CONVERSATIONAL UI

Dialog Management & Response Generation

Dr. Aobo Wang
Email: isswan@nus.edu.sg

© 2019 NUS. All rights reserved.

Agenda

- **Dialog Management**
  - **Interaction Strategies**
  - **Error Handling and Confirmation Strategies**
  - **Dialogue State Tracking**
  - **Dialogue policy**
- **Response Generation**
- **Summary**

© 2019 NUS. All rights reserved.

Dialog Management (DM)

- Decide “what to say” and “what to do”
- There is no universally agreed definition
- The complexity of DM depends on the specific tasks
- Largely responsible for user satisfaction

- DM Tasks:
  - Interaction Strategies
  - Error Handling and Confirmation Strategies
  - **Dialogue State Tracking**
  - Dialogue policy

© 2019 NUS. All rights reserved.

Dialog Management (DM)

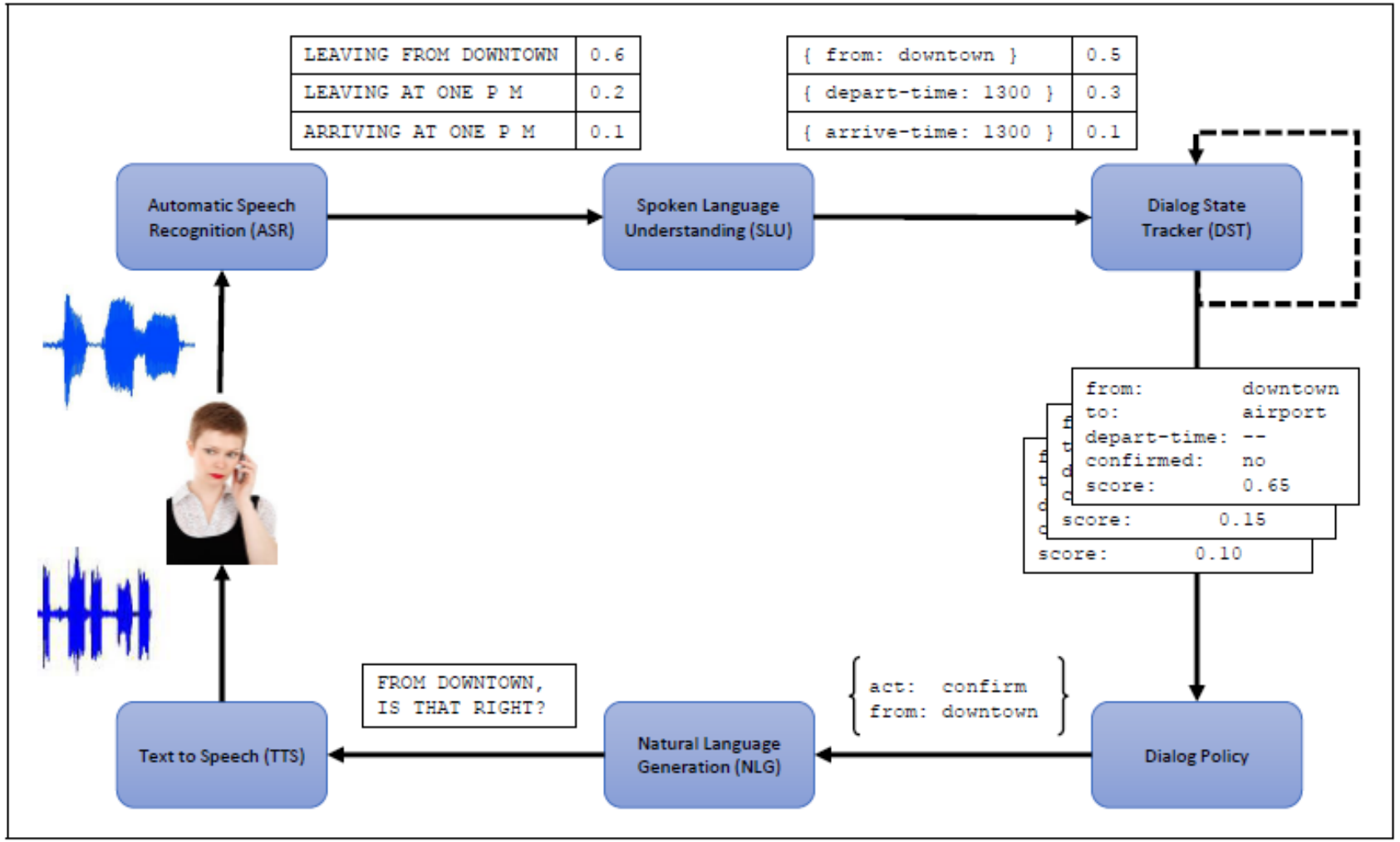


Figure 26.11 Architecture of a dialogue-state system for task-oriented dialogue from Williams et al. (2016).

© 2019 NUS. All rights reserved.

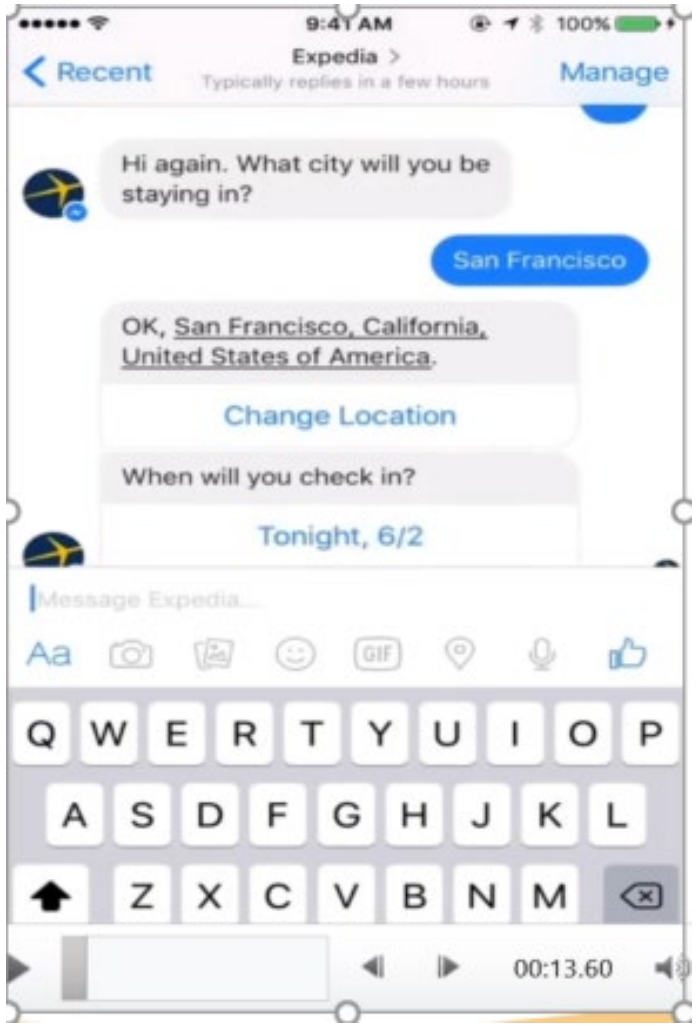
Interaction Strategies

Who takes the initiative in the dialog

© 2019 NUS. All rights reserved.

System-directed initiative

- Search and Booking Service
- Asking questions to navigate the conversation
- User just answers its queries
  - Efficient dialogs
  - Lack of flexibility

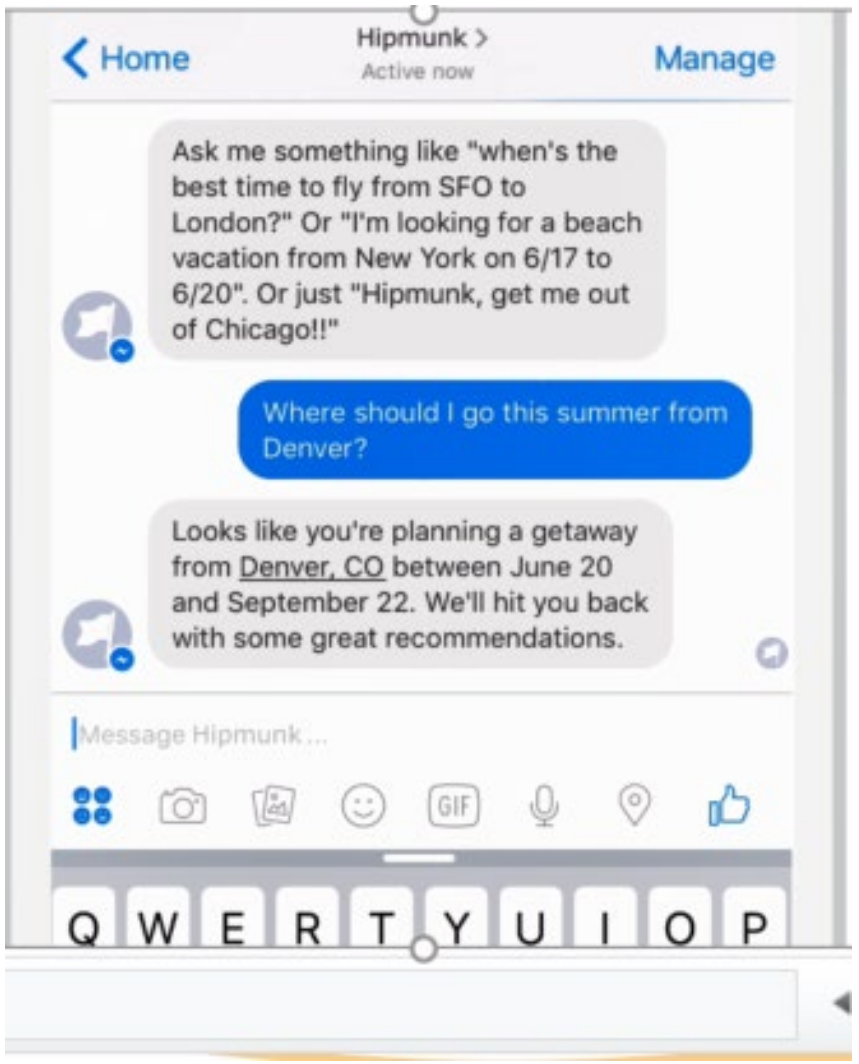


© 2019 NUS. All rights reserved.

User-directed initiative

- User has the initiative
- System responds to the user's queries
  - Natural and Flexible
  - Loss constrain

- Faked by user guide



© 2019 NUS. All rights reserved.

Mixed-initiative strategy

- Both the user and the system can take the initiative in the dialog

System: Do you want timetables for next Friday?
User: Are there trains before 6 in the morning?

© 2019 NUS. All rights reserved.

Mixed-initiative strategy

- Both the user and the system can take the initiative in the dialog
- User can take the initiative by
  - asking questions
  - introducing new topics
  - providing over-informative responses

System: Do you want timetables for next Friday?
User: Are there trains before 6 in the morning?

- Maintain and monitor the dialog history and the system's agenda

© 2019 NUS. All rights reserved.

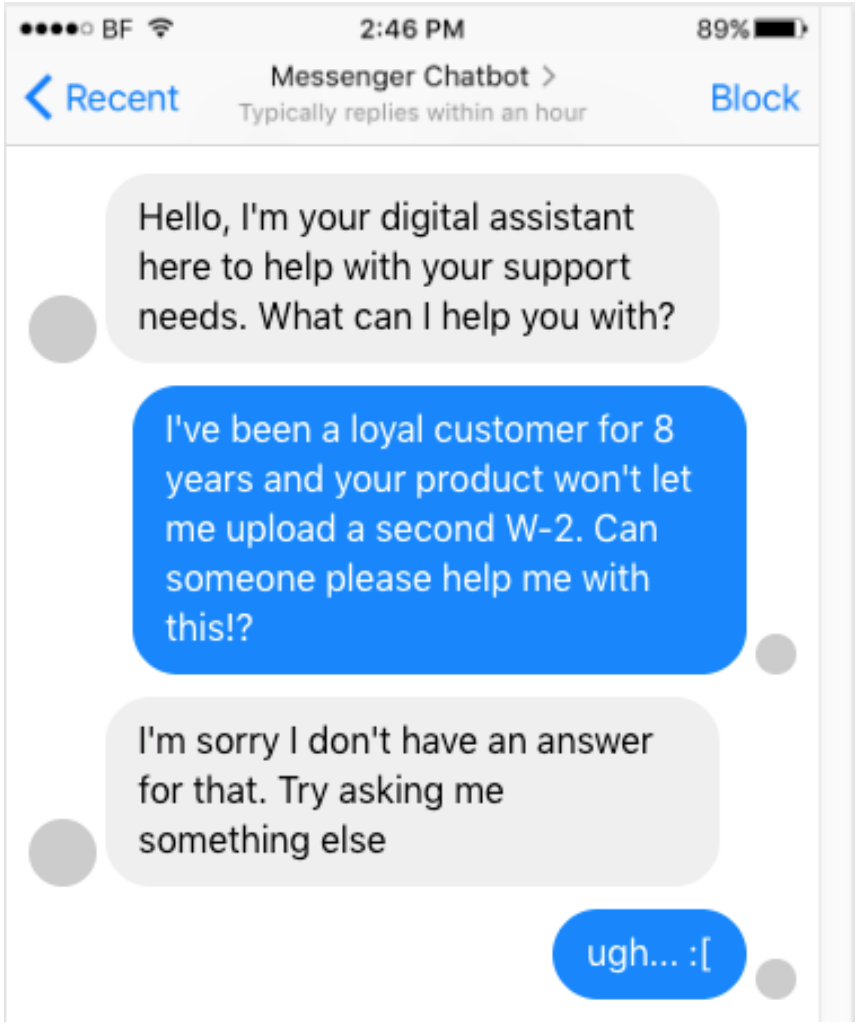
Error Handling and Confirmation Strategies

© 2019 NUS. All rights reserved.

When Things Are Not Working

- Ability to understand complicated/unseen utterances

  *“ I’ve been a loyal customer for 8 years and your product won’t let me **upload a second W-2**. Can someone please help me with this!?”*



© 2019 NUS. All rights reserved.

Error Handling and Confirmation Strategies

User: I want to know timetables from Madrid.
System: Do you want to leave from Madrid?
User: Yes.

© 2019 NUS. All rights reserved.

Error Handling and Confirmation Strategies

- Intents and slots could be uncertain or ambiguous
- Errors from ASR and NLU might be propagated
- Confidence scores with a threshold
  - below a threshold indicates confirmation or even rejection needed

User: I want to know timetables from Madrid.
System: Do you want to leave from Madrid?
User: Yes.

- explicit confirmation
- dialog tends to be lengthy

© 2019 NUS. All rights reserved.

Error Handling and Confirmation Strategies

- over-informative responses are challenging

> User: I want to know timetables from Madrid.
> System: What time do you want to leave from Madrid?

> User: No, I just wanted to know about times from Madrid but I might be departing from somewhere else depending on whether I have the use of the car next Friday.

© 2019 NUS. All rights reserved.

When Things Are Not Working

- Ability to understand complicated/unseen utterances

  *" I've been a loyal customer for 8 years and your product won't let me **upload a second W-2**. Can someone please help me with this!?"*

- **Confirmation Strategy**
- **Sentiment Analysis**
- **Switch to Human Assistance**

© 2019 NUS. All rights reserved.

Dialog Management

- DM Tasks:
  - Interaction Strategies
  - Error Handling and Confirmation Strategies
  - **Dialogue State Tracking**
    - Handcrafted Approaches
    - Statistical approaches

© 2019 NUS. All rights reserved.

Dialogue State Tracking

The system's belief of "what the user wants"

© 2019 NUS. All rights reserved.

Dialog State Tracking

- Example of the Output of DST
  - **Cross turn**

User: I'm looking for a cheaper restaurant
→ `inform(price=cheap)`
System: Sure. What kind - and where?
User: Thai food, somewhere downtown
→ `inform(price=cheap, food=Thai, area=centre)`
System: The House serves cheap Thai food
User: Where is it?
→ `inform(price=cheap, food=Thai, area=centre); request(address)`
System: The House is at 106 Regent Street

© 2019 NUS. All rights reserved.

Dialog State Tracking

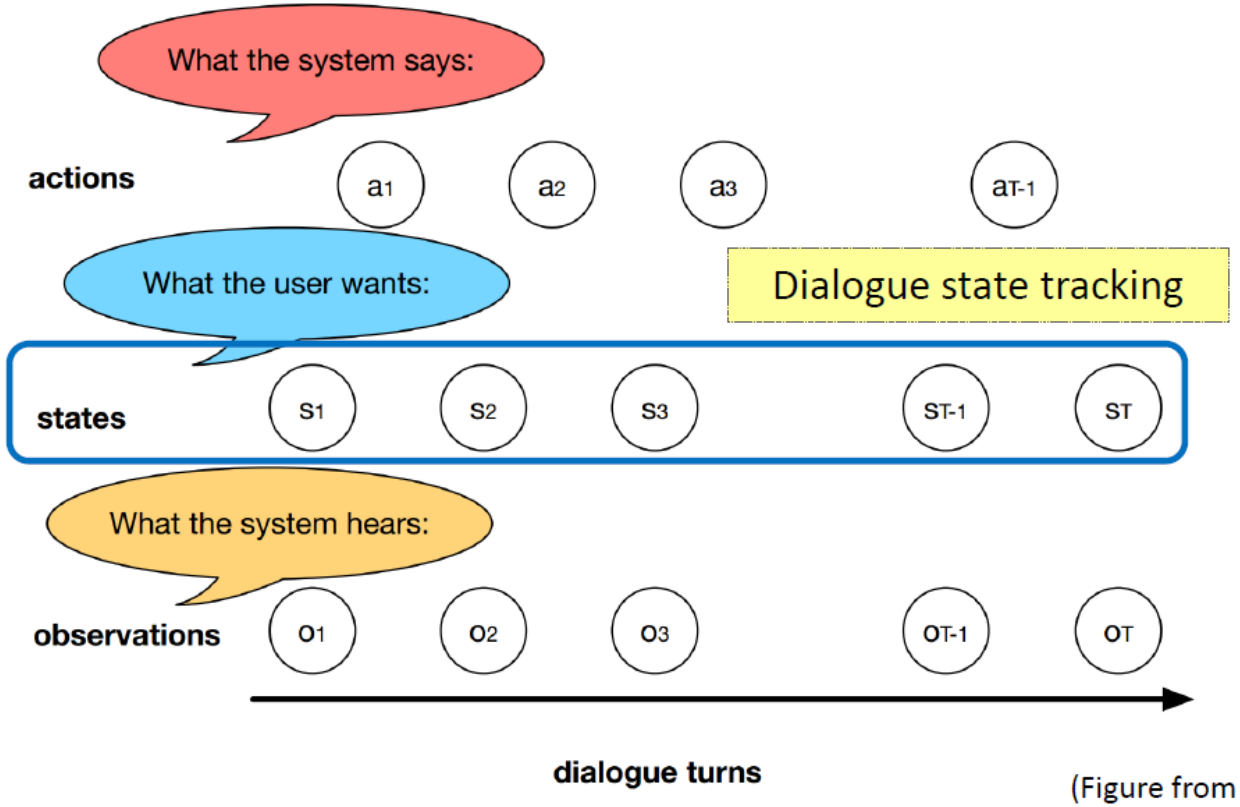
- Pre-defined Domain Ontology

Slots	Values	*informable*	*Requestable*
FoodType	*local/Indian/Chinese*	Y	**N**
RestName	*KFC/LongBeach/SushiHouse*	Y	Y
PriceRange	***Any/Cheap/Medium/Expensive***	Y	**N**
Area	*East/West/Center/North/South*	Y	**N**
NumOfGuests	*any integer*	Y	**N**
Address	*address of restaurant*	**N**	Y

© 2019 NUS. All rights reserved.

Dialog State Tracking

- Layers of DST
- System's internal representation of the state of the conversation
- The system's belief of "what the user wants" at each turn



(Figure from Gašić)

© 2019 NUS. All rights reserved.

Dialog State Tracking

- Pre-defined Action List
  - What is the user's request?
  - Then what should the system do?

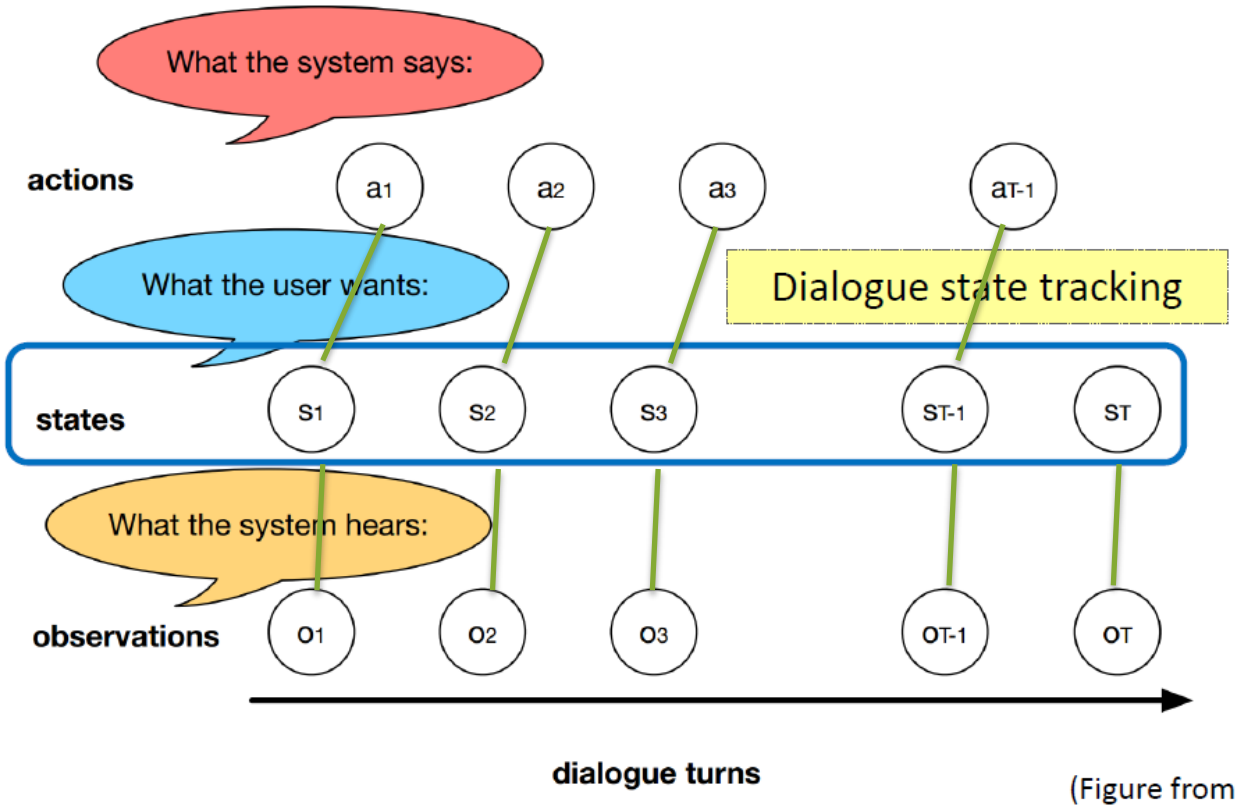
act field	slots field	Description
ack	empty list	An acknowledgement e.g. "okay"
affirm	empty list	An affirmation e.g. "yes"
bye	empty list	Trying to end the dialog e.g. "good-bye"
hello	empty list	Greeting the system e.g. "hi"
help	empty list	Trying to solicit general help from the system e.g. "what can I say?"
negate	empty list	A negation e.g. "no"
null	empty list	Something not understandable to the system; outside its domain e.g. "pineapple"
repeat	empty list	A request for the system to repeat what it just said e.g. "please repeat that"
reqalts	empty list	Requesting for alternative suggestions e.g. "are there any others"
reqmore	empty list	Asking for more information in general e.g. "tell me more"
restart	empty list	Asking the system to start' from the beginning e.g. "let's start again"
silence	empty list	User actually said nothing
thankyou	empty list	User thanking the system e.g. "thanks"

confirm	one slot, value pair- (s, v)	s must be an informable slot and v a possible value for s as specified in the ontology. This corresponds to the user confirming that the constraint $s = v$ has been understood. E.g. "Is it in the west?" is a case with $(s, v) =$ ("area", "west").
deny	one slot, value pair- (s, v)	s must be an informable slot and v a possible value for s as specified in the ontology. This is the user saying their goal for s is not v. E.g. "I don't want something in the west"
inform	one slot, value pair- (s, v)	Again s must be an informable slot and v a possible value for s as specified in th ontology. This is the user specifying their goal for s as v. E.g. "It must be in the west"
request	one pair: ("slot", s)	s must be requestable according to the ontology. This is the user asking for the value of s from the system E.g. "what part of town is it?"

© 2019 NUS. All rights reserved.

Dialog Status Tracking

- Handcrafted Approaches
  - Define s_i based on domain knowledge
  - Determine s_i based on NLU
  - Define actions a_i attached to the States s_i



(Figure from Gašić)

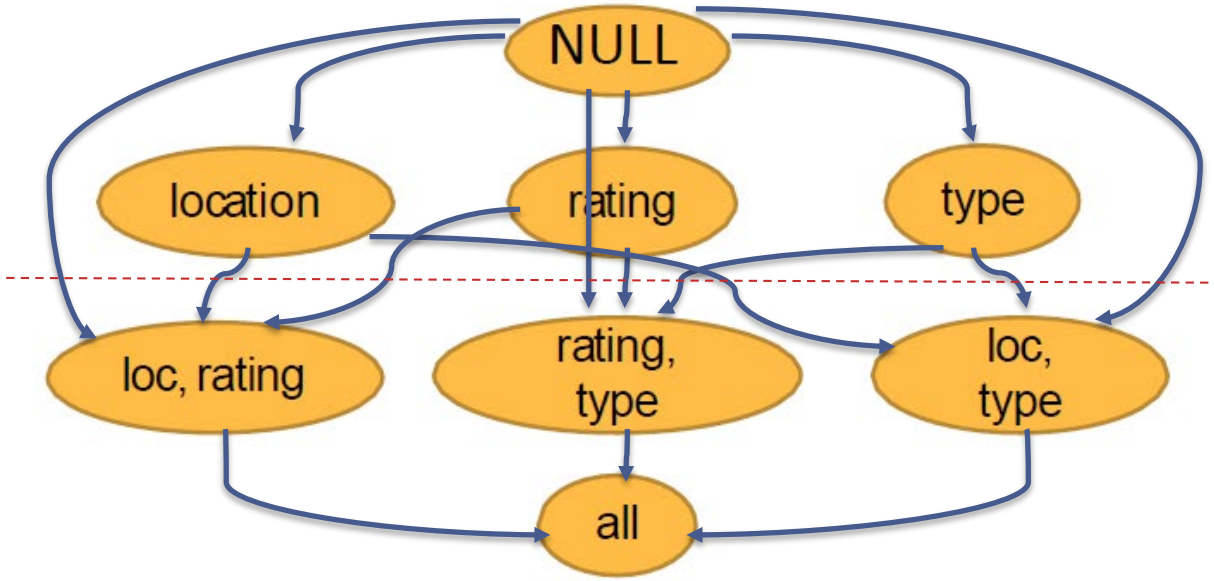
© 2019 NUS. All rights reserved.

Handcrafted Approaches

- Task-oriented Scenario

Intelligent Agent

Handcrafted States Diagram



Actions

AskNext

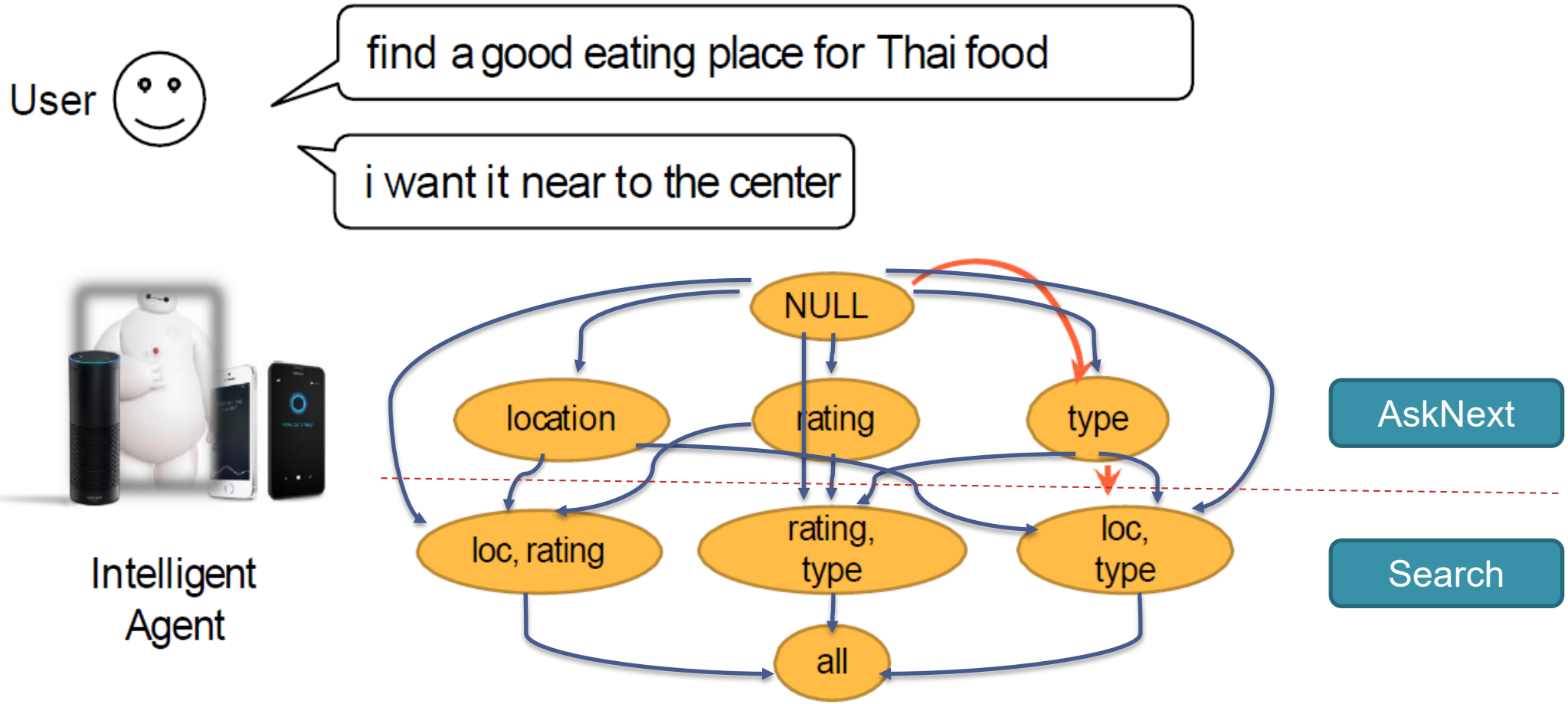
Search

https://sites.google.com/view/deepdial/

© 2019 NUS. All rights reserved.

Handcrafted Approaches

- Restaurant searching scenario



https://sites.google.com/view/deepdial/

© 2019 NUS. All rights reserved.

Handcrafted Approaches

- Action Types
  - Task-independent behaviours
    - error correction and confirmation
  - Task-specific behaviours
    - logic associated actions e.g., *search, book, send*
  - Task interface behaviours
    - e.g., *prompt selection*

© 2019 NUS. All rights reserved.

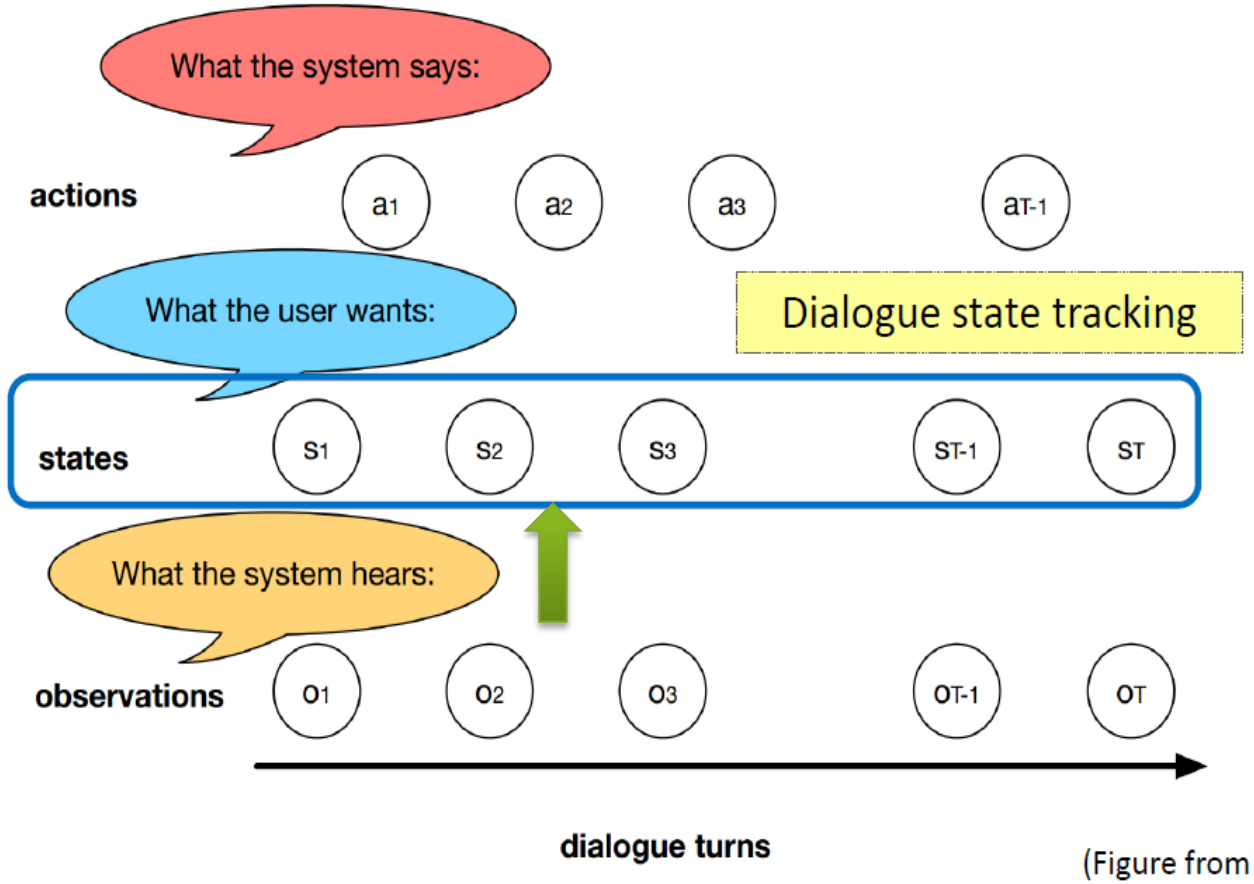
Handcrafted Approaches

- Efficient and Accurate
- Suitable for narrow domain problem
- **Avoid pretending to be smart**
- Challenging to anticipate every possible flows
- Effort to literately refine and tune the dialog strategies
- Non-transferable to new domain

© 2019 NUS. All rights reserved.

Statistical Approaches

- Research Topic
  - Data driven
  - Better scalability
  - Probability distribution



(Figure from Gašić)

https://sites.google.com/view/deepdial/

© 2019 NUS. All rights reserved.

Statistical Approaches

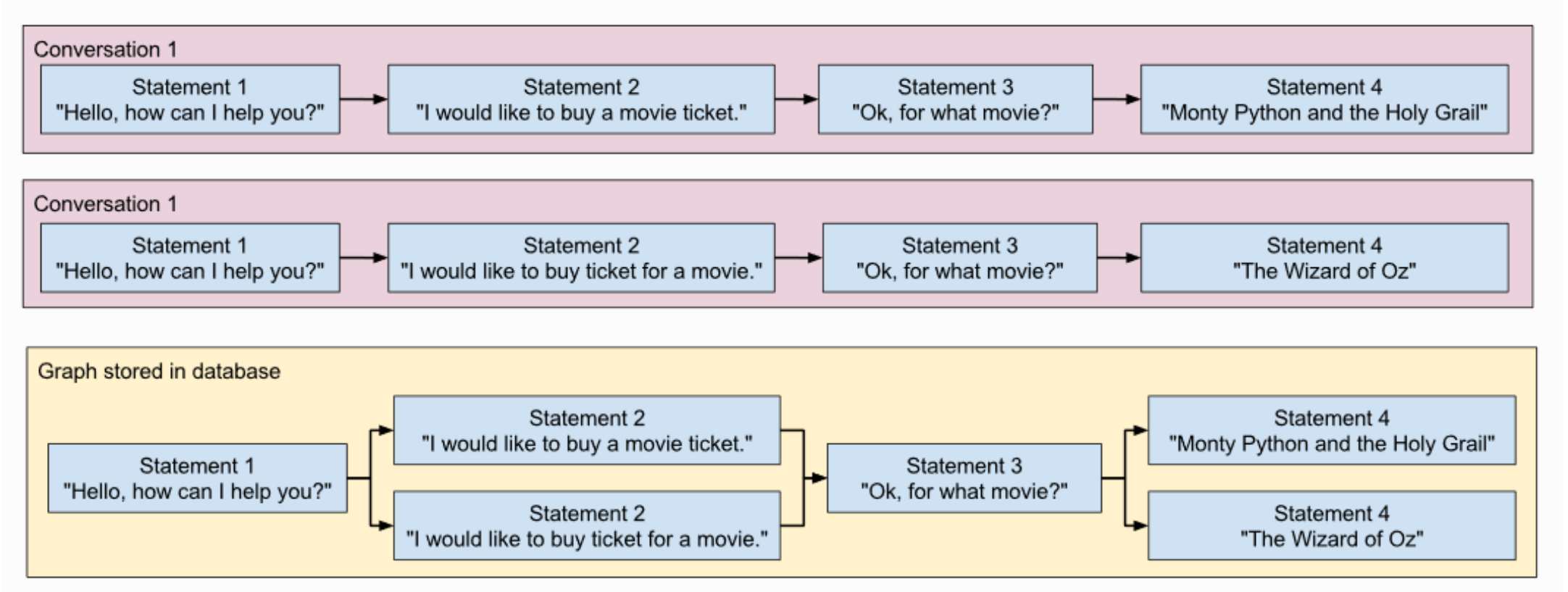
- **Example-based** approaches-- Chatterbot

```
from chatterbot.trainers import ListTrainer

conversation = [
    "Hello",
    "Hi there!",
    "How are you doing?",
    "I'm doing great.",
    "That is good to hear",
    "Thank you.",
    "You're welcome."
]

trainer = ListTrainer(chatbot)

trainer.train(conversation)
```

© 2019 NUS. All rights reserved.

Statistical Approaches

- Example-based approaches -- Chatterbot
  - Finite State Machines



© 2019 NUS. All rights reserved.

Statistical Approaches

- Recent Research Topic
  - Data driven
  - Adaptation to deal with new domains (if **data** available in new domains)
  - Better scalability (if more **data** available)
  - Improved probability distribution
  - Joint NLU with DST
  - Heavily depending on the quality and coverage of **labeled data**
  - Time consuming to collect corpus and annotations

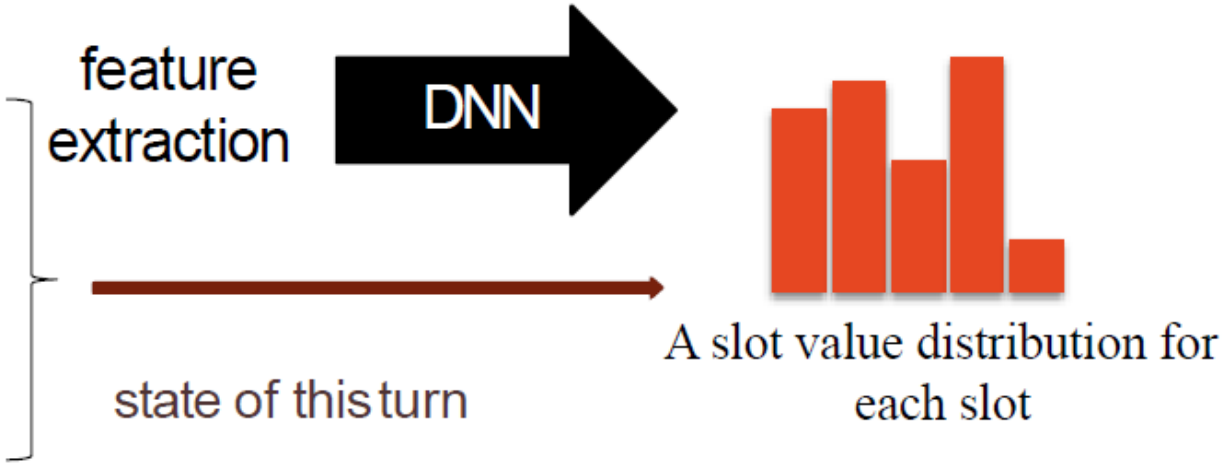
https://sites.google.com/view/deepdial/

© 2019 NUS. All rights reserved.

Statistical Approaches

- NN for DST

multi-turn conversation



Slot Name	Slot Value	Prob.
food	Thai	0.95
food	Chinese	0.01
area	center	0.99
…	…	…

https://sites.google.com/view/deepdial/

© 2019 NUS. All rights reserved.

Statistical Approaches

- Joint NLU with DST

"I am looking for good pizza"

FoodType	Local
FoodType	Indian
FoodType	Chinese
FoodType	ITALIAN
PriceRange	Cheap
…	…

FoodType	Local
FoodType	Indian
FoodType	Chinese
FoodType	ITALIAN
PriceRange	Cheap
…	…

  - The action types, slots and values are predefined

© 2019 NUS. All rights reserved.

Statistical Approaches (I)

- **Word-based RNN**

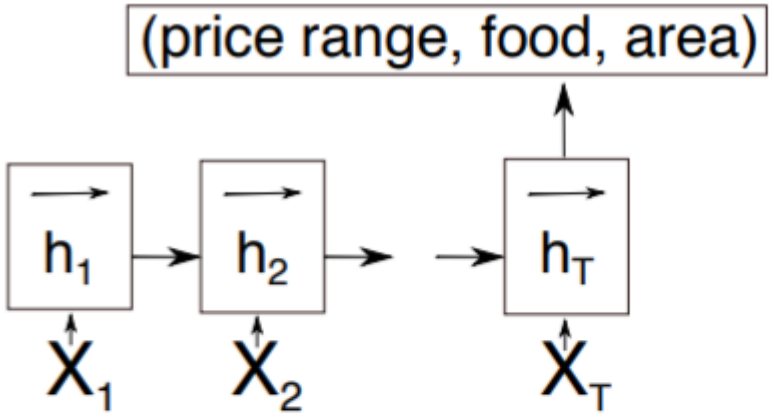


Figure 2: The joint label predictions using RNN from last hidden state h_T. The h_T represents the whole dialog history of T words. The RNN takes as input for each word i an embedding and binary features concatenated to vector X_i.

System: *What part of town do you have in mind?*
User: *West part of town.*
System: *What kind of food would you like?*
User: *Indian*

X_1= **f**(What)
X_2= **f**(part)
...
X_T= **f**(Indian)

wordvec for word	U	S	IsF	IsR	IsL	IsP
1.2,2.4,5.9,0.1,9.1	1	0	1	1	0	0

X_T= **f**(*Indian*)

© 2019 NUS. All rights reserved.

Statistical Approaches (I)

- Delexicalisation

"I am looking for good pizza"

FoodType	Local
FoodType	Indian
FoodType	Chinese
FoodType	ITALIAN
PriceRange	Cheap
…	…

- ***construct semantic dictionaries***

FOOD=CHEAP: [affordable, budget, low-cost, low-priced, inexpensive, cheaper, economic, ...]

RATING=HIGH: [best, high-rated, highly rated, top-rated, cool, chic, popular, trendy, ...]

AREA=CENTRE: [center, downtown, central, city centre, midtown, town centre, ...]

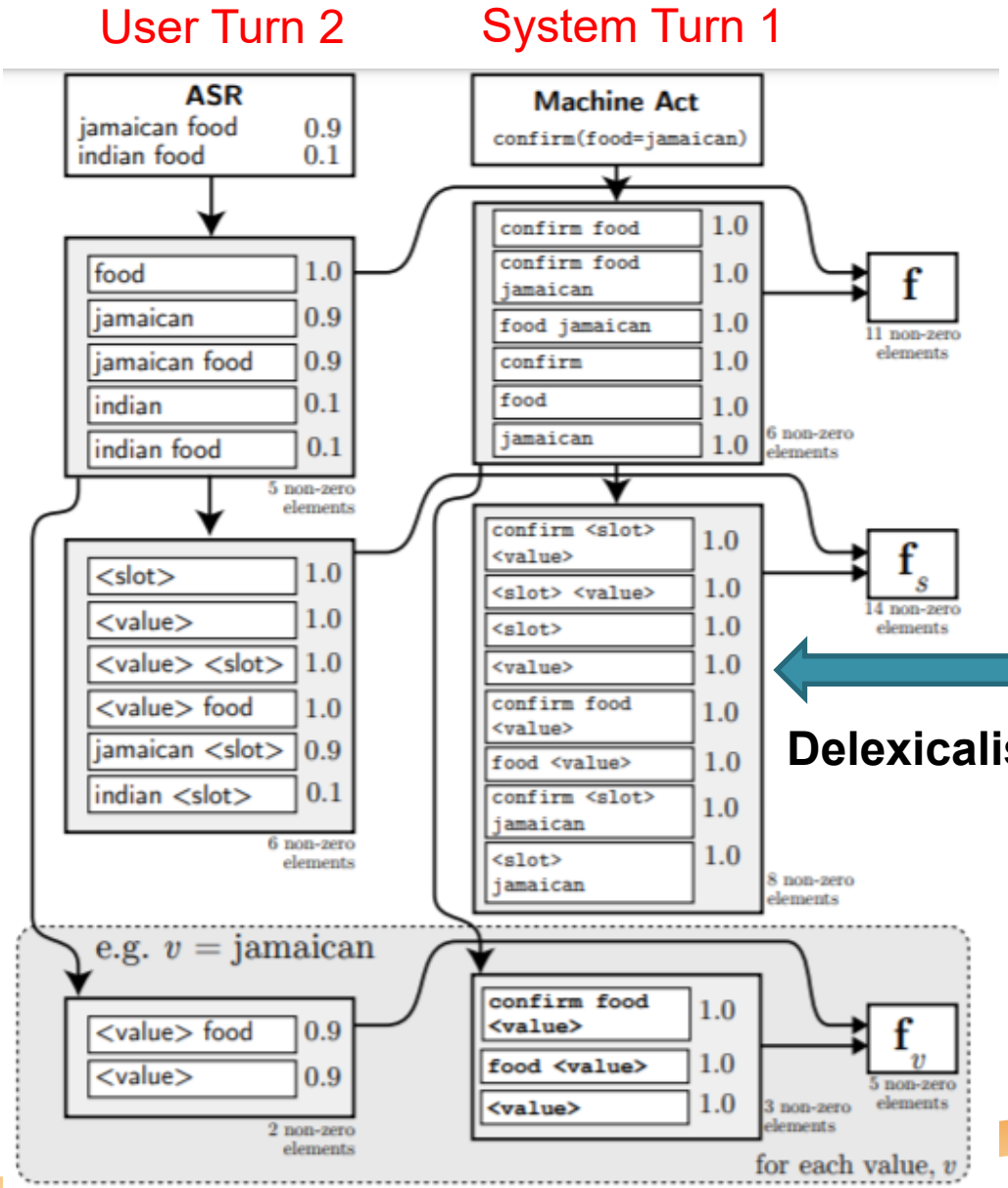
– **Delexicalisation** whereby slots and values mentioned are replaced with generic labels

- [*want tagged-value food*].

© 2019 NUS. All rights reserved.

Statistical Approaches (I)

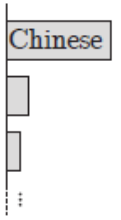
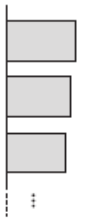
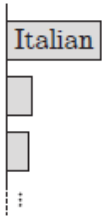
- Word-Based RNN with **Delexicalisation**
  - **n-grams** from utterances
  - **n-grams** from **dialog acts**
    - acttype(slot=value)



© 2019 NUS. All rights reserved.

Statistical Approaches (I)

- Word-Based RNN with **Delexicalisation**
  - One such RNN is used per slot
  - Slot = Food
  - the most recent user input
  - the most recent machine dialog act
  - Predict the likelihood for all pairs of (value, slot)

Dialog Turn	y^{init}	Notes
System: What type of food would you like? *User*: Chinese food.		Here an initial model is likely to output a confident low entropy distribution correctly identifying the food goal as 'Chinese'.
System: There are no matching Chinese restaurants. *User*: Any serving pizza?		The system has requested the food slot, and the user's response included the term 'serving'. This gives evidence that the user has informed the food slot, but the system cannot recognise which food type is correct. Therefore it is likely that an initial model would output a high entropy distribution for the food slot.
System: Sorry, what type of food would you like? *User*: Um, Italian food.		If the user explicitly says 'Italian', which the system is able to match in its ontology, then an initial model can predict with high confidence the correct value is food.

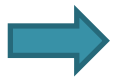
© 2019 NUS. All rights reserved.

Statistical Approaches (II)

- Neural Belief Tracker: Data-Driven

  - No more feature engineering or Delexicalisation with lexicon resources
  - Couple SLU/NLU with DST
  - Fully based on Pre-trained wordvectors
  - **Match** the performance of delexicalisation-based models
  - better-suited to **scaling** where the creation of such domain-specific lexicons would be infeasible

wordvec for word	U	S	IsF	IsR	IsL	IsP
1.2,2.4,5.9,0.1,9.1	1	0	1	1	0	0

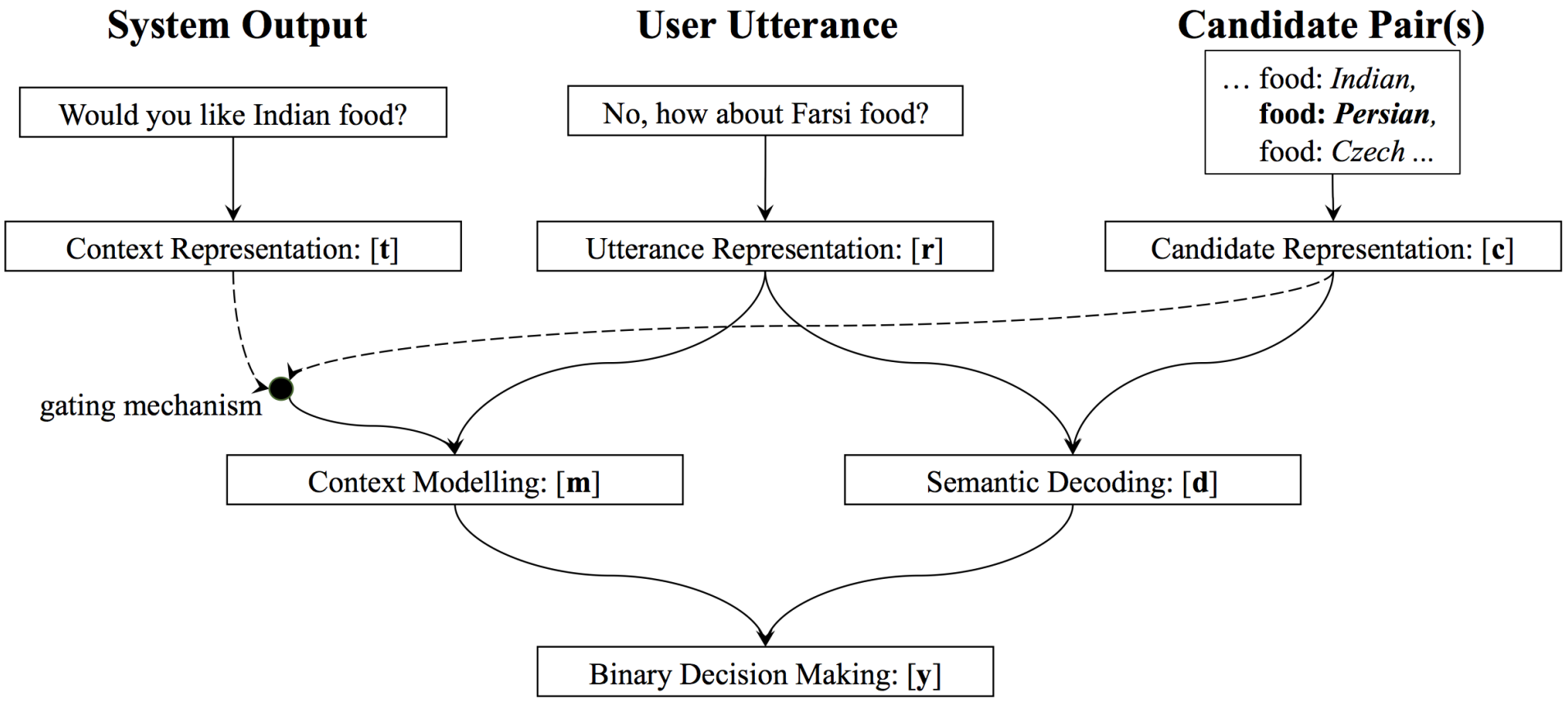


wordvec for context
3.2 4.2 9.6 9.1 6.2 8.4 0.9 0.3 7.1

© 2019 NUS. All rights reserved.

Statistical Approaches (II)

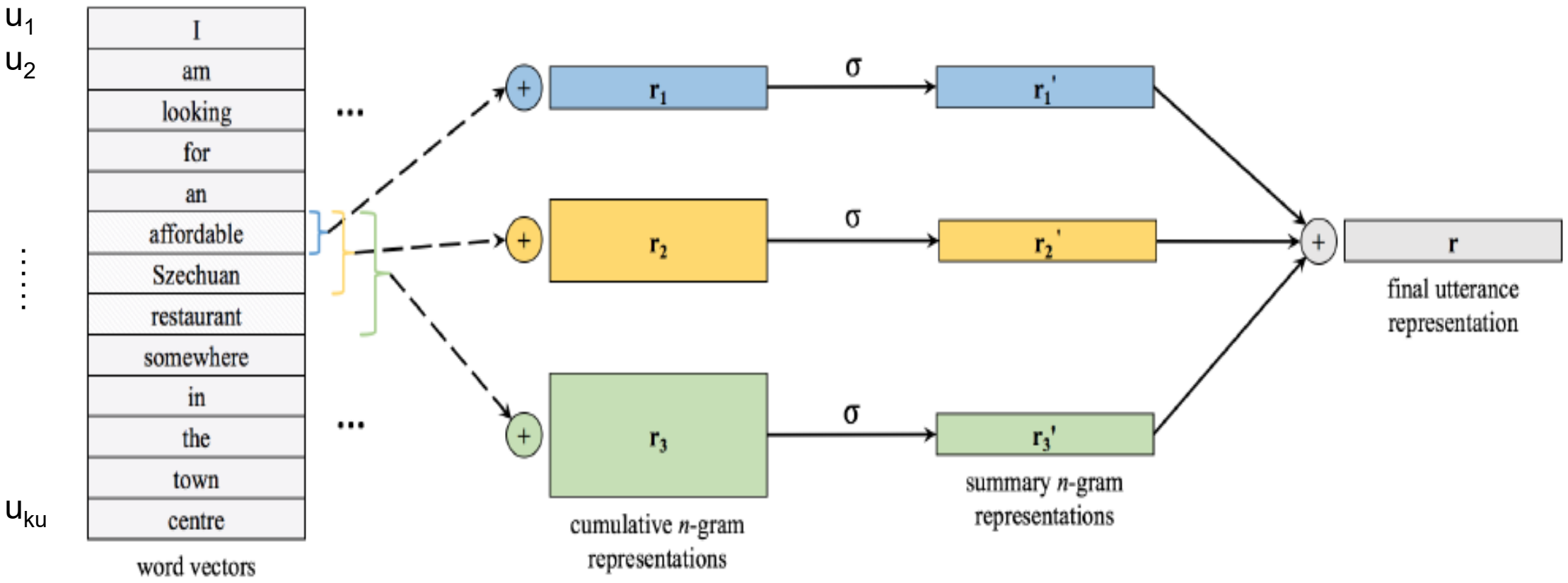
- Neural Belief Tracker: Data-Driven



© 2019 NUS. All rights reserved.

Statistical Approaches (II)

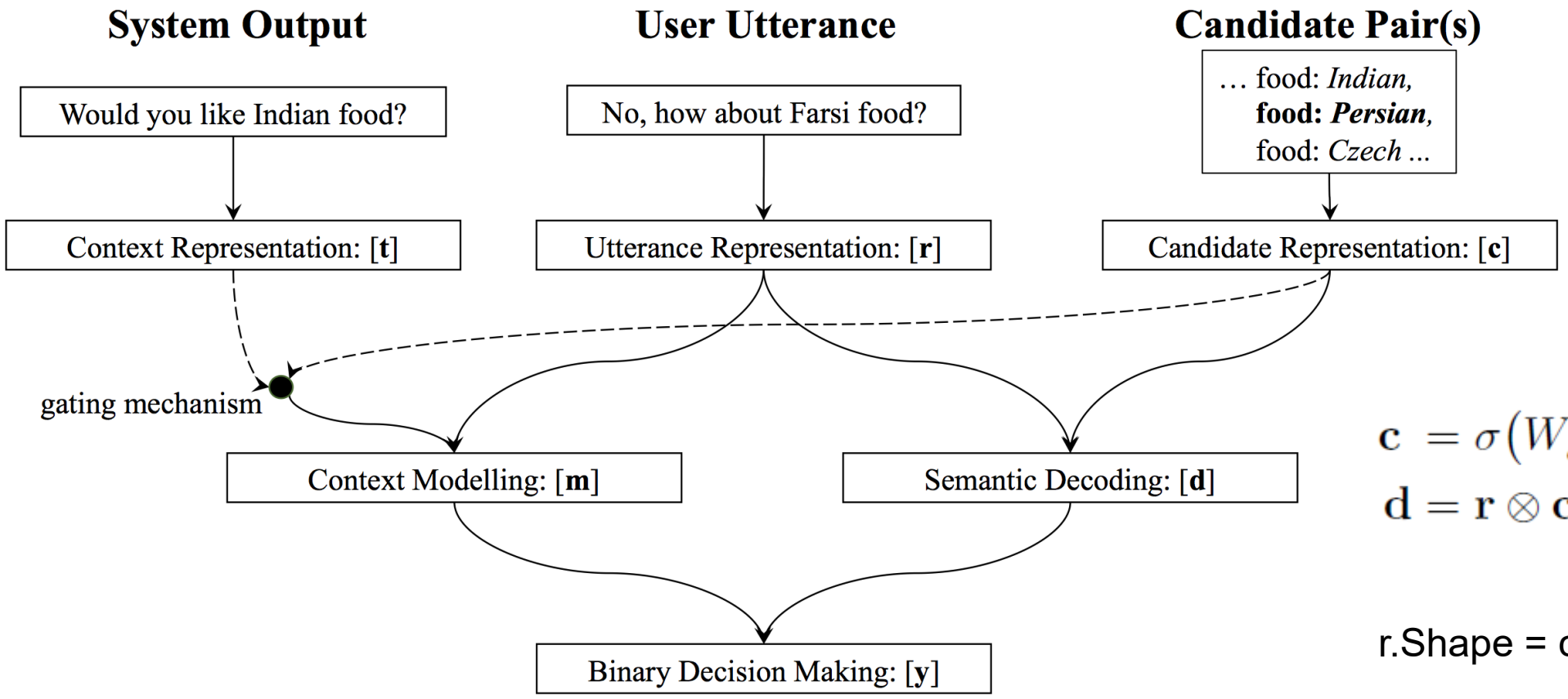
- Neural Belief Tracker: Utterance representation



© 2019 NUS. All rights reserved.

Statistical Approaches (II)

- Neural Belief Tracker: Semantic Decoding



$$\mathbf{c} = \sigma\left(W_c^s(\mathbf{c_s} + \mathbf{c_v}) + b_c^s\right)$$

$$\mathbf{d} = \mathbf{r} \otimes \mathbf{c}$$

r.Shape = c.shape = d.shape

© 2019 NUS. All rights reserved.

Statistical Approaches (II)

- Neural Belief Tracker: Context modelling
  - System Acts
    - System Requests : *"what price range would you prefer?"* → t_q
    - System Confirmation : *"how about thai food ? "* → (t_s , t_v)

$$\mathbf{m_r} = (\mathbf{c_s} \cdot \mathbf{t_q})\mathbf{r}$$

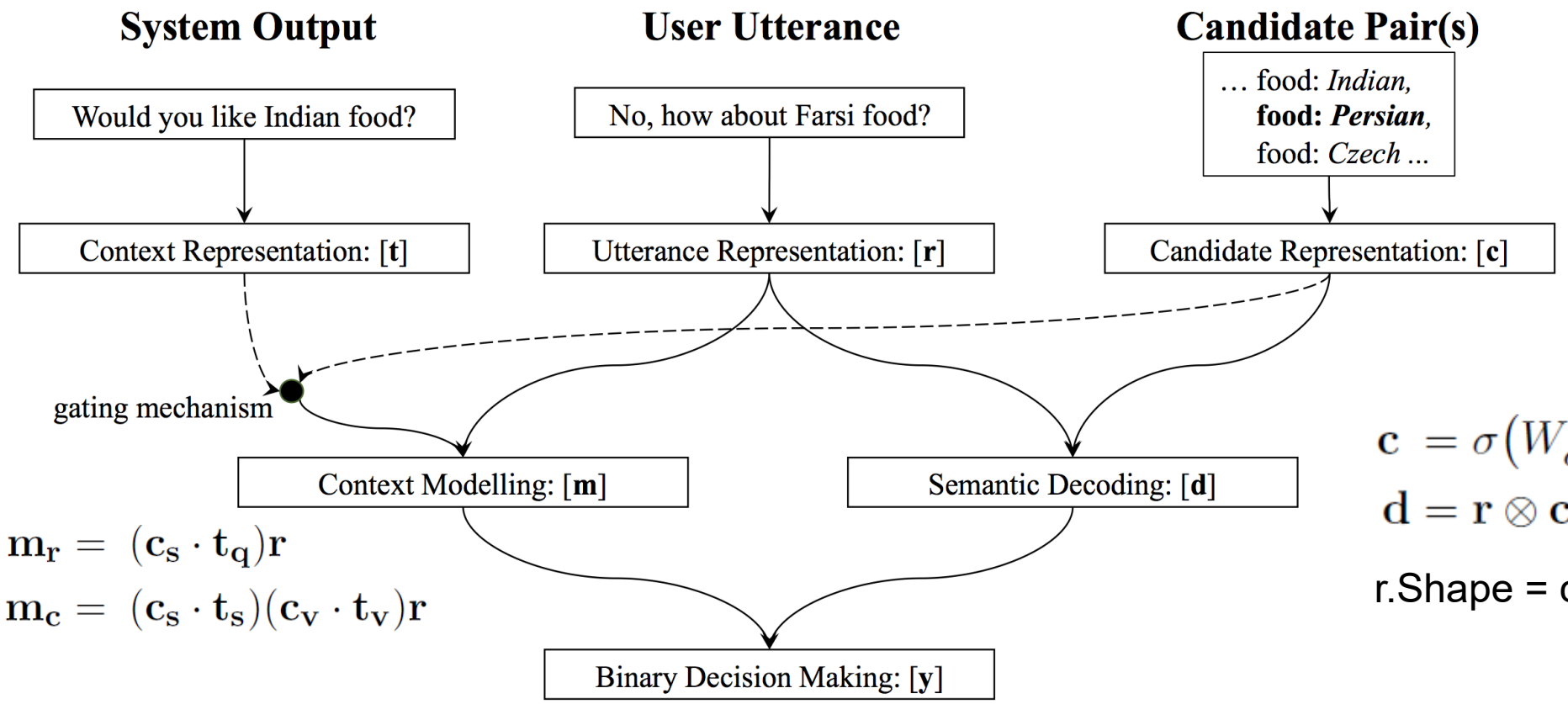
$$\mathbf{m_c} = (\mathbf{c_s} \cdot \mathbf{t_s})(\mathbf{c_v} \cdot \mathbf{t_v})\mathbf{r}$$

https://sites.google.com/view/deepdial/

© 2019 NUS. All rights reserved.

Statistical Approaches (II)

- Neural Belief Tracker: Binary Decision Making



$$\mathbf{m_r} = (\mathbf{c_s} \cdot \mathbf{t_q})\mathbf{r}$$

$$\mathbf{m_c} = (\mathbf{c_s} \cdot \mathbf{t_s})(\mathbf{c_v} \cdot \mathbf{t_v})\mathbf{r}$$

$$\mathbf{c} = \sigma\left(W_c^s(\mathbf{c_s} + \mathbf{c_v}) + b_c^s\right)$$

$$\mathbf{d} = \mathbf{r} \otimes \mathbf{c}$$

r.Shape = c.shape = d.shape

$$\mathbf{y} = \phi_2\left(\phi_{100}(\mathbf{d}) + \phi_{100}(\mathbf{m_r}) + \phi_{100}(\mathbf{m_c})\right)$$

© 2019 NUS. All rights reserved.

More about DST

Dialogue Acts

https://sites.google.com/view/deepdial/

© 2019 NUS. All rights reserved.

More about DST

- Action List

act field	slots field	Description
ack	empty list	An acknowledgement e.g. "okay"
affirm	empty list	An affirmation e.g. "yes"
bye	empty list	Trying to end the dialog e.g. "good-bye"
hello	empty list	Greeting the system e.g. "hi"
help	empty list	Trying to solicit general help from the system e.g. "what can I say?"
negate	empty list	A negation e.g. "no"
null	empty list	Something not understandable to the system; outside its domain e.g. "pineapple"
repeat	empty list	A request for the system to repeat what it just said e.g. "please repeat that"
reqalts	empty list	Requesting for alternative suggestions e.g. "are there any others"
reqmore	empty list	Asking for more information in general e.g. "tell me more"
restart	empty list	Asking the system to start' from the beginning e.g. "let's start again"
silence	empty list	User actually said nothing
thankyou	empty list	User thanking the system e.g. "thanks"

confirm	one slot, value pair- (s, v)	s must be an informable slot and v a possible value for s as specified in the ontology. This corresponds to the user confirming that the constraint $s = v$ has been understood. E.g. "Is it in the west?" is a case with $(s, v) =$ ("area", "west").
deny	one slot, value pair- (s, v)	s must be an informable slot and v a possible value for s as specified in the ontology. This is the user saying their goal for s is not v. E.g. "I don't want something in the west"
inform	one slot, value pair- (s, v)	Again s must be an informable slot and v a possible value for s as specified in th ontology. This is the user specifying their goal for s as v. E.g. "It must be in the west"
request	one pair: ("slot", s)	s must be requestable according to the ontology. This is the user asking for the value of s from the system E.g. "what part of town is it?"

© 2019 NUS. All rights reserved.

Dataset for DST

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

https://sites.google.com/view/deepdial/

© 2019 NUS. All rights reserved.

Dialog Policy

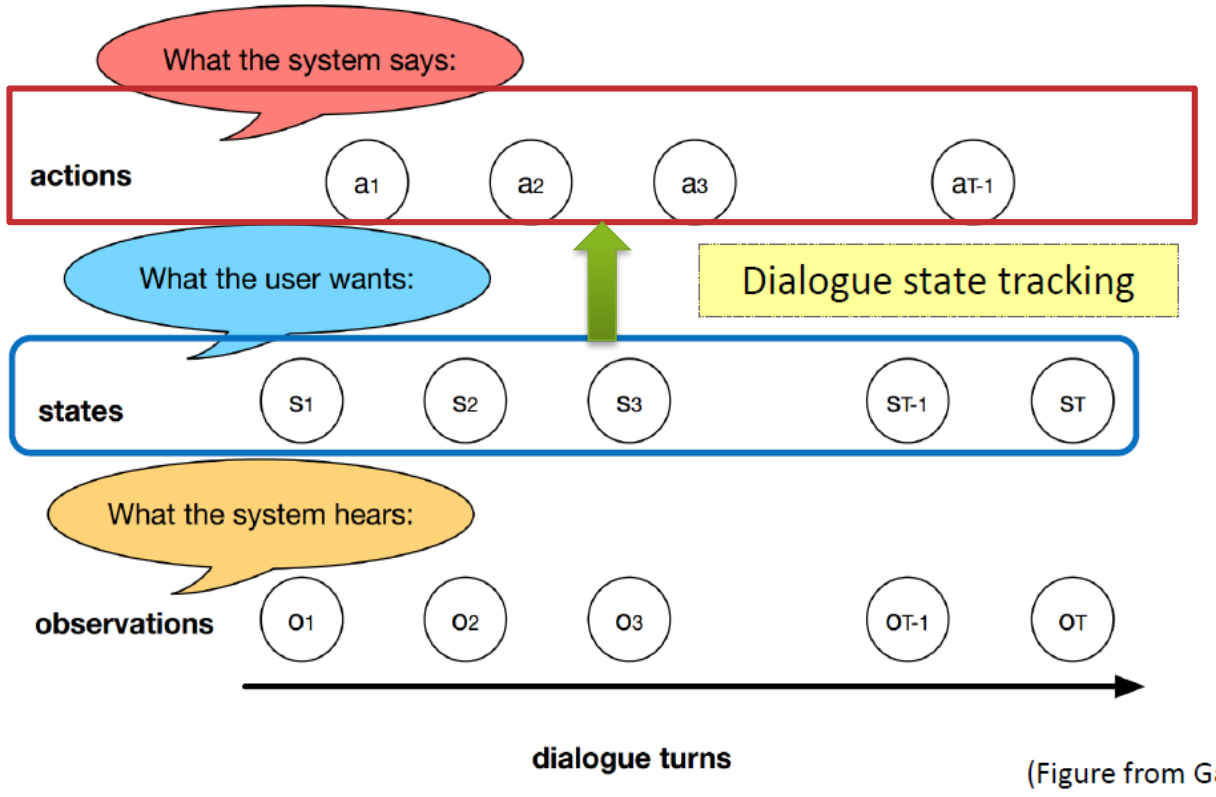
What the system should do next

© 2019 NUS. All rights reserved.

Dialogue Policy

- Research Topic
  - Rule based or Frame-based
  - Statistical Approaches
  - Reinforcement learning

$$\hat{A}_i = \underset{A_i \in A}{\operatorname{argmax}} P(A_i | (A_1, U_1, ..., A_{i-1}, U_{i-1})$$



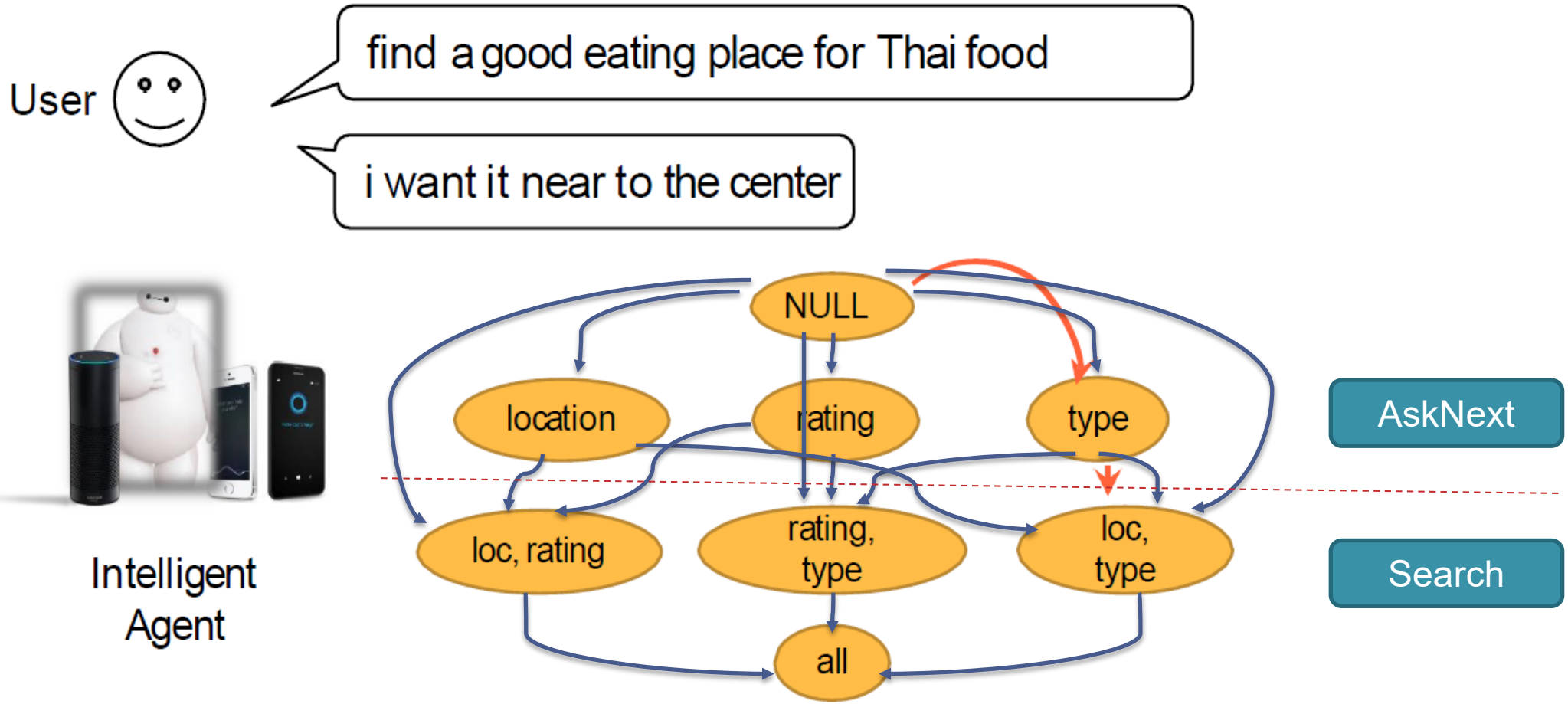
(Figure from Gašić)

https://sites.google.com/view/deepdial/

© 2019 NUS. All rights reserved.

Handcrafted Approaches

- Restaurant searching scenario



https://sites.google.com/view/deepdial/

© 2019 NUS. All rights reserved.

Statistical Approaches

- Learn to predict next **State/Action**
  - Data driven
  - Encode the State/Action with numbers

List of Actions

- AskType
- AskLoc
- AskRating
- Search

https://sites.google.com/view/deepdial/

© 2019 NUS. All rights reserved.

Statistical Approaches

- **Classification** based on Encoding

Observations (After NLU)

	"Welcome, ready to order?"	*"I want two Chicken Pizza"*	
• Type of order (delivery or pick up),	0	0	unknown
• Number of pizzas,	0	1	Above threshold
• Types of pizzas,	0	2	below threshold
• Sizes of pizzas,	0	0	
• Types of pizza dough, and	0	0	
• Drinks (optional field).	0	0	
• Acceptance,	0	0	
• Rejection, and	0	0	
• Not-understood.	0	0	

© 2019 NUS. All rights reserved.

Statistical Approaches

- **Classification** based on Encoding
  - predict the next system response after each user turn.

S_1 "Welcome…"

U_1 "I want two Chicken Pizza"

S_2 What is the Action for S_2 ?

Action = 1

0
1
2
0
0
0
0
0
0

Predict the next System action

1. Welcome (*Opening*).
2. Ask the type of order (*Ask_Type_Order*).
3. Ask the number of pizzas (*Ask_Number_Pizzas*).
4. Ask the types of pizzas (*Ask_Types_Pizzas*).
5. Ask the sizes of pizzas (*Ask_Sizes_Pizzas*).
6. Ask the types of dough (*Ask_Types_Doughs*).
7. Ask the drinks (*Ask_Drinks*).
8. Confirm the type of order (*Confirm_Type_Order*).
9. Confirm the number of pizzas (*Confirm _Number_Pizzas*).
10. Confirm the types of pizza (*Confirm _Types_Pizzas*).
11. Confirm the sizes of pizzas (*Confirm _Sizes_Pizzas*).
12. Confirm the types of dough (*Confirm _Types_Doughs*).
13. Confirm the drinks (*Confirm _Drinks*).
14. Closing (*Closing*).

© 2019 NUS. All rights reserved.

Statistical Approaches

- **Classification** based on Encoding

System1: Welcome to the Pizzeria. How can I help you?

```
A1: (Opening)
DR0: 000000-000
```

U1: I want two Sicilian pizzas

```
Number_Pizzas: Two        [0.91]
Types_Pizzas:  Sicilian   [0.23]
DR1: 012000-000

(...)
```

Active task-independent information	
Acceptance	0
Rejection	0
Non-understood	0

Dialog register (1 per user and turn)	
Active attributes	
Type_Order	0
Number_Pizzas	1
Types_Pizzas	2
Sizes_Pizzas	0
Types_Dough	0
Drink	0

© 2019 NUS. All rights reserved.

Statistical Approaches

- **Classification** based on Encoding

Previous Action item

```
#Dialog001
1,1,1,0,0,0,0,0,0,0,4
4,1,1,1,0,0,0,0,0,0,5
5,1,1,1,1,0,0,0,0,0,6
6,1,1,1,1,1,0,0,0,0,7
7,1,1,1,1,1,1,0,0,0,14
#Dialog002
1,2,1,0,0,0,0,0,0,0,8
8,1,1,0,0,0,0,0,0,0,4
4,1,1,1,0,0,0,0,0,0,5
5,1,1,1,1,0,0,0,0,0,6
6,1,1,1,1,1,0,0,0,0,7
7,1,1,1,1,1,1,0,0,0,14
```

Current Action item

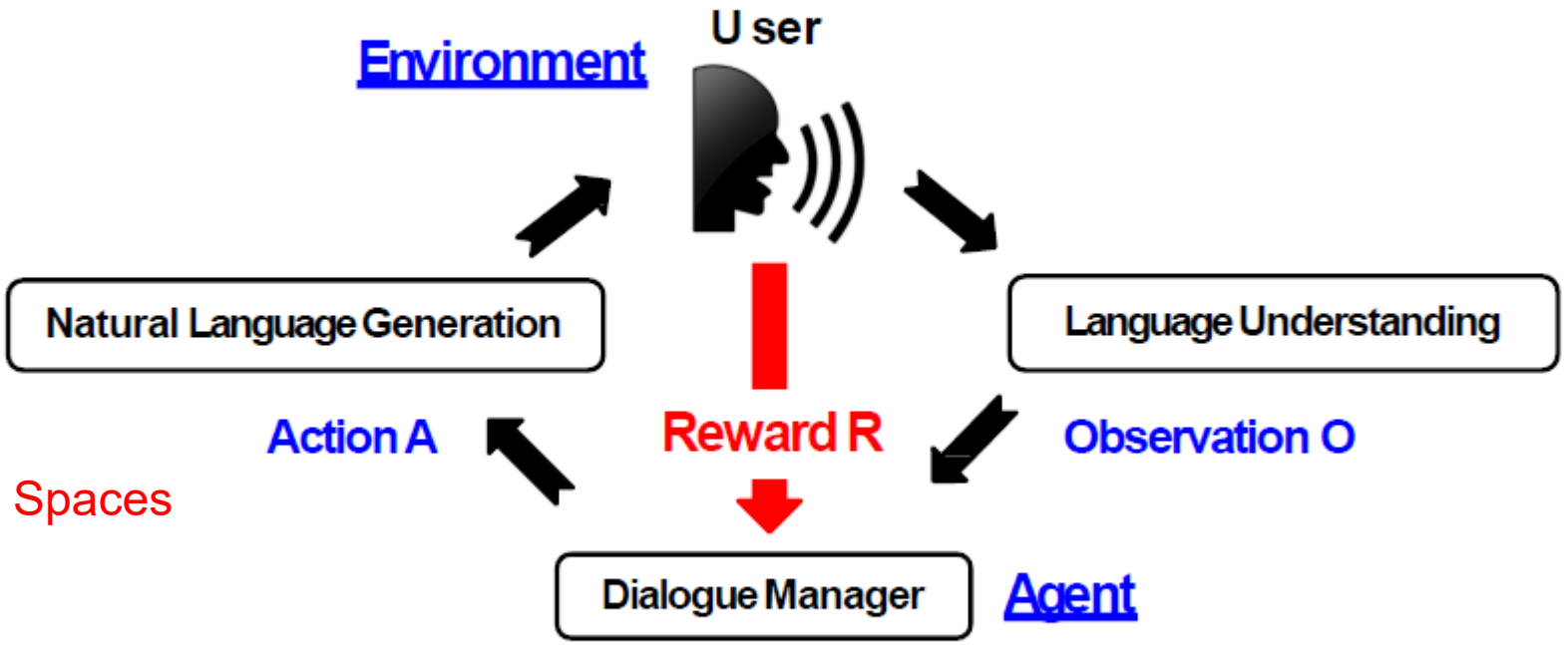
Current States

1. Welcome (*Opening*).
2. Ask the type of order (*Ask_Type_Order*).
3. Ask the number of pizzas (*Ask_Number_Pizzas*).
4. Ask the types of pizzas (*Ask_Types_Pizzas*).
5. Ask the sizes of pizzas (*Ask_Sizes_Pizzas*).
6. Ask the types of dough (*Ask_Types_Doughs*).
7. Ask the drinks (*Ask_Drinks*).
8. Confirm the type of order (*Confirm_Type_Order*).
9. Confirm the number of pizzas (*Confirm _Number_Pizzas*).
10. Confirm the types of pizza (*Confirm _Types_Pizzas*).
11. Confirm the sizes of pizzas (*Confirm _Sizes_Pizzas*).
12. Confirm the types of dough (*Confirm _Types_Doughs*).
13. Confirm the drinks (*Confirm _Drinks*).
14. Closing (*Closing*).

© 2019 NUS. All rights reserved.

Reinforcement learning

- Dialogue management in a RL framework



Limited Action Spaces

The optimized dialogue policy selects the best action that **maximizes the future reward**

© 2019 NUS. All rights reserved.

Reinforcement learning

- 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 system training before deployment

© 2019 NUS. All rights reserved.

Response Generation

What to say to users

© 2019 NUS. All rights reserved.

Response Generation

Natural Language Generation (NLG)

- Mapping dialogue acts into natural language

inform(name=Seven_Days, foodtype=Chinese)

Seven Days is a nice Chinese restaurant

© 2019 NUS. All rights reserved.

Response Generation

Template-Based NLG

- Define a set of rules to map frames to natural language

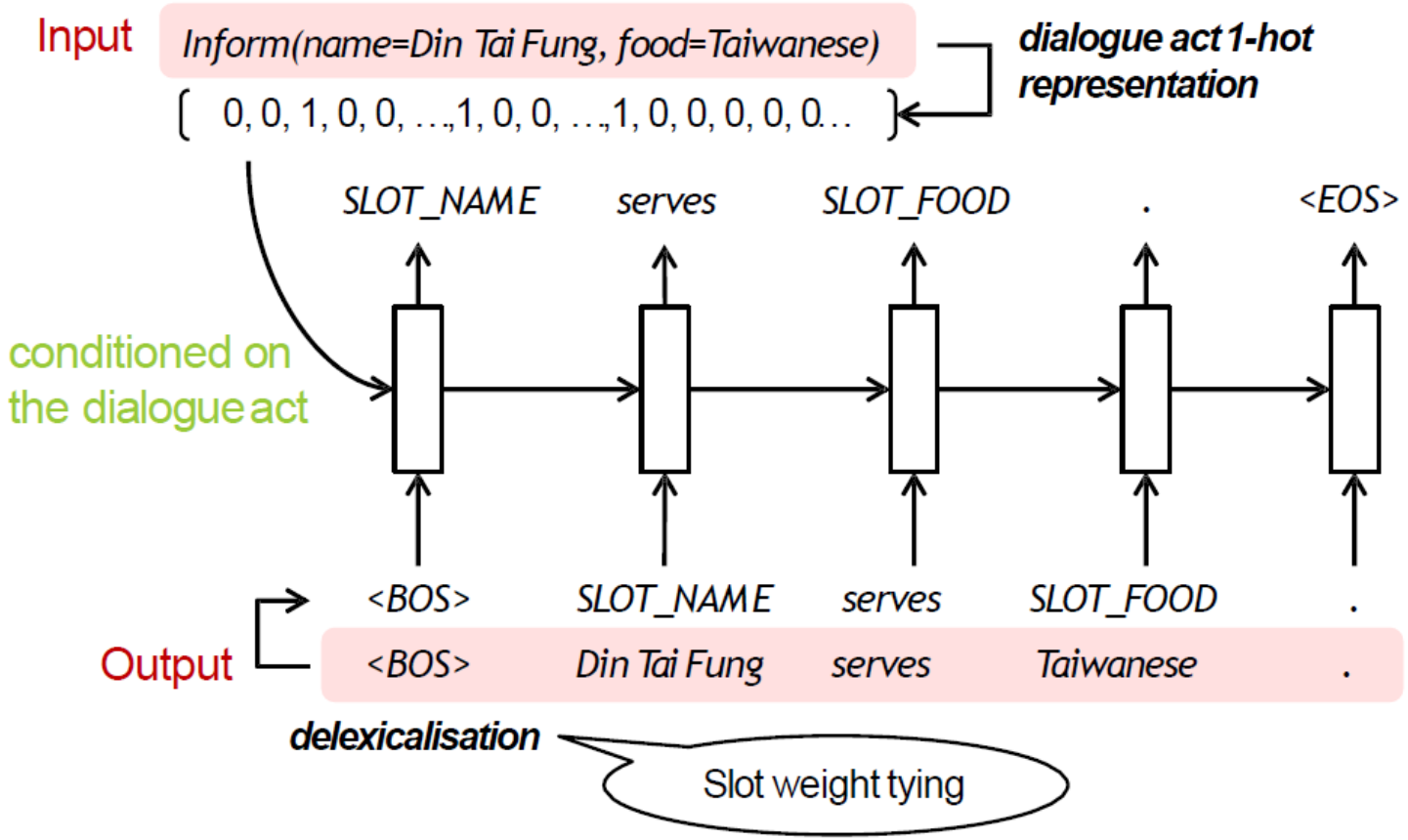
Semantic Frame	Natural Language
confirm()	"Please tell me more about the product your are looking for."
confirm(area=$V)	"Do you want somewhere in the $V?"
confirm(food=$V)	"Do you want a $V restaurant?"
confirm(food=$V,area=$W)	"Do you want a $V restaurant in the $W."

Pros: simple, error-free, easy to control
Cons: time-consuming, rigid, poor scalability

© 2019 NUS. All rights reserved.

Response Generation

RNN-Based LM NLG [wen+ 15]



© 2019 NUS. All rights reserved.

References

- Multi-domain neural network language generation for spoken dialogue systems. In Proceedings of the 2016 Conference of the North American Chapter of the Association for
- Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP),
- A network-based end-to-end trainable task-oriented dialogue system. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics
- Multi-domain dialog state tracking using recurrent neural networks. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics
- Neural belief tracker: Data-driven dialogue state tracking. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL),
- http://deeppavlov.ai/
- https://sites.google.com/view/deepdial/

Acknowledgement: Some of the slides are adapted from the dialogue tutorial given by Jianfeng Gao and Vivian Chen.

© 2019 NUS. All rights reserved.