Deep Unsupervised Learning of Syntactic Structure

Yoon Kim

(work with Chris Dyer, Alexander Rush)
Language has structure
Language has structure

Bojan Tunguz
@tunguz

Watching a model train can be very calming and satisfying.
Language has structure

watching a model train

watching a model train
Neurobiological Evidence (Fedorenko et al. 2012)

Different neural activity for Jabberwocky sentences versus non-word lists

“after the bonter mellowed the perlen
he mested to weer on colmition”

“was during cusrists fick prell pront
the pome villpa and wornetist she”
Neurobiological Evidence (Ding et al. 2015)
Neurobiological Evidence (Ding et al. 2015)
Neurobiological Evidence (Ding et al. 2015)
Unsupervised Parsing

i like superhero movies
the dog was hungry
stocks rose on tuesday
he is a big fan of football
it is snowing in boston
time flies like an arrow
i saw an elephant in my pajamas

\[
\begin{align*}
\text{the dog was hungry} \\
\text{stocks rose on tuesday}
\end{align*}
\]
Unsupervised Parsing

i like superhero movies
the dog was hungry
stocks rose on tuesday
he is a big fan of football
it is snowing in boston
time flies like an arrow
i saw an elephant in my pajamas

\[ \vdots \]

the dog was hungry

stocks rose on tuesday
Grammar Induction for Unsupervised Parsing

- Classic approach: Hypothesize a **formal grammar** that generates natural language

![Grammar Induction Diagram]

- (Parse tree implied by the grammar)
Goal of Grammar Induction

- Learning the syntax of human language
- Longstanding problem in AI/NLP
Review: Context-Free Grammars (CFG) for Natural Language

```
s \rightarrow np \hspace{0.2cm} vp
np \rightarrow det \hspace{0.2cm} n
vp \rightarrow tv \hspace{0.2cm} np
       \rightarrow iv
det \rightarrow the
       \rightarrow a
       \rightarrow an
n \rightarrow giraffe
       \rightarrow apple
iv \rightarrow dreams
tv \rightarrow eats
       \rightarrow dreams
```
Review: CFG Formal Description

\[ G = (S, \mathcal{N}, \mathcal{P}, \Sigma, \mathcal{R}) \text{ where} \]

\[ \mathcal{N} : \text{ Set of nonterminals (constituent labels)} \]
\[ \mathcal{P} : \text{ Set of preterminals (part-of-speech tags)} \]
\[ \Sigma : \text{ Set of terminals (words)} \]
\[ S : \text{ Start symbol} \]
\[ \mathcal{R} : \text{ Set of rules} \]

Each rule \( r \in \mathcal{R} \) is one of the following:

\[ S \rightarrow A \quad A \in \mathcal{N} \]
\[ A \rightarrow B \ C \quad A \in \mathcal{N}, \ B, C \in \mathcal{N} \cup \mathcal{P} \]
\[ T \rightarrow w \quad T \in \mathcal{P}, \ w \in \Sigma \]
Review: CFG Formal Description

\[ G = (S, N, P, \Sigma, R) \]

where

- \( N \): Set of nonterminals (constituent labels)
- \( P \): Set of preterminals (part-of-speech tags)
- \( \Sigma \): Set of terminals (words)
- \( S \): Start symbol
- \( R \): Set of rules

Each rule \( r \in R \) is one of the following:

- \( S \to A \) \quad \( A \in N \)
- \( A \to B \ C \) \quad \( A \in N, \quad B, C \in N \cup P \)
- \( T \to w \) \quad \( T \in P, \quad w \in \Sigma \)
Review: Probabilistic Context-Free Grammars (PCFG)

- Associate probabilities $\pi = \{\pi_r\}_{r \in \mathcal{R}}$ for each rule $r \in \mathcal{R}$.
- Probability of a tree $t$ is given by multiplying the probabilities of rules used in the derivation

$$p_{\pi}(t) = \prod_{r \in t_{\mathcal{R}}} \pi_r$$

where $t_{\mathcal{R}}$ is set of rules used to derive $t$
Review: PCFG Example

\[ S \rightarrow A_1, \ A_1 \rightarrow T_4 A_3, \]
\[ A_3 \rightarrow T_2 T_7, \ T_4 \rightarrow \text{Jon}, \]
\[ T_2 \rightarrow \text{knows}, \ T_7 \rightarrow \text{nothing} \]

\[ p_\pi(t) = \pi_{S \rightarrow A_1} \times \pi_{A_1 \rightarrow T_4 A_3} \times \pi_{A_3 \rightarrow T_2 T_7} \times \pi_{T_4 \rightarrow \text{Jon}} \times \pi_{T_2 \rightarrow \text{knows}} \times \pi_{T_7 \rightarrow \text{nothing}} \]
Review: PCFG Example

$A_i$: nonterminals

$T_j$: preterminals

$$t_R = \{ S \rightarrow A_1, \ A_1 \rightarrow T_4 \ A_3, \ A_3 \rightarrow T_2 \ T_7, \ T_4 \rightarrow \text{Jon}, \ T_2 \rightarrow \text{knows}, \ T_7 \rightarrow \text{nothing} \}$$

$$p_{\pi}(t) = \pi_{S\rightarrow A_1} \times \pi_{A_1\rightarrow T_4 \ A_3} \times \pi_{A_3\rightarrow T_2 \ T_7} \times \pi_{T_4\rightarrow \text{Jon}} \times \pi_{T_2\rightarrow \text{knows}} \times \pi_{T_7\rightarrow \text{nothing}}$$
Review: Grammar Induction with PCFGs

- Specify broad grammar structure: number of nonterminals ($|\mathcal{N}| = 30$), preterminals ($|\mathcal{P}| = 60$), set of context-free rules.

- Maximize log likelihood (Expectation-Maximization)
  - Given corpus of sentences $x^{(1)}, \ldots, x^{(N)}$,
    \[
    \max_{\pi} \sum_{n=1}^{N} \log p_{\pi}(x^{(n)})
    \]
  - Sum over unobserved trees,
    \[
    p_{\pi}(x) = \sum_{t \in \mathcal{T}(x)} p_{\pi}(t)
    \]
    where $\mathcal{T}(x) =$ set of trees whose leaves are $x$. 
Results from PCFG Induction

Unlabeled $F_1$ against gold trees on PTB.

<table>
<thead>
<tr>
<th>Model</th>
<th>$F_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Trees</td>
<td>19.5</td>
</tr>
<tr>
<td>PCFG</td>
<td>35.0</td>
</tr>
</tbody>
</table>
Results from PCFG Induction

Unlabeled $F_1$ against gold trees on PTB.

<table>
<thead>
<tr>
<th>Model</th>
<th>$F_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Trees</td>
<td>19.5</td>
</tr>
<tr>
<td>PCFG</td>
<td>35.0</td>
</tr>
<tr>
<td>Right Branching</td>
<td>39.5</td>
</tr>
</tbody>
</table>

Long history of work showing that MLE with PCFGs fails to discover linguistically meaningful tree structures [Lari and Young 1990].

*Common wisdom: “MLE with PCFGs doesn’t work”*
Rich Prior Work on Unsupervised Constituency Parsing

- **Modified objectives** [Klein and Manning 2002, 2004; Smith and Eisner 2004].
- **Use priors/nonparametric models** [Liang et al. 2007; Johnson et al. 2007].
- **Handcrafted features** [Huang et al. 2012; Golland et al. 2012].
- **Other types of regularization (e.g. on recursion depth)** [Noji et al. 2016; Jin et al. 2018].
- **Activation analysis from neural language models** [Shen et al. 2018, 2019]
This Talk: Revisit Core Assumptions about Grammar Induction

1. PCFG with an embedding parameterization can induce meaningful grammars with MLE.

2. Develop more flexible grammars through auxiliary sentence vector + neural variational inference.

3. Learn structured language models with induced trees.
This Talk: Revisit Core Assumptions about Grammar Induction

1. **PCFG with an embedding parameterization can induce meaningful grammars with MLE.**

2. Develop more flexible grammars through auxiliary sentence vector + neural variational inference.

3. Learn structured language models with induced trees.
Simple Modification: PCFG Parameterization

- **Scalar Parameterization**: Associate probabilities $\pi_T$ to each rule such that they are valid probability distributions.

  $$\pi_{T \rightarrow w} \geq 0 \quad \sum_{w' \in \Sigma} \pi_{T \rightarrow w'} = 1$$

- **“Neural” Parameterization**: Associate symbol embeddings $w_N$ to each symbol $N$ on left hand side of a rule.

  $$\pi_{T \rightarrow w} = \text{NEURALNET}(w_T) = \frac{\exp(u_w^T f(w_T))}{\sum_{w' \in \Sigma} \exp(u_{w'}^T f(w_T))}$$

  (Similar parameterizations for $A \rightarrow BC'$)
Simple Modification: PCFG Parameterization

- **Scalar Parameterization**: Associate probabilities $\pi_r$ to each rule such that they are valid probability distributions.

  $\pi_{T \rightarrow w} \geq 0 \quad \sum_{w' \in \Sigma} \pi_{T \rightarrow w'} = 1$

- **“Neural” Parameterization**: Associate symbol embeddings $w_N$ to each symbol $N$ on left hand side of a rule.

  $\pi_{T \rightarrow w} = \text{NEURALNET}(w_T) = \frac{\exp(u_w^T f(w_T))}{\sum_{w' \in \Sigma} \exp(u_{w'}^T f(w_T))}$

  (Similar parameterizations for $A \rightarrow BC$)
Simple Modification: Neural PCFG

\[ \pi_{T \rightarrow w} \propto \exp \left( \mathbf{u}_w^\top \right) \]

- Model parameters \( \theta \) given by input embeddings, output embeddings, and parameters of neural net \( f \).

- Analogous to count-based vs neural language models: parameter sharing through distributed representations (word embedding vs symbol embedding).

Same model assumptions, different parameterization.
Simple Modification: Neural PCFG

\[ \pi_{T \to w} \propto \exp \left( \mathbf{u}_w^\top \mathbf{w} \right) \]

- Model parameters \( \theta \) given by input embeddings, output embeddings, and parameters of neural net \( f \).
- Analogous to count-based vs neural language models: parameter sharing through distributed representations (word embedding vs symbol embedding).

**Same model assumptions, different parameterization.**
Neural PCFG: Training

- Maximum likelihood (EM) with dynamic programming for marginalization.
- Practical details: Stochastic gradient ascent on log marginal likelihood with Inside algorithm + Autodiff

\[ \theta_{\text{new}} = \theta_{\text{old}} + \lambda \nabla_{\theta} \log p_{\theta}(x) \]

- (PyTorch-Struct includes GPU-optimized implementations of these (and many other) algorithms.)
## Neural PCFG: Results

<table>
<thead>
<tr>
<th>Model</th>
<th>$F_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Trees</td>
<td>19.5</td>
</tr>
<tr>
<td>Right Branching</td>
<td>39.5</td>
</tr>
<tr>
<td>Scalar PCFG</td>
<td>35.0</td>
</tr>
</tbody>
</table>

(English Penn Treebank)
## Neural PCFG: Results

<table>
<thead>
<tr>
<th>Model</th>
<th>$F_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Trees</td>
<td>19.5</td>
</tr>
<tr>
<td>Right Branching</td>
<td>39.5</td>
</tr>
<tr>
<td>Scalar PCFG</td>
<td>35.0</td>
</tr>
<tr>
<td>Neural PCFG</td>
<td>52.6</td>
</tr>
</tbody>
</table>

(English Penn Treebank)
## Neural PCFG Results

<table>
<thead>
<tr>
<th>Model</th>
<th>$F_1$</th>
<th>Training/Test PPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Trees</td>
<td>19.5</td>
<td>—</td>
</tr>
<tr>
<td>Right Branching</td>
<td>39.5</td>
<td>—</td>
</tr>
<tr>
<td>Scalar PCFG</td>
<td>35.0</td>
<td>$\approx 350$</td>
</tr>
<tr>
<td>Neural PCFG</td>
<td>52.6</td>
<td>$\approx 250$</td>
</tr>
</tbody>
</table>
This Talk: Revisit Core Assumptions about Grammar Induction

1. PCFG with an embedding parameterization can induce meaningful grammars with MLE.
2. Develop more flexible grammars through auxiliary sentence vector + neural variational inference.
3. Learn structured language models with induced trees.
Review: Limitations of simple PCFGs

No sensitivity to lexical context

Review: Limitations of simple PCFGs

No sensitivity to lexical context
Review: Limitations of simple PCFGs

No sensitivity to structural context

Review: Limitations of simple PCFGs

Johnson et al. [2007]: Supervised PCFG + Unsupervised fine tuning decreases parsing accuracy while corpus likelihood improves!

“It is easy to demonstrate that the poor quality of the PCFG models is the cause of these problems rather than search or other algorithmic issues. If one initializes either the IO or Bayesian estimation procedures with treebank parses and then runs the procedure using the yields alone, the accuracy of the parses uniformly decreases while the (posterior) likelihood uniformly increases with each iteration, demonstrating that improving the (posterior) likelihood of such models does not improve parse accuracy.”
Classic Solutions: Lexicalization

- No sensitivity to lexical context $\implies$ Lexicalized PCFGs [Collins 1997]
- Rules are lexicalized, e.g.

\[
A \rightarrow BC \implies A(w) \rightarrow B(w)C(h)
\]

\[w, h \in \Sigma\]

- Integrates notion of headedness
Classic Solutions: Higher-order Grammars

- No sensitivity to structural context $\implies$ Horizontal/Vertical Markovization \cite{Klein and Manning 2003}
- Richer dependencies through grandparents/siblings.
Classic Solutions: Enriching PCFGs

- Lexicalized PCFG [Collins 1997]
- Horizontal/Vertical Markovization [Klein and Manning 2003]
- Latent Variable PCFG [Petrov et al. 2006]

Expensive to apply in the unsupervised case due to explosion in number of rules.
Goal: Capture these in a soft manner.

Compound generative process (Bayesian PCFG):

1. \( z \sim \mathcal{N}(0, I) \)
2. \( \pi_z = \text{NEURALNETWORK}([w_N; z]) \), for example,
   \[
   \pi_{z,T \rightarrow w} = \frac{\exp(u_w^T f([w_T; z]))}{\sum_{w' \in \Sigma} \exp(u_{w'}^T f([w_T; z]))}
   \]
3. \( t \sim \text{PCFG}(\pi_z) \)
4. \( x = \text{yield}(t) \)
Goal: Capture these in a soft manner.

**Compound** generative process (Bayesian PCFG):

1. \( z \sim \mathcal{N}(0, I) \)
2. \( \pi_z = \text{NEURALNETWORK}([w_N; z]) \), for example,
   \[
   \pi_{z,T \rightarrow w} = \frac{\exp(u^\top_w f([w_T; z]))}{\sum_{w' \in \Sigma} \exp(u_{w'}^\top f([w_T; z]))}
   \]
3. \( t \sim \text{PCFG}(\pi_z) \)
4. \( x = \text{yield}(t) \)
Compound PCFG

\[ \pi_{z,T \rightarrow w} \propto \exp( u_w^T f( [w_T ; z] )) \]

- Input/output embeddings and neural net \( f \) shared across sentences, but rule probabilities for each sentence can vary through \( z \)
- Intuition: \( z \) can encode lexical/structural information specific to the sentence.
Neural PCFG vs. Compound PCFG

Neural PCFG

Compound PCFG

\[ z \sim p_\gamma(z) \]
Sentence-level latent vector
Neural PCFG vs. Compound PCFG

The model reduces to a PCFG conditioned on $z$
Compound PCFG: Training and Inference

For maximum likelihood, log marginal likelihood given by

$$\log p_\theta(x) = \log \left( \int \sum_{t \in T(x)} p_\theta(t \mid z) p(z) \, dz \right)$$

- Intractable due to integral over $z$. 
Variational Inference: Introduce variational posterior for $z$

$$\log p_\theta(x) \geq \mathbb{E}_{q_\phi(z \mid x)} \left[ \log \sum_{t \in T(x)} p_\theta(t \mid z) \right] - \text{KL}[q_\phi(z \mid x) \parallel p(z)]$$

- Inference network over $x$ produces parameters for the Gaussian variational posterior $q_\phi(z \mid x)$.
- Given a sample $z$, can calculate with dynamic programming

$$p_\theta(x \mid z) = \sum_{t \in T(x)} p_\theta(t \mid z)$$
Variational Inference: Introduce variational posterior for $z$

$$\log p_\theta(x) \geq \mathbb{E}_{q_\phi(z \mid x)} \left[ \log \sum_{t \in \mathcal{T}(x)} p_\theta(t \mid z) \right] - \text{KL}[q_\phi(z \mid x) \parallel p(z)]$$

- Inference network over $x$ produces parameters for the Gaussian variational posterior $q_\phi(z \mid x)$.
- Given a sample $z$, can calculate with dynamic programming

$$p_\theta(x \mid z) = \sum_{t \in \mathcal{T}(x)} p_\theta(t \mid z)$$
Compound PCFG: Training and Inference

Collapsed Variational Inference

\[
\log p_\theta(x) \geq \mathbb{E}_{q_\phi(z \mid x)} \left[ \log p_\theta(x \mid z) \right] - \text{KL} \left[ q_\phi(z \mid x) \parallel p(z) \right]
\]

reparameterized sample inside algorithm analytic KL between 2 Gaussians

"VAE with a PCFG decoder"
Collapsed Variational Inference

$$\log p_\theta(x) \geq \mathbb{E}_{q_\phi(z|x)} \left[ \log p_\theta(x|z) \right] - \text{KL} \left[ q_\phi(z|x) \| p(z) \right]$$

reparameterized sample \hspace{1cm} inside algorithm \hspace{1cm} analytic KL between 2 Gaussians

“VAE with a PCFG decoder”
## Compound PCFG: Results on PTB

<table>
<thead>
<tr>
<th>Model</th>
<th>$F_1$</th>
<th>Training/Test PPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Trees</td>
<td>19.5</td>
<td>–</td>
</tr>
<tr>
<td>Right Branching</td>
<td>39.5</td>
<td>–</td>
</tr>
<tr>
<td>Scalar PCFG</td>
<td>35.0</td>
<td>≈ 350</td>
</tr>
<tr>
<td>Neural PCFG</td>
<td>52.6</td>
<td>≈ 250</td>
</tr>
<tr>
<td><strong>Compound PCFG</strong></td>
<td>60.1</td>
<td>≈ 190</td>
</tr>
</tbody>
</table>
## Compound PCFG: Comparison against other unsupervised parsers

<table>
<thead>
<tr>
<th>Model</th>
<th>English (PTB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRPN [Shen et al. 2018]</td>
<td>38.1</td>
</tr>
<tr>
<td>Ordered Neurons [Shen et al. 2019]</td>
<td>49.4</td>
</tr>
<tr>
<td>DIORA [Drozdov et al. 2019]</td>
<td>58.9</td>
</tr>
<tr>
<td>Constituency Tests [Cao et al. 2020]</td>
<td>62.8</td>
</tr>
<tr>
<td>Right Branching</td>
<td>39.5</td>
</tr>
<tr>
<td>Scalar PCFG</td>
<td>35.0</td>
</tr>
<tr>
<td>Neural PCFG</td>
<td>52.6</td>
</tr>
<tr>
<td>Compound PCFG</td>
<td>60.1</td>
</tr>
</tbody>
</table>
## Compound PCFG: Results on other languages

<table>
<thead>
<tr>
<th>Model</th>
<th>English</th>
<th>Chinese</th>
<th>Japanese</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Trees</td>
<td>19.5</td>
<td>16.0</td>
<td>15.3</td>
</tr>
<tr>
<td>Left Branching</td>
<td>8.7</td>
<td>9.7</td>
<td>25.5</td>
</tr>
<tr>
<td>Right Branching</td>
<td>39.5</td>
<td>20.0</td>
<td>1.2</td>
</tr>
<tr>
<td>Scalar PCFG</td>
<td>35.0</td>
<td>15.0</td>
<td>15.7</td>
</tr>
<tr>
<td>Neural PCFG</td>
<td>52.6</td>
<td>29.5</td>
<td>44.6</td>
</tr>
<tr>
<td>Compound PCFG</td>
<td>60.1</td>
<td>39.8</td>
<td>47.4</td>
</tr>
</tbody>
</table>
tugh ghaH vllegh 'e' tamvad ylja' tell Tom that I will see him soon
Model Analysis: Nonterminal Alignment ($|\mathcal{N}| = 30$)
Model Analysis: Nonterminal Alignment ($|N| = 30$)

| SBAR | A_1 | A_2 | A_3 | A_4 | A_5 | A_6 | A_7 | A_8 | A_9 | A_{10} | A_{11} | A_{12} | A_{13} | A_{14} | A_{15} | A_{16} | A_{17} | A_{18} | A_{19} | A_{20} | A_{21} | A_{22} | A_{23} | A_{24} | A_{25} | A_{26} | A_{27} | A_{28} | A_{29} | A_{30} |
|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| NP   |     |     |     |     |     |     |     |     |     |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |
| VP   |     |     |     |     |     |     |     |     |     |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |
| PP   |     |     |     |     |     |     |     |     |     |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |
| ADJP |     |     |     |     |     |     |     |     |     |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |
| ADVP |     |     |     |     |     |     |     |     |     |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |
| OTHER|     |     |     |     |     |     |     |     |     |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |        |
| Accuracy | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

- the level of hypoglycemia
- a measure of inflation
- an act of god
- the isle of man
- the first of december

- the organization of american states
- its acquirer for half price
- a decline in brazilian interest
- his trial on perjury charges
- each plan including the assumptions
**Model Analysis: Nonterminal Alignment** (|\(\mathcal{N}\)| = 30)

| SBAR | A₁  | A₂  | A₃  | A₄  | A₅  | A₆  | A₇  | A₈  | A₉  | A₁₀ | A₁₁ | A₁₂ | A₁₃ | A₁₄ | A₁₅ | A₁₆ | A₁₇ | A₁₈ | A₁₉ | A₂₀ | A₂₁ | A₂₂ | A₂₃ | A₂₄ | A₂₅ | A₂₆ | A₂₇ | A₂₈ | A₂₉ | A₃₀ |
|------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| NP   | A₂  | A₃  | A₄  | A₅  | A₆  | A₇  | A₈  | A₉  | A₁₀ | A₁₁ | A₁₂ | A₁₃ | A₁₄ | A₁₅ | A₁₆ | A₁₇ | A₁₈ | A₁₉ | A₂₀ | A₂₁ | A₂₂ | A₂₃ | A₂₄ | A₂₅ | A₂₆ | A₂₇ | A₂₈ | A₂₉ | A₃₀ |
| VP   | A₃  | A₄  | A₅  | A₆  | A₇  | A₈  | A₉  | A₁₀ | A₁₁ | A₁₂ | A₁₃ | A₁₄ | A₁₅ | A₁₆ | A₁₇ | A₁₈ | A₁₉ | A₂₀ | A₂₁ | A₂₂ | A₂₃ | A₂₄ | A₂₅ | A₂₆ | A₂₇ | A₂₈ | A₂₉ | A₃₀ |
| PP   | A₄  | A₅  | A₆  | A₇  | A₈  | A₉  | A₁₀ | A₁₁ | A₁₂ | A₁₃ | A₁₄ | A₁₅ | A₁₆ | A₁₇ | A₁₈ | A₁₉ | A₂₀ | A₂₁ | A₂₂ | A₂₃ | A₂₄ | A₂₅ | A₂₆ | A₂₇ | A₂₈ | A₂₉ | A₃₀ |
| ADJP | A₅  | A₆  | A₇  | A₈  | A₉  | A₁₀ | A₁₁ | A₁₂ | A₁₃ | A₁₄ | A₁₅ | A₁₆ | A₁₇ | A₁₈ | A₁₉ | A₂₀ | A₂₁ | A₂₂ | A₂₃ | A₂₄ | A₂₅ | A₂₆ | A₂₇ | A₂₈ | A₂₉ | A₃₀ |
| ADVP | A₆  | A₇  | A₈  | A₉  | A₁₀ | A₁₁ | A₁₂ | A₁₃ | A₁₄ | A₁₅ | A₁₆ | A₁₇ | A₁₈ | A₁₉ | A₂₀ | A₂₁ | A₂₂ | A₂₃ | A₂₄ | A₂₅ | A₂₆ | A₂₇ | A₂₈ | A₂₉ | A₃₀ |
| OTHER| A₇  | A₈  | A₉  | A₁₀ | A₁₁ | A₁₂ | A₁₃ | A₁₄ | A₁₅ | A₁₆ | A₁₇ | A₁₈ | A₁₉ | A₂₀ | A₂₁ | A₂₂ | A₂₃ | A₂₄ | A₂₅ | A₂₆ | A₂₇ | A₂₈ | A₂₉ | A₃₀ |
| Accuracy | A₈  | A₉  | A₁₀ | A₁₁ | A₁₂ | A₁₃ | A₁₄ | A₁₅ | A₁₆ | A₁₇ | A₁₈ | A₁₉ | A₂₀ | A₂₁ | A₂₂ | A₂₃ | A₂₄ | A₂₅ | A₂₆ | A₂₇ | A₂₈ | A₂₉ | A₃₀ |

- **Model**: in the chemicals sector as the death rate by a corresponding amount of our own institutions to the creditors committee.
- **Diagram**: through another rise in the base rate of the roads into the bay area on the funding of its original transaction for a portion of the buying public as an amendment to a pending measure.
Model Analysis: Nonterminal Alignment ($|\mathcal{N}| = 30$)

<table>
<thead>
<tr>
<th></th>
<th>A_1</th>
<th>A_2</th>
<th>A_3</th>
<th>A_4</th>
<th>A_5</th>
<th>A_6</th>
<th>A_7</th>
</tr>
</thead>
<tbody>
<tr>
<td>SBAR</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>VP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADJP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADVP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OTHER</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Accuracy:

- did his work scored surprising gains made emergency loans got three trucks

- caused an injury or death carried a change of clothing illustrated this mix of power founded the company in chicago sold his stake in texaco
Model Analysis: Nonterminal Alignment ($|N| = 30$)
Model Analysis: Preterminal Alignment ($|\mathcal{P}| = 60$)
Model Analysis: What does $z$ learn?

Nearest neighbors based on variational posterior mean vector

\begin{align*}
\langle \text{unk} \rangle \text{ corp. received an N million army contract for helicopter engines} \\
\text{boeing co. received a N million air force contract for developing cable systems for the } \langle \text{unk} \rangle \text{ missile} \\
\text{general dynamics corp. received a N million air force contract for } \langle \text{unk} \rangle \text{ training sets} \\
\text{grumman corp. received an N million navy contract to upgrade aircraft electronics} \\
\text{thomson missile products with about half british aerospace 's annual revenue include the } \langle \text{unk} \rangle \langle \text{unk} \rangle \langle \text{unk} \rangle \text{ missile family} \\
\text{already british aerospace and french } \langle \text{unk} \rangle \langle \text{unk} \rangle \langle \text{unk} \rangle \text{ on a british missile contract and on an air-traffic control} \\
\end{align*}
Model Analysis: What does $z$ learn?

Cluster 1
- of the company’s capital structure in the company’s divestiture program by the company’s new board in the company’s core business

Cluster 2
- above the treasury’s N-year note
- above the treasury’s seven-year note
- above the treasury’s comparable note
- above the treasury’s five-year note
This Talk: Revisit Core Assumptions about Grammar Induction

1. PCFG with an embedding parameterization can induce meaningful grammars with MLE.
2. Develop more flexible grammars through auxiliary sentence vector + neural variational inference.
3. **Learn structured language models with induced trees.**
## Compound PCFG as a Language Model

<table>
<thead>
<tr>
<th>Model</th>
<th>$F_1$</th>
<th>Test PPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scalar PCFG</td>
<td>35.0</td>
<td>≈ 350</td>
</tr>
<tr>
<td>Neural PCFG</td>
<td>52.6</td>
<td>≈ 250</td>
</tr>
<tr>
<td>Compound PCFG</td>
<td>60.1</td>
<td>≈ 190</td>
</tr>
</tbody>
</table>
Compound PCFG as a Language Model

<table>
<thead>
<tr>
<th>Model</th>
<th>$F_1$</th>
<th>Test PPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scalar PCFG</td>
<td>35.0</td>
<td>$\approx$ 350</td>
</tr>
<tr>
<td>Neural PCFG</td>
<td>52.6</td>
<td>$\approx$ 250</td>
</tr>
<tr>
<td>Compound PCFG</td>
<td>60.1</td>
<td>$\approx$ 190</td>
</tr>
<tr>
<td>RNN LM</td>
<td>–</td>
<td>86.2</td>
</tr>
</tbody>
</table>

Good parser, poor language model.
Review: Recurrent Neural Network Grammars (RNNG) [Dyer et al. 2016]

- Structured joint generative model of sentence $x$ and tree $z$

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

- Generate next word conditioned on partially-completed syntax tree

- Like RNN LM, no independence assumptions.
Review: RNN LMs

“Flat” left-to-right generation

\[ x_t \sim p_\theta(x \mid x_1, \ldots, x_{t-1}) = \text{softmax}(W h_{t-1} + b) \]
Introduce binary variables $\mathbf{z} = [z_1, \ldots, z_{2T-1}]$ (unlabeled binary tree)

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

$$z_t \sim \text{Bernoulli}(p_t)$$

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

Sample word from context representation
(Similar to standard RNNLMs)

\[ x \sim \text{softmax}(W_h^{\text{prev}} + b) \]
Obtain new context representation with $e_{\text{hungry}}$

$$h_{\text{new}} = \text{LSTM}(e_{\text{hungry}}, h_{\text{prev}})$$
RNNG [Dyer et al. 2016]

\[ h_{\text{new}} = \text{LSTM}(e_{\text{cat}}, h_{\text{prev}}) \]
If $z_t = \text{REDUCE}$
If $z_t = \text{REDUCE}$

Pop last two elements
RNNG [Dyer et al. 2016]

Obtain new representation of constituent

\[ e_{\text{hungry cat}} = \text{TreeLSTM}(e_{\text{hungry}}, e_{\text{cat}}) \]
RNNG [Dyer et al. 2016]

Move the new representation onto the stack

$$h_{\text{new}} = \text{LSTM}(e_{(\text{hungry cat})}, h_{\text{prev}})$$
**Compound PCFG + RNNG**

- Compound PCFG to parse training set, train an RNNG on induced trees, fine-tune with unsupervised RNNG.

<table>
<thead>
<tr>
<th>Model</th>
<th>Test PPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neural PCFG</td>
<td>252.6</td>
</tr>
<tr>
<td>Compound PCFG</td>
<td>196.3</td>
</tr>
<tr>
<td>RNN LM</td>
<td>86.2</td>
</tr>
<tr>
<td>URNNG + Compound PCFG</td>
<td>83.7</td>
</tr>
<tr>
<td>URNNG + Gold Trees</td>
<td>78.3</td>
</tr>
</tbody>
</table>
**Compound PCFG + RNNG**

- Compound PCFG to parse training set, train an RNNG on induced trees, fine-tune with unsupervised RNNG.

<table>
<thead>
<tr>
<th>Model</th>
<th>Test PPL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neural PCFG</td>
<td>252.6</td>
</tr>
<tr>
<td>Compound PCFG</td>
<td>196.3</td>
</tr>
<tr>
<td>RNN LM</td>
<td>86.2</td>
</tr>
<tr>
<td>URNNG + Compound PCFG</td>
<td>83.7</td>
</tr>
<tr>
<td>URNNG + Gold Trees</td>
<td>78.3</td>
</tr>
</tbody>
</table>
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.
<table>
<thead>
<tr>
<th>Model</th>
<th>Test PPL</th>
<th>Syntactic Eval.</th>
</tr>
</thead>
<tbody>
<tr>
<td>RNN LM</td>
<td>86.2</td>
<td>60.9%</td>
</tr>
<tr>
<td>URNNG + Compound PCFG</td>
<td>83.7</td>
<td>76.1%</td>
</tr>
<tr>
<td>URNNG + Gold Trees</td>
<td>78.3</td>
<td>76.1%</td>
</tr>
</tbody>
</table>

**Syntactic Evaluation [Marvin and Linzen 2018]**
Compound PCFG Extensions

- Lexicalized Compound PCFG [Zhu et al. 2020]
- Visually Grounded Compound PCFG [Zhao and Titov 2020]
Discussion

Limitations

- Can be slower to train due to DP.
- Latent vector to approximate richer grammars.

“We assume that the goal of learning a context-free grammar needs no justification.”

[Carroll and Charniak 1992]

- What is the role of grammars (and other linguistic structures) in ELMo/BERT era?
Discussion

Limitations

- Can be slower to train due to DP.
- Latent vector to approximate richer grammars.

“We assume that the goal of learning a context-free grammar needs no justification.”

[Carroll and Charniak 1992]

- What is the role of grammars (and other linguistic structures) in ELMo/BERT era?
Future Work

- Separation of “what to say” from “how to say it” for structured generation.
- Some languages are provably not context-free $\implies$ neural parameterizations of mildly context-sensitive formalisms (e.g. tree-adjoining grammars).
- Investigate why MLE with scalar parameterization fails but neural parameterization works.


Slav Petrov, Leon Barret, Romain Thibaux, and Dan Klein. 2006. Learning Accurate, Compact, and Interpretable Tree Annotation. In Proceedings of ACL.


