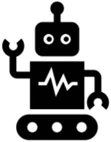
ConvLab: An Open Source Dialog System Platform

- An open source Multi-Domain End-to-End Dialog System Platform provided along with the 8th Dialog System Technology Challenge (DSTC8)

Fully annotate data
for training individual components or end-to-end models with supervision

Speaker	Utterance	Annotation
User	am looking for a place to to stay that has cheap price range it should be in a type of hotel	**Dialog acts** {"Hotel-Inform": [["Pricerange", "cheap"]]}
System	Okay, do you have a specific area you want to stay in?	**State** {"hotel": { "name": "not mentioned", "area": "not mentioned", "parking": "not mentioned", "pricerange": "cheap", "stars": "not mentioned", "internet": "not mentioned", "type": "hotel"}}} ... **Dialog acts** "Hotel-Request": [["Area", "?"]]
User	no, i just need to make sure it's cheap. oh, and i need parking	**Dialog acts** {"negate", "Hotel-Inform": [["Pricerange", "cheap"], ["Parking", "yes"]]}
System	I found 1 cheap hotel for you that includes parking. Do you like me to book it?	**State** {"hotel": { "name": "not mentioned", "area": "not mentioned", "parking": "yes", "pricerange": "cheap", "stars": "not mentioned", "internet": "not mentioned", "type": "hotel"}}} ... **Dialog acts** {"Hotel-Inform": [["Price", "cheap"], ["Choice", "1"], ["Parking", "none"]]}

User Simulators
for reinforcement learning
1 rule-based simulator
2 data-driven simulators



SOTA Baselines
Multiple models for each component
Multiple end-to-end system recipes

Jianfeng Gao, Michel Galley, Neural Approaches to Conversational AI (slides), ICML 2019

ConvLab: An Open Source Dialog System Platform

- Paper: `https://arxiv.org/pdf/1904.08637.pdf`
- Codes and data: `https://github.com/ConvLab/ConvLab`
- A good start point to learn how to build an end-to-end dialog system.
- Here we use some examples from ConvLab to demonstrate the details of a task-oriented dialog system.

ConvLab: Dialog Acts

general-bye	Booking-Book	Hotel-Inform
general-greet	Booking-Inform	Hotel-NoOffer
general-reqmore	Booking-NoBook	Hotel-Recommend
general-thank	Booking-Request	Hotel-Request
general-welcome	Police-Inform	Hotel-Select
Attraction-Inform	Police-Request	Restaurant-Inform
Attraction-NoOffer	Train-Inform	Restaurant-NoOffer
Attraction-Recommend	Train-NoOffer	Restaurant-Recommend
Attraction-Request	Train-OfferBook	Restaurant-Request
Attraction-Select	Train-OfferBooked	Restaurant-Select
Hospital-Inform	Train-Request	Taxi-Inform
Hospital-Request	Train-Select	Taxi-Request

ConvLab: Dialog Slots

Fee	Ref	Day	none
Addr	Food	Name	Depart
Area	Type	Car	People
Stars	Price	Leave	Dest
Internet	Stay	Time	Parking
Department	Phone	Arrive	Open
Choice	Post	Ticket	Id

ConvLab: Training data (Multiwoz)

sessID	MsgID	Text	DialogAct
PMUL1032	0	What kind of attractions are available in the centre?	Attraction-Inform(Area=centre)
PMUL1032	1	There is the Holy Trinity Church on Market Street. It is free to get in.	Attraction-Inform(Addr=Market Street;Fee=free;Name=Holy Trinity Church)
PMUL1032	3	Sorry there are no listings for multiple sports,can I check in another area?	Attraction-NoOffer(Type=multiple sports)
PMUL1032	4	How about any that are about architecture?	Attraction-Inform(Type=architecture)
PMUL1032	8	Yes, may I have the phone number and postcode? Also, is there an entrance fee?	Attraction-Request(Post;Phone;Fee)
PMUL1032	9	The phone number is 01223332320 and the entrance fee is free. The post code is cb21tt.	Attraction-Inform(Fee=free;Phone=01223332320;Post=cb21tt)
PMUL1032	10	Thank you that is all I needed.	general-thank()
PMUL1033	0	I need to book a train to cambridge on Monday.	Train-Inform(Dest=cambridge)
PMUL1033	1	Where will you be departing from?	Train-Request(Depart)
PMUL1033	2	I will be departing out of Stevenage.	Train-Inform(Depart=stevenage)
PMUL1033	4	I want to arrive by 16:45.	Train-Inform(Arrive=16:45)
PMUL1033	6	Sorry, I looked at the calendar. I need a Thursday train, not a Monday. Can you please find a train on that day instead?	Train-Inform(Day=thursday)
PMUL1033	7	Okay, no problem. The TR1163 train leaves at 05:54. Will that work for you?	Train-Inform(Id=TR1163;Leave=05:54)
PMUL1033	8	What time will the train arrive in Cambridge?	Train-Inform(Dest=cambridge)
PMUL1033	9	It arrives at 06:43.	Train-Inform(Arrive=06:43)
PMUL1033	10	Ok please book that for 5 people.	Train-Inform(People=5)

a small piece

ConvLab: Distribution of Dialog Acts & Slots

User Dialog Act:			System Dialog Act:		
Attraction-Inform			Attraction-Inform		
	Area	2084		Addr	2664
	Name	1447		Area	2365
	Type	2477		Choice	2241
	none	269		Fee	1922
Attraction-Request				Name	3072
	Addr	1236		Open	13
	Area	379		Phone	1741
	Fee	1109		Post	1579
	Phone	1314		Price	48
	Post	1401		Type	2589
	Type	258		none	28
Hospital-Inform			Attraction-NoOffer		
	Department	95		Addr	1
	none	231		Area	324

a small piece

NLU: Natural Language Understanding

- Tasks:
  - Intent (dialog act) detection: text classification
  - Slot filling: sequence labeling
- Technologies:
  - Intent detection: SVM, CNN, CNN-LSTM, BERT
  - Slot filling: CRF, Bi-LSTM, Bi-LSTM-CRF, ELMo/BERT
- Challenges:
  - Low resource
  - Domain adaptation
  - Out-of-domain problem

DST: Dialog State Tracking

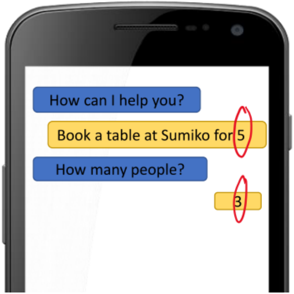
- Tasks:
  - Update the dialog states with the lastest input
- Technologies:
  - Rule-based method: when you have a high quality NLU module, a simple rule-based method to incorporate the NLU result with exisitng states (adding new slots or updating existing slots) will work well.
  - Neural method: it is a common practice to construct a neural DST module which has NLU included.

Belief state: distribution of dialog states

- Maintain a probabilistic distribution instead of a 1-best prediction for <u>better robustness to LU errors or ambiguous input</u>

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

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



Jianfeng Gao, Michel Galley, Neural Approaches to Conversational AI (Slides), ICML 2019

Young, S., Gašić, M., Keizer, S., Mairesse, F., Schatzmann, J., Thomson, B. and Yu, K., 2010. The hidden information state model: A practical framework for POMDP-based spoken dialogue management. Computer Speech & Language, 24(2), pp.150-174.

Neural Belief Tracker: Data-Driven Dialogue State Tracking

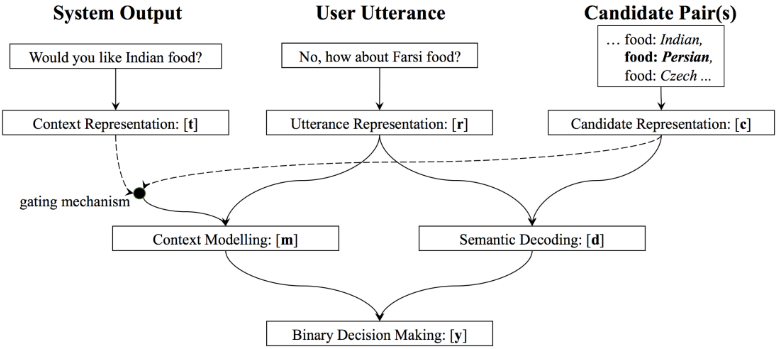


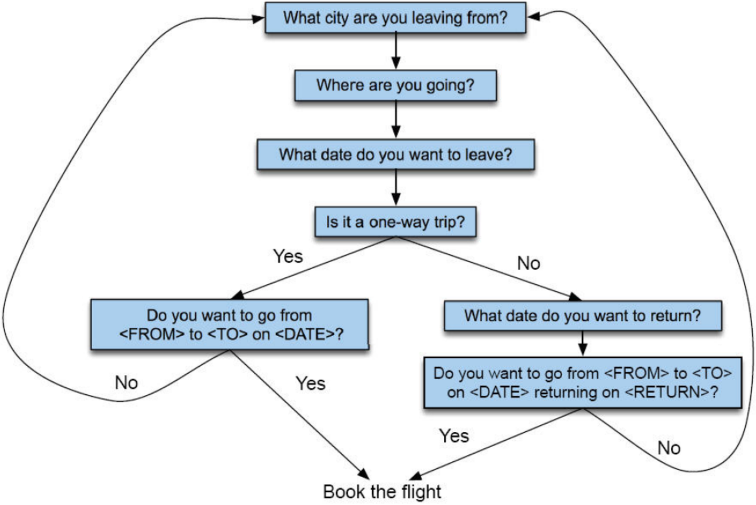
Figure 3: Architecture of the NBT Model. The implementation of the three representation learning subcomponents can be modified, as long as these produce adequate vector representations which the downstream model components can use to decide whether the current candidate slot-value pair was expressed in the user utterance (taking into account the preceding system act).

Mrkšić, N., Séaghdha, D.O., Wen, T.H., Thomson, B. and Young, S., 2016. Neural belief tracker: Data-driven dialogue state tracking. arXiv preprint arXiv:1606.03777.

DP: Dialog Policy

- Dialog policy decides what and how a taskoriented dialog system will respond, according to the current dialog states.
- Dialoy policy plays a vital role in delivering effective conversations.
- Approaches for dialog policy:
  - Rule-based policy
  - Supervised learning policy
  - Reinforcement learning policy

Rule-based policy



Original slides by Dan Jurafsky

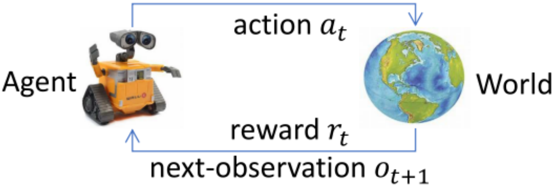
Rule-based policy

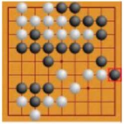
- Simple and straightforward. Easy to implement.
- System takes the control of the whole conversation.
  - However, in a real conversatipon, the user may not strictly follow the pre-defined flow. For example:
    - The user may prefer to give the details in a different order than the pre-defined one;
    - The user may give multiple details in one utterance;
    - The user may ask, confirm, or clarify something.
  - In such a scenario, even if the user's response is pertinent to the task, the system will not be able to complete the task.
- The conversation will be inflexible and unfriendly to users.

Policy by supervised learning

- The policy may be learned with a supervised learning methods.
- Given the current dialog states, learn a policy to predict the action (dialog act and slots) from the training data.
- The supervised learning for dialog policy is limited by the small size of the training data and the lack of exploration of the dialog state space.
- Since dialog systems interact with the enviroment constantly, reforcement learning would be a better choice in such a scenario.

Reinforcement Learning (RL)





Goal of RL

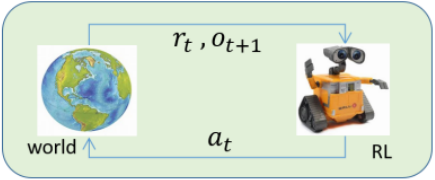
At each step t, given history so far s_t, take action a_t to maximize long-term reward ("return"):

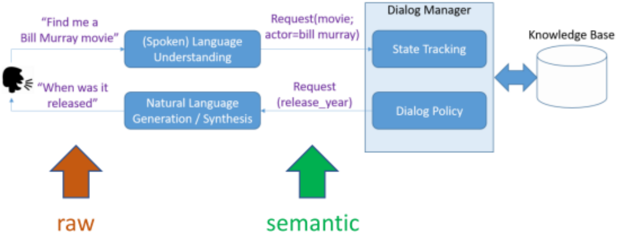
$$R_t = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \cdots$$

"Reinforcement Learning: An Introduction", 2nd ed., Sutton & Barto

Jianfeng Gao, Michel Galley, Neural Approaches to Conversational AI (slides), ICML 2019

Conversation as RL





Jianfeng Gao, Michel Galley, Neural Approaches to Conversational AI (slides), ICML 2019

Conversation as RL

- State and action
  - Raw representation
    (utterances in natural language form)
  - Semantic representation
    (intent-slot-value form)
- Reward
  - +10 upon successful termination
  - -10 upon unsuccessful termination
  - -1 per turn
  - ...

Early work:

Esther Levin, Roberto Pieraccini, and Wieland Eckert. A stochastic model of human-machine interaction for learning dialog strategies. IEEE Transactions on speech and audio processing 8.1 (2000): 11-23.

Pietquin, Olivier. A framework for unsupervised learning of dialogue strategies. Presses univ. de Louvain, 2005.

Jason D. Williams and Steve Young. Partially observable Markov decision processes for spoken dialog systems. Computer Speech & Language 21.2 (2007): 393-422.

Jianfeng Gao, Michel Galley, Neural Approaches to Conversational AI (slides), ICML 2019

Conversation as RL

- RL for dialog policy is very promising and has sparkled a lot of research:
  - Value-based RL
  - Policy-based RL
  - Model-based RL
- Despite of the success of RL in dialog system research, rule-based policy is still commonly used in industries because of its simplicity and being easy to handle.

User simulation

- One of the research lines is to use user simulators:
    - Train a user simulator off-line by rule-based methods or supervised learning on the training corpus;
    - Train the dialog policy on-line with the responses from the user simulator by reinforcement learning.
- ConvLab provides two types of user simulator:
    - rule-based simulators, and
    - data-driven simulators.

NLG: Natural Language Generation

- NLG module generate the system utterance according to the speech act (or action, including the intent and slot-value pairs), which is provided by the dialoy policy module.
- For task-oriented dialog systems, if the task is not too complex, a rule-based generation should work well.
- In recent years, deep learning based NLG for dialog systems has become popular and can generate texts in rather good quality.

End-to-End training for dialog systems

- The training of a pipeline dialog system is complex and hard to adopt to new domain.
- End-to-end training for dialog systems has been proposed and become a popular paradigm.
- The data for end-to-end training contains not only the dialog utterances from the user and the dialog system, but also the dialog acts and slot-value pairs for each utterance, as we have seen in the ConvLab system.

End-to-End training for dialog systems

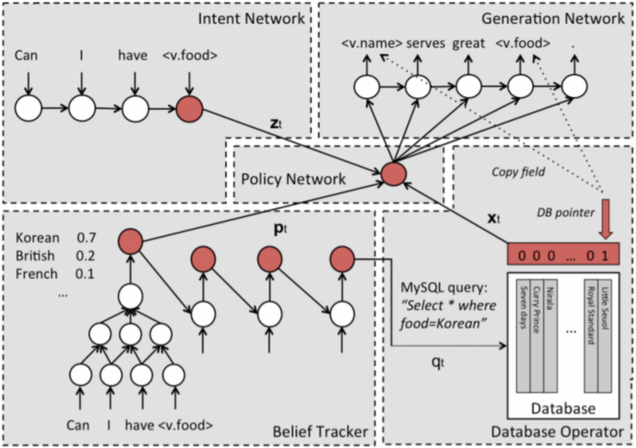


Figure 1: The proposed end-to-end trainable dialogue system framework

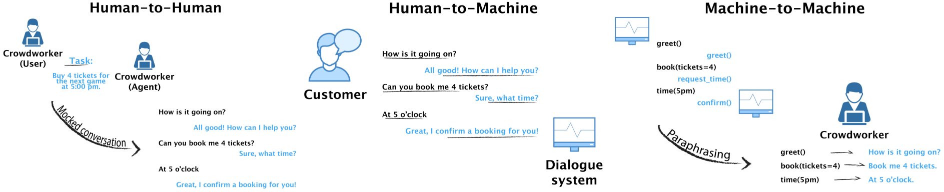
Tsung-Hsien Wen, et al. A Network-based End-to-End Trainable Task-oriented Dialogue System. EACL 2017.

Data Collection and Wizard-of-Oz

- To collect dialog data from real human-to-human conversation is really difficult.
- A Wizard-of-Oz (WoZ) paradigm is proposed to solve this problem:
  - a human (e.g., a crowdworker) takes the role of a user with a specic task in mind: for instance, she wants to book a restaurant through a call center, or has a question for a customer service.
  - The role of the system agent (so called the wizard) is then played by another crowdworker who has access to the knowledge (e.g., databases, FAQs, the user's history) required to complete the task.
  - An actual conversation is set within a particular domain and “mocked” between the two parties.

Data Collection and Wizard-of-Oz

- WoZ has been used to create two largest research datasets for task-oriented dialogue today: Stanford multi-domain dataset (SMD) and Multi-Domain Wizard-of-Oz (MultiWOZ)
- Improved versions of WoZ experiments have been proposed and used to collect dialog data:



The Magic Triangle of Dialogue Data Collection, PolyAI

Evaluation for task-oriented dialogs

- Many metrics can be applied for evaluation of task-oriented dialog system, which are mainly categorized as:
  - Human evaluation:
    - Evaluation done by real human users.
    - Reliable but expensive.
    - Metrics: Task success rate, dialog length, irrelevant turn rate, redundant turn rate, user satisfaction score, etc.
  - Automatic evaluation:
    - Evaluation by the user simulator.
    - Cheap but not so reliable.
    - Metrics: Task success rate, dialog length, average rewards, etc.

Content

Chitchat dialog systems

- A chitchat dialog system (social bot) is expected to chat with the user freely on any topics.
- It had been an extremely difficult task and not practically feasible until the neural sequence-to-sequence was applied:
  - L. Shang, Z. Lu, and H. Li. Neural responding machine for short-text conversation. ACL-IJCNLP 2015.
  - O. Vinyals and Q. Le. A neural conversational model. arXiv preprint arXiv:1506.05869, 2015.

Seq2seq generation for chitchat dialogs

- The sequence-to-sequence model used for chitchat dialogs is similar to neural machine translation models:
    - An encoder takes the dialog history as the input and encodes it into an internal representation;
    - An decocer takes the internal representation as the input and decodes it into an output sequence as the response.
- Huge open domain conversation data can be collected from social media, which provide almost unlimited resource to build open domain chitchat dialog systems.

Seq2seq generation for chitchat dialogs

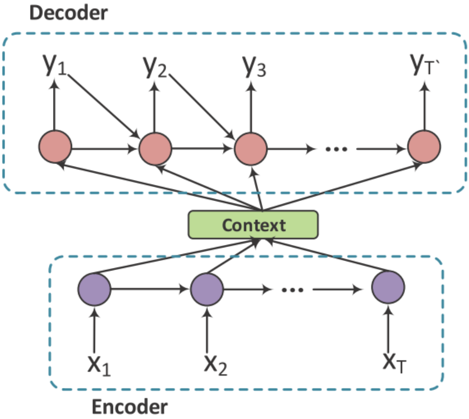


Figure 2: An Illustration of the Encoder-Decoder Model.

Hongshen Chen et al., A Survey on Dialogue Systems: Recent Advances and New Frontiers, ACM SIGKDD Explorations Newsletter 19.2 (2017)

A piece of dialog generated by a seq2seq model

Neural model learns the general shape of conversation, and the system output is situationally appropriate and coherent...	
H:	Hi
M:	**How are you?**
H:	Not bad
M:	**What's wrong?**
H:	Nothing really
M:	**Wanna talk about it? I'm here if you wanna talk x**
H:	Well, I'm just tired
M:	**Me too, I'm here if you wanna talk**
H:	Thanks, what are you doing?
M:	**You're welcome and I'm watching TV and you?**
H:	On Netflix?
M:	**Yeah, I'm watching it on Netflix.**
H:	What are you watching?
M:	**A TV show.**
H:	Nice, did you have dinner already?
M:	**No, not yet. what about you?**

Jianfeng Gao, Michel Galley, Neural Approaches to Conversational AI, ICML 2019

Problems for seq2seq dialog generation

- Although neural seq2seq has made great progress in chitchat dialog generation, there are still many problems for this method:
  - Universal response: the system frequently generate universal response like "I don't know.", "Ok", etc.
  - Inconsistant: the system often provides contradictory information in the same session: "I'm 14.", "I'm 16."
  - Long context: it is difficult to encode very long dialog history.
  - External knowledge: it is difficult to incorporate external knowledge into the dialog generation process.
  - Persona: it is hard to assign a specific personality to the system.
- Intensive research has been conducted in this research area in recent years.

Retrieval-based chitchat generation

- Another effective to produce natural chitchat dialog is to use retrieval-based method, if the dialog data is large enough.
- Retrieval-based method and generation-based method are complementary to each other and are often used togethor to form a hybrid method.

Evaluation for chitchat dialogs

- Automatic evaluation of chitchat dialog remains an open question.
- Evaluation metrics borrowed from machine translation and text summarizaiton like BLUE, METEOR and ROUGH are offen used, although not satisfactory.
- New metrics using word embeddings or neural networks are also proposed.

Content