficient

Phrase-Based NLG (Mairesse et al, 2010)

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



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

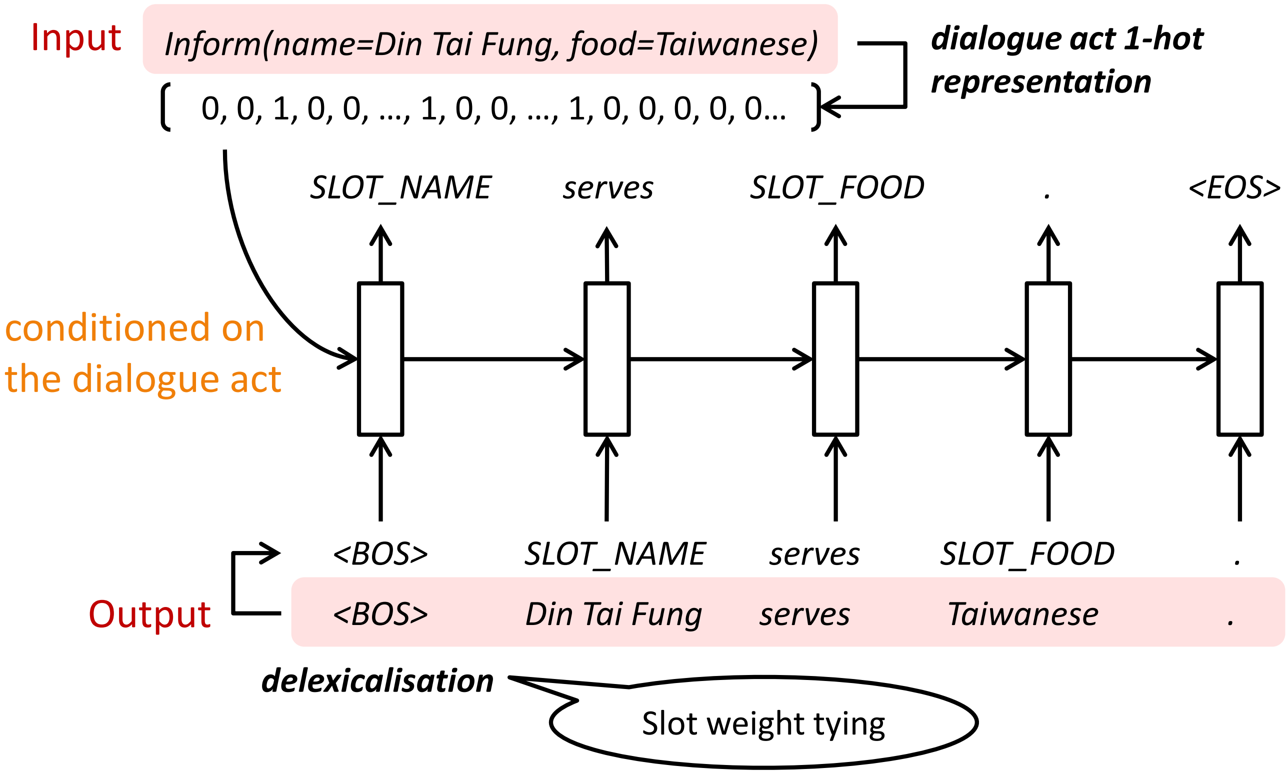
realization phrase semantic stack

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)

http://www.anthology.aclweb.org/W/W15/W15-46.pdf#page=295



Handling Semantic Repetition

- 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)

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

- Original LSTM cell

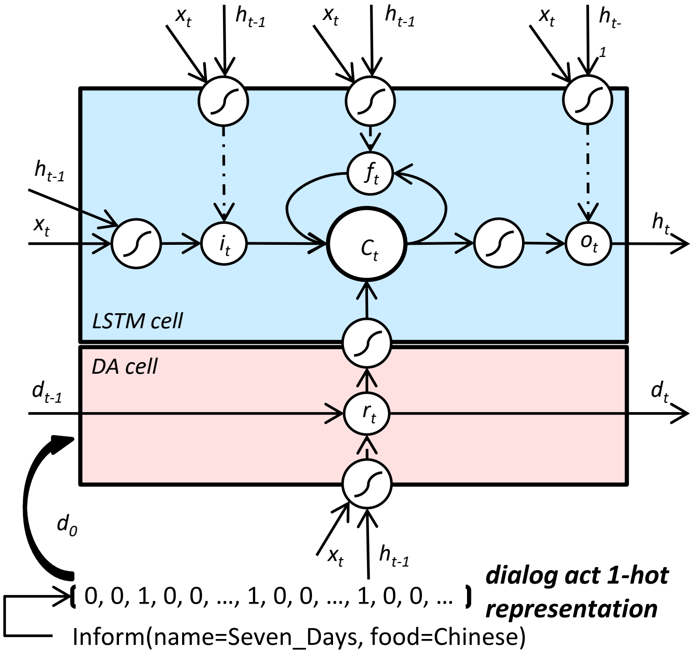
$$\mathbf{i}_t = \sigma(\mathbf{W}_{wi}\mathbf{x}_t + \mathbf{W}_{hi}\mathbf{h}_{t-1})$$
$$\mathbf{f}_t = \sigma(\mathbf{W}_{wf}\mathbf{x}_t + \mathbf{W}_{hf}\mathbf{h}_{t-1})$$
$$\mathbf{o}_t = \sigma(\mathbf{W}_{wo}\mathbf{x}_t + \mathbf{W}_{ho}\mathbf{h}_{t-1})$$
$$\hat{\mathbf{c}}_t = \tanh(\mathbf{W}_{wc}\mathbf{x}_t + \mathbf{W}_{hc}\mathbf{h}_{t-1})$$
$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \hat{\mathbf{c}}_t$$
$$\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t)$$

- 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 Ct

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

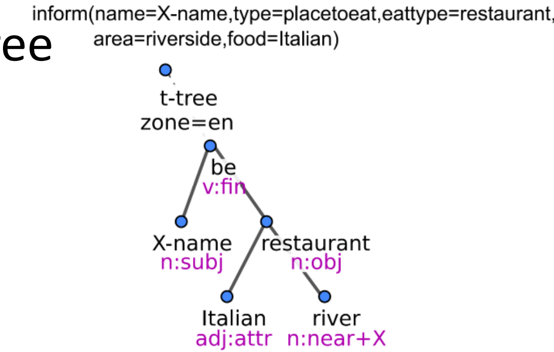


Idea: using gate mechanism to control the generated semantics (dialogue act/slots)

Structural NLG (Dušek and Jurčíček, 2016)

https://www.aclweb.org/anthology/P/P16/P16-2.pdf#page=79

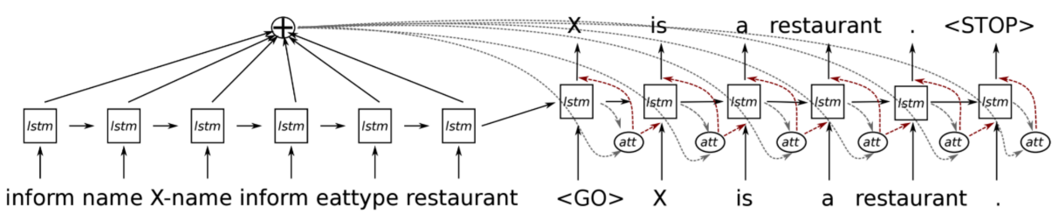
- Goal: NLG based on the syntax tree
  - Encode trees as sequences
  - Seq2Seq model for generation



(<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

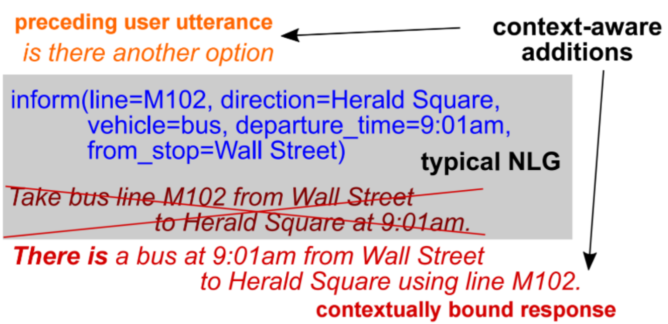
X is an Italian restaurant near the river.

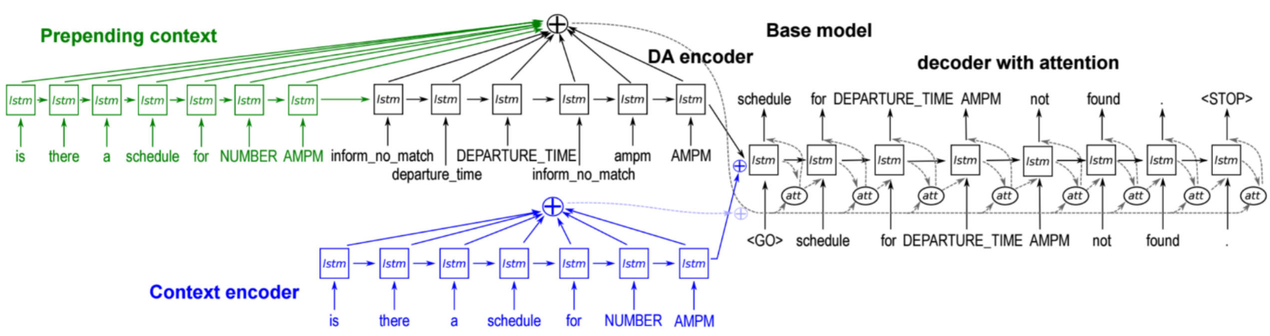


Contextual NLG (Dušek and Jurčíček, 2016)

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

- 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

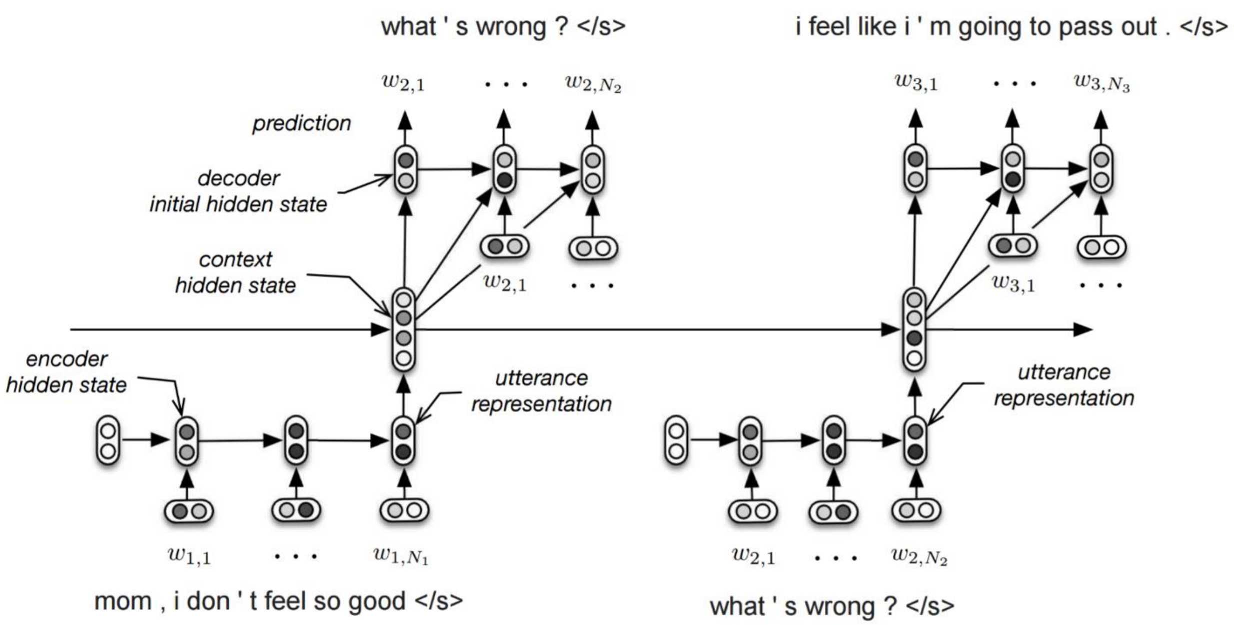
Outline

ChitChat Hierarchical Seq2Seq

(Serban et.al., 2016)

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

- A hierarchical seq2seq model for generating dialogues

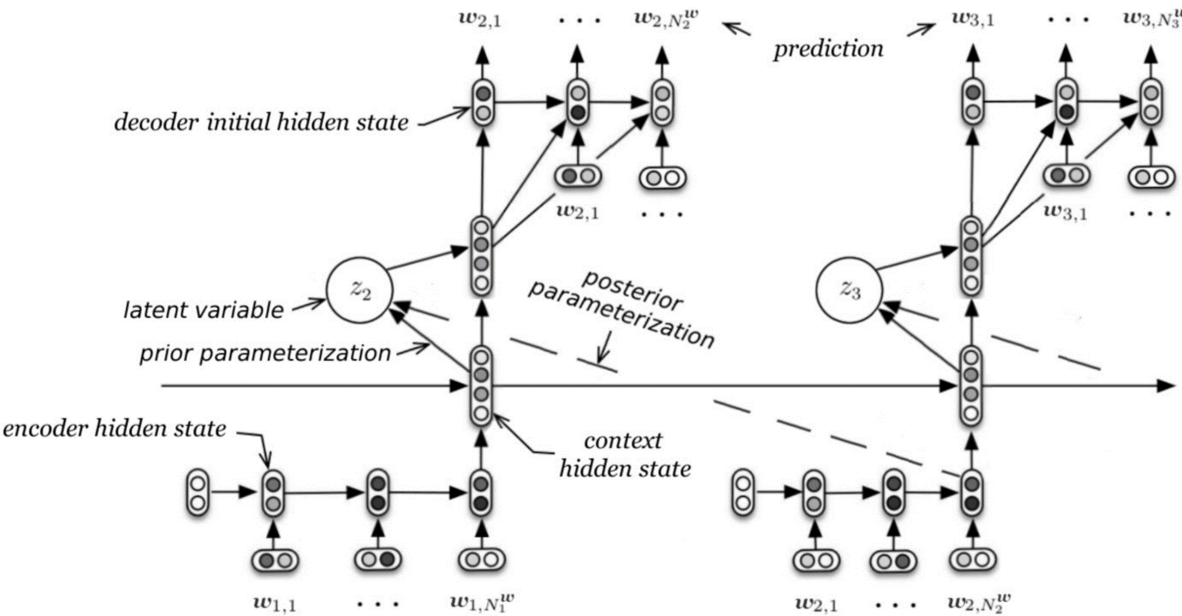


ChitChat Hierarchical Seq2Seq

(Serban et.al., 2017)

https://arxiv.org/abs/1605.06069

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

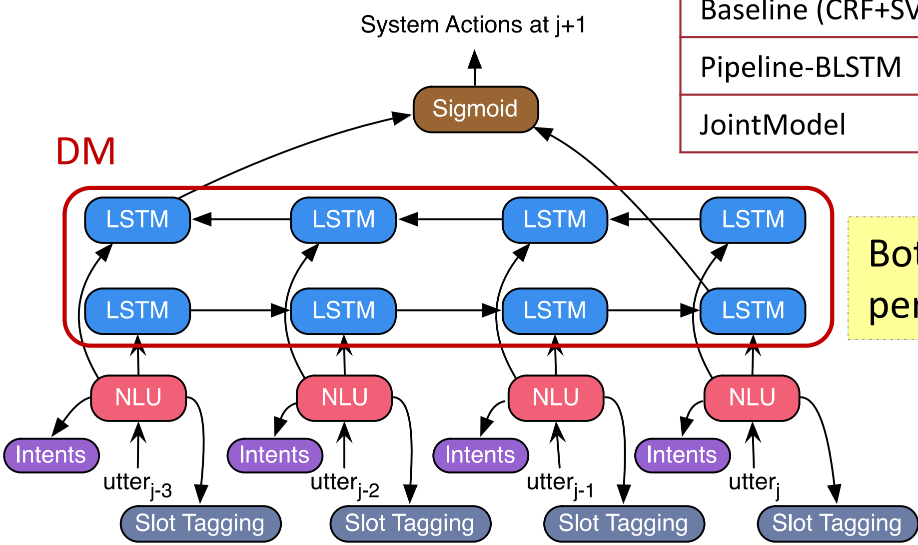


E2E Joint NLU and DM (Yang et al., 2017)

https://arxiv.org/abs/1612.00913

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

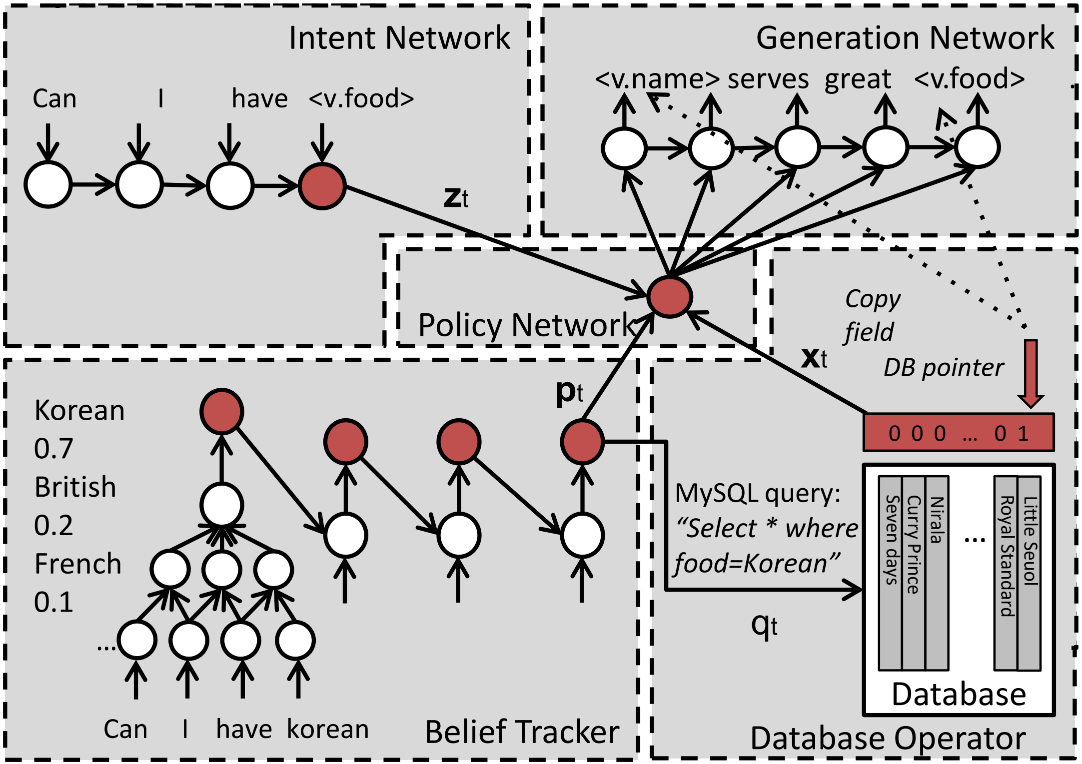
Model	DM	NLU
Baseline (CRF+SVMs)	7.7	33.1
Pipeline-BLSTM	12.0	36.4
JointModel	**22.8**	**37.4**



Both DM and NLU performance is improved

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

https://arxiv.org/abs/1604.04562

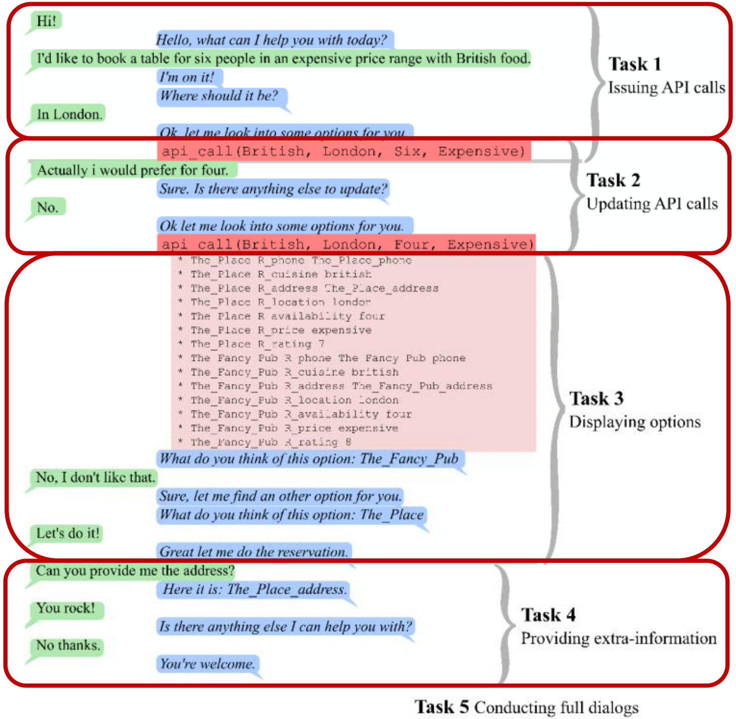


E2E MemNN for Dialogues (Bordes et al., 2016)

https://arxiv.org/abs/1605.07683

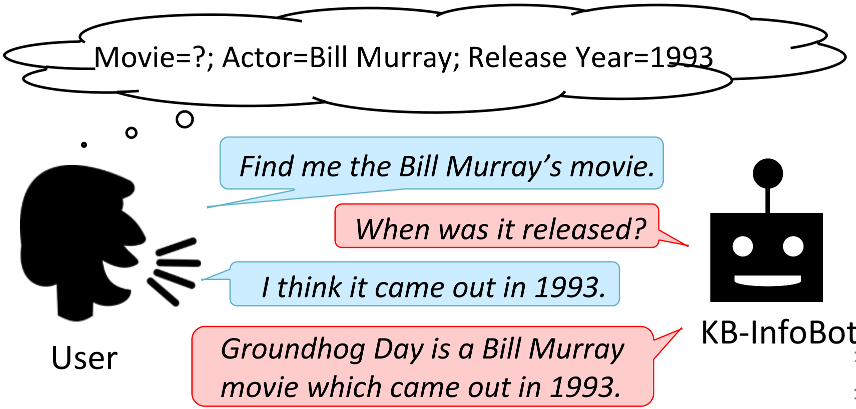
- Split dialogue system actions into subtasks
  - API issuing
  - API updating
  - Option displaying
  - Information informing

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



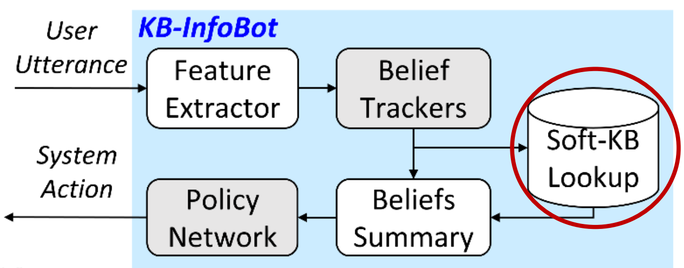
E2E RL-Based Info-Bot (Dhingra et al., 2016)

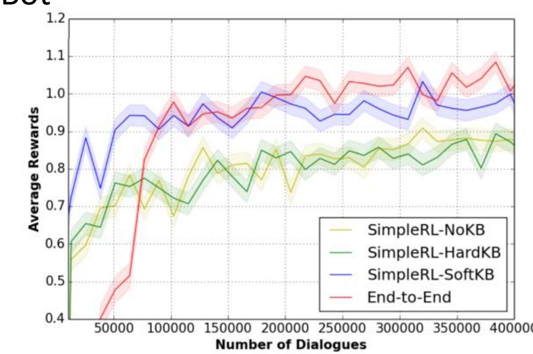
https://arxiv.org/abs/1609.00777



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)



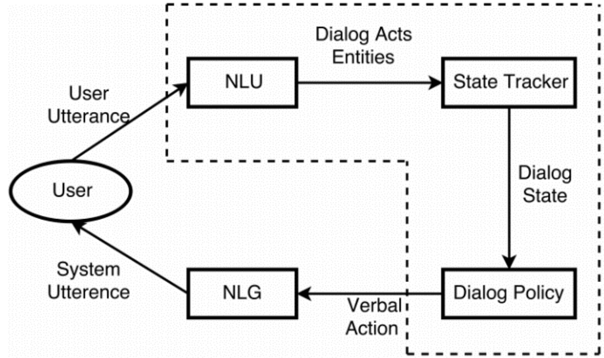


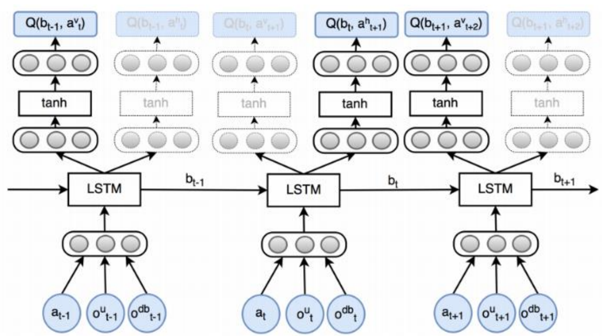
Idea: differentiable database for propagating the gradients

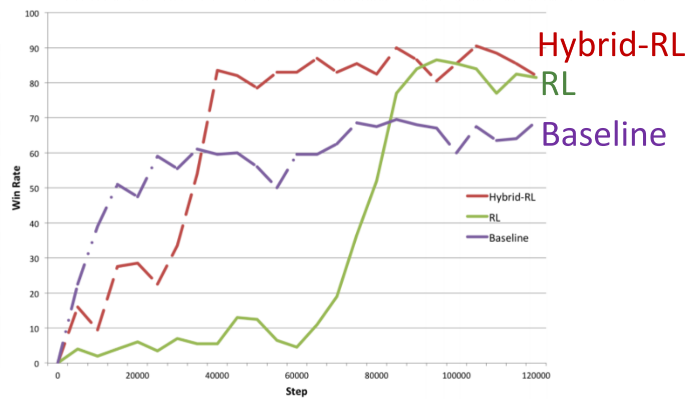
E2E RL-Based System (Zhao and Eskenazi, 2016)

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



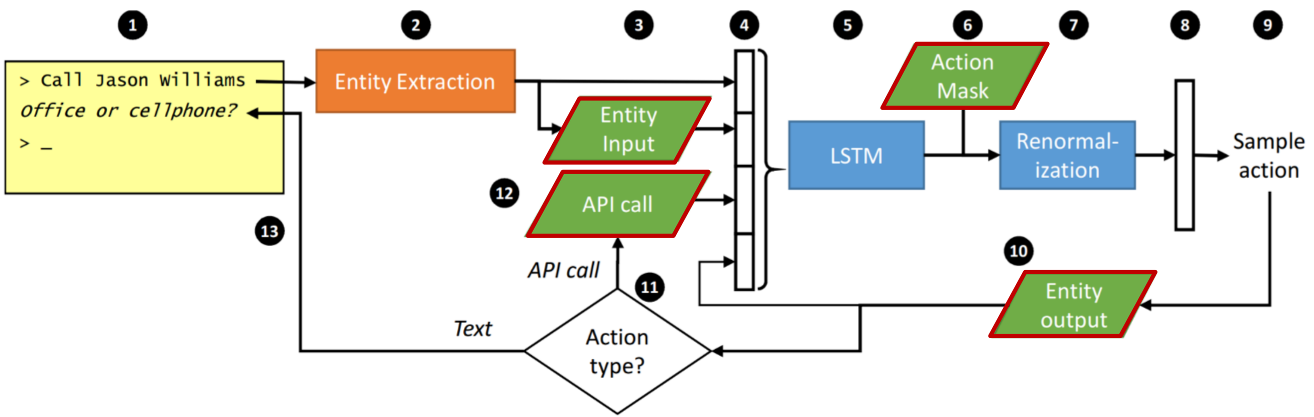




E2E LSTM-Based Dialogue Control
(Williams and Zweig, 2016)

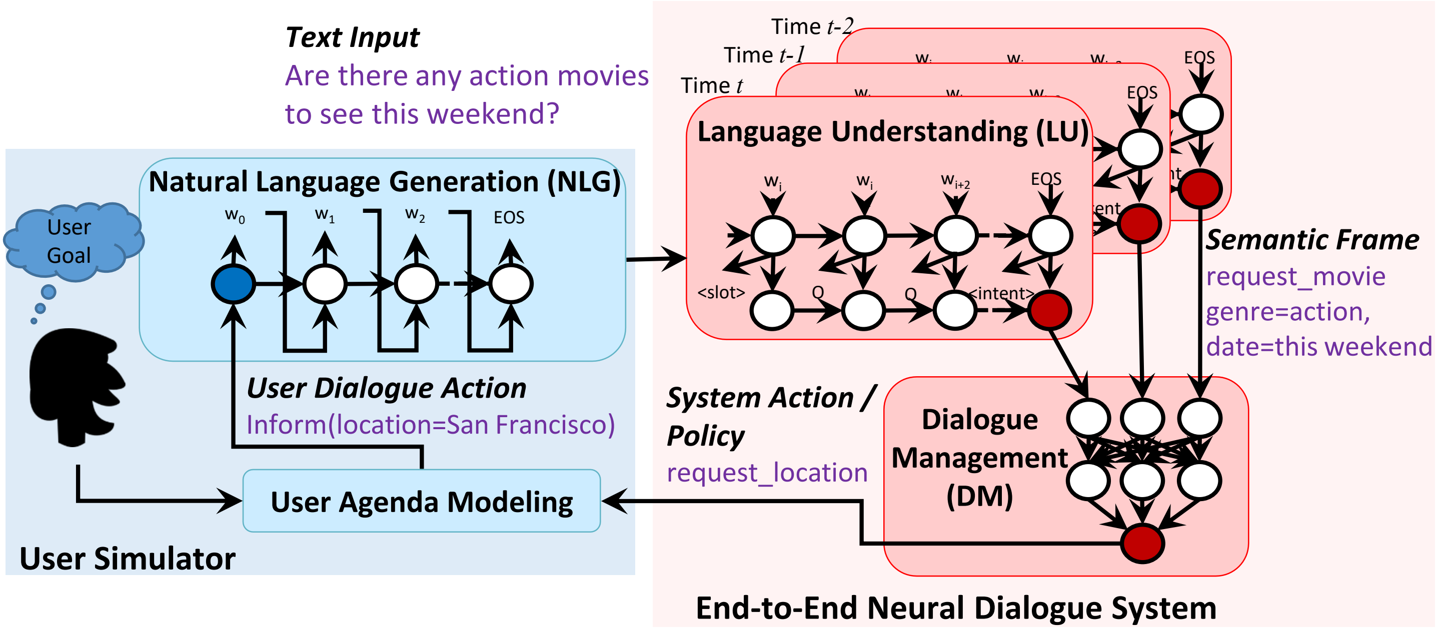
https://arxiv.org/abs/1606.01269

- 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)

https://arxiv.org/abs/1703.01008



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

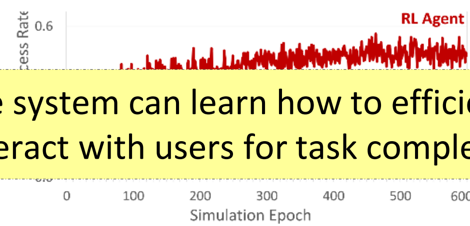
E2E Task-Completion Bot (TC-Bot) (Li et al., 2017)

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

Part III

Evaluation

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

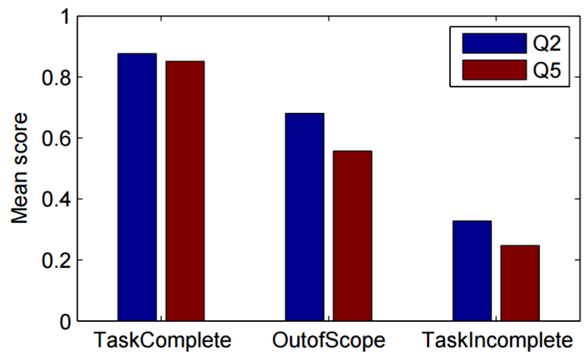
Dialogue System Evaluation

- 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)

http://www-scf.usc.edu/~zhaojuny/docs/SDSchapter_final.pdf

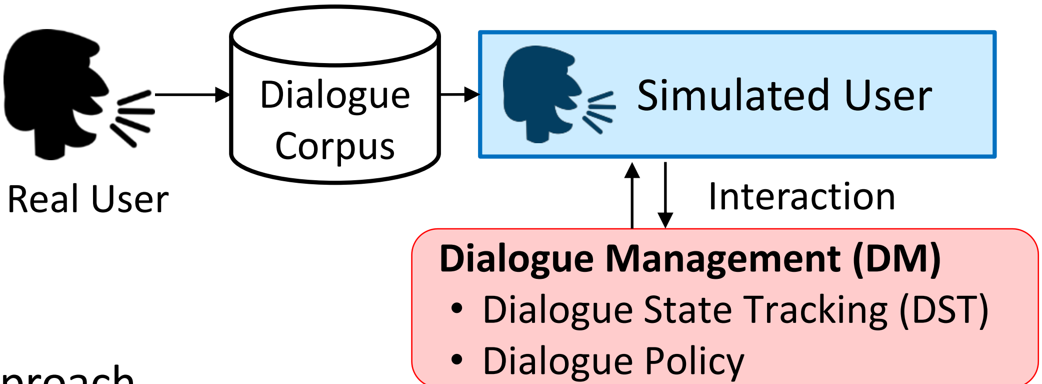
Q1	**Do you think you understand from the dialog what the user wanted?**
Opt	1) No clue 2) A little bit 3) Somewhat 4) Mostly 5) Entirely
Aim	*elicit the Worker's confidence in his/her ratings.*
Q2	**Do you think the system is successful in providing the information that the user wanted?**
Opt	1) Entirely 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

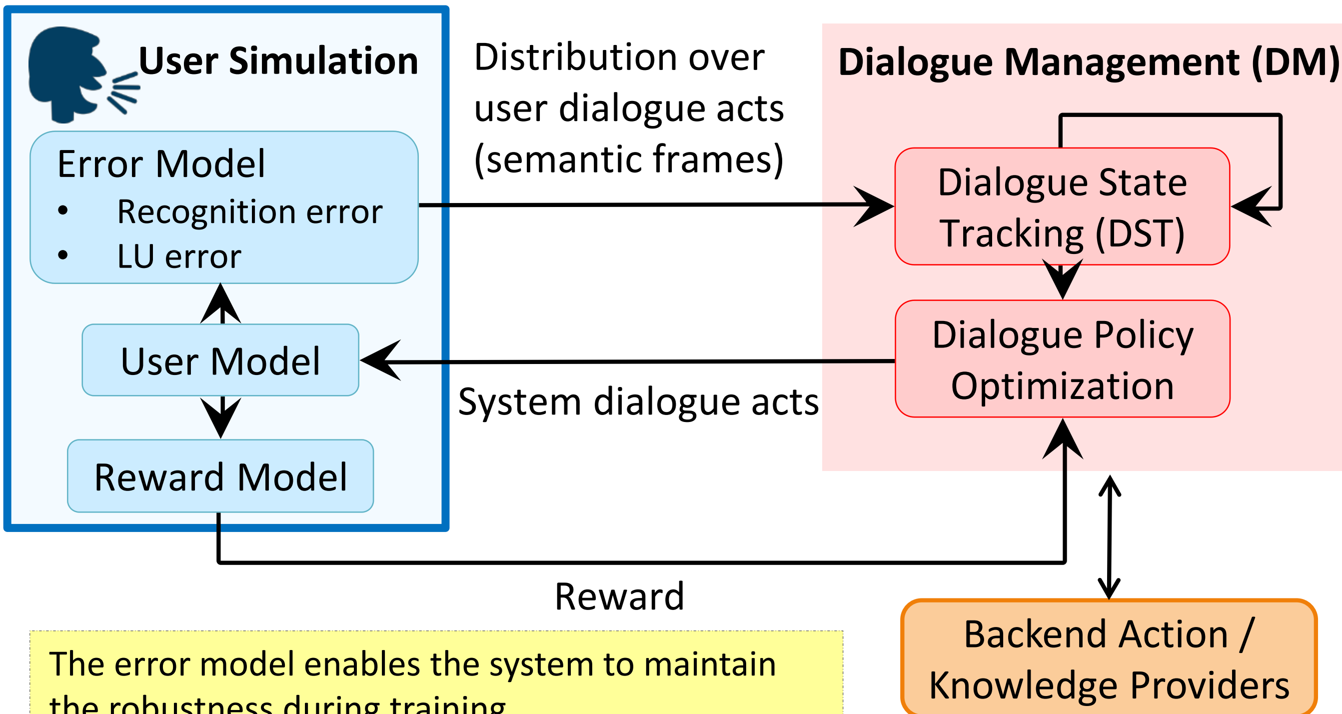
User Simulation

- 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



The error model enables the system to maintain the robustness during training

Rule-Based Simulator for RL Based System

(Li et.al., 2016)

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

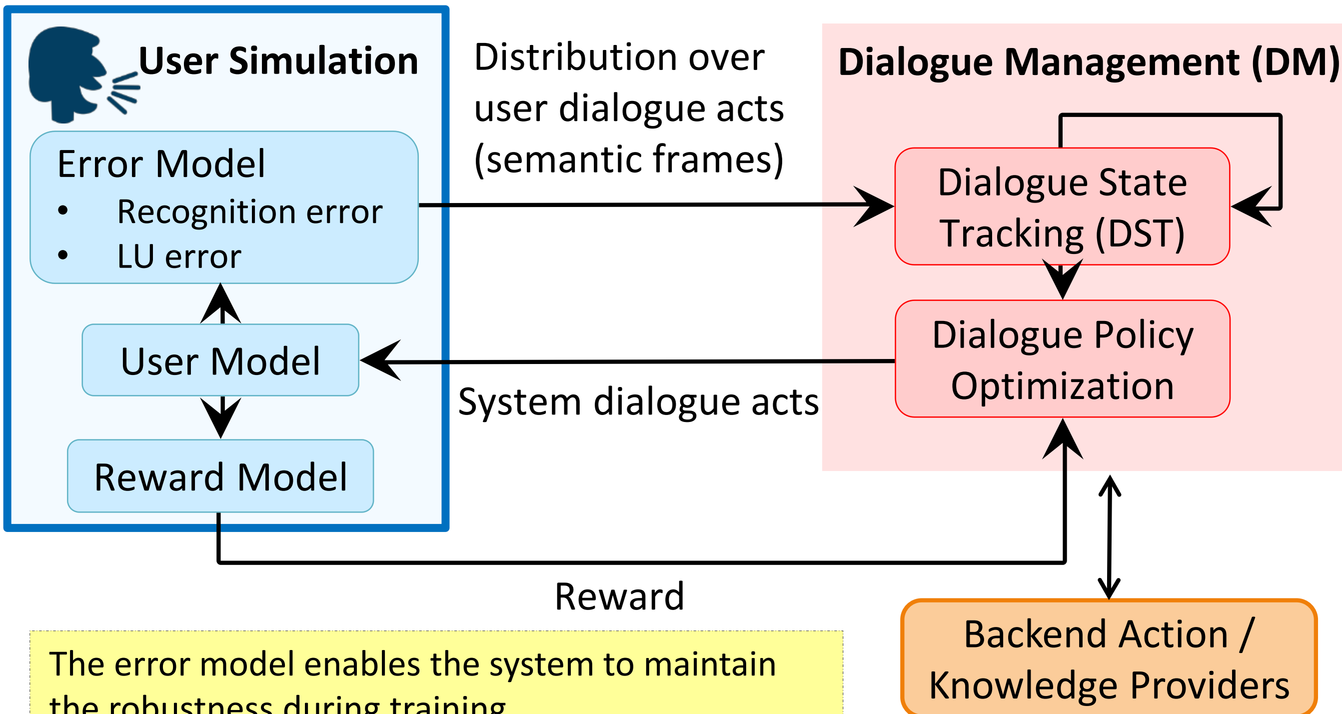
```python
class AgentDQN(Agent):
    def run_policy(self, representation):
        """ epsilon-greedy policy """

        if random.random() < self.epsilon:
            return random.randint(0, self.num_actions - 1)
        else:
            if self.warm_start == 1:
                if len(self.experience_replay_pool) > self.experience_replay_pool_size:
                    self.warm_start = 2
                return self.rule_policy()
            else:
                return self.dqn.predict(representation, {}, predict_model=True)

    def train(self, batch_size=1, num_batches=100):
        """ Train DQN with experience replay """

        for iter_batch in range(num_batches):
            self.cur_bellman_err = 0
            for iter in range(len(self.experience_replay_pool)/(batch_size)):
                batch = [random.choice(self.experience_replay_pool) for i in xrange(batch_size)]
                batch_struct = self.dqn.singleBatch(batch, {'gamma': self.gamma}, self.clone_dqn)
```

Elements of User Simulation



The error model enables the system to maintain the robustness during training

Rule-Based Simulator for RL Based System

(Li et.al., 2016)

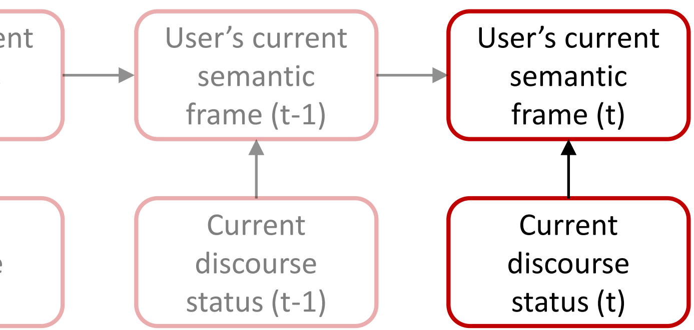
http://arxiv.org/abs/1612.05688

- 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)

- Three step process
  1) User intention simulator

request+search_loc



features (DD+DI)

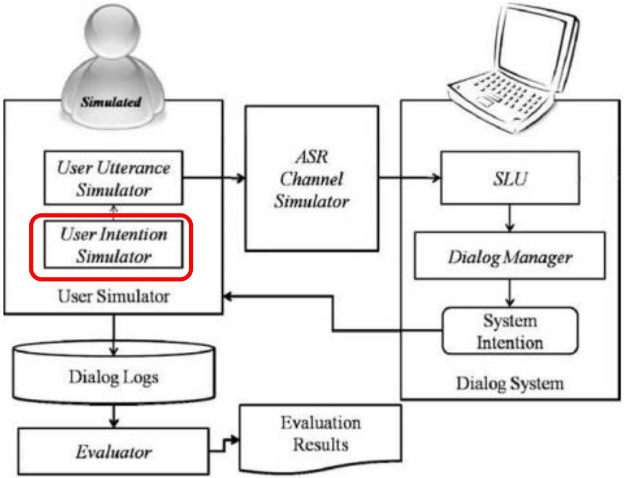


Fig. 1. Overall architecture of dialog simulation.

(*) 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)

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

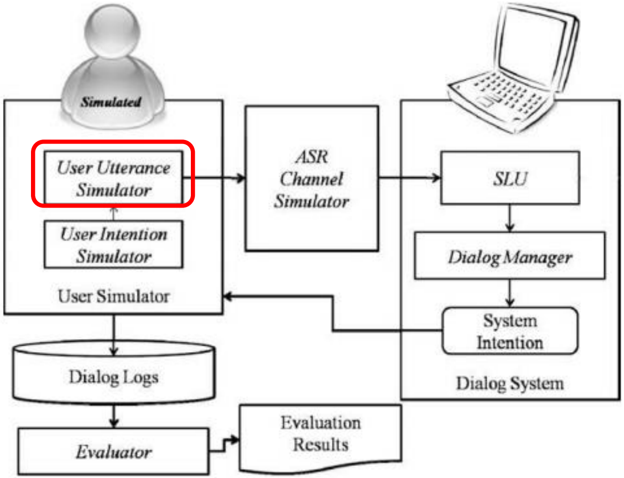


Fig. 1. Overall architecture of dialog simulation.

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)

- 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)

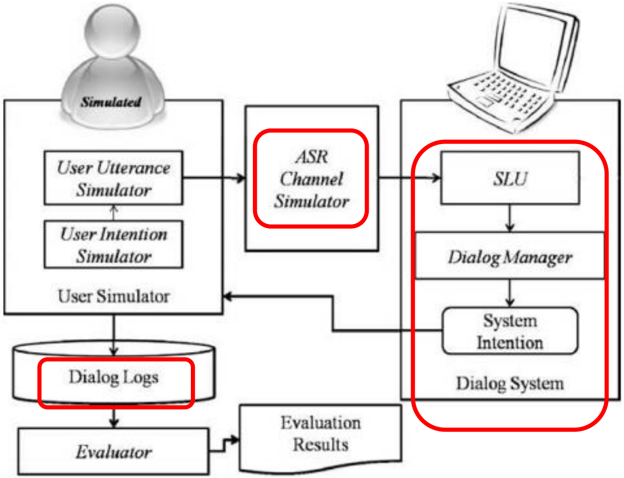
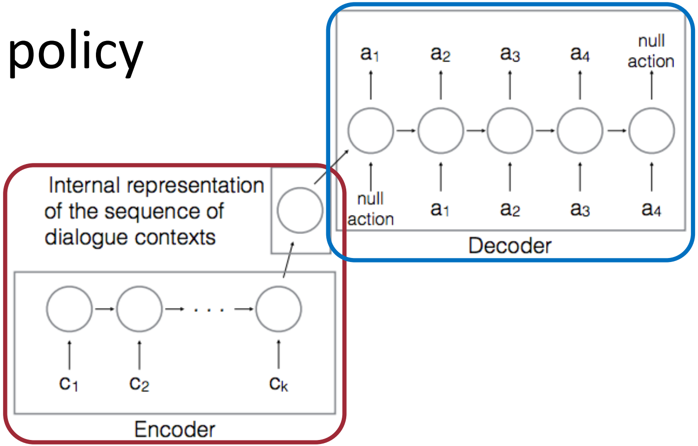


Fig. 1. Overall architecture of dialog simulation.

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

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

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)

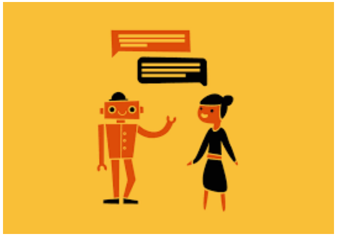
Action Appropriateness

- 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)

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)

- 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!*