# Talking to Machines: The System

Christian Saam, Trinity College Dublin

The ADAPT Centre is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund.

# Overview



- Motivation
- Short overview of classic rule-based vs. stochastic
- Neural network dialogue systems
- User modeling

# Motivation



Spoken Dialog Systems - hot topic

- rise of conversational agents Siri, Cortana, Alexa etc.
- rise of web-based chat-bots
- promise of virtual companions
- performance improvements in base technology ASR
- ease/naturalness of interaction
- proliferation of mobile devices and ubiquitous use

# Motivation...



Application of spoken dialogue systems

| task-oriented | non task-oriented |
| --- | --- |
| use of dialogues to accomplish a task | conversational interaction but not necessarily to achieve task |
| provide information (interactive question answering), complete task (book train), personal assistant | social dialog |

# Why Machine Learning

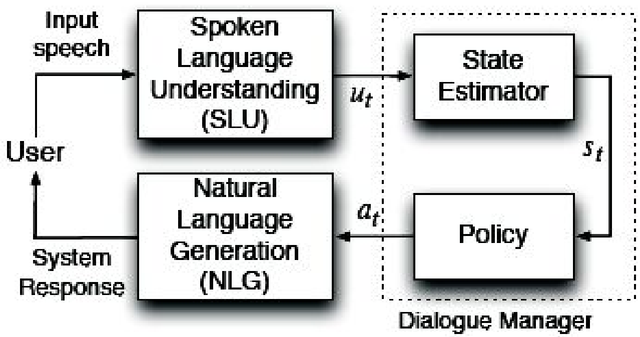


- Mixed and user initiative hard to control - system initiative often restrictive and inflexible
- More open domains require extensive effort in rule-based systems
- Inherent uncertainty
  - speech can be noisy
  - language is ambiguous
  - user is undecided

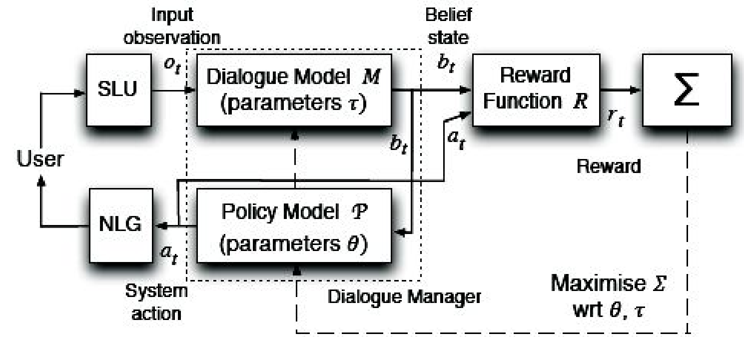
  Uncertainty is well modeled with probabilities

# (Partially Observable) Decision Process






Finite state spoken dialogue system (Young et al., 2013)




POMDP-based spoken dialogue system (Young et al., 2013)

| | |
|---|---|
| convert input speech to an abstract representation of the user's intent $u_t$ | decode input speech into noisy observation $o_t$ of underlying intent $u_t$ |
| update dialogue state $s_t$ | update belief distribution $b_t$ over all possible states |
| apply deterministic decision rule (*policy*) mapping state to response action $a_t$ | apply stochastic rule to choose $a_t$ |
| rules of State Estimator and Policy are hand-crafted | parameters of *M* and *P* can be optimised with reinforcement learning given appropriate reward function |

# Rule-based vs. Stochastic

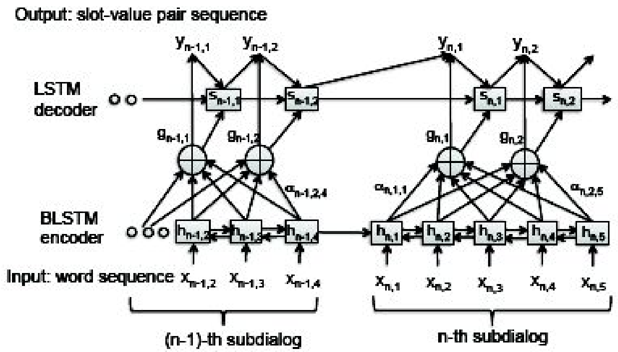


| | |
|---|---|
| need expertise | need lots of data |
| design error handling strategies or restrict interaction | learn from interaction |
| in real-world scenarios many rules | in real-world scenarios state-space explodes and becomes hard to handle requiring complex algorithms and computational power |
| need to fine-tune | can learn optimal decision policies |

# Cascaded Modular Neural DS



- simple case, replicate traditional modular approach with a pipeline of trained neural networks
- need labels at every module
- can use S2S approach for all models
- example: state tracker

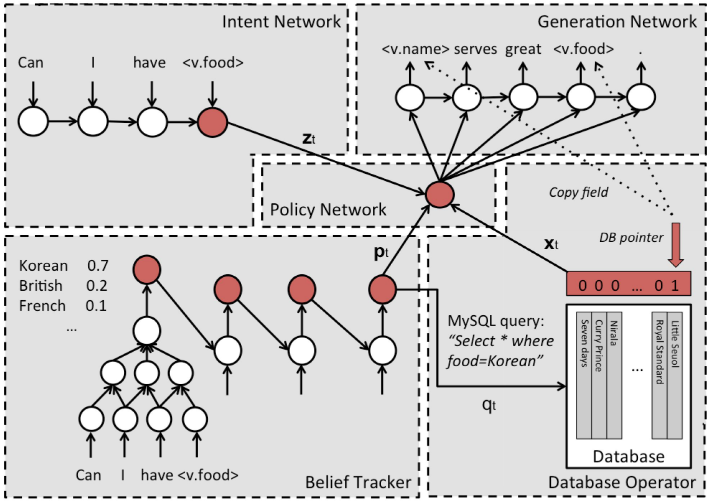



Dialogue state tracking with attention-based sequence to sequence learning
(Hori et al., 2016)

# End-to-end: Modular Neural Network



- classic modules in S2S approach
- belief tracker pre-trained with cross entropy error btw slot-value labels and predictions
- other params optimised with cross entropy error from GN predictions
- extra database operator (not differentiable, not trained)

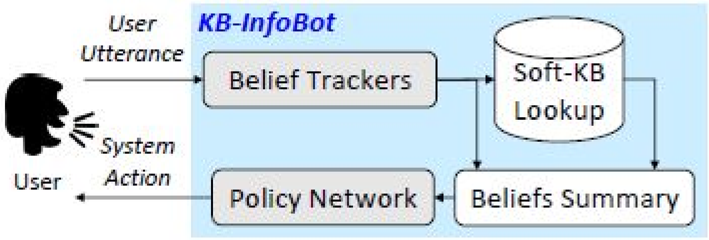



End-to-end trainable dialogue system (Wen et al., 2016)

# End-to-end: Differentiable Information Access



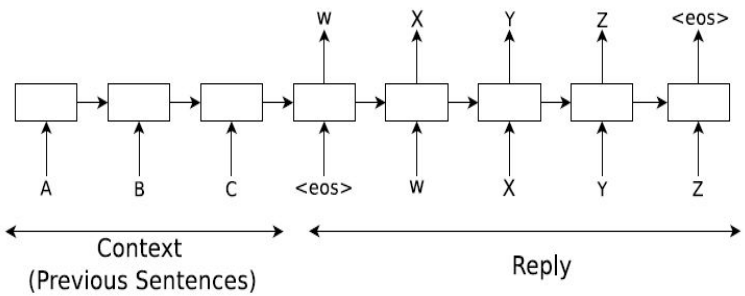
- replace symbolic querying with induced "soft" posterior distribution over knowledge base
- can integrate learning of DB access in reinforcement learning



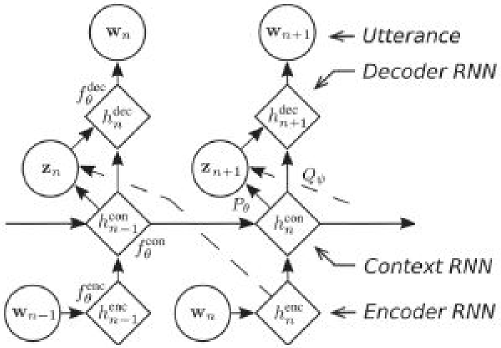

End-to-end infobot trained with RL
(Dinghra et al., 2017)

# End-to-End: Sequence-to-Sequence Models






Seq2Seq Conversational Dialogue Model
(Vinyals, Le, 2015)



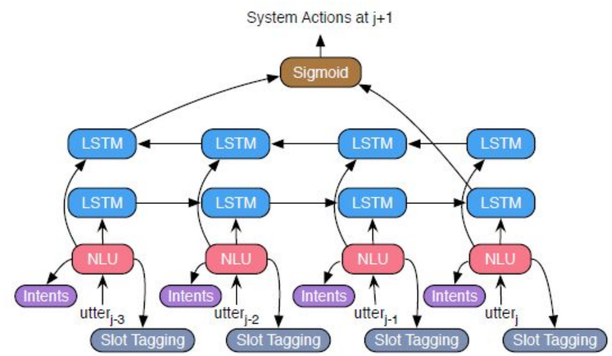

Latent Variable Hierarchical Recurrent
Encoder-Decoder (Serban et al., 2017)

- tend to generate repetitive generic answers
  - hierarchical encoder to represent levels of dialogue organisation
  - incorporate latent variable model to capture variation
  - diversity enhancing decoding
  - augment supervised training with RL to account for future
  - adversarial training
- cannot easily incorporate domain knowledge/constraints

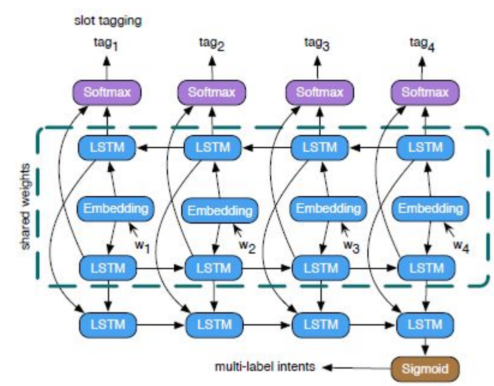
# End-to-end: Unifying Models



- NLU error prone
- noisy NLU output affects DM
- retraining of one component may adversely affect other component
- unify NLU and DM in one model




End-to-end joint NLU-DM Model (Yang et al., 2016)

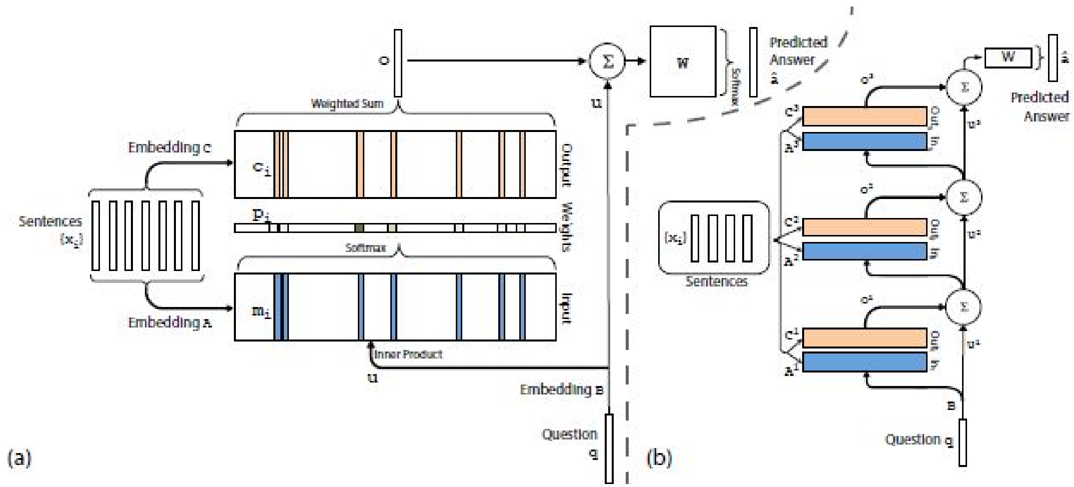



Detail NLU model (Yang et al., 2016)

# End-to-End: Memory Augmentation



- plain end-to-end s2s models attractive as simple
- success on chit-chat doesn't carry over to tasks
- alternative non-modular end-to-end approach

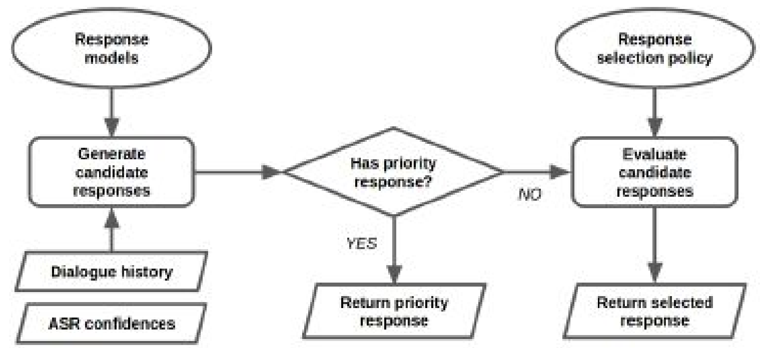



Single (a) and multi-layer (b) End-to-end Memory Networks (Sukhbaatar et al., 2015) used in Bordes et al., 2017

# Conversational DS: Ensemble



- to cope with topic variability train many dialog systems (conversational, info-seeking, story telling etc.)
- use a reinforcement learned meta-process to choose best move among all responses

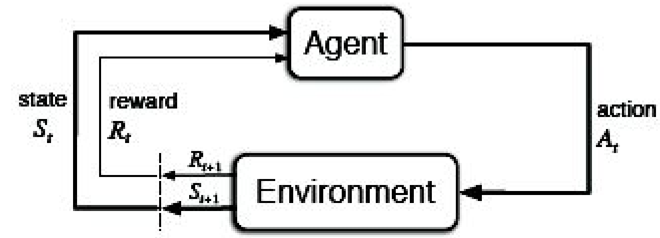



Large ensemble model conversational agent Dialogue Manager control flow (Serban et al., 2017)

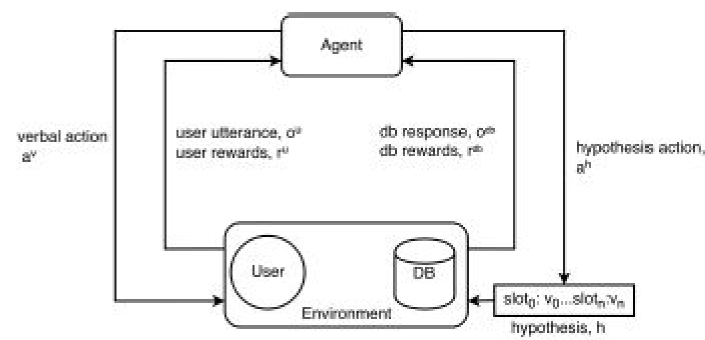
# Reinforcement Learning



- Dialogue Policy model takes dialogue state $s_n$ as input and produces system action $a_n$ as output.
- Dialogue Policy modeled by MDP $\{S, A, P, R, \gamma\}$
- Policy Learning can be cast as a Reinforcement Learning problem
- Cumulative Reward is $G_n = \Sigma_k \gamma^k r_{n+k}$
- The objective is to maximize Cumulative Reward
- 3 popular Policy Learning frameworks
    - Value-based RL
    - Policy-based RL
    - Actor-critic RL




Agent-environment interaction in a Markov Decision Process (Sutton, Barto, 2017)




End-to-end task-oriented Dialog Manager (NLU+DST+DM) (Zhao, Eskenazi, 2016)

# User Modeling



- So far mostly simple preferences
- For companions need long term deep understanding of user
  - current state vs. slow changing base state
- Probably need to model ontology of complex social relations
- Desirable to acquire this ontology from text mining