Unsupervised Recurrent Neural Network Grammars

Yoon Kim Alexander Rush Lei Yu
Adhiguna Kuncoro Chris Dyer Gábor Melis

Code: `https://github.com/harvardnlp/urnng`

Language Modeling & Grammar Induction

- Goal of **Language Modeling**: assign high likelihood to held-out data
- Goal of **Grammar Induction**: learn linguistically meaningful tree structures without supervision
- Incompatible?
  - For good language modeling performance, need little independence assumptions and make use of flexible models (e.g. deep networks)
  - For grammar induction, need strong independence assumptions for tractable training and to imbue inductive bias (e.g. context-freeness grammars)

Language Modeling & Grammar Induction

- Goal of **Language Modeling**: assign high likelihood to held-out data
- Goal of **Grammar Induction**: learn linguistically meaningful tree structures without supervision
- Incompatible?
  - For good language modeling performance, need little independence assumptions and make use of flexible models (e.g. deep networks)
  - For grammar induction, need strong independence assumptions for tractable training and to imbue inductive bias (e.g. context-freeness grammars)

This Work: Unsupervised Recurrent Neural Network Grammars

- Use a flexible generative model without any explicit independence assumptions (RNNG) $\implies$ good LM performance
- Variational inference with a structured inference network (CRF parser) to regularize the posterior $\implies$ learn linguistically meaningful trees

Background: Recurrent Neural Network Grammars [Dyer et al. 2016]

- Structured joint generative model of sentence $\mathbf{x}$ and tree $\mathbf{z}$

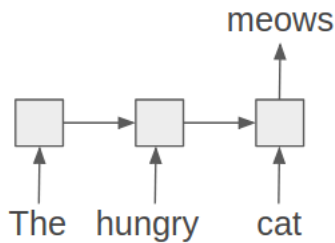
$$p_\theta(\mathbf{x}, \mathbf{z})$$

- Generate next word conditioned on partially-completed syntax tree
- Hierarchical generative process (cf. flat generative process of RNN)

Background: Recurrent Neural Network Language Models

Standard RNNLMs: flat left-to-right generation

$$x_t \sim p_\theta(x \mid x_1, \ldots, x_{t-1}) = \text{softmax}(\mathbf{W}\mathbf{h}_{t-1} + \mathbf{b})$$



Background: RNNG [Dyer et al. 2016]

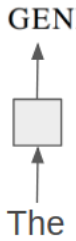
Introduce binary variables $\mathbf{z} = [z_1, \ldots, z_{2T-1}]$ (unlabeled binary tree)

Sample action $z_t \in \{\text{GENERATE}, \text{REDUCE}\}$ at each time step:

$$z_t \sim \text{Bernoulli}(p_t) \qquad p_t = \sigma(\mathbf{w}^\top \mathbf{h}_{\text{prev}} + b)$$

If $z_t = \text{GENERATE}$

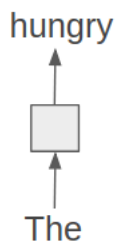
Sample word from context representation



Background: RNNG [Dyer et al. 2016]

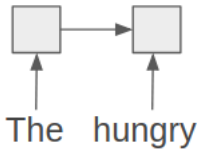
(Similar to standard RNNLMs)

$$x \sim \text{softmax}(\mathbf{W}\mathbf{h}_{\text{prev}} + \mathbf{b})$$



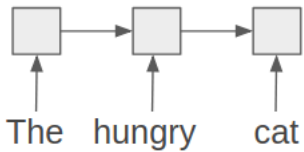
Obtain new context representation with $\mathbf{e}_{\text{hungry}}$

$$\mathbf{h}_{\text{new}} = \text{LSTM}(\mathbf{e}_{\text{hungry}}, \mathbf{h}_{\text{prev}})$$

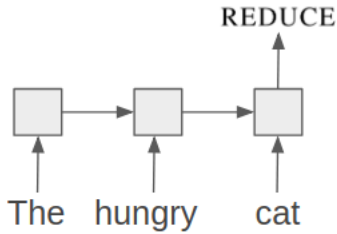


Background: RNNG [Dyer et al. 2016]

$$\mathbf{h}_{\text{new}} = \text{LSTM}(\mathbf{e}_{\text{cat}}, \mathbf{h}_{\text{prev}})$$

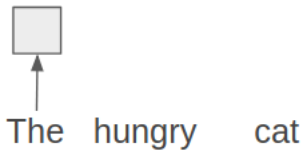


If $z_t = \text{REDUCE}$



If $z_t = \text{REDUCE}$

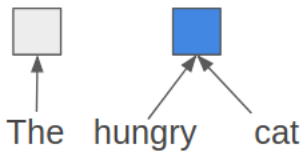
Pop last two elements



Background: RNNG [Dyer et al. 2016]

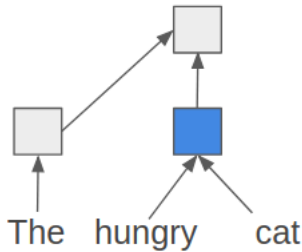
Obtain new representation of constituent

$$\mathbf{e}_{(\text{hungry cat})} = \text{TreeLSTM}(\mathbf{e}_{\text{hungry}}, \mathbf{e}_{\text{cat}})$$



Move the new representation onto the stack

$$\mathbf{h}_{\text{new}} = \text{LSTM}(\mathbf{e}_{(\text{hungry cat})}, \mathbf{h}_{\text{prev}})$$



Background: RNNG [Dyer et al. 2016]

Different inductive biases from RNN LMs $\implies$ learn different generalizations about the observed sequence of terminal symbols in language

- Lower perplexity than neural language models [Dyer et al. 2016]
- Better at syntactic evaluation tasks (e.g. grammaticality judgment) [Kuncoro et al. 2018; Wilcox et al. 2019]
- Correlate with electrophysiological responses in the brain [Hale et al. 2018]

(All require supervised training on annotated treebanks)

Unsupervised Recurrent Neural Network Grammars

- RNNG as a tool to learn structured, syntax-aware generative model of language
- Variational inference for tractable training and to imbue inductive bias

URNNG: Issues

Approach to unsupervised learning: maximize log marginal likelihood

$$\log p_\theta(\mathbf{x}) = \log \sum_{\mathbf{z} \in \mathcal{Z}_T} p_\theta(\mathbf{x}, \mathbf{z})$$

Intractability

- $\mathcal{Z}_T$: exponentially large space
- No dynamic program

$$z_j \sim p_\theta(z \mid \mathbf{x}_{\text{all previous words}}, \mathbf{z}_{\text{all previous actions}})$$

URNNG: Issues

Approach to unsupervised learning: maximize log marginal likelihood

$$\log p_\theta(\mathbf{x}) = \log \sum_{\mathbf{z} \in \mathcal{Z}_T} p_\theta(\mathbf{x}, \mathbf{z})$$

Intractability

- $\mathcal{Z}_T$: exponentially large space
- No dynamic program

$$z_j \sim p_\theta(z \mid \mathbf{x}_{\text{all previous words}}, \mathbf{z}_{\text{all previous actions}})$$

URNNG: Issues

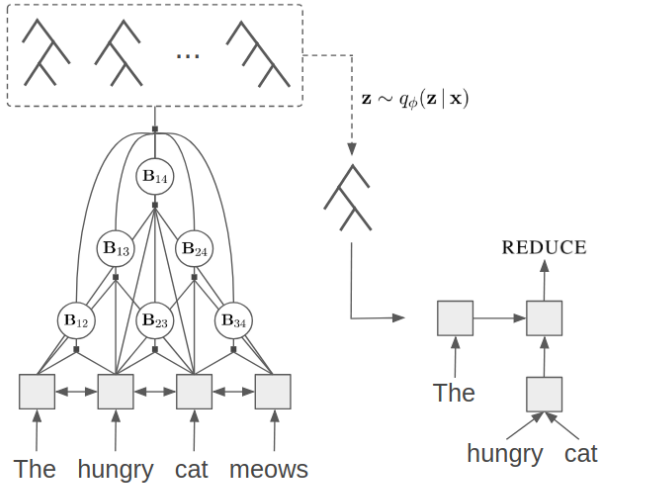
Approach to unsupervised learning: maximize log marginal likelihood

$$\log p_\theta(\mathbf{x}) = \log \sum_{\mathbf{z} \in \mathcal{Z}_T} p_\theta(\mathbf{x}, \mathbf{z})$$

Unconstrained Latent Space

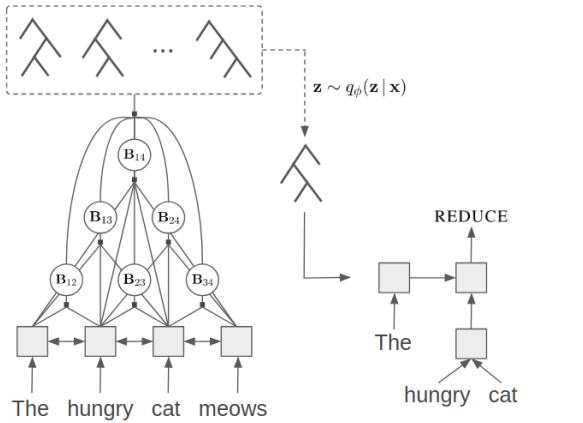
- Little **inductive bias** for meaningful trees to emerge through maximizing likelihood (cf. PCFGs)
- Preliminary experiments on exhaustive marginalization on short sentences (length $<$ 10) were not successful

URNNG: Overview



Inference Network $q_\phi(\mathbf{z} \mid \mathbf{x})$ Generative Model $p_\theta(\mathbf{x}, \mathbf{z})$

URNNG: Tractable Training

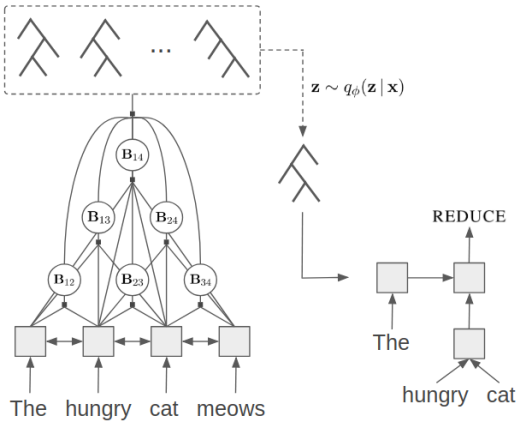


Inference Network $q_\phi(\mathbf{z} \mid \mathbf{x})$ Generative Model $p_\theta(\mathbf{x}, \mathbf{z})$

Tractability

$$\log p_\theta(\mathbf{x}) \geq \mathbb{E}_{q_\phi(\mathbf{z} \mid \mathbf{x})} \left[\log \frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\phi(\mathbf{z} \mid \mathbf{x})} \right]$$
$$= \mathrm{ELBO}(\theta, \phi; \mathbf{x})$$

- Define variational posterior $q_\phi(\mathbf{z} \mid \mathbf{x})$ with an inference network ϕ
- Maximize lower bound on $\log p_\theta(\mathbf{x})$ with sampled gradient estimators



Inference Network $q_\phi(\mathbf{z} \mid \mathbf{x})$ Generative Model $p_\theta(\mathbf{x}, \mathbf{z})$

Unconstrained Latent Space

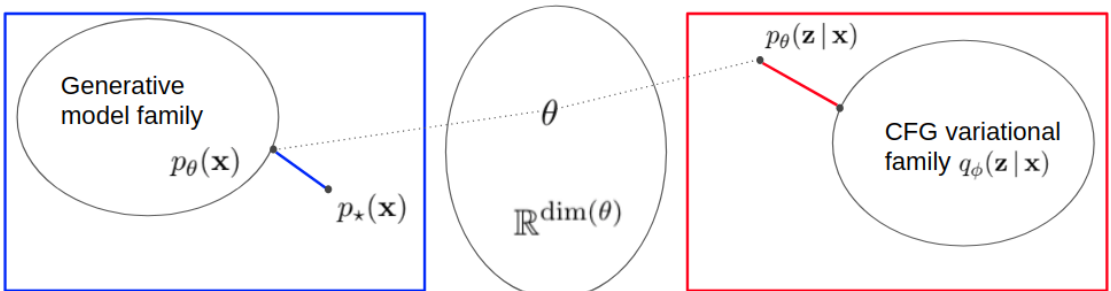
$$\max_\theta \mathrm{ELBO}(\theta, \phi; \mathbf{x}) =$$

$$\min_\theta \; -\log p_\theta(\mathbf{x}) + \mathrm{KL}[q_\phi(\mathbf{z} \mid \mathbf{x}) \,\|\, p_\theta(\mathbf{z} \mid \mathbf{x})]$$

- Structured inference network with context-free assumptions (CRF parser)
- Combination of language modeling and posterior regularization objectives

Posterior Regularization [Ganchev et al. 2010]

$$\min_{\theta} \; -\log p_\theta(\mathbf{x}) + \mathrm{KL}[q_\phi(\mathbf{z} \mid \mathbf{x}) \,||\, p_\theta(\mathbf{z} \mid \mathbf{x})]$$



Inference Network Parameterization

Inference network: CRF constituency parser [Finkel et al. 2008; Durrett and Klein 2015]

- Bidirectional LSTM over $\mathbf{x}$ to get hidden states

$$\overrightarrow{\mathbf{h}}, \overleftarrow{\mathbf{h}} = \mathrm{BiLSTM}(\mathbf{x})$$

- Score $s_{ij} \in \mathbb{R}$ for an unlabeled constituent spanning x_i to x_j

$$s_{ij} = \mathrm{MLP}([\overrightarrow{\mathbf{h}}_{j+1} - \overrightarrow{\mathbf{h}}_i; \overleftarrow{\mathbf{h}}_{i-1} - \overleftarrow{\mathbf{h}}_j])$$

- Similar score parameterization to recent works [Wang and Chang 2016; Stern et al. 2017; Kitaev and Klein 2018]

Training

$$\begin{aligned}\mathrm{ELBO}(\theta, \phi;\, \mathbf{x}) &= \mathbb{E}_{q_\phi(\mathbf{z}\,|\,\mathbf{x})}\left[\log \frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\phi(\mathbf{z}\,|\,\mathbf{x})}\right] \\ &= \mathbb{E}_{q_\phi(\mathbf{z}\,|\,\mathbf{x})}[\log p_\theta(\mathbf{x}, \mathbf{z})] + \mathbb{H}[q_\phi(\mathbf{z}\,|\,\mathbf{x})]\end{aligned}$$

Gradient-based optimization with Monte Carlo estimators

$$\begin{aligned}\nabla_\theta \,\mathrm{ELBO}(\theta, \phi;\, \mathbf{x}) &= \mathbb{E}_{q_\phi(\mathbf{z}\,|\,\mathbf{x})}[\nabla_\theta \log p(\mathbf{x}, \mathbf{z})] \\ \nabla_\phi \,\mathrm{ELBO}(\theta, \phi;\, \mathbf{x}) &= \nabla_\phi \mathbb{E}_{q_\phi(\mathbf{z}\,|\,\mathbf{x})}\left[\log \frac{p_\theta(\mathbf{x}, \mathbf{z})}{q_\phi(\mathbf{z}\,|\,\mathbf{x})}\right] \\ &= \underbrace{\mathbb{E}_{q_\phi(\mathbf{z}\,|\,\mathbf{x})}[\log p_\theta(\mathbf{x}, \mathbf{z}) \nabla_\phi \log q_\phi(\mathbf{z}\,|\,\mathbf{x})]}_{\text{score function gradient estimator}} + \underbrace{\nabla_\phi \mathbb{H}[q_\phi(\mathbf{z}\,|\,\mathbf{x})]}_{O(T^3)\text{ dynamic program}}\end{aligned}$$

Sampling from $q_\phi(\mathbf{z}\,|\,\mathbf{x})$ with forward-filtering backward-sampling in $O(T^3)$

Experimental Setup

- Tasks and Evaluation
  - Language Modeling: Perplexity
  - Unsupervised Parsing: Unlabeled F_1
- Data
  - English: Penn Treebank (40K sents, 24K word types). Different from standard LM setup from Mikolov et al. [2010].
  - Chinese: Chinese Treebank (15K sents, 17K word types)
  - Preprocessing: Singletons replaced with UNK. Punctuation is retained

Experimental Setup: Baselines

- LSTM Language Model: same size as the RNNG
- Parsing Predict Reading Network (PRPN) [Shen et al. 2018]: neural language model with gated layers to induce binary trees
- Supervised RNNG: RNNG trained on binarized gold trees

Language Modeling

	Perplexity	
Model	PTB	CTB
LSTM LM	93.2	201.3

Language Modeling

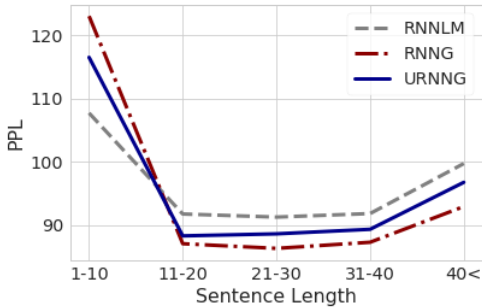
Model	Perplexity PTB	CTB
LSTM LM	93.2	201.3
PRPN (default)	126.2	290.9
PRPN (tuned)	96.7	216.0

Language Modeling

Model	Perplexity PTB	CTB
LSTM LM	93.2	201.3
PRPN (default)	126.2	290.9
PRPN (tuned)	96.7	216.0
Unsupervised RNNG	90.6	195.7
Supervised RNNG	88.7	193.1

Language Modeling

Perplexity on PTB by Sentence Length



Grammar Induction

Unlabeled F_1 with `evalb`

Model	Unlabeled F_1 PTB	CTB
Right Branching Trees	34.8	20.6
Random Trees	17.0	17.4
PRPN (default)	32.9	32.9
PRPN (tuned)	41.2	36.1
Unsupervised RNNG	40.7	29.1
Oracle Binary Trees	82.5	88.6

Grammar Induction

Using evaluation setup from Drozdov et al. [2019]

	F_1	+PP Heuristic
PRPN-LM [Shen et al. 2018]	42.8	42.4
ON-LSTM [Shen et al. 2019]	49.4	–
DIORA [Drozdov et al. 2019]	49.6	56.2
PRPN (tuned)	49.0	49.9
Unsupervised RNNG	52.4	52.4

+PP Heuristic attaches trailing punctuation directly to root

Grammar Induction

Label Recall

Label	URNNG	PRPN
SBAR	74.8%	28.9%
NP	39.5%	63.9%
VP	76.6%	27.3%
PP	55.8%	55.1%
ADJP	33.9%	42.5%
ADVP	50.4%	45.1%

Syntactic Evaluation [Marvin and Linzen 2018]

Two minimally different sentences:

The senators near the assistant are old

*The senators near the assistant is old

Model must assign higher probability to the correct one

	RNNLM	PRPN	URNNG	RNNG
Perplexity	93.2	96.7	90.6	88.7
Syntactic Eval.	62.5%	61.9%	64.6%	69.3%

Syntactic Evaluation [Marvin and Linzen 2018]

Two minimally different sentences:

The senators near the assistant are old

*The senators near the assistant is old

Model must assign higher probability to the correct one

	RNNLM	PRPN	URNNG	RNNG
Perplexity	93.2	96.7	90.6	88.7
Syntactic Eval.	62.5%	61.9%	64.6%	69.3%

Limitations

- Unable to improve on right-branching baseline on unpunctuated corpus
- Slower to train due to the $O(T^3)$ dynamic program and multiple samples for gradient estimators
- Requires various optimization strategies: KL annealing, different optimizers for θ and ϕ, etc.

Conclusion

- Flexible generative model + structured inference network = low perplexity + meaningful structure
- Role of language structure & latent variable modeling in deep learning?

Andrew Drozdov, Patrick Verga, Mohit Yadev, Mohit Iyyer, and Andrew McCallum. 2019. Unsupervised Latent Tree Induction with Deep Inside-Outside Recursive Auto-Encoders. In *Proceedings of NAACL*.

Greg Durrett and Dan Klein. 2015. Neural CRF Parsing. In *Proceedings of ACL*.

Chris Dyer, Adhiguna Kuncoro, Miguel Ballesteros, and Noah A. Smith. 2016. Recurrent Neural Network Grammars. In *Proceedings of NAACL*.

Jenny Rose Finkel, Alex Kleeman, and Christopher D. Manning. 2008. Efficient, Feature-based, Conditional Random Field Parsing. In *Proceedings of ACL*.

Kuzman Ganchev, João Graça, Jennifer Gillenwater, and Ben Taskar. 2010. Posterior Regularization for Structured Latent Variable Models. *Journal of Machine Learning Research*, 11:2001–2049.

John Hale, Chris Dyer, Adhiguna Kuncoro, and Jonathan R. Brennan. 2018. Finding Syntax in Human Encephalography with Beam Search. In *Proceedings of ACL*.

Nikita Kitaev and Dan Klein. 2018. Constituency Parsing with a Self-Attentive Encoder. In *Proceedings of ACL*.

Adhiguna Kuncoro, Chris Dyer, John Hale, Dani Yogatama, Stephen Clark, and Phil Blunsom. 2018. LSTMs Can Learn Syntax-Sensitive Dependencies Well, But Modeling Structure Makes Them Better. In *Proceedings of ACL*.

Rebecca Marvin and Tal Linzen. 2018. Targeted Syntactic Evaluation of Language Models. In *Proceedings of EMNLP*.

Tomas Mikolov, Martin Karafiat, Lukas Burget, Jan Cernocky, and Sanjeev Khudanpur. 2010. Recurrent Neural Network Based Language Model. In *Proceedings of INTERSPEECH*.

Yikang Shen, Zhouhan Lin, Chin-Wei Huang, and Aaron Courville. 2018. Neural Language Modeling by Jointly Learning Syntax and Lexicon. In *Proceedings of ICLR*.

Yikang Shen, Shawn Tan, Alessandro Sordoni, and Aaron Courville. 2019. Ordered Neurons: Integrating Tree Structures into Recurrent Neural Networks. In *Proceedings of ICLR*.

Mitchell Stern, Jacob Andreas, and Dan Klein. 2017. A Minimal Span-Based Neural Constituency Parser. In *Proceedings of ACL*.

Wenhui Wang and Baobao Chang. 2016. Graph-based Dependency Parsing with Bidirectional LSTM. In *Proceedings of ACL*.

Ethan Wilcox, Peng Qian, Richard Futrell, Miguel Ballesteros, and Roger Levy. 2019. Structural Supervision Improves Learning of Non-Local Grammatical Dependencies. In *Proceedings of NAACL*.