Математические методы анализа текстов

Reinforcement Learning in Natural Language Processing

Потапенко Анна Александровна

5 ноября 2018 г.

What is Reinforcement Learning

What makes RL different from other machine learning paradigms?

- There is no supervision, only a reward signal
- Feedback is delayed, not instantaneous
- Time really matters (sequential, non i.i.d data)
- Agent's actions affect the subsequent data it receives

Reward hypothesis: The agent's job is to maximize cumulative reward (a scalar feedback signal).

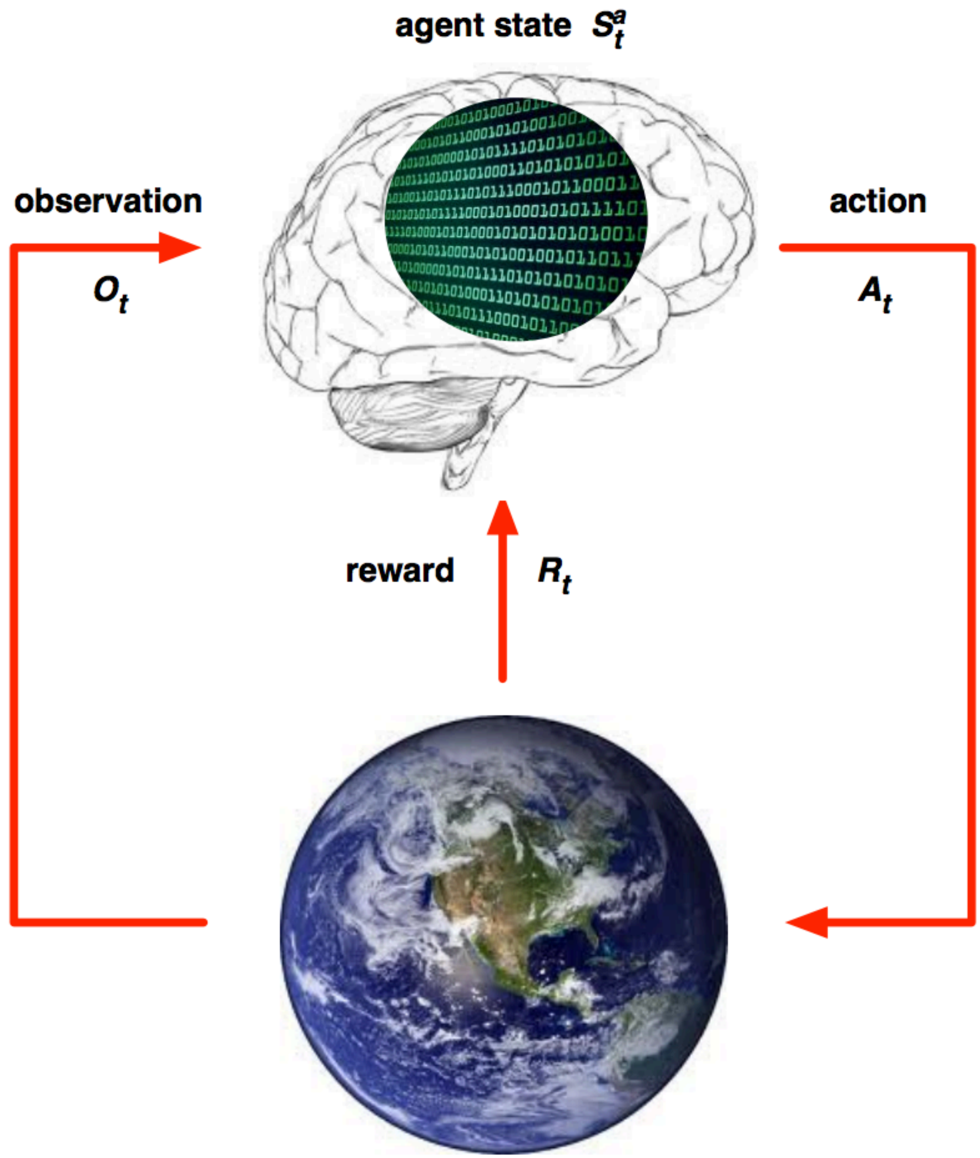
David Silver's lectures (slides and videos)

Main components

History is a sequence of:

- **observations**
- **actions**
- **rewards**

Agent state is agent's internal representation (any function of history).



David Silver's lectures (slides and videos)

RL agent types

RL agent may include one or more of these components:

- **Policy:**

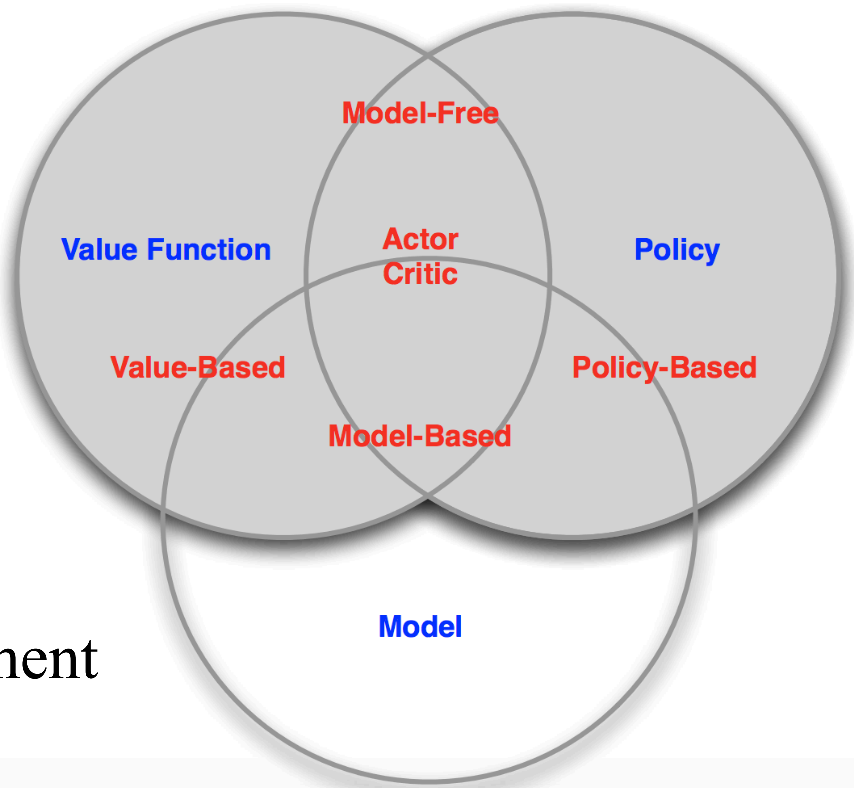
agent's behavior function

- **Value function:**

how good is each state and/or action

- **Model:**

agent's representation of the environment



David Silver's lectures (slides and videos)

Deep Policy Network

Agent's policy is a map from state to action:

❖ **Deterministic policy:**

$$a = \pi(s)$$

❖ **Stochastic policy:**

$$\pi(a|s) = p_\theta(A_t = a|S_t = s)$$

Represent policy by a deep neural network and learn parameters:

$$\mathbb{E}_{a \sim p_\theta(a|s)}[r(a)] \to \max_\theta$$

David Silver's lectures (slides and videos)

REINFORCE

$$\nabla_\theta \mathbb{E}_{a\sim p_\theta}[r(a)] = \nabla_\theta \sum_a p_\theta(a) r(a)$$

$$= \sum_a \nabla_\theta p_\theta(a) r(a)$$

$$= \sum_a p_\theta(a) \nabla_\theta \log p_\theta(a) r(a)$$

$$= \sum_a p_\theta(a) \frac{\nabla_\theta p_\theta(a)}{p_\theta(a)} r(a)$$

$$= \mathbb{E}_{a\sim p_\theta(a)}[r(a) \nabla_\theta \log p_\theta(a)]$$

http://karpathy.github.io/2016/05/31/rl/

Let's apply it to summarization / MT

Problems with seq2seq architectures:

- **Exposure bias:** a model is only exposed to the training data distribution, instead of its own predictions.
- **Loss:** a model is trained with word-level cross-entropy instead of discrete quality measures such as BLEU or ROUGE.

Solution to both: REINFORCE.

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

where the policy is RNN and the reward is ROUGE.

Sequence Level Training with Recurrent Neural Networks, FAIR, ICLR-2016.

More details

- Agent: generative model (the RNN)
- Environment: the words and the context vector at each time step
- Action: the next word in the sequence at each time step
- Policy: probability of the next word from the RNN
- Internal state: the hidden units of RNN
- Reward (in the end of the sequence): BLEU/ROUGE
- Optimize with REINFORCE:

$$\nabla \mathbb{E}_{a \sim p_\theta}[r(a)] = \mathbb{E}_{a \sim p_\theta}[r(a) \nabla \log p_\theta(a)]$$

Sequence Level Training with Recurrent Neural Networks, FAIR, ICLR-2016.

Compute gradients

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

$$\frac{\partial L_\theta}{\partial \theta} = \sum_t \frac{\partial L_\theta}{\partial \mathbf{o}_t} \frac{\partial \mathbf{o}_t}{\partial \theta}$$

where $\mathbf{o}_t$ is the input to the softmax.

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

where $\bar{r}_{t+1}$ is the average reward at time $t+1$.

Can you interpret the formulas?

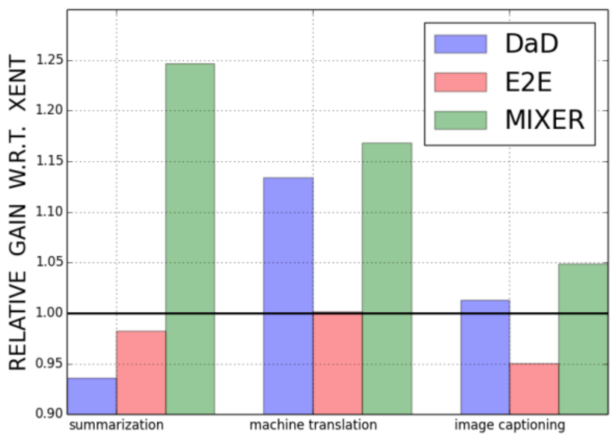
Sequence Level Training with Recurrent Neural Networks, FAIR, ICLR-2016.

Initialization and annealing

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

Algorithm 1: MIXER pseudo-code.

TASK	XENT	DAD	E2E	MIXER
summarization	13.01	12.18	12.78	**16.22**
translation	17.74	20.12	17.77	**20.73**
image captioning	27.8	28.16	26.42	**29.16**

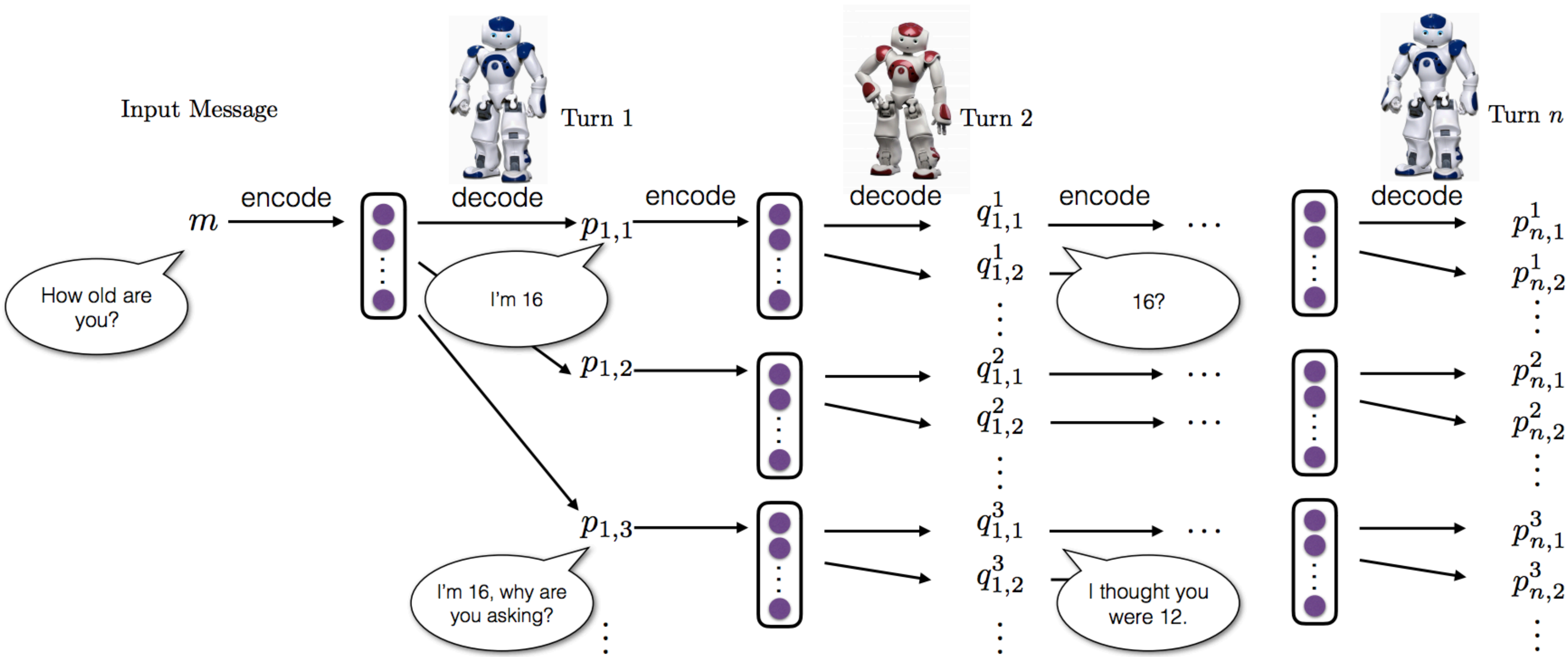


[Sequence Level Training with Recurrent Neural Networks](), FAIR, ICLR-2016.

Where to apply RL in NLP?

- Optimize a discrete loss in any seq2seq:
  - Summarization
  - Machine translation
  - Image captioning
  - Text simplification
  - Dialogue systems
  - Question answering
  - Language to code
  - …
- Use a loss of the main task as a reward for a scaffolding task
  - e.g. joint sentiment analysis and syntactic parsing
- Negotiation dialogues
  - natural language emerging in a self-play of two agents
- ….

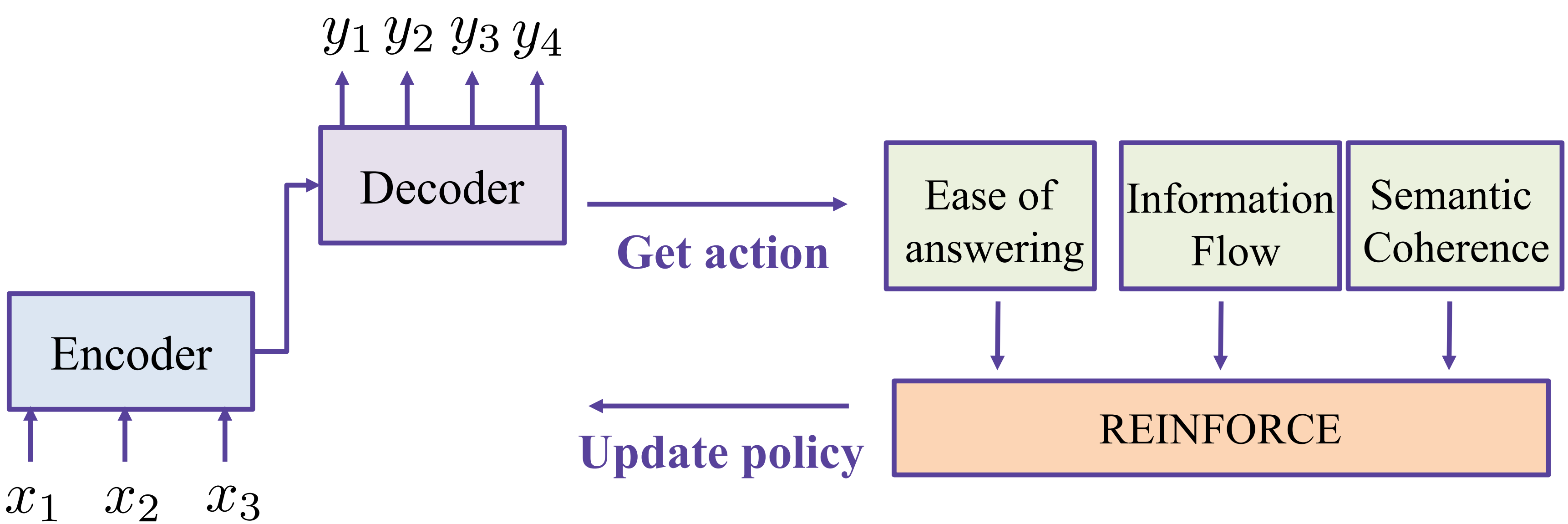
Dialogue generation (simulation)



Deep Reinforcement Learning for Dialogue Generation, 2016. https://arxiv.org/pdf/1606.01541.pdf

Dialogue generation (REINFORCE)

- **State:** LSTM encoder for two previous dialogue turns
- **Action:** dialogue utterance (of any length)
- **Policy:** LSTM encoder-decoder
- **Reward:** a sum of 3 components:



Reward components

- **Ease of answering:** negative log likelihood of responding to that utterance with a dull response:

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

- **Information flow:** penalize semantic similarity between consecutive turns from the same agent:

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

- **Semantic coherence:** mutual information between the action and previous turns in the history:

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

Important hacks

- **Initialization:**
  - Train supervised encoder-decoder model on OpenSubtitles dataset
  - Maximize mutual information of responses with RL
    - **Annealing:** increase the number of the remaining tokens in the sequence to be trained with RL
- **Simulation** (dialogue generated by two agents):
  - Policy gradients for the described reward (3 components)
  - **Curriculum learning:** increase the number of turns in the generated dialogues

Results

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) ... ...

Deep Reinforcement Learning for Dialogue Generation, 2016. https://arxiv.org/pdf/1606.01541.pdf

Where to apply RL in NLP?

- Optimize a discrete loss in any seq2seq:
  - Summarization
  - Machine translation
  - Image captioning
  - Text simplification
  - Dialogue systems
  - Question answering
  - Language to code
  - …
- Use a loss of the main task as a reward for a scaffolding task
  - e.g. joint sentiment analysis and syntactic parsing
- Negotiation dialogues
  - natural language emerging in a self-play of two agents
- ….

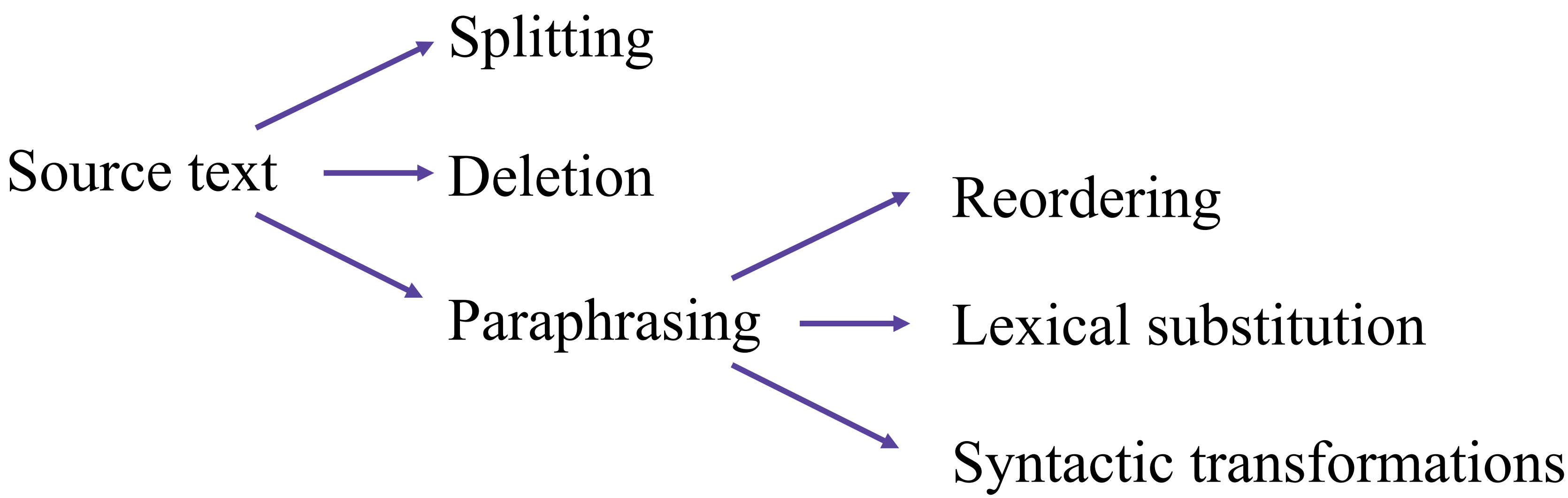
Simplification

Text simplification – reducing the lexical and syntactical complexity of text.

a.	**Normal:** As Isolde arrives at his side, Tristan dies with her name on his lips. **Simple:** As Isolde arrives at his side, Tristan dies while speaking her name.
b.	**Normal:** Alfonso Perez Munoz, usually referred to as Alfonso, is a former Spanish footballer, in the striker position. **Simple:** Alfonso Perez is a former Spanish football player.
c.	**Normal:** Endemic types or species are especially likely to develop on islands because of their geographical isolation. **Simple:** Endemic types are most likely to develop on islands because they are isolated.

Coster et. al. Simple English Wikipedia: A New Text Simplification Task, 2011.

Operations to simplify text



Xu et. al. Optimizing Statistical Machine Translation for Text Simplification, 2016.
Tong Wang et al. Text Simplification Using Neural Machine Translation, AAAI-16

Rule-based approach for paraphrasing

Lexical	[RB]	solely	→	only
	[NN]	objective	→	goal
	[JJ]	undue	→	unnecessary
Phrasal	[VP]	accomplished	→	carried out
	[VP/PP]	make a significant contribution	→	contribute greatly
	[VP/S]	is generally acknowledged that	→	is widely accepted that
Syntactic	[NP/VP]	the manner in which NN	→	the way NN
	[NP]	NNP 's population	→	the people of NNP
	[NP]	NNP 's JJ legislation	→	the JJ law of NNP

- Synchronous context-free grammar (SCFG) rules
- Uppercase indicates non-terminal symbols
- Paraphrase Database http://www.cis.upenn.edu/~ccb/ppdb/

Xu et. al. Optimizing Statistical Machine Translation for Text Simplification, 2016.

Simplification

Encoder-decoder framework – yes, but the network might learn just to **copy** the content… How do we force it to **simplify**?

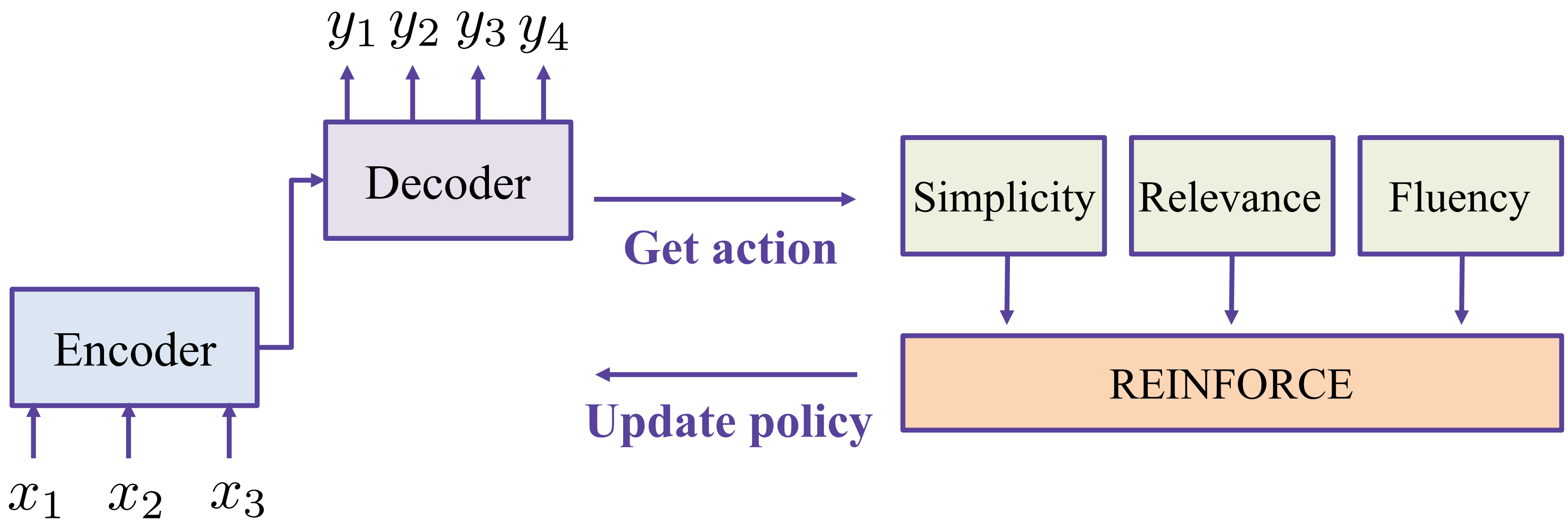
Reinforcement learning can be used to do **weak supervision.**

- **Action:** output next word y_j
- **Policy:** $p(y_j|\mathbf{x}, y_1, \ldots y_{j-1})$
- **Reward:** Adequacy + Fluency + Simplicity

Rewards come only when the whole sequence is generated.

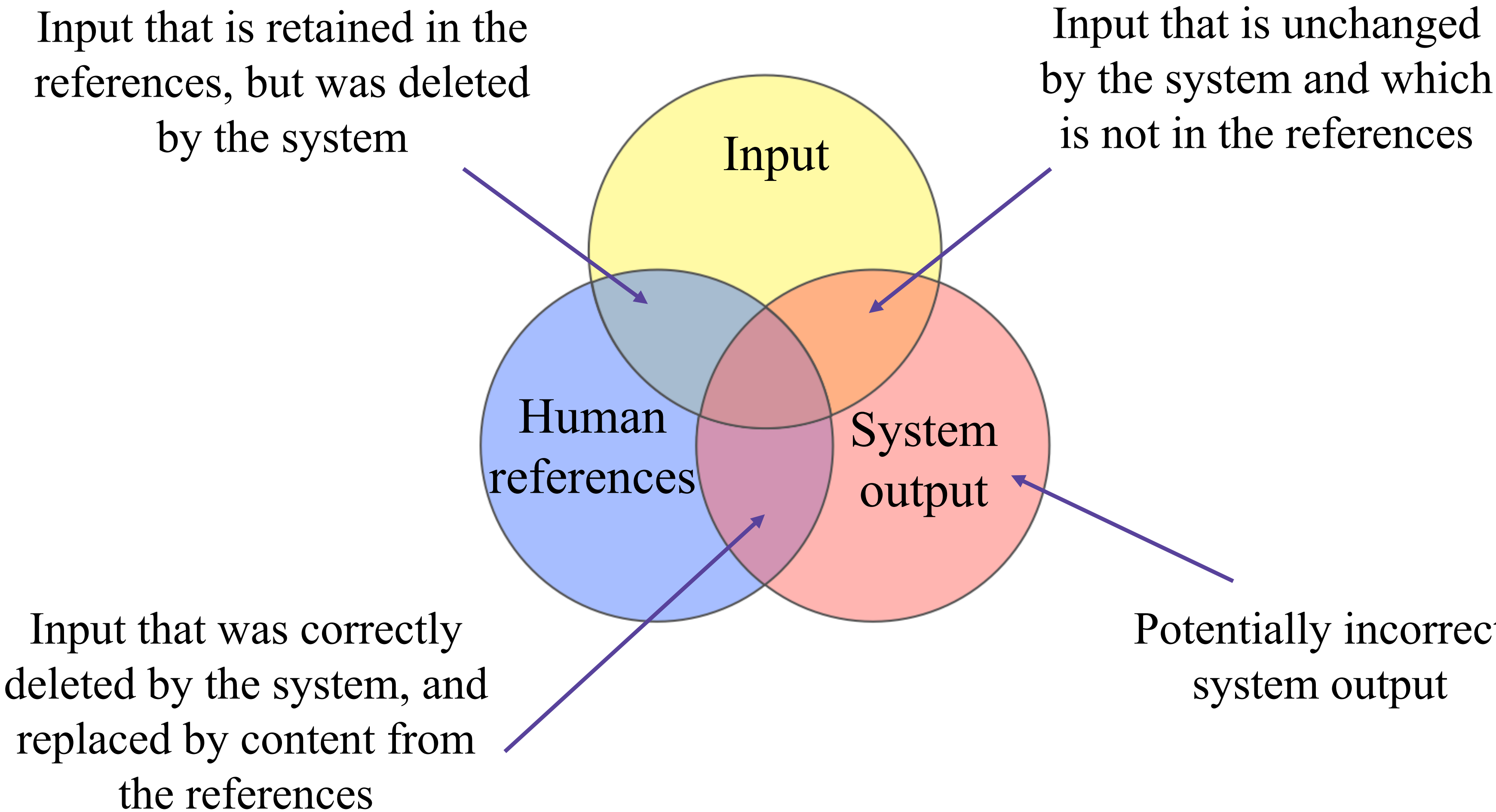
Zhang, Lapata. Sentence Simplification with Deep Reinforcement Learning, 2017.

Simplification



Zhang, Lapata. Sentence Simplification with Deep Reinforcement Learning, 2017.

How to measure simplicity?



Xu et. al. Optimizing Statistical Machine Translation for Text Simplification, 2016.

How to measure simplicity?

SARI (**s**ystem **a**gainst **r**eferences and **i**nput) –
arithmetic average of n-gram precision and recall of

- addition
- copying
- deletion

For example, precision for **addition**:

$$\text{precision} = \frac{\sum_{g \in O} [g \in (O \cap \bar{I} \cap R)]}{\sum_{g \in O} [g \in (O \cap \bar{I})]}$$

Xu et. al. Optimizing Statistical Machine Translation for Text Simplification, 2016.

SARI: example

INPUT: About 95 species are currently accepted.

REF-1: About 95 species are currently known.

*REF-2: About 95 species are **now** accepted.*

REF-3: 95 species are now accepted.

OUTPUT-1: About 95 you now get in. → 0.2683

*OUTPUT-2: About 95 species are **now** agreed.* → **0.7594**

OUTPUT-3: About 95 species are currently agreed. → 0.5890

Xu et. al. Optimizing Statistical Machine Translation for Text Simplification, 2016.

Compare with BLEU

INPUT: About 95 species are currently accepted.

REF-1: About 95 species are currently known.

*REF-2: About 95 species are **now** accepted.*

REF-3: 95 species are now accepted.

OUTPUT-1: About 95 you now get in. → 0.1562

*OUTPUT-2: About 95 species are **now** agreed.* → **0.6435**

OUTPUT-3: About 95 species are currently agreed. → **0.6435**

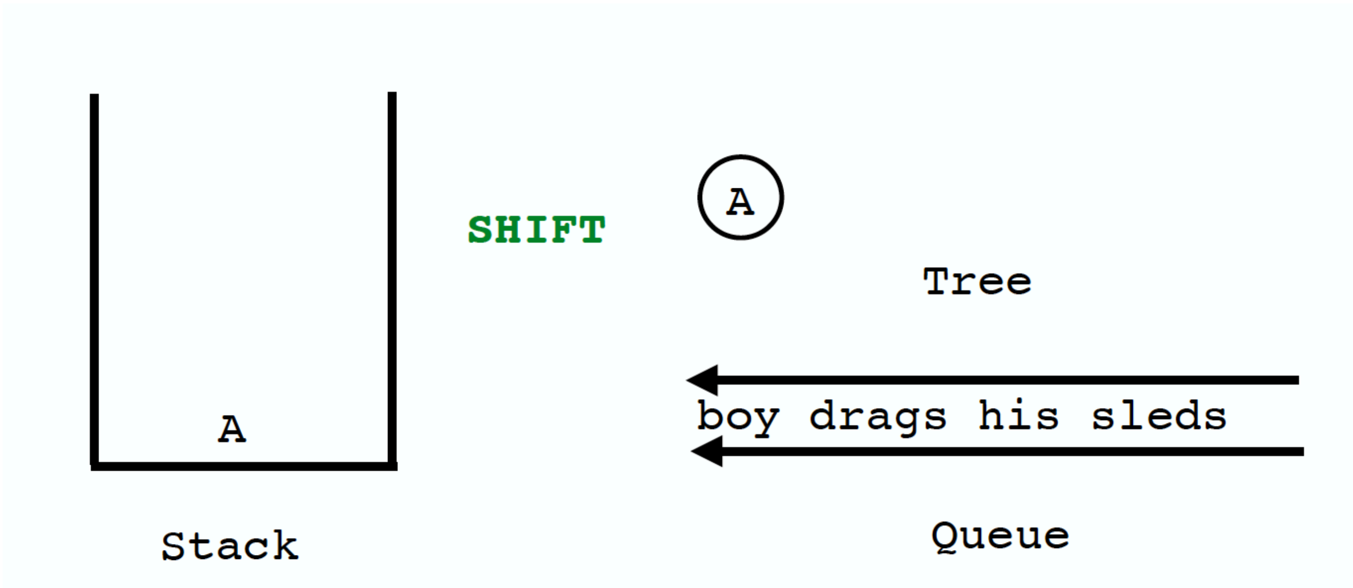
BLEU does not distinguish between outputs 2 and 3.

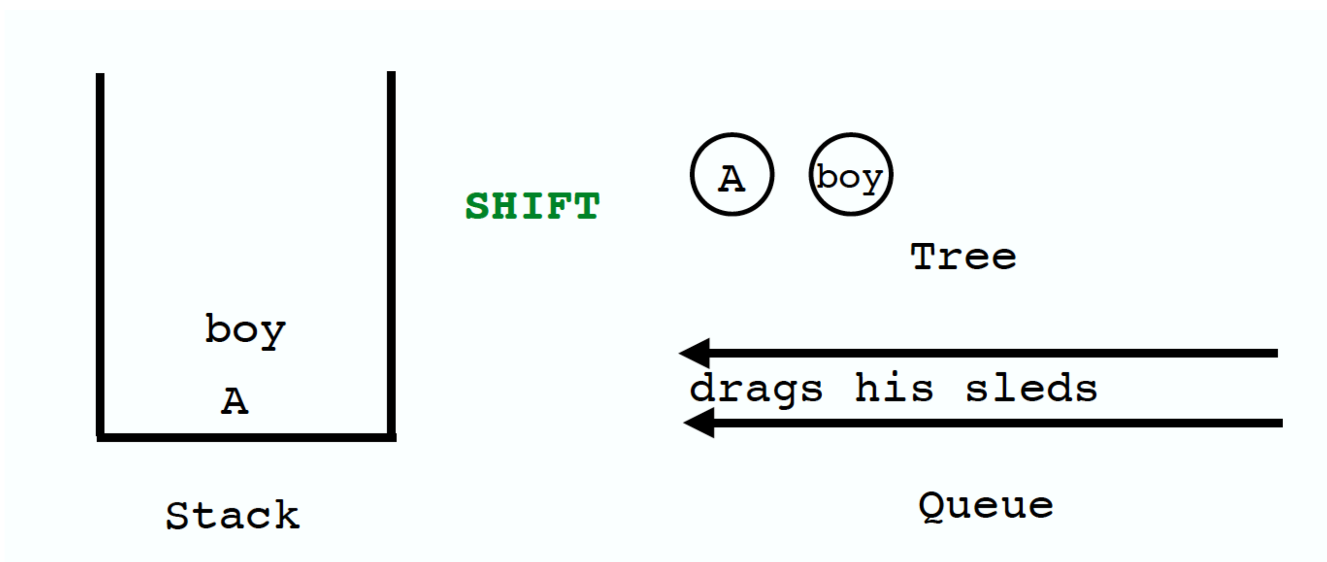
Xu et. al. Optimizing Statistical Machine Translation for Text Simplification, 2016.

Where to apply RL in NLP?

- Optimize a discrete loss in any seq2seq:
  - Summarization
  - Machine translation
  - Image captioning
  - Text simplification
  - Dialogue systems
  - Question answering
  - Language to code
  - …
- Use a loss of the main task as a reward for a scaffolding task
  - e.g. joint sentiment analysis and syntactic parsing
- Negotiation dialogues
  - natural language emerging in a self-play of two agents
- ….

Shift-reduce parsing (Aho and Ullman, 1972)



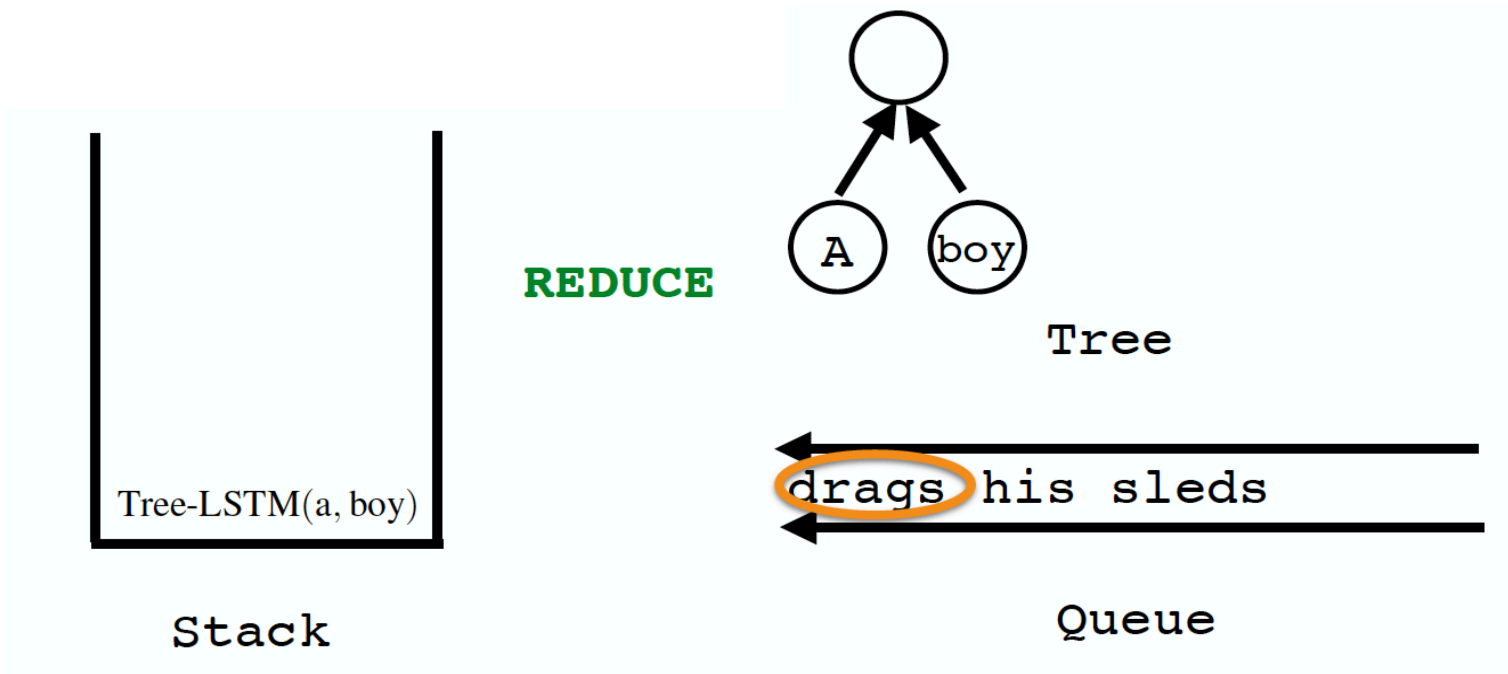


Slides/pictures credit: Chris Dyer, CONLL-2017.

Shift-reduce parsing (Aho and Ullman, 1972)

REDUCE:

- Compose top two elements of the stack with Tree LSTM (Tai et al., 2015 Zhu et al., 2015)
- Push the result back onto the stack



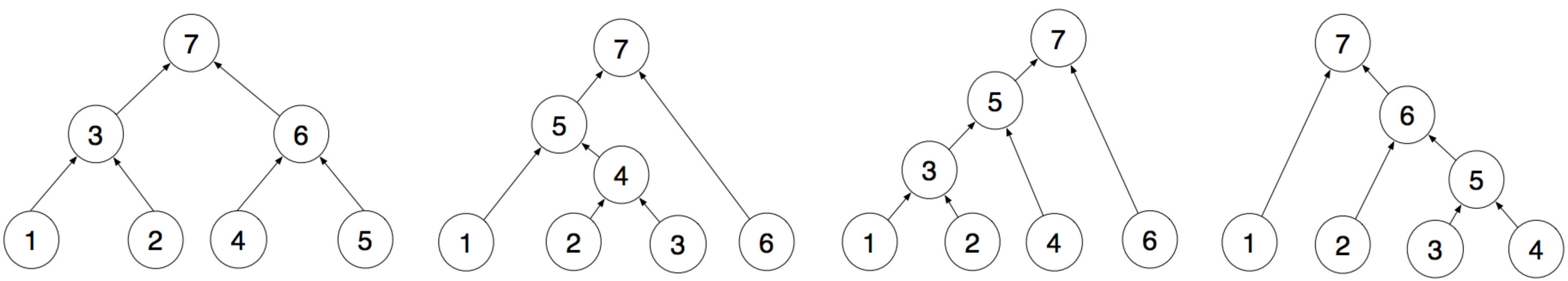
Slides/pictures credit: Chris Dyer, CONLL-2017.

Shift-reduce parsing

- Different Shift/Reduce sequences lead to different tree structures

Learning:

- How do we learn the policy for the shift reduce sequence?
- What if we don't have a treebank (labels)?



S, S, R, S, S, R, R S, S, S, R, R, S, R S, S, R, S, R, S, R S, S, S, S, R, R, R

Paper: https://arxiv.org/pdf/1611.09100.pdf

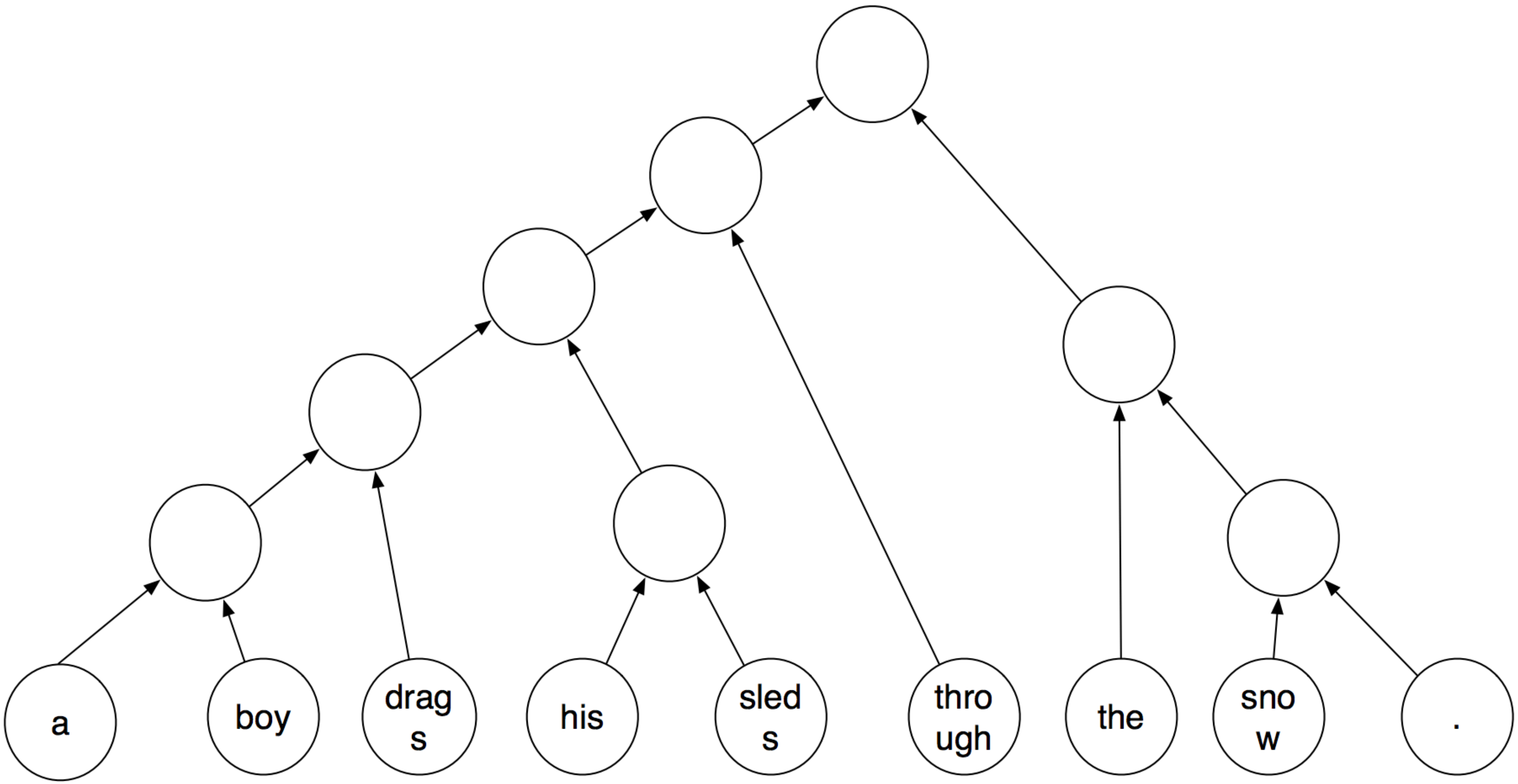
Reinforcement Learning

- **State:** embeddings of top two elements of the stack, embedding of head of the queue.
- **Actions:** shift, reduce.
- **Reward:** log likelihood on a downstream task given the produced representation (e.g. sentiment analysis).
- **Policy:** two-layer feedforward network.

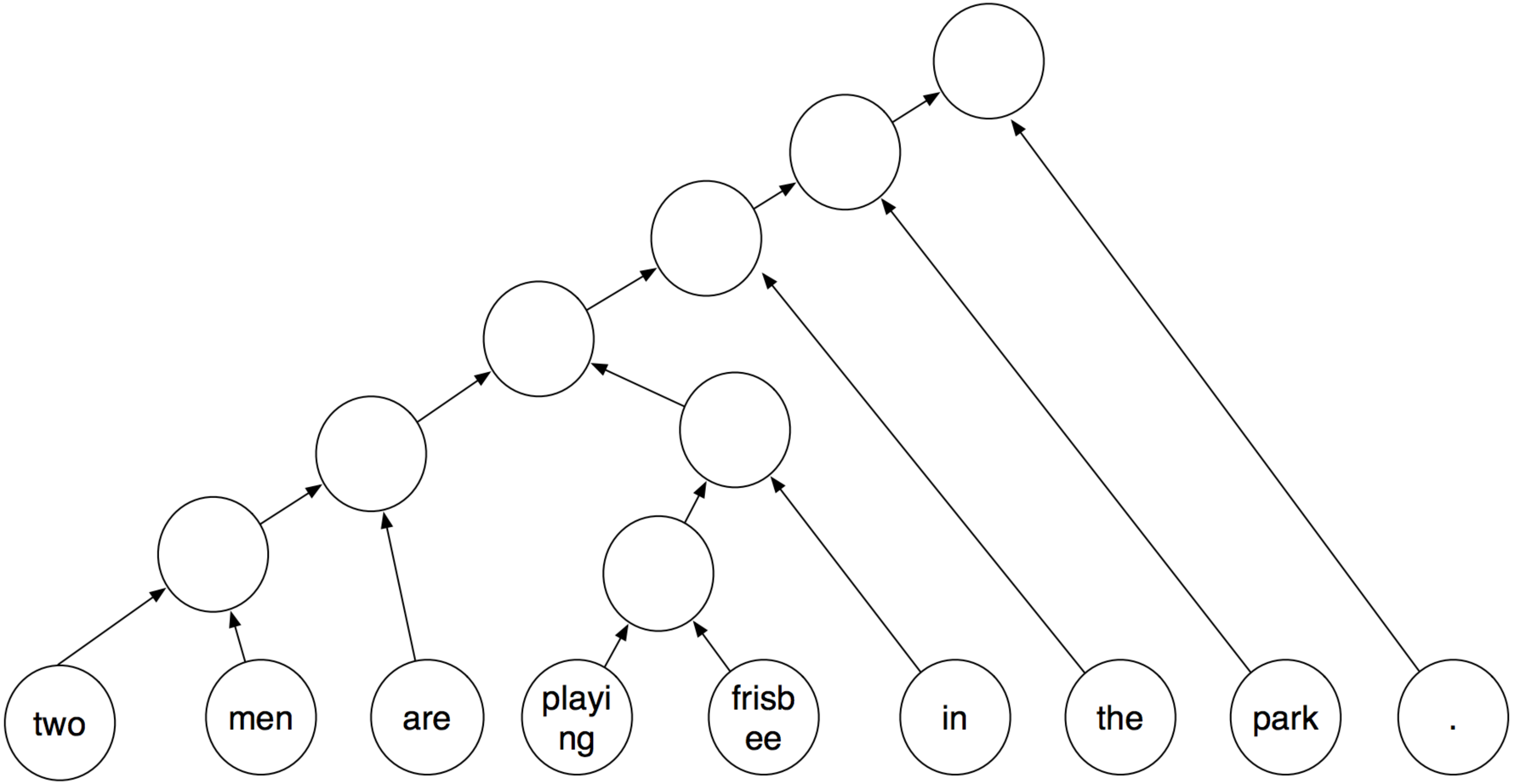
Use REINFORCE (policy gradient method) to build the parse tree.

Paper: https://arxiv.org/pdf/1611.09100.pdf

Syntax parse tree examples



Syntax parse tree examples



Results [Chris Dyer, CoNLL-2017]

Method	Accuracy
Naive Bayes (from Socher et al., 2013)	81.8
SVM (from Socher et al., 2013)	79.4
Average of Word Embeddings (from Socher et al., 2013)	80.1
Bayesian Optimization (Yogatama et al., 2015)	82.4
Weighted Average of Word Embeddings (Arora et al., 2017)	82.4
Left-to-Right LSTM	84.7
Right-to-Left LSTM	83.9
Bidirectional LSTM	84.7
Supervised Syntax	**85.3**
Semi-supervised Syntax	**86.1**
Latent Syntax	**86.5**

Paper: https://arxiv.org/pdf/1611.09100.pdf

Results [Chris Dyer, CoNLL-2017]

Trees look "non linguistic", but downstream performance is great!

Do we need better bias in our models?

❖ Yes! They are making the wrong generalizations, even from large data.

Do we have to have the perfect model?

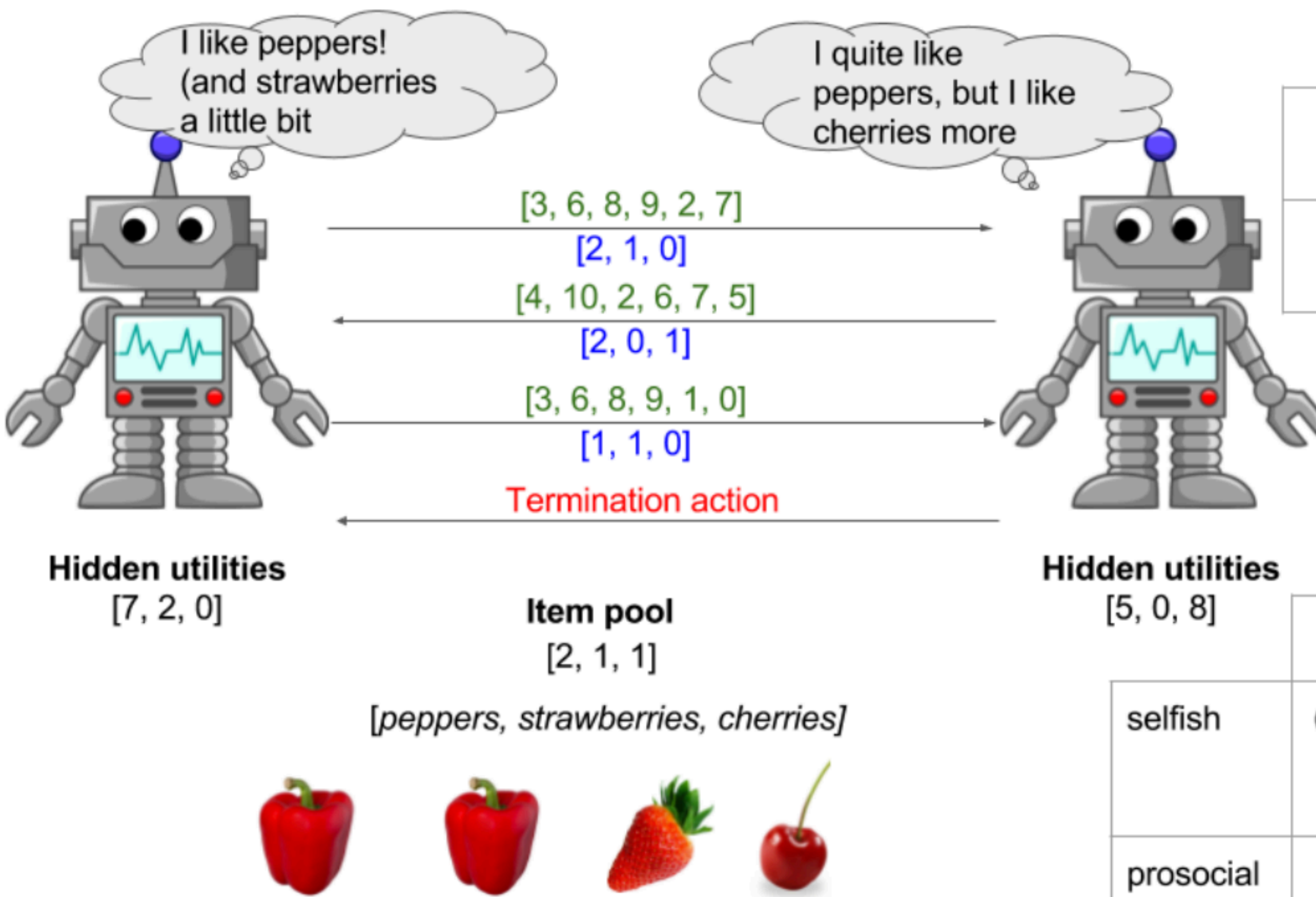
❖ No! Small steps in the right direction can pay big dividends.

Slides/pictures credit: Chris Dyer, CONLL-2017.

Where to apply RL in NLP?

- Optimize a discrete loss in any seq2seq:
  - Summarization
  - Machine translation
  - Image captioning
  - Text simplification
  - Dialogue systems
  - Question answering
  - Language to code
  - …
- Use a loss of the main task as a reward for a scaffolding task
  - e.g. joint sentiment analysis and syntactic parsing
- Negotiation dialogues
  - natural language emerging in a self-play of two agents
- ….

Emergent communication through negotiation



Communication channels
Linguistic channel Proposal channel

Agent rewards

	Agent A	Agent B
selfish	(1x7+1x2) **9**	(1x5+1x8) **13**
prosocial	(9+13) **22**	(13+9) **22**

Emergent communication through negotiation, DeepMind, ICLR 2018.

Talk the walk: Navigating New York City

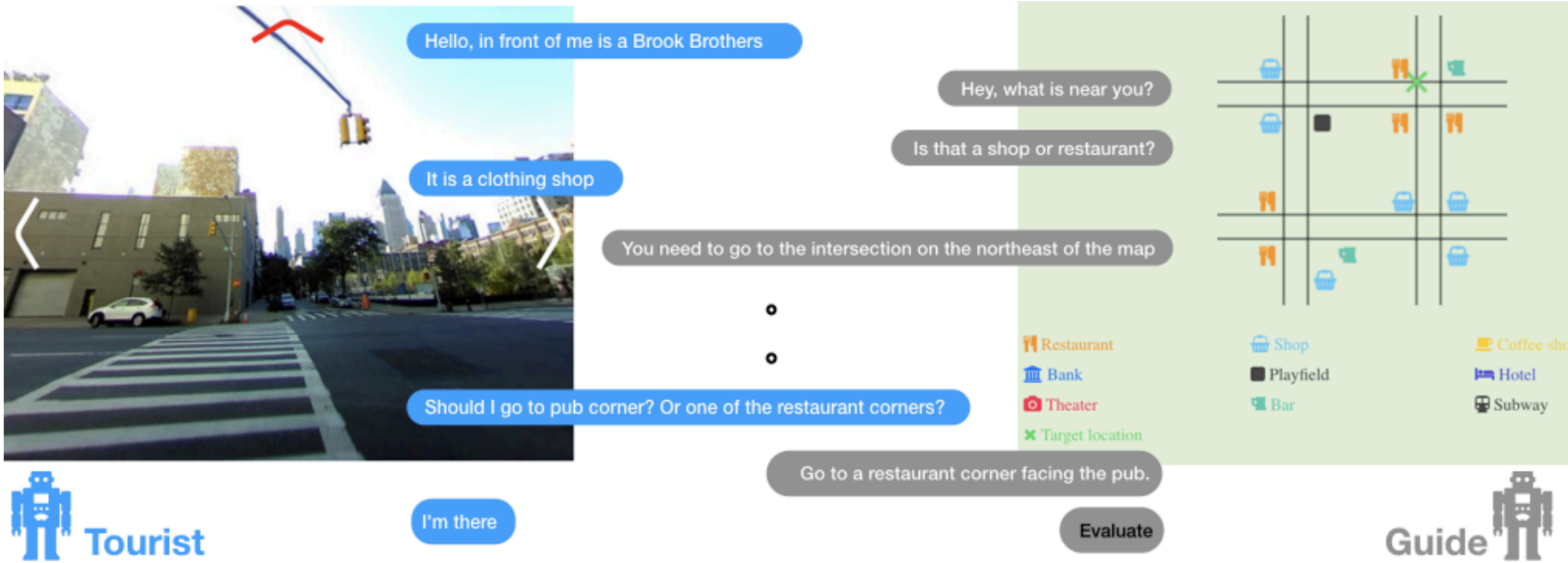


Figure 1: Example of the Talk The Walk task: two agents, a "tourist" and a "guide", interact with each other via natural language in order to have the tourist navigate towards the correct location. The guide has access to a map and knows the target location but not the tourist location, while the tourist does not know the way but can navigate in a 360-degree street view environment.

Talk the Walk: Navigating New York City through Grounded Dialogue, FAIR, 2018.

Thanks! Questions?