Integrating Domain-Knowledge into Deep Learning

Russ Salakhutdinov

Machine Learning Department
Carnegie Mellon University
Canadian Institute for Advanced Research

Domain knowledge

- Two key ingredients of a Statistical Machine Learning system
  - Model architecture/class
  - Learning algorithms to learn from data
- How do we incorporate domain knowledge into either or both these ingredients?
- We can consider three classes of domain knowledge:
  - Relational
  - Logical
  - Scientific

Relational Knowledge

- Simple relations among entities
  - (father, Bob, Alice)
- Available via relational databases, or knowledge graphs

- Statistical Relational Models
  - Probabilistic Graphical Models (PGMs) to model relationships amongst entities
  - Probabilistic Relational Models (via Bayes Nets), Relational Dependency Networks
- Embeddings
  - Instead of distributional semantics, represent entities via vectors in some vector space
  - Learn these vector representations via predicting an entity given its "context"

- We show how to incorporate relational information in Deep Learning via knowledge graph propagation

Ravikumar, Salakhutdinov, 2019

Logical Knowledge

- Propositional and First Order Logic (FOL) based knowledge
  - In contrast to simpler tuple based relational knowledge
  - E.g. if object has a wing, and a beak, it is a bird

- Encode logical knowledge into Probabilistic Graphical Models
- Bayesian Networks from Horn clauses, Probabilistic Context Free Grammars, Markov Logic Networks

- We incorporate logical information (and more general constraints) into Deep Learning via distillation (student-teacher) framework

Scientific Knowledge

- Partial and Stochastic Differential Equations
  - Newton Laws of Motion
  - Navier-Stokes fluid dynamics equations
  - ...
- Conservation laws and principles, Invariances

- Learning PDEs from data
- Regularizing dynamical system (e.g. state space models) via PDEs

Ravikumar, Salakhutdinov, 2019

Reading Comprehension

Her plain face broke into a huge smile when she saw Terry.
“Terry!” she called out.
She rushed to meet him and they embraced.
“Hon, I want you to meet an old friend, Owen McKenna.
Owen, please meet Emily."
She gave me a quick nod and turned back to **X**

LAMBADA dataset, Paperno et al., 2016

Reading Comprehension

Her plain face broke into a huge smile when she saw **Terry**.
“Terry!” she called out.
She rushed to meet him and they embraced.
“Hon, I want you to meet an old friend, **Owen** McKenna.
Owen, please meet **Emily**."
She gave me a quick nod and turned back to **X**

LAMBADA dataset, Paperno et al., 2016

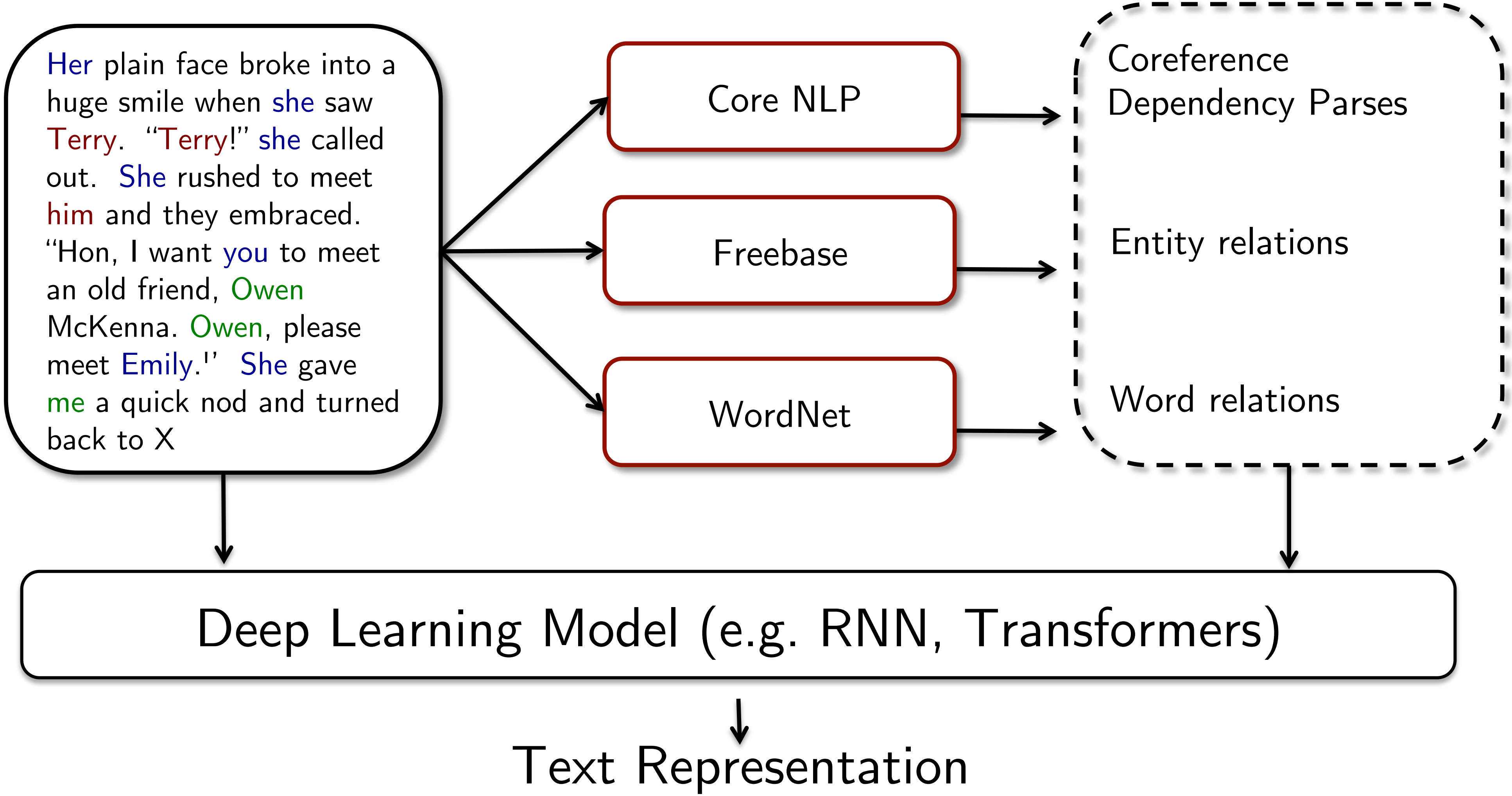
Reading Comprehension

Her plain face broke into a huge smile when she saw **Terry**.
“Terry!” she called out.
She rushed to meet him and they embraced.
“Hon, I want you to meet an old friend, **Owen** McKenna.
Owen, please meet **Emily**."
She gave me a quick nod and turned back to **X**

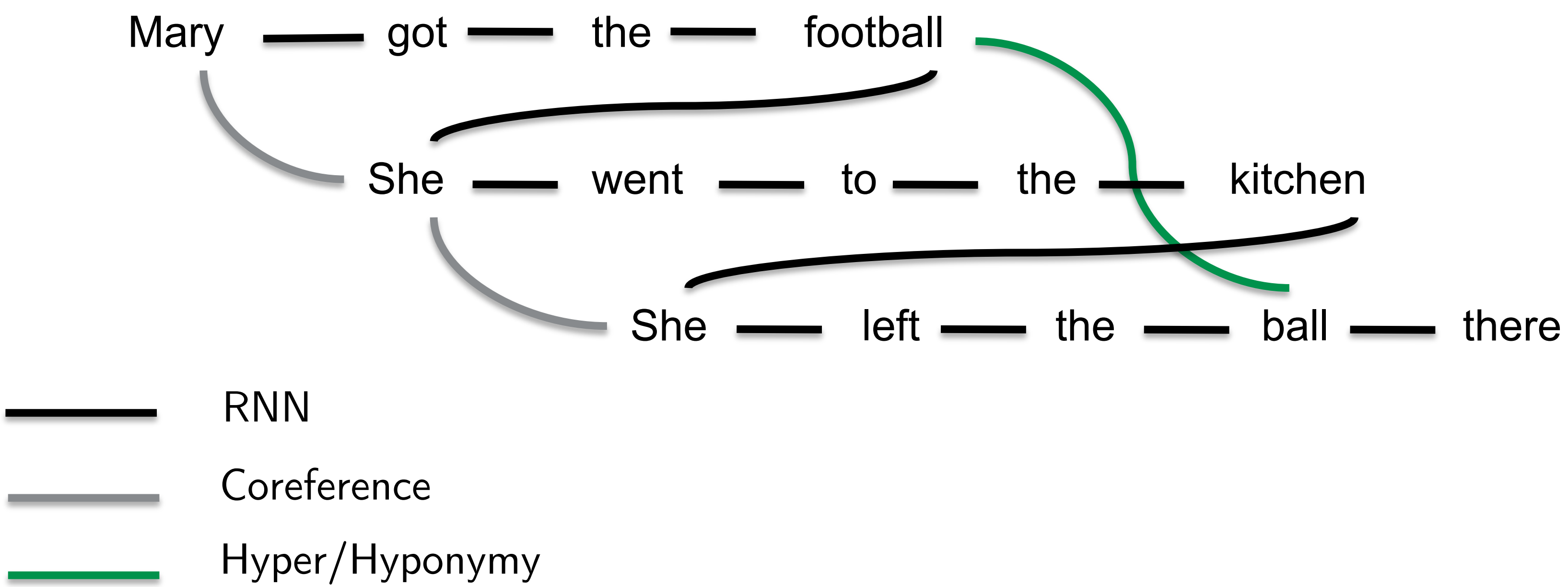
X = **Terry**

LAMBADA dataset, Paperno et al., 2016

Incorporating Prior Knowledge

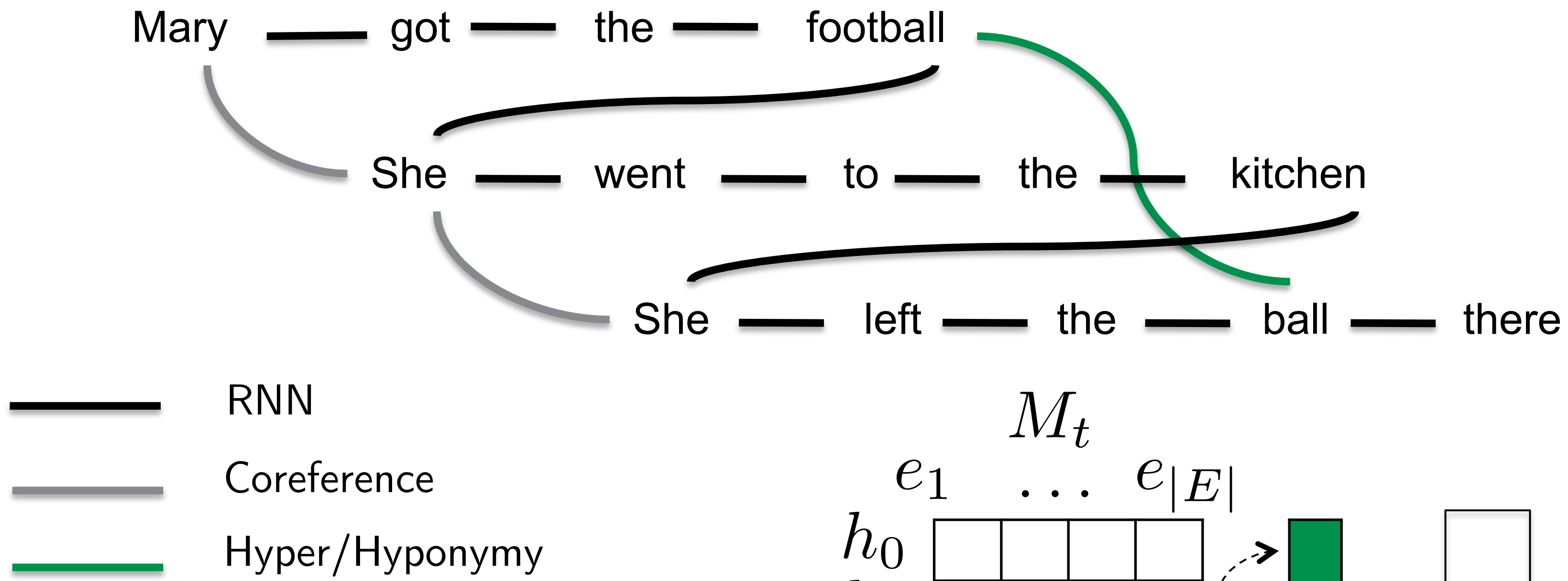


Explicit Memory

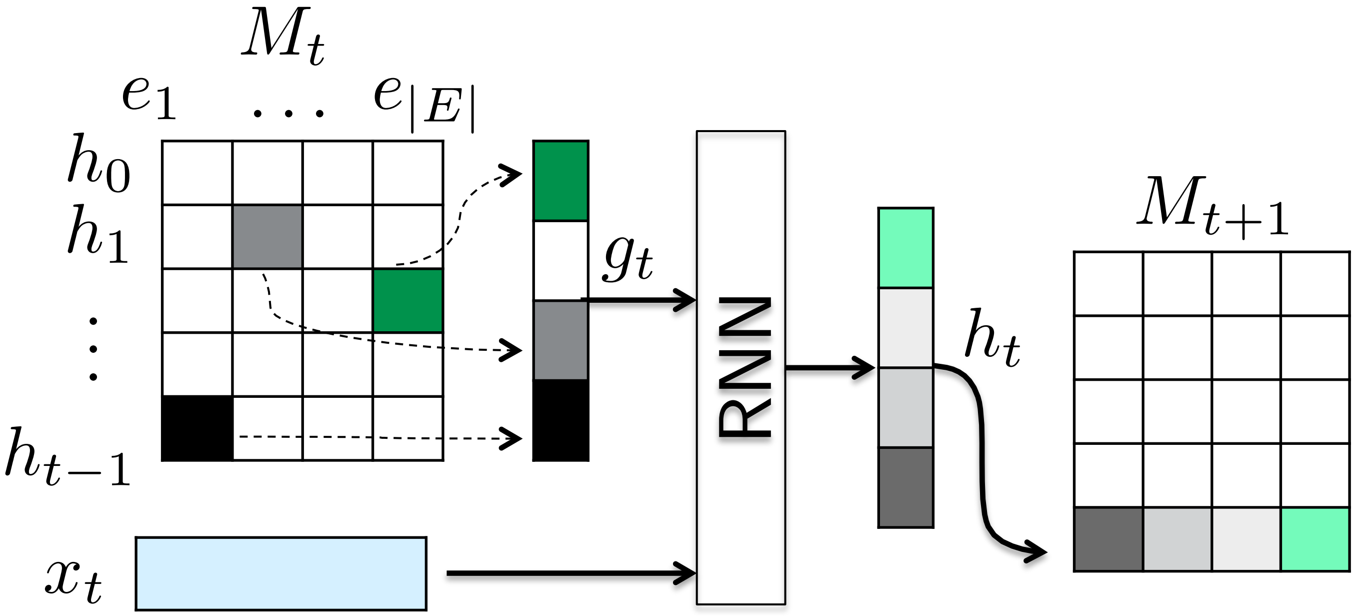


Dhingra, Jin, Yang, et al, NAACL 2018

Explicit Memory



Memory as **A**cyclic **G**raph
Encoding (MAGE) - RNN



Dhingra, Jin, Yang, et al, NAACL 2018

Open Domain Question Answering

- Finding answers to factual questions posed in Natural Language:

Who voiced Meg in Family Guy?

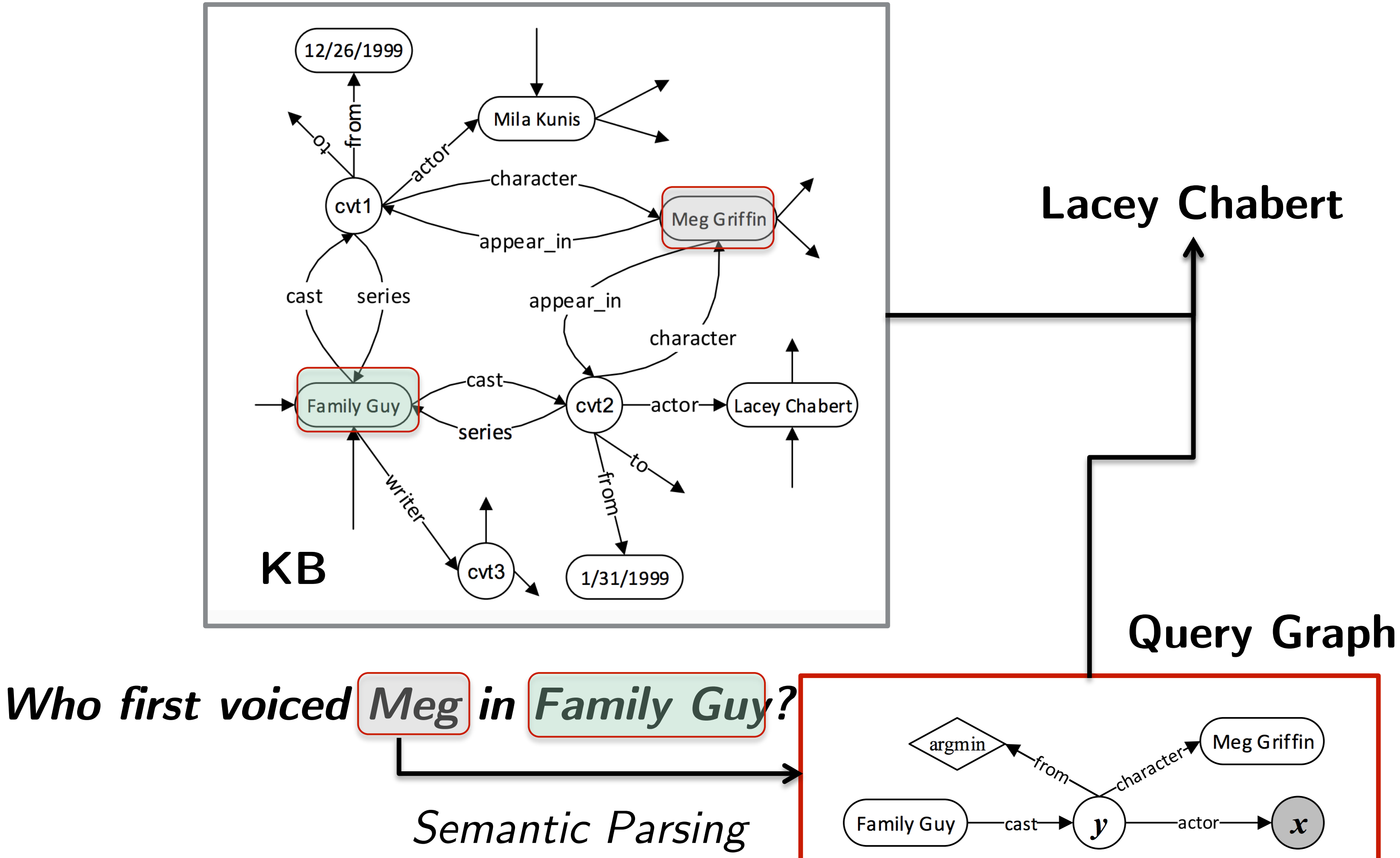
A. Lacey Chabert, Mila Kunis

Who first voiced Meg in Family Guy?

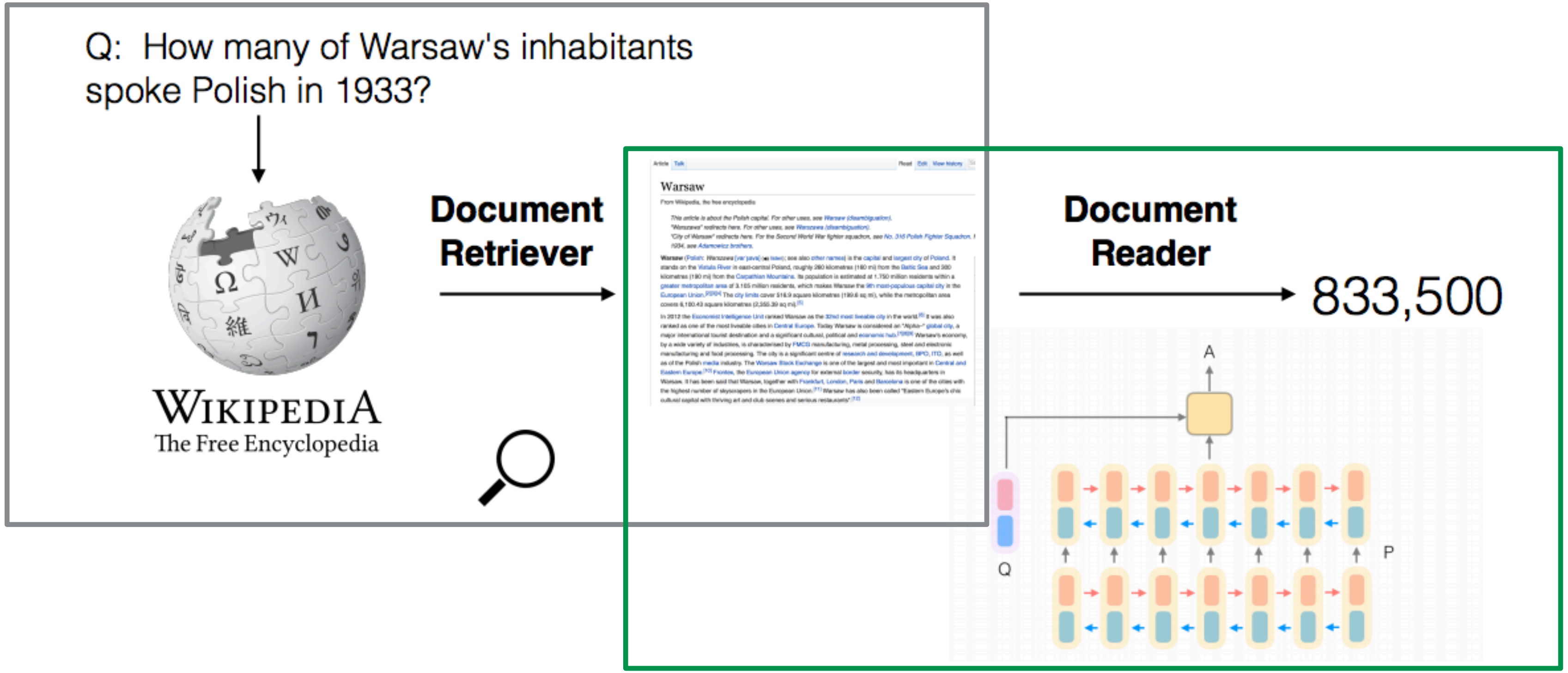
A. Lacey Chabert

Dhingra, Sun, et al., EMNLP 2018

Knowledge Base as a Knowledge Source

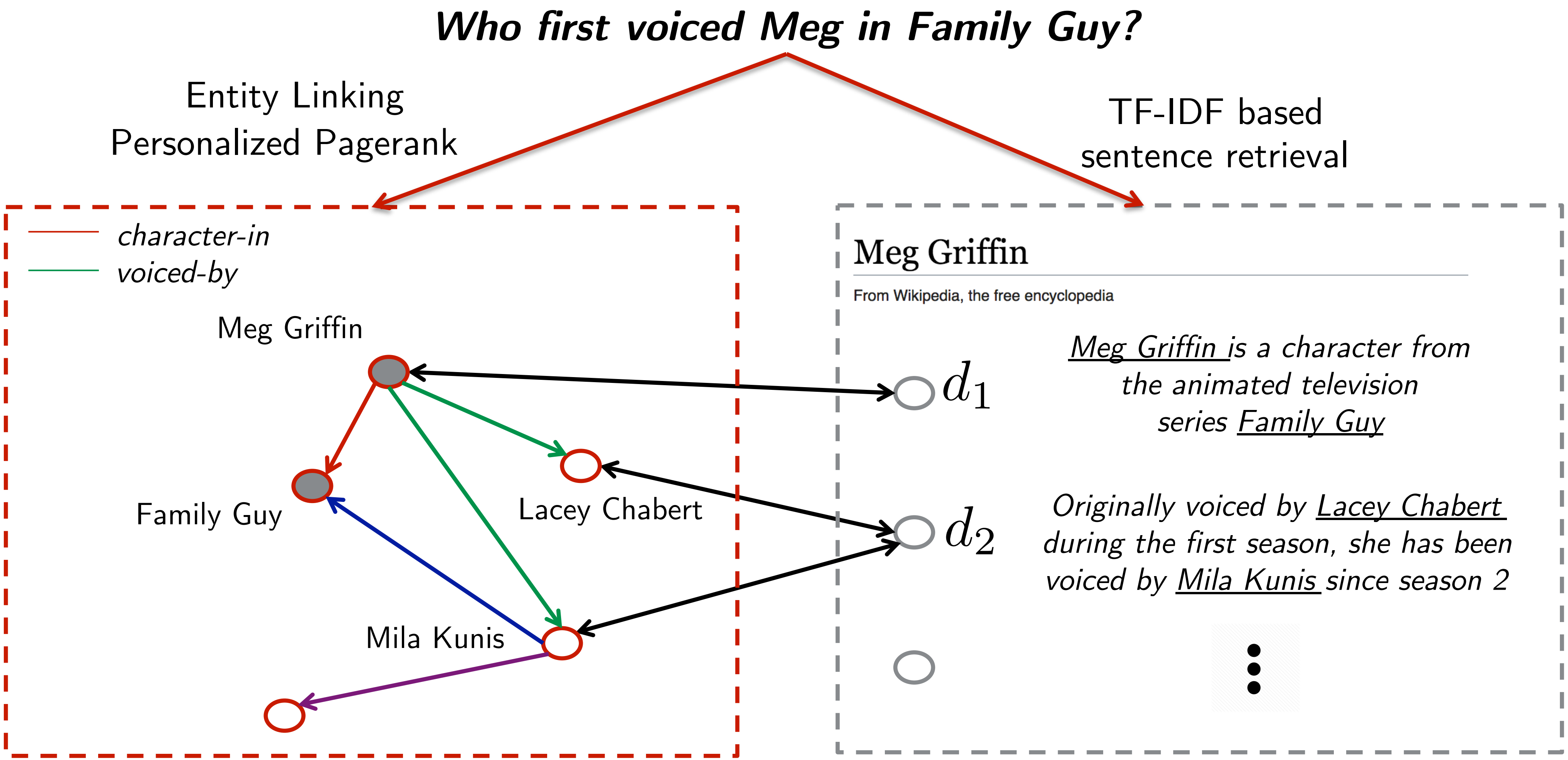


Unstructured Text as a Knowledge Source



Step 1 (Information Retrieval):
Retrieve passages relevant to the Question using shallow methods
Step 2 (Reading Comprehension):
Perform deep reading of passages to extract answers

Text Augmented Knowledge Graph (Dhingra, Sun, et al., 2018)



Dhingra, Sun, et al., EMNLP 2018

Reading Graphs

Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a natural language question $q = (w_1, \dots, w_T)$ learn a function $y_v = f(v) \forall v \in \mathcal{V}$, s.t. $y_v \in \{0, 1\}$ and $y_v = 1$ if and only if v is an answer for q.

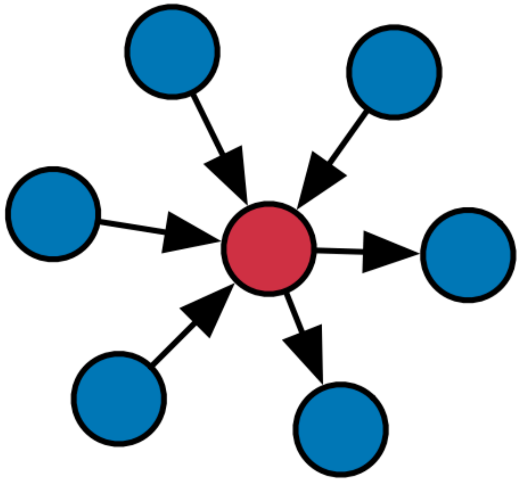
$$P(y_v = 1 | \mathcal{G}, q) = \frac{\exp h_q^T h_v}{\sum_{v'} \exp h_q^T h_{v'}}$$

h_q -- Question Representation from an LSTM

h_v -- Node Representation from a Graph Convolution Network

Dhingra, Sun, et al., EMNLP 2018

Graph Convolution Network



For each v:

Initialize $h_v^{(0)}$

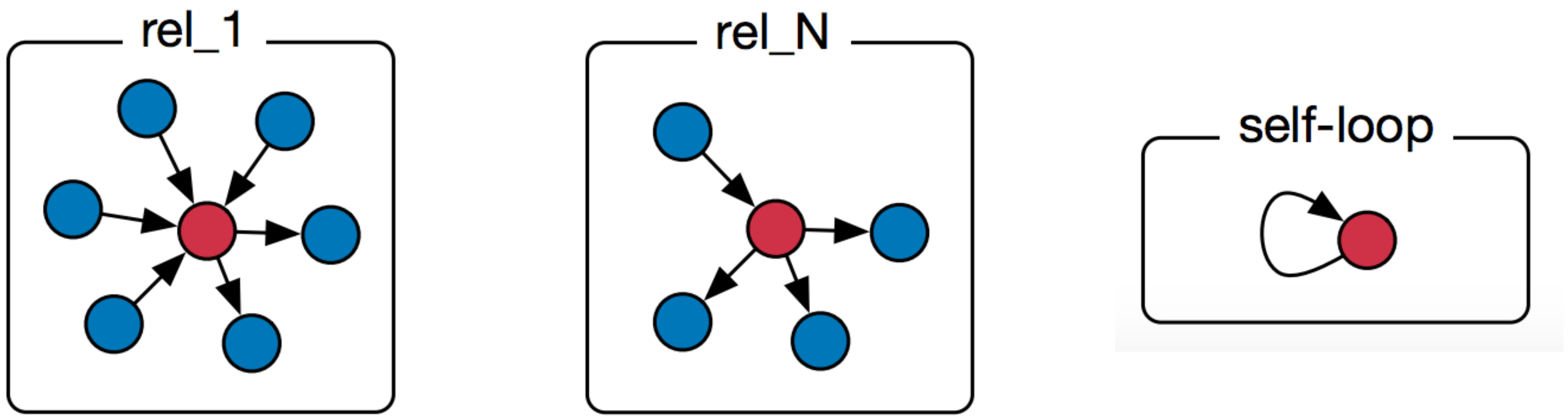
$$h_v^{(t)} = f(W_1 h_v^{(t-1)} + W_2 \sum_{v' \in N(v)} \alpha_{v'} h_{v'}^{(t-1)})$$

Repeat for $t = 1, \ldots, T$

Kipf et al., 2016

Relational Graph Convolution Network

Graphs with edge types



$$h_v^{(t)} = f\left(\sum_r W_1 h_v^{(t-1)} + W_2^r \sum_{v' \in N_r(v)} \alpha_{v'} h_{v'}^{(t-1)}\right)$$

Schlichtkrull et al. 2017

Graph Propagation / Graph Convolution

Entities

$\circ\ e$

Meg Griffin

Lookup Table

$$h_e^{(0)} = L(e) \in \mathbb{R}^p$$

Documents

$\circ\ d$

Meg Griffin is a character from the animated television series Family Guy

BiDirectional LSTM

$$h_d^{(0)} = \mathrm{LSTM}(d_{w_1}, \ldots, d_{w_T}) \in \mathbb{R}^{T \times p}$$

Dhingra, Sun, et al., EMNLP 2018

Graph Propagation / Graph Convolution

Entities — Documents

$e \longrightarrow d$

Meg Griffin — *Meg Griffin is a character from the animated television series Family Guy*

$$h_d^{(t)} = \mathrm{LSTM}(h_{d_1}^{(t-1)} || e_{w_1}^{(t-1)}, \ldots, h_{d_T}^{(t-1)} || e_{w_T}^{(t-1)})$$

$e \longleftarrow d$

Meg Griffin — *Meg Griffin is a character from the animated television series Family Guy*

$$h_e^{(t)} = f(W_1 h_e^{(t-1)} + \sum_r \sum_{v' \in N_r(v)} W_2^r h_{v'}^{(t-1)} + W_3 \sum_{d: e \in d} h_{d_w}^{(t-1)})$$

- Relational information via KB propagation

Dhingra, Sun, et al., EMNLP 2018

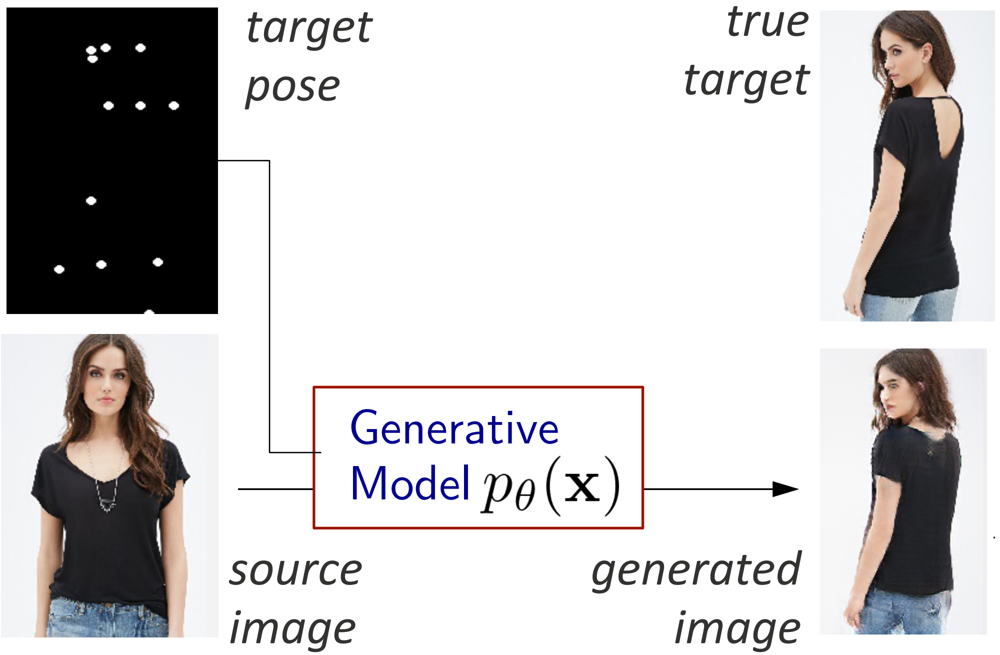
Domain knowledge

- We consider three classes of domain knowledge:
  - Relational
  - Logical (constraints)
  - Scientific

Incorporating Constraints

- Consider a statistical model $\mathbf{x} \sim p_\theta(\mathbf{x})$
- Consider a constraint function, $f_\phi(\mathbf{x}) \in \mathbb{R}$ parameterized by ϕ
  - Higher $f_\phi(\mathbf{x})$ value, better $\mathbf{x}$ w.r.t the knowledge

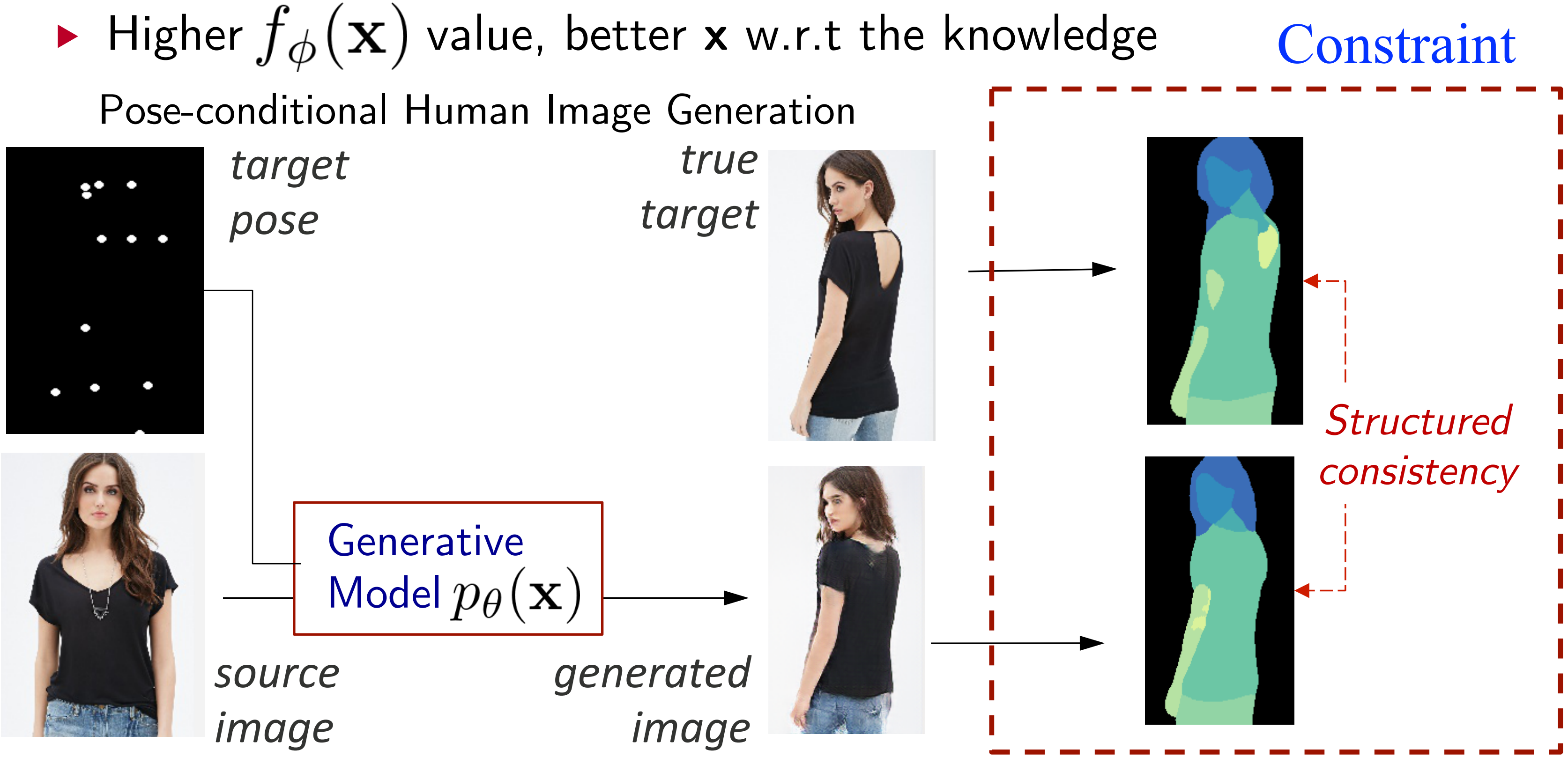
Pose-conditional Human Image Generation



DeepFashion, Liu et.al., CVPR 2016

Incorporating Constraints

- Consider a statistical model $\mathbf{x} \sim p_\theta(\mathbf{x})$
- Consider a constraint function, $f_\phi(\mathbf{x}) \in \mathbb{R}$ parameterized by ϕ
  - Higher $f_\phi(\mathbf{x})$ value, better $\mathbf{x}$ w.r.t the knowledge



DeepFashion, Liu et.al., CVPR 2016

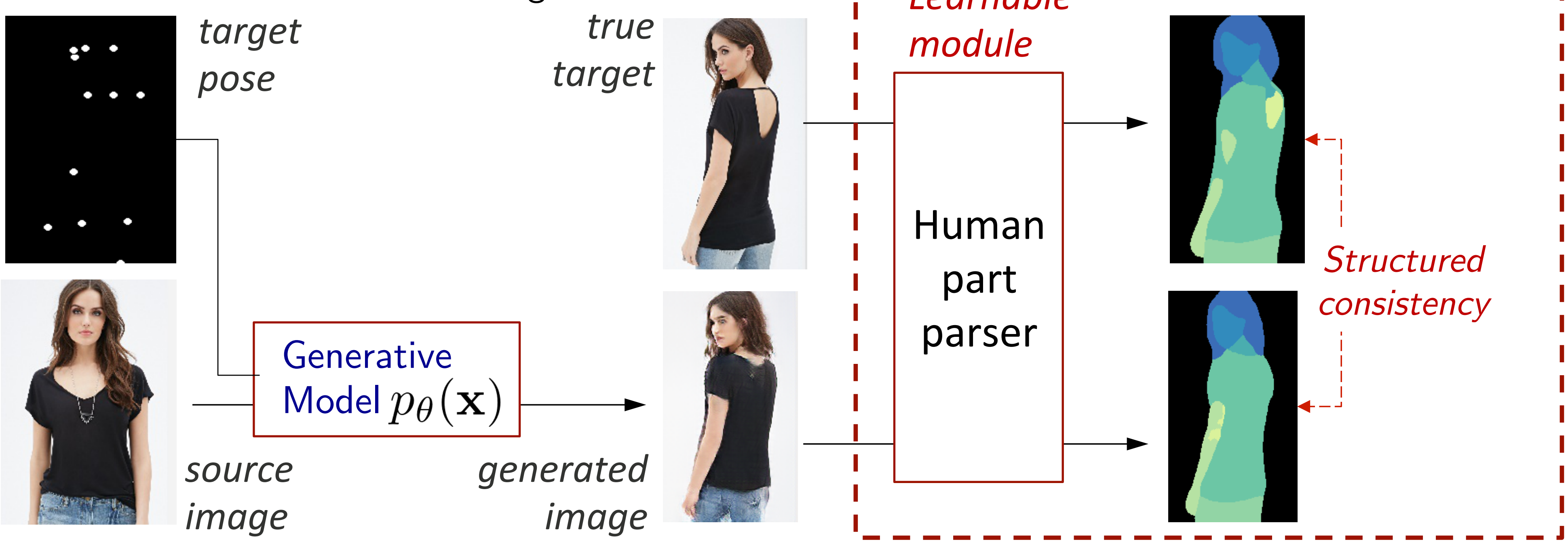
Hu, Yang, et al., NeurIPS 2018

Learning with Constraints

- Consider a statistical model $\mathbf{x} \sim p_\theta(\mathbf{x})$
- Consider a constraint function, $f_\phi(\mathbf{x}) \in \mathbb{R}$ parameterized by ϕ
  - Higher $f_\phi(\mathbf{x})$ value, better $\mathbf{x}$ w.r.t the knowledge

Pose-conditional Human Image Generation

Constraint



DeepFashion, Liu et.al., CVPR 2016

Hu, Yang, et al., NeurIPS 2018

Learning with Constraints

- Consider a statistical model $\mathbf{x} \sim p_\theta(\mathbf{x})$
- Consider a constraint function, $f_\phi(\mathbf{x}) \in \mathbb{R}$ parameterized by ϕ
  - Higher $f_\phi(\mathbf{x})$ value, better $\mathbf{x}$ w.r.t the knowledge
- Sentiment prediction:
  - This was a terrific movie, but the director could have done better
- Logical Rules:
  - Sentence *S* with structure *A-but-B*: => sentiment of *B* dominates

Learning with Constraints

- Consider a statistical model $\mathbf{x} \sim p_\theta(\mathbf{x})$
- Consider a constraint function, $f_\phi(\mathbf{x}) \in \mathbb{R}$ parameterized by ϕ
  - Higher $f_\phi(\mathbf{x})$ value, better **x** w.r.t the knowledge

- One way to impose the constraint is to maximize: $\mathbb{E}_{p_\theta}[f_\phi(\mathbf{x})]$
- Objective:

$$\min_\theta \left(\mathcal{L}(\theta) - \alpha \mathbb{E}_{p_\theta}[f_\phi(\mathbf{x})]\right)$$

Regular objective (e.g. cross-entropy loss, etc.)

Regularization: imposing constraints – difficult to compute

Posterior Regularization (Ganchev et al., 2010)

- Consider a statistical model $\mathbf{x} \sim p_\theta(\mathbf{x})$
- Consider a constraint function, $f_\phi(\mathbf{x}) \in \mathbb{R}$ parameterized by ϕ

$$\min_\theta \left(\mathcal{L}(\theta) - \alpha \boxed{\mathbb{E}_{p_\theta}[f_\phi(\mathbf{x})]}\right)$$

$$\mathcal{L}(\theta, q) = \mathrm{KL}\big(q(\mathbf{x}) || p_\theta(\mathbf{x})\big) - \lambda \mathbb{E}_q[f_\phi(\mathbf{x})]$$

- Introduce variational distribution q, which is encouraged to stay close to p

- Objective:

$$\min_{\theta, q} \left(\mathcal{L}(\theta) + \alpha \mathcal{L}(\theta, q)\right)$$

Posterior Regularization (Ganchev et al., 2010)

$$\min_{\theta,q} \left(\mathcal{L}(\theta) + \alpha \mathcal{L}(\theta, q)\right)$$

$$\mathcal{L}(\theta, q) = \mathrm{KL}\big(q(\mathbf{x}) || p_\theta(\mathbf{x})\big) - \lambda \mathbb{E}_q[f_\phi(\mathbf{x})]$$

- Optimal solution for q:

$$q^*(\mathbf{x}) = p_\theta(\mathbf{x}) \exp\left(\lambda f_\phi(\mathbf{x})\right)/\mathcal{Z}$$

Higher value -- higher probability under q – "soft constraint"

- How do we fit our model parameters θ ?

Logical Rule Formulation (Zhiting Hu et al., 2016)

- Consider a supervised learning: $p_\theta(y|\mathbf{x})$, e.g. deep neural network
- Input-Target space (X,Y)
- First-order logic rules: (r, λ)
  - $r(X, Y) \in [0, 1]$, could be soft
  - λ is the confidence level of the rule

- Within PR framework given l rules

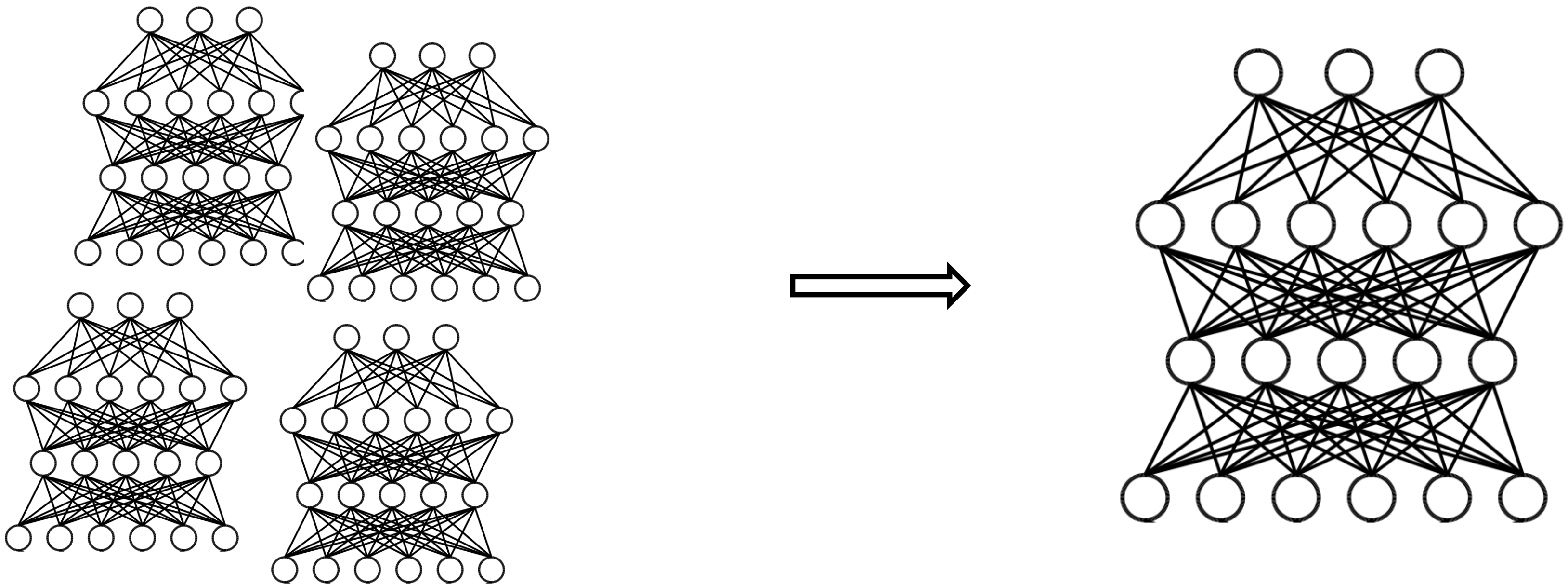
$$q^*(y|\mathbf{x}) = p_\theta(y|\mathbf{x}) \exp\left(\sum_l \lambda_l r_l(y, \mathbf{x})\right) / \mathcal{Z}$$

- How to train a neural network: Knowledge Distillation [Hinton et al., 2015; Bucilu et al., 2006].

Zhiting Hu et.al., ACL 2016

Knowledge Distillation

$q(y|\mathbf{x})$ Match soft predictions of the teacher network and student network $p_\theta(y|\mathbf{x})$



Teacher (Ensemble)

Student

Knowledge Distillation [Hinton et al., 2015; Bucilu et al., 2006].

Rule Knowledge Distillation

- Deep neural network $p_\theta(y|\mathbf{x})$
- Train to imitate the outputs of the rule-regularized teacher network
- At iteration t:

true hard label

soft prediction of $p_\theta(y|\mathbf{x})$

$$\theta^{(t+1)} = \underset{\theta}{\operatorname{argmin}} \frac{1}{N} \sum_{n=1}^{N} \ell(y_n, \sigma_\theta(\mathbf{x})) + \alpha \ell(s_n^{(t)}, \sigma_\theta(\mathbf{x}))$$

balancing parameter

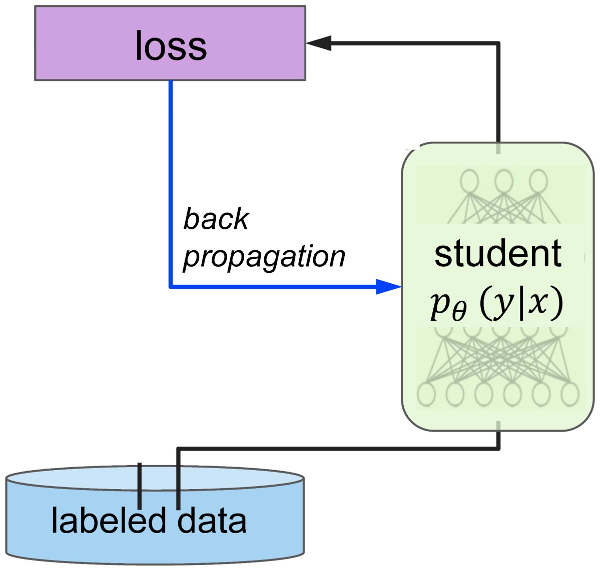
soft prediction of the teacher network q.

$$q^*(y|\mathbf{x}) = p_\theta(y|\mathbf{x}) \exp\left(\sum_l \lambda_l r_l(y, \mathbf{x})\right) / Z$$

Zhiting Hu et.al., ACL 2016

Rule Knowledge Distillation (Zhiting Hu et al., 2016)

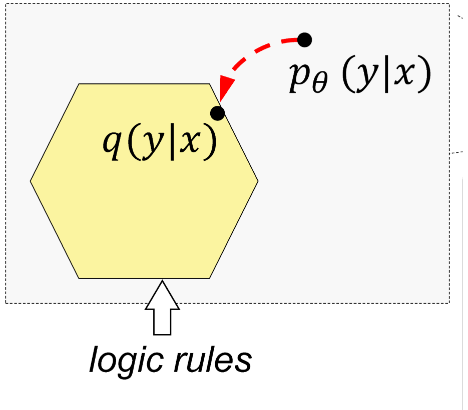
- Deep neural network $p_\theta(y|\mathbf{x})$

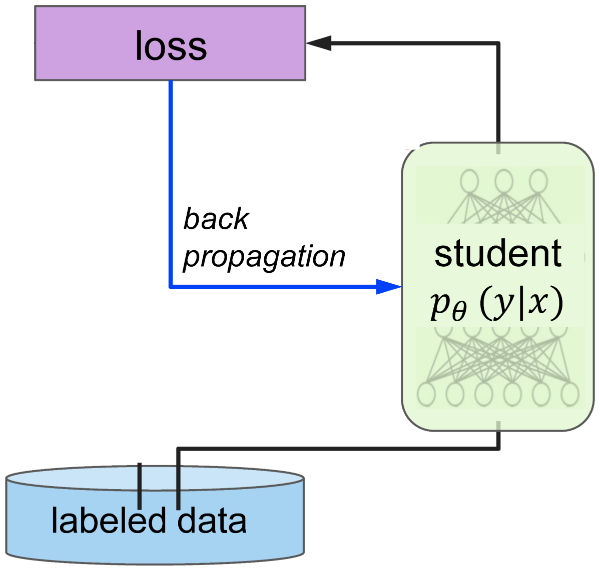


Zhiting Hu et.al., ACL 2016

Rule Knowledge Distillation (Zhiting Hu et al., 2016)

- Deep neural network $p_\theta(y|\mathbf{x})$

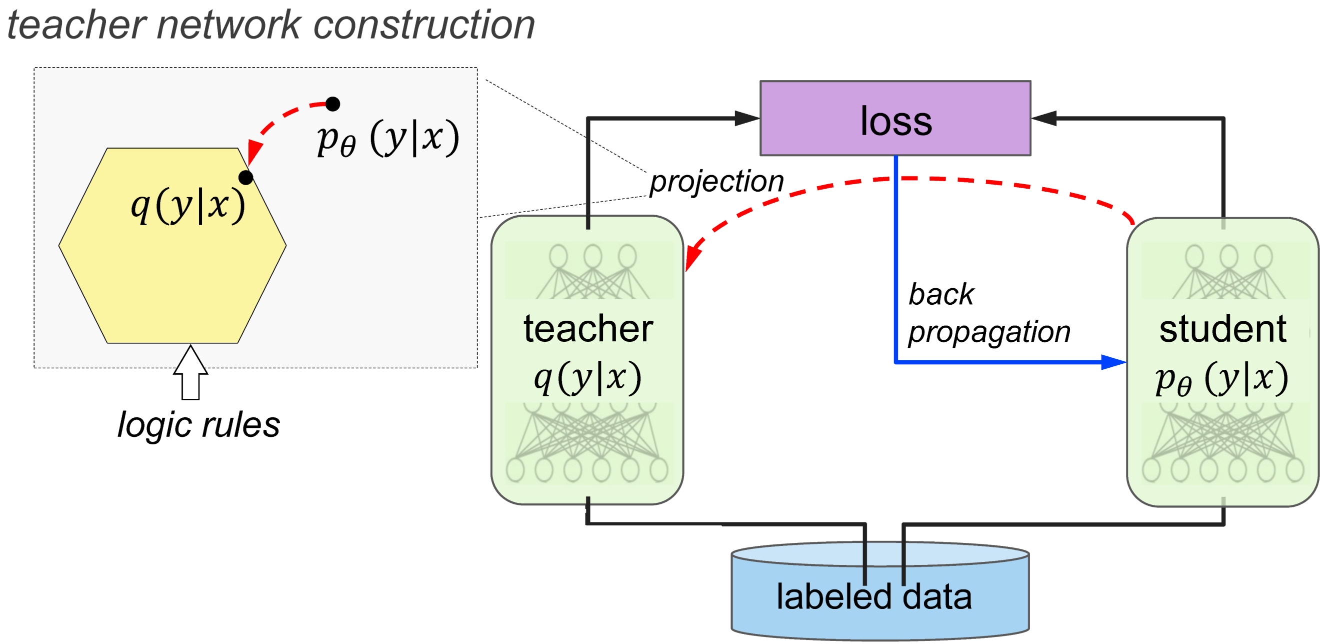




Zhiting Hu et.al., ACL 2016

Rule Knowledge Distillation (Zhiting Hu et al., 2016)

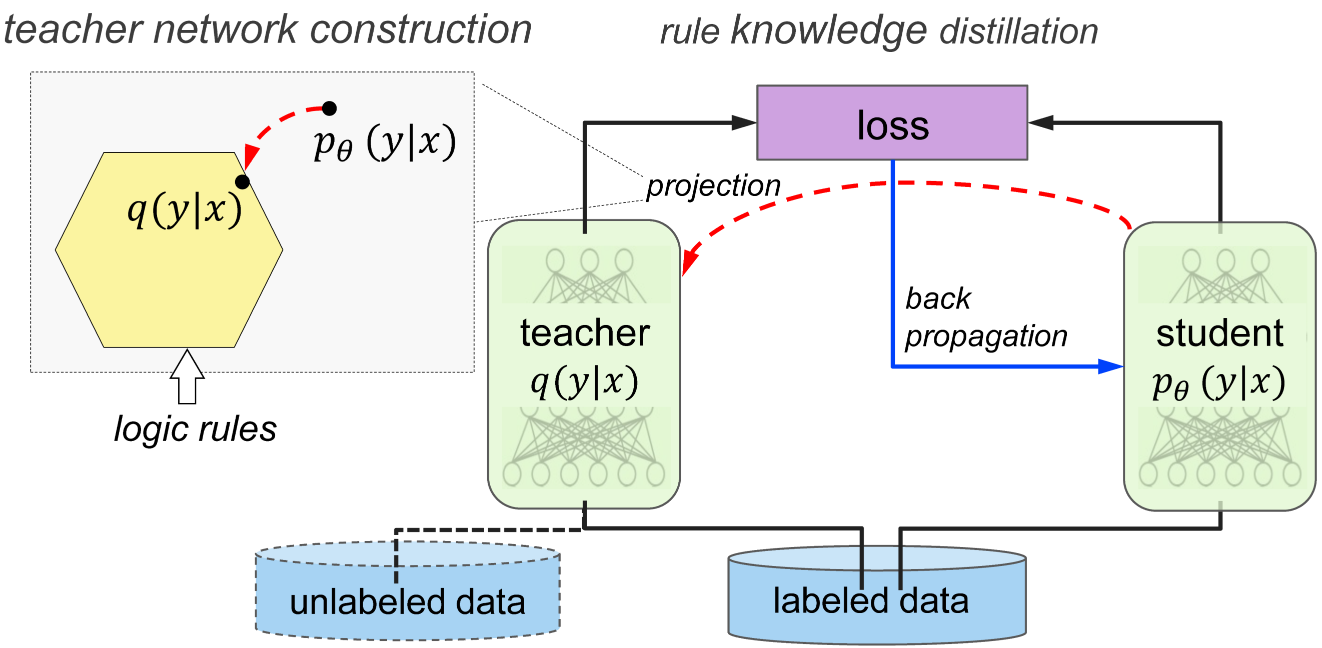
- Deep neural network $p_\theta(y|\mathbf{x})$
- At each iteration:
  - Construct a teacher network q(y|x) with “soft constraints”
  - Train DNN to emulate the teacher network



Zhiting Hu et.al., ACL 2016

Rule Knowledge Distillation (Zhiting Hu et al., 2016)

- Deep neural network $p_\theta(y|\mathbf{x})$
- At each iteration:
  - Construct a teacher network q(y|x) with "soft constraints"
  - Train DNN to emulate the teacher network



- Sentiment classification,
- Named entity recognition

Zhiting Hu et.al., ACL 2016

Learning Rules / Constraints

$$q^*(y|\mathbf{x}) = p_\theta(y|\mathbf{x}) \exp\left(\sum_l \lambda_l r_l(y, \mathbf{x})\right) / \mathcal{Z}$$

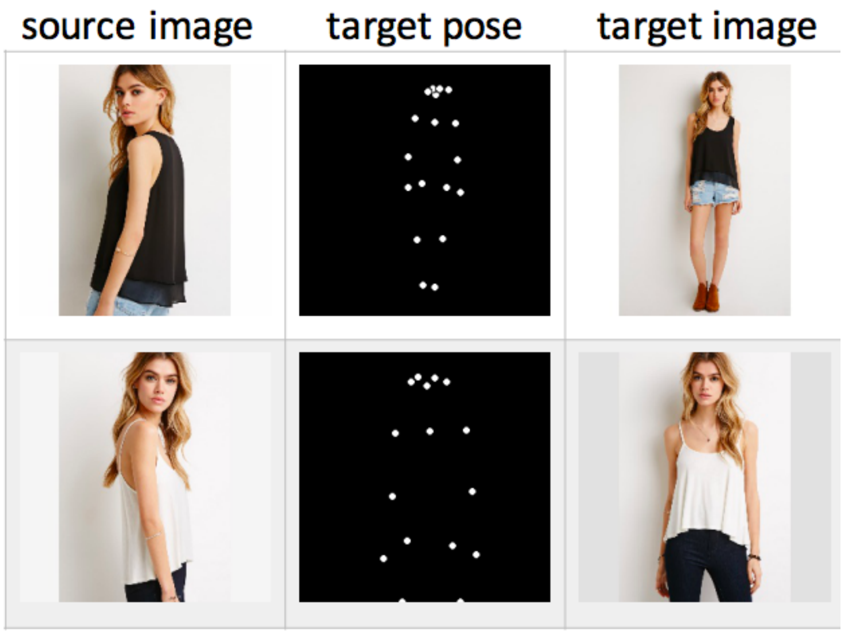
- We can also learn the "confidence" values λ_l for logical rules
- More generally, we can optimize parameters of the constraint function $f_\phi(\mathbf{x})$

$$q^*(\mathbf{x}) = p_\theta(\mathbf{x}) \exp\left(\lambda f_\phi(\mathbf{x})\right) / \mathcal{Z}$$

  - Treat $f_\phi(\mathbf{x})$ as the reward function to be learned within the MaxEnt Inverse Reinforcement Learning

Zhiting Hu et.al., EMNLP 2016, NeurIPS2018

Pose-conditional Human Image Generation



Samples generated by the models. Enforcing learned human part constraint generates correct poses and better preserves human body structure

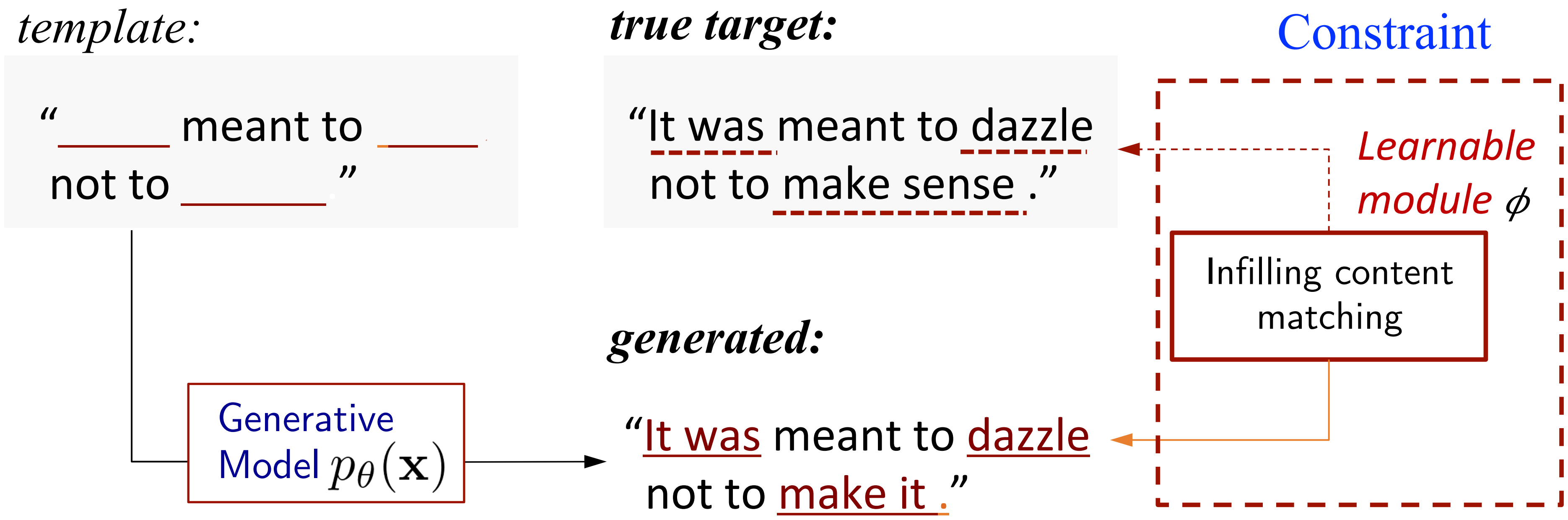
	Method	SSIM	Human
1	Ma et al. [38]	0.614	—
2	Pumarola et al. [44]	0.747	—
3	Ma et al. [37]	0.762	—
4	Base model	0.676	0.03
5	With fixed constraint	0.679	0.12
6	With learned constraint	**0.727**	**0.77**

Results of image generation using Structural Similarity (SSIM) between generated and true images

Hu, Yang, et al., NeurIPS 2018

Template-guided Sentence Generation

- Task: Given a template, generate a complete sentence following the template
- Constraint: force to match between infilling content of the generated sentence with the true content



Hu, Yang, et al., NeurIPS 2018

Template-guided Sentence Generation

	Model	Perplexity	Human
1	Base model	30.30	0.19
2	With binary D	30.01	0.20
3	With constraint updated in M-step (Eq.5)	31.27	0.15
4	With learned constraint	**28.69**	**0.24**

Samples by the full model are considered as of higher quality in 24% cases.

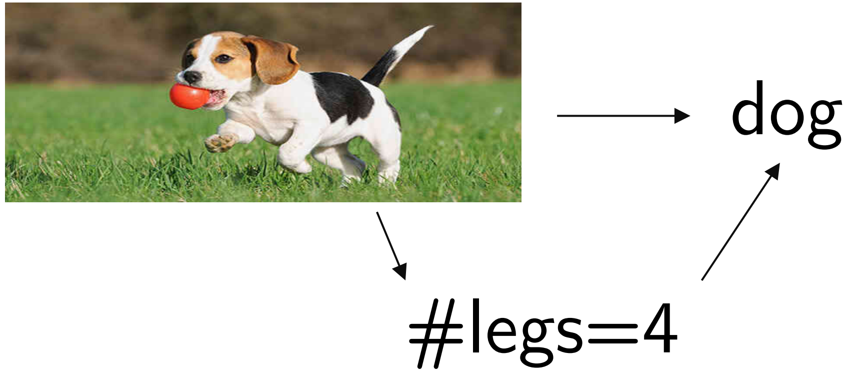
______	acting	______
the	acting	is the acting .
the	acting	is also very good .
______	out of 10 .	
10	out of 10 .	
I will give the movie 7	out of 10 .	

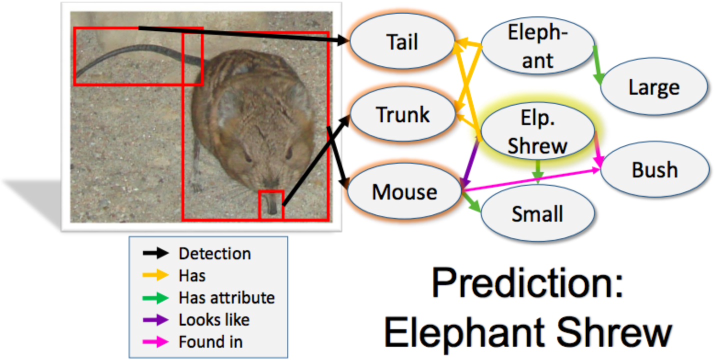
Two test examples, including the template, the sample by the base model, and the sample by the constrained model.

Hu, Yang, et al., NeurIPS 2018

Summary So Far

- Limitations: We considered very simple forms of domain knowledge: relational, logical, simple constraints
- Human Knowledge: abstract, fuzzy, build on high-level concepts
  - e.g. dogs have 4 legs





Example of how semantic knowledge about the world aids classification.

Marino et al., CVPR 2017

- How do we encode this knowledge and how do we efficiently integrate this into deep learning models

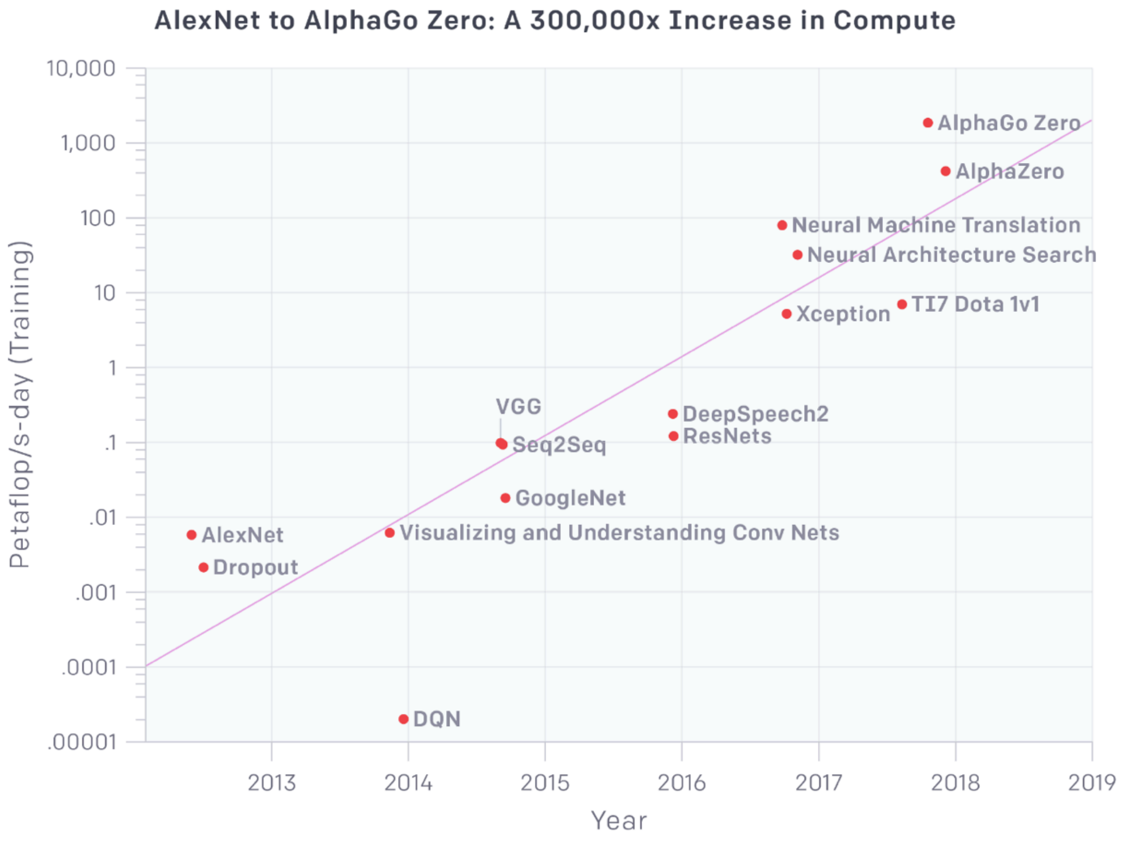
MineRL

Towards Sample Efficient Reinforcement Learning

William H. Guss* , Brandon Houghton* , Nicholay Topin , Phillip Wang , Cayden Codel , Manuela Veloso and Ruslan Salakhutdinov

The growing problem of sample inefficiency in RL

- The **number of environment samples** to train policies on domains of increasing complexity is **growing exponentially**

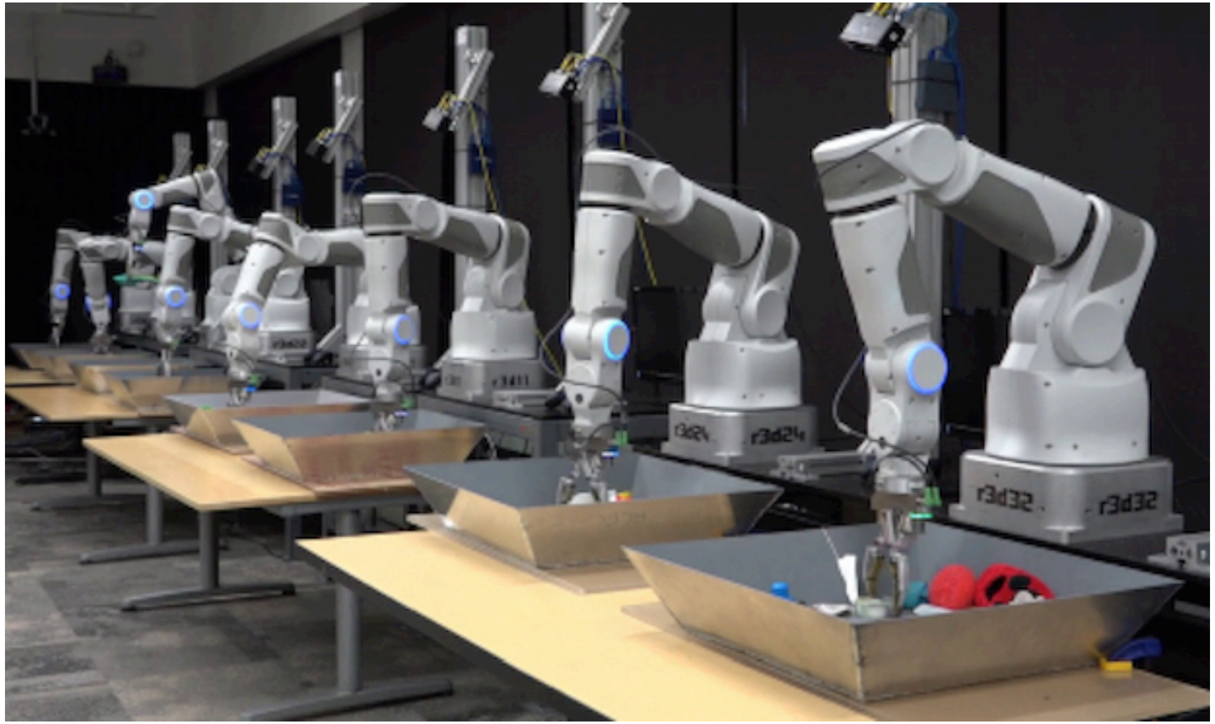


Dario Amodei & Danny Hernandez Open AI 2019.

The growing problem of sample inefficiency in RL

- The **number of environment samples** to train policies on domains of increasing complexity is **growing exponentially**

- Training **complex policies** in real-world environments is quickly becoming **intractable**, without significant infrastructure

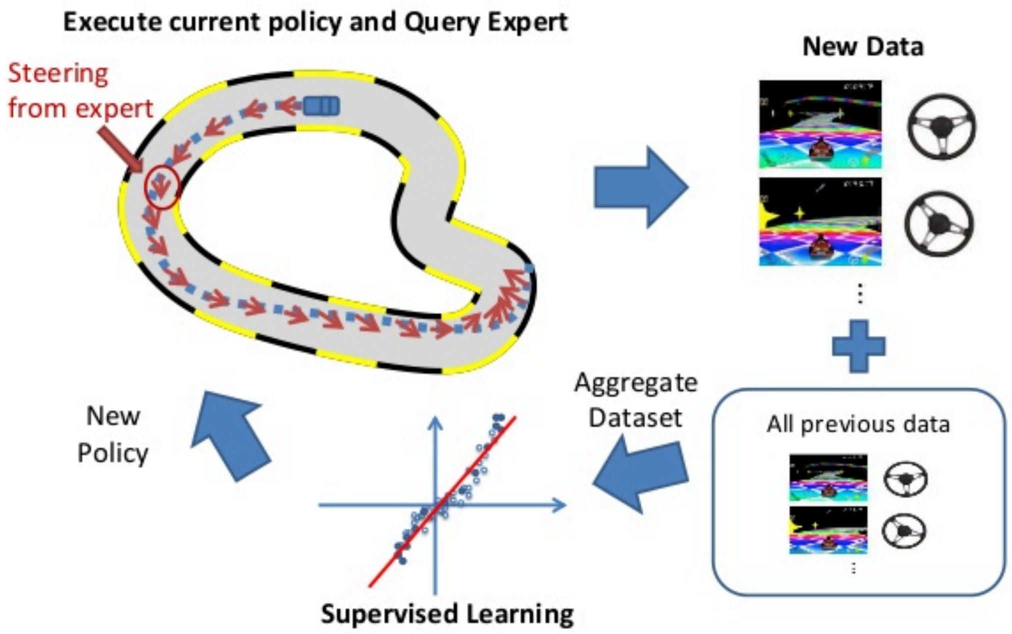


Levine et. al. 2016

Demonstration as an Answer to Sample Inefficiency

- The number of samples required can be drastically reduced using expert demonstrations.
- No open, large-scale dataset of demonstrations across a variety of open/closed world tasks exists

Stéphane. Ross, Geoffrey J. Gordon, and J. Andrew. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In , 2011.



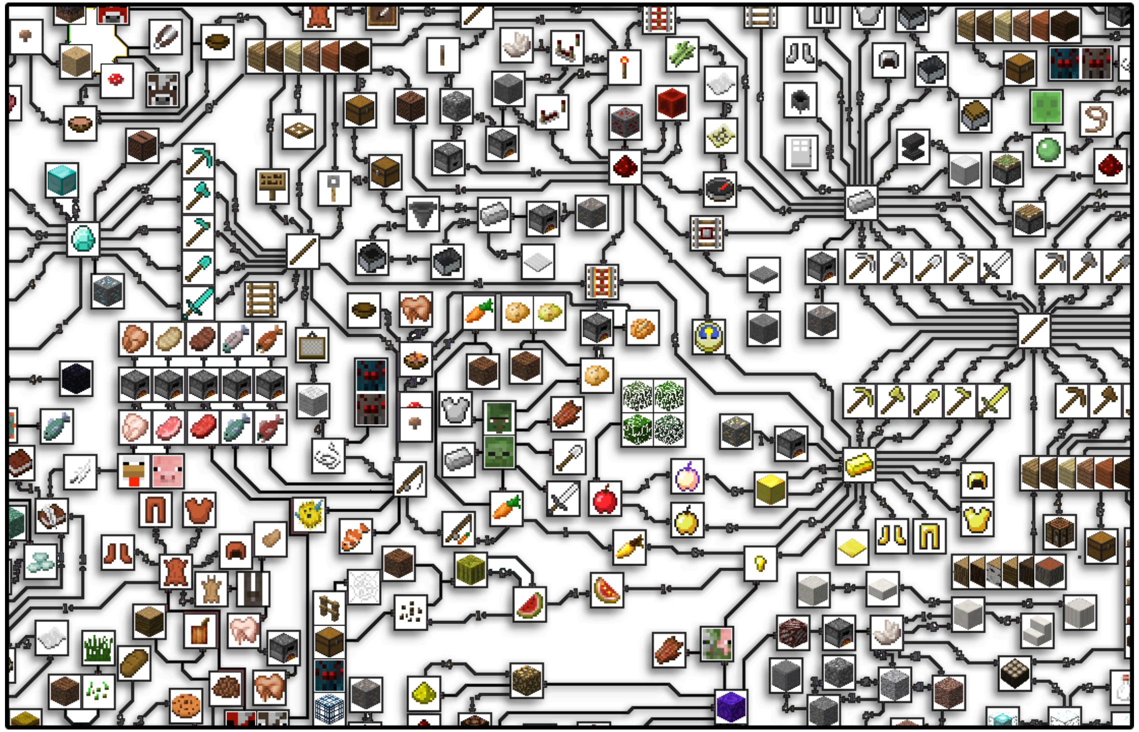
MineRL: A Large-Scale Dataset of Minecraft Demonstrations

William H. Guss∗ , Brandon Houghton∗ , Nicholay Topin , Phillip Wang , Cayden Codel , Manuela Veloso and **Ruslan Salakhutdinov.** IJCAI 2019.

- We have created one of the largest imitation learning datasets with over **60 million frames of recorded** human player data across **6+** complex tasks in Minecraft.

MineRL: Why Minecraft?

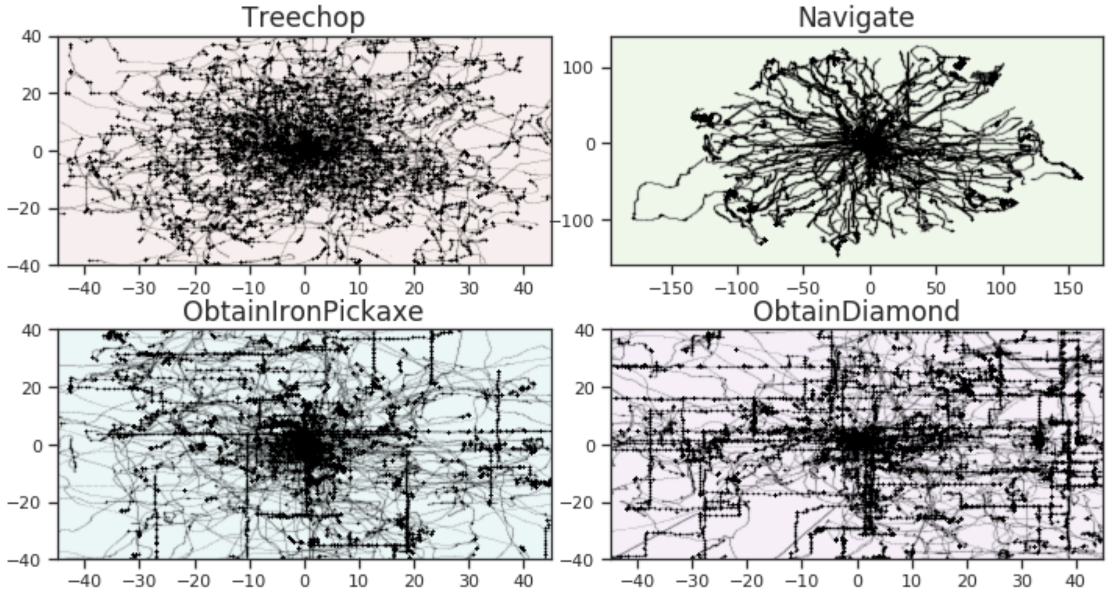
- Open-world, infinite/procedurally generated
- Sparse & dense rewards
- Many innate task hierarchies and subgoals
- Encompasses many of problems we must solve as we approach the problem of general AI.



A glimpse into the Minecraft item hierarchy

MineRL: Dataset Details

- Consists of over 500+ hours of human demonstrations over 1000+ unique player sessions.
- Rich set of annotations including: **subtask completion, rewards, player meta-data, gamestate.**
- **Rerenderable!** We record game-state not just player-pixels

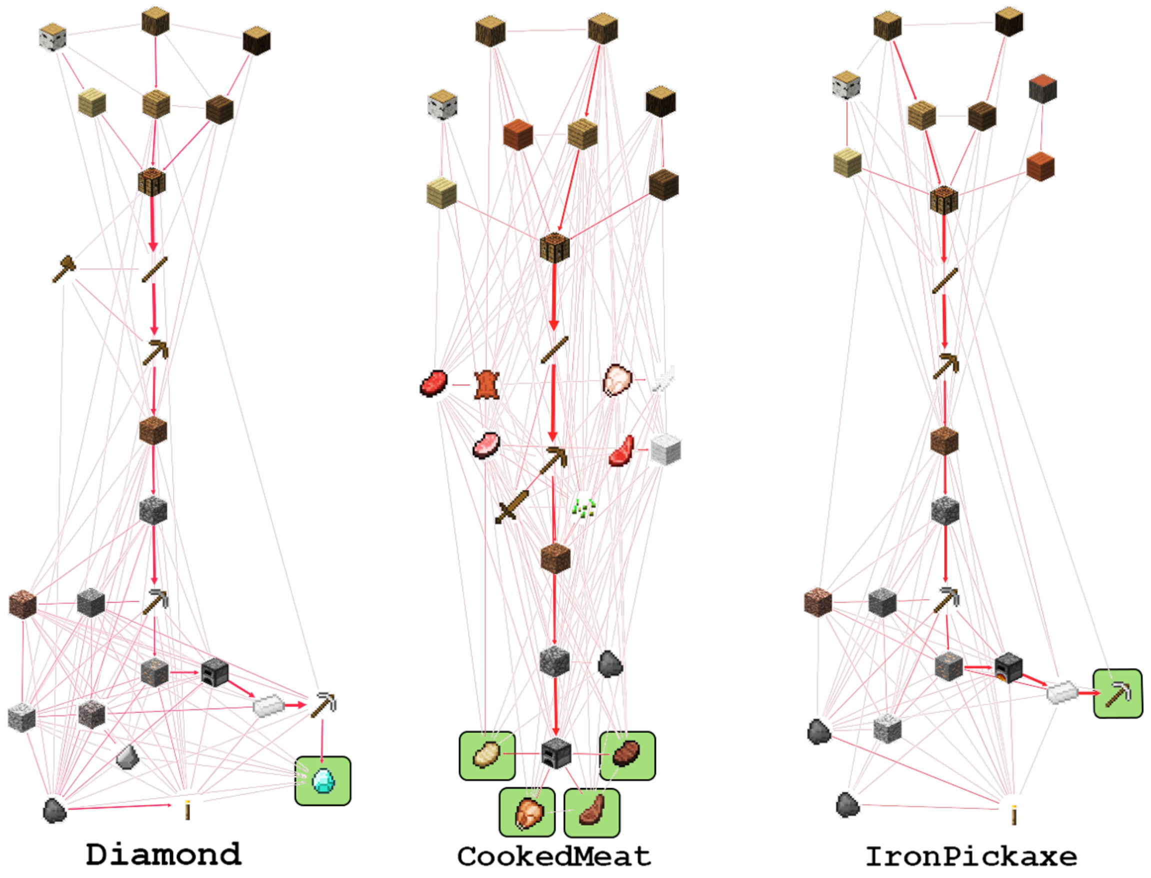


Plots of XY positions of players in several tasks (diversity & rich annotations)

MineRL: Hierarchality of Data

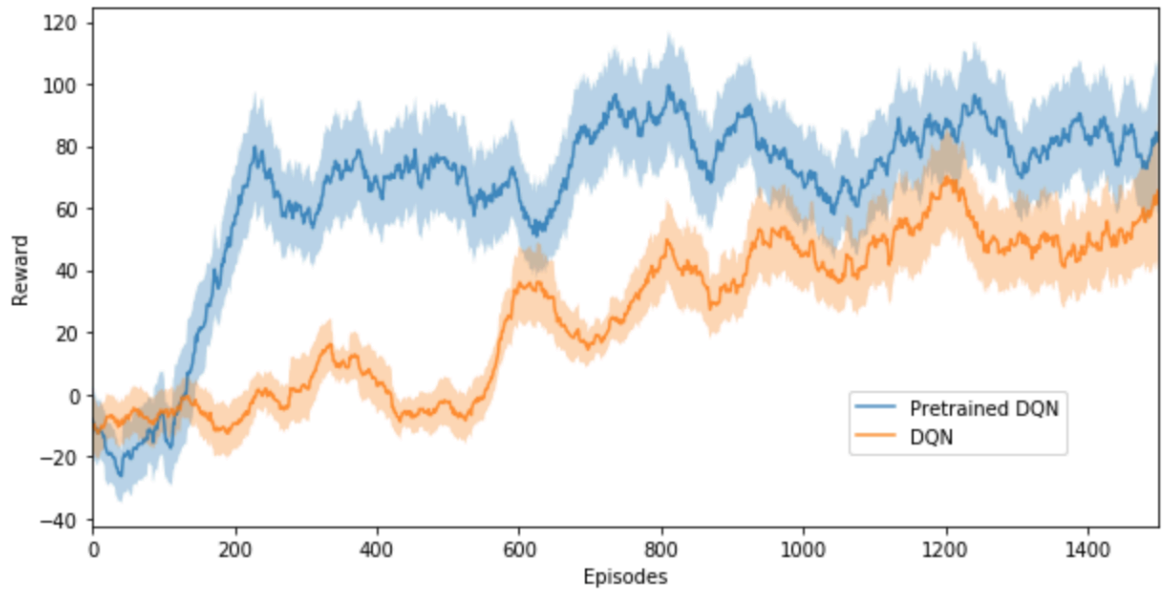
- Players complete sparsely rewarded tasks **following a specific task hierarchy/dependence graph.**

- Many ways to obtain an item, but data exhibits the existence of **canonical pathways.**



MineRL: Expert demonstrations help

- On the **Navigate** task, using the MineRL-v0 dataset helps **drastically reduce** the number of samples for standard algorithms.
- However, **better algorithms still need to be developed**, especially for the long-term, hierarchical tasks exhibited in Minecraft.



MineRL: NeurIPS 2019 Competition

- To foster research in this direction, we are hosting the **MineRL Competition on Sample Efficient Reinforcement Learning** at NeurIPS 2019**!**

- Competitors must learn to obtain a diamond in under 4-days of training.

https://www.youtube.com/watch?v=KFMuI4TfC7c

MineRL: Get started now!

http://minerl.io/

Thank you