Conversational Recommendation: Formulation, Methods, and Evaluation

Wenqiang Lei, Xiangnan He, Maarten de Rijke, Tat-Seng Chua

wenqianglei@gmail.com, hexn@ustc.edu.cn, derijke@uva.nl, dcscts@nus.edu.sg

slides will be available at: https://core-tutorial.github.io

A literature survey based on this tutorial as well as other materials will be available soon.

• Information Seeking

information overload

Information explosion problem?

- Information seeking requirements
  - ➢ E-commerce(Amazon and Alibaba)
  - ➢ Social networking(Facebook and Wechat)
  - ➢ Content sharing platforms(Instagram and Pinterest)

Two major types of information seeking techniques

Search

RECOMMENDED

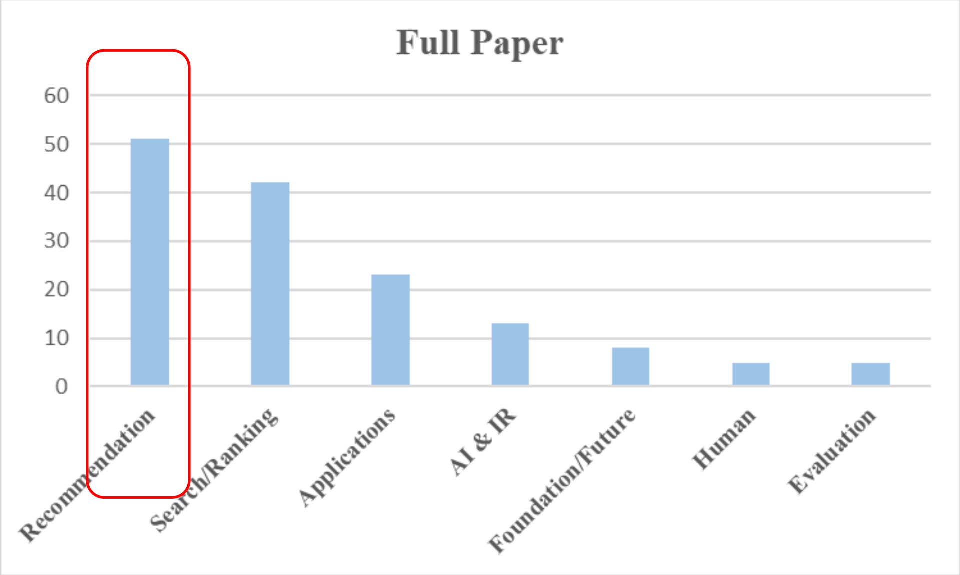
Recommendation

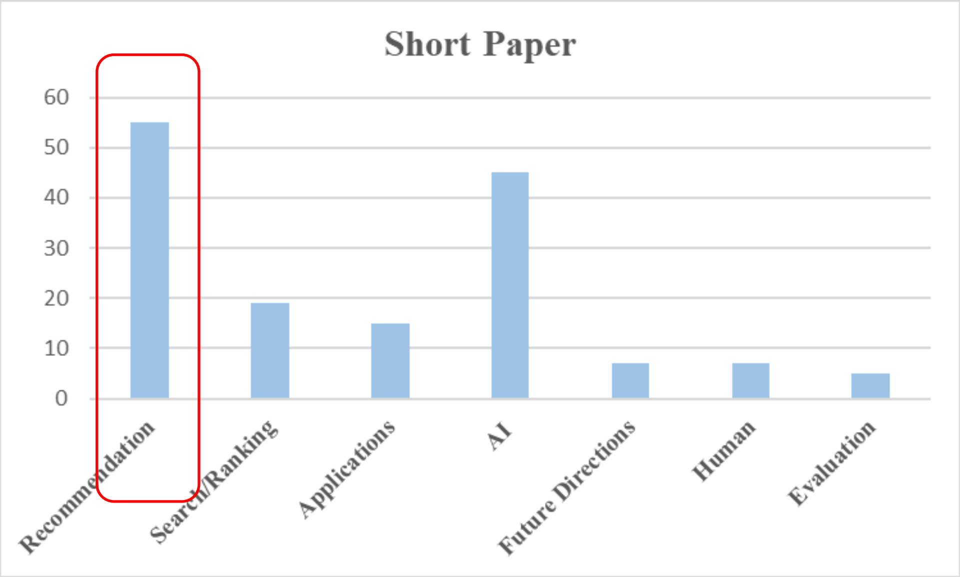
How to handle?

• Recommendation Has Become Prevalent in IR Community

Recommendation becomes the most popular track

2019 SIGIR Hot Topics





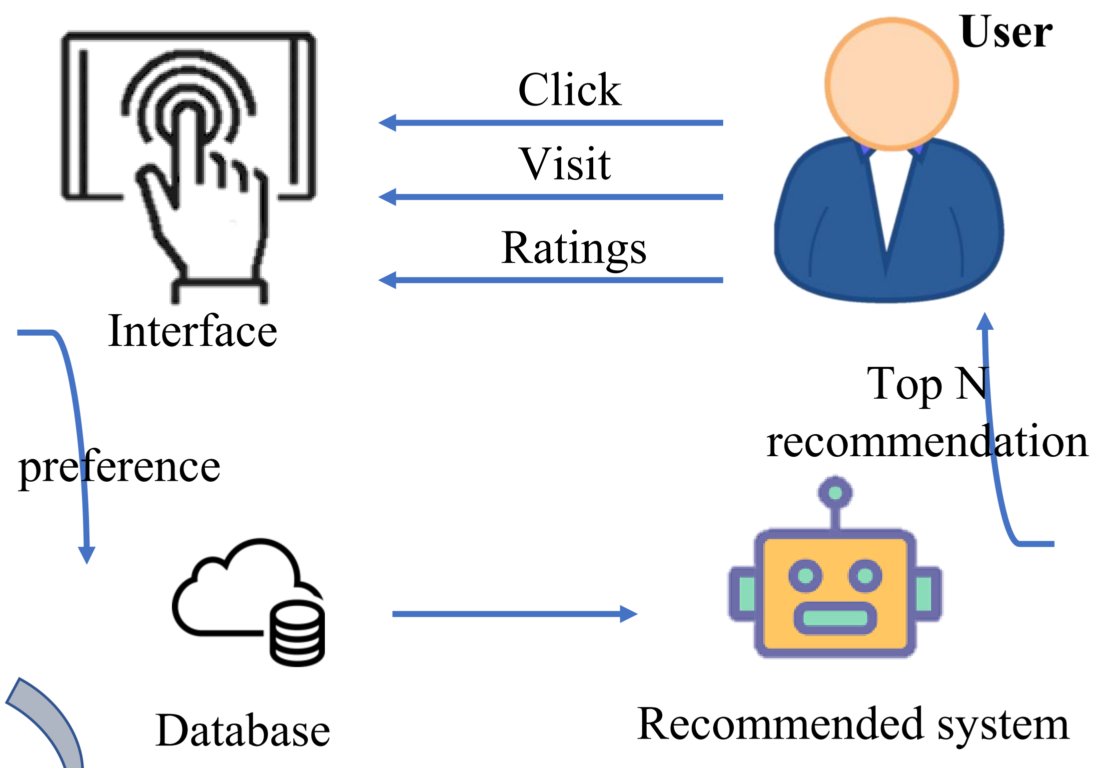
SIGIR of different Topics were received in 2020

• Typical Recommender Systems

Recommender systems

- predict a user's preference towards an item by analyzing their past behavior

（e.g., click history, visit log, ratings on items, etc）



Implicit

• Existing Static Recommendation： Collaborative Filtering

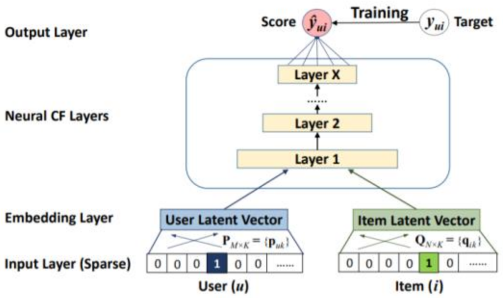
➢ Collaborative filtering

- matrix factorization and factorization machines

Factorization Machines

➢ Deep learning approaches

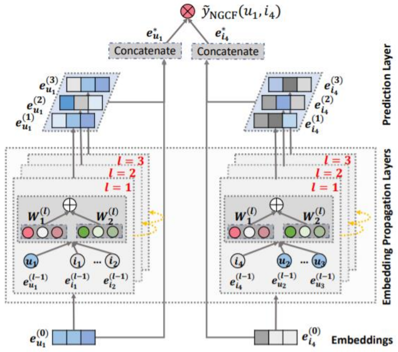
- neural factorization machines & deep interest networks



Neural Collaborative Filtering

➢ Graph-based approaches

- expressiveness and explainability of graphs

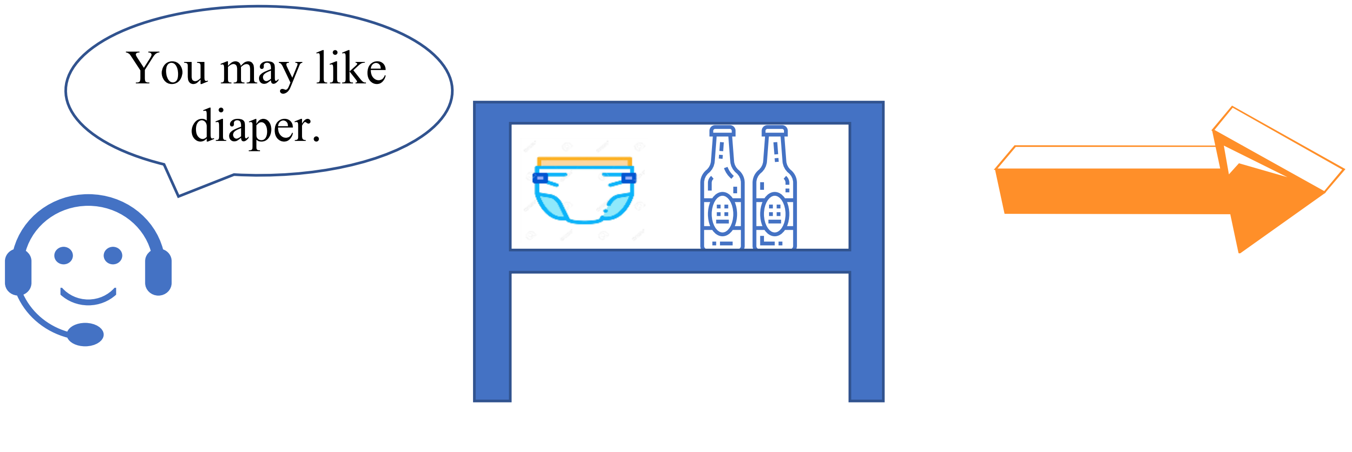


Neural Graph Collaborative Filtering

• Limitation: Information Asymmetry

Key Problems for Recommendation: Information Asymmetry

- Information asymmetry
  - A system can only **estimate** users’ preferences based on their historical data
- Intrinsic limitation
  - Users’ preferences often **drift** over times.
  - It is hard to find **accurate reasons** to recommendation



• Existing Online Recommendation: Bandit

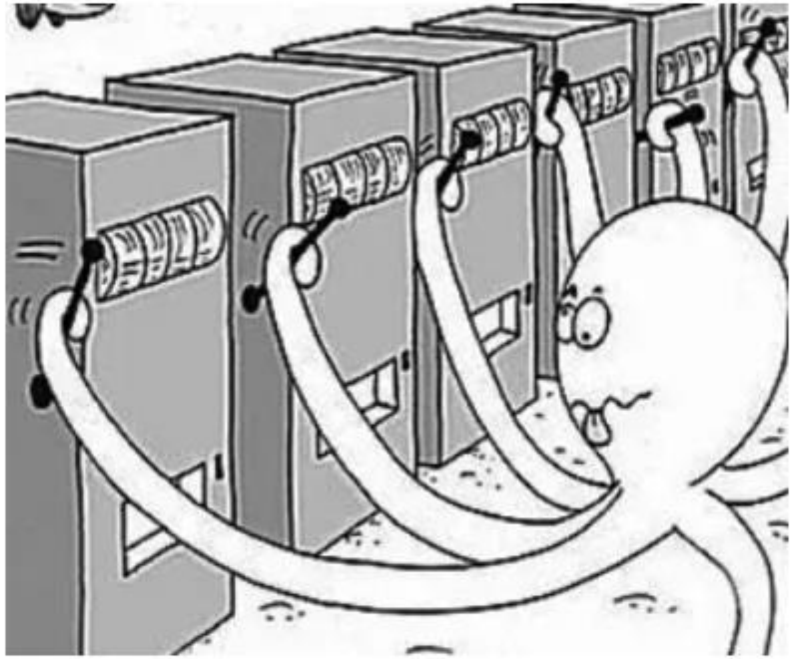
Exploration and
Exploitation Balance

Online Recommendation:

Arm	→	Item/ Item Category
Reward	→	User feedback
Environment	→	User

❑ **Bandit Algorithm**:

- Exploit-Explore problem
- Cold-Start problem



Multi-Armed Bandit

• Limitation：Lack of Explainability

A model still has no channel to know find the exact reason why a user prefer an item.

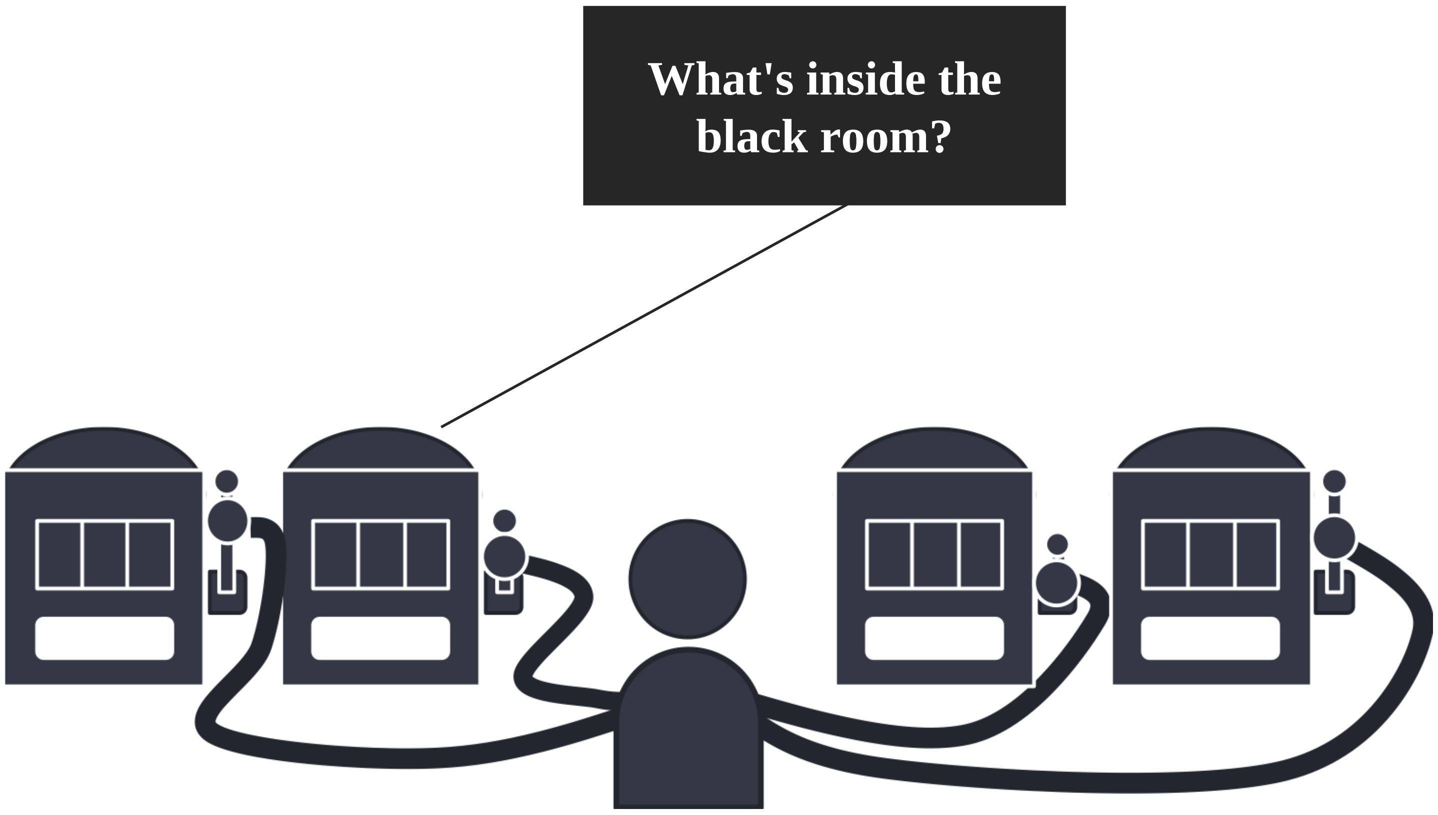
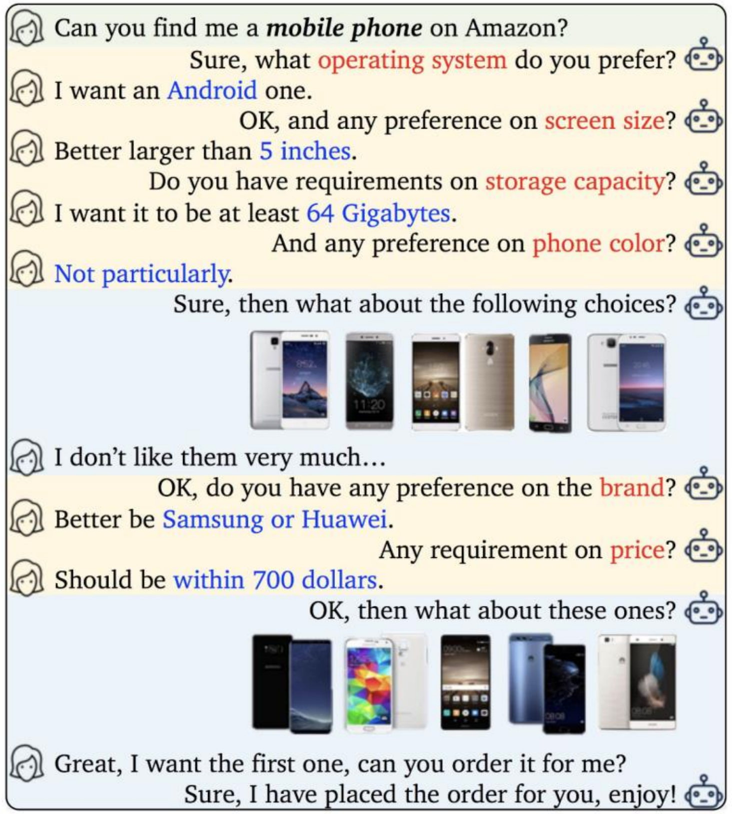
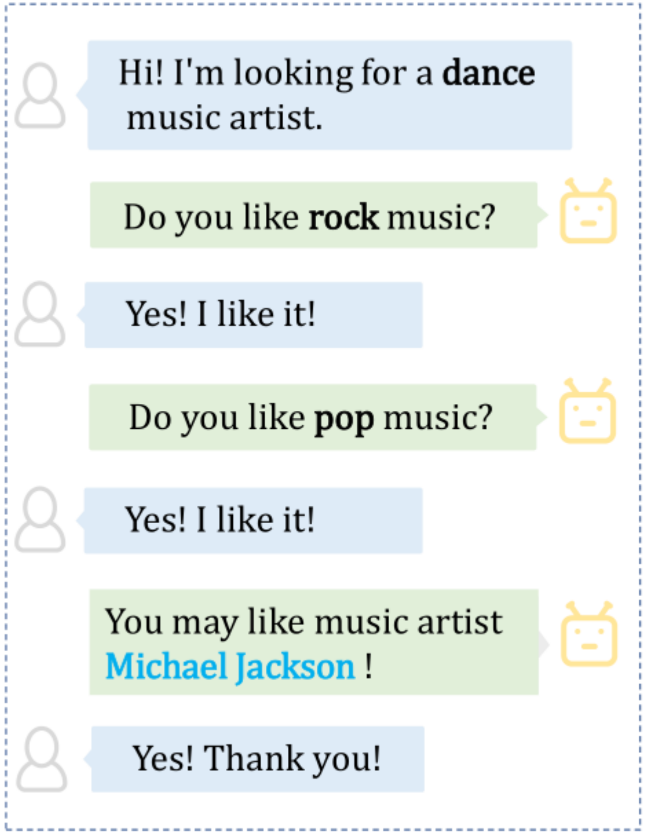


Figure credit: *Spotx*

• Conversation Brings Revolution





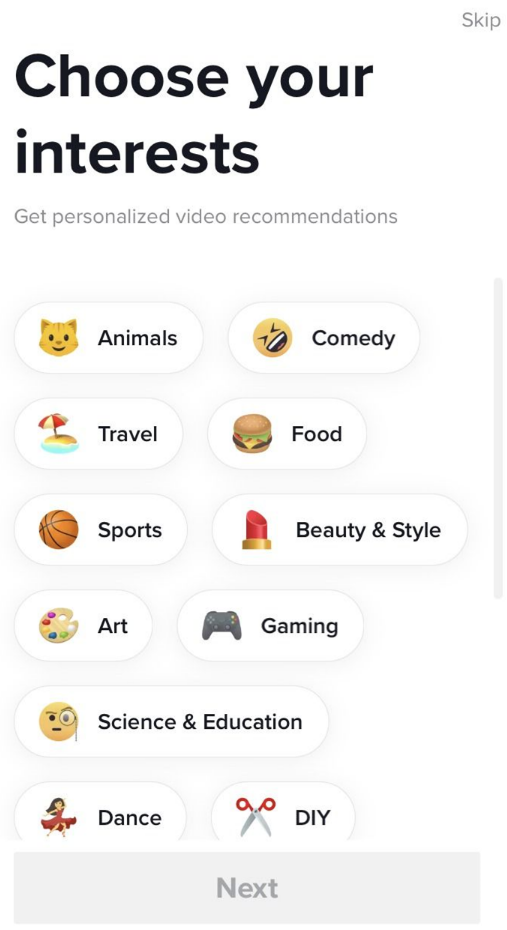
The example of a conversational recommender system

Conversational Recommender Systems

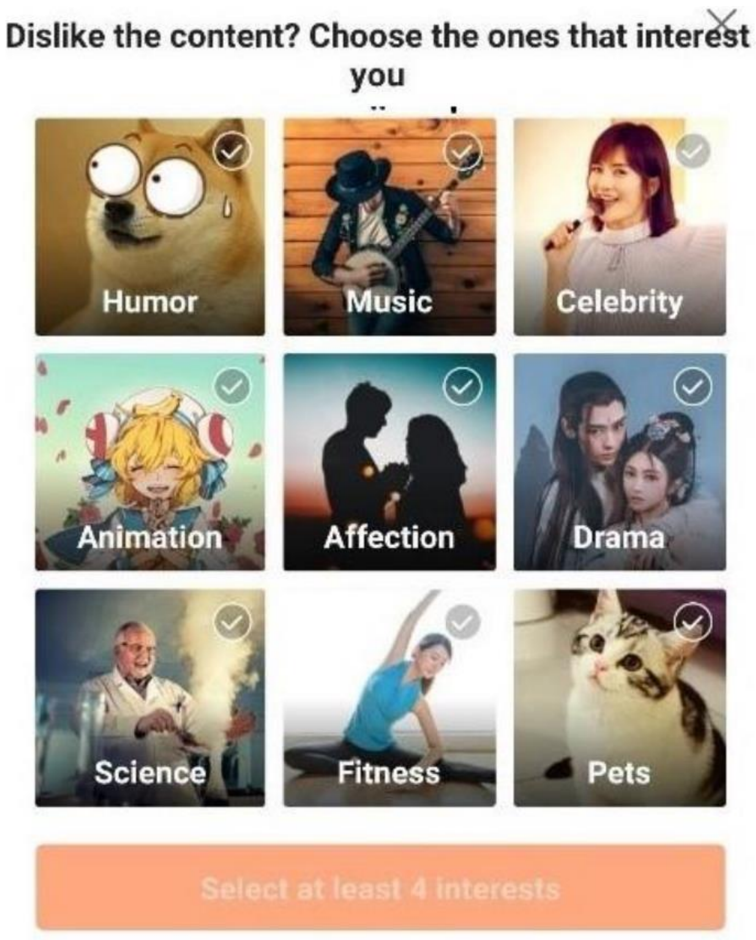
- ➢Interactive recommendation
- ➢Using natural languages

• Conversational Recommender Systems In a Broader Perspective

• Tag-based Interaction



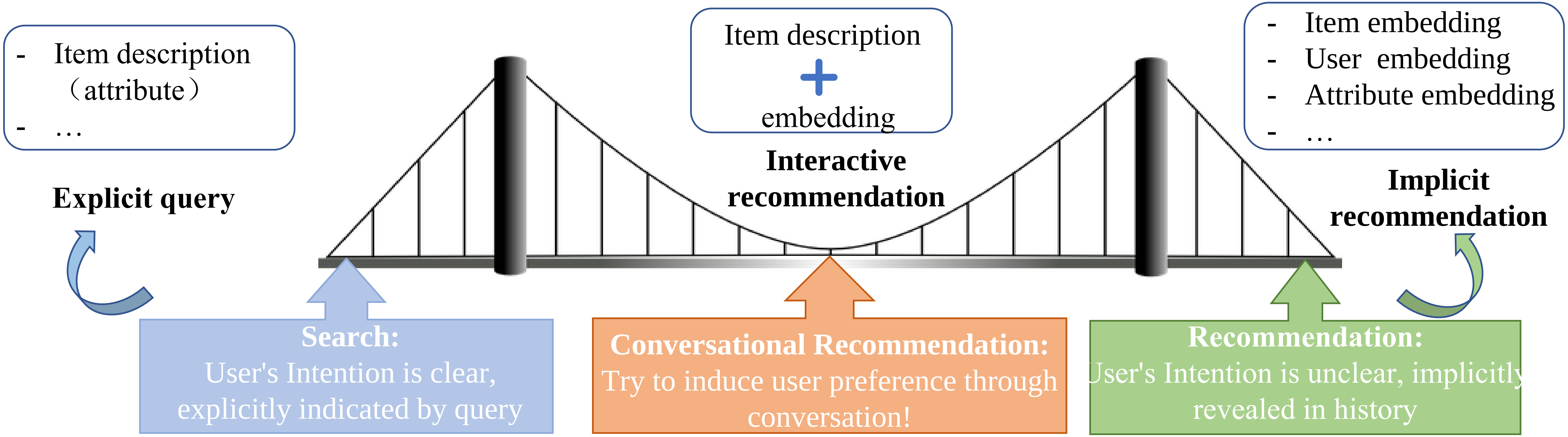
The example of tag-based interaction on tiktok



The example of tag-based interaction on kuaishou

• Conversational Recommendation Bridges Search and Recommendation

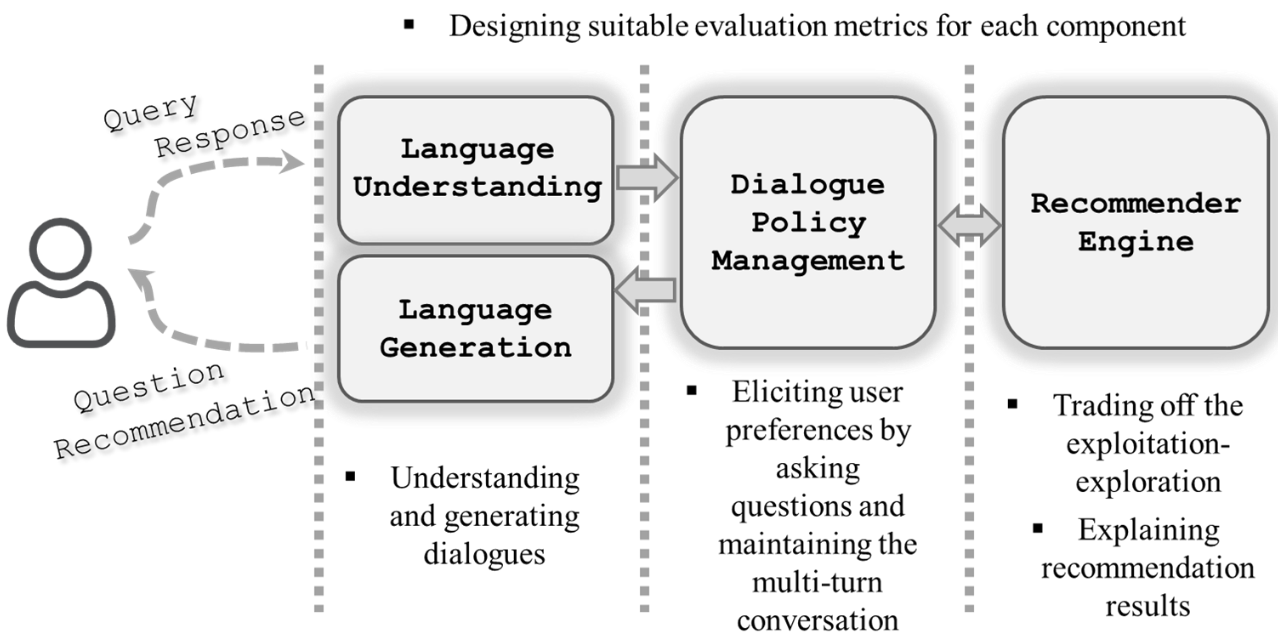
Traditional paradigms for information-seeking:
Search (pull) **or** **Recommendation (push)**



• Conversational Recommender Systems

Four Directions being Explored

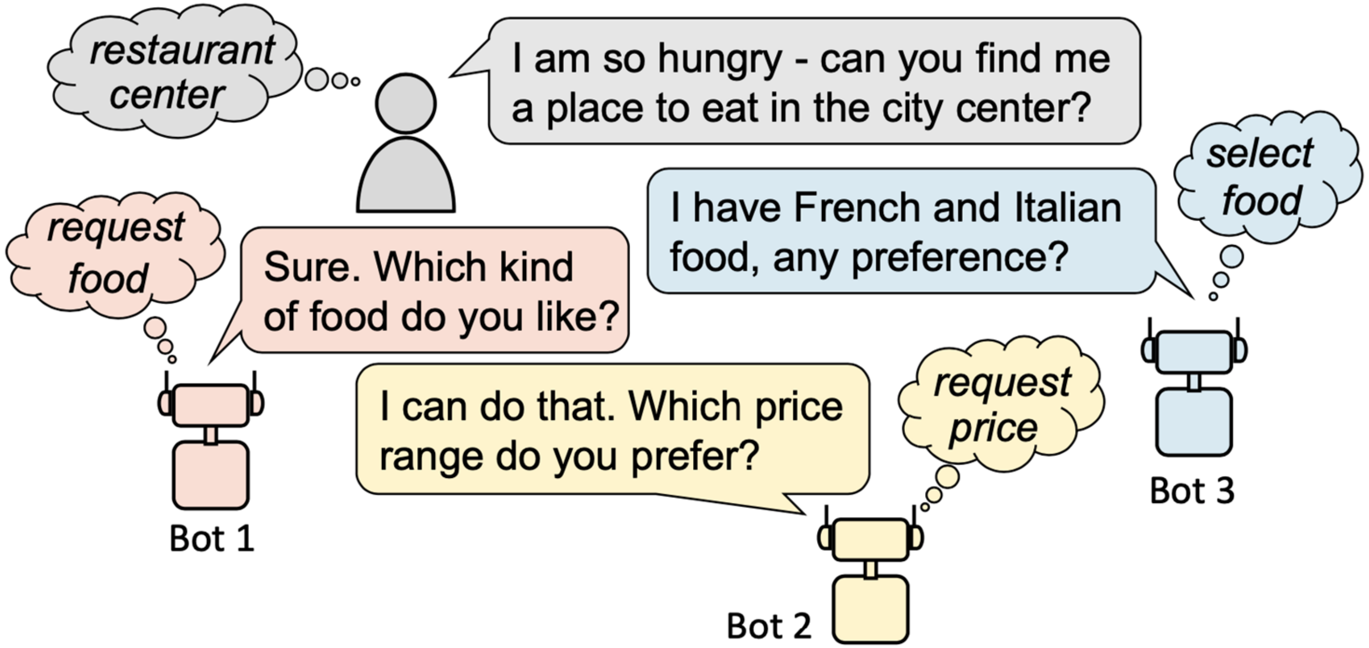
1. Question Driven Approaches
2. Multi-turn Conversational Recommendation Strategy
3. Exploitation-Exploration Trade-offs for Cold Users
4. Dialogue Understanding and Generation



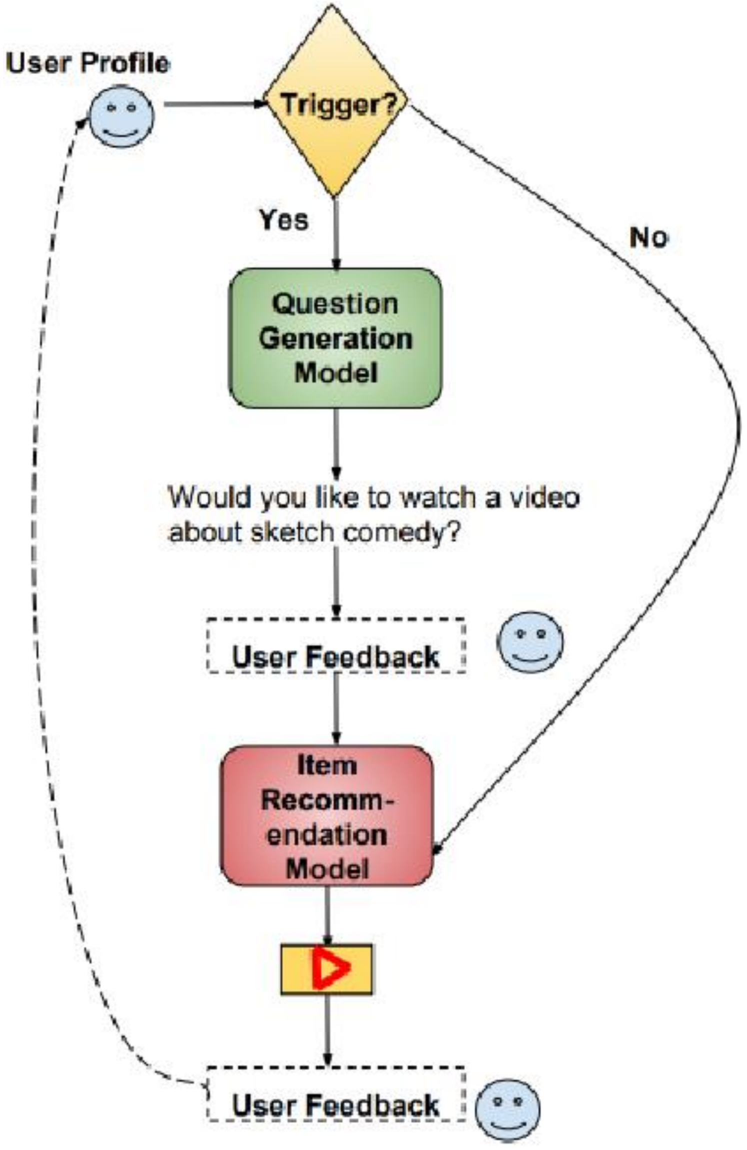
• Question Driven Approaches in CRS

The key advantage of conversational recommendation: being able to ask questions.

- **Ask about attributes/topics/categories of items to narrow down the recommended candidates.**



Zhang et al. Task-Oriented Dialog Systems that Consider Multiple Appropriate Responses under the Same Context (AAAI' 20)

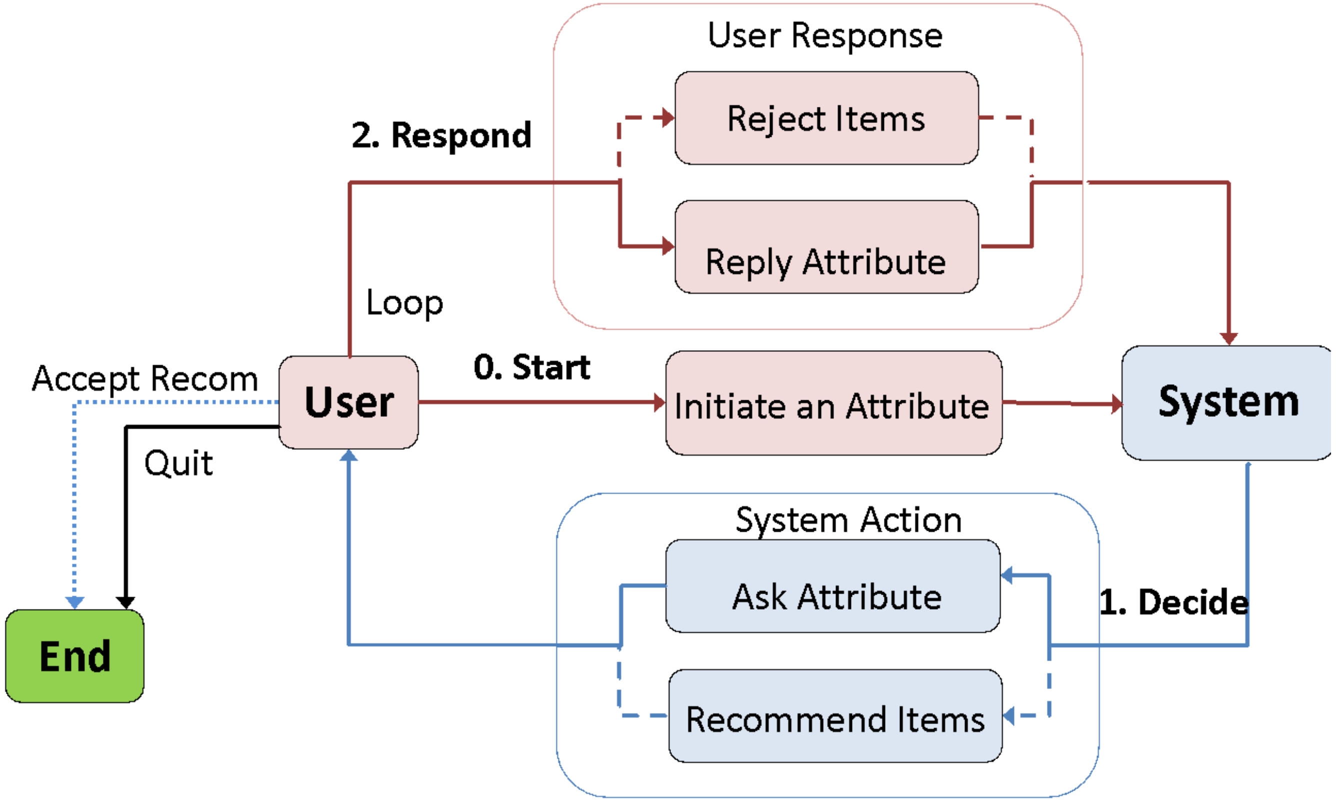


Christakopoulou et al. "Q&R: A Two-Stage Approach toward Interactive Recommendation"(KDD' 18)

•Multi-turn Conversational Recommendation Strategy

A System needs to choose to ask questions and make recommendations in a multi-turn conversation

- **Purpose**: making successful recommendations with less turns of interactions
- **Challenges to address:**
  1. Which items or attributes to recommend?
  2. When to ask questions and when to make recommendations?
  3. How to adapt user feedback

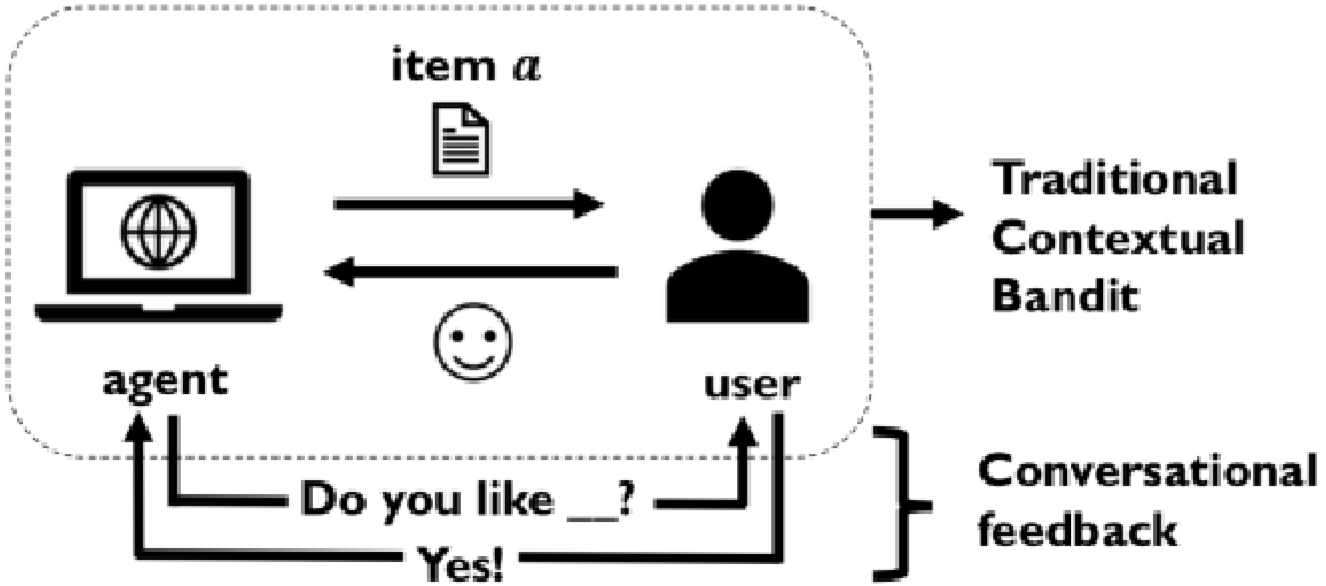


Lei et al. "Estimation–Action–Reflection: Towards Deep Interaction Between Conversational and Recommender Systems" (WSDM'20)

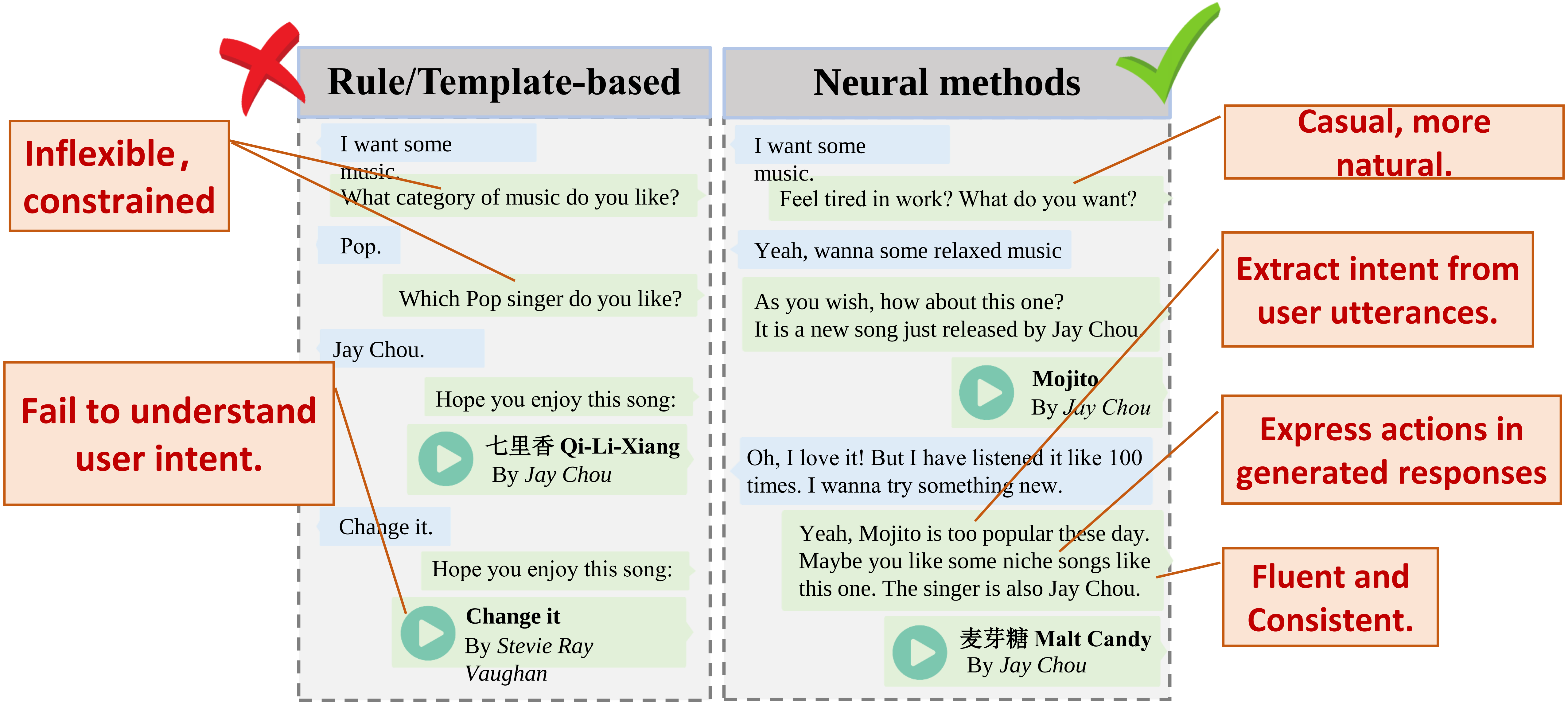
• Exploitation-Exploration Trade-offs for Cold Users

Exploration (Learning)	Trade-off	Exploitation (Earning)
Take some risk to collect information about unknown options		Takes advantage of the best option that is known.

✔ **Leverage the dynamics of CRS to benefit the E&E trade-off for cold users/items.**



• Dialogue Understanding and Generation

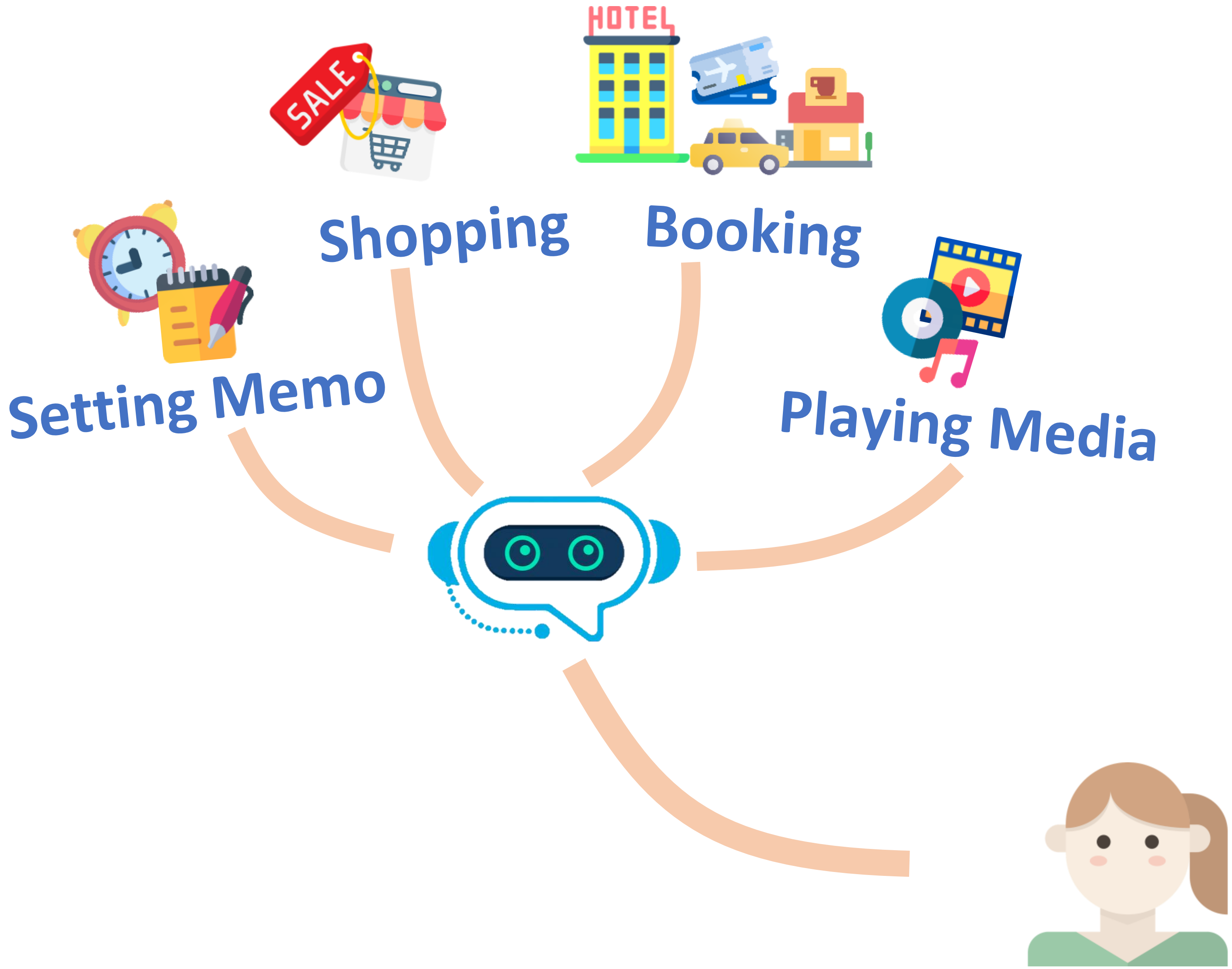


• Tutorial Outline

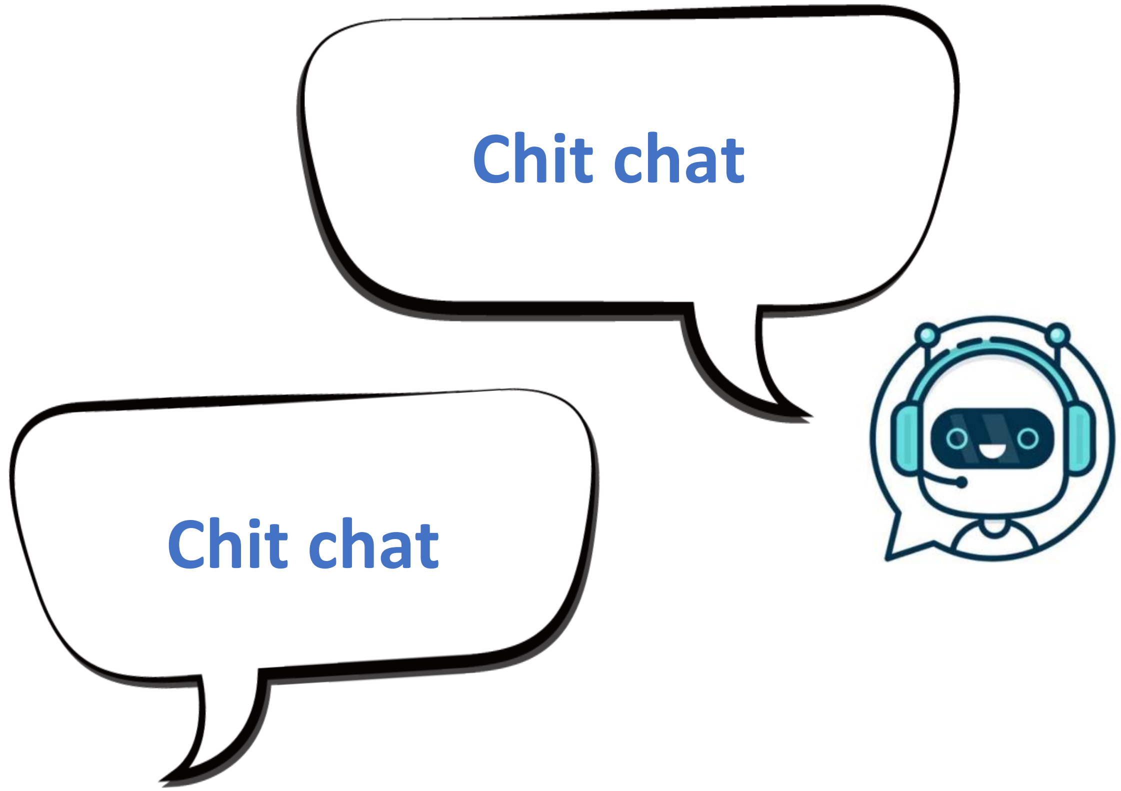
- ❑ **A Glimpse of Dialogue System**
- ❑ Four research directions in conversational recommendation system
  - ❑ Question Driven Approaches
  - ❑ Multi-turn Conversational Recommendation Strategy
  - ❑ Dialogue Understanding and Generation
  - ❑ Exploitation-Exploration Trade-offs for Cold Users
- ❑ Summary of Formalizations and Evaluations

• Two Types of Dialogue Systems

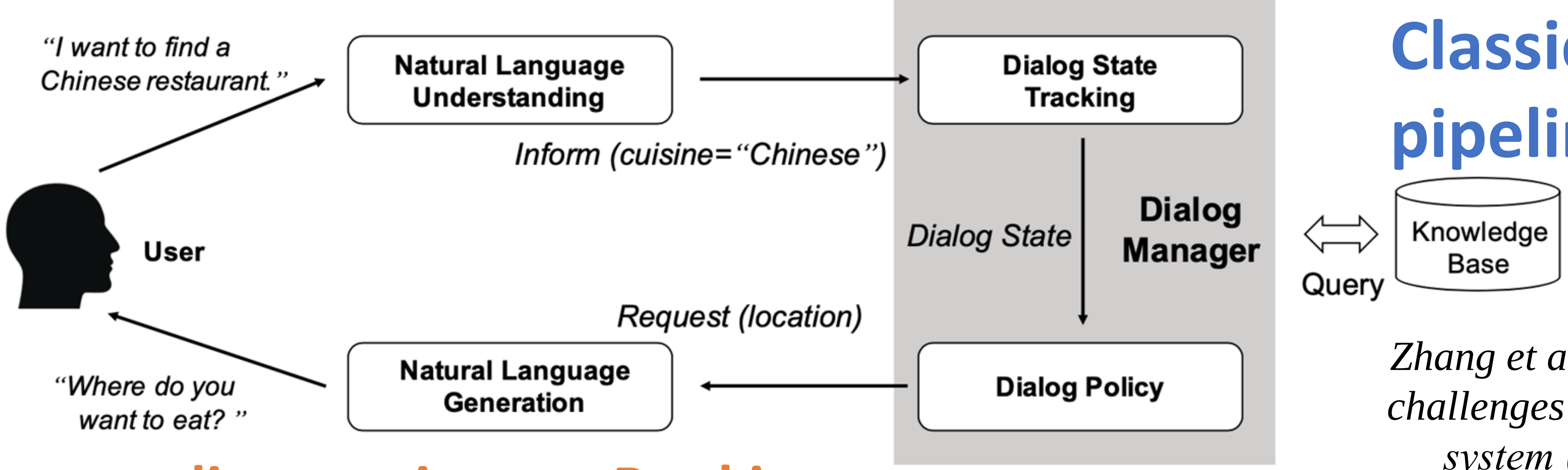
- **Task-oriented Dialogue System**



- **Non-task-oriented Dialogue System (Chatbot)**



• Typical Structure of Task-oriented Dialogue System

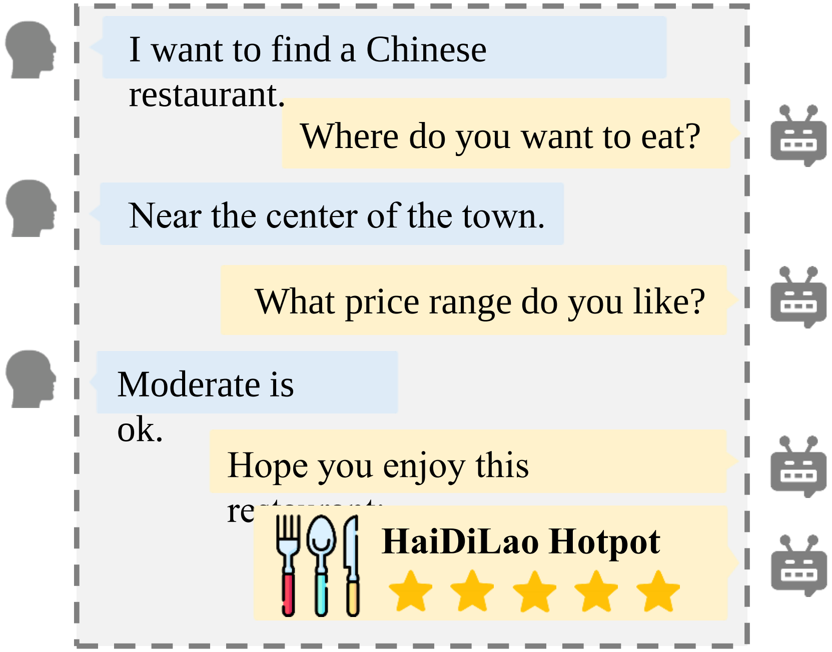


Classical pipeline structure

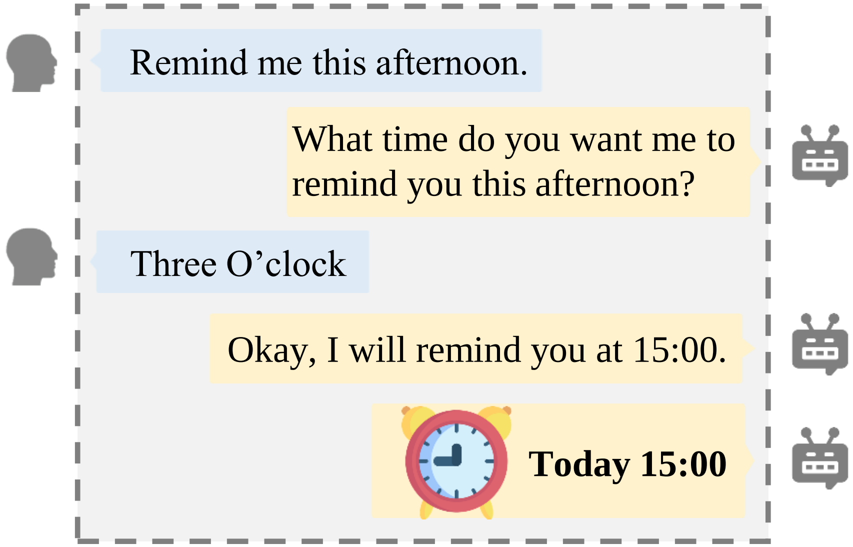
Zhang et al. Recent advances and challenges in task-oriented dialog system (Science China' 20)

Recommending music

Booking restaurants



Setting alarms



• Natural Language Understanding

• **Three Purpose:**

1. **Domain detection**
2. **Intent detection**
3. **Slot value extraction**

An example utterance with annotations in IOB format

W	find	recent	comedies	by	james	cameron
	↓	↓	↓	↓	↓	↓
S	O	B-date	B-genre	O	B-dir	I-dir
D	movies					
I	find_movie					

where:
S: semantic slots.
D: domain.
I: intent.

In IOB format:
O: a token belongs to no chunk.
B-: the beginning of every chunk.
I-: a token inside a chunk

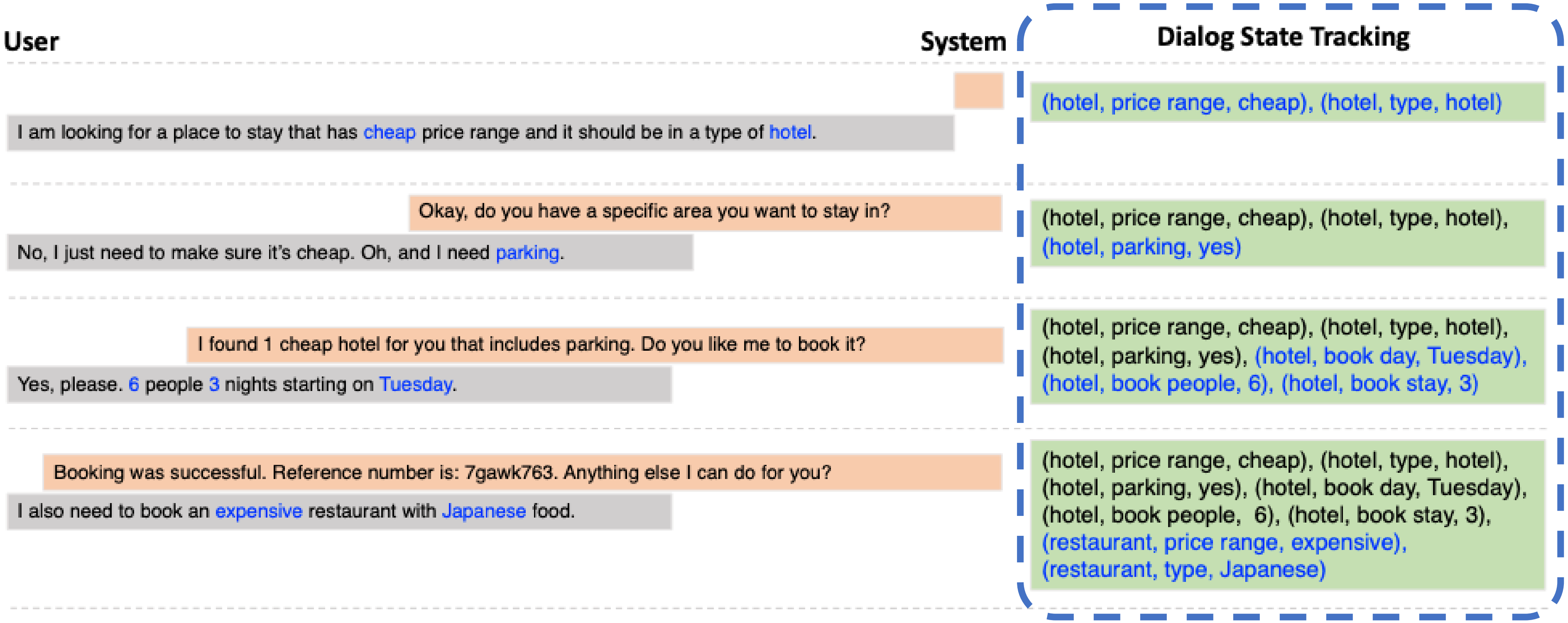
Hakkani-T ̈ur et al. Is Your Goal-Oriented Dialog Model Performing Really Well? Empirical Analysis of System-wise Evaluation (INTER- SPEECH’ 20)

• Dialogue State Tracking

Aiming to track all the states accumulated across the conversational turns

Recent solutions: latent vector-based methods

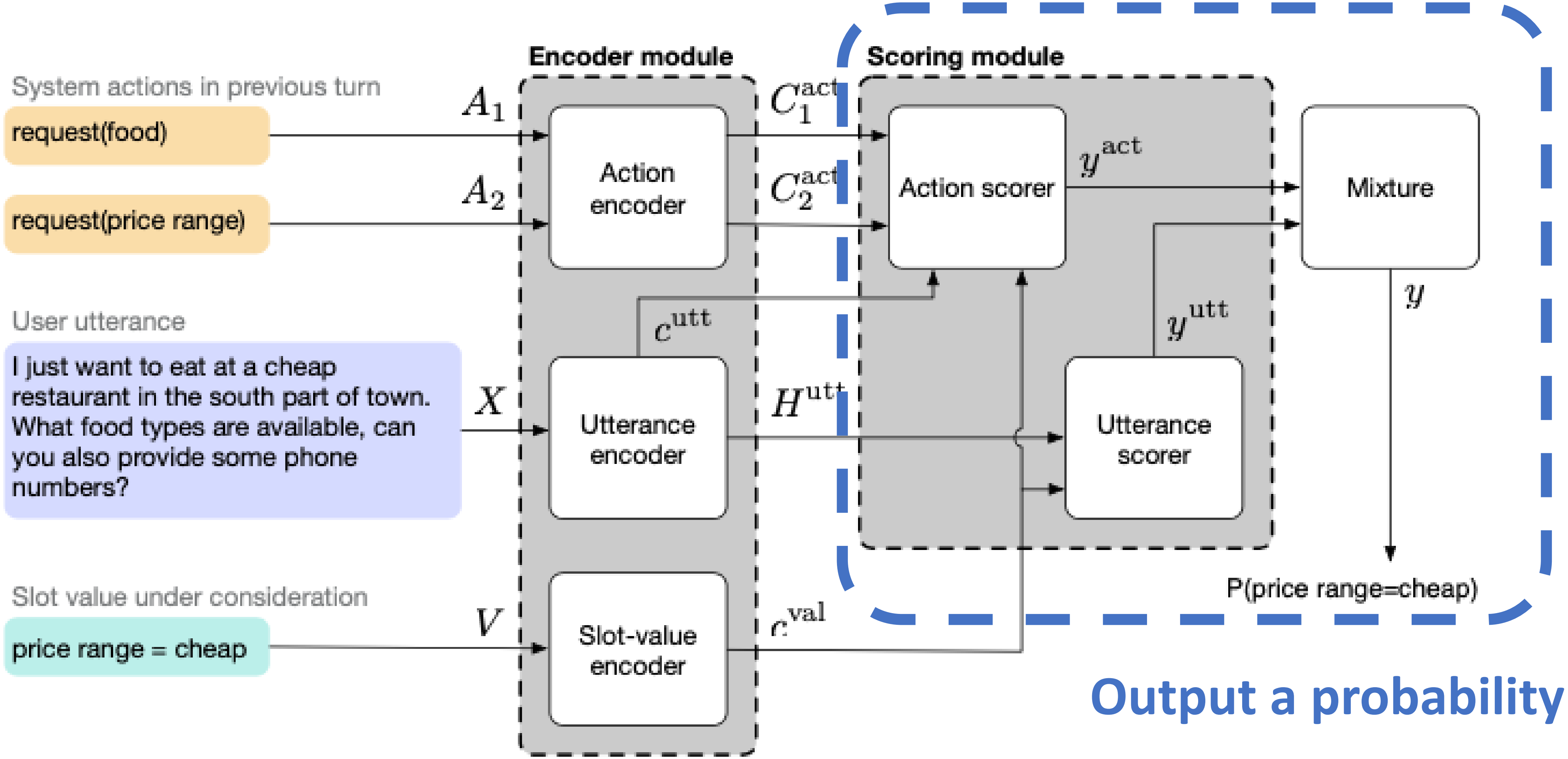
1. Classification (picklist-based).
2. Copying (generative)



Zhang et al. Find or Classify? Dual Strategy for Slot-Value Predictions on Multi-Domain Dialog State Tracking (Arxiv' 19)

• Jointly Solving Natural Language Understanding and Dialogue State Tracking -- Classification

• Using a classifier as dialogue state tracker

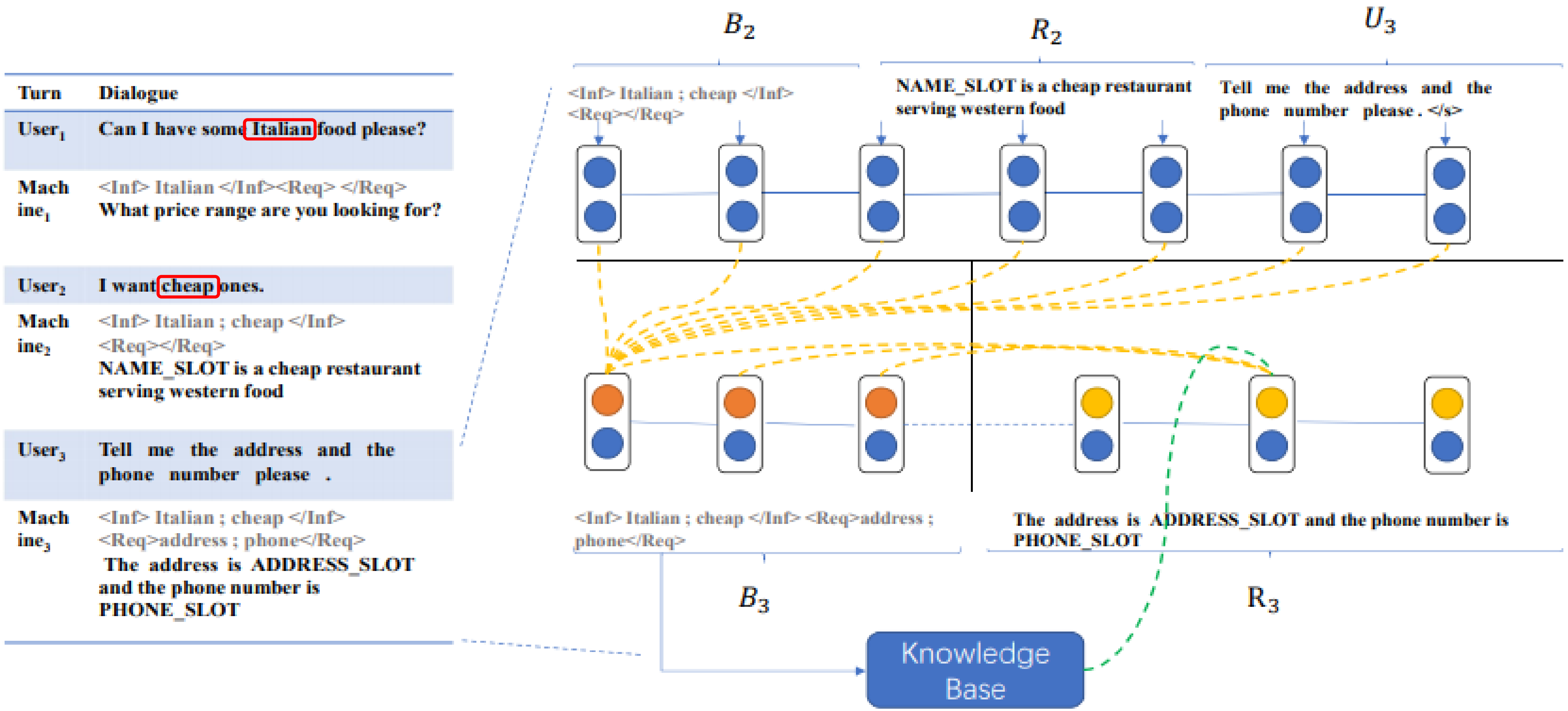


Output a probability of state

Zhong et al. Global-Locally Self-Attentive Encoder for Dialogue State Tracking (ACL' 18)

• Jointly Solving Natural Language Understanding and Dialogue State Tracking -- Copying

- **Find the text span in original utterances.**

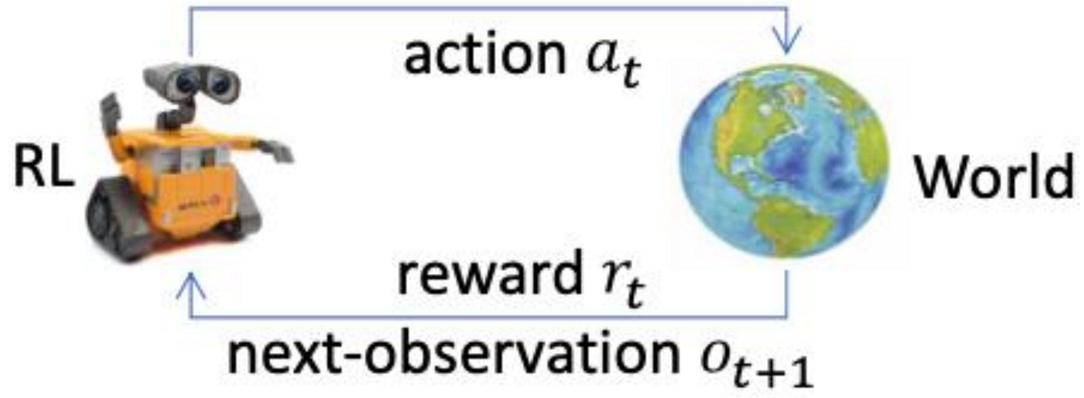


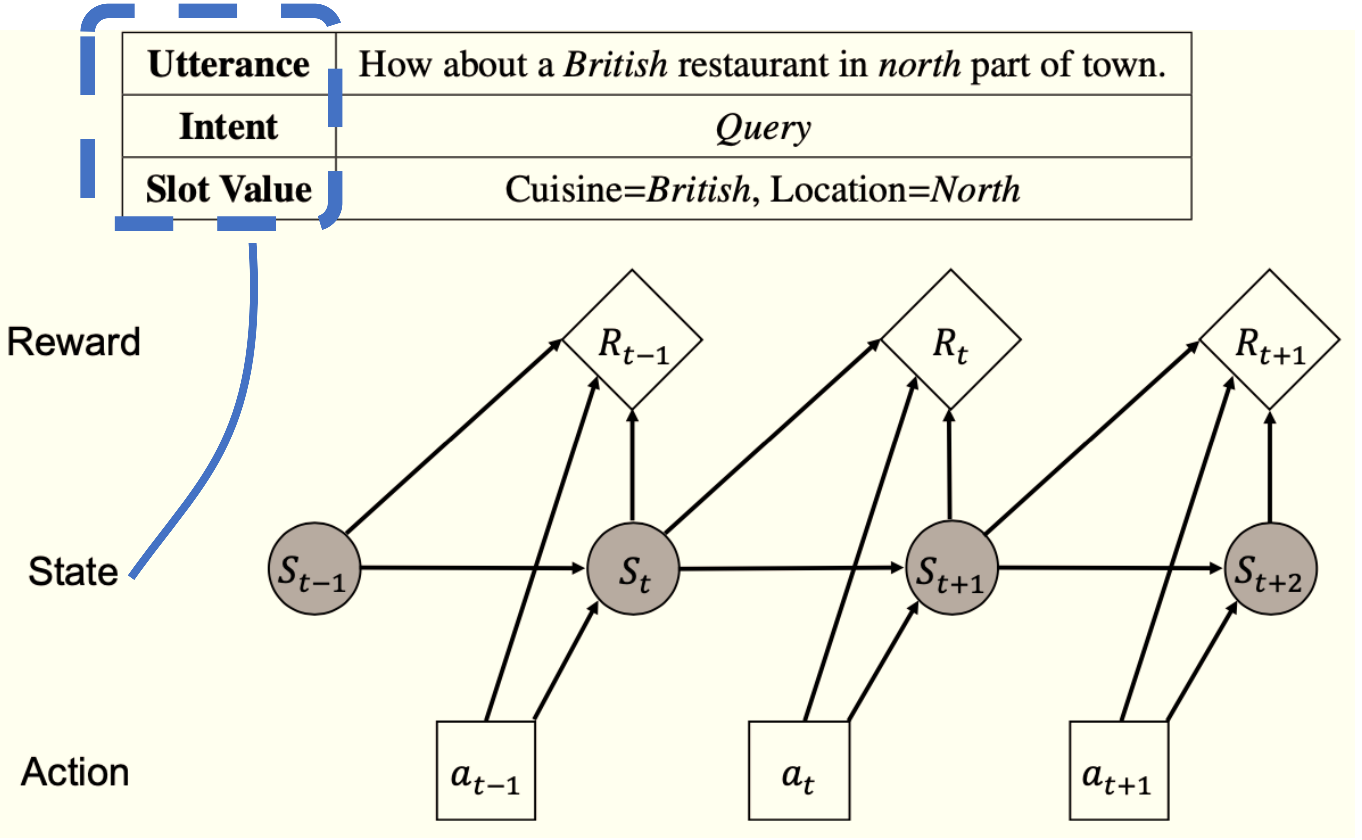
Lei et al. Sequicity: Simplifying Task-oriented Dialogue Systems with Single Sequence-to-sequence Architectures (ACL' 18)

• Dialogue Policy

- Dialogue act in a session are generated sequentially, so it is formulated as a **Markov Decision Process (MDP)**

- Can be address by **Supervised Learning** or **Reinforcement Learning**





A framework of MDP.

Zhang et al. Recent advances and challenges in task-oriented dialog system (Sci China Tech Sci' 20)

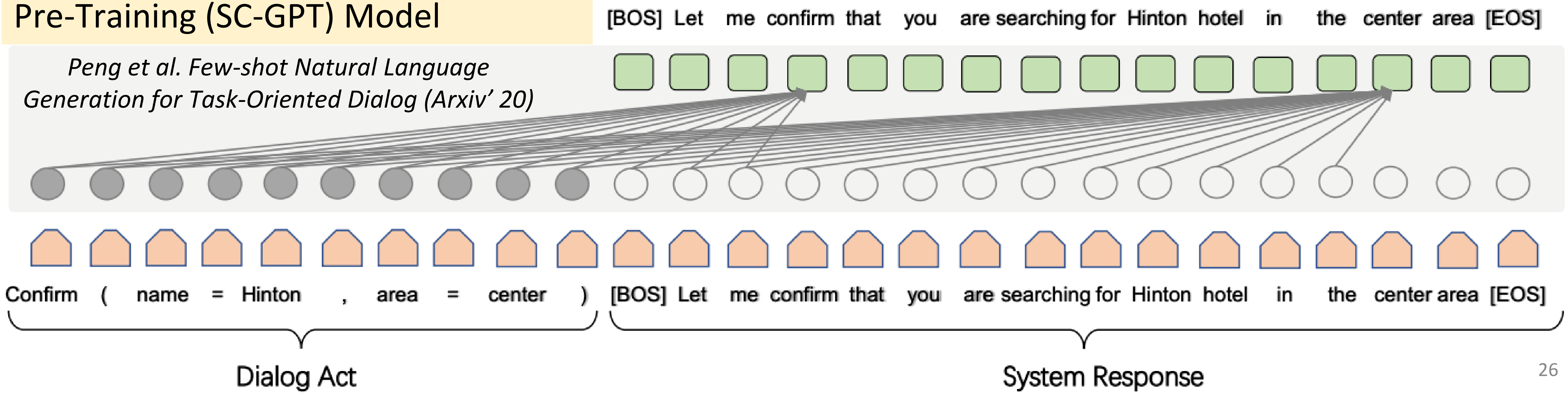
• Natural Language Generation

- **Strategies:**
  - Surface realization
  - Conditioned language generation (RNN-based neural network)

- **Challenges:**
  - **Adequacy**: meaning equivalence,
  - **Fluency**: syntactic correctness,
  - **Readability**: efficacy in context,
  - **Variation**: different expression.

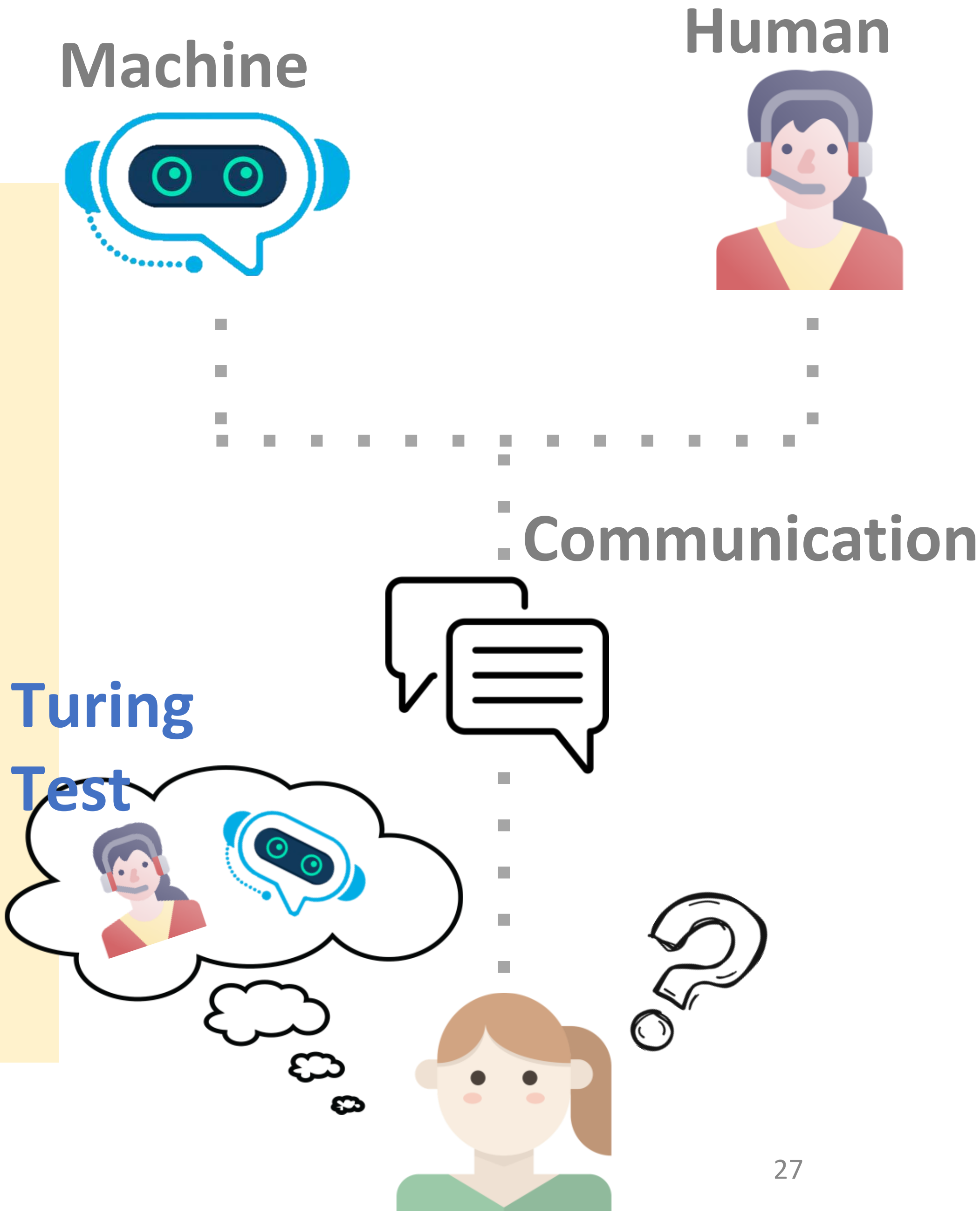
Semantically-Conditioned Generative Pre-Training (SC-GPT) Model

Peng et al. Few-shot Natural Language Generation for Task-Oriented Dialog (Arxiv' 20)

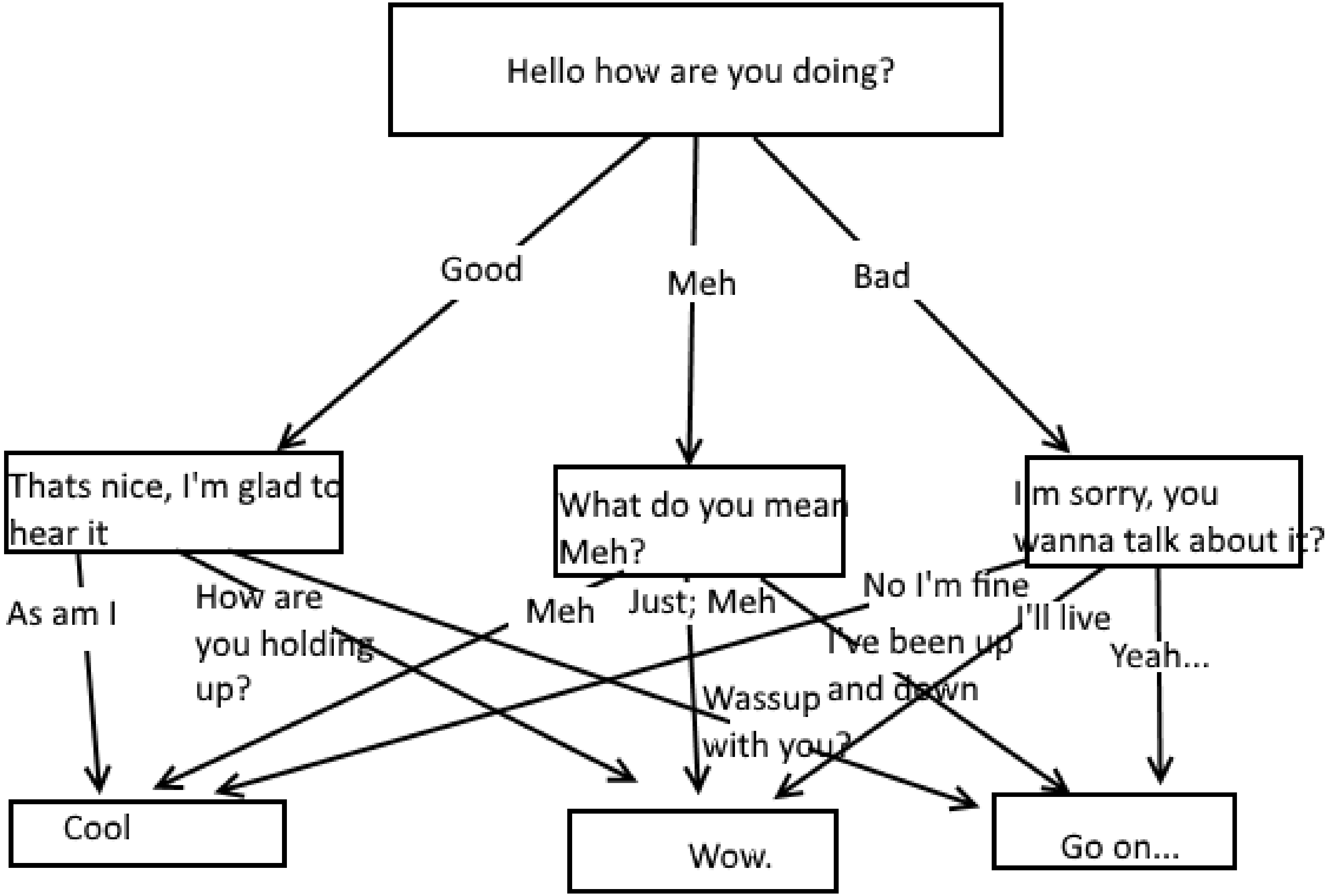


• Non-task-oriented Dialogue System

- **Chit-chat:** casual and non-goal-oriented.
- **Open domain and open ended**
- **Challenges:**
  - Coherence
  - Diversity
  - Engagement
  - …
- **Ultimate goal:** to pass Turing Test



• Template-based (Rule-based) Solution

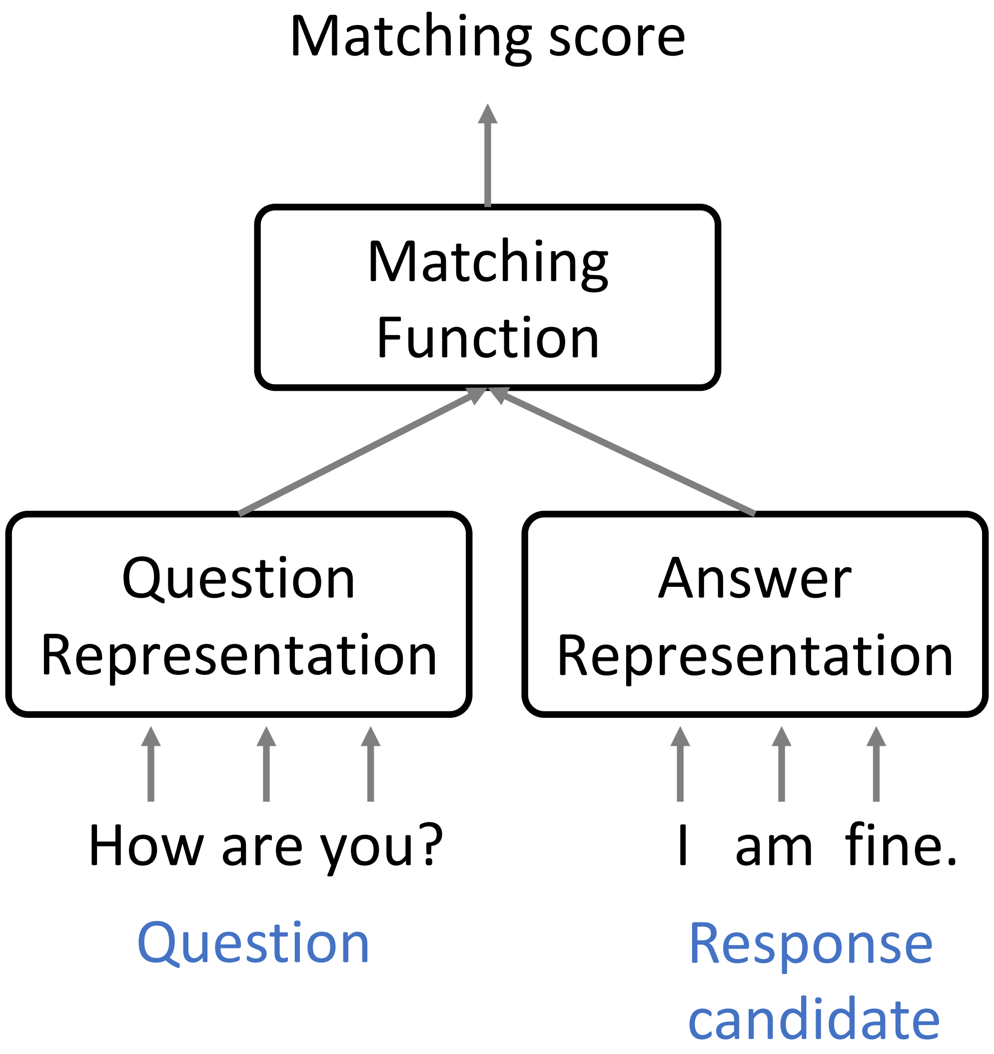


- **Unscalable**: require human labor
- **Inflexible**: hard to adopt to unseen topic

• Retrieval-based Solution

❑**Assumption:**

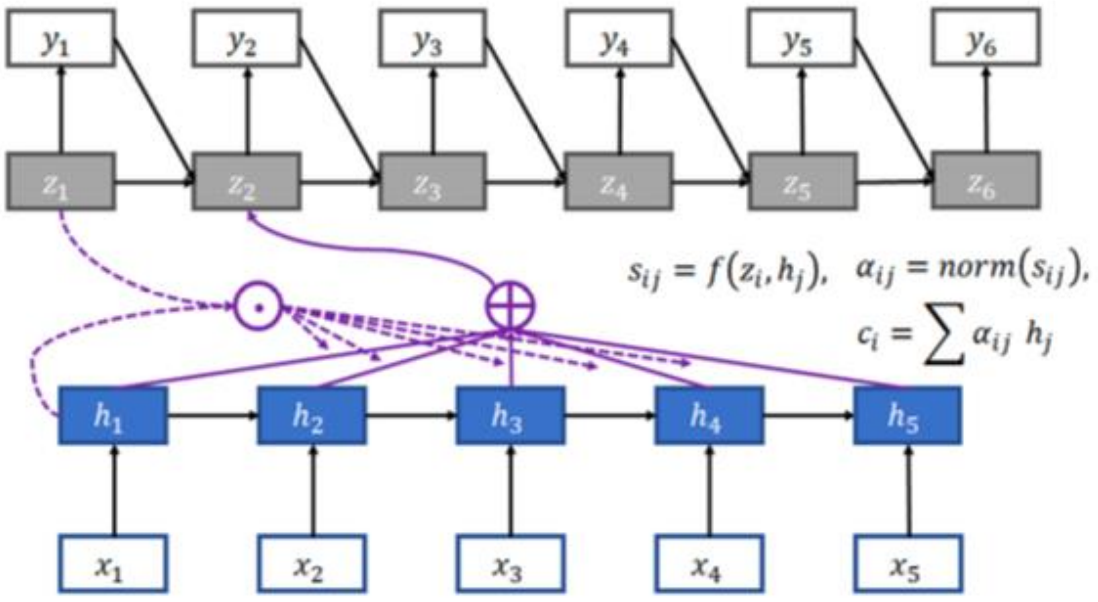
- A large candidate response set such that all input utterances can get a proper response.



• Generation-based Solution -- Classical Sequence to Sequence

- **Challenges:**
  - **Blandness**
    - Basic models tend to generate generic responses like ``I see'' and ``OK''.
  - **Consistency**
    - Logical self-consistent across multiple turns, e.g., persona, sentiment
  - **Lack of Knowledge**
    - Typical sequence-to-sequence models only mimic surface level sequence ordering patterns without understanding world knowledges deeply.

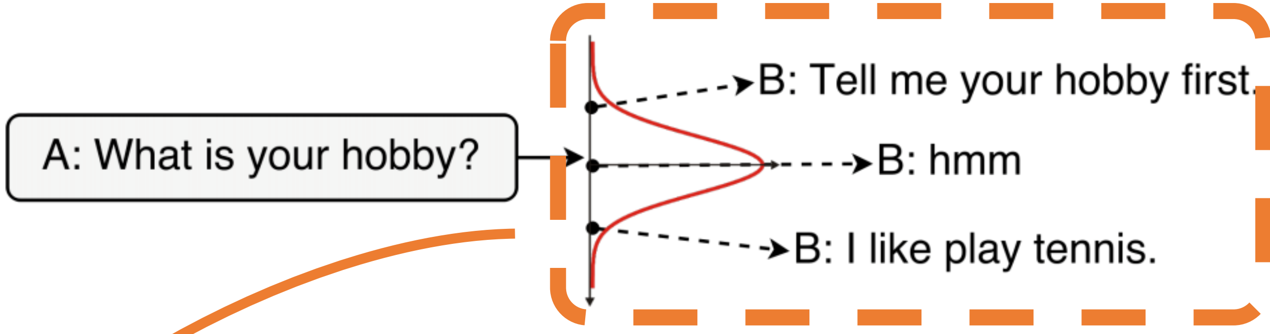
A Basic Model:
Encoder-Attention-Decoder



Wu et al. Deep Chit-Chat: Deep Learning for ChatBots (EMNLP' 18)

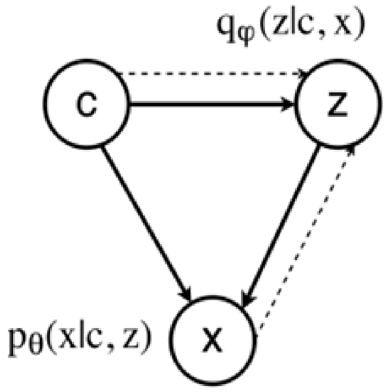
• Blandness：VAE-based solution

- **Problem in chatbot:**
  - The lack of diversity: often generate dull and generic response.



- **Solution:**
  - Using latent variables to learn a **distribution over potential conversation actions.**
  - Using **Conditional Variational Autoencoders (CVAE)** to infer the latent variable.

(CVAE)



- c: dialog history information
- x: the input user utterance
- z: latent vector of distribution of intents
- y: linguistic feature knowledge

Zhao et al. "Learning Discourse-level Diversity for Neural Dialog Models using Conditional Variational Autoencoders?"(ACL' 17)

• Consistency: Persona chat

- **Motivation:**
  - The lack of a consistent personality
  - A tendency to produce non-specific answers like "I don't know"
- **Solution: endowing machines with a configurable and consistent persona (profile), making chats condition on:**
  1. The machine' own given profile information.
  2. Information about the person the machine is talking to.

Persona of two interlorlocutors

Persona 1	Persona 2
I like to ski	I am an artist
My wife does not like me anymore	I have four children
I have went to Mexico 4 times this year	I recently got a cat
I hate Mexican food	I enjoy walking for exercise
I like to eat cheetos	I love watching Game of Thrones

[PERSON 1:] Hi
[PERSON 2:] Hello ! How are you today ?
[PERSON 1:] I am good thank you , how are you.
[PERSON 2:] Great, thanks ! My children and I were just about to watch Game of Thron
[PERSON 1:] Nice ! How old are your children?
[PERSON 2:] I have four that range in age from 10 to 21. You?
[PERSON 1:] I do not have children at the moment.
[PERSON 2:] That just means you get to keep all the popcorn for yourself.
[PERSON 1:] And Cheetos at the moment!
[PERSON 2:] Good choice. Do you watch Game of Thrones?
[PERSON 1:] No, I do not have much time for TV.
[PERSON 2:] I usually spend my time painting: but, I love the show.

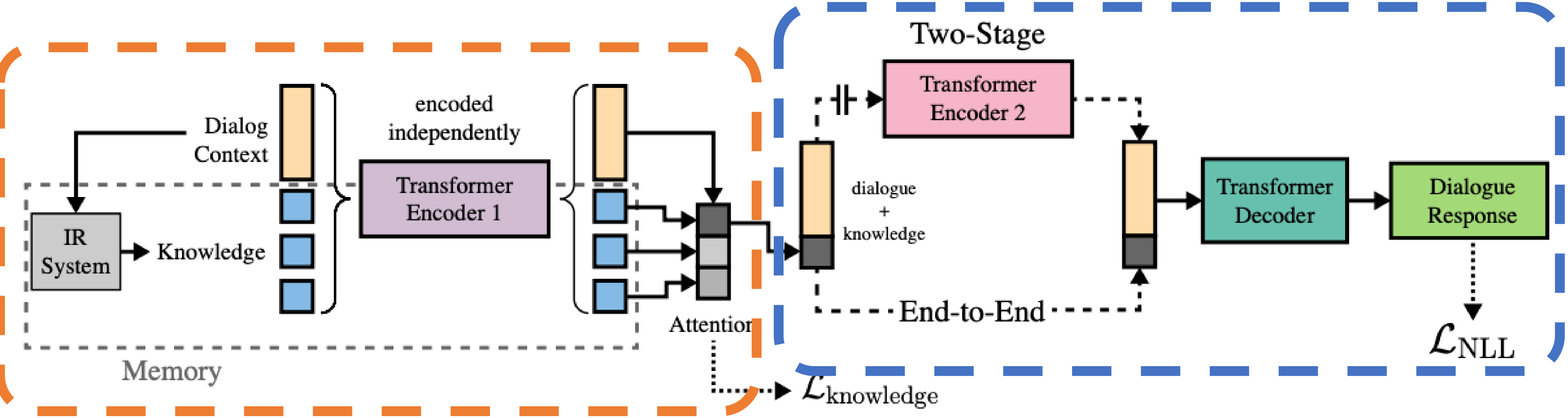
Wu et al. "Personalizing Dialogue Agents: I have a dog, do you have pets too?"(EMNLP' 18)

• Lack of background knowledge: Knowledge grounded dialogue response generation -- Text

- **Solution:** Knowledge retrieval from texts (e.g., Wikipedia) into dialogue responses

Response generated by integrating knowledge

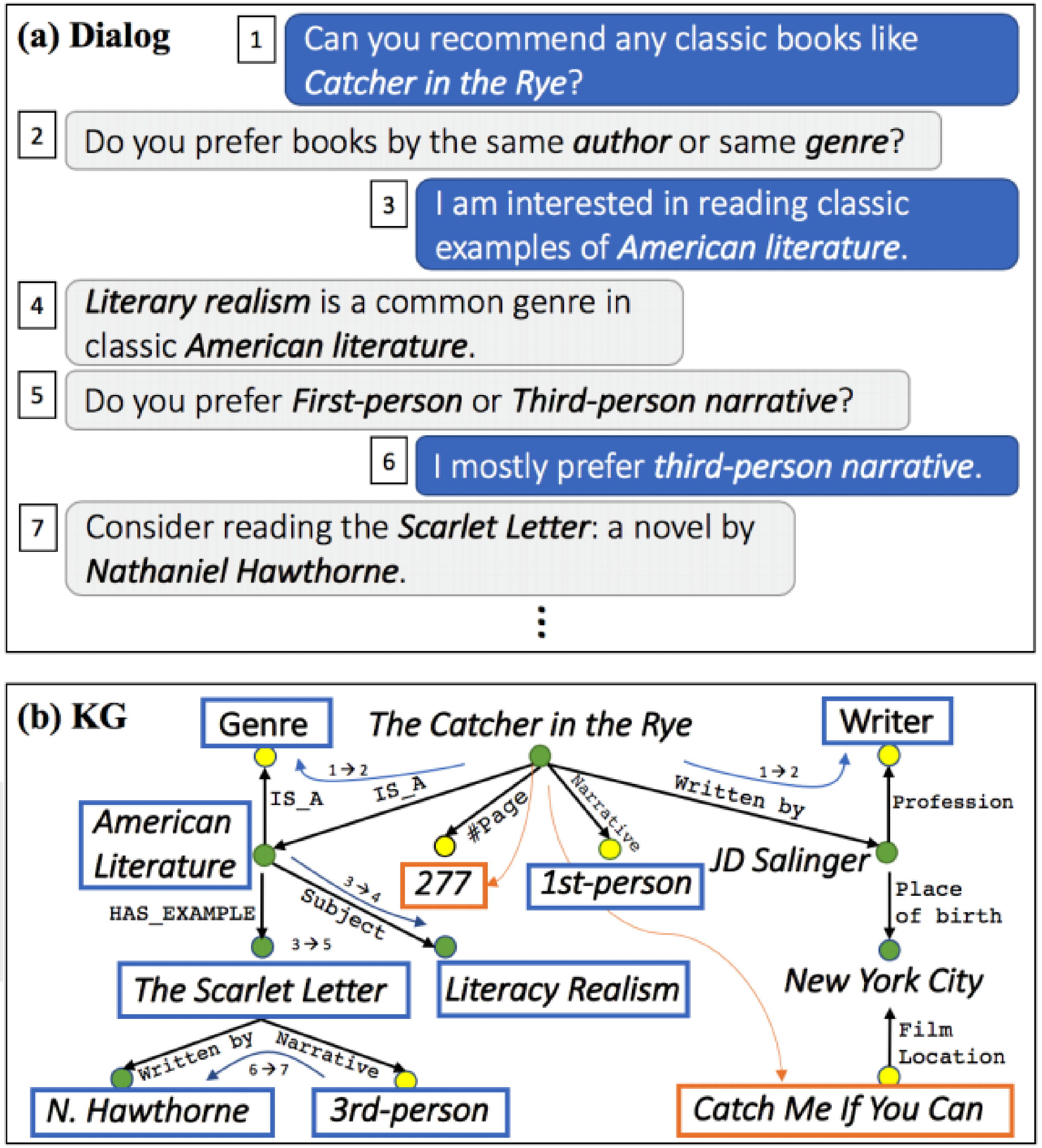
Knowledge retrieval module



Dinan et al. "Wizard of Wikipedia: Knowledge-Powered Conversational agents" (ICLR' 19)

- **Lack of background knowledge: Knowledge grounded dialogue response generation -- KG**

- **Solution: Walking within a large knowledge graph to**
  - track dialogue states.
  - to guide dialogue planning



Blue arrow: walkable paths led to engaging dialogues
Orange arrow: non-ideal paths that never mentioned (Should be pruned)

Moon et al. "OpenDialKG: Explainable Conversational Reasoning with Attention-based Walks over Knowledge Graphs" (ACL' 19)

• Tutorial Outline

- ❑A Glimpse of Dialogue System
- ❑Four research directions in conversational recommendation system
  - ❑**Question Driven Approaches**
  - ❑Multi-turn Conversational Recommendation Strategy
  - ❑Dialogue Understanding and Generation
  - ❑Exploitation-Exploration Trade-offs for Cold Users
- ❑ Summary of Formalizations and Evaluations

• System Ask – User Respond (SAUR) - Formalization

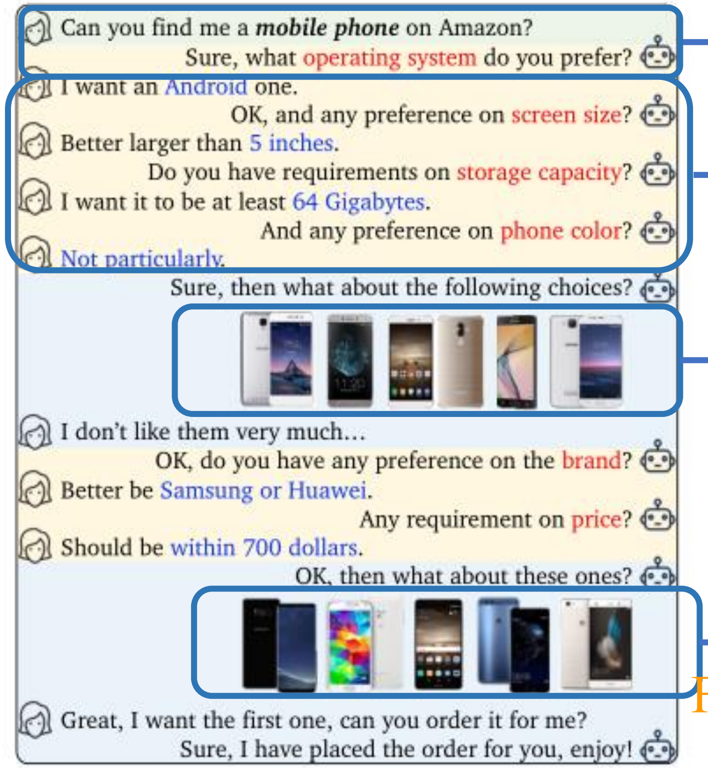


Figure 1: Example for conversational search in e-commerce product search or recommendation scenario (best in color).

Initial request

Get feedback

Feels confident

Research Question -- Given the requests specified in dialogues, the system needs to predict:

1. What questions to ask
2. What items to recommend

1. Initiation

   User initiates a conversation

2. Conversation

   Asks the user preferences on product aspects

3. Display

   Display product to the user

three stages

Zhang et al. "Towards Conversational Search and Recommendation: System Ask, User Respond"(CIKM' 18)

• SAUR – Method -- Representation

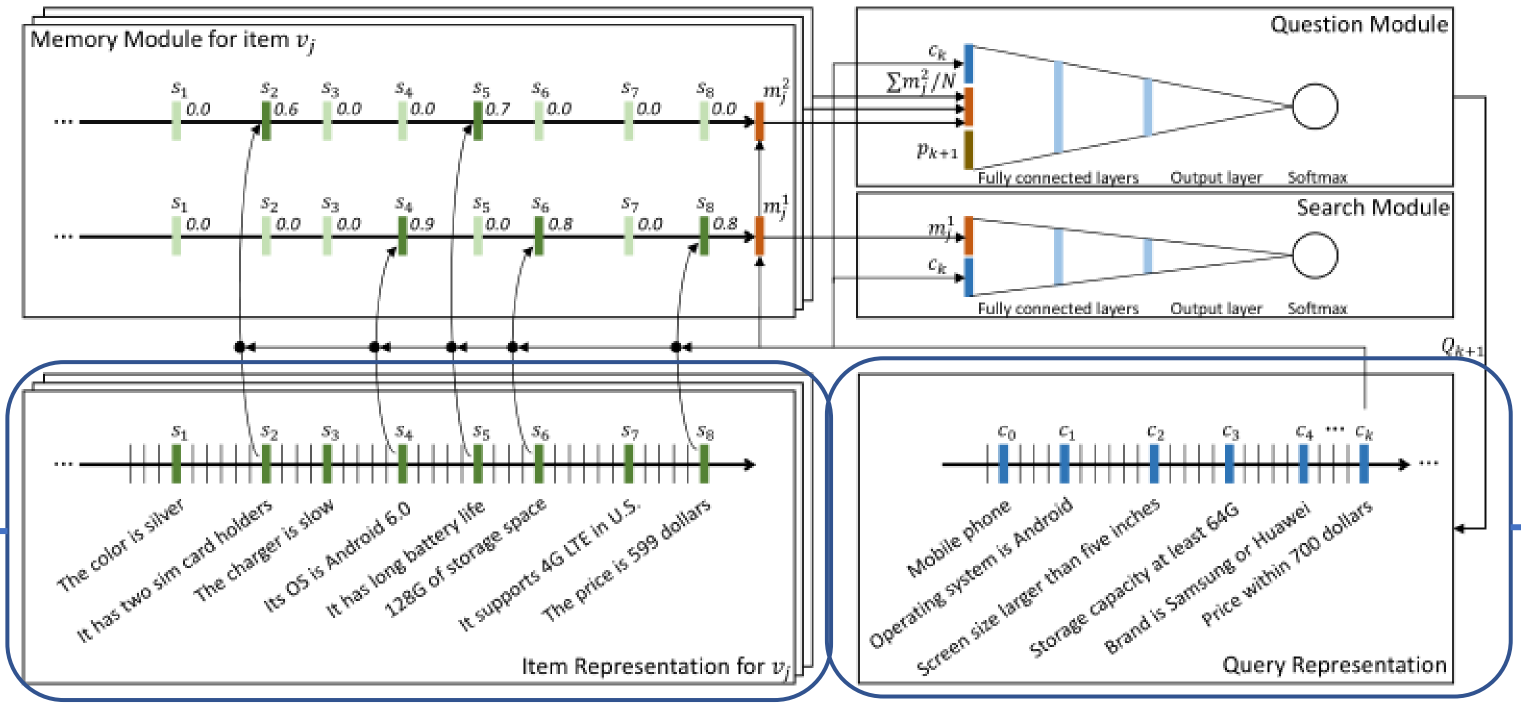


Figure 3: The Multi-Memory Network (MMN) architecture for conversational search and recommendation, including five components: query and item representations, the memory module, as well as the question and search modules.

Item Representations

- Apply a gated recurrent unit (GRU) on the text description of each item
- The hidden states of each sentence as the representation of $\boldsymbol{T}_j(s_1, s_2 \ldots s_\tau)$

Query Representation

- Also a gated recurrent unit (GRU)
- Query sequence c1, c2 … is extracted in conversations

Zhang et al. "Towards Conversational Search and Recommendation: System Ask, User Respond"(CIKM' 18)

• SAUR - Method

Joint optimize

Question Loss + Search (item) Loss → The Unified Architecture

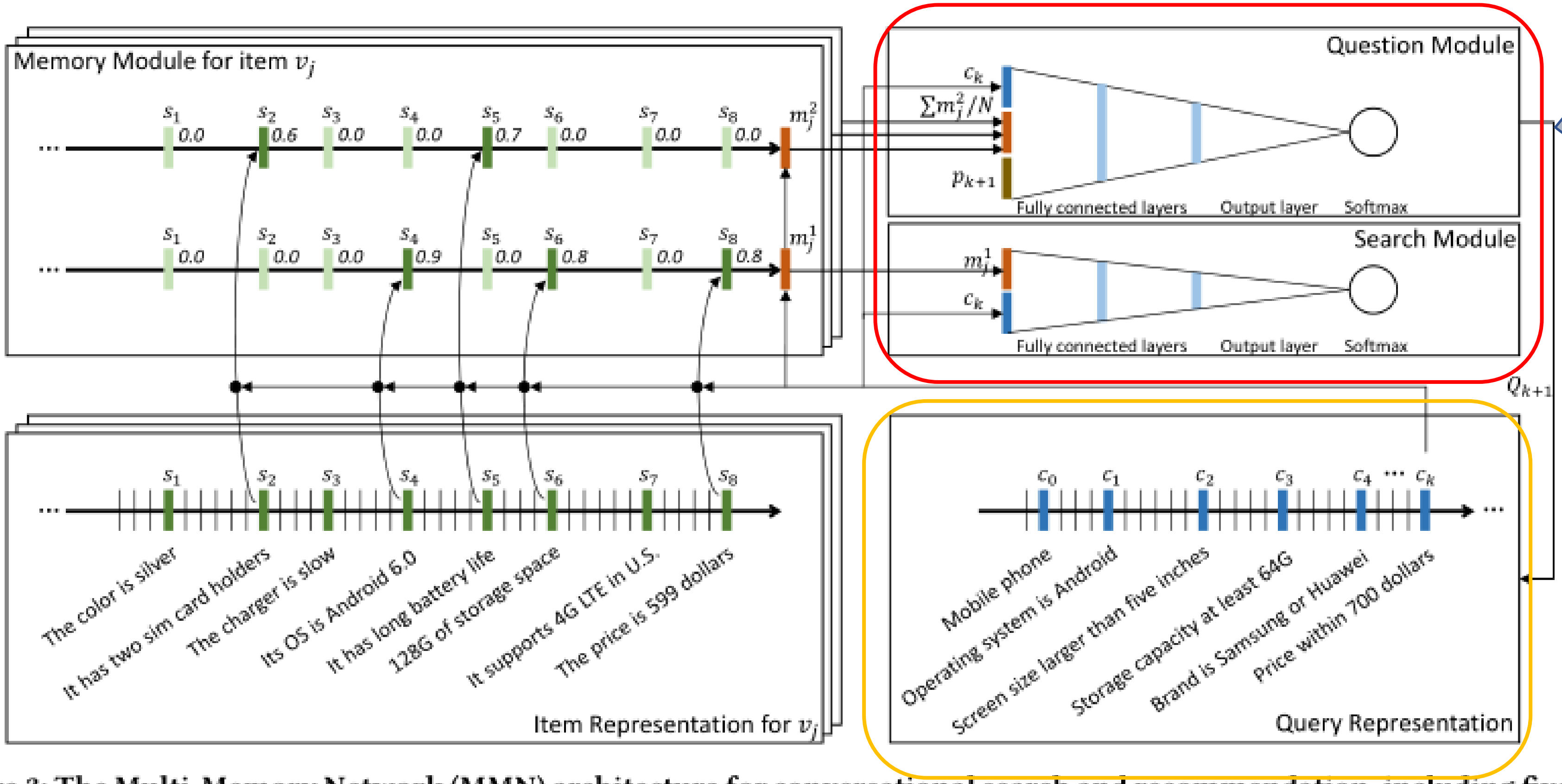


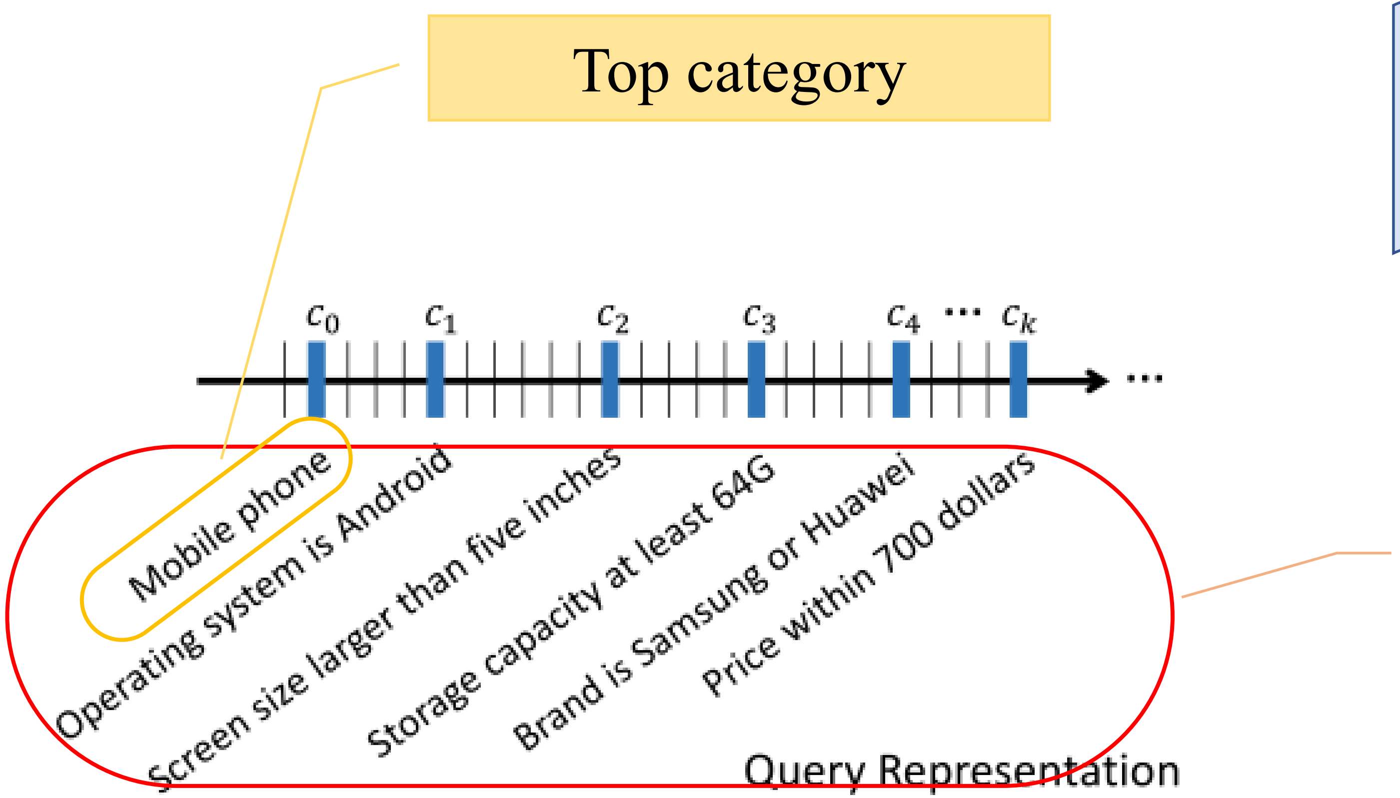
Figure 3: The Multi-Memory Network (MMN) architecture for conversational search and recommendation, including five components: query and item representations, the memory module, as well as the question and search modules.

c_0 : constructed from the product category c

c_k : Question-answer sequence (The order mentioned in the review)

Zhang et al. “Towards Conversational Search and Recommendation: System Ask, User Respond”(CIKM’ 18)

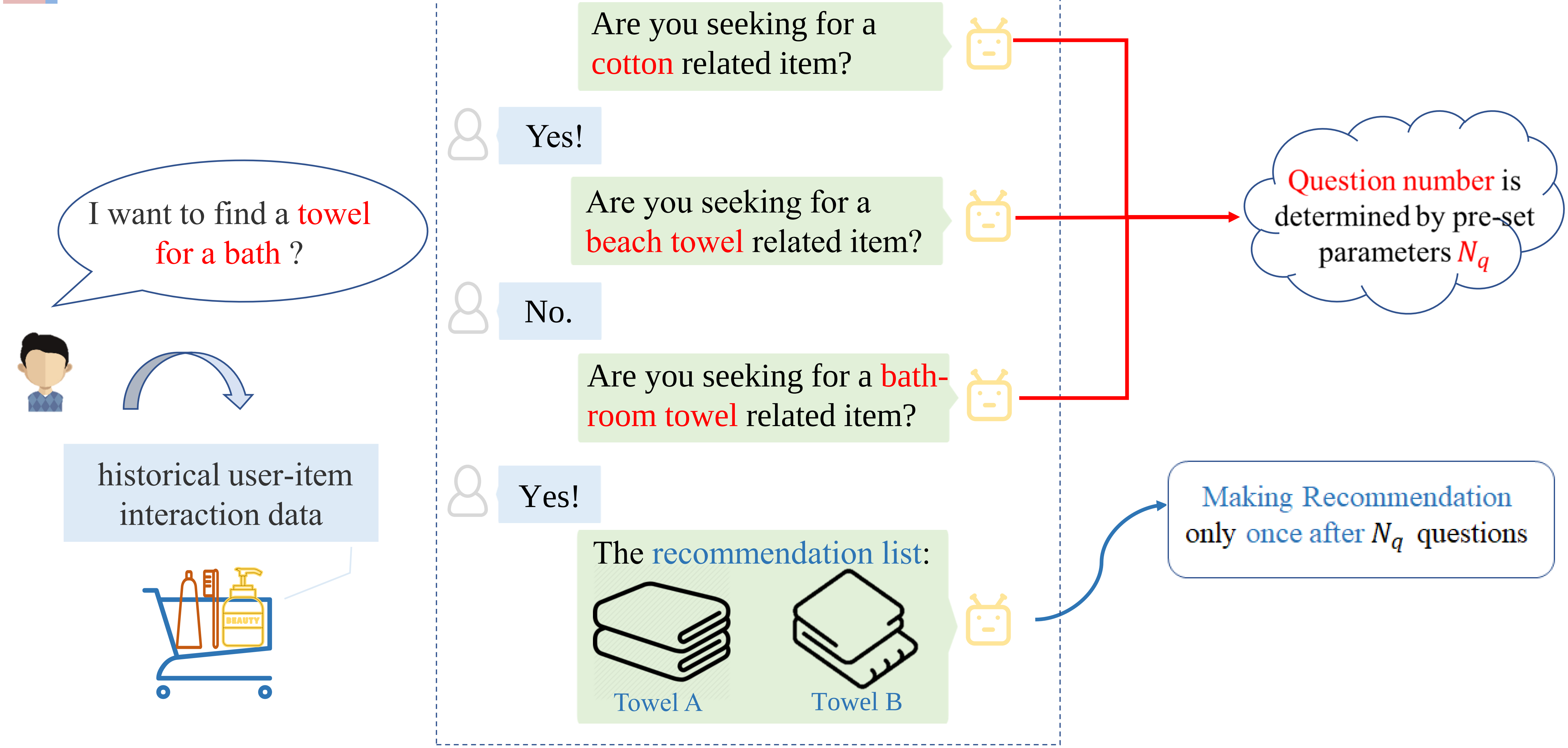
• SAUR - Evaluation



Evaluation Criteria:

1. Query prediction
2. Item prediction (e.g., NDCG)

• Question-based recommendation(Qrec) - Formalization



Zou et al. "Towards Question-based Recommender Systems"(SIGIR' 20)

• Qrec - Method -- Offline and Online Optimization

Latent Factor Recommendation

Y_{ij}: the number of attributes of item j that satisfy the user i in current conversation.
For example, if a user specify [cotton] and [towel], but the item only gets [cotton], Y=1.

⇓

$$-\log p(\mathbf{U},\mathbf{V}\mid \mathbf{R},\mathbf{Y},\Theta) \propto \boxed{\frac{1}{2}\sum_{i,j\in \mathrm{R}}\left(R_{ij}-\mathbf{p}^{\top}(\mathbf{u}_i\circ\mathbf{v}_j)\right)^2} + \frac{\gamma}{2}\boxed{\sum_{i,j\in \mathrm{Y}}\left(Y_{ij}-\mathbf{q}^{\top}(\mathbf{u}_i\circ\mathbf{v}_j)\right)^2} + \sum_{i=1}^{M}\frac{\lambda_u}{2}\|\mathbf{u}_i\|_2^2 + \sum_{j=1}^{N}\frac{\lambda_v}{2}\|\mathbf{v}_j\|_2^2 + \frac{\lambda_p}{2}\|\mathbf{p}\|_2^2 + \frac{\lambda_q}{2}\|\mathbf{q}\|_2^2, \quad (2)$$

Offline Optimization

Online Optimization（feedback from user, (i.e. Y) ）

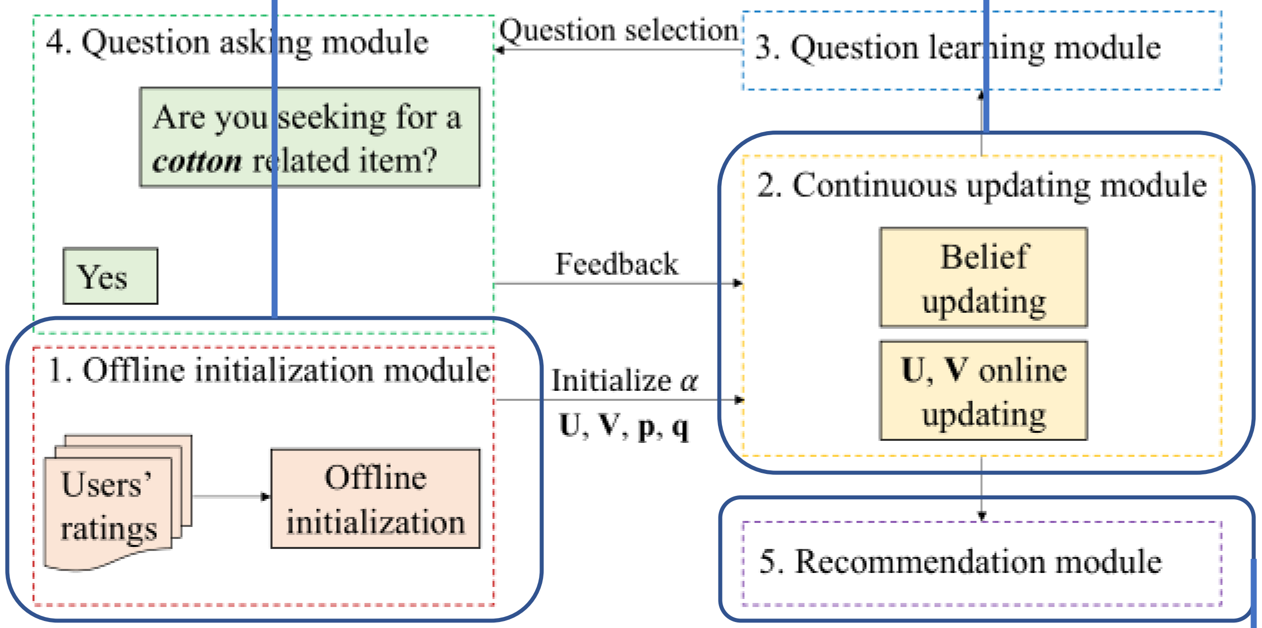


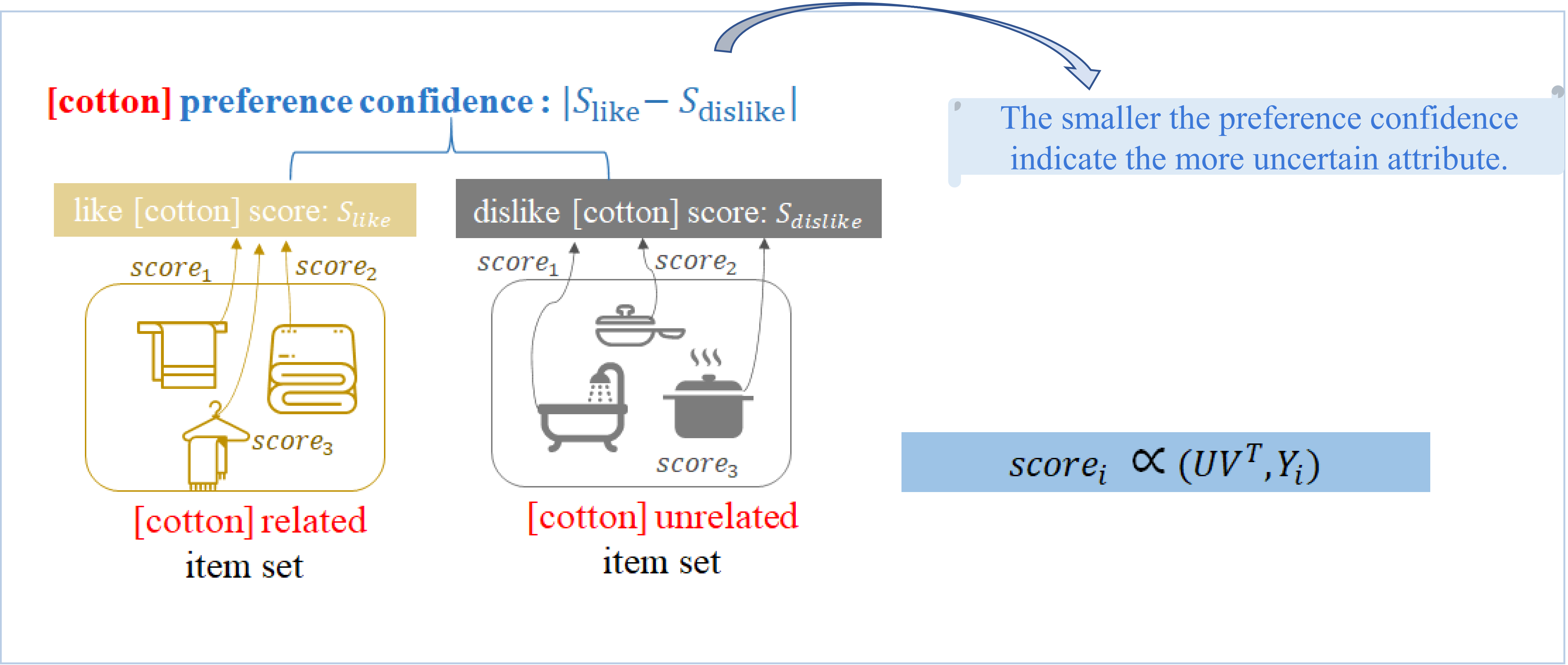
Figure 1: Framework of our proposed question-based recommendation model, Qrec. *Cotton* is an extracted entity (informative term), U, V, p, q are model variables, and α is a hyper-parameter of user belief.

Ranking UV^T ⇒ Recommendation list

Zou et al. "Towards Question-based Recommender Systems"(SIGIR' 20)

• Qrec - Method -- Choosing Questions to Ask

Attribute Choosing criteria: Finding the most uncertain [attribute] to ask.



Zou et al. "Towards Question-based Recommender Systems"(SIGIR' 20)

• Qrec - Evaluation

Evaluation Measures:
recall@5, MRR, NDCG
only on items!
No questions are evaluated, but if question asking strategy is bad, the item recommendation results will not be good.

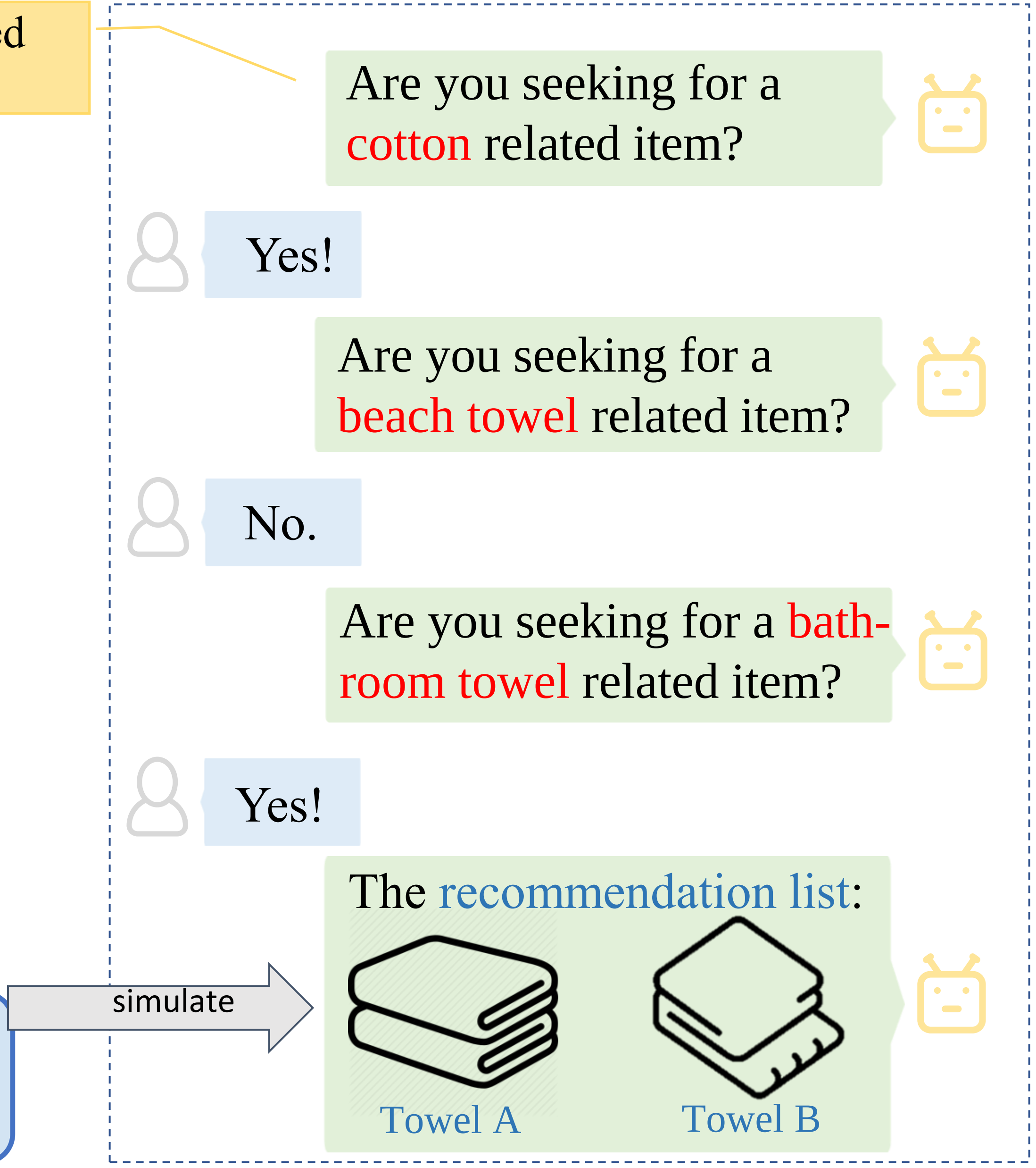
Dataset: *Amazon product dataset*

- Using TAGME (an entity linking tool) to find the entities in the product description page as the attributes.

Item Name: "*Cotton Hotel spa Bathroom Towel*"
Item Attributes: [cotton, bathroom, hand towels]

Simulating Users

Template-based question



Zou et al. "Towards Question-based Recommender Systems"(SIGIR' 20)

• Question & Recommendation(Q&R) - Formalization

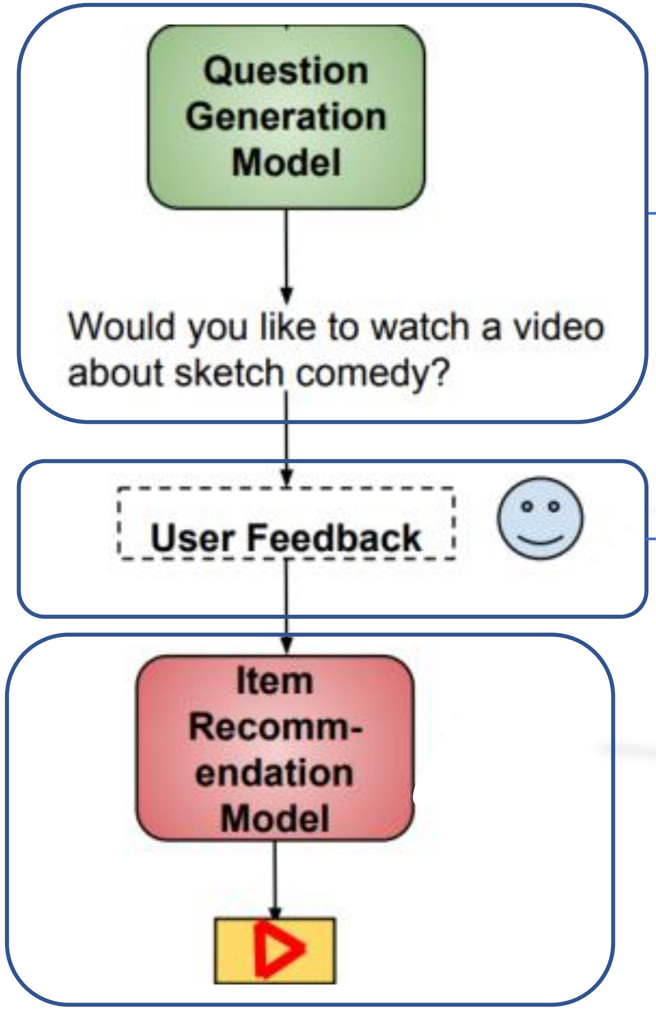


Figure 1: High-level *Q&R* overview.

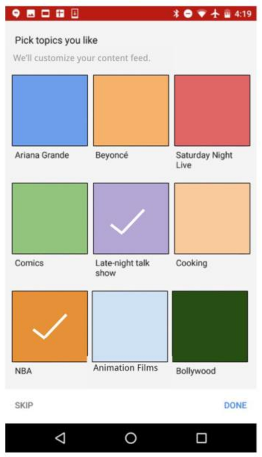


Figure 2: User Onboarding UI.

Only asking question once and make one recommendation

Positive-only type of feedback (click topics)

User is prompted to choose as many topics as they like

Incorporates the user feedback to improve video recommendations

Christakopoulou et al. "Q&R: A Two-Stage Approach toward Interactive Recommendation"(KDD' 18)

• Q&R - Method

Two Main Tasks

feedback

building better user profiles

i.e., predicting the sequential future (interested topic)

What to ask

step1

How to respond

step2

given the video(user interests)

i.e., predicting the video that the user be most interested in

the sequence of watch videos

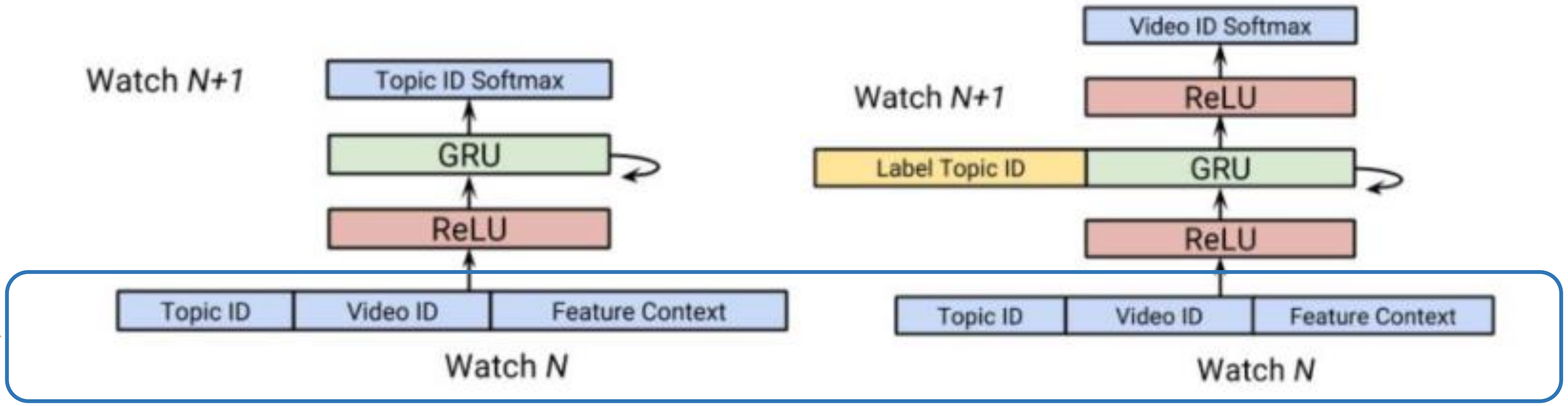


Figure 3: *Left:* Topic Prediction (Question Ranking) Model. *Right:* Post-Fusion Approach for Response Model.

Christakopoulou et al. "Q&R: A Two-Stage Approach toward Interactive Recommendation"(KDD' 18)

• Q&R - Evaluation

Offline Evaluation

Data

YouTube user watch sequences

1. The watch sequence of a user up until the previous to last step
2. The video ID and topic ID of the user's last watch event

watched video id (until t)

watched video topic id (until t)

feature context (until t)

video topic id (t+1)

Target video id (t+1)

Online Evaluation

Figure 2: User Onboarding UI.

Christakopoulou et al. "Q&R: A Two-Stage Approach toward Interactive Recommendation"(KDD' 18)

• Tutorial Outline

- ❑A Glimpse of Dialogue System
- ❑Four research directions in conversational recommendation system
  - ❑Question Driven Approaches
  - ❑**Multi-turn Conversational Recommendation Strategy**
  - ❑Dialogue Understanding and Generation
  - ❑Exploitation-Exploration Trade-offs for Cold Users
- ❑ Summary of Formalizations and Evaluations

• CRM - Formalization

Scenario: single round of a conversation between a user and the system

Recommend once and break the dialogue

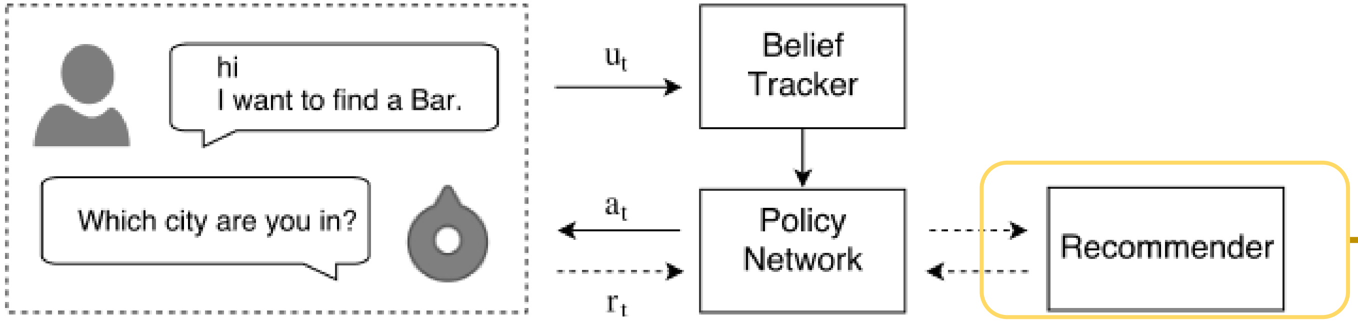


Figure 1: The conversational recommender system overview

Recommender System

- Make a recommendation only once after asking question.

Sun et al. "Conversational Recommender System"(SIGIR' 18)

• CRM - Method -- Dialogue Component

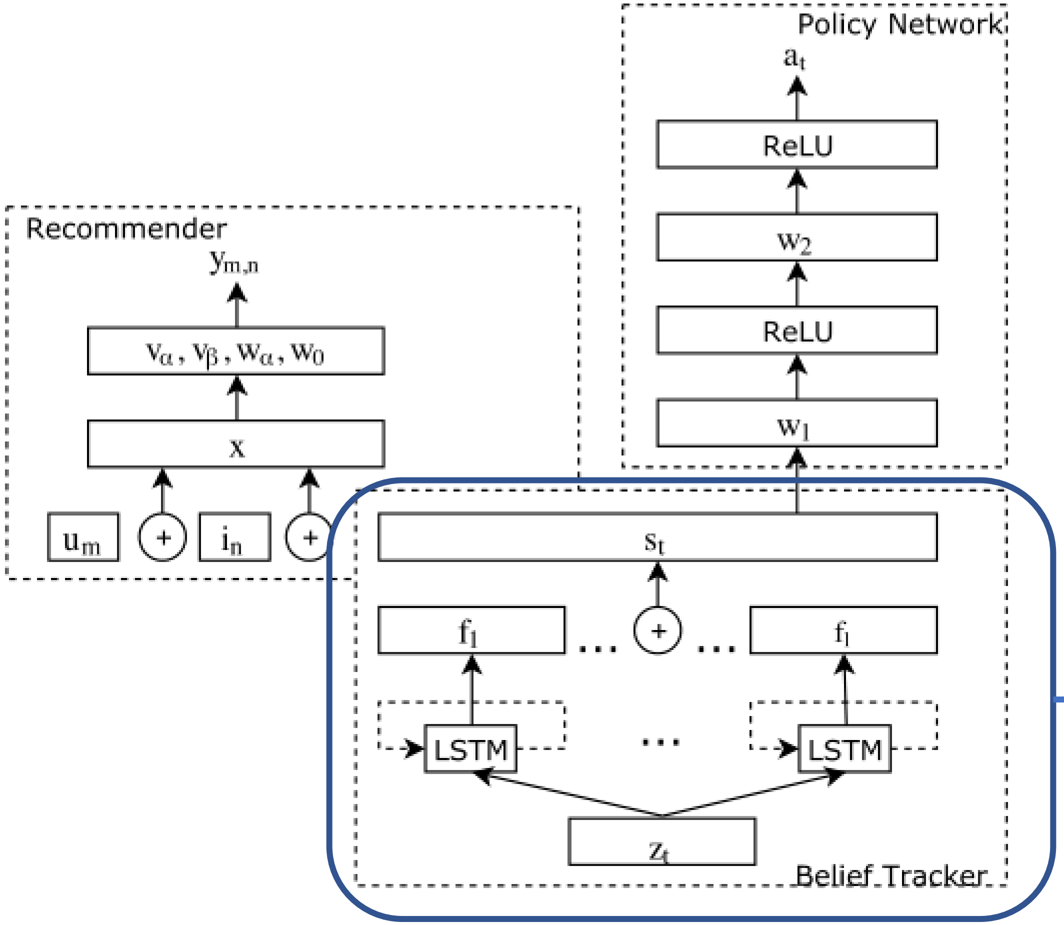


Figure 2: The structure of the proposed conversational recommender model. The bottom part is the belief tracker, the top left part is the recommendation model, and the top right part is the deep policy network.

Belief Tracker

- **Input:** the current and the past user utterances representation Zt

LSTM

- **Output:** a probability distribution of facets

$$s_t = \mathbf{f}_1 \oplus \mathbf{f}_2 \ldots \oplus \mathbf{f}_l$$

vector representation for the facet i

the agent's current belief of the dialogue state

Sun et al. "Conversational Recommender System"(SIGIR' 18)

• CRM - Method

User feedback is not encoded

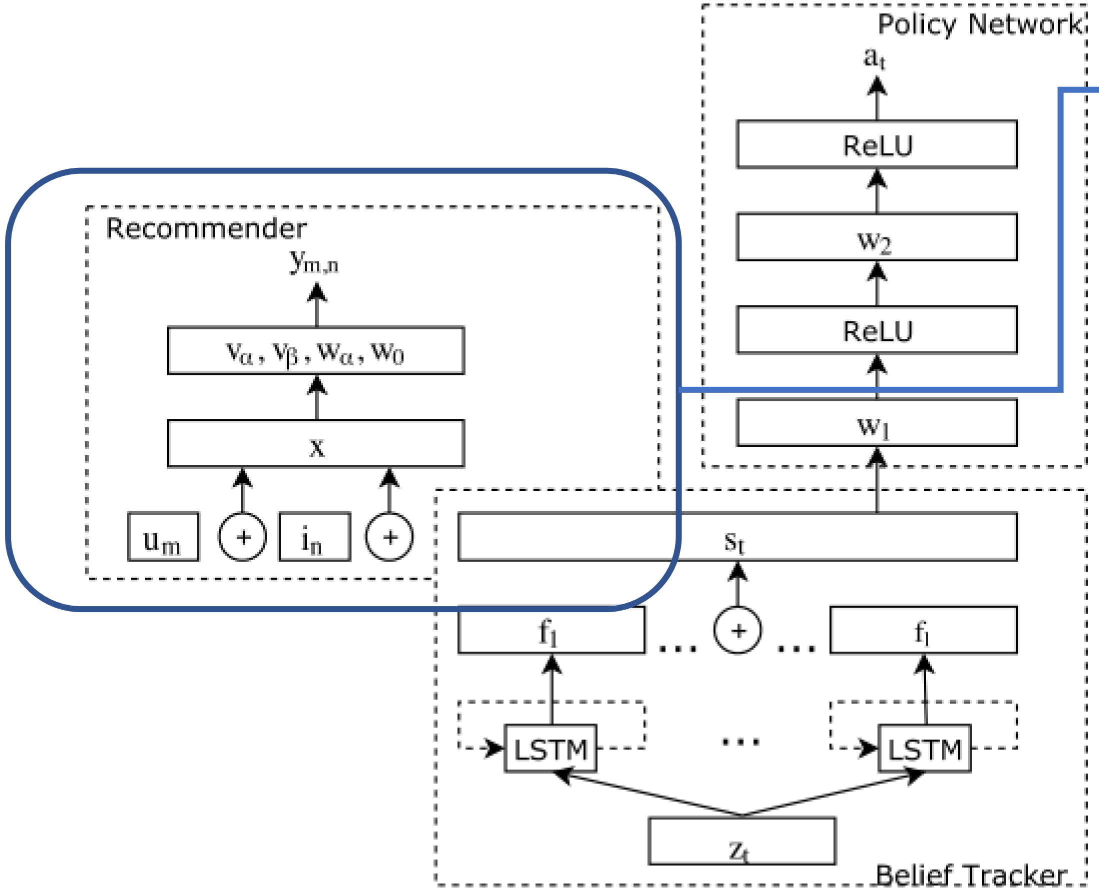


Figure 2: The structure of the proposed conversational recommender model. The bottom part is the belief tracker, the top left part is the recommendation model, and the top right part is the deep policy network.

Recommender System

- **Input:** $\mathbf{x} = \mathbf{u}_m \oplus \mathbf{i}_n \oplus \mathbf{s}_t$

u_m : user
i_n : item

1-hot encoded user/item vector

Factorization Machine (FM)

- **Output:** $y_{m,n}$

a rating score

Sun et al. "Conversational Recommender System"(SIGIR' 18)

• CRM - Method

Decisions based only on the belief tracker

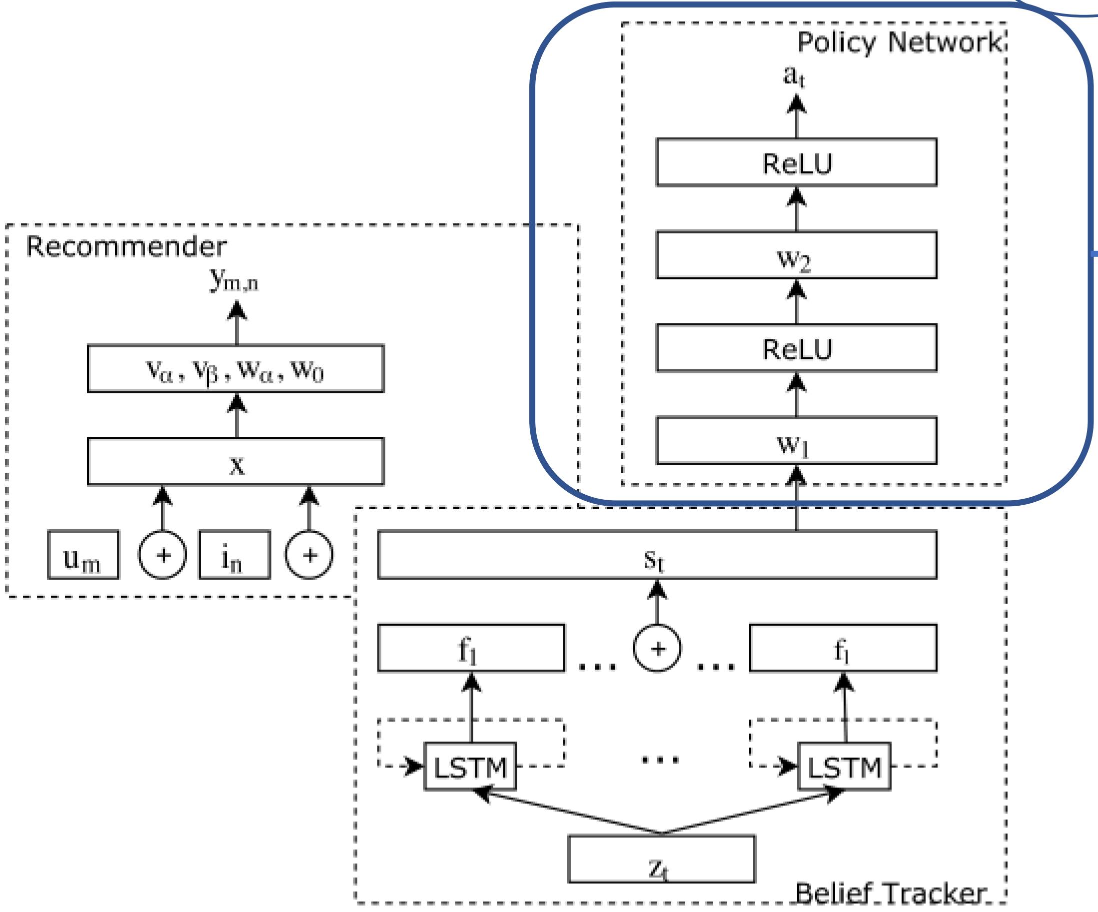


Figure 2: The structure of the proposed conversational recommender model. The bottom part is the belief tracker, the top left part is the recommendation model, and the top right part is the deep policy network.

Sun et al. "Conversational Recommender System"(SIGIR' 18)

Deep Policy Network

- **State:** $s_t = \{\mathbf{f}_1 \oplus \mathbf{f}_2 \ldots \oplus \mathbf{f}_l\}.$

  Description of the conversation context

- **Action:** $\{a_1, a_2, ..., a_l\},$ request the value of a facet; $a_{rec},$ make a personalized recommendation

- **Reward:** benefit/penalty the agent gets from interacting with its environment

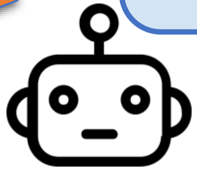
- **Policy:** $\pi(a_t|s_t),$ two fully connected layers as the policy network

Adopt the policy gradient method of reinforcement learning

• CRM - Evaluation

User Simulation

Yelp (the restaurants and food data)



Item Name: "*Small Italy Restaurant*"
Item Attributes: [Italian, San Diego, California, cheap, rating>=3.5]

(city="Italian", category="San Diego")

I'm looking for Italian food in San Diego.

Which **state** are you in?

I'm in California. *(state="CA")*

Which **price** range do you like?

Low price *(price_range="cheap")*

What **rating** range do you want?

3.5 or higher. *(rating_range>="3.5")*

Do you want "Small Italy Restaurant"?

thank you!

Evaluation Metrics

Evaluation Matrices:

- SR @ k (Success rate at k-th turn)
- AT (Average Turns)

$$SR = \frac{\#sucessful\,dialogues}{\#dialogues} \cdot 100\%$$

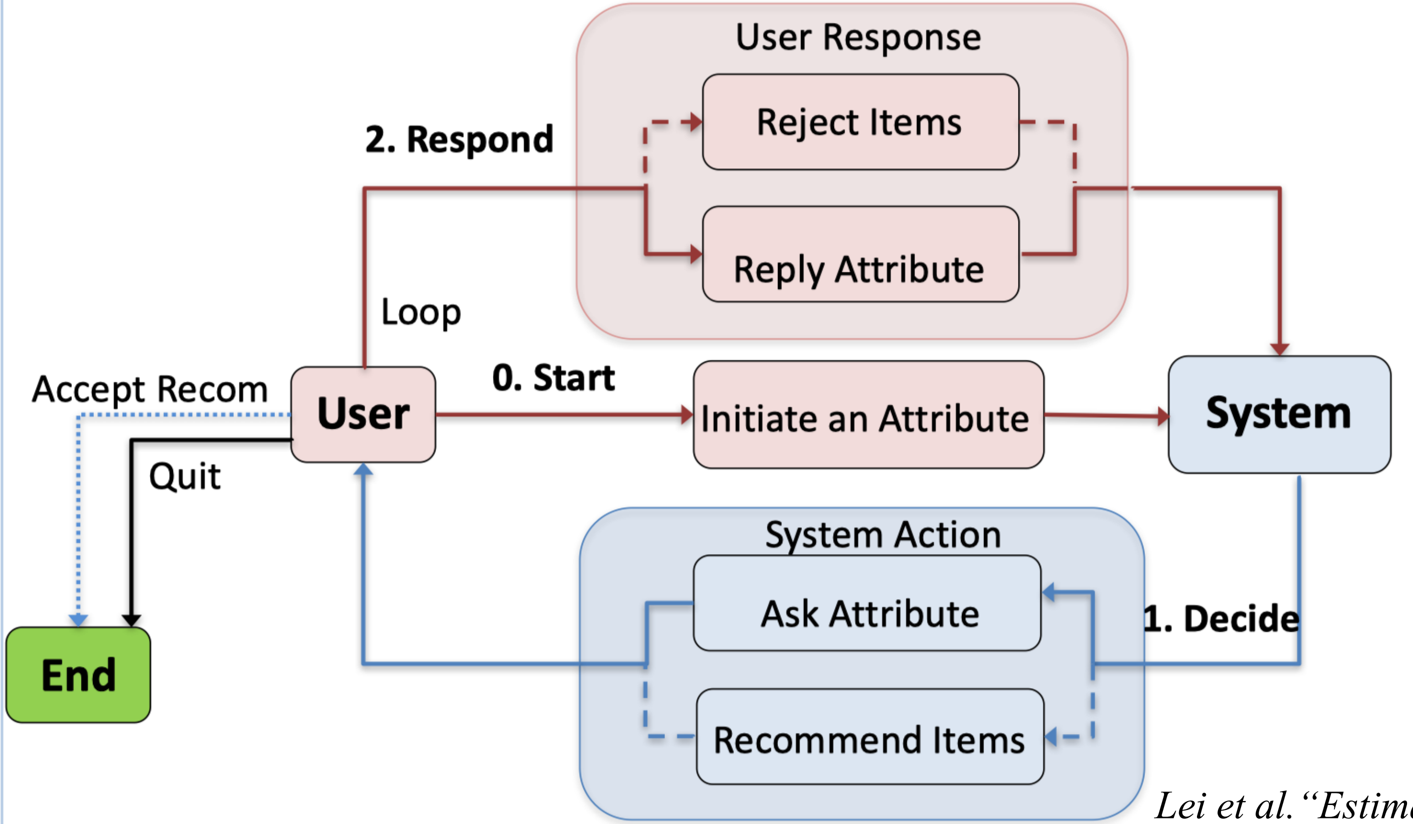
$$AT = \overline{dialogue\,length.}$$

Sun et al. "Conversational Recommender System"(SIGIR' 18)

• Estimation–Action–Reflection(EAR) - Formalization

Workflow of Multi-round Conversational Recommendation (MCR)

Objective:
Recommend desired items to user in shortest turns



- Key Research Questions

1. What item/attribute to recommend/ask?
1. Strategy to ask and recommend?
1. How to adapt to user's online feedback?

Lei et al. "Estimation–Action–Reflection: Towards Deep Interaction Between Conversational and Recommender Systems" (WSDM'20)

• EAR - Method -- What Item to Recommend and What Attribute to Ask

Method: Attribute-aware FM for Item Prediction and Attribute Preference Prediction

Notation	Meaning
$\boldsymbol{p}$	A given attribute
$\boldsymbol{u}$	User embedding
$\boldsymbol{\mathcal{P}_u}$	User's known preferred attributes

Notation	Meaning
(Neg. 1) $\mathcal{V}_u^- := \mathcal{V} \setminus \mathcal{V}_u^+$	The ordinary negative sample as in standard BPR.
(Neg. 2) $\widehat{\mathcal{V}_u^-} := \mathcal{V}_{cand} \setminus \mathcal{V}_u^+$	$\mathcal{V}_{cand}$ is the set of candidate items satisfying user's preferred attributes.
$\mathcal{D}_1 := \{(u, v, v') \mid v' \in \mathcal{V}_u^-\}$	Paired sample for first kind of negative sample
$\mathcal{D}_2 := \{(u, v, v') \mid v' \in \widehat{\mathcal{V}_u^-}\}$	Paired sample for second kind of negative sample

$$\hat{y}(u, v, \mathcal{P}_u) = \mathbf{u}^T \mathbf{v} + \sum_{p_i \in \mathcal{P}_u} \mathbf{v}^T \mathbf{p_i}$$

Score function for item prediction

$$L_{item} = \sum_{(u,v,v') \in \mathcal{D}_1} -\ln \sigma\left(\hat{y}(u, v, \mathcal{P}_u) - \hat{y}(u, v', \mathcal{P}_u)\right) + \sum_{(u,v,v') \in \mathcal{D}_2} -\ln \sigma\left(\hat{y}(u, v, \mathcal{P}_u) - \hat{y}(u, v', \mathcal{P}_u)\right) + \lambda_\Theta \|\Theta\|^2$$

ordinary negative example

The items satisfying the specified attribute but still are not clicked by the user

Lei et al. "Estimation–Action–Reflection: Towards Deep Interaction Between Conversational and Recommender Systems" (WSDM'20)

• EAR - Method -- What Item to Recommend and What Attribute to Ask

Method: Attribute-aware FM for Item Prediction and Attribute Preference Prediction

Notation	Meaning
$\boldsymbol{p}$	A given attribute
$\boldsymbol{u}$	User embedding
$\boldsymbol{\mathcal{P}_u}$	User's known preferred attributes

Notation	Meaning
(Neg. 1) $\mathcal{V}_u^- := \mathcal{V} \setminus \mathcal{V}_u^+$	The ordinary negative sample as in standard BPR.
(Neg. 2) $\widehat{\mathcal{V}_u^-} := \mathcal{V}_{cand} \setminus \mathcal{V}_u^+$	$\mathcal{V}_{cand}$ is the set of candidate items satisfying user's preferred attributes.
$\mathcal{D}_1 := \{(u, v, v') \mid v' \in \mathcal{V}_u^-\}$	Paired sample for first kind of negative sample
$\mathcal{D}_2 := \{(u, v, v') \mid v' \in \widehat{\mathcal{V}_u^-}\}$	Paired sample for second kind of negative sample

$$\hat{g}(p|u, \mathcal{P}_u) = \mathbf{u}^T p + \sum_{p_i \in \mathcal{P}_u} \mathbf{P}^T \mathbf{P}_i$$

Score function for attribute preference prediction

$$L_{attr} = \sum_{(u,p,p') \in \mathcal{D}_3} - \ln \sigma \left(\hat{g}(p|u, \mathcal{P}_u) - \hat{g}(p'|u, \mathcal{P}_u) \right) + \lambda_\Theta \|\Theta\|^2$$

$$L = L_{item} + L_{attr}$$

Multi-task Learning: Optimize for item ranking and attribute ranking simultaneously.

Lei et al. "Estimation–Action–Reflection: Towards Deep Interaction Between Conversational and Recommender Systems" (WSDM'20)

• EAR - Method -- Action stage

Method: Strategy to Ask and Recommend? (Action Stage)

We use reinforcement learning to find the best strategy.

- policy gradient method
- simple policy network (2-layer feedforward network)

- **State Vector**
- $s_{entropy}$: **The entropy of attribute is important.**
- $s_{prefrence}$: **User's preference on each attribute.**
- $s_{history}$: **Conversation history is important.**
- s_{length}: **Candidate item list length.**

Note: 3 of the 4 information come from Recommender Part

Action Space: $|\mathcal{P}| + 1$

Reward

$r_{success}$: Give the agent a big reward when it successfully recommend!

r_{ask}: Give the agent a small reward when it ask a correct attribute.

r_{quit}: Give the agent a big negative reward when the user quit (the conversation is too long)

$r_{prevent}$: Give each turn a relatively small reward to prevent the conversation goes too long.

Lei et al. "Estimation–Action–Reflection: Towards Deep Interaction Between Conversational and Recommender Systems" (WSDM'20)

• EAR - Method -- Reflection

Method: How to Adapt to User's Online Feedback? (Reflection stage)

Solution: We treat the recently rejected 10 items as negative samples to re-train the recommender, to adjust the estimation of user preference.

$$L_{ref} = \sum_{(u,v,v')\in\mathcal{D}_4} -\ln\sigma\left(\hat{y}(u,v,\mathcal{P}_u) - \hat{y}(u,v',\mathcal{P}_u)\right) + \lambda_\Theta\|\Theta\|^2$$

Notation	Meaning
$\mathcal{V}^t$	Recently rejected item set.
$\mathcal{D}_4 \coloneqq \{(u,v,v') \mid v' \in \mathcal{V}_u^+ \wedge v' \in \mathcal{V}^t\}$	Paired sample for online update.

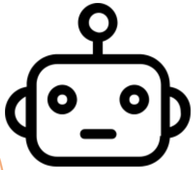
Lei et al.“Estimation–Action–Reflection: Towards Deep Interaction Between Conversational and Recommender Systems” (WSDM’20)

• EAR - Evaluation

Table 1: Dataset statistics.

Dataset	#users	#items	#interactions	#attributes
Yelp	27,675	70,311	1,368,606	590
LastFM	1,801	7,432	76,693	33

Item Name: "*Small Italy Restaurant*"
Item Attributes: [Pizza, Nightlife, Wine, Jazz]



I'd like some Italian food.

Got you, do you like some pizza?

Yes!

Got you, do you like some nightlife?

Yes!

Do you want "Small Paris"?

Rejected!

Got you, do you like some Rock Music?

No!

Do you want "Small Italy Restaurant"?

Accepted!

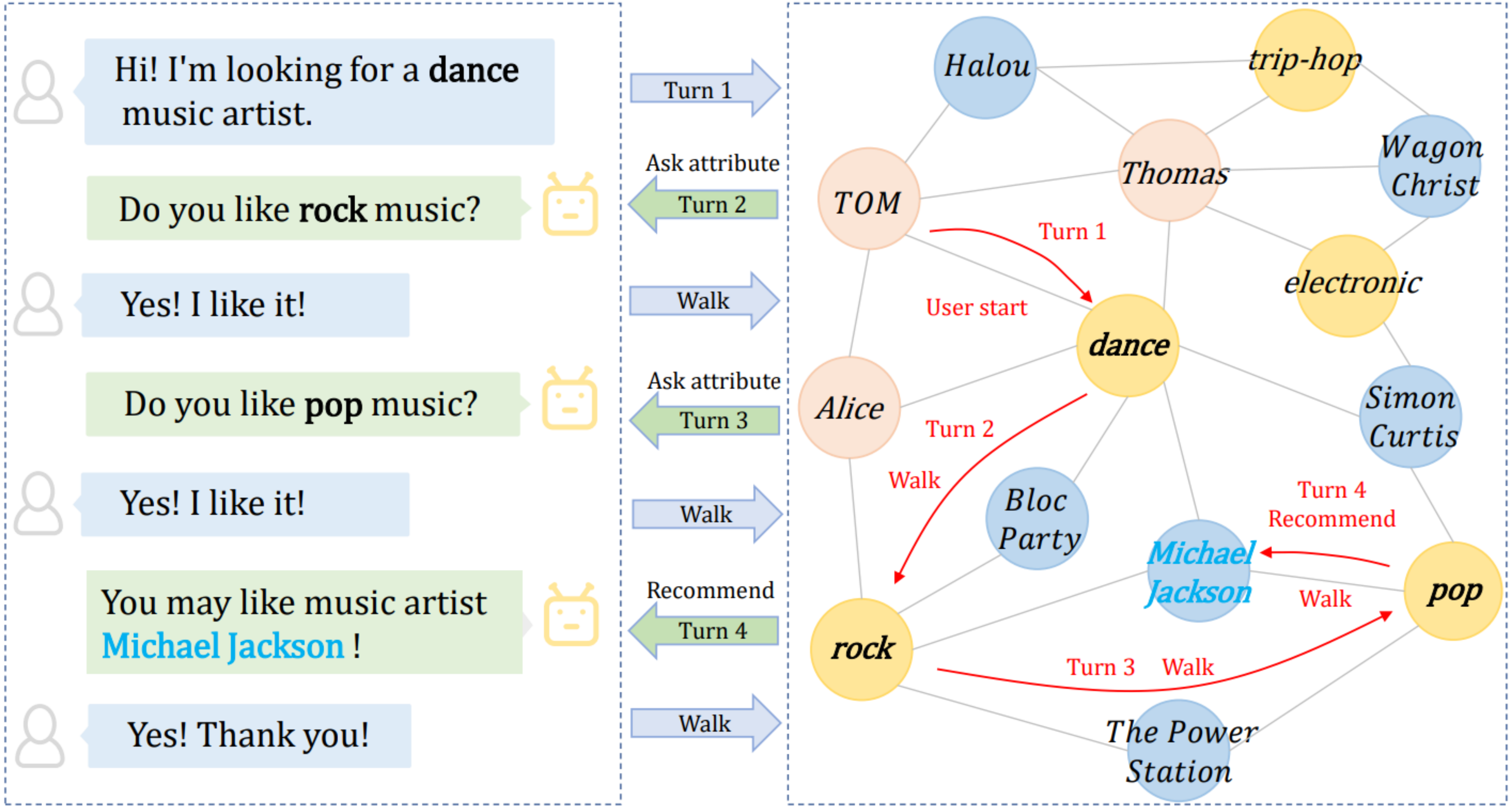
Check, I don't want "Small Paris"

Check, I don't want "Rock Music"

Template-based utterances

Lei et al. "Estimation–Action–Reflection: Towards Deep Interaction Between Conversational and Recommender Systems" (WSDM'20)

• CPR - Motivation



Lei et al. "Interactive Path Reasoning on Graph for Conversational Recommendation" (KDD'20)

• CPR - Method

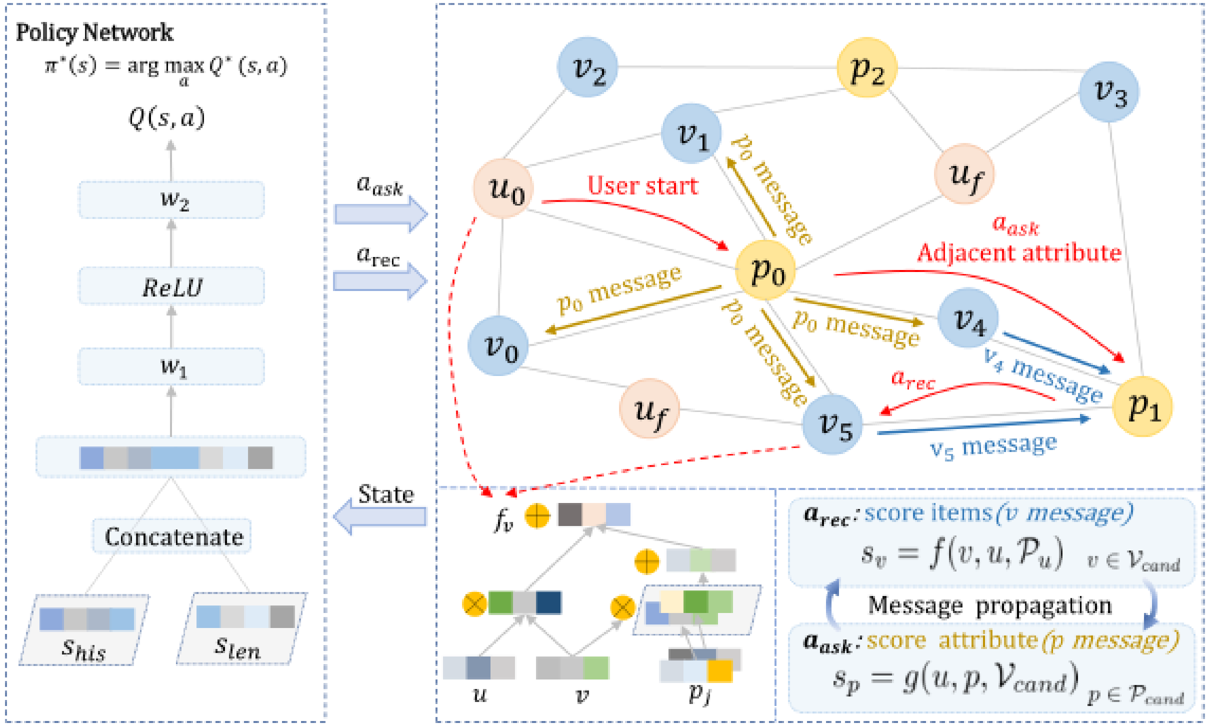


Figure 2: CPR framework overview. It starts from the user u_0 and walks over adjacent attributes, forming a path (the red arrows) and eventually leading to the desired item. The policy network (left side) determines whether to ask an attribute or recommend items in a turn. Two reasoning functions f and g score attributes and items, respectively.

CPR Framework

- **Assuming**
  - Current path $P = p_0, p_1, p_2 \ldots, p_t$
  - u: user v: item p: attribute
  - $\mathcal{P}_u$: user's preferred attributes
  - $\mathcal{V}_{cand}$: candidate items
- **Reasoning**
  - Score items to recommend (v message):

    $$s_v = f(v, u, \mathcal{P}_u)$$
  - Score attribute to ask (p message):

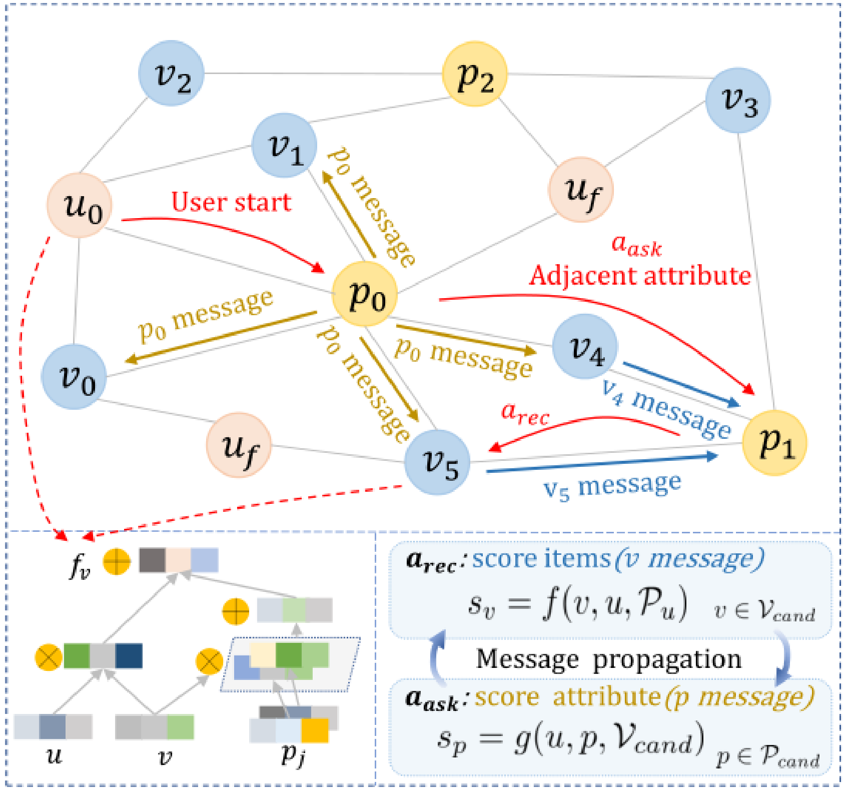
    $$s_p = g(u, p, \mathcal{V}_{cand})$$
- **Consultation**
  - Policy network (choose to ask or rec)
- **Transition**
  - Extended path

    $$P = p_0, p_1, p_2 \ldots p_t\ p_{t+1}$$
  - Update candidate item /attribute set ($\mathcal{V}_{cand}/\mathcal{P}_{cand}$)

Lei et al. "Interactive Path Reasoning on Graph for Conversational Recommendation" (KDD'20)

• CPR - Method

An instantiation of CPR Framework



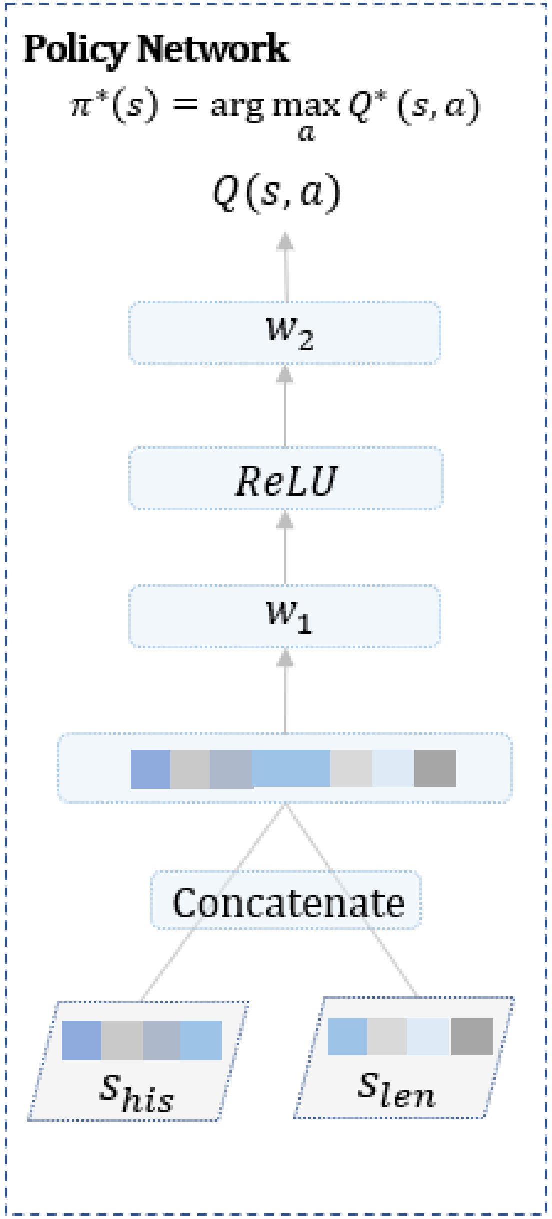
- u: user v: item p: attribute
- P_u: user's preferred attributes
- L_{item}: item prediction loss
- L_{atti}: attribute prediction loss

Message propagation from attributes to items	Message propagation from items to attributes
<u>Factorization Machine in EAR</u> • Item prediction $$f(v, u, \mathcal{P}_u) = \mathbf{u}^T \mathbf{v} + \sum_{p \in \mathcal{P}_u} \mathbf{v}^T \mathbf{p},$$ • Optimization: Bayesian Personalized Ranking	<u>Information entropy strategy</u> • Weighted attribute information entropy $$g(u, p, \mathcal{V}_{cand}) = -\text{prob}(p) \cdot \log_2(\text{prob}(p)),$$ $$\text{prob}(p) = \frac{\sum_{v \in \mathcal{V}_{cand} \cap \mathcal{V}_p} \sigma(s_v)}{\sum_{v \in \mathcal{V}_{cand}} \sigma(s_v)}$$

The same with the recommender model in EAR

Lei et al. "Interactive Path Reasoning on Graph for Conversational Recommendation" (KDD'20)

• CPR - Method



Input

s_{his}: encodes the conversation history
s_{len}: encodes the size of candidate items

Output

$$Q(s,a)$$

$Q(s,a)$: the value of action a in state s
a_{rec}: the action of recommendation
a_{ask}: the action of asking attribute

DQN method

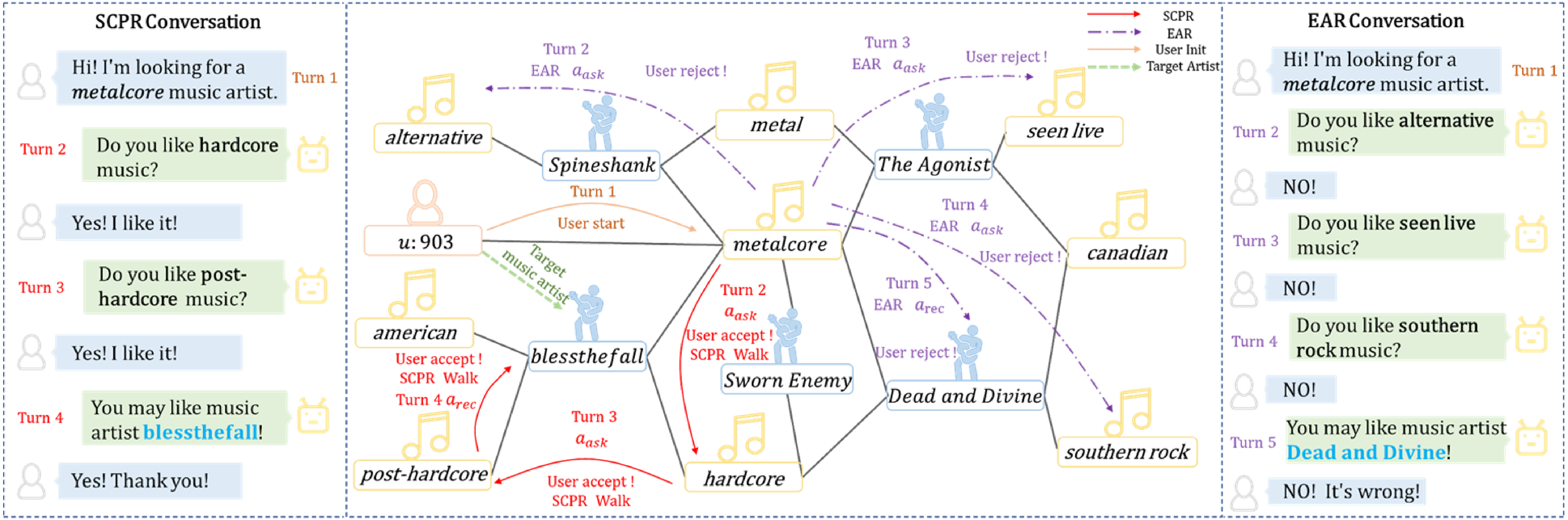
Policy: $\pi^*(s) = \arg\max_a Q^*(s,a)$

TD loss: $\delta = Q(s,a) - \left(R + \gamma \max_a Q(s',a)\right)$

Lei et al. "Interactive Path Reasoning on Graph for Conversational Recommendation" (KDD'20)

• CPR - Evaluation

CPR can make the reasoning process explainable and easy-to-interpret!



Sample conversations generated by SCPR (left) and EAR (right) and their illustrations on the graph (middle).

Lei et al. "Interactive Path Reasoning on Graph for Conversational Recommendation" (KDD'20)

• Tutorial Outline

- ❑ A Glimpse of Dialogue System
- ❑ Four research directions in conversational recommendation system
  - ❑ Question Driven Approaches
  - ❑ Multi-turn Conversational Recommendation Strategy
  - ❑ **Dialogue Understanding and Generation**
  - ❑ Exploitation-Exploration Trade-offs for Cold Users
- ❑ Summary of Formalizations and Evaluations

• ReDial - Formalization

HUMAN:	hello, can i help you find a movie ?
HRED:	hi what kind of movies do you like
OURS:	hi, how can i help you ?
SEEKER:	hello ! i would like to find a suspenseful, but clean family friendly movie .
...	...
SEEKER:	2001 : a space odyssey might be a great option. anything else that you would suggest ?
HUMAN:	you can try planet of the apes the older one is quite suspenseful and family friendly .
HRED:	i haven't seen that one yet but i heard it was good. i haven't seen that one. have you seen the last house on the left ?
OURS:	star wars : the force awakens is also a good one return of the jedi all good movies
SEEKER:	those sound good ! i 'm going to look into those movies.
HUMAN:	i hope you enjoy, have a nice one
HRED:	have you seen foxcatcher ? it 's about a man who has a rich guy.
OURS:	i hope i was able to help you find a good movie to watch
SEEKER:	thank you for your help ! have a great night ! good bye

Conversational recommendation through natural language (in movie domain)

- Seeker: explain what kind of movie he/she likes, and asks for movie suggestions
- Recommender: understand the seeker's movie tastes, and recommends movies

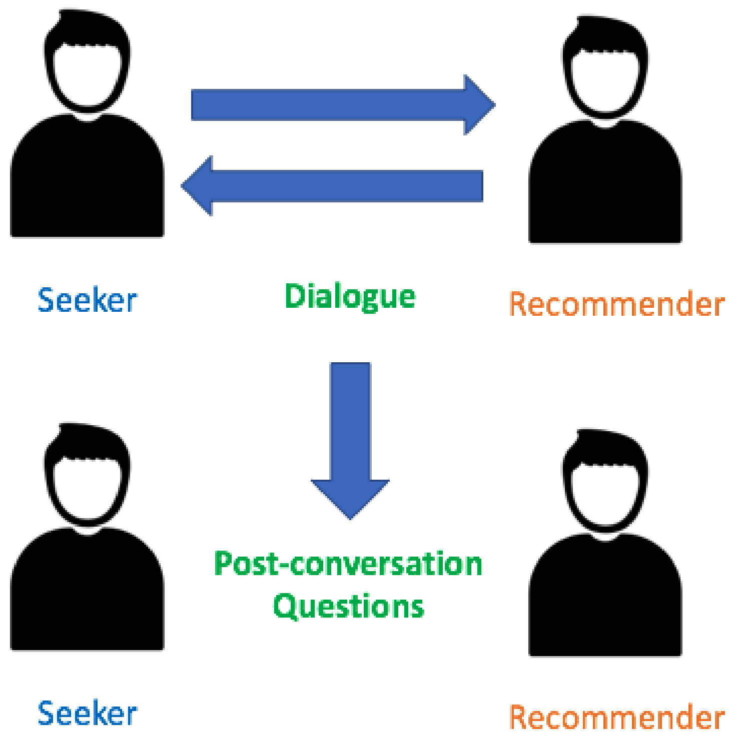
Li et al. "Towards Deep Conversational Recommendations" (NIPS' 18)

• ReDial – Formalization -- Dataset Collection

# conversations	10006
# utterances	182150
# users	956
# movie mentions	51699
seeker mentioned	16278
recommender suggested	35421
not seen	16516
seen	31694
did not say	3489
disliked (4.9%)	2556
liked (81%)	41998
did not say (14%)	7145

Data annotation on Amazon Mturk Platform

- 2 turkers: **Seeker** and **recommender** converse with each other.



Li et al. "Towards Deep Conversational Recommendations" (NIPS' 18)

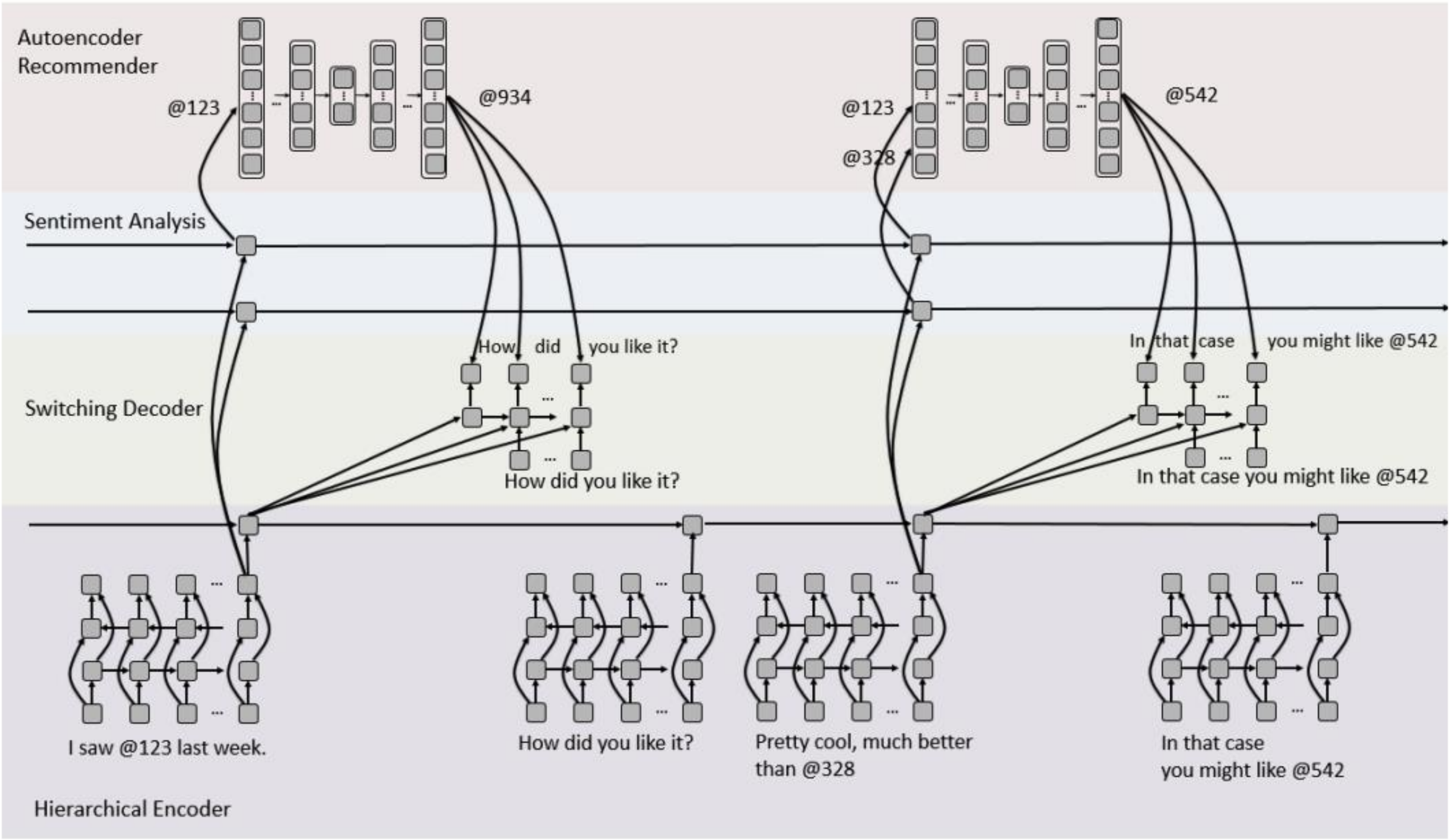
• ReDial – Methods – Overall

3
Recommender

2
Sentiment Analysis

4
Switching Decoder

1
Encoder

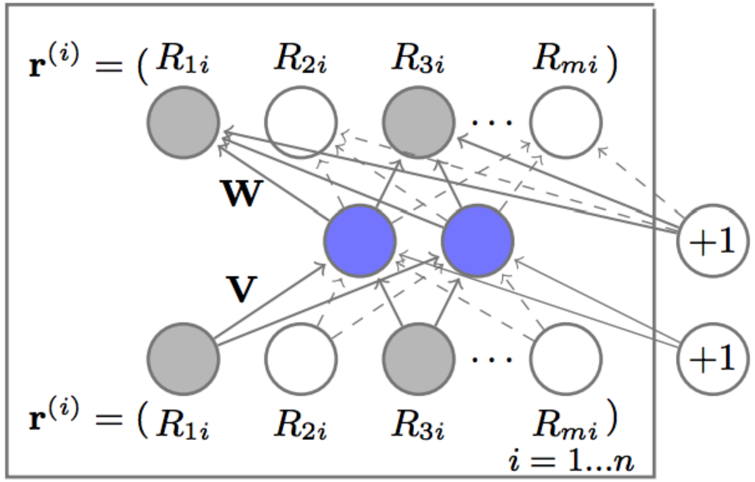


Li et al. "Towards Deep Conversational Recommendations" (NIPS' 18)

• ReDial – Methods – The Autoencoder Recommender

Notations:

- We have |M| users and |V'| movies.
- User-movie Rating Matrix: $\mathbf{R} \in \mathbb{R}^{M \times |V'|}$ Scale: -1 - 1
- A user can be represented by $\mathbf{r}^{(u)} = (\mathbf{R}_{u,1}, \ldots, \mathbf{R}_{u,|V'|})$



AutoRec: Autoencoders Meet Collaborative Filtering (WWW15)

Retrieve the full representation from the lower dimension representation

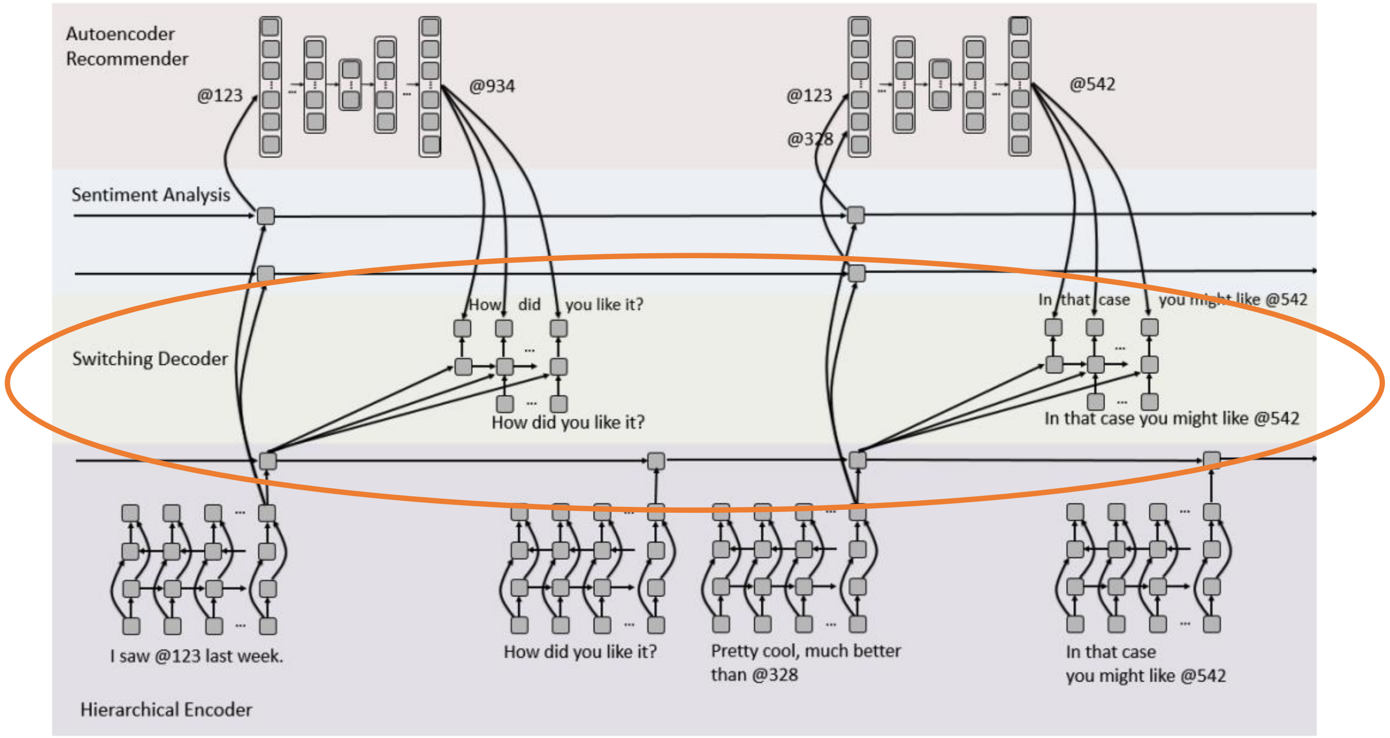
Partially observed user representation fed into a FC layer to lower dimension.

- Then Loss function:

$$L_{\mathbf{R}}(\theta) = \sum_{u=1}^{M} \|\mathbf{r}^{(u)} - h(\mathbf{r}^{(u)}; \theta)\|_{\mathcal{O}}^2 + \lambda \|\theta\|^2$$

Li et al. "Towards Deep Conversational Recommendations" (NIPS' 18)

• ReDial – Methods – Decoder with a Movie Recommendation Switching Mechanism



Responsibility:

- When decoding the next token, decide to mention a movie name, or an ordinary word.

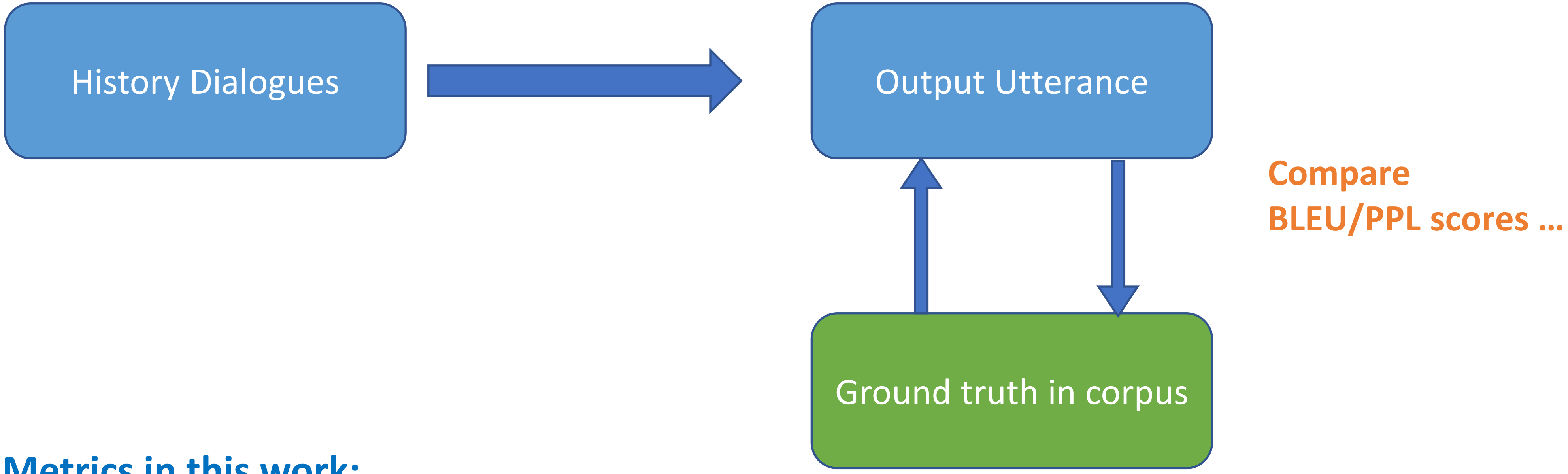
Purpose:

- Such a switching mechanism allows to include an explicit recommendation system in the dialogue agent.

Li et al. "Towards Deep Conversational Recommendations" (NIPS' 18)

• ReDial – Evaluation – Formalization

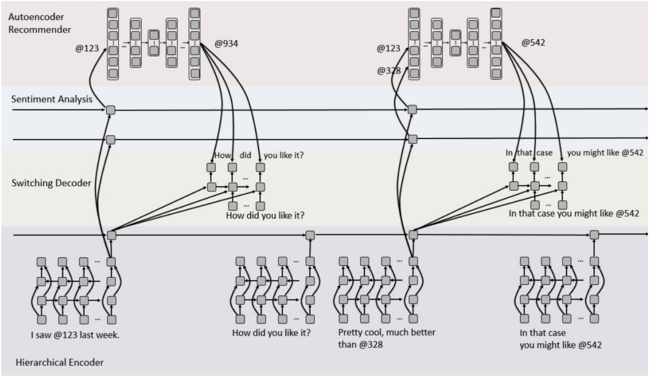
Evaluation settings:
Corpus-based evalution. (Similar to the evaluation in dialogue system)



Evaluation Metrics in this work:
- **Kappa score: Sentiment analysis subtask**
- **RMSE score: Recommendation subtask**
- **Human evaluation: Dialogue generation**

Li et al. "Towards Deep Conversational Recommendations" (NIPS' 18)

• KBRD – Motivation

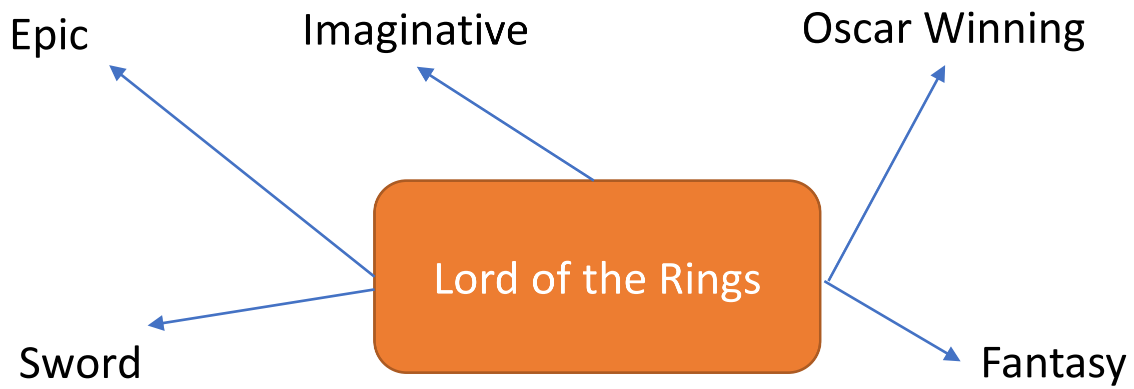


The ReDial (NIPS18) paper has two shortage:

- Only mentioned items are used for recommender system.

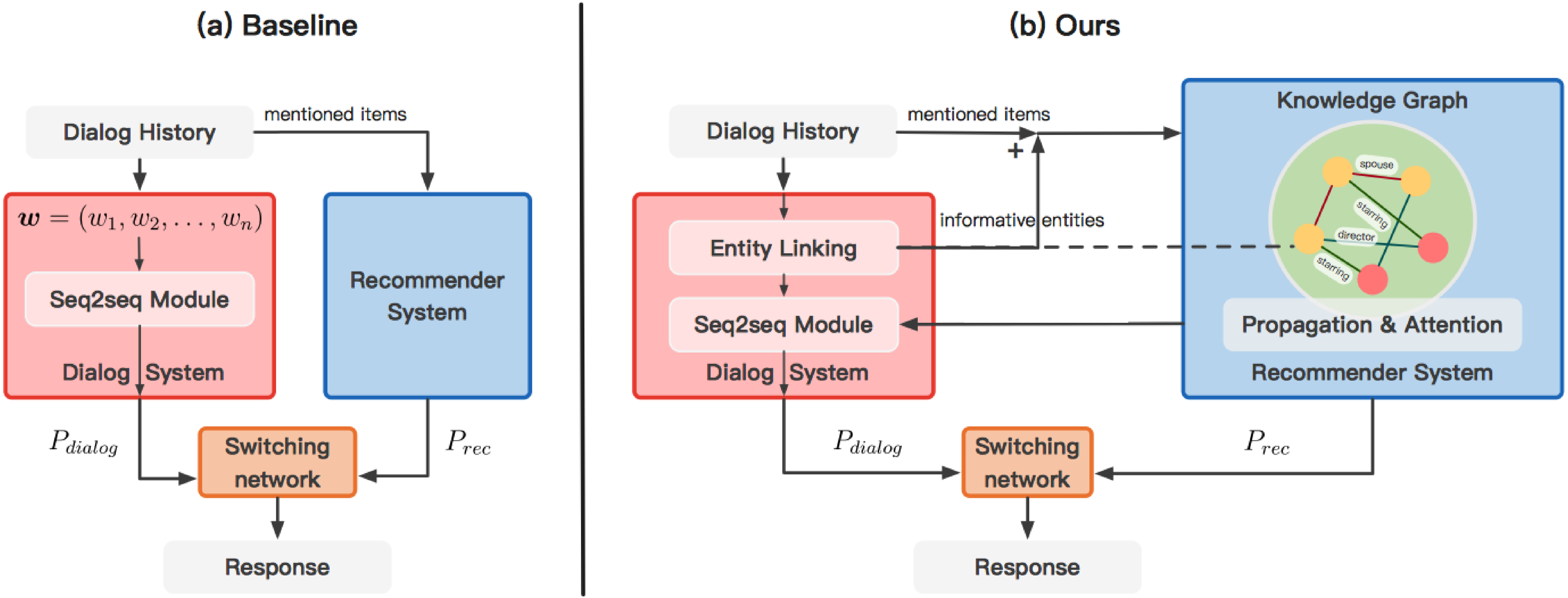
Lord of the Rings is really my all-time-favorite! In fact, I love all **J. R. R. Tolkien**'s work!

- Recommender cannot help generate better dialogue.



Chen et al. "Towards Knowledge-Based Recommender Dialog System" (EMNLP' 19)

• KBRD – Method – Overall



Chen et al. "Towards Knowledge-Based Recommender Dialog System" (EMNLP' 19)

• KBRD – Experiments – Does Recommendation Help Dialog?

Movie	1	2	3	4	5	6	7	8
Star Wars	space	alien	sci-fi	star	sci	robot	smith	harry
The Shining	creepy	stephen	gory	horror	scary	psychological	haunted	thriller
The Avengers (2012)	marvel	superhero	super	dc	wait	batman	thor	take
Beauty and the Beast	cute	disney	animated	live	music	child	robin	kids

Recommendation-Aware Dialog

$$P_{\text{dialog}} = \operatorname{softmax}(Wo + b)$$

$$b_u = \mathcal{F}(t_u)$$

Vocabulary Bias

$$P_{\text{dialog}} = \operatorname{softmax}(Wo + b + b_u)$$

- We select words with Top 8 vocabulary bias. We can see that these words have strong connection with the movie.

Chen et al. "Towards Knowledge-Based Recommender Dialog System" (EMNLP' 19)

• MGCG – Formalization

Goal planning: QA about <Stolen life>, chitchat about Xun Zhou, recommending the movie <The message>, recommending <Don't cry, Nanking!>

1.User: 你知道电影『生死劫』的主演是谁吗? (Who is the star of the movie <stolen life>?)

2.Bot: 是周迅哦。(It is Xun Zhou.)

3.User:是我女神演的呀! (She is my goddess.)

4.Bot: 你真有眼光啊，周迅是中国电视金鹰奖最受欢迎女演员呢。(You have good taste. Zhou Xun is the most popular actress in the Golden Eagle Award of China TV.)

5.User:是呢，我特别喜欢她的，她的演技也很厉害。(I like her very much. She's very good at acting.)

6.Bot:她还获得过亚洲电影大奖最佳女演员的，演技是得到过国际认可的。(She also won the best actress of the Asian Film Awards. Her acting is internationally recognized.)

7.User:嗯嗯，反正她真的很优秀。(Anyway, she's really good.)

8.Bot:那你要看看她演的《风声》吗？该片具有精炼的人物、波谲云诡的情节。很好看哦！(Do you want to see her movie <The message>? It has refined characters and capricious plots.)

9.User:可我现在更喜欢『刘若英』的电影!(But now I prefer Rene Liu's films!)

10.Bot:那刘若英的《南京1937》你可以看看，电影所展现的是一个导演对战争、对民族、对人性的思考。(You can see the movie <Don't cry, Nanking>. It shows the director's thinking on war, nation, and human nature.)

11.User: 历史片吗？(It is historical film?)

12.Bot:历史战争片。(Yes. It is historical war film.)

13.User:那我肯定要看看了。(I will watch it!)

14.Bot:嗯，你肯定会喜欢的。(Sure, you will like it!)

15.User: 嗯，谢谢你呀，下次聊吧! (Thank you! See you later!)

16.Bot:好的，再见！ (OK! Bye!)

DuRecDial Dataset

Recap the settings in NIPS 18:

- Seeker: explain what kind of movie he/she likes, and asks for movie suggestions
- Recommender: understand the seeker's movie tastes, and recommends movies

The dialogue types are very limited!

In this work, 4 types of dialogues:

- Recommendation
- Chitchat
- QA
- Task

Liu et al. "Towards Conversational Recommendation over Multi-Type Dialogues" (ACL' 20)

• MGCG – Formalization -- Dataset Collection

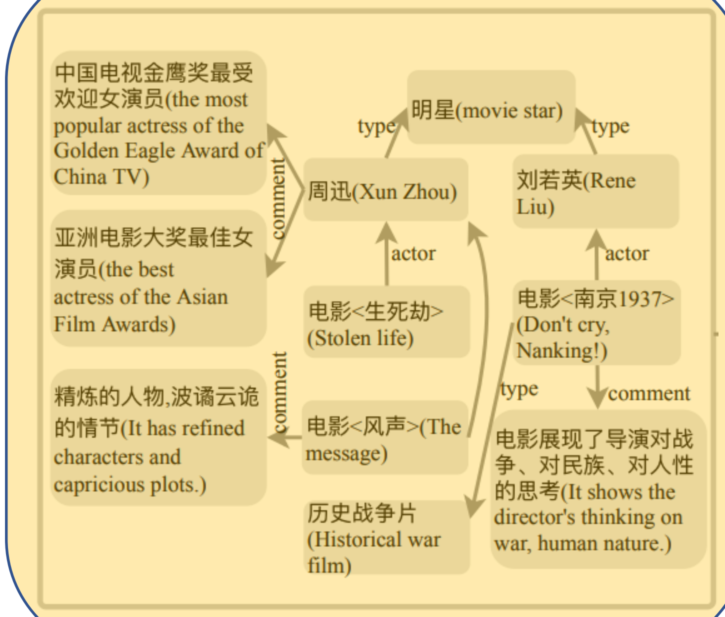
Very similar to the dataset collection process as in NIPS 18: Two workers, one for seeker, one for recommender.
It is further supported by following elements:

Goals	Goal description
Goal1: QA (dialog type) about the movie <Stolen life> (dialog topic)	The seeker takes the initiative, and asks for the information about the movie <Stolen life>; the recommender replies according to the given knowledge graph; finally the seeker provides feedback.
Goal2: chitchat about the movie star Xun Zhou	The recommender proactively changes the topic to movie star Xun Zhou as a short-term goal, and conducts an in-depth conversation;
Goal3: Recommendation of the movie <The message>	The recommender proactively changes the topic from movie star to related movie <The message>, and recommend it with movie comments, and the seeker changes the topic to Rene Liu's movies;
Goal4: Recommendation of the movie <Don't cry, Nanking!>	The recommender proactively recommends Rene Liu's movie <Don't cry, Nanking!> with movie comments. The seeker tries to ask questions about this movie, and the recommender should reply with related knowledge. Finally the user accepts the recommended movie.

Name: 杨凡宇(Fanyu Yang)
Gender: 男(Male)
Age: 20
Domains that the user likes: movie, music
Stars that the user likes: 周迅(Xun Zhou), 刘若英(Rene Liu)
Recommendation accepted: <生死劫>(Stolen Life)
Recommendation rejected: <小王子>(The little prince)

Explicit Seeker Profile
- For the consistency

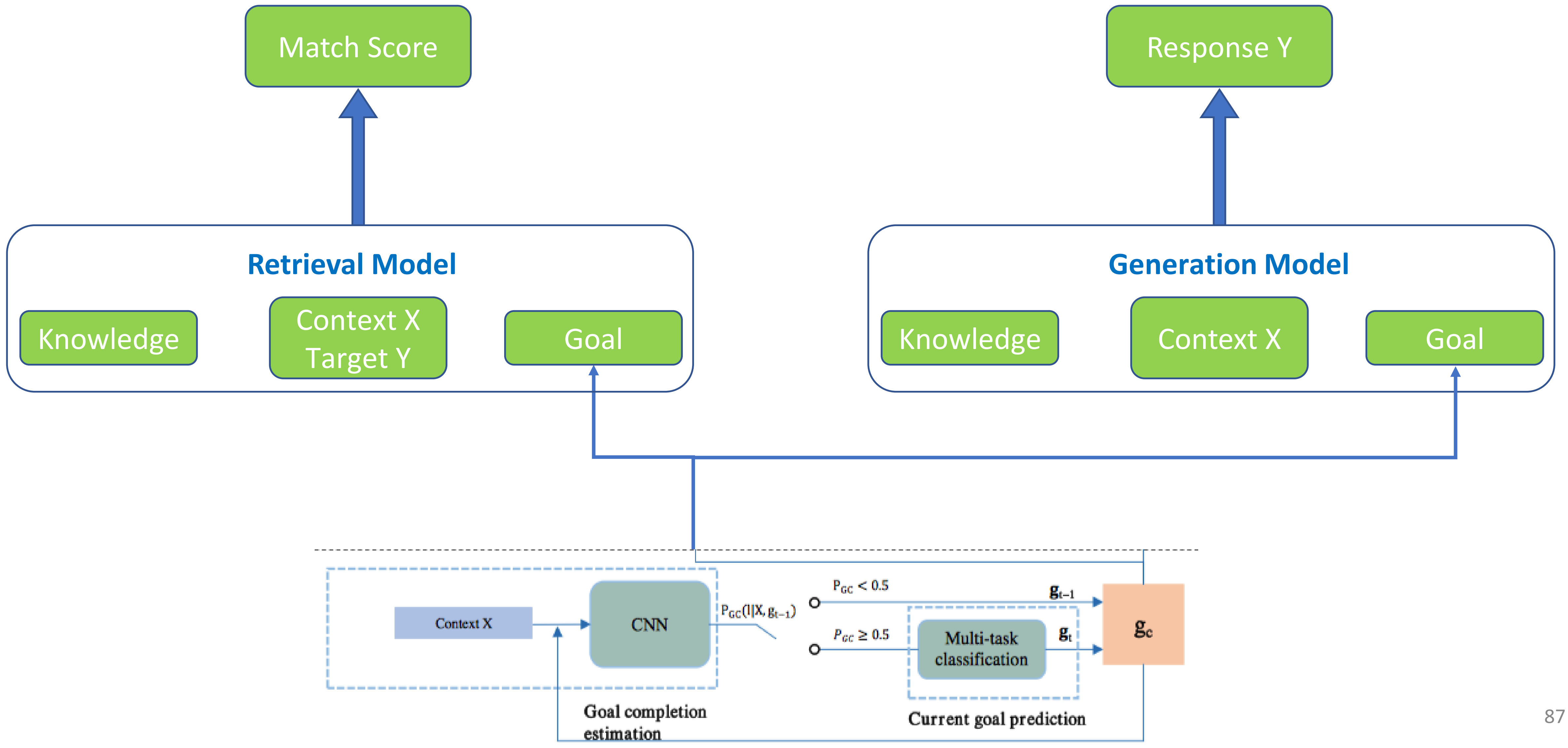
Task Template
- Constrain the complicated task



Knowledge Graph:
- Further assist the workers

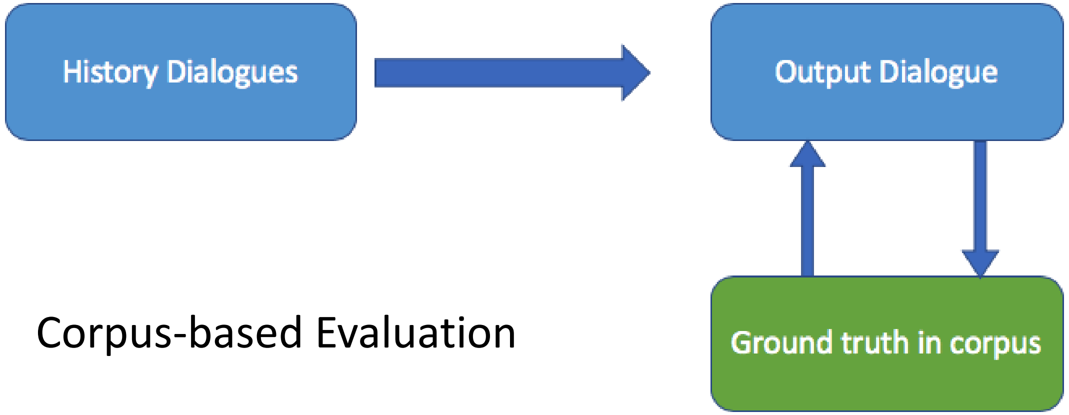
Liu et al. "Towards Conversational Recommendation over Multi-Type Dialogues" (ACL' 20)

• MGCG – Methods



Liu et al. "Towards Conversational Recommendation over Multi-Type Dialogues" (ACL' 20)

• MGCG – Evaluation – Setting



Corpus-based Evaluation

Evaluation Metrics:

Dialogue generation:

- BLEU – Relevence
- Perplexity – Fluency
- DIST – Diversity
- Hits@1/3 -- Retrieval model (1 ground truth, 9 randomly sampled.)

Humam Evaluation:

- Turn level: fluency, appropriateness, informativeness, and proactivity.
- Dialogue level: Goal success rate and Coherence

Liu et al. "Towards Conversational Recommendation over Multi-Type Dialogues" (ACL' 20)

• KMD – Motivation and Formalization

Motivation:

Existing dialogue systems only utilize textual information, which is not enough for full understanding of the dialogue.

- What is "these"?
- What is "it"?

Background: Fashion Match!

User utterance $u_1, u_2, \cdots, u_t$

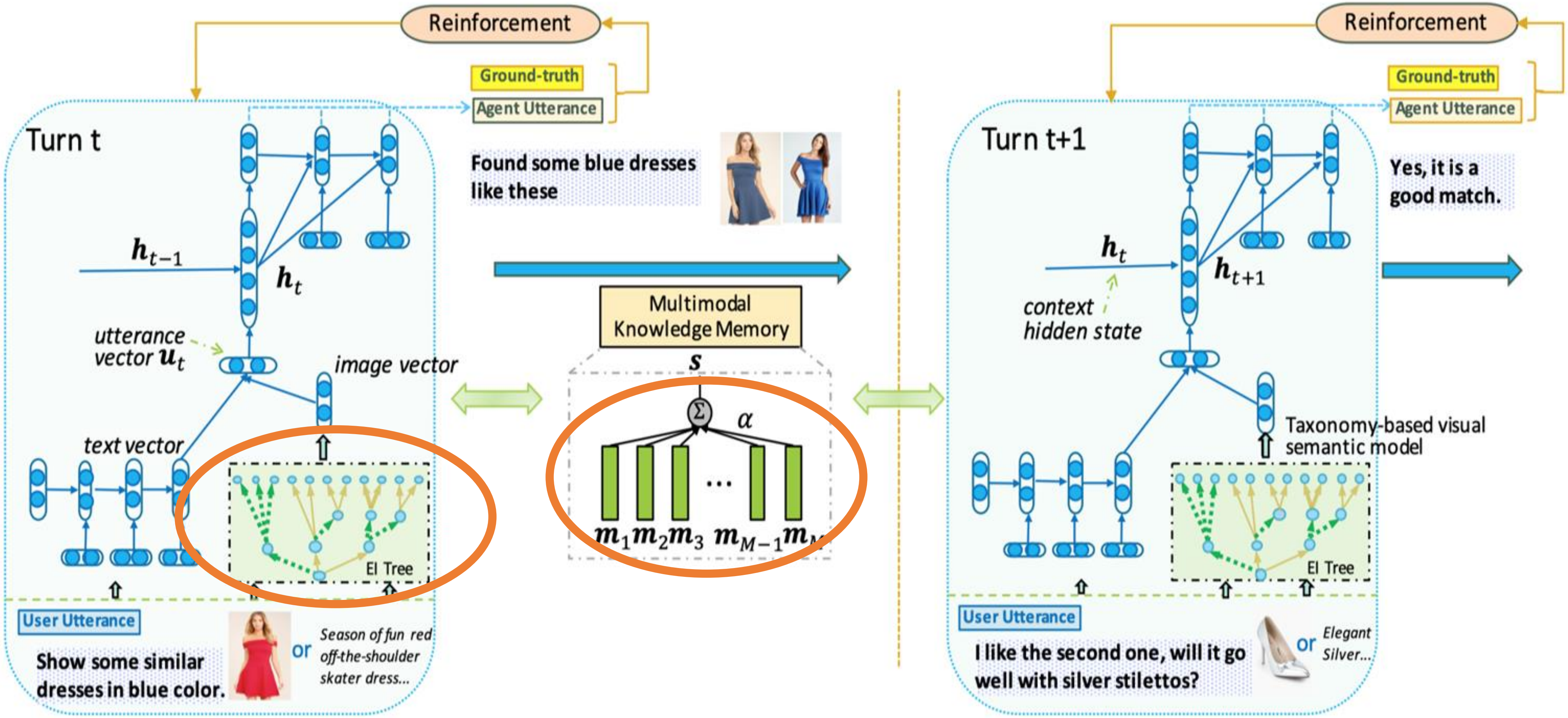
Agent utterance $\hat{u}_1, \hat{u}_2, \cdots, \hat{u}_{t-1}$

$\longrightarrow \hat{u}_t.$

u be both Text and Image modality

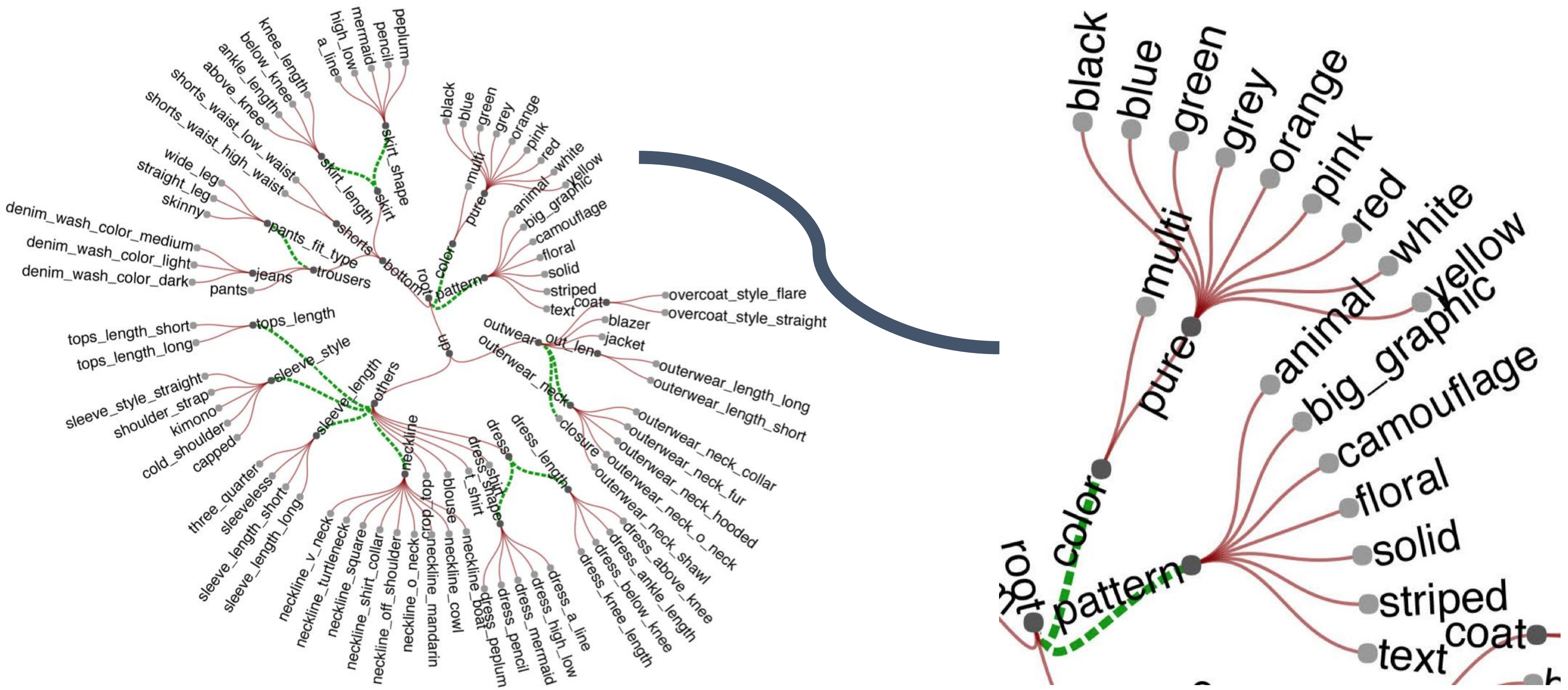
Liao et al. "Knowledge-aware Multimodal Dialogue Systems" (MM 20)

• KMD – Method – Overview



Liao et al. "Knowledge-aware Multimodal Dialogue Systems" (MM 20)

• KMD – Method – Exclusive & Inclusive Tree (EI Tree)



Instead of CNN to capture image feature, they used taxonomy-based feature. They argued that CNN only captures generic features, but they want to capture the rich domain knowledge in specific domain.

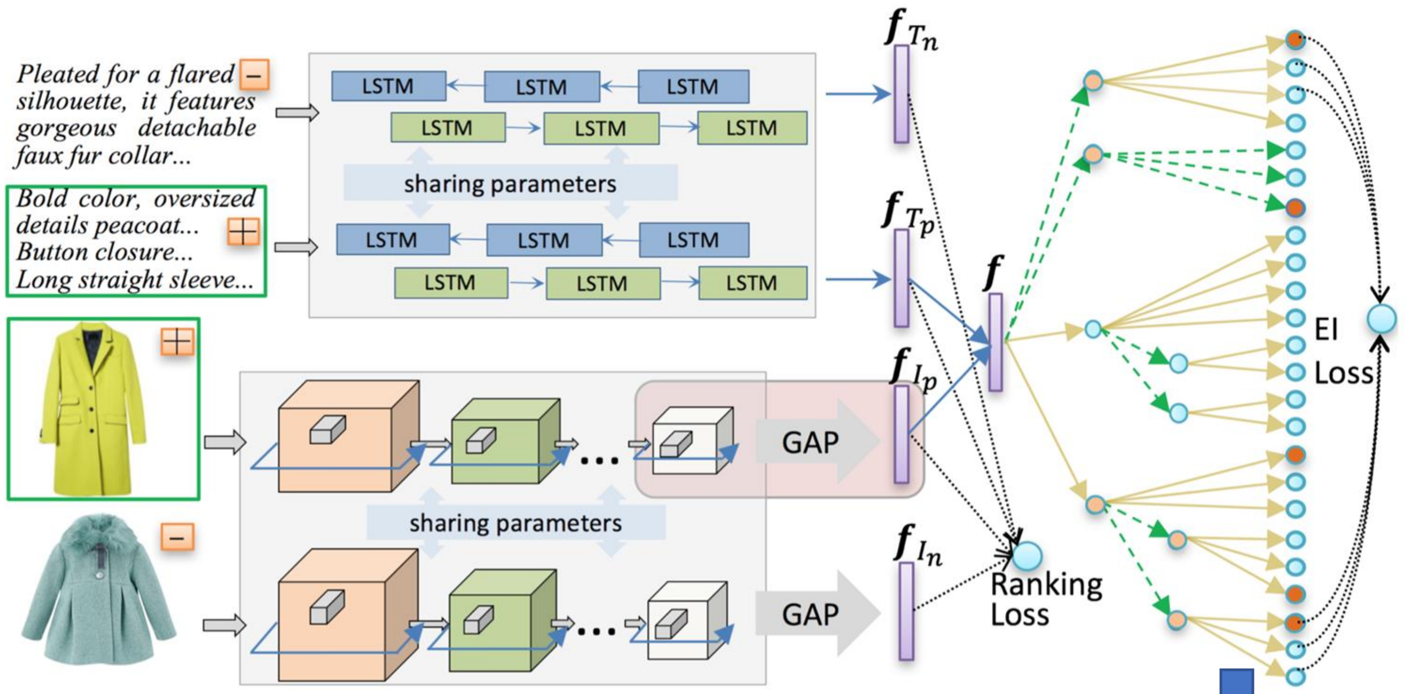
Liao et al. "Knowledge-aware Multimodal Dialogue Systems" (MM 20)

• KMD – Method – EI Tree

Encode text features

Encode image features

A sequence of steps along the path.



$$p(c_n|\, c_0 \rightarrow c_n, \mathbf{f}, \mathbf{W}_{EI}) = p(c_1|\, c_0, \mathbf{f}, \mathbf{W}_{EI}) \cdot p(c_2|\, c_1, \mathbf{f}, \mathbf{W}_{EI}) \cdots p(c_n|\, c_{n-1}, \mathbf{f}, \mathbf{W}_{EI}),$$

Optimization:

- **EI Loss**: Compare the predicted leaf node against ground truth, and optimize the cross entropy loss.
- **Pairwise ranking loss** is used to regularize the model to match text and image feature.

Liao et al. "Knowledge-aware Multimodal Dialogue Systems" (MM 20)

• KMD – Method – Incorporation of Domain Knowledge

Fashion Tips: if the user asks for advice about matching tips of **NUS hoodie**, the matching candidates such as the **Livi's jeans** might not co-occur with it in the whole training corpus or conversation history.

Liao et al. "Knowledge-aware Multimodal Dialogue Systems" (MM 20)

• KMD – Method – Incorporation of Domain Knowledge

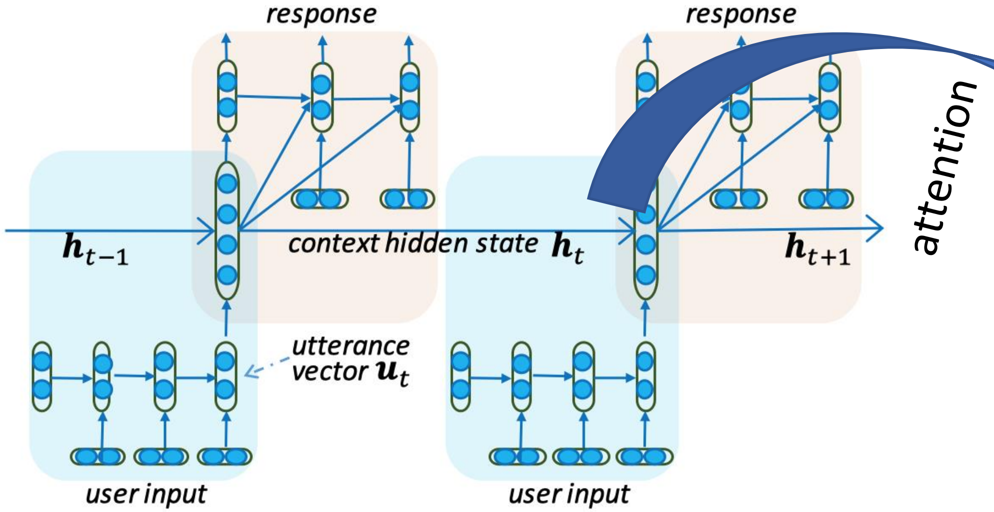
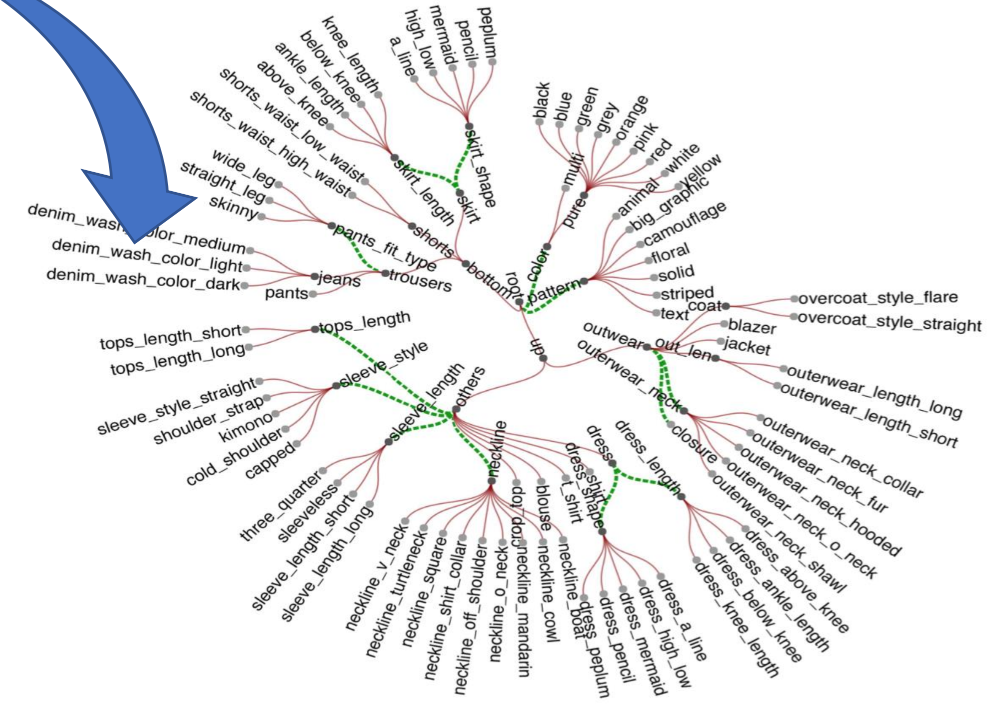


Figure 5: An illustration of text-based HRED backbone.

Each EI tree leaf gets a memory vector: the averaging of the image representation corresponds to the leaf node



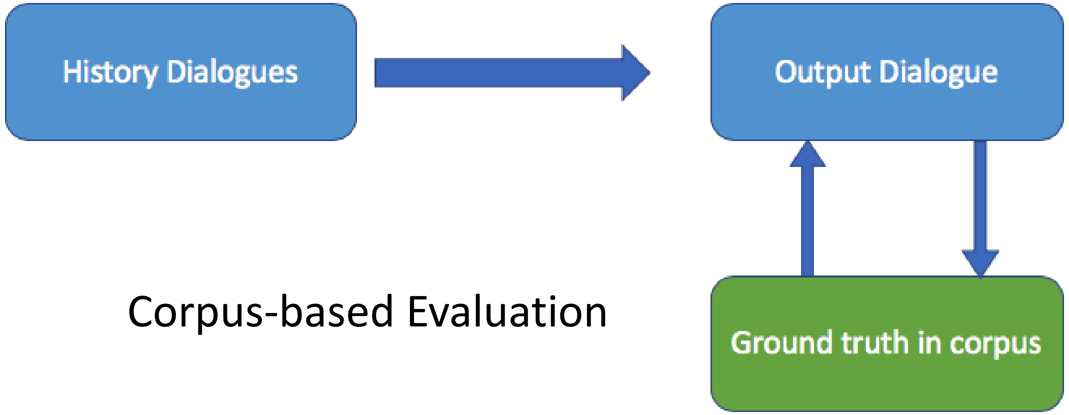
They incorporated knowledge into HRED model (hierarchical recurrent encoder-decoder)

$$\mathbf{h}'_t = \mathbf{h}_t + \mathbf{s}.$$

S is the weighted sum of the memory vector

Liao et al. "Knowledge-aware Multimodal Dialogue Systems" (MM 20)

• KMD – Evaluation – Formalization



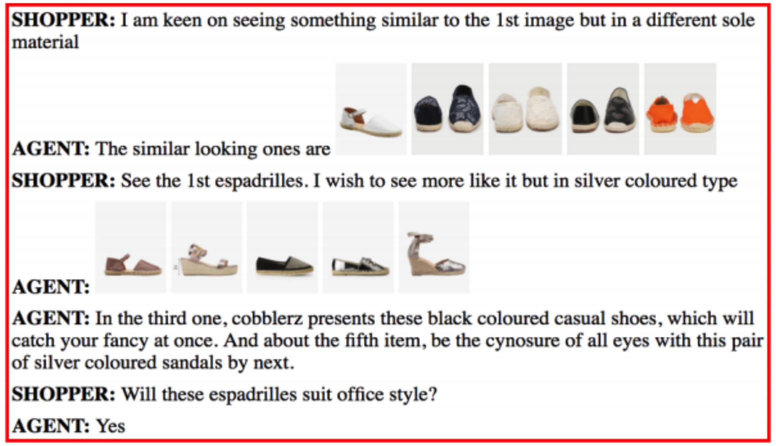
Evaluation Metrics:

Text generation:

- **BLEU Score**
- **Diversity (unigram)**

Image response generation:

- **Recall @ K**



SHOPPER: I am keen on seeing something similar to the 1st image but in a different sole material

AGENT: The similar looking ones are

SHOPPER: See the 1st espadrilles. I wish to see more like it but in silver coloured type

AGENT:

AGENT: In the third one, cobblerz presents these black coloured casual shoes, which will catch your fancy at once. And about the fifth item, be the cynosure of all eyes with this pair of silver coloured sandals by next.

SHOPPER: Will these espadrilles suit office style?

AGENT: Yes

Liao et al. "Knowledge-aware Multimodal Dialogue Systems" (MM 20)

Towards Building Large Scale Multimodal Domain-Aware Conversation Systems (AAAI 18) MMD Dataset

• Tutorial Outline

- ❑A Glimpse of Dialogue System
- ❑Four research directions in conversational recommendation system
  - ❑Question Driven Approaches
  - ❑Multi-turn Conversational Recommendation Strategy
  - ❑Dialogue Understanding and Generation
  - ❑**Exploitation-Exploration Trade-offs for Cold Users**
- ❑ Summary of Formalizations and Evaluations

• Bandit algorithms for Exploitation-Exploration trade-off

Exploration (Learning)		Exploitation (Earning)
✔ **Take some risk to collect information about unknown options**	Trade-off	✔ **Takes advantage of the best option that is known.**

Multi-armed bandit example: which arm to select next?

#(Successes) / #(Trials))

Arm 1	Arm 2	Arm 3	Arm 4	
2/5	0/1	3/8	1/3	• • •

Common intuitive ideas:

- **Greedy:** trivial exploit-only strategy
- **Random:** trivial explore-only strategy
- **Epsilon-Greedy:** combining Greedy and Random.
- **Max-Variance**: only exploring w.r.t. uncertainty.

• Upper Confidence Bounds (UCB) - Method

Arm selection strategy:

Exploitation Exploration

$$\hat{a} = \arg\max_a \boxed{\hat{Q}(a)} + \boxed{\Delta(a)}$$

$\boxed{\hat{Q}(a)} = \frac{1}{N}\sum_{t=1}^{N_a} r_{t,a}$: The estimated mean of reward of arm a.

$\boxed{\Delta(a)}$: The uncertainty of $\hat{Q}(a)$.

Estimating rewards by averaging the observed rewards:

#(Successes) / #(Trials)

Arm 1	Arm 2	Arm 3	Arm 4	
$\hat{Q}(a) = 2/5$	$\hat{Q}(a) = 0/1$	$\hat{Q}(a) = 3/8$	$\hat{Q}(a) = 1/3$	...

$$\hat{a} = \arg\max_a \boxed{\hat{Q}(a)} + \boxed{\alpha\sqrt{\frac{\log T}{t_a}}}$$

• A Contextual-Bandit Approach with Linear Reward (LinUCB) - Method

Estimating reward by introducing the feature vector $\mathbf{X}_{t,a}^{\mathrm{T}}$: $\hat{Q} = \mathbf{X}_{t,a}^{\mathrm{T}}\boldsymbol{\theta}_a$

#(Successes) / #(Trials)

Arm 1	Arm 2	Arm 3	Arm 4	
$\mathbf{X}_{t,a}^{\mathrm{T}}\boldsymbol{\theta}_a =$ 2/5	$\mathbf{X}_{t,a}^{\mathrm{T}}\boldsymbol{\theta}_a =$ 0/1	$\mathbf{X}_{t,a}^{\mathrm{T}}\boldsymbol{\theta}_a =$ 3/8	$\mathbf{X}_{t,a}^{\mathrm{T}}\boldsymbol{\theta}_a =$ 1/3	…

The arm selection strategy is:

Exploitation Exploration

$$a_t \overset{\text{def}}{=} \arg\max_a \left(\boxed{\mathbf{X}_{t,a}^{\mathrm{T}}\boldsymbol{\theta}_a} + \boxed{\alpha\sqrt{\mathbf{X}_{t,a}^{\mathrm{T}}\mathbf{A}_a^{-1}\mathbf{X}_a}} \right)$$

where $\mathbf{A}_a \overset{\text{def}}{=} \mathbf{D}_a^{\mathrm{T}}\mathbf{D}_a + \mathbf{I}_a$

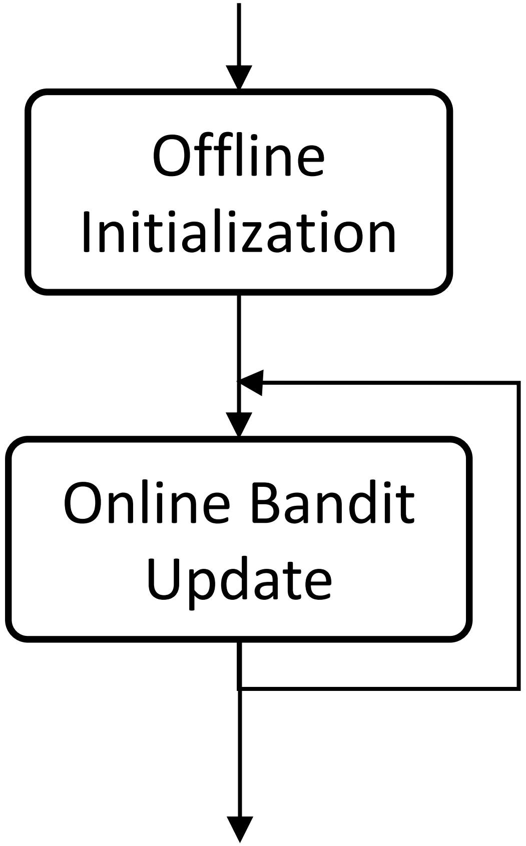
$\alpha = 1 + \sqrt{\ln(2/\delta)/2}$

Li et al. "A Contextual-Bandit Approach to Personalized News Article Recommendation " (WWW' 10)

• Bandit algorithm in Conversational Recommendation System - Formalization

Setting:

- For cold start users, the user embedding is initialized as the average embedding of existing users.
- Asking only whether a user likes items (no attributes questions).
- The model updates its parameters at each turn.



only ask about Items!

Christakopoulou et al. "Towards Conversational Recommender Systems" (KDD' 16)

• Bandit algorithm in Conversational Recommendation System - Method

Method:

Traditional recommendation model + bandit model

• Terminology: trait=embedding

Absolute Model. First, let us assume that we have observed tuples of the form (user i, item j, 1/0).[4] The model estimates the *affinity* of user i to item j based on the biases and traits. The generative procedure is:

1. User i has traits $\mathbf{u}_i \sim \mathcal{N}(\mathbf{0}, \sigma_1^2\mathbf{I})$, bias $\alpha_i \sim \mathcal{N}(0, \sigma_2^2)$.
2. Item j has traits $\mathbf{v}_j \sim \mathcal{N}(\mathbf{0}, \sigma_1^2\mathbf{I})$, bias $\beta_j \sim \mathcal{N}(0, \sigma_2^2)$.
3. (a) The (unobserved) affinity is

$$y_{ij} = \alpha_i + \beta_j + \mathbf{u}_i^T \mathbf{v}_j. \quad (1)$$

Observations are modeled as the noisy estimate $\hat{y}_{ij} \sim \mathcal{N}(y_{ij}, \epsilon_{ij})$, where ϵ_{ij} models the affinity variance, accounting for noise in user preferences. This yields an observation of whether the user likes an item ($\hat{r}_{ij}$):

$$\hat{r}_{ij} = \mathbf{1}[\hat{y}_{ij} > 0]. \quad (2)$$

Traditional MF-based recommendation model

+

Greedy: $j^* = \arg\max_j y_{ij}$
A trivial *exploit*-only strategy: Select the item with highest estimated affinity mean.

Random: $j^* = \text{random}(1,N)$
A trivial *explore*-only strategy.

Maximum Variance (MV): $j^* = \arg\max_j \epsilon_{ij}$
A *explore*-only strategy, variance reduction strategy: Select the item with the highest noisy affinity variance.

Maximum Item Trait (MaxT): $j^* = \arg\max_j \|\mathbf{v}_j\|_2$
Select the item whose trait vector $\mathbf{v}_j$ contains the most information, namely has highest L2 norm $\|\mathbf{v}_j\|_2 = \sqrt{v_{j1}^2 + v_{j2}^2 + \ldots + v_{jd}^2}$.

Minimum Item Trait (MinT): $j^* = \arg\min_j \|\mathbf{v}_j\|_2$
Select the item with trait vector with least information.

Upper Confidence (UCB): $j^* = \arg\max_j y_{ij} + \epsilon_{ij}$
Based on UCB1 [3]: Pick the item with the highest upper confidence bound, namely mean plus variance (95% CI)

Thompson Sampling (TS) [5]: $j^* = \arg\max_j \hat{y}_{ij}$
For each item, sample the noisy affinity from the posterior. Select item with the maximum sampled value.

Common bandit strategies

Christakopoulou et al. "Towards Conversational Recommender Systems" (KDD' 16)

• Bandit algorithm in Conversational Recommendation System - Evaluation

Setting: Offline initialization + Online updating

- Online stage: Ask 15 questions of 10 items. Each question is followed by a recommendation.
- Metric: Average precision *AP@10,* which is a widely used recommendation metric.

Real data: collected from restaurant searching logs

- Offline learning on collected **3549 users 289 restaurants, and 9330 positive observations.**
- Recruit **28 users** to rate on the **selected 10 restaurants**.
- Online cold-start user study: **each one of the 28 users rates 10 carefully selected restaurants, based on which his/her preference u_i is inferred. Then, run 50 times:**
    1. Sample a user i.
    2. Sample $\hat{u}_i \sim u_i$.
    3. Use $\hat{u}_i$ to simulate reward of each restaurant.

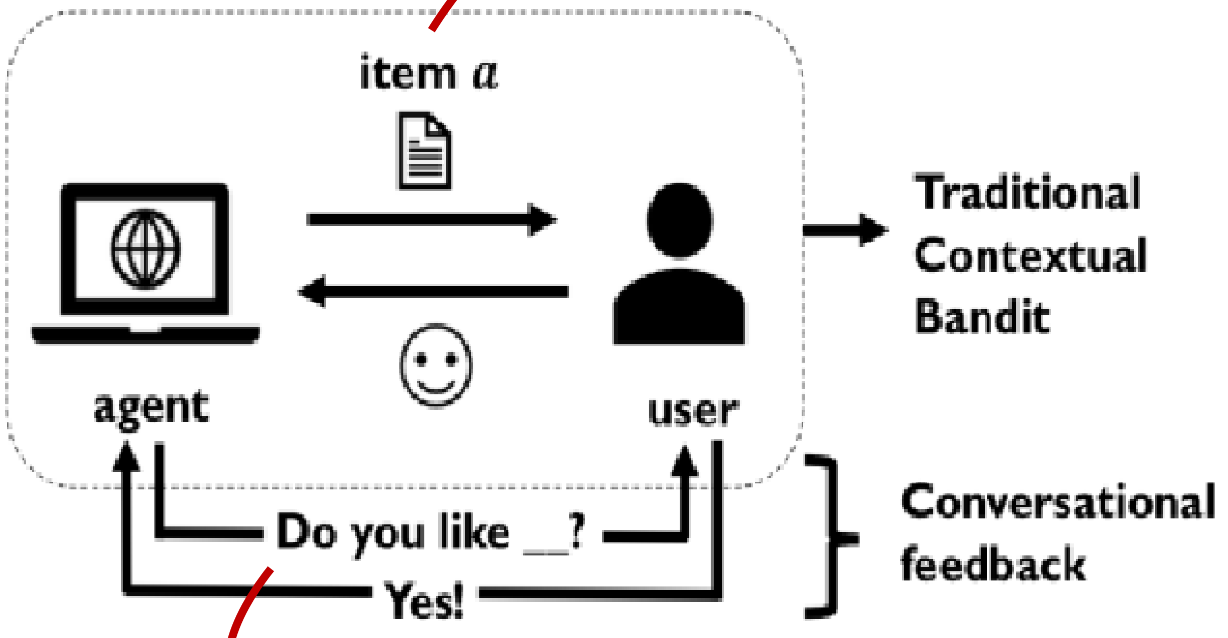
Christakopoulou et al. "Towards Conversational Recommender Systems" (KDD' 16)

• Conversational UCB algorithm(ConUCB) - Formalization

Setting:

- Asking questions about not only the **bandit arms (items),** but also the **key-terms (categories, topics).**
- **One key-term is related to a subset of arms. Users' preference on key-terms can propagate to arms.**
- **Each arm has its own features.**

Select an arm to recommend



Select one or more key-terms to query or not

Zhang et al. "Conversational Contextual Bandit: Algorithm and Application" (WWW' 20)

• ConUCB - Method -- Overview

Select **attributes (key-terms)** to query

Select **an item (arm)** to recommend

Algorithm 1: General algorithm of ConUCB

Input: arms $\mathcal{A}$, key-terms $\mathcal{K}$, graph $(\mathcal{A}, \mathcal{K}, W)$, $b(t)$.

for $t = 1, \ldots, T$ **do**

  observe contextual vector $\boldsymbol{x}_{a,t}$ of each arm $a \in \mathcal{A}_t$;

  If conversation is allowed at round t, i.e., $q(t) = 1$, select key-terms to conduct conversations and receive conversational feedbacks $\{\tilde{r}_{k,t}\}$;

  select an arm $a_t = \arg\max_{a \in \mathcal{A}_t} \boxed{\tilde{R}_{a,t}} + \boxed{C_{a,t}}$;

  receive a reward $r_{a_t,t}$;

  update model ;

Exploitation Exploration

Zhang et al. “Conversational Contextual Bandit: Algorithm and Application” (WWW’ 20)

• ConUCB - Method

When to query the key-terms:

- Define a function $b(t)$, which determines:
  (1) whether to converse at round t.
  (2) the number of conversations until round t.

- Consider the function $q(t)$:

$$q(t) = \begin{cases} 1, & b(t) - b(t-1) > 0, \\ 0, & \text{otherwise.} \end{cases}$$

- **If $q(t) = 1$, query about $b(t) - b(t-1)$ key-terms;**
- **If $q(t) = 0$, does not query about a key-term;**

- For users' experience, key-term-level conversations should be less frequent than arm-level interactions, i.e., $b(t) \le t, \forall t$.

Examples:

1) The agent makes k conversations in every m rounds.

$$b(t) = k \left\lfloor \frac{t}{m} \right\rfloor, m \ge 1, k \ge 1,$$

1) The agent makes a conversation with a frequency represented by the logarithmic function of t.

$$b(t) = \lfloor \log(t) \rfloor$$

1) There is no conversation between the agent and the user.

$$b(t) \equiv 0$$

Zhang et al. "Conversational Contextual Bandit: Algorithm and Application" (WWW' 20)

• ConUCB - Method

The core strategy to select arms and key-terms:

- **Selecting the arm** with the largest upper confidence bound derived from both arm-level and key-term-level feedback, and receives a reward.

User preference computed on key-term-level rewards

$$\tilde{\boldsymbol{\theta}}_t = \arg\min_{\tilde{\boldsymbol{\theta}}} \sum_{\tau=1}^{t} \sum_{k \in \mathcal{K}_\tau} \left(\frac{\sum_{a \in \mathcal{A}} w_{a,k} \tilde{\boldsymbol{\theta}}^T \boldsymbol{x}_{a,\tau}}{\sum_{a \in \mathcal{A}} w_{a,k}} - \tilde{r}_{k,\tau} \right)^2 + \tilde{\lambda} \|\tilde{\boldsymbol{\theta}}\|_2^2,$$

User preference computed on arm-level rewards

$$\boldsymbol{\theta}_t = \arg\min_{\boldsymbol{\theta}} \ \lambda \sum_{\tau=1}^{t-1} (\boldsymbol{\theta}^T \boldsymbol{x}_{a_\tau,\tau} - r_{a_\tau,\tau})^2 + (1-\lambda) \|\boldsymbol{\theta} - \tilde{\boldsymbol{\theta}}_t\|_2^2.$$

Constrain $\boldsymbol{\theta}$ to be close to $\tilde{\boldsymbol{\theta}}$

• ConUCB - Method

The strategy of arm selection is

$$a_t = \arg\max_{a \in \mathcal{A}_t} \underbrace{x_{a,t}^T \boldsymbol{\theta}_t}_{\tilde{R}_{a,t}} + \underbrace{\lambda \alpha_t \left\|\boldsymbol{x}_{a,t}\right\|_{\boldsymbol{M}_t^{-1}} + (1-\lambda)\tilde{\alpha}_t \left\|\boldsymbol{x}_{a,t}^T \boldsymbol{M}_t^{-1}\right\|_{\tilde{\boldsymbol{M}}_t^{-1}}}_{C_{a,t}}$$

Exploitation

Exploration

where M_t is the function of $\boldsymbol{\theta}$ and $\tilde{\theta}$

111

The core strategy to select arms and key-terms:

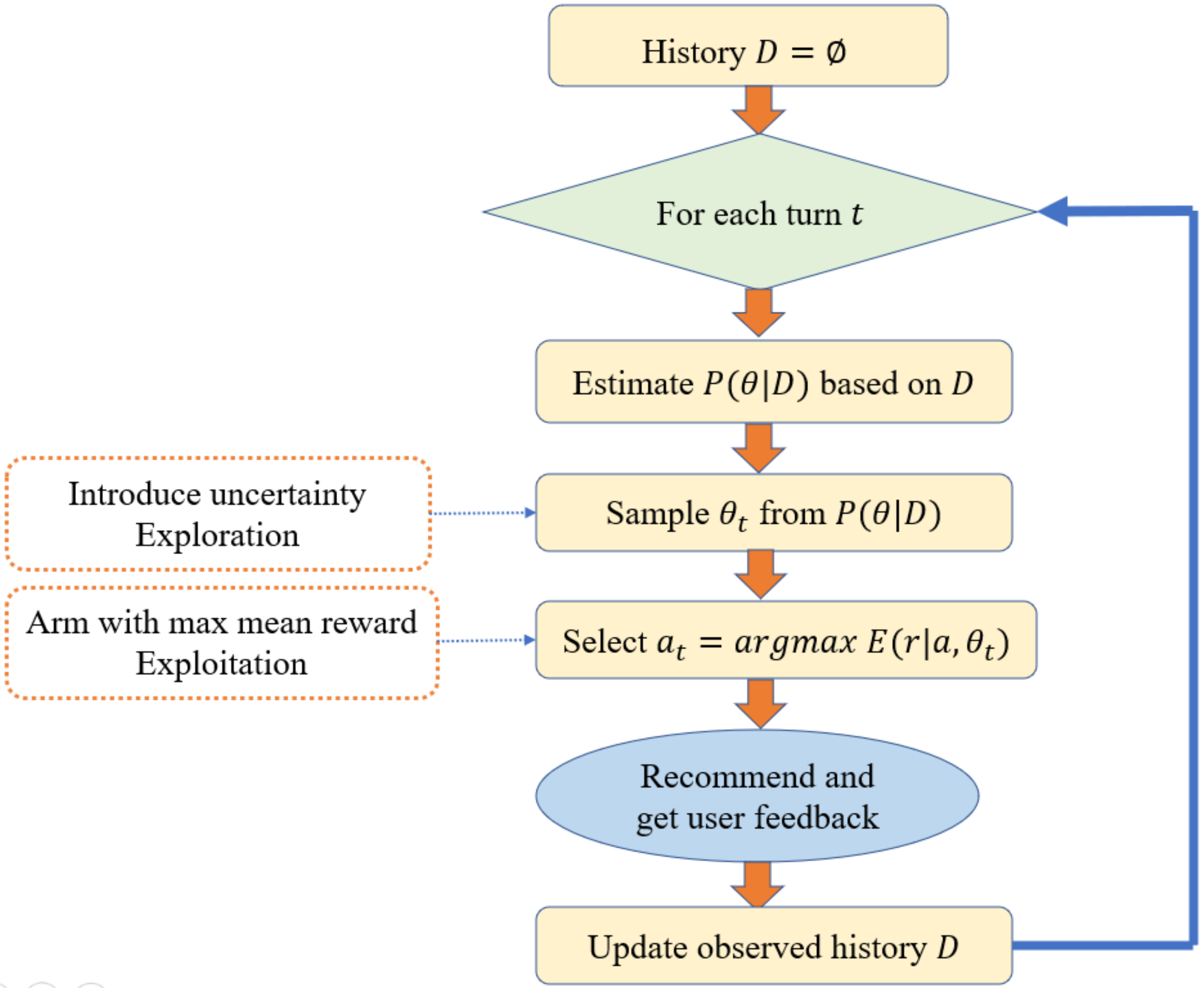
- **Selecting the key-terms** that maximum the reward of the corresponding items.

$$k = \arg\max_{k'} \left\| \boldsymbol{X}_t \boldsymbol{M}_t^{-1} \tilde{\boldsymbol{M}}_{t-1}^{-1} \tilde{\boldsymbol{x}}_{k',t} \right\|_2^2 \Big/ \left(1 + \tilde{\boldsymbol{x}}_{k',t}^T \tilde{\boldsymbol{M}}_{t-1}^{-1} \tilde{\boldsymbol{x}}_{k',t}\right)$$

$$\text{where } \tilde{\mathbf{x}}_{k,t} = \sum_{a \in \mathcal{A}} \frac{w_{a,k}}{\sum_{a' \in \mathcal{A}} w_{a',k}} \mathbf{x}_{a,t}.$$

Zhang et al. "Conversational Contextual Bandit: Algorithm and Application" (WWW' 20)

- **Thompson Sampling**
  - **Bayesian bandit problem**: instead of modeling the probability of reward as a scalar, Thompson Sampling assumes the user preference comes from a distribution



•Contextual Thompson Sampling

- Assume that user preference comes from a **multidimensional Gaussian distribution**.

Arm selection strategy:

$$\hat{a} = \arg max_a \boldsymbol{x}_{\boldsymbol{a}}^{\boldsymbol{T}} \boldsymbol{\theta}_{\boldsymbol{u}}$$

$\boldsymbol{\theta}_{\boldsymbol{u}}$ denotes user preference. In each turn, it is sampled from a Gaussian distribution:

$$\mathcal{N}(\boldsymbol{\mu}_u, l^2 \mathbf{B}_u^{-1})$$

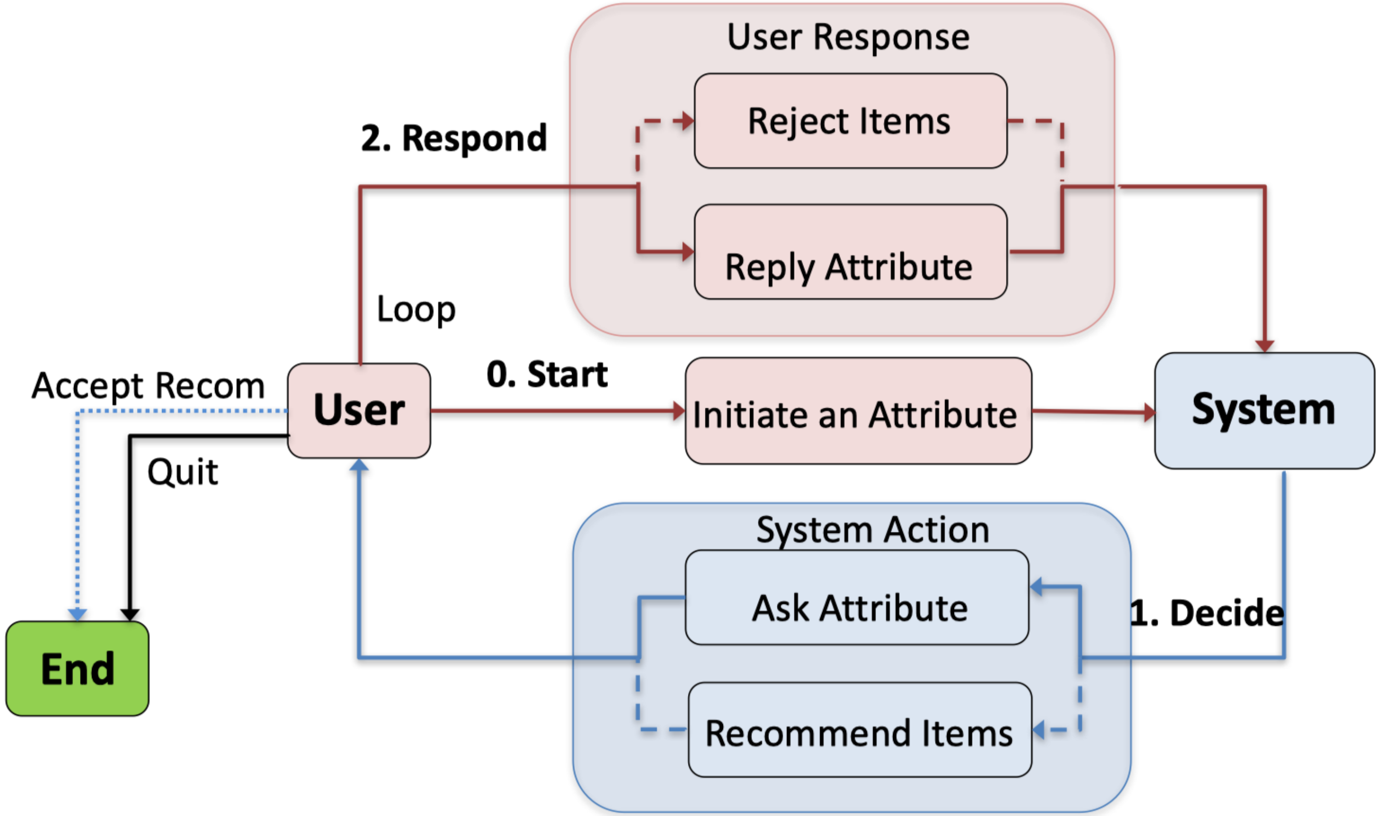
exploitation

exploration

• Revisit Multi-Round Conversational Recommendation Scenario

This time, we focus on cold-start users

Objective:
Recommend desired items to user in shortest turns

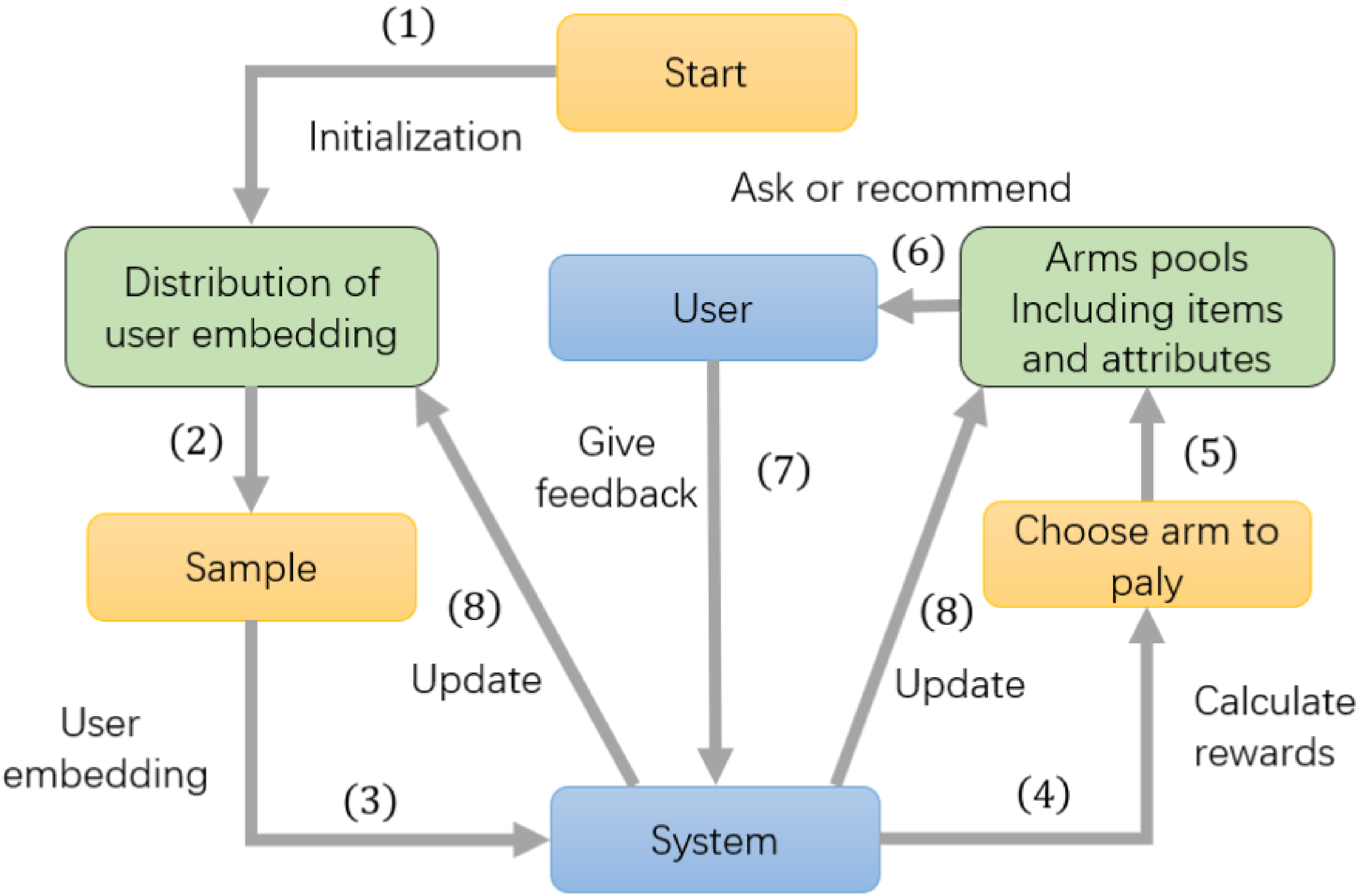


Lei et al. "Estimation–Action–Reflection: Towards Deep Interaction Between Conversational and Recommender Systems" (WSDM'20)

• ConTS (Conversational Thompson Sampling) -- Workflow

Treat items and attributes as indiscriminate arms.

Make theoretical customization for contextual TS to adapt to cold-start users in conversational recommendation.



Li et al. Seamlessly Unifying Attributes and Items: Conversational Recommendation for Cold-Start Users (arxiv' 20)

• ConTS -- Method -- Arm Choosing

$$\mathbb{E}[r(a,u,\mathcal{P}_u)] = \mathbf{u}^{\mathsf{T}}\mathbf{x}_a + \sum_{p_i \in \mathcal{P}_u} \mathbf{x}_a{}^{T}\mathbf{p}_i,$$

Arm Choosing: It is very simple, selecting the arm with highest reward.

Indiscriminate arms for items and attributes:

- If the arm with highest reward is attribute: system asks.
- If the arm with highest reward is item: system recommends top K items.

We addresses the strategy for recommendation issue by our indiscriminate designs of arms.

Li et al. Seamlessly Unifying Attributes and Items: Conversational Recommendation for Cold-Start Users (arxiv' 20)

• ConTS -- Method -- Update

$$\mathbb{E}[r(a,u,\mathcal{P}_u)] = \mathbf{u}^\top \mathbf{x}_a + \sum_{p_i \in \mathcal{P}_u} \mathbf{x}_a{}^T \mathbf{p}_i,$$

Update of Arm Pool: $\mathcal{P}_u$

- If user rejects an item / attribute: remove them from arm pool.
- If user likes an attribute: append it to the known attribute set for better estimation and narrow down the candidate item pool accordingly.

Update parameters of : $\mathcal{N}(\boldsymbol{\mu}_u, l^2 \mathbf{B}_u^{-1})$

$$\mathbf{B}_u = \mathbf{B}_u + \mathbf{x}_{a(t)} \mathbf{x}_{a(t)}{}^T$$
$$r'_a = r_a - \mathbf{x}_{a(t)}{}^T (\mathbf{u}_{init} + \textstyle\sum_{p_i \in \mathcal{P}_u} \mathbf{p}_i)$$
$$\boldsymbol{f}_u = \boldsymbol{f}_u + r'_a * \mathbf{x}_{a(t)}$$
$$\boldsymbol{\mu}_u = \mathbf{B}_u^{-1} \boldsymbol{f}_u$$

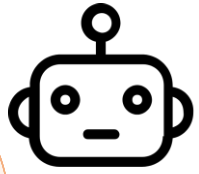
The known preferred attributes are used to estimate reward of arms as well as narrow down the candidate item pool.

Li et al. Seamlessly Unifying Attributes and Items: Conversational Recommendation for Cold-Start Users (arxiv' 20)

• ConTS -- Evaluation -- User Simulator

User ID: 333, Item ID: 666

Item Name: “*Small Italy Restaurant*”
Item Attributes: [Pizza, Nightlife, Wine, Jazz]



I'd like some **Italian** food.

Got you, do you like some **pizza**?

Yes!

Got you, do you like some **nightlife**?

Yes!

Do you want “Small Paris”?

Check, I don’t want “Small Paris”

Rejected!

Got you, do you like some **Rock Music**?

No!

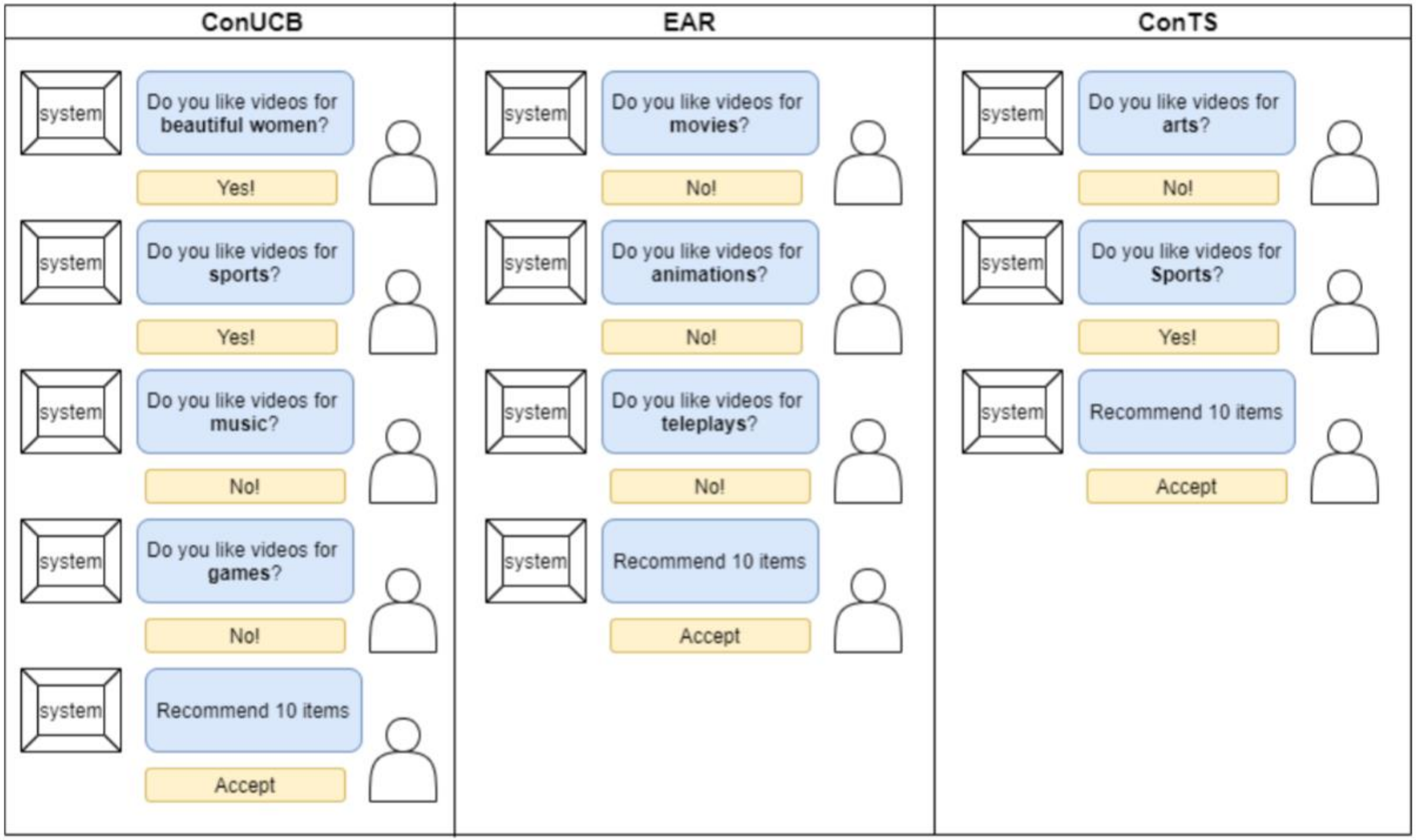
Check, I don’t want “Rock Music”

Do you want “Small Italy Restaurant”?

Accepted!

Template-based utterances

• ConTS -- Evaluation-- Case Study on Kuaishou



ConTS unifies items and attributes and keeps EE balance.

Li et al. Seamlessly Unifying Attributes and Items: Conversational Recommendation for Cold-Start Users (arxiv' 20)

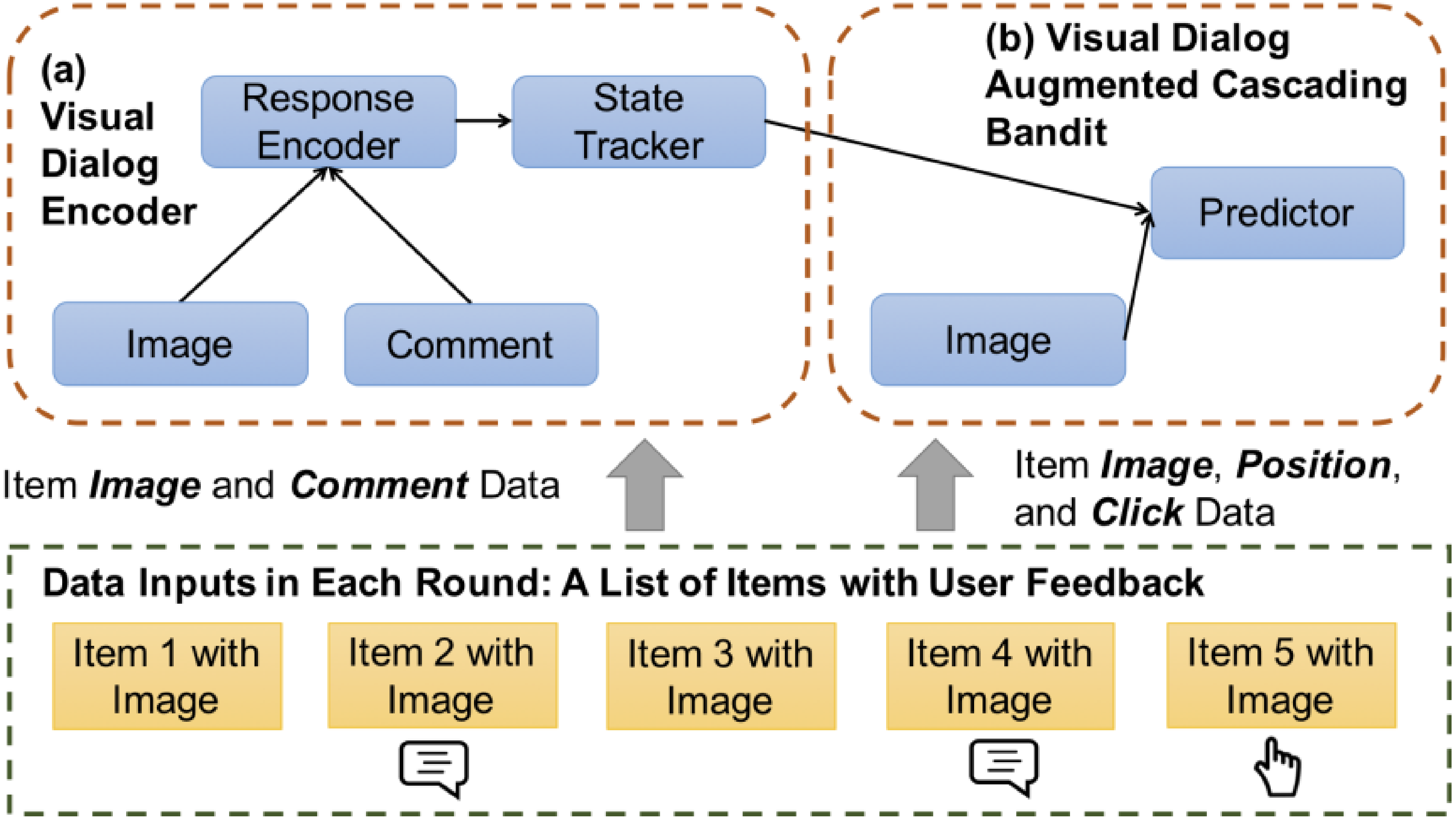
• VDA IRS -- Formalization

A Visual Dialog Augmented Interactive Recommender System

Yu et al. (KDD' 19)

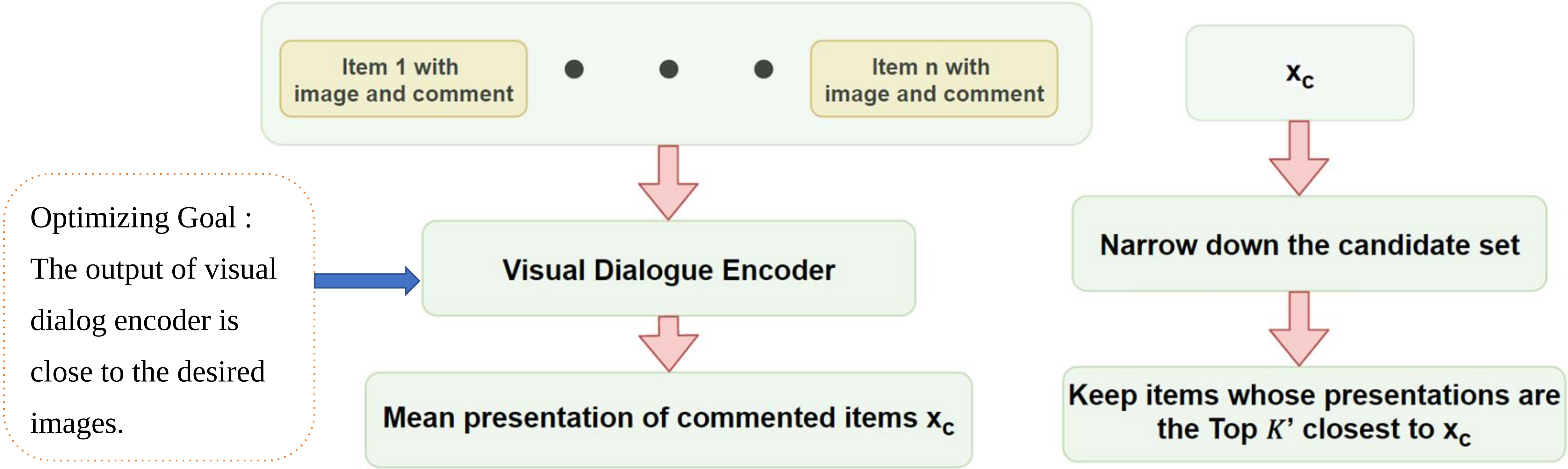
Yu et al. A Visual Dialog Augmented Interactive Recommender System (KDD' 19)

• VDA IRS -- Workflow



Yu et al. A Visual Dialog Augmented Interactive Recommender System (KDD' 19)

• VDA IRS -- Method -- Visual Dialog Encoder



The comments and images are encoded to help elicit the user preferences and narrow down the candidate set.

Yu et al. A Visual Dialog Augmented Interactive Recommender System (KDD' 19)

• VDA IRS --Method--Visual Dialog Augmented Cascading Bandit

forall $t = 1, \cdots, n$ **do**

Sample the parameter θ from its posterior

$\theta_t \sim \mathcal{N}(\bar{\theta}_t, S_t)$

forall $k = 1, \cdots, K$ **do**

$a_k^t \leftarrow \arg\max_{e \in \mathcal{L} - \{a_1^t, \cdots, a_{k-1}^t\}} x_e^\top \theta_t$

end

$A_t \leftarrow (a_1^t, \cdots, a_k^t)$

Observe click $C_t \in \{1, \cdots, K, \infty\}$

Each turn, the model will recommend a list of the Top K items.

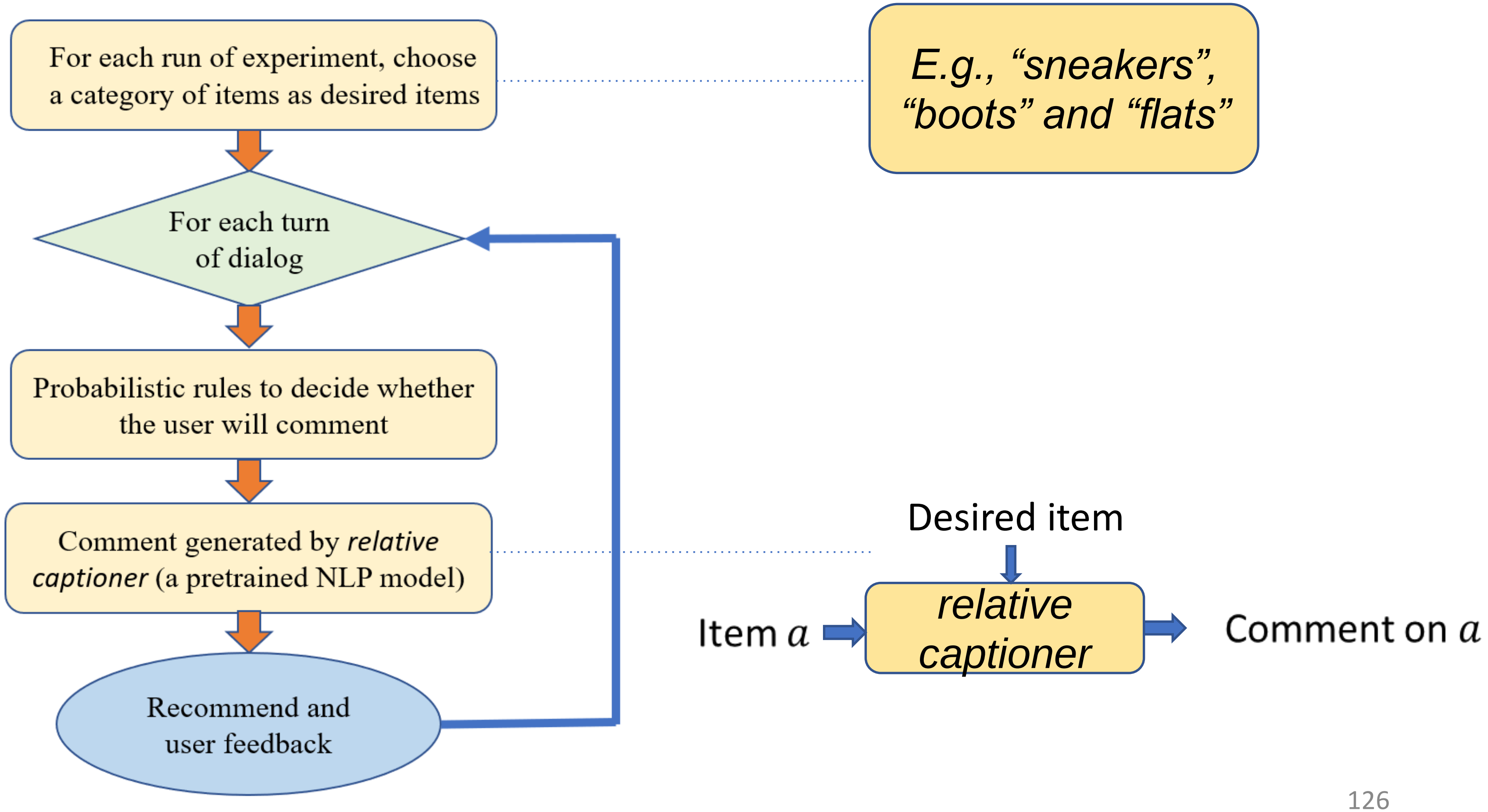
Yu et al. A Visual Dialog Augmented Interactive Recommender System (KDD' 19)

• VDA IRS -- Evaluation

Dataset:

- A footwear dataset where 10,000 images for offline training the visual dialog encoder and 4,658 images for evaluating different interactive recommenders.

User simulator:



Yu et al. A Visual Dialog Augmented Interactive Recommender System (KDD’ 19)

• Strategies in the conversational recommendation bandit (ConUCB)

Evaluation setting for real data:

- **How to simulate users' ground-truth rewards on unobserved arms?**

1. Use interactions of test set as known rewards $r_{a,t}$
2. Given users' feature $x_{a,t}$ on an arm a.
3. Estimate users' preferences θ using ridge regression:

$$\theta = \arg\min_{\theta} \sum_{t=1}^{|T_a|} (x_{a,t}^T \theta - r_{a,t})^2 + ||\theta||^2$$

4. Simulate the ground-truth reward on unobserved arm and key terms of by this estimated θ .

Zhang et al. "Conversational Contextual Bandit: Algorithm and Application" (WWW' 20)

• Tutorial Outline

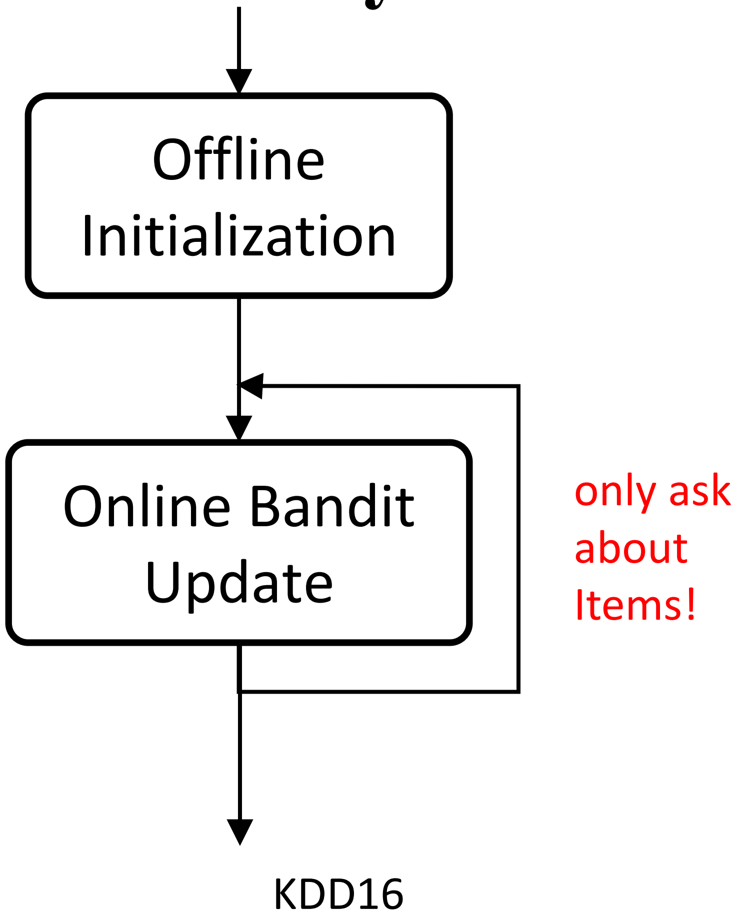
- ❑A Glimpse of Dialogue System
- ❑Four research directions in conversational recommendation system
  - ❑Question Driven Approaches
  - ❑Multi-turn Conversational Recommendation Strategy
  - ❑Dialogue Understanding and Generation
  - ❑Exploitation-Exploration Trade-offs for Cold Users
- ❑ **Summary of Formalizations and Evaluations**

• Summary – Formalization

Mainstream settings for CRS:

- Only consult on items.
- Ask 1 turn, recommend 1 turn.
- Ask X turn, recommend 1 turn (X is predefined).
- Ask X turn, recommend 1 turn (The system need to decide X).
- Ask X turn, recommend X turn.
- Natural Language Understanding and Generation.

• Summary – Formalization – Only Consulting on Items



- The system only consult users on their preference on items.
- Cannot leverage on the advantage of explicitly consulting on attributes.

Liao et al. "Knowledge-aware Multimodal Dialogue Systems" (MM 20)

• Summary – Formalization – Ask 1 Turn, Recommend 1 Turn

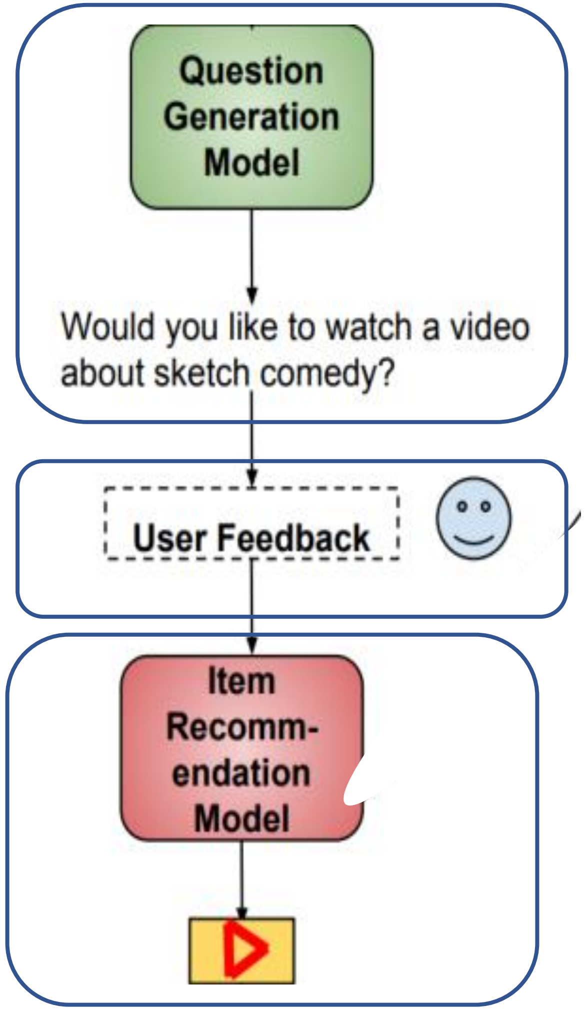


Figure 1: High-level *Q&R* overview.

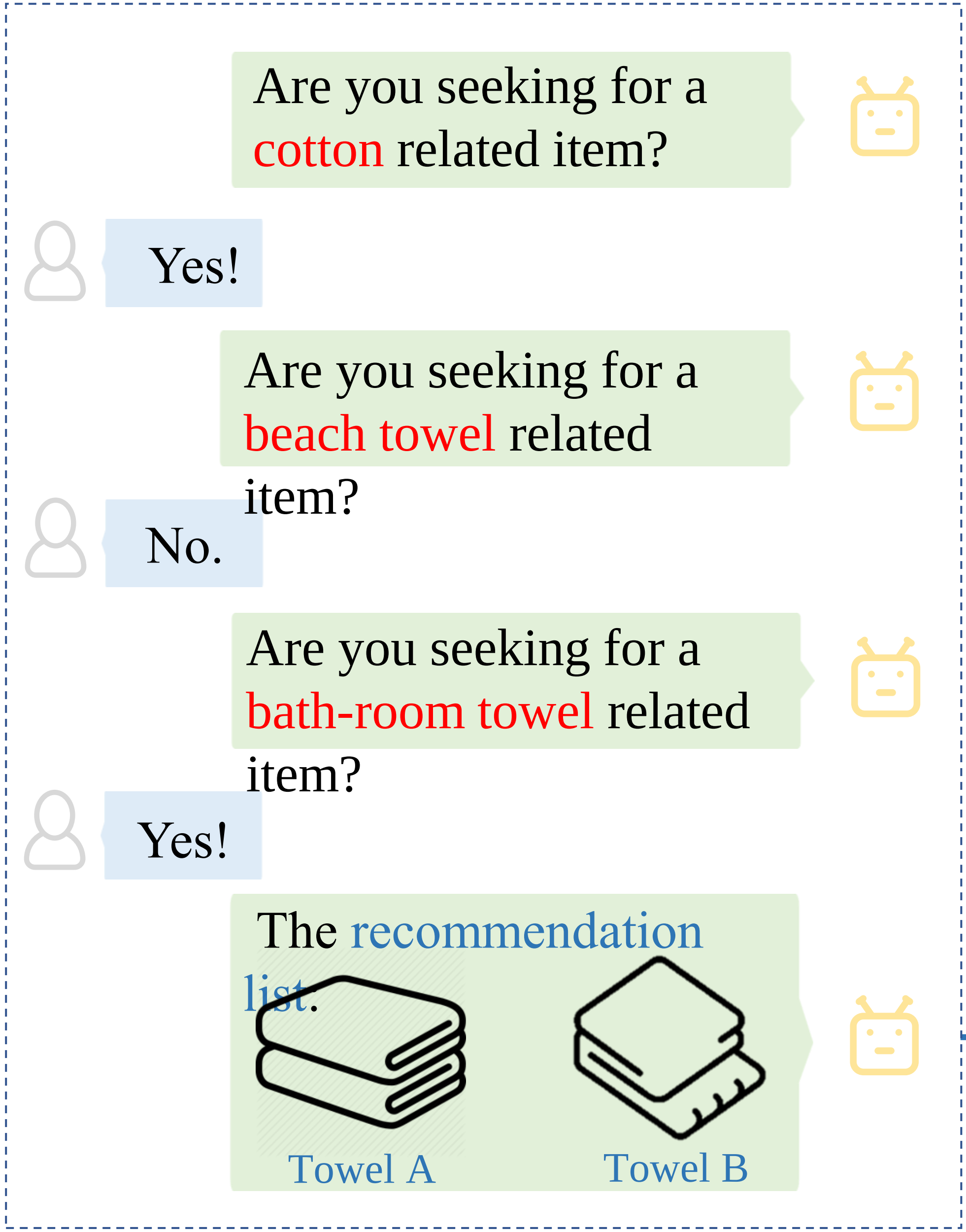
Christakopoulou et al. "Q&R: A Two-Stage Approach toward Interactive Recommendation"(KDD' 18)

- The session will end regardless the recommendation successes or not.

- The session will continue till the recommendation successes.

Yu et al. A Visual Dialog Augmented Interactive Recommender System (KDD' 19)

• Summary – Formalization – Ask X Turns, Recommend 1 Turn



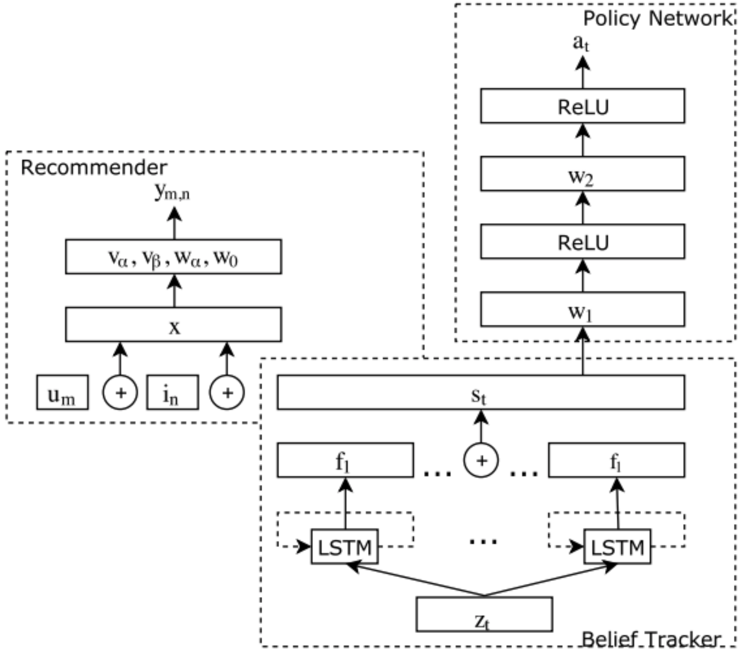
The number of questions $X = 3$ is a hyper-parameter which is specified.

- Ask K question and then recommend one batch of items. (**X is pre-defined**)
- Do not take long-term strategy into account.

Making Recommendation only once N_q questions

Zou et al. "Towards Question-based Recommender Systems"(SIGIR' 20)

• Summary – Formalization – Ask X Turn, Recommend 1 turn



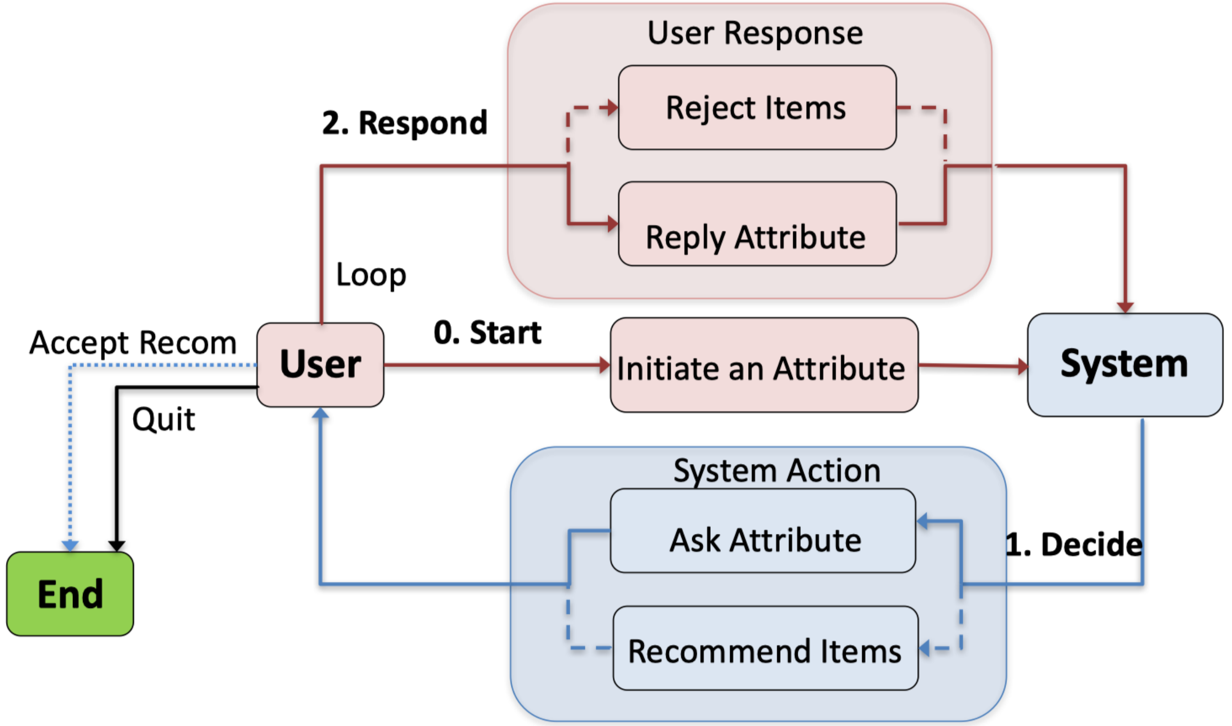
Zhang et al. Conversational Recommender System SIGIR18

-Ask X question and then recommend one batch of items. (**X is decided by model**)

-The session will end regardless the recommendation succeeds or not.

-Only consider strategy in a shallow way (e.g. after asking 3, 4 or 5 question, should I recommend?)

• Summary – Formalization – Ask X turn, Recommend X turn



Lei et al. "Estimation–Action–Reflection: Towards Deep Interaction Between Conversational and Recommender Systems" (WSDM'20)

- Ask X question and then recommend one batch of items.
- The session **will go on** even it the recommendation is not successful!

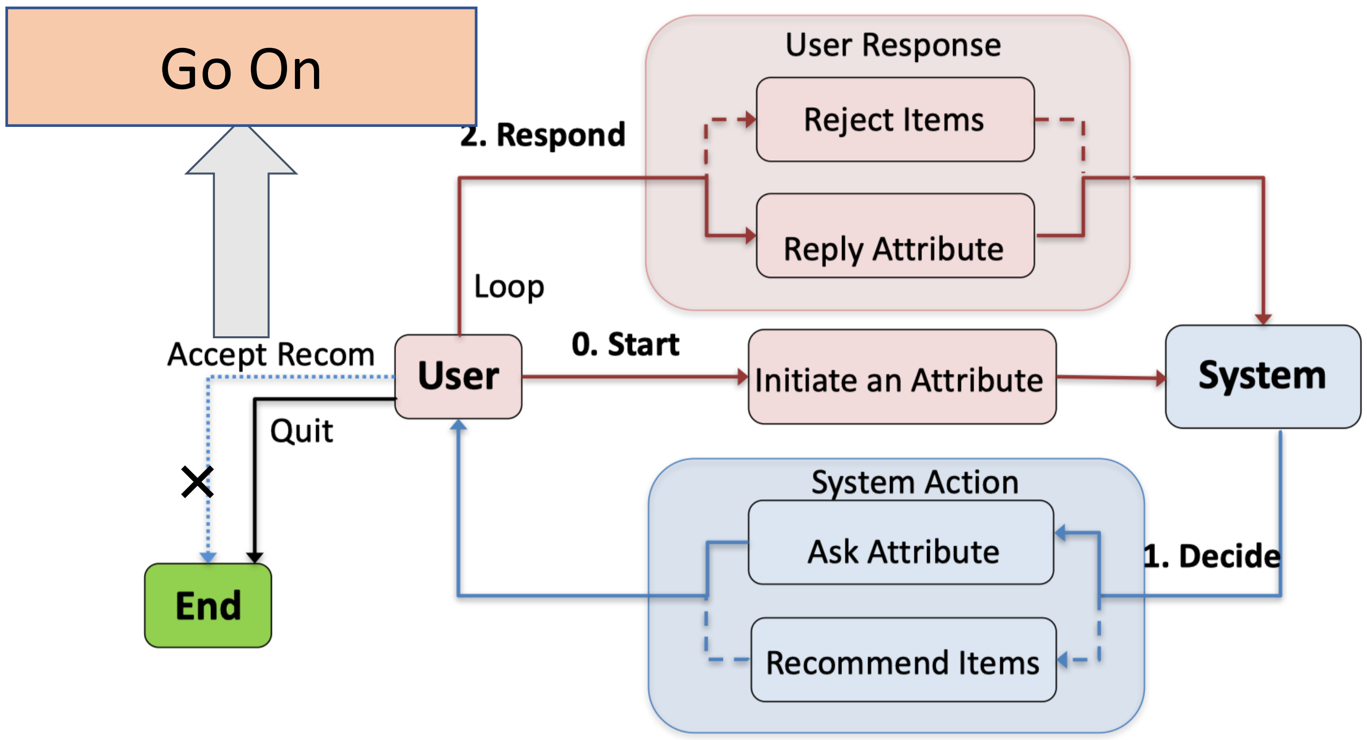
• Summary – Formalization – Natural Language Understanding and Generation

HUMAN:	hello, can i help you find a movie ?
HRED:	hi what kind of movies do you like
OURS:	hi, how can i help you ?
SEEKER:	hello ! i would like to find a suspenseful, but clean family friendly movie .
...	...
SEEKER:	2001 : a space odyssey might be a great option. anything else that you would suggest ?
HUMAN:	you can try planet of the apes the older one is quite suspenseful and family friendly .
HRED:	i haven't seen that one yet but i heard it was good. i haven't seen that one. have you seen the last house on the left ?
OURS:	star wars : the force awakens is also a good one return of the jedi all good movies
SEEKER:	those sound good ! i 'm going to look into those movies.
HUMAN:	i hope you enjoy, have a nice one
HRED:	have you seen foxcatcher ? it 's about a man who has a rich guy.
OURS:	i hope i was able to help you find a good movie to watch
SEEKER:	thank you for your help ! have a great night ! good bye

- This is more likely to be a special type of dialogue system. More popular in NLP community.

Li et.al. "Towards Deep Conversational Recommendations" (NIPS'18)

• Summary – Formalization – Future Directions



The session will go on even if the recommendation is successful.

- Maximize Profit
- Increase the time users stay

• Summary – User Preference Simulation

Mainstream approaches to simulate user preference:

- **User click history:** EAR (WSDM20), CPR (KDD20), CRM(SIGIR18)
- **Generalize to the full datasets:** (KDD16) ConUCB (WWW20)
- **Extract from user review:** SAUR (CIKM18)
- **Corpus based:** the line of NLU/NLG works

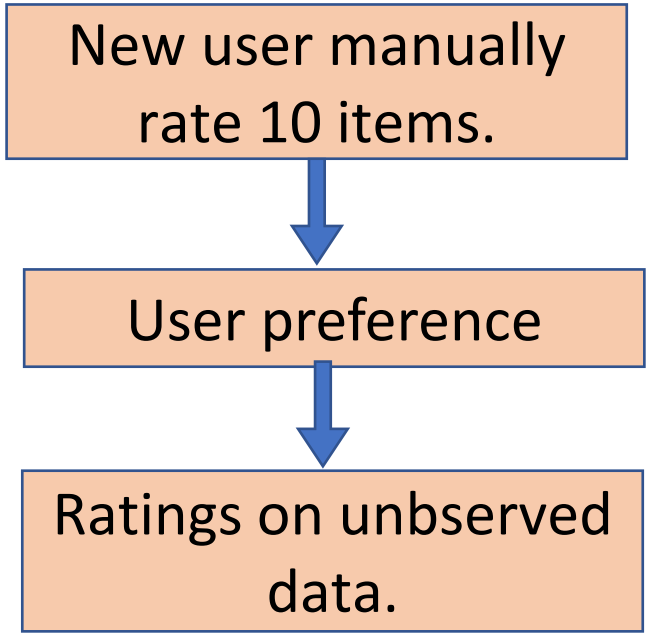
• Summary – User preference simulation – User Click History

User Click History:

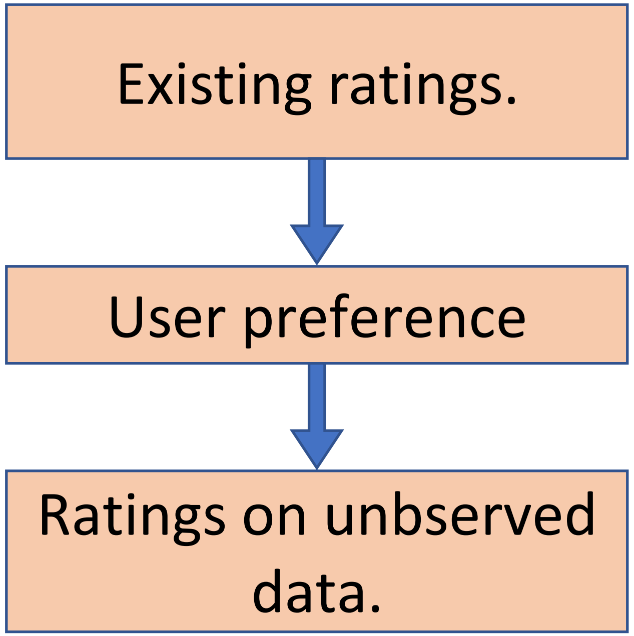
- Observed (user – item) pairs are used as positive samples, unobserved once as negative samples.
- During one conversation session, we sample one (user – item) pair.
  - During this session, the user will only like this item.
  - During this session, the user will only like the attributes of this item.

• Summary – User preference simulation – Generalize to the Whole Candidate Testing Set

- Get user’s ground-truth preference score on a small amount of data.
- Infer user’s preference for the full dataset.



Christakopoulou et al. “Towards Conversational Recommender Systems” (KDD’ 16)

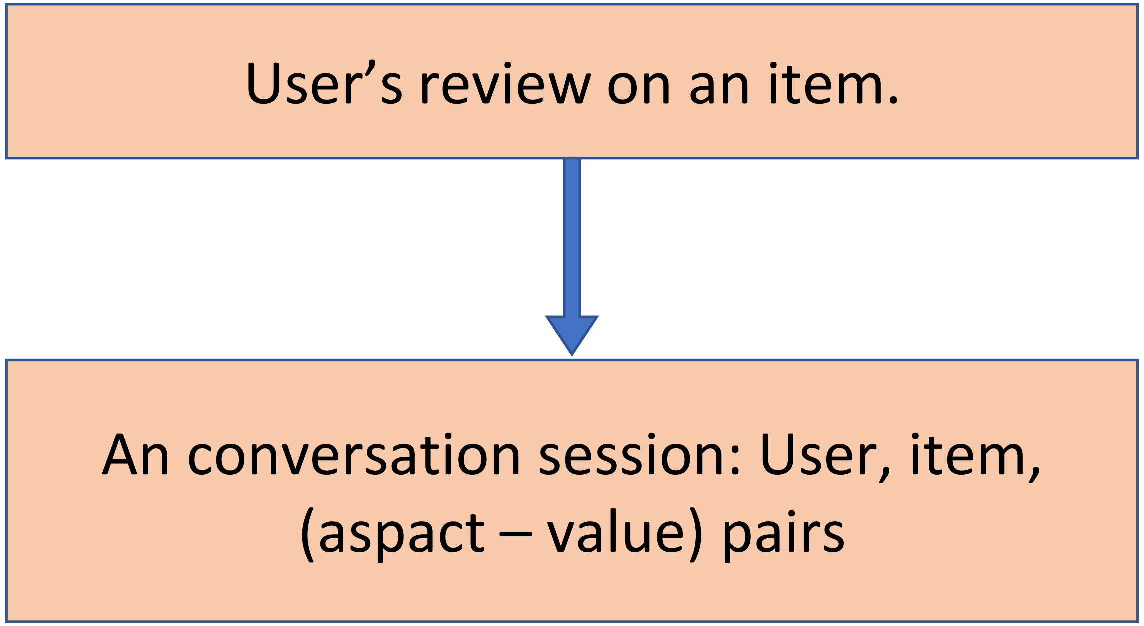


Zhang et al. “Conversational Contextual Bandit: Algorithm and Application” (WWW’ 20)

• Summary – User preference simulation – Extract from User Review

Extract from user review:

- Each review will be used to generate a conversation session.
- “Aspect – Value” pairs would be extracted from the review (e.g. “price” = “high”, ‘OS” = “Android”).



Zou et al. “Towards Question-based Recommender Systems”(SIGIR’ 20)
Zhang et al. “Towards Conversational Search and Recommendation: System Ask, User Respond”(CIKM’ 18)

• Summary – User preference simulation – Corpus based

HUMAN:	hello, can i help you find a movie ?
HRED:	hi what kind of movies do you like
OURS:	hi, how can i help you ?
SEEKER:	hello ! i would like to find a suspenseful, but clean family friendly movie .
...	...
SEEKER:	2001 : a space odyssey might be a great option. anything else that you would suggest ?
HUMAN:	you can try planet of the apes the older one is quite suspenseful and family friendly .

i.e. corpus

User actually likes "Star Wars" and dislikes "the planet of the apes".

Conversational recommendation through natural language.

- User's preference is recorded "as is" in the corpus. The evaluation is actually biased on responses in the corpus (which is often generated on AMTurker).

Li et.al. "Towards Deep Conversational Recommendations" (NIPS'18)

• Discussion on Future Researches

Formalization (problem setting):

- If a user accepts the recommendation, is it possible to recommend more?
- Can we optimize other goals other than clicking? For example, maximizing profits in E-commerce; maximizing total time spending in video sharing platform ...

Evaluation (simulating user preferences):

- How to reliably simulate user preferences and action in conversational recommendation scenarios!