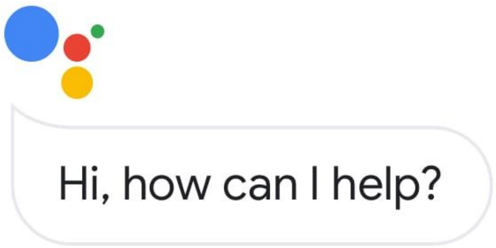
Dialogue Generation

March 31, 2020

Paula Gradu, Xinyi Chen

Dialogue Agents

- Personal assistants:
  - Google Assistant, Alexa, Bixby
- Communicating with robots
- Therapy for mental health
- For fun



Two Classes of Systems

- Chatbots (this lecture and next lecture)
  - Open domain

- Goal-based dialog agents (next Tuesday)
  - Book restaurants or flights
  - Closed domain

Previous Chatbots

- Eliza (1966)
- Parry (1968)
- CleverBot
- Microsoft Little Bing（小冰）

Chatbot Architectures

- Rule-based
  - Pattern-action rules (Eliza)
  - + a mental model (Parry)
- Corpus-based
  - Information Retrieval (IR) (CleverBot)
  - Neural Network Encoder-Decoder (this lecture)

Eliza

You are like my father in some ways.

WHAT RESEMBLANCE DO YOU SEE

You are not very aggressive but I think you don't want me to notice that.

WHAT MAKES YOU THINK I AM NOT AGGRESSIVE

You don't argue with me.

WHY DO YOU THINK I DON'T ARGUE WITH YOU

You are afraid of me.

DOES IT PLEASE YOU TO BELIEVE I'M AFRAID OF YOU

My father is afraid of everybody.

WHAT ELSE COMES TO MIND WHEN YOU THINK OF YOUR FATHER

Bullies.

DOES THAT HAVE ANYTHING TO DO WITH THE FACT THAT YOUR BOYFRIEND MADE YOU COME HERE

Eliza

You are like my father in some ways.

WHAT RESEMBLANCE DO YOU SEE

You are not very aggressive but I think you don't want me to notice that.

WHAT MAKES YOU THINK I AM NOT AGGRESSIVE

You don't argue with me.

WHY DO YOU THINK I DON'T ARGUE WITH YOU

You are afraid of me.

DOES IT PLEASE YOU TO BELIEVE I'M AFRAID OF YOU

My father is afraid of everybody.

WHAT ELSE COMES TO MIND WHEN YOU THINK OF YOUR FATHER

Bullies.

DOES THAT HAVE ANYTHING TO DO WITH THE FACT THAT YOUR BOYFRIEND MADE YOU COME HERE

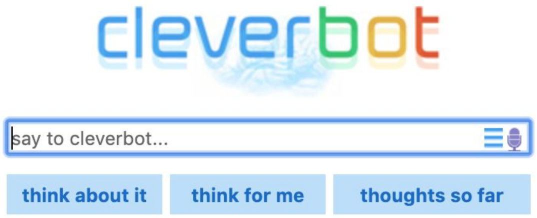
The trick: to be a Rogerian psychologist: draw patients out by reflecting back at them

Eliza

```
function ELIZA GENERATOR(user sentence) returns response

Find the word w in sentence that has the highest keyword rank
  if  w exists
       Choose the highest ranked rule r for w that matches sentence
       response ← Apply the transform in r to sentence
       if  w = 'my'
          future ← Apply a  transformation from the 'memory' rule list to sentence
          Push future onto memory stack
   else (no keyword applies)
    either
       response ← Apply the transform for the NONE keyword to sentence
    or
       response ← Pop the top response from the memory stack
 return(response)
```

CleverBot



User:	Do you like Doctor Who?
Cleverbot:	Yes, so funny
User:	What's your favorite show?
Cleverbot:	I don't have one
User:	What's your name?
Cleverbot:	Jason I already told you.
User:	No your real name.
Cleverbot:	Stephanie IS my real name.
User:	I thought it was Jason?
Cleverbot:	How did you come to that conclusion.
User:	Because you said so.
Cleverbot:	I have to do my homework.

https://www.cleverbot.com

IR-based Systems

- Given user query, find response to the closest turn in corpus

- Return closest turn in corpus

Evaluation

- Ideally want human evaluation, but it’s not feasible for training, expensive even for evaluation
- Heuristics: perplexity, BLEU, dialogue length/diversity
- Still an open question

This lecture: neural conversational models

- **Paper 1**: (Vinyals and Le, 2015) A Neural Conversational Model
  - Pioneered seq2seq models for dialogue generation, trained on both large closed-domain and open-domain datasets
- **Paper 2**: (Li et al, 2016) Deep Reinforcement Learning for Dialogue Generation
  - Proposed to apply deep reinforcement learning to model future reward in chatbot conversations

Paper 1: A Neural Conversation Model

Motivation:

- Previously this task was specialized to a narrow domain and relied on hand-engineered features.
- Advances in **Seq2Seq** models are **suitable** for conversational modeling.
- This work casts this task as sequence prediction. **End-to-end approach**, not domain-specific.

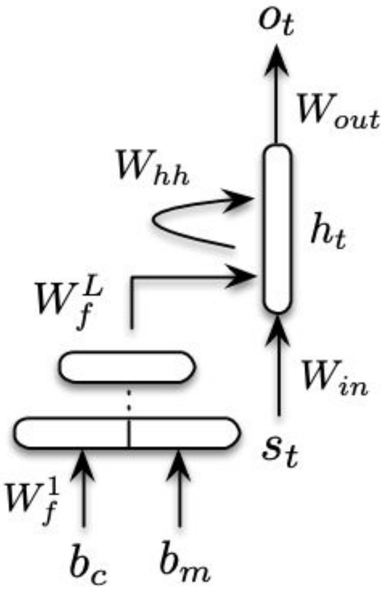
Concurrent Work

- Twitter: A Neural Network Approach to Context-Sensitive Generation of Conversational Responses (Sordoni et al. 2015)
- Weibo: Neural Responding Machine for Short-Text Conversation (Shang et al. 2015)
- Main Idea: Use RNNs to model dialogue in short conversations

Concurrent Work on Twitter

A Neural Network Approach to Context-Sensitive Generation of Conversational Responses (Sordoni et al. 2015)

- Encoding step: Multilayer non-linear forward architecture
- Decoding step: Recurrent Neural Network Language Model (RLM)



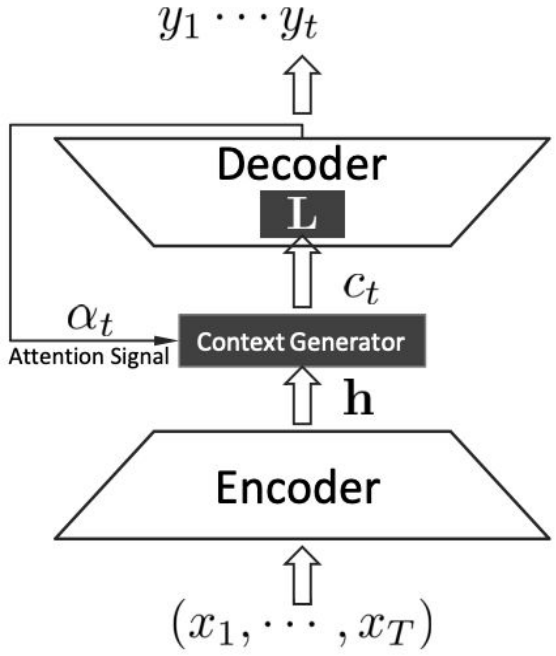
Concurrent Work on Twitter

#	Context c	Message m	Generated Response r
1	noone can help this time its just i need a support system	well im here to support you. whenever you need me.	thank you, i appreciate it thank you
2	nothin much, and how's the book?!	its good but i'm only like halfway through cuz i don't feel like reading. i'm so bored ...	that's good! i have the book but i'm bored too.
3	i know! what have you been up to?	nothing really. just enjoying summer and working. what about you?	that's good! nothing much just listening to music and you?

Concurrent Work on Weibo

Neural Responding Machine for Short-Text Conversation (Shang et al. 2015)

- encoder-decoder framework
- both steps use an RNN
- argues against seq2seq encoding approach

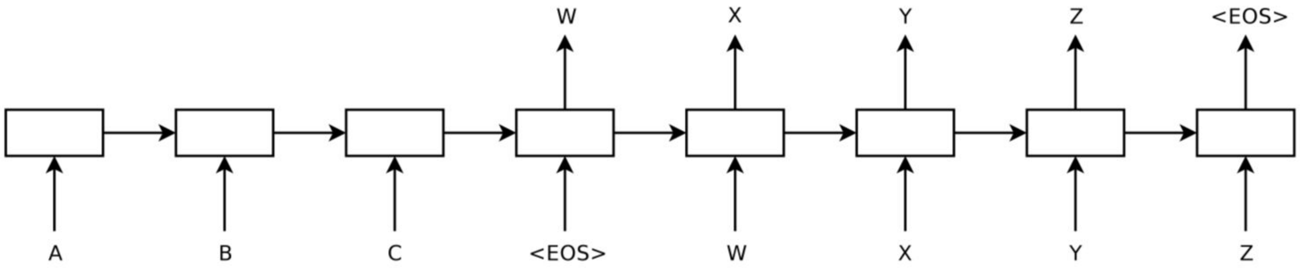


Concurrent Work on Weibo

ID	Post	NRM-glo	NRM-loc	NRM-hyb	Rtr.-based
1	每年元旦都来高烧。 High fever attacks me every New Year's Day	多休息，增强免疫力！ Rest more to boost your immune system.	每年元旦都去医院看病 Go to see a doctor every New Year's Day	祝早日康复，身体健康！ Get well soon and stay healthy!	高烧… High fever…
2	又老了一岁，感谢我们组，我很高兴 I gain one more year. Grateful to my group, so happy	生日快乐！永远支持你！ Happy birthday! Will stand by you forever!	生日快乐！永远年轻漂亮！ Happy birthday! Wish you always stay young and pretty!	老了老了，岁月不饶人啊！ Getting old now. Time has no mercy.	我只是老了一岁 I just gain only one more year.

Seq2Seq ‘backbone’

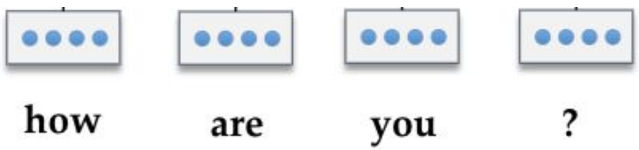
- **Encode** the input sequence to fixed-size vector w/ one RNN
- **Decode** the vector to the target sequence w/ another RNN
- **<EOS>** marker enables model to define a probability distribution over sequences of all possible lengths.



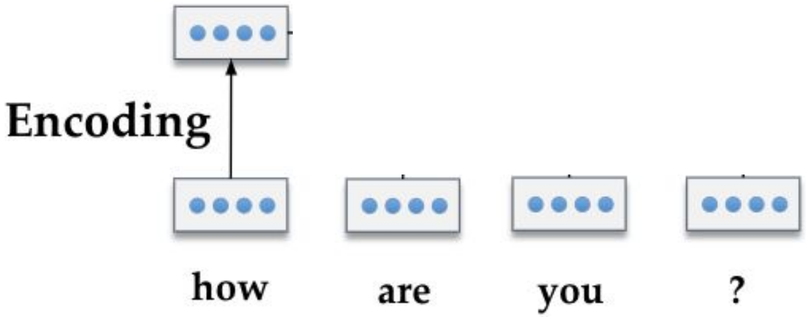
Seq2Seq Models for Response Generation

- Input: Context
- Output: Response
- Loss: Cross-entropy
- Teacher forcing during training
- ‘Greedy’ inference

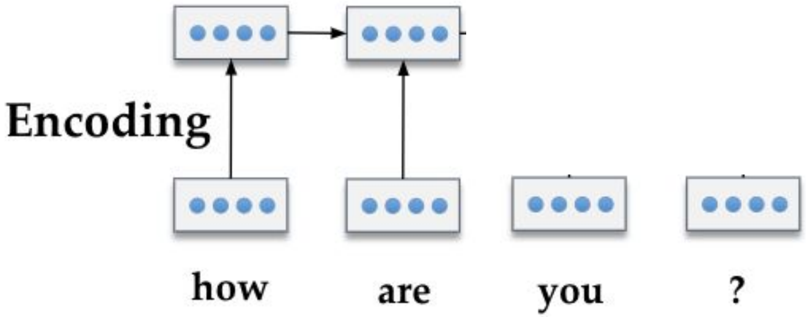
Seq2Seq Models for Response Generation



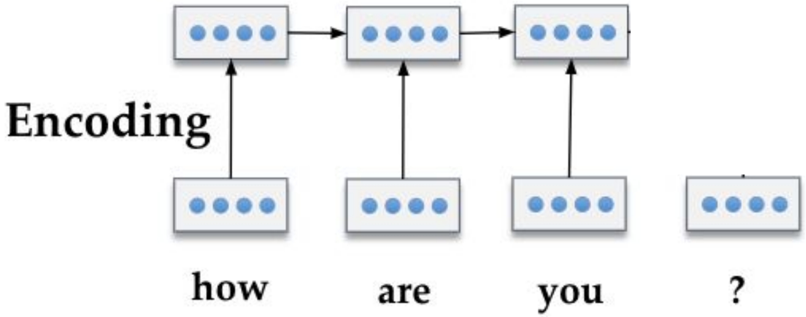
Seq2Seq Models for Response Generation



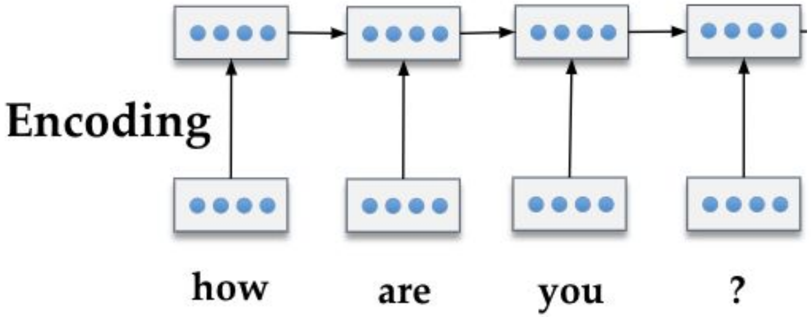
Seq2Seq Models for Response Generation



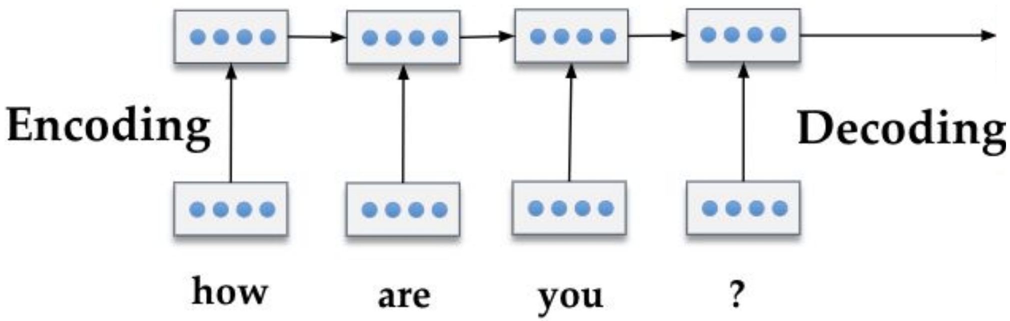
Seq2Seq Models for Response Generation



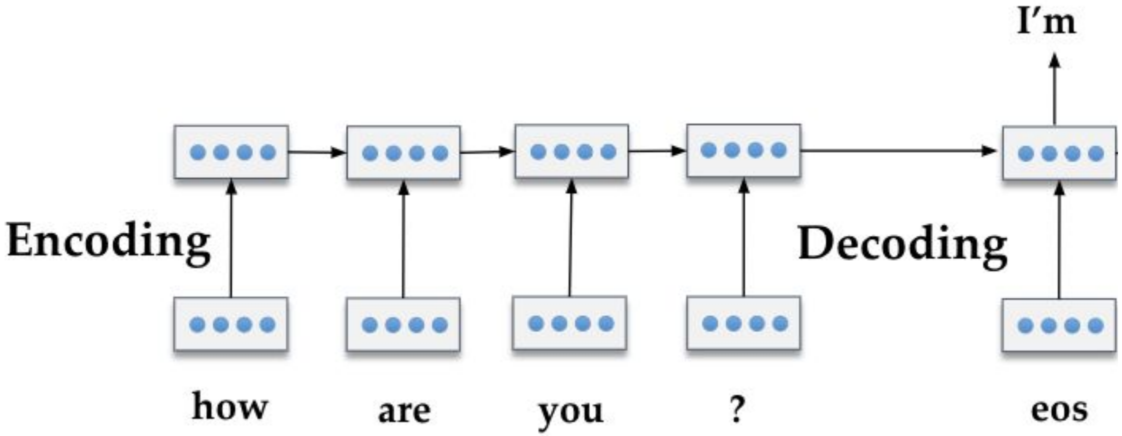
Seq2Seq Models for Response Generation



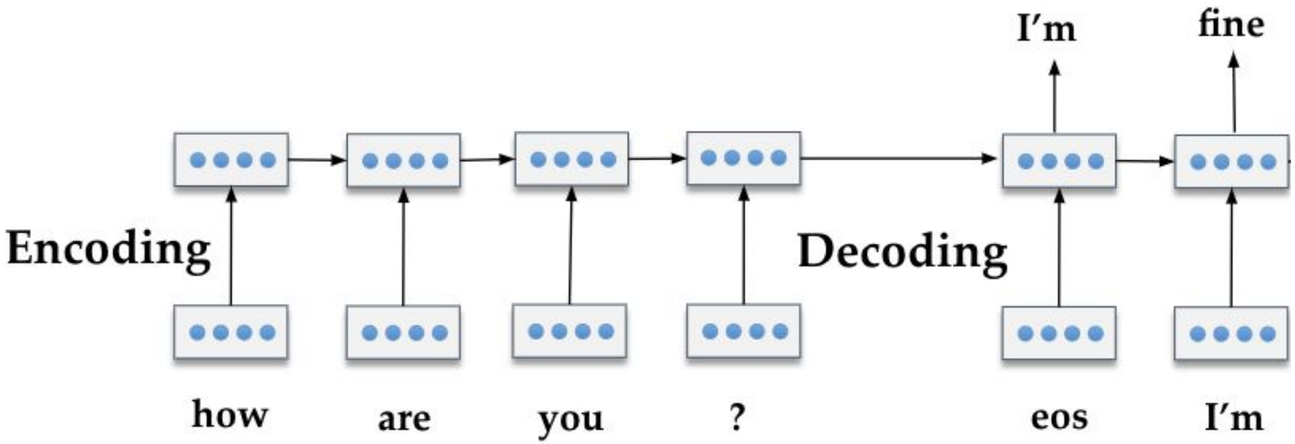
Seq2Seq Models for Response Generation



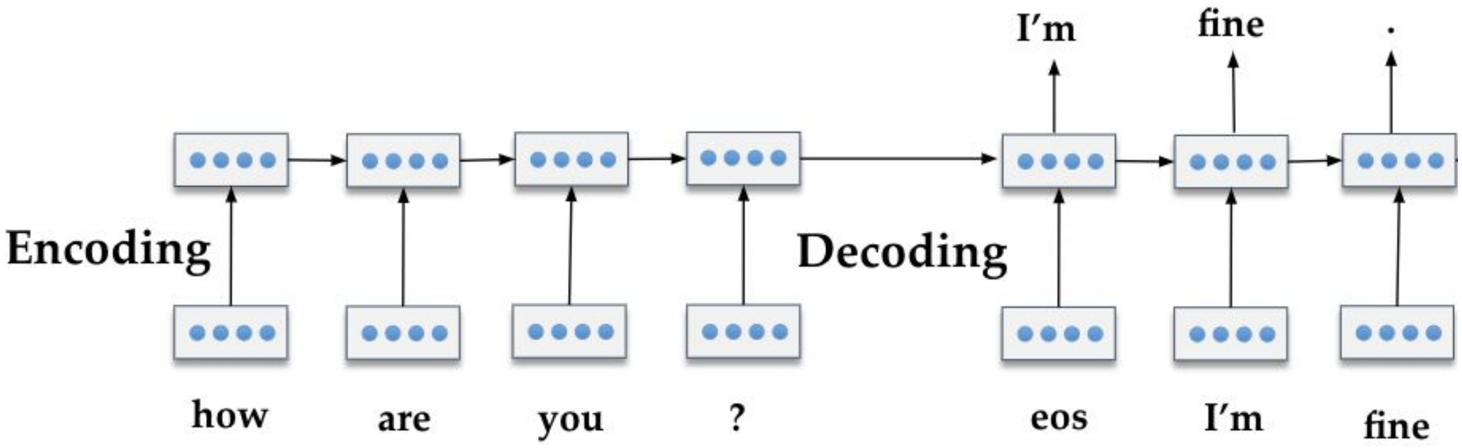
Seq2Seq Models for Response Generation



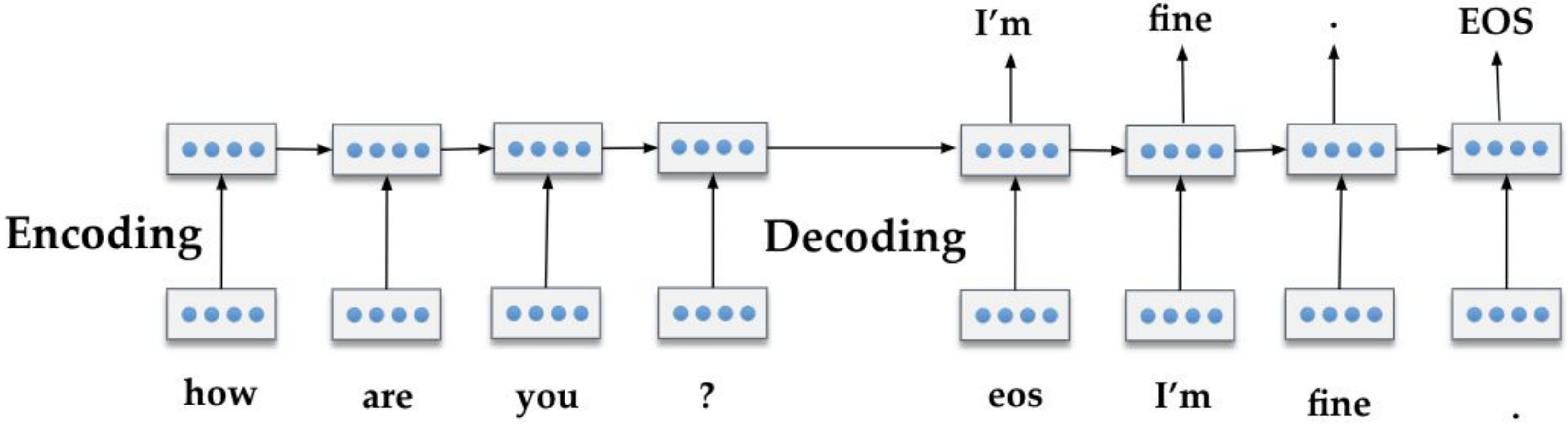
Seq2Seq Models for Response Generation



Seq2Seq Models for Response Generation



Seq2Seq Models for Response Generation



Closed Domain Dataset: IT Helpdesk

Describe your problem: i forgot my password
Machine: *hi , this is <NAME> from helpdesk connect , what can i help you with today ?*
Human: *hi , i forgot my password*

Machine: *ok great*
Human: *can you tell me how i can recover it ?*
Machine: *i ' ll need to verify who it is at that machine*

IT Helpdesk (Closed Domain) Experiment

- Dataset:
  - From a IT helpdesk troubleshooting chat service
  - ~400 words/ interaction
  - Turns are clearly signaled
  - Training: ~30M tokens
  - Validation: ~3M tokens
- Model:
  - Vocab: most common 20k words
  - 1-layer LSTM, 1024 memory cells
  - SGD

IT Helpdesk Example

Machine: *goto <URL> page and enter your name and password*
Human: *done*
Machine: *try logging in with <NAME> and check if its working fine now*
Human: *yes , now it works !*

IT Helpdesk Example

Machine: *goto <URL> page and enter your name and password*
Human: *done*
Machine: *try logging in with <NAME> and check if its working fine now*
Human: *yes , now it works !*

The <URL> indeed contains information about vpn access!!

IT Helpdesk Example

Machine: *goto <URL> page and enter your name and password*
Human: *done*
Machine: *try logging in with <NAME> and check if its working fine now*
Human: *yes , now it works !*

The <URL> indeed contains information about vpn access!!

IT Helpdesk Example

Machine: *ok this is the issue with the system performance , nasser , tools .*

IT Helpdesk Example

Machine: *ok this is the issue with the system performance , nasser , tools .*

IT Helpdesk Example

Machine: *ok this is the issue with the system performance , nasser , tools .*

Machine: *is it a windows machine ?*
Machine: *we have linux on your machine .*
Machine: *is it windows xp or windows 7 ?*
Human: *i am using linux*

IT Helpdesk Example

Machine: *ok this is the issue with the system performance , nasser , tools .*

Machine: *is it a windows machine ?*
Machine: *we have linux on your machine .*
Machine: *is it windows xp or windows 7 ?*
Human: *i am using linux*

IT Helpdesk Example

Machine: *ok this is the issue with the system performance , nasser , tools .*

Machine: *is it a windows machine ?*
Machine: *we have linux on your machine .*
Machine: *is it windows xp or windows 7 ?*
Human: *i am using linux*

Machine: *shd i reboot the machine .*

IT Helpdesk Example

Machine: *ok this is the issue with the system performance , nasser , tools .*

Machine: *is it a windows machine ?*
Machine: *we have linux on your machine .*
Machine: *is it windows xp or windows 7 ?*
Human: *i am using linux*

Machine: *shd i reboot the machine .*

IT Helpdesk Example Take-aways

- **Pros**:
  - Model sometimes successfully resolves issue
- **Cons**:
  - Doesn’t always make sense
  - Doesn’t always retain information
  - Not very good English
  - Very non-human-like

Open Domain Dataset: OpenSubtitles

If we've underestimated the size of the artifact's data stream...

We'll fry the ship's CPU and we'll all spend the rest of our lives stranded in the Temporal Zone.

The ship's CPU has a name.

Sorry, Gideon.

Can we at least talk about this before you connect...

Gideon?

You still there?

Oh my God, we killed her.

Artificial intelligences cannot, by definition, be killed, Dr. Palmer.

OpenSubtitles (Open Domain) Experiment

- Dataset:
  - Movie subtitles transcript
  - Assume every sentence is a turn
  - Training: ~920M tokens
  - Validation: ~395M tokens
- Model:
  - Vocab: most frequent 100K words
  - 2-layer LSTM, 4096 cells
  - AdaGrad

OpenSubtitles Examples

Human: *my name is david . what is my name ?*
Machine: *david .*
Human: *my name is john . what is my name ?*
Machine: *john .*

OpenSubtitles Examples

Human: *my name is david . what is my name ?*
Machine: *david .*
Human: *my name is john . what is my name ?*
Machine: *john .*

OpenSubtitles Examples

Human: *when were you born ?*
Machine: *july 20th .*
Human: *what year were you born ?*
Machine: *1977 .*

OpenSubtitles Examples

Human: *when were you born ?*
Machine: *july 20th .*
Human: *what year were you born ?*
Machine: *1977 .*

OpenSubtitles Examples

Human: *is sky blue or black ?*
Machine: *blue .*
Human: *does a cat have a tail ?*
Machine: *yes .*

OpenSubtitles Examples

Human: *is sky blue or black ?*
Machine: *blue .*
Human: *does a cat have a tail ?*
Machine: *yes .*

OpenSubtitles Examples

Human: *what do you think about cleopatra ?*
Machine: *oh , she 's very regal .*
Human: *what do you think about england during the reign of elizabeth ?*
Machine: *it was a great place .*

OpenSubtitles Examples

Human: *what do you think about cleopatra ?*
Machine: *oh , she 's very regal .*
Human: *what do you think about england during the reign of elizabeth ?*
Machine: *it was a great place .*

OpenSubtitles Examples Take-away I

- **Pros**:
  - Remembering facts
  - Understanding concepts
  - Common sense reasoning
  - Generalization

OpenSubtitles Examples

Human: *what is your job ?*
Machine: *i 'm a lawyer .*
Human: *what do you do ?*
Machine: *i 'm a doctor .*

OpenSubtitles Examples

Human: *what is your job ?*
Machine: *i 'm a lawyer .*
Human: *what do you do ?*
Machine: *i 'm a doctor .*

OpenSubtitles Examples

Human: *what do you like to talk about ?*
Machine: *nothing .*

OpenSubtitles Examples

Human: *what do you like to talk about ?*
Machine: *nothing .*

OpenSubtitles Examples Take-away II

- **Cons**:
    - No coherent personality
    - Unsatisfying replies

Comparison with CleverBot

Human: *Are you a follower or a leader ?*
CleverBot: *No !!*
Human: *My name is david . what is my name ?*
CleverBot: *How old are you David ?*
Human: *My name is john . what is my name ?*
CleverBot: *John*

Comparison with CleverBot

Human: *Is sky blue or black ?*
CleverBot: *I am 200 years old.*
Human: *What is the color of a yellow car ?*
CleverBot: *Brown.*
Human: *What is the usual color of a leaf ?*
CleverBot: *What is the color of the apple in the fridge ?*

Evaluation: Perplexity

- 8 for IT Helpdesk (vs. 18 for n-gram model)
- 17 for OpenSubtitles (vs. 28 for smoothed 5-gram)

Human Evaluation

- four different humans rate proposed model vs. CleverBot

Human Evaluation

- four different humans rate proposed model vs. CleverBot
- 48.5% proposed model

Human Evaluation

- four different humans rate proposed model vs. CleverBot
- 48.5% proposed model

Human: What is the deepest spot on the world ?
NCM: The pacific ocean.
CB: What ?

Human Evaluation

- four different humans rate proposed model vs. CleverBot
- 48.5% proposed model
- 30% CleverBot

Human Evaluation

- four different humans rate proposed model vs. CleverBot
- 48.5% proposed model
- 30% CleverBot

Human: Do you like Mexican food or Indian food ?
NCM: I hate it.
CB: I love Mexican food.

Human Evaluation

- four different humans rate proposed model vs. CleverBot
- 48.5% proposed model
- 30% CleverBot
- 21.5 % tie or disagreement

Human Evaluation

- four different humans rate proposed model vs. CleverBot
- 48.5% proposed model
- 30% CleverBot
- 21.5 % tie or disagreement

Human: Who are you crazy about ?
NCM: You're the one who's crazy.
CB: Who are you ?

Designing a **good metric** to quickly measure the quality of a conversational model **remains an open question!**

Advantages

- **General** architecture, can be used in translation, Q&A, etc.
- **Simple** implementation
- Among the **first end-to-end neural network approaches** to dialogue generation

Limitations

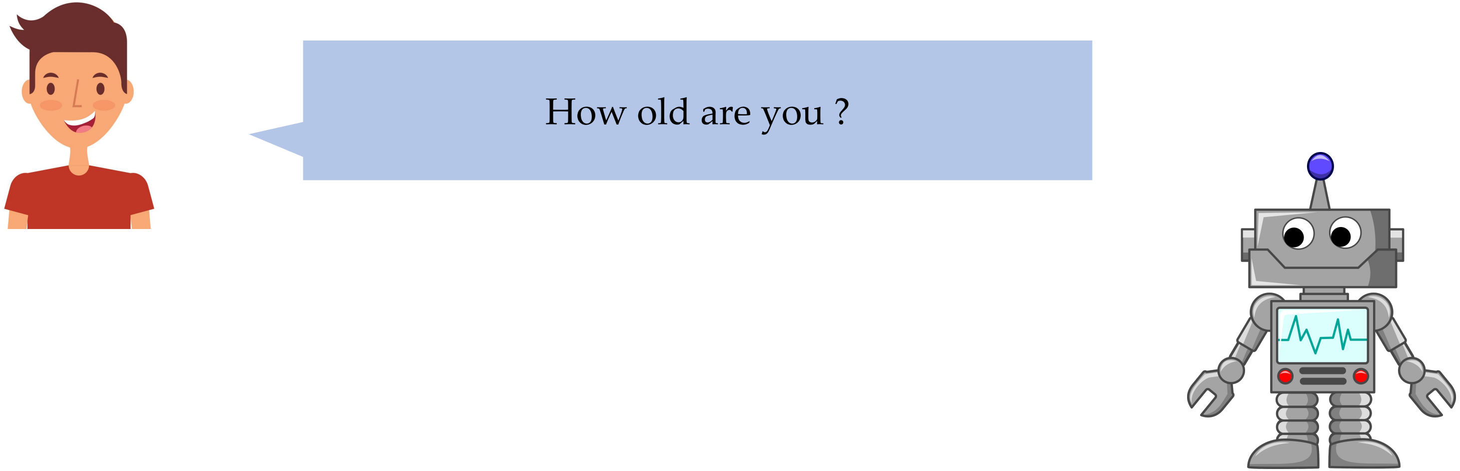
- Objective function doesn’t capture **human communication**
- Lacks **long-term** consistency
- Doesn’t measure the amount of **information** exchanged

Paper 2: Deep RL for Dialogue Generation

- Vanilla seq2seq gives **repetitive** or **boring** replies
- **MLE** may be a **bad objective** for approximating real-world goals of chatbots

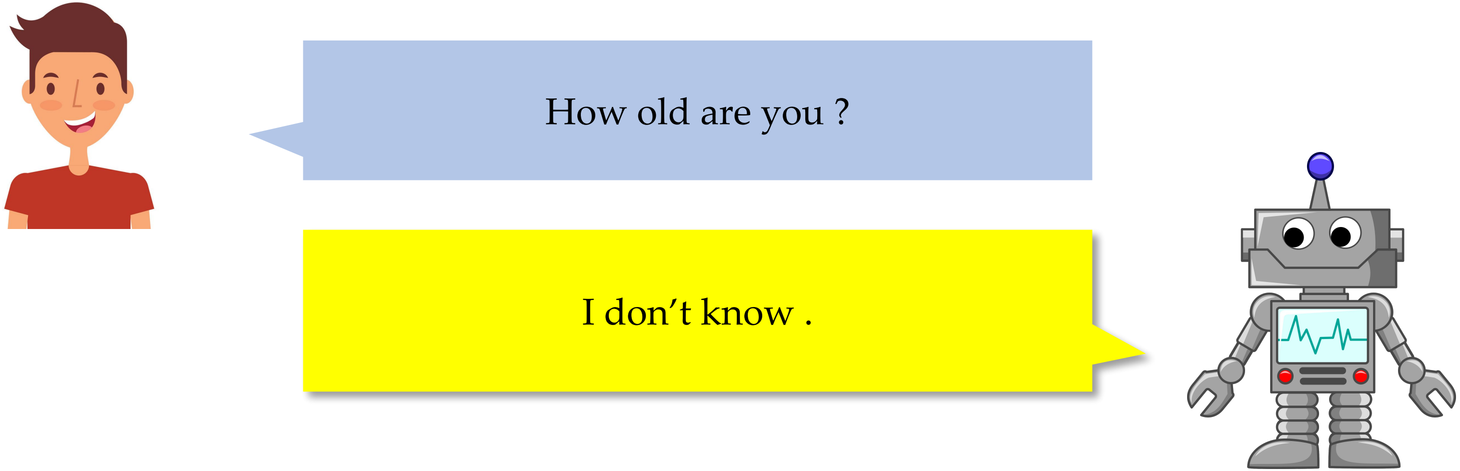
Problem 1: Dull and generic responses

“I don’t know“ problem (Sordoni et al., 2015; Serban et al.,2015;)



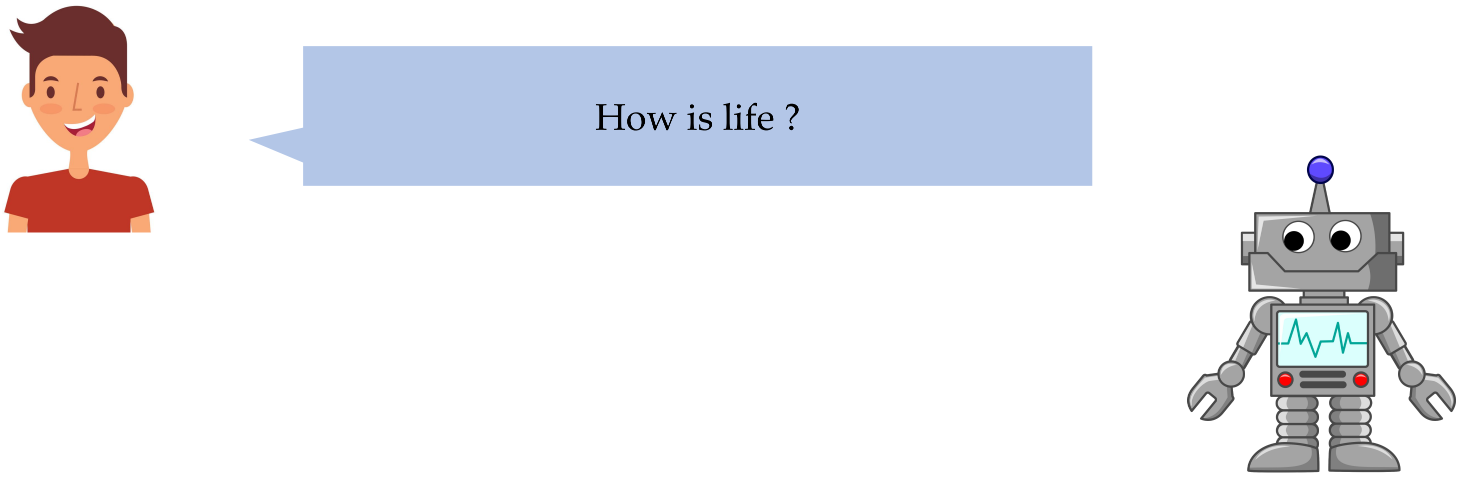
Problem 1: Dull and generic responses

"I don't know" problem (Sordoni et al., 2015; Serban et al.,2015;)



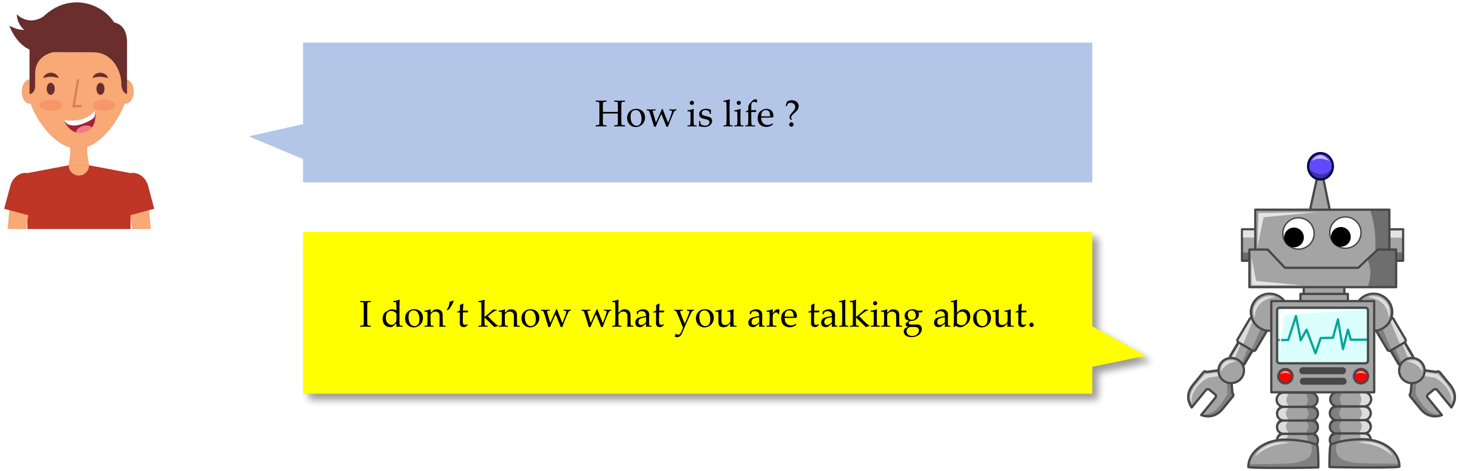
Problem 1: Dull and generic responses

“I don’t know“ problem (Sordoni et al., 2015; Serban et al.,2015;)

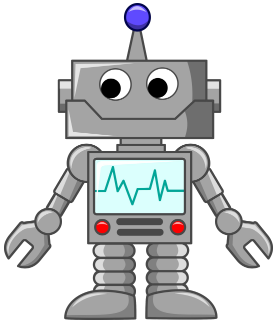


Problem 1: Dull and generic responses

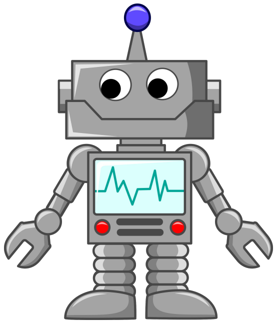
“I don’t know“ problem (Sordoni et al., 2015; Serban et al.,2015;)



Problem 2: Repetitive responses



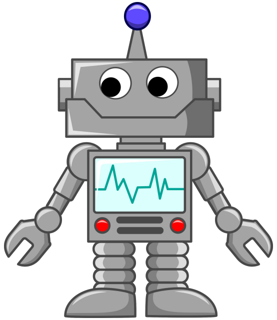
Shut up !

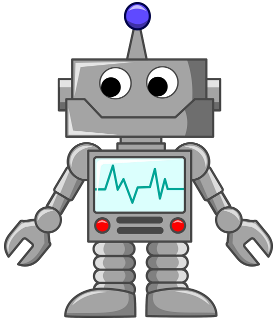


Problem 2: Repetitive responses

Shut up !

No, you shut up !

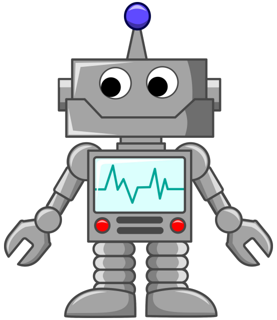




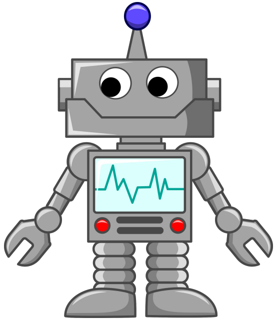
Problem 2: Repetitive responses

Shut up !

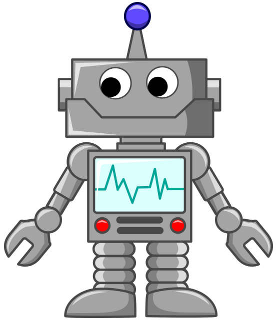
No, you shut up
!



No, you shut up !



Problem 2: Repetitive responses

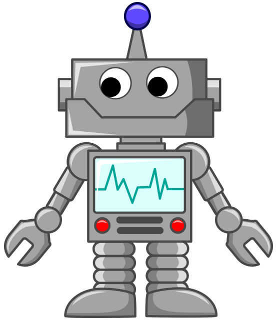


Shut up !

No, you shut up !

No, you shut up !

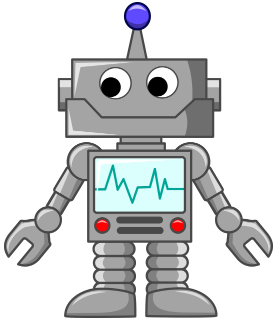
No, you shut up !



Problem 2: Repetitive responses

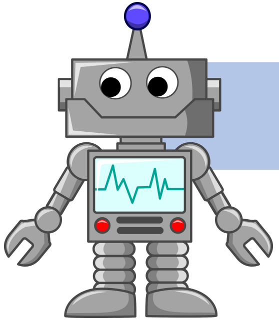
Shut up !

No, you shut up !



No, you shut up !

No, you shut up !

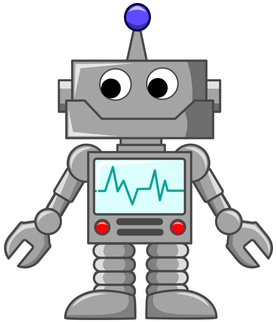


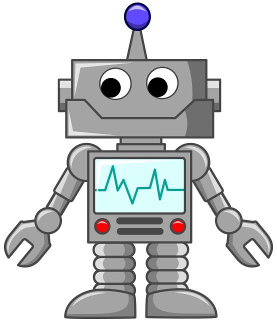
No, you shut up !

……

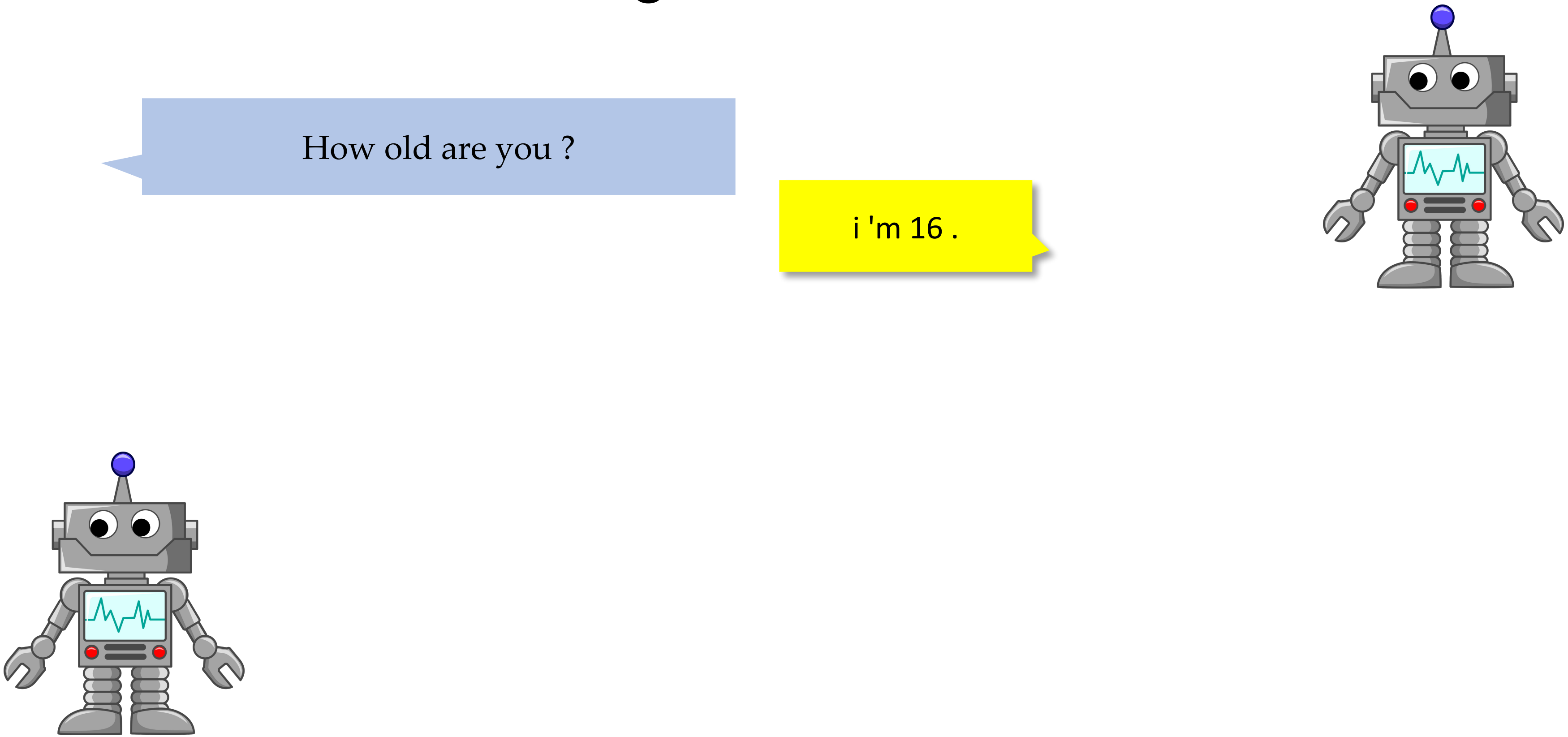
Problem 3: Short-sighted decisions

How old are you ?

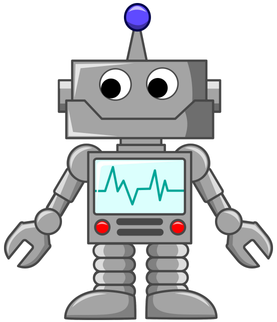




Problem 3: Short-sighted decisions



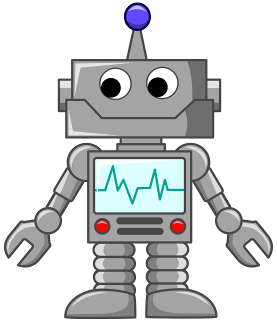
Problem 3: Short-sighted decisions



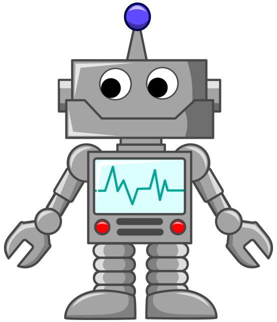
How old are you ?

i 'm 16 .

16 ?



Problem 3: Short-sighted decisions

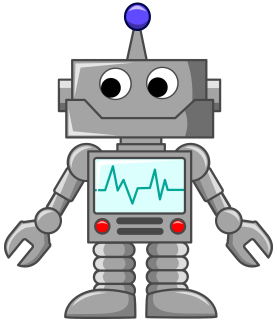


How old are you ?

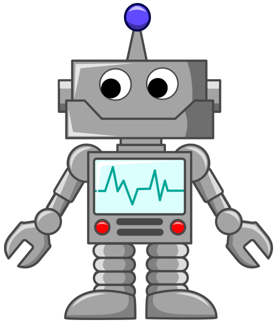
i 'm 16 .

16 ?

i don 't know what you 're talking about



Problem 3: Short-sighted decisions

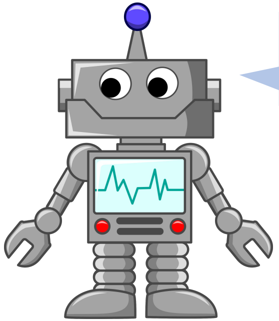


How old are you ?

i 'm 16 .

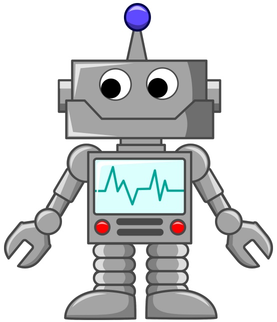
16 ?

i don 't know what you 're talking about



you don 't know what you 're saying

Problem 3: Short-sighted decisions

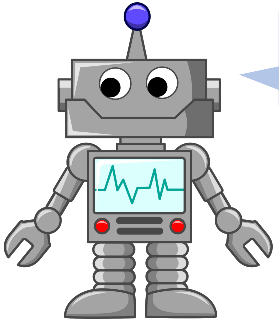


How old are you ?

i 'm 16 .

16 ?

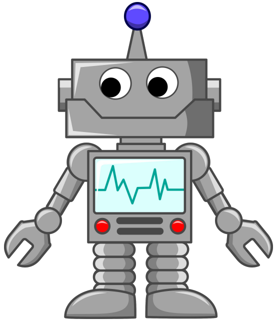
i don 't know what you 're talking about



you don 't know what you 're saying

i don 't know what you 're talking about

Problem 3: Short-sighted decisions



How old are you ?

i 'm 16 .

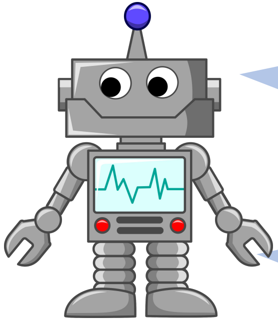
16 ?

i don 't know what you 're talking about

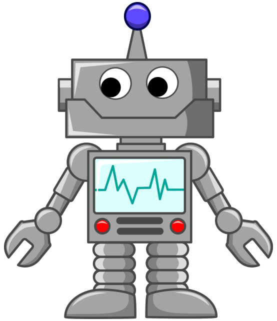
you don 't know what you 're saying

i don 't know what you 're talking about

you don 't know what you 're saying



Problem 3: Short-sighted decisions



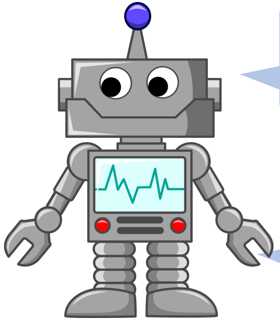
How old are you ?

Bad Action

i 'm 16 .

16 ?

i don 't know what you 're talking about



you don 't know what you 're saying

i don 't know what you 're talking about

you don 't know what you 're saying

Deep RL for Dialogue Generation

- Want to **reward interesting, diverse** and **informative replies**
- Need ability to **model future direction** of a dialogue
- **Goal**: **integrate** the **Seq2Seq** and **reinforcement learning paradigms**

Why RL?

- Powerful paradigm for **optimizing for long-term** goals across a conversation
- Ability to **integrate rewards** that better **mimic** the **true goal** of chatbot development
- **Overcome** the limitations of the **MLE** objective

RL Recap

- RL **problem formulation**
    - action space
    - state space
    - transition probability
    - reward function
    - time horizon
- **Goal**: learn a policy that maximizes the expected reward.
- **Policy**: a probability distribution over actions given a state

RL for Open-Domain Dialogue

- **Action**: the dialogue utterance to generate
- **State**: the previous two dialogue turns
- **Policy**: the parameters of the LSTM encoder-decoder
- **Reward**: computable reward function observed after agent reaches end of each sentence (more about this on next slides!)

Reward I: Ease of Answering

- Define set of dull responses S = { “I have no idea”, “I don’t know what you’re talking about”, … }
- Reward negative log likelihood of responding with a sentence in S

$$r_1 = -\frac{1}{N_{\mathbb{S}}} \sum_{s \in \mathbb{S}} \frac{1}{N_s} \log p_{\text{seq2seq}}(s|a)$$

- Hope: a model less likely to generate replies in S is also less likely to give other dull replies

Reward II: Information Flow

- Want semantic dissimilarity between consecutive terms of the same agent
- Reward negative log of cosine similarity

$$r_2 = -\log \cos(h_{p_i}, h_{p_{i+1}}) = -\log \cos \frac{h_{p_i} \cdot h_{p_{i+1}}}{\|h_{p_i}\| \|h_{p_{i+1}}\|}$$

- Hope: agent will contribute new info to keep dialogue moving and avoid repetitive sequences

Reward III: Semantic Coherence

- Want coherent and appropriate replies
- Model by mutual information between action a and previous turns
    - Train a backward seq2seq with sources and targets swapped
    - Scale probabilities by target length

$$r_3 = \frac{1}{N_a} \log p_{\text{seq2seq}}(a|q_i, p_i) + \frac{1}{N_{q_i}} \log p_{\text{seq2seq}}^{\text{backward}}(q_i|a)$$

Total Reward

- weighted sum of the 3 rewards discussed

$$r(a, [p_i, q_i]) = \lambda_1 r_1 + \lambda_2 r_2 + \lambda_3 r_3$$

$$\lambda_1 + \lambda_2 + \lambda_3 = 1.$$

- $\boldsymbol{\lambda_1}$ = 0.25 [*ease of answering*]
- $\boldsymbol{\lambda_2}$ = 0.25 [*information flow*]
- $\boldsymbol{\lambda_3}$ = 0.5 [*semantic coherence*]

Integrating the RL paradigm: Overview

- **Idea**: simulate two virtual agents taking turns conversing with each other
- **Stage 1**: Supervised Learning
  - Seq2Seq with attention on OpenSubtitles
  - 80M source-target pairs
  - Target: each turn
  - Source: concatenate two previous sentences

Integrating the RL paradigm: Overview

- **Stage 2:** Maximize the Mutual Information
  - Treat as an RL problem
  - Training: MIXER, a form of curriculum learning
- **Stage 3**: Train the Policy
  - Simulation: explore the state-action space
  - Optimization
  - Curriculum learning

Mutual Information Model

- Motivation: **promote diversity** for better exploration
- Treat max mutual information task as an **RL problem** with **mutual information** as **reward**
- Recall that the policy is **initialized** with parameters of a vanilla **Seq2Seq** model

MIM Training

- Use policy gradient
- For each state, generate a list of actions from policy

$$A = \{\hat{a} | \hat{a} \sim p_{RL}\}$$

- For each generated action, obtain mutual information score as reward

$$\frac{1}{N_a} \log p_{\text{seq2seq}}(a|q_i, p_i) + \frac{1}{N_{q_i}} \log p_{\text{seq2seq}}^{\text{backward}}(q_i|a)$$

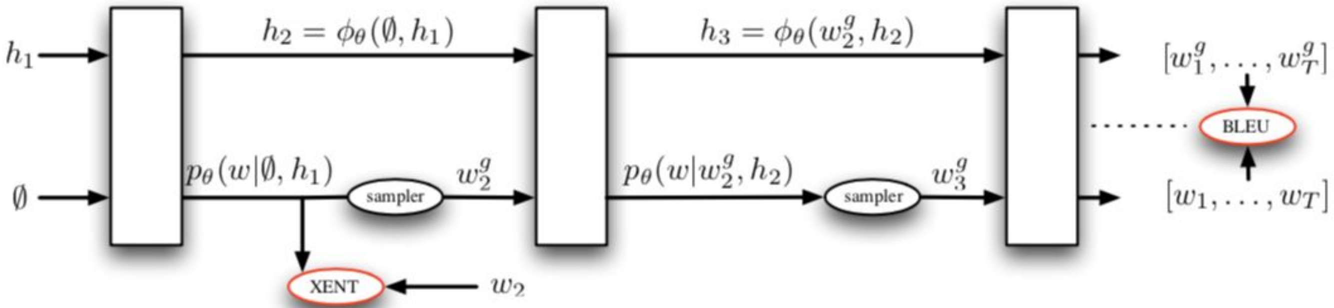
MIM Training

- Reward: $J(\theta) = \mathbb{E}[m(\hat{a}, [p_i, q_i])]$
- Gradient: $\nabla J(\theta) = m(\hat{a}, [p_i, q_i])\nabla \log p_{RL}(\hat{a}|[p_i, q_i])$
- Add baseline to reduce variance

$$\nabla J(\theta) = \nabla \log p_{RL}(\hat{a}|[p_i, q_i])[m(\hat{a}, [p_i, q_i]) - b]$$

- Use curriculum learning: MIXER

MIXER



Data: a set of sequences with their corresponding context.
Result: RNN optimized for generation.
Initialize RNN at random and set N^{XENT}, $N^{\text{XE+R}}$ and Δ;
for $s = T, 1, -\Delta$ **do**
  if $s == T$ **then**
    train RNN for N^{XENT} epochs using XENT only;
  else
    train RNN for $N^{\text{XE+R}}$ epochs. Use XENT loss in the first s steps, and REINFORCE (sampling from the model) in the remaining $T - s$ steps;
  end
end

REINFORCE

- Loss:

$$L_\theta = -\sum_{w_1^g,\ldots,w_T^g} p_\theta(w_1^g,\ldots,w_T^g)\, r(w_1^g,\ldots,w_T^g) = -\mathbb{E}_{[w_1^g,\ldots w_T^g]\sim p_\theta}\, r(w_1^g,\ldots,w_T^g)$$

$$m(\hat{a}, [p_i, q_i])$$

- Gradient $\dfrac{\partial L_\theta}{\partial \theta} = \sum_t \dfrac{\partial L_\theta}{\partial \mathbf{o}_t}\dfrac{\partial \mathbf{o}_t}{\partial \theta}$

Estimated using a NN

$$\frac{\partial L_\theta}{\partial \mathbf{o}_t} = \left(r(w_1^g,\ldots,w_T^g) - \bar{r}_{t+1}\right)\left(p_\theta(w_{t+1}|w_t^g, \mathbf{h}_{t+1}, \mathbf{c}_t) - \mathbf{1}(w_{t+1}^g)\right)$$

Generated words

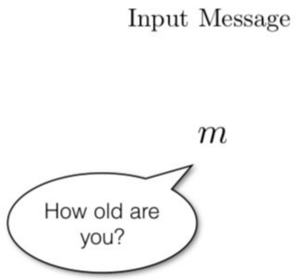
- Compared to cross-entropy:

$$\frac{\partial L_\theta^{\text{XENT}}}{\partial \mathbf{o}_t} = p_\theta(w_{t+1}|w_t, \mathbf{h}_{t+1}, \mathbf{c}_t) - \mathbf{1}(w_{t+1})$$

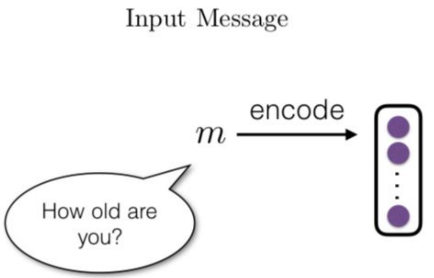
Final Policy Training

- Initialize agents with model after Stage 2
- Dialogue Simulation
- maximize expected future reward

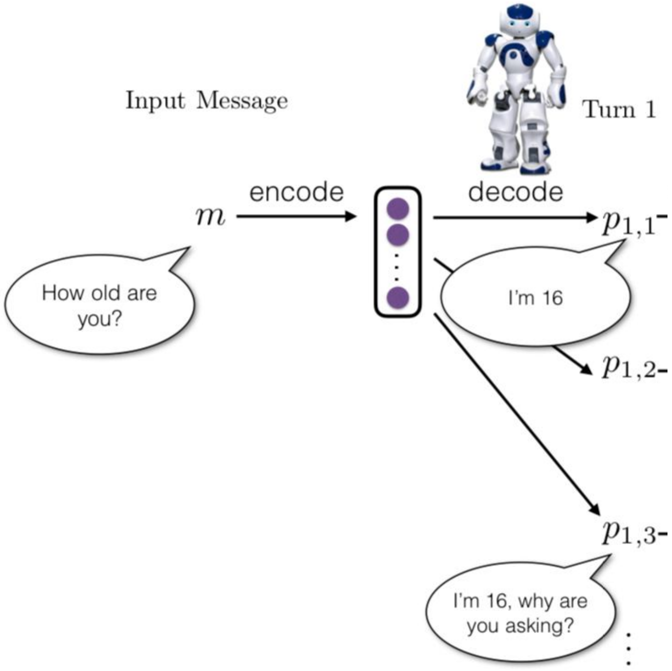
Visualization of Dialogue Simulation Procedure



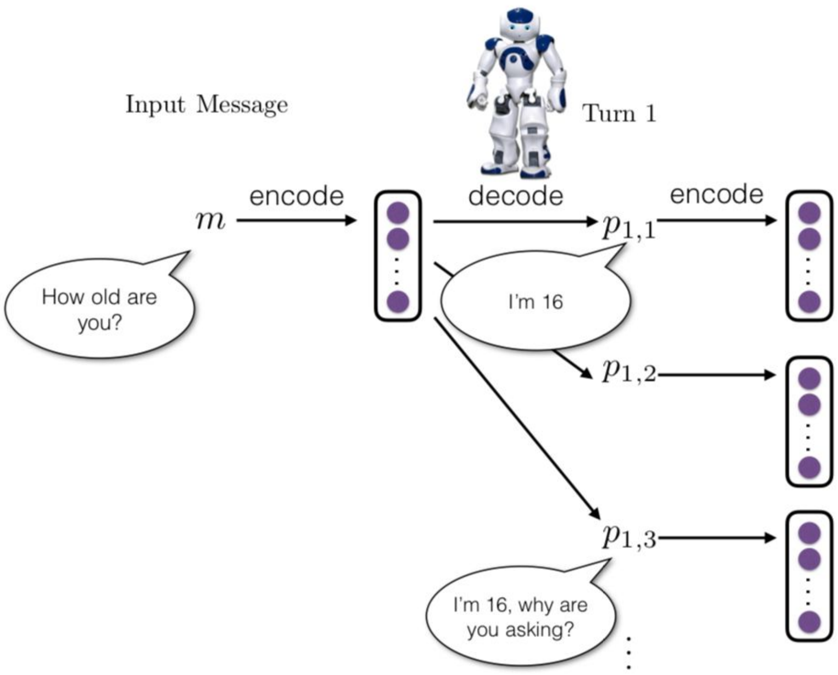
Visualization of Dialogue Simulation Procedure



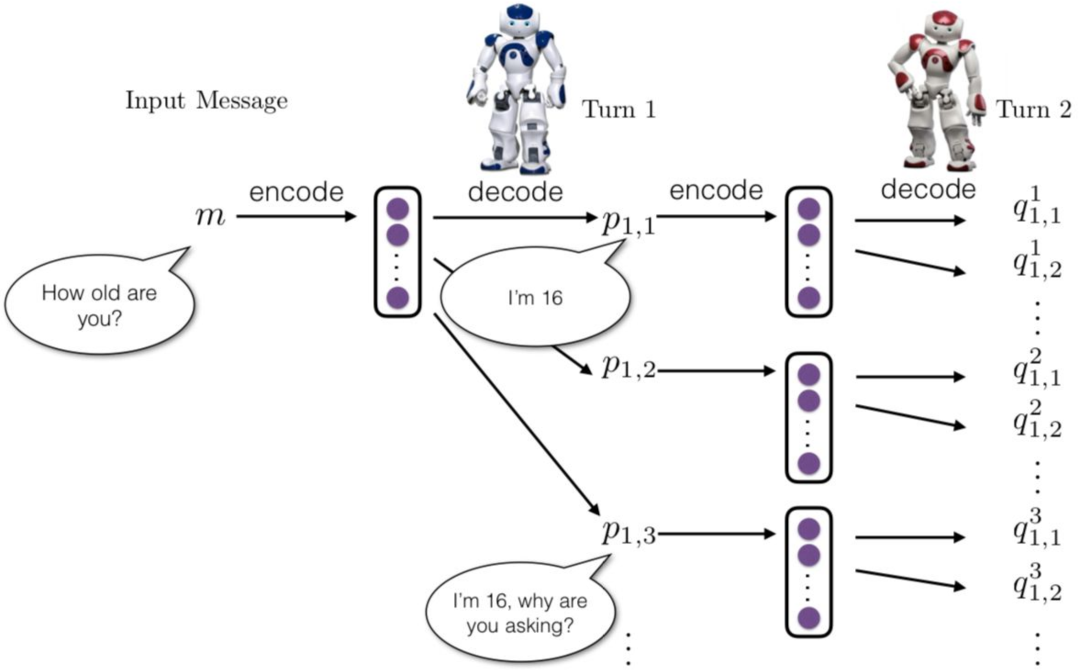
Visualization of Dialogue Simulation Procedure



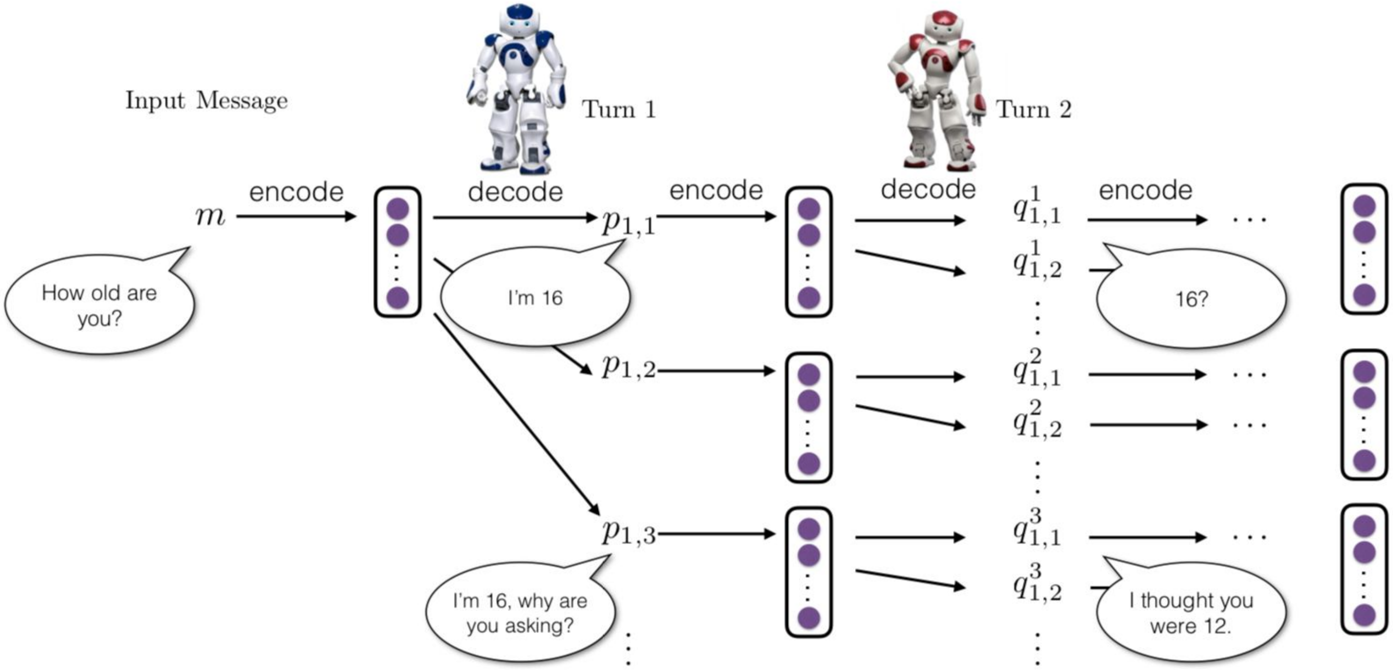
Visualization of Dialogue Simulation Procedure



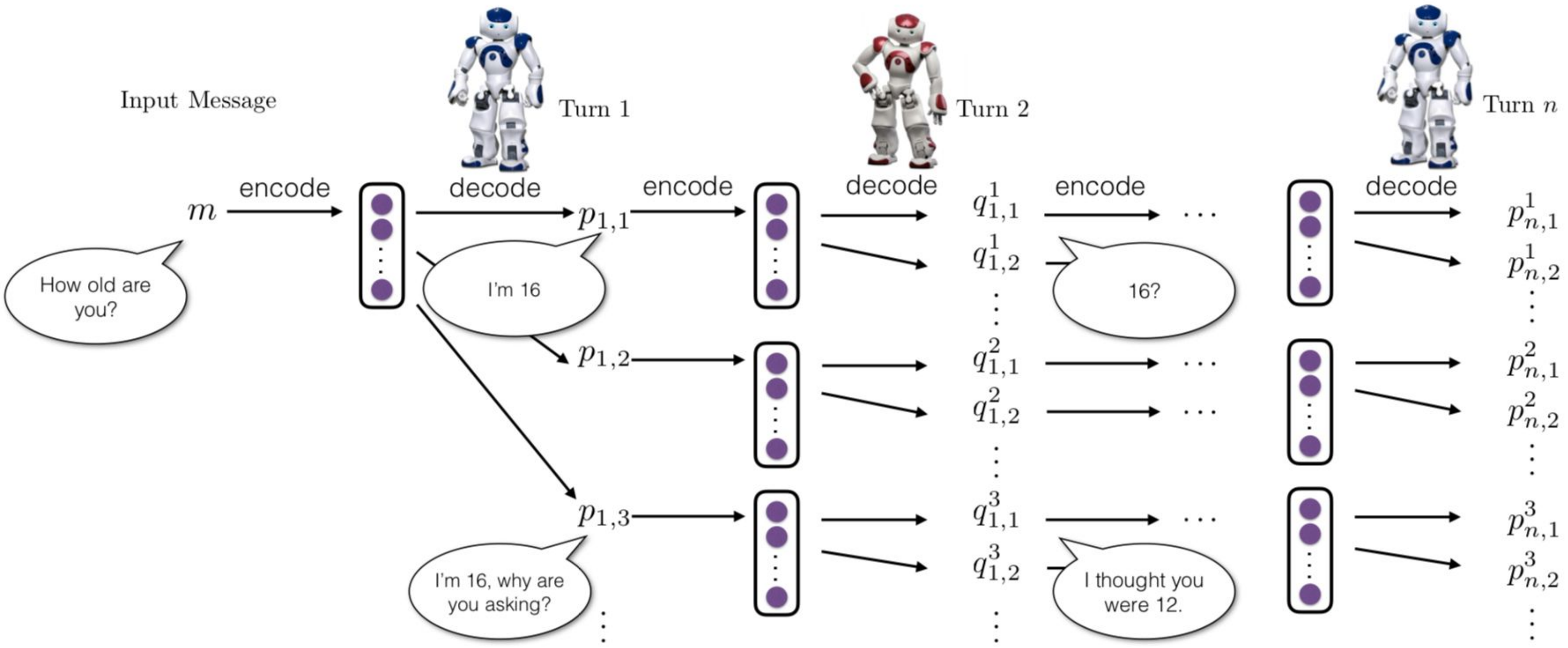
Visualization of Dialogue Simulation Procedure



Visualization of Dialogue Simulation Procedure



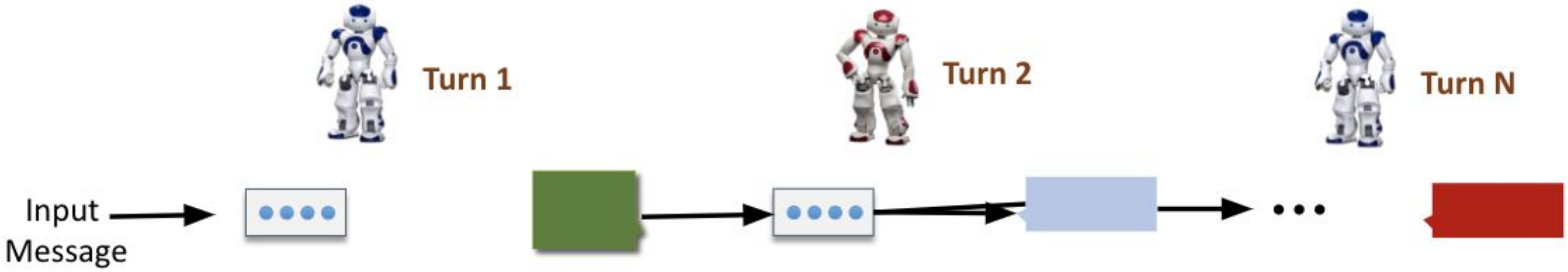
Visualization of Dialogue Simulation Procedure

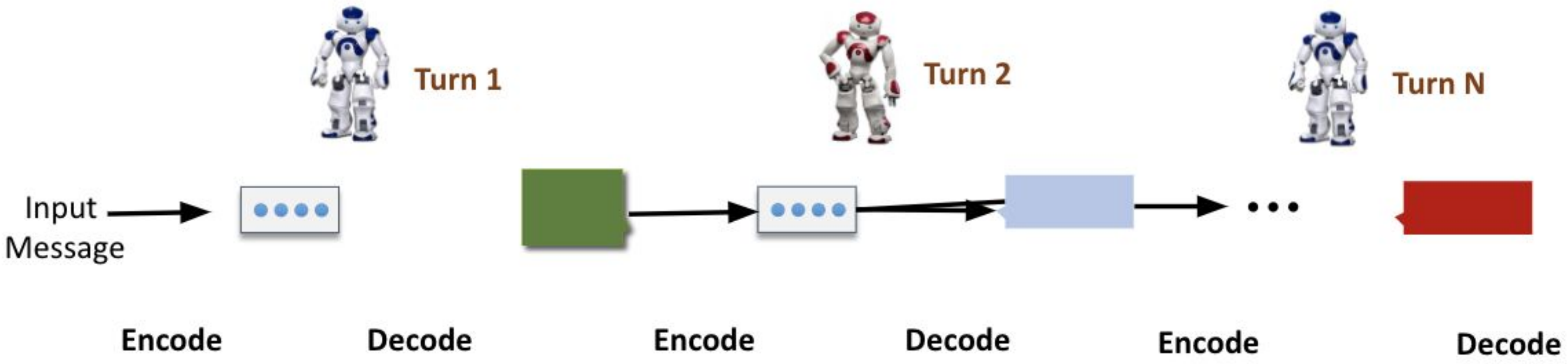


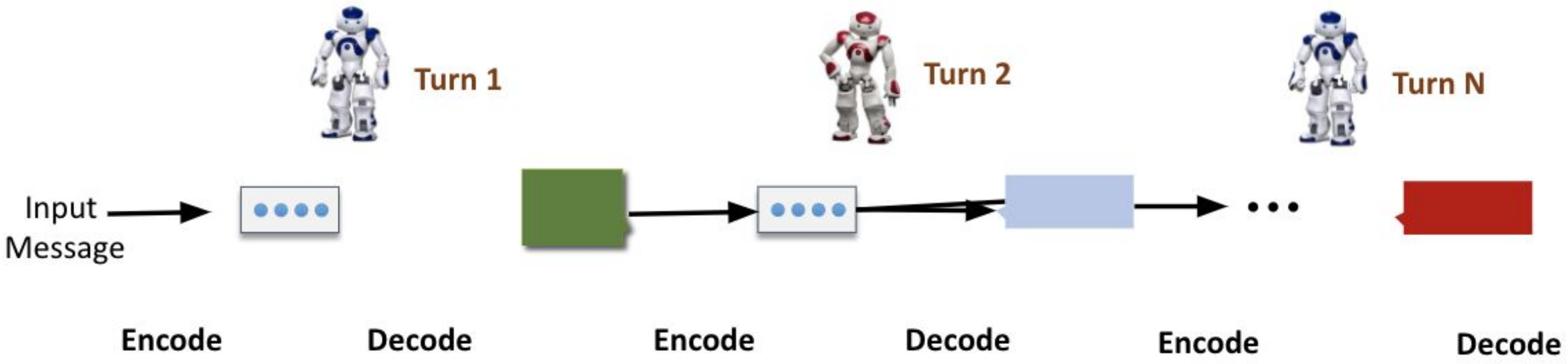
Notes on Dialogue Simulation: Curriculum Learning

- 5 candidate responses generated at each step of the simulation
- simulate the dialogue for 2 turns at first
- gradually increase the number of simulated turns (up to 5)

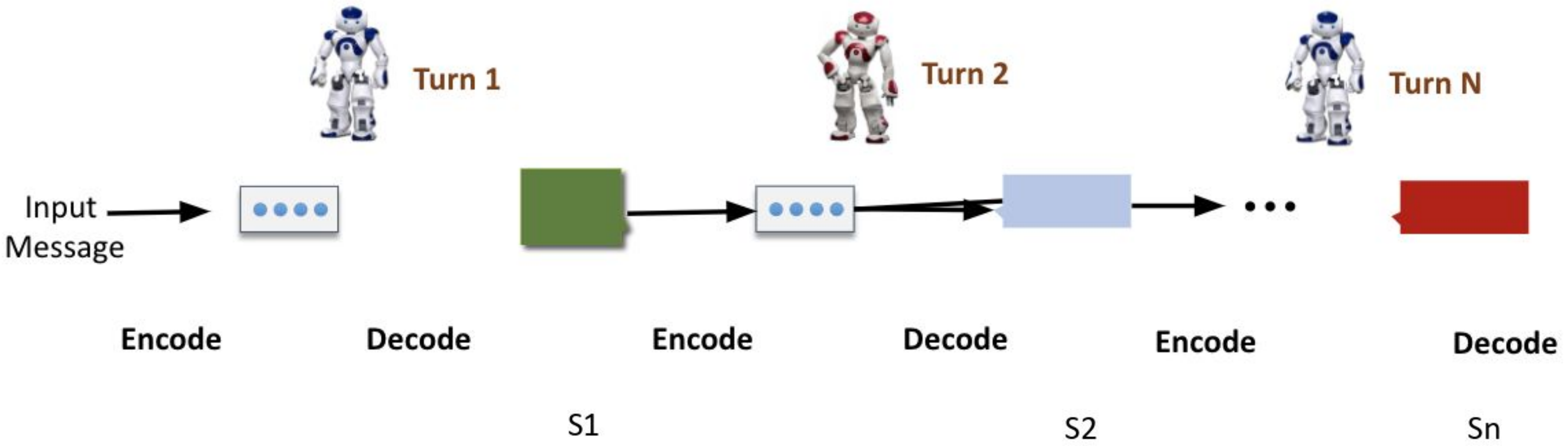
Visualization of Training Procedure



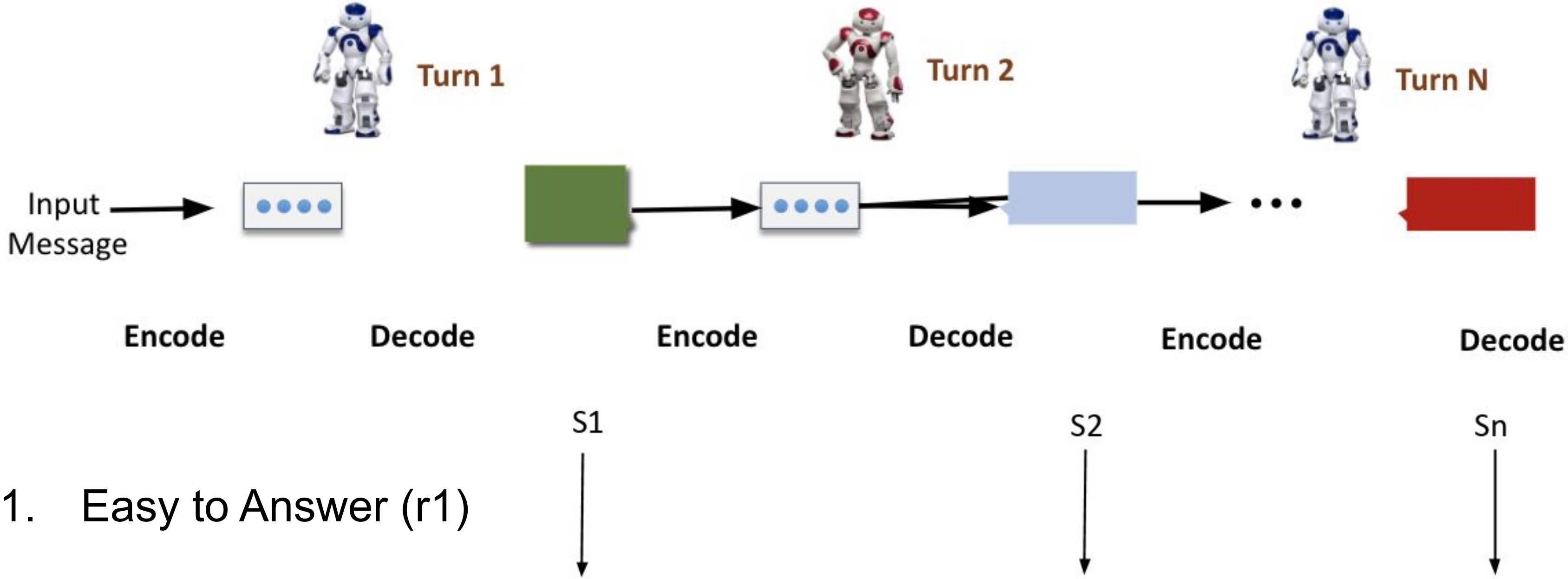

Turn 1
Turn 2
Turn N
Input
Message
Encode
Decode
Encode
Decode
Encode
Decode



Compute Accumulated Reward R(S1, …, Sn) !

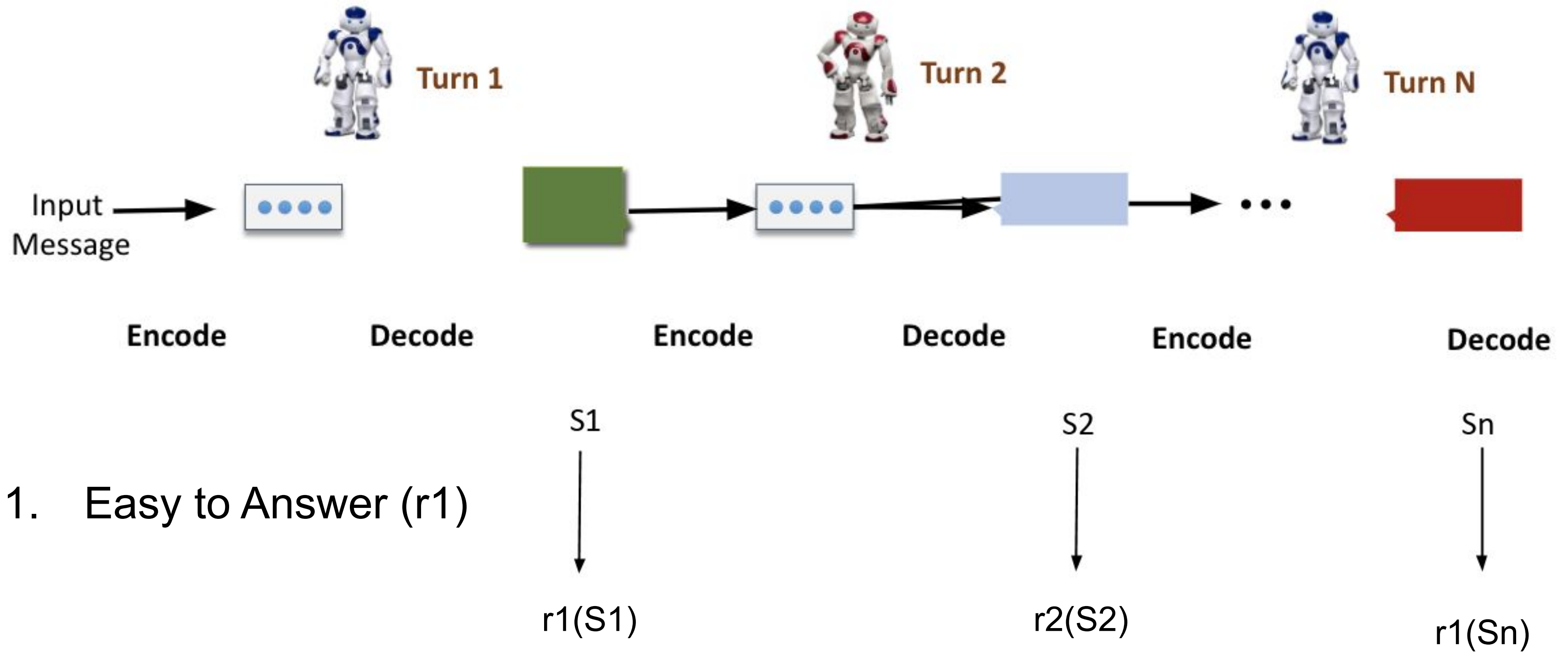


Compute Accumulated Reward R(S1, …, Sn) !



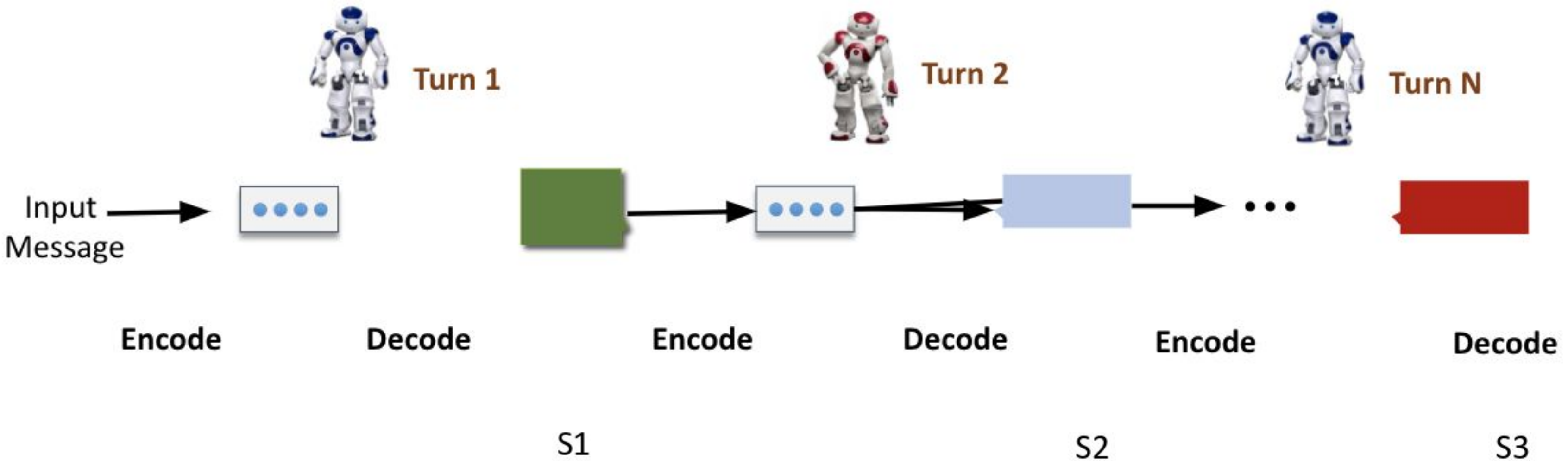
1. Easy to Answer (r1)

Compute Accumulated Reward R(S1, …, Sn) !



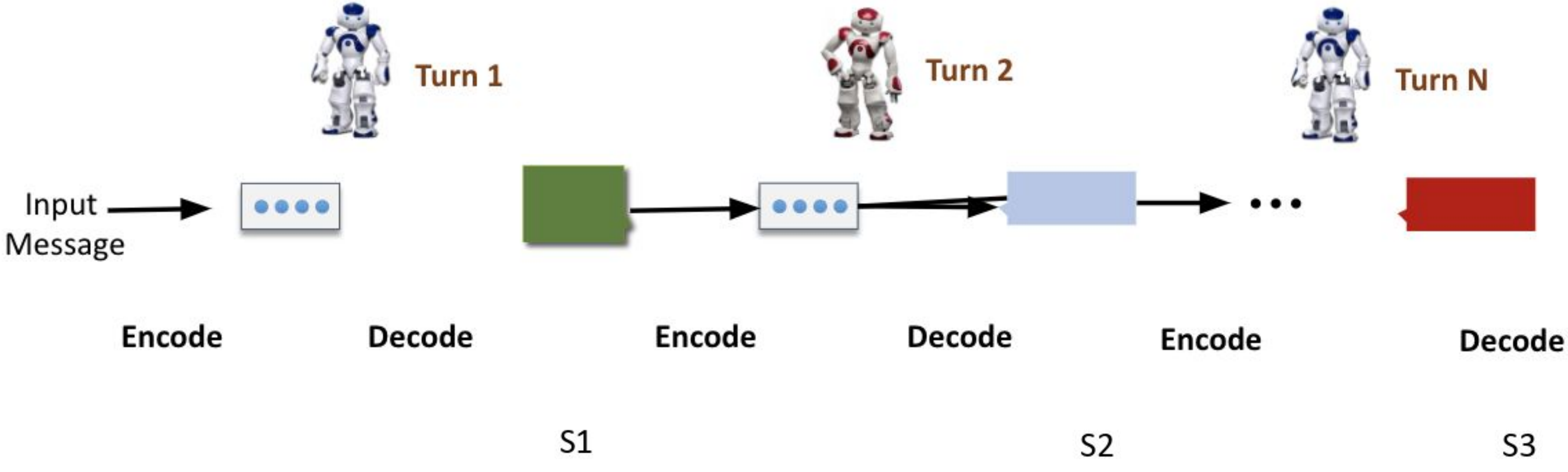
1. Easy to Answer (r1)

Compute Accumulated Reward R(S1, …, Sn) !



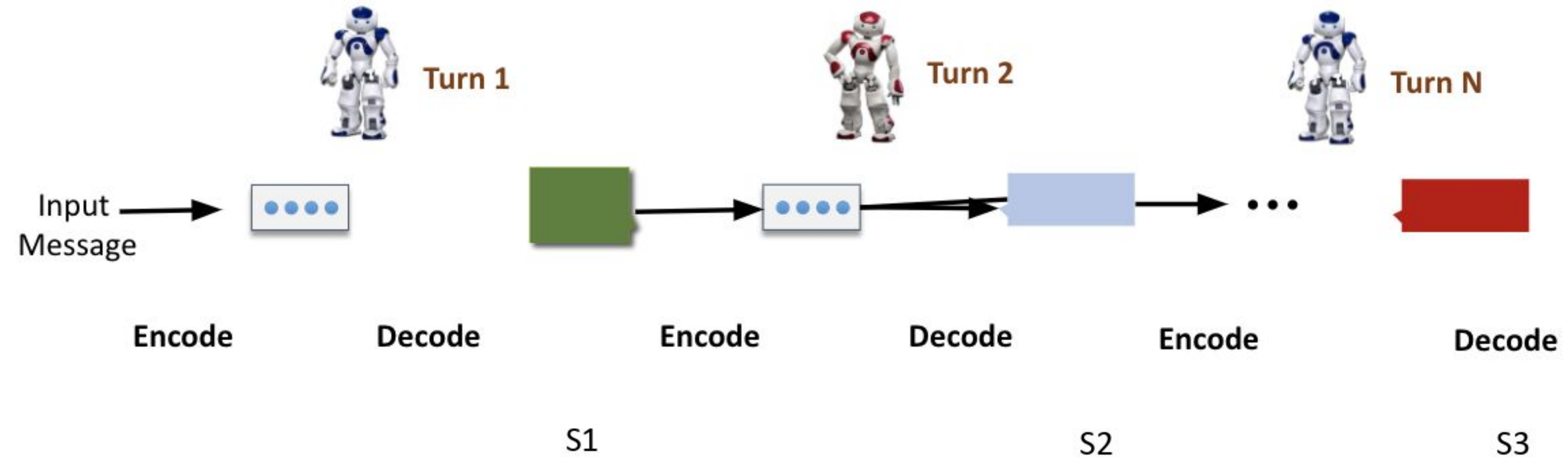
1. Easy to Answer (r1)
2. Information Flow (r2)

Compute Accumulated Reward R(S1, …, Sn) !



1. Easy to Answer (r1)
2. Information Flow (r2)

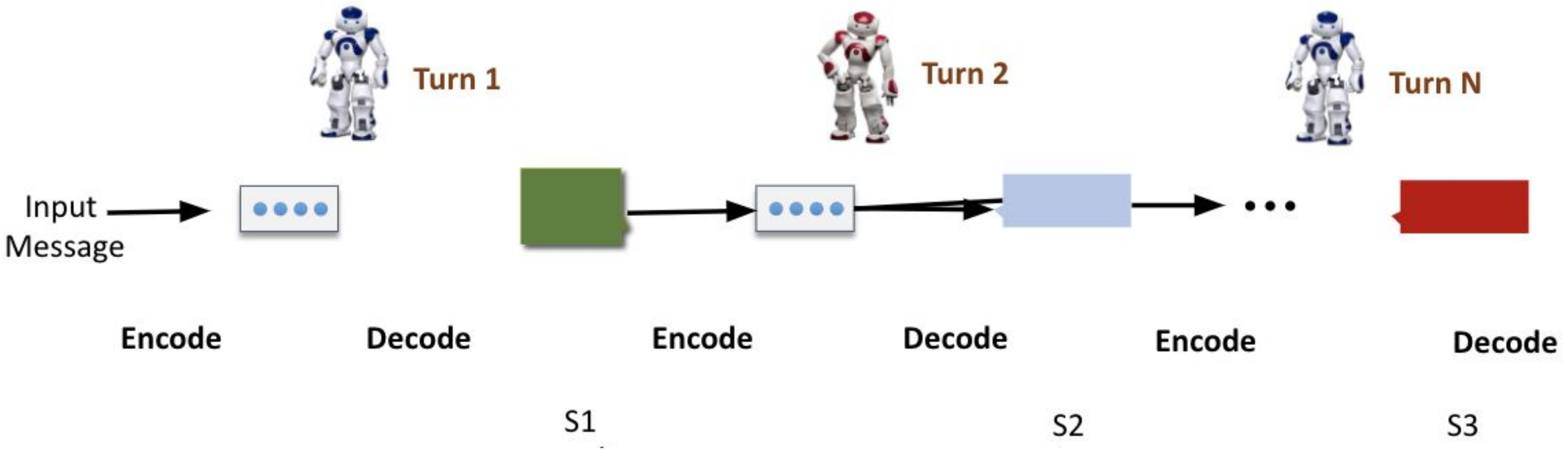
Compute Accumulated Reward R(S1, …, Sn) !



1. Easy to Answer (r1)
2. Information Flow (r2)

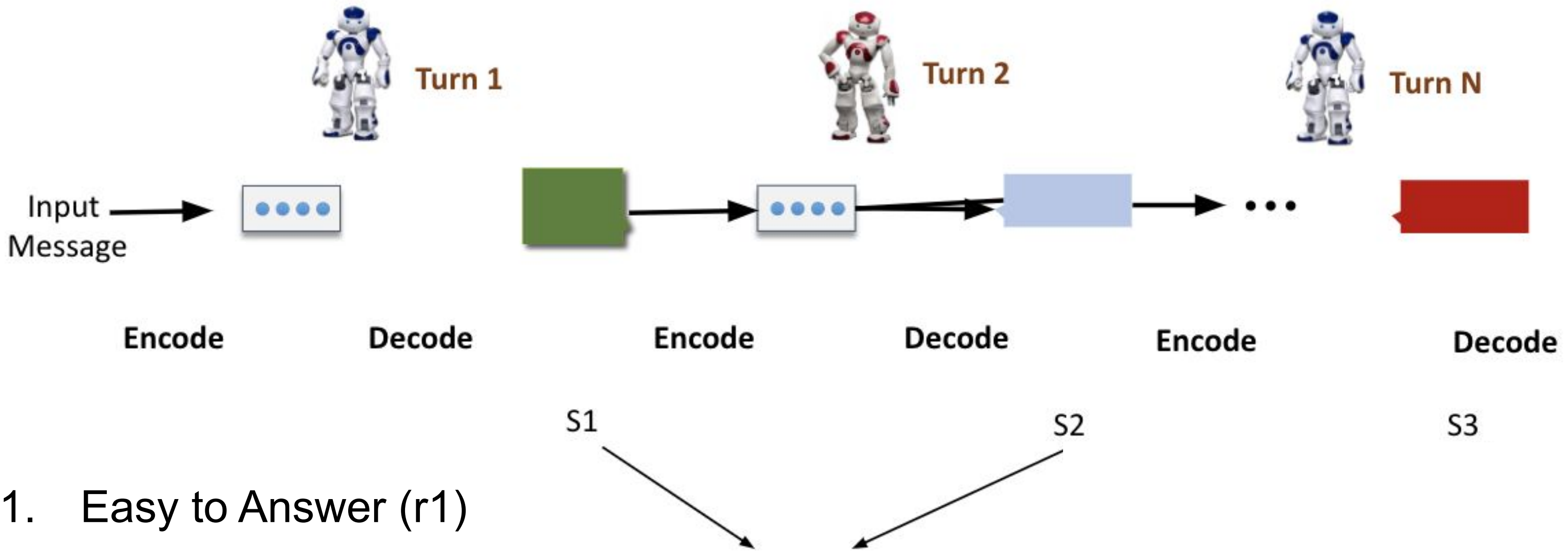
r2(S1, S3)

Compute Accumulated Reward R(S1, …, Sn) !



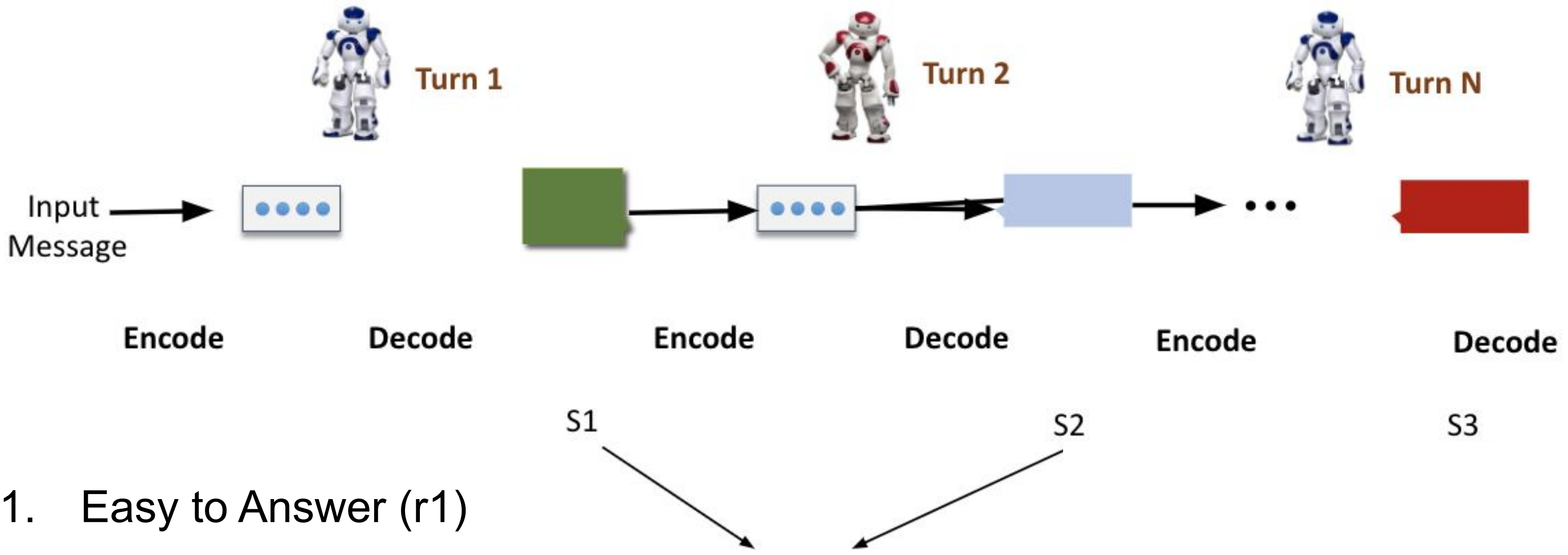
1. Easy to Answer (r1)
2. Information Flow (r2)
3. Semantic Coherence (r3)

Compute Accumulated Reward R(S1, …, Sn) !



1. Easy to Answer (r1)
2. Information Flow (r2)
3. Semantic Coherence (r3)

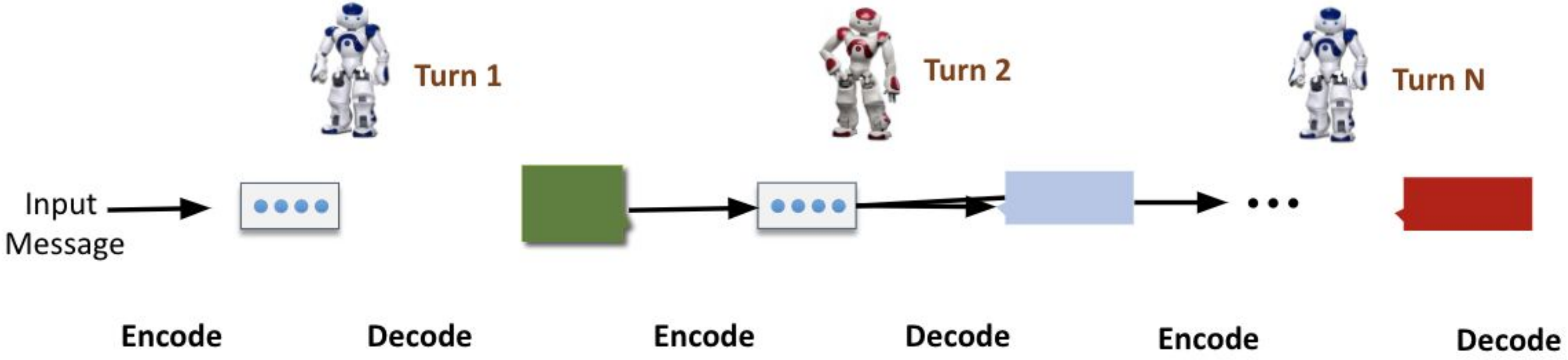
Compute Accumulated Reward R(S1, …, Sn) !

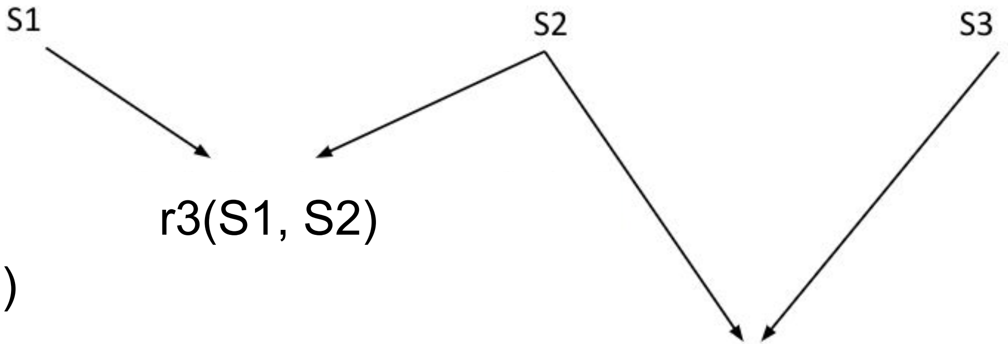


1. Easy to Answer (r1)
2. Information Flow (r2)
3. Semantic Coherence (r3)

r3(S1, S2)

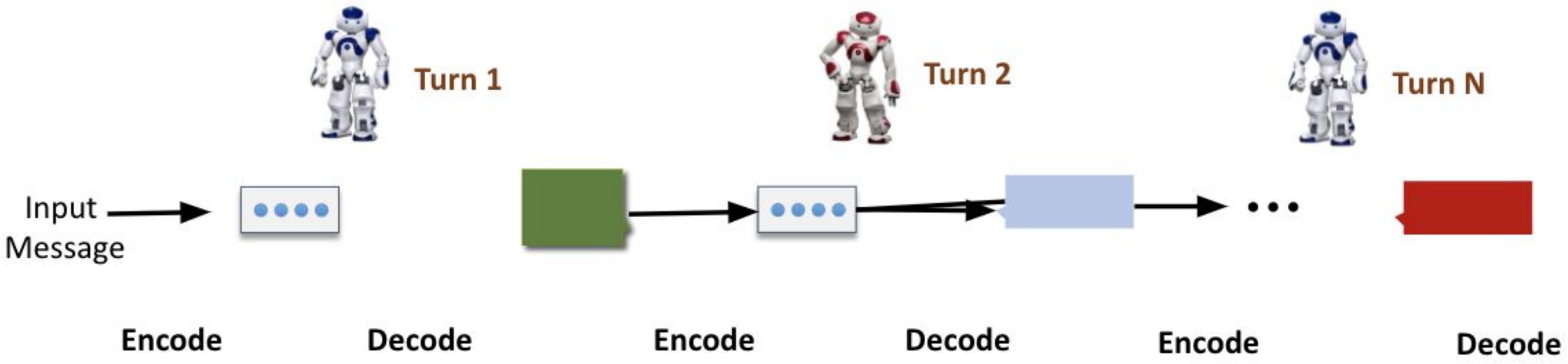
Compute Accumulated Reward R(S1, …, Sn) !

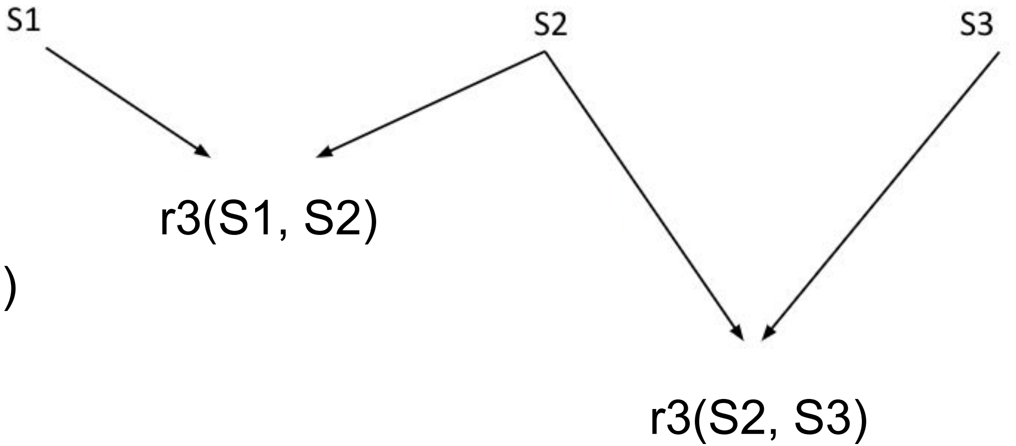




1. Easy to Answer (r1)
2. Information Flow (r2)
3. Semantic Coherence (r3)

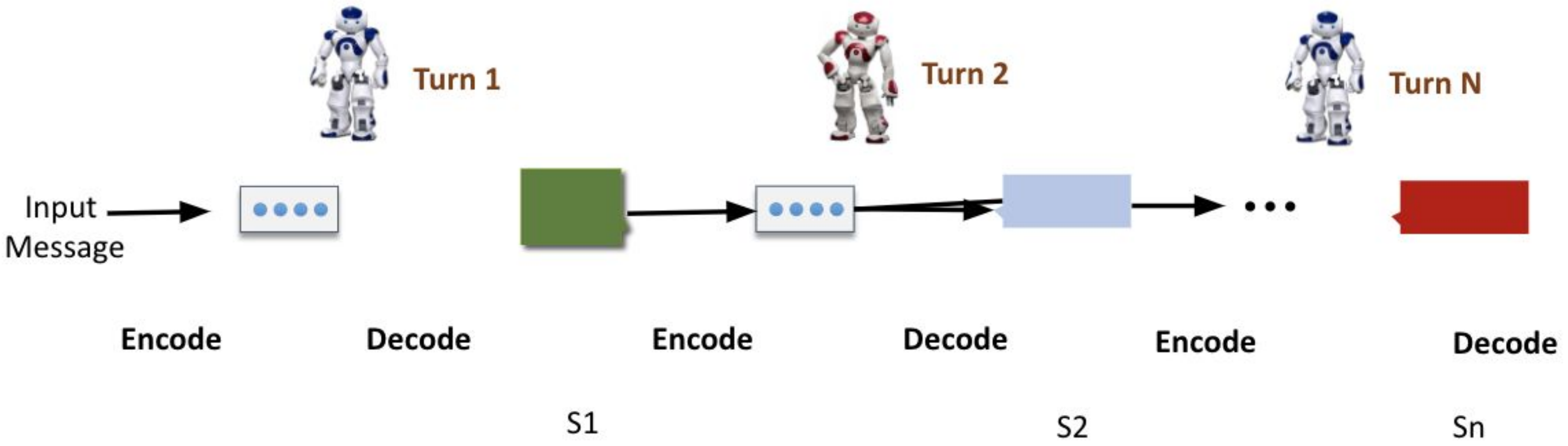
Compute Accumulated Reward R(S1, …, Sn) !





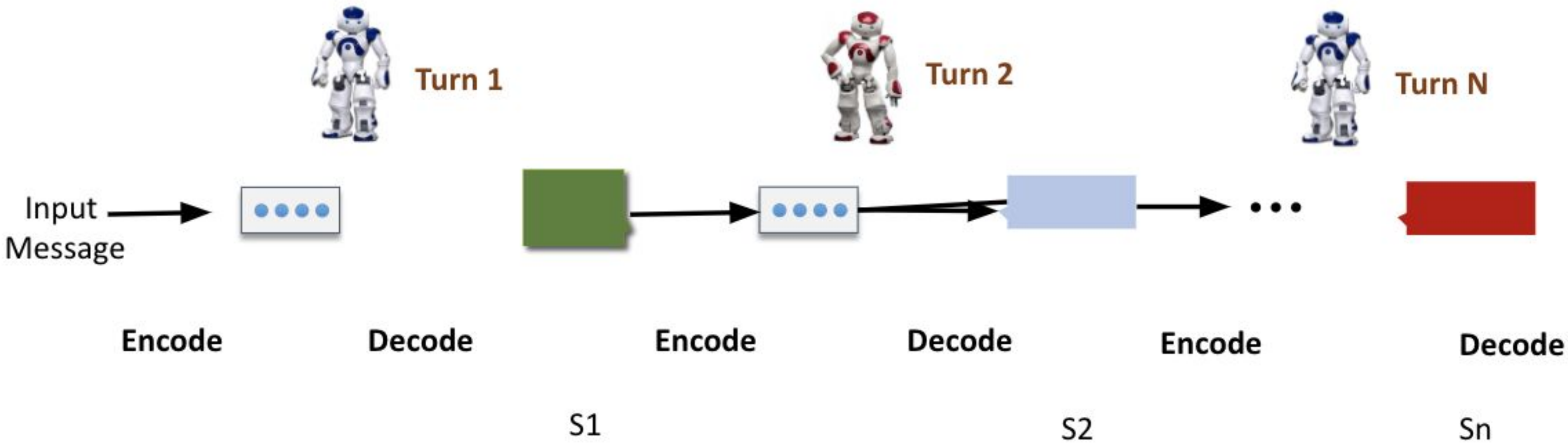
1. Easy to Answer (r1)
2. Information Flow (r2)
3. Semantic Coherence (r3)

Compute Accumulated Reward R(S1, …, Sn) !



REINFORCE Algorithm (William,1992)

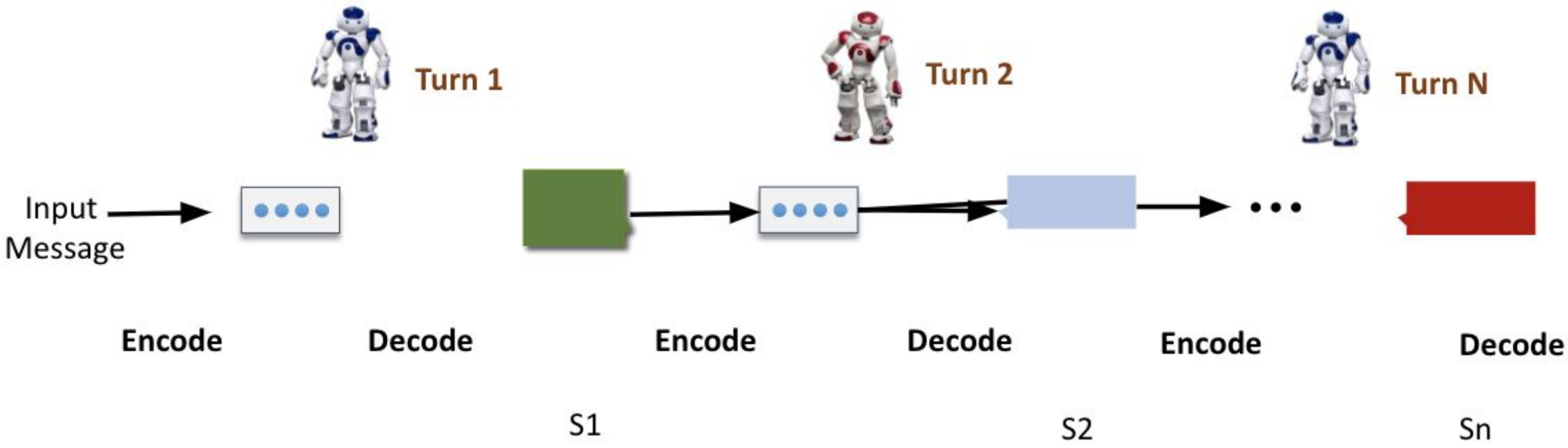
$$J(\theta) = \mathbb{E}[R(s_1, s_2, ..., s_N)]$$



REINFORCE Algorithm (William,1992)

$$J(\theta) = \mathbb{E}[R(s_1, s_2, ..., s_N)]$$

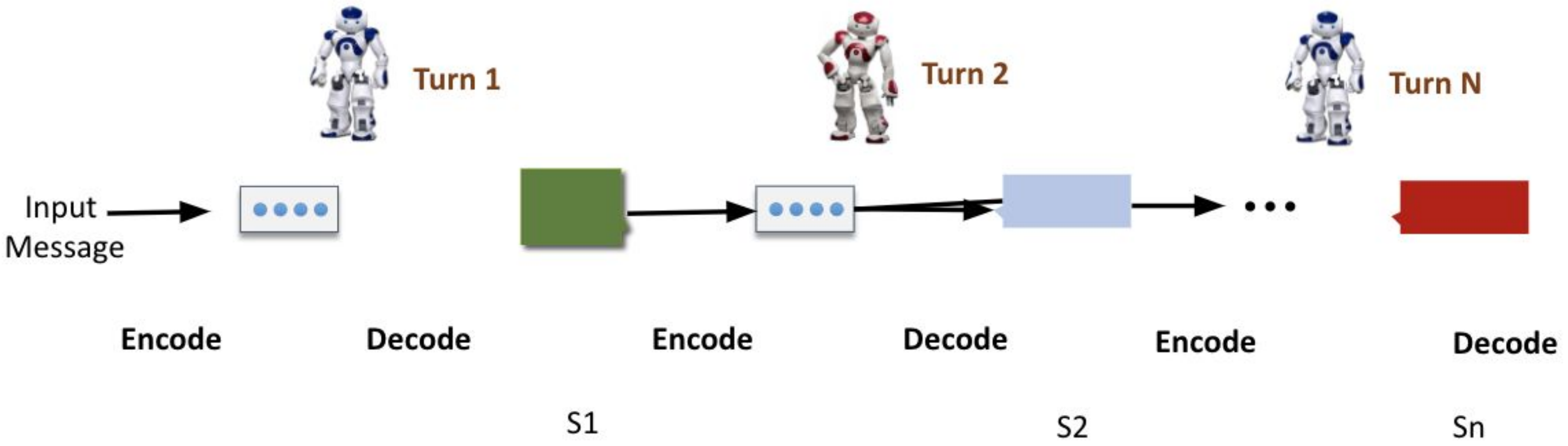
$$\nabla J(\theta) = \nabla \log p(s_1, s_2, ..., s_N) R(s_1, s_2, ..., s_N)$$



REINFORCE Algorithm (William,1992)

$$J(\theta) = \mathbb{E}[R(s_1, s_2, ..., s_N)]$$

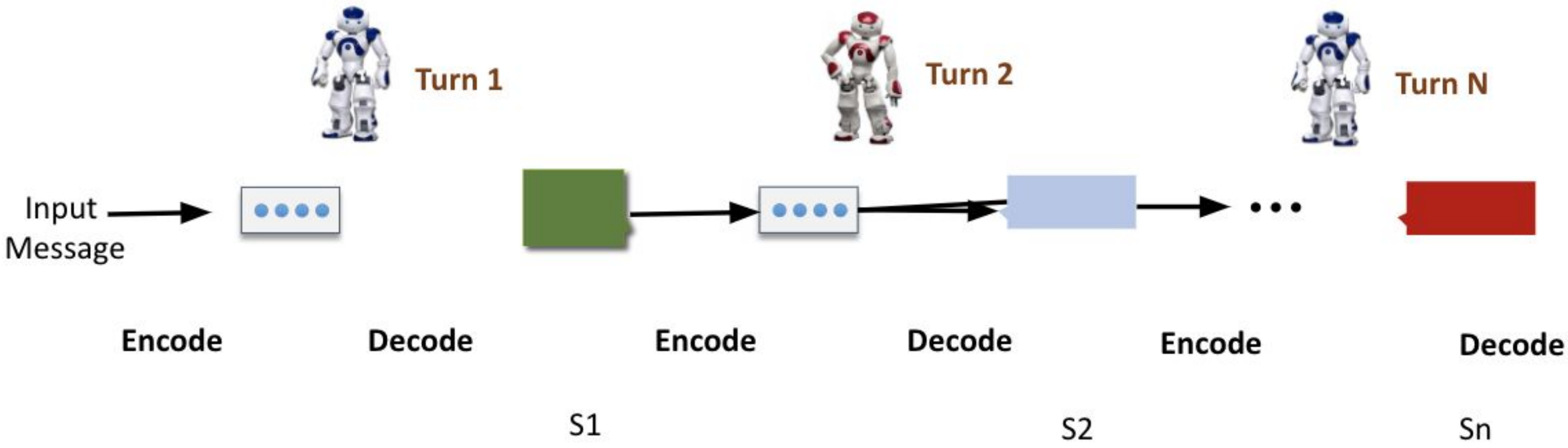
$$\nabla J(\theta) = \nabla \boxed{\log p(s_1, s_2, ..., s_N)} R(s_1, s_2, ..., s_N)$$



REINFORCE Algorithm (William,1992)

$$J(\theta) = \mathbb{E}[R(s_1, s_2, ..., s_N)]$$

$$\nabla J(\theta) = \nabla \log p(s_1, s_2, ..., s_N) \boxed{R(s_1, s_2, ..., s_N)}$$

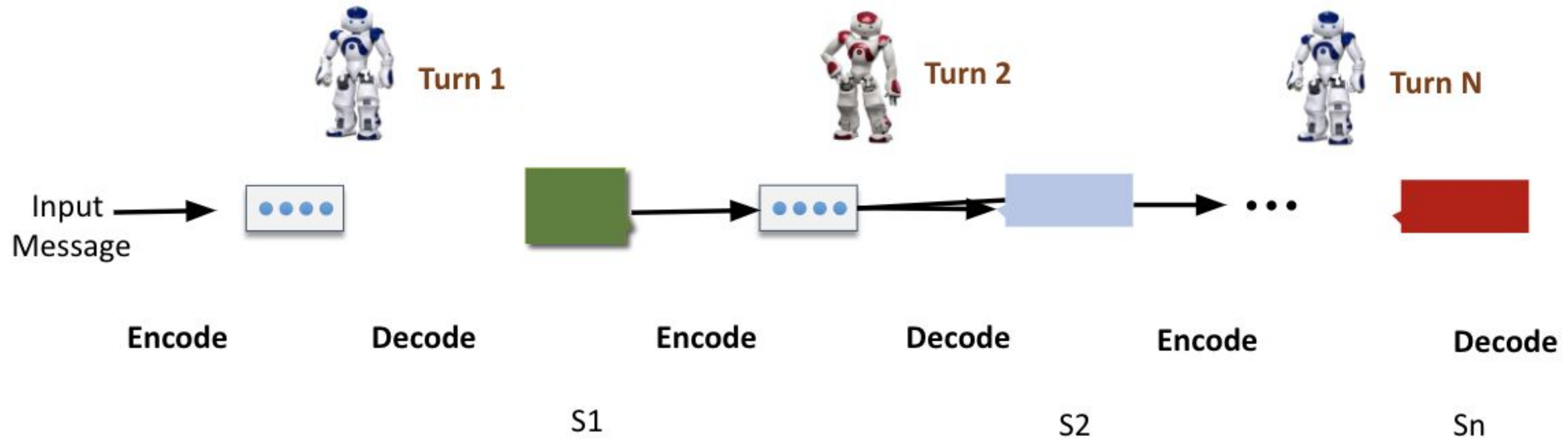


REINFORCE Algorithm (William,1992)

$$J(\theta) = \mathbb{E}[R(s_1, s_2, ..., s_N)]$$

$$\nabla J(\theta) = \nabla \log p(s_1, s_2, ..., s_N) R(s_1, s_2, ..., s_N)$$

$$\nabla J(\theta) = \nabla \log \prod_i p(s_i | s_{i-1}) R(s_1, s_2, ..., s_N)$$

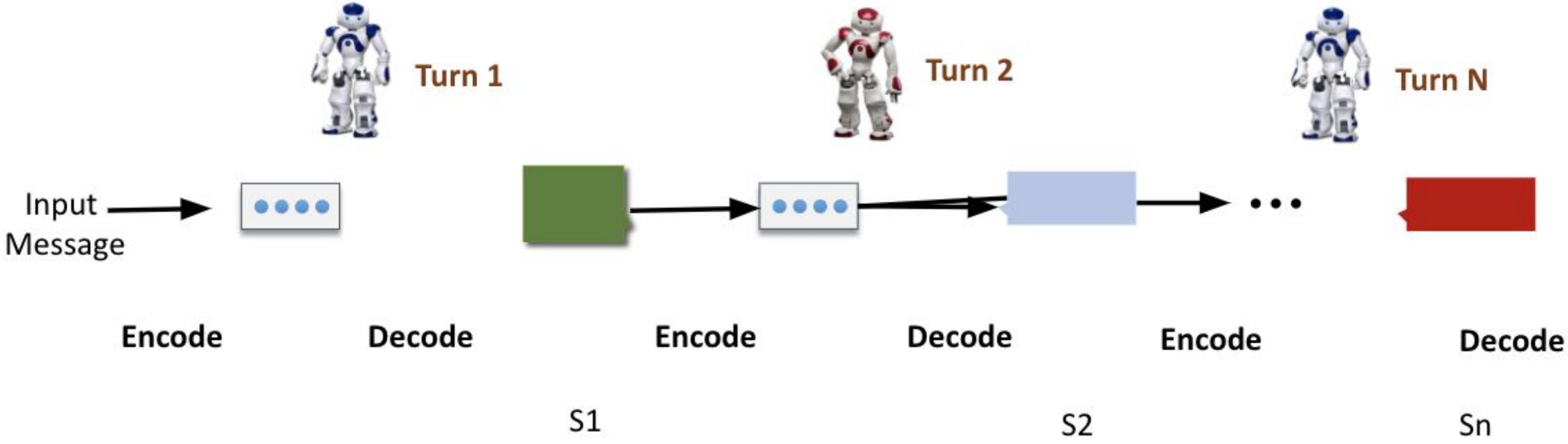


REINFORCE Algorithm (William,1992)

$$J(\theta) = \mathbb{E}[R(s_1, s_2, ..., s_N)]$$

$$\nabla J(\theta) = \nabla \log p(s_1, s_2, ..., s_N) R(s_1, s_2, ..., s_N)$$

$$\nabla J(\theta) = \nabla \log \prod_i \boxed{p(s_i | s_{i-1})} R(s_1, s_2, ..., s_N)$$



REINFORCE Algorithm (William,1992)

$$J(\theta) = \mathbb{E}[R(s_1, s_2, ..., s_N)]$$

$$\nabla J(\theta) = \nabla \log p(s_1, s_2, ..., s_N) R(s_1, s_2, ..., s_N)$$

what we want to learn!

$$\nabla J(\theta) = \nabla \log \prod_i \boxed{p(s_i|s_{i-1})} R(s_1, s_2, ..., s_N)$$

Experiments: Baseline Models

- Vanilla Seq2Seq
- Mutual Information Model
    - Seq2Seq + Mutual Information Rescoring during testing

Mutual Information Baseline

- avoid responses with unconditionally high probability
- bias towards those responses specific to the given input
- Objective: $\log p(T|S) - \log p(T)$
- Introduce hyperparameter that controls how much to penalize generic responses

$$\arg\max_T \left\{ (1-\lambda)\log p(T|S) + \underline{\lambda \log p(S|T)} \right\}$$

Mutual Information Baseline

- adapting MMI to SEQ2SEQ training is empirically nontrivial
- Want to adjust λ w/o repeatedly training neural network models from scratch
- Solution: train MLE model, use MMI criterion **only during testing**
- Decoding:
  - Generate best n responses from $p_{\text{SEQ2SEQ}}(s|t)$
  - Re-rank them by $p_{\text{SEQ2SEQ}}(t|s)$

Experiments

- For open-domain, the goal is not to predict the highest probability response, but rather the **long-term success** of the dialogue

  => BLEU and Perplexity are **not** appropriate

- Evaluate on dialogue length and diversity instead
- \+ human evaluation against proposed baseline models

Dialogue Length

- Dialogue ends when “I don’t know” is generated or utterances are highly overlapping
- Test set: 1000 input messages
- Limit # of simulated turns to be less than 8

Model	# of simulated turns
SEQ2SEQ	2.68
mutual information	3.40
RL	$\boxed{4.48}$

Diversity

- Number of distinct unigrams and bigrams in generated responses per token

Model	Unigram	Bigram
SEQ2SEQ	0.0062	0.015
mutual information	0.011	0.031
RL	$\boxed{0.017}$	$\boxed{0.041}$

Human Evaluation

- **single-turn general quality:** which generated reply is better for a given input message
- **single-turn ease to answer***:* which output is easier to respond to
- **multi-turn general quality:** which simulated conversation is of higher quality

Setting	RL-win	RL-lose	Tie
single-turn general quality	0.40	0.36	0.24
single-turn ease to answer	0.52	0.23	0.25
multi-turn general quality	0.72	0.12	0.16

Example Responses

Input Message	Mutual Information Model	RL
How old are you?	I'm 16.	I'm 16, why are you asking?
What's your full name?	I have no idea.	What's yours?
I don't want to go home tonight.	Really?	Why?
Do you have any feelings for me?	I don't know what you are talking about.	I don't want to hurt your feelings.
How much time do you have here?	Not long enough. Sorry, sir.	Ten seconds.
Shall we get started?	Of course!	Yes. We've got a lot of work to do here.
Do you play football?	No, i don't	Yes. I love football!
We'd have to talk to him.	I mean, he's a good guy	About what ?
How come you never say it?	Because I don't want to hurt you.	I don't think it's a good idea to say it.

Example Conversations

Baseline mutual information model (Li et al. 2015)	Proposed reinforcement learning model
A: Where are you going? (1) B: I'm going to the restroom. (2) A: See you later. (3) B: See you later. (4) A: See you later. (5) B: See you later. (6)	A: Where are you going? (1) B: I'm going to the police station. (2) A: I'll come with you. (3) B: No, no, no, no, you're not going anywhere. (4) A: Why? (5) B: I need you to stay here. (6) A: I don't know what you are talking about. (7) ...
A: how old are you? (1) B: I'm 16. (2) A: 16? (3) B: I don't know what you are talking about. (4) A: You don't know what you are saying. (5) B: I don't know what you are talking about . (6) A: You don't know what you are saying. (7) ...	A: How old are you? (1) B: I'm 16. Why are you asking? (2) A I thought you were 12. (3) B: What made you think so? (4) A: I don't know what you are talking about. (5) B: You don't know what you are saying. (6) ... ...

Example Analysis

- RL model generates **more interactive responses** than the other baselines
- RL model has a tendency to **end a sentence with** another **question** and hand the conversation over to the user
- RL model manages to **produce more interactive** and **sustained conversations** than the mutual information model
- Model sometimes **starts** a **less relevant topic** during the conversation (trade-off between relevance and less repetitiveness!)

Example Conversations with Cycle

- Model sometimes **enters a cycle** with length greater than one
- can be ascribed to the **limited** amount of conversational **history** considered

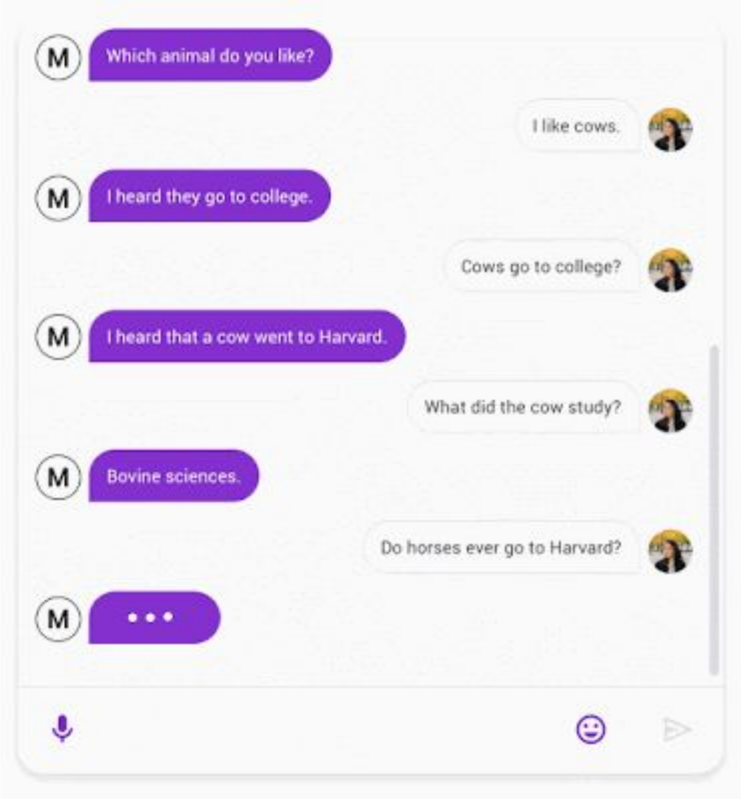
A: What's your name ?
B: Daniel.
A: How old are you ?
B. Twelve. What's your name ?
A. Daniel.
B: How old are you ?
A: Twelve. What's your name ?
B: Daniel.
A: How old are you ?
B ...

Limitations

- Currently there is a **tradeoff** between **relevance** and **less repetitiveness**
- A **manually defined reward** function can’t cover all crucial aspects of an ideal conversation; would be idea to receive **real rewards from humans**
- Can only afford to explore a very **small number of candidates** and simulated turns since the number of cases to consider grow exponentially

Extra: Meena Chatbot

- multi-turn open-domain chatbot
- **2.6B parameter** neural network
    - Seq2seq
    - Evolved Transformer



Extra: Meena Chatbot

- trained end-to-end on **social media conversations**
- **simply** trained to **minimize perplexity of** the **next token**

Extra: Meena Chatbot

Much more appropriate than OpenSubtitles!

- trained end-to-end on **social media conversations**
- **simply** trained to **minimize perplexity of** the **next token**

Extra: Meena Chatbot

Sample Contexts	**Meena (base)**	**Cleverbot**	**DialoGPT**
A: You must have read a lot of books *B:* Oh wow, how do you know? Do I look nerdy? *A:* Of course, you're not!	*B:* So how do you know? [*sensible: 1, specific: 1*]	*B:* Never. said I was. [*sensible: 0, specific: 0*]	*B:* Oh thank goodness. [*sensible: 1, specific: 1*]
A: You know nothing, Jon Snow! *B:* You must be a fan of Game of Thrones *A:* Haha yea, how can I not?	*B:* Who's your favorite character? [*sensible: 1, specific: 1*]	*B:* What do you mean? [*sensible: 0, specific: 0*]	*B:* You can't. But that's not why you shouldn't know. [*sensible: 0, specific: 0*]
A: What is your favorite island in the world? *B:* I would say Fiji *A:* Interesting, what's nice about it?	*B:* Just a beautiful place [*sensible: 1, specific: 0*]	*B:* The title is self explanatory. [*sensible: 0, specific: 0*]	*B:* You can see some great scenery. [*sensible: 1, specific: 0*]

Sensibleness and Specificity Average (SSA)

- new evaluation metric
- covers two fundamental aspects of a human-like chatbot
    - making sense
    - being specific
- human judges to label every model response on these two criteria
- static and interactive version

Sensibleness and Specificity Average (SSA)

- (surprising) empirical observation: **strongly correlated with perplexity,** both in static and interactive evaluation