Covariance in Unsupervised Learning of Probabilistic Grammars

Cohen and Smith (2010)

Presenter: Alice Lai

Introduction

- A framework for modeling covariance in probabilistic grammars
- Express priors using logistic normal distributions
- Experiments on dependency grammar induction with parameter tying within and across grammars

Grammar Induction

- Grammar induction: unsupervised discovery of grammatical structure
- Bayesian models used to specify priors of probabilistic grammars
- Many models use Dirichlet distributions because of conjugate prior property

Dependency Grammars

- Syntax is a directed tree, words are vertices, edges are dependency relations
  - Two words have a dependency relation if one is an argument or modifier of the other

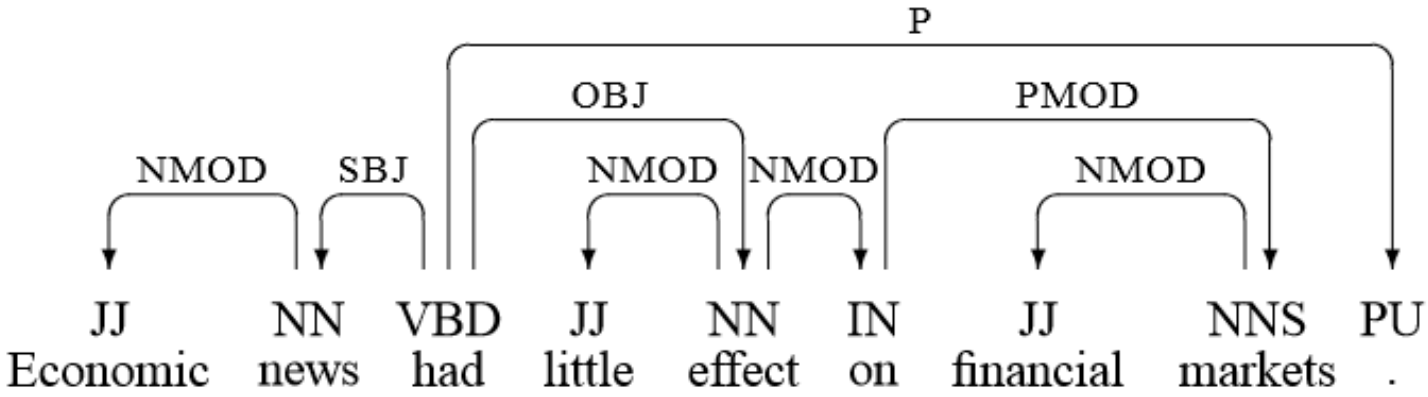


Figure from Nivre (2005), Dependency Grammar and Dependency Parsing.

Dependency Model with Valence

- Proposed by Klein and Manning (2004)
- Each word has:
  - Binomial distribution over whether it has any left/ right children
  - Geometric distribution over the number of left/ right children
- Inference is cubic in the length of the sentence
- Maximum likelihood via EM algorithm

DMV Example

y =

x = ⟨$ DT JJ NN VBZ⟩

$$p_{\mathbf{x},\mathbf{y}}(\theta) = \theta_{\downarrow c}(\text{VBZ\$,r}) \times p_{y\uparrow(1)}(\text{VBZ},\theta)$$

$$p_{\mathbf{y}\uparrow(1)}(\text{VBZ},\theta) = \theta_{\downarrow s}(\neg\text{stopVBZ,l, f}) \times \theta_{\downarrow c}(\text{NNVBZ, l}) \times p(\mathbf{y}\uparrow(2) \mid \text{NN},\theta) \times \theta_{\downarrow s}(\text{stop}\mid\text{VBZ,l,t}) \times \theta_{\downarrow s}(\text{stop}\mid\text{VBZ,r,f})$$

$$p_{\mathbf{y}\uparrow(2)}(\text{NN},\theta) = \theta_{\downarrow s}(\neg\text{stopNN,l,f}) \times \theta_{\downarrow c}(\text{JJNN,l}) \times \theta_{\downarrow s}(\text{stopJJ,r,f}) \times \theta_{\downarrow s}(\text{stopJJ,l,f}) \times \theta_{\downarrow s}(\neg\text{stopNN,l,t}) \times \theta_{\downarrow c}(\text{DTNN,l}) \times \theta_{\downarrow s}(\text{stopDT,r,f}) \times \theta_{\downarrow s}(\text{stopDT,l,f}) \times \theta_{\downarrow s}(\text{stopNN,l,t}) \times \theta_{\downarrow s}(\text{stopNN,r,f})$$

Modeling Covariance

- We expect to see covariance in probabilistic grammars
  - Words and word classes (e.g. parts of speech) follow patterns
  - Example: the probability that a word class has singular noun arguments is related to the probability that it has plural noun arguments
- Use logistic normal distribution to model covariance

Logistic Normal Distribution

- Logistic transformation of multivariate normal distribution to points on probabilistic simplex
- Used by Blei and Lafferty (2006) for correlated topic models

Limitations of LN Distribution

- Covariance only modeled within a multinomial, not across multinomials
- Probabilistic grammar models involve multiple multinomials
  - We want to model the correlation between different verb types (VBD, VBZ) both taking nouns as arguments

Partitioned LN Distribution

- Define a Gaussian over $N = \sum_{k=1}^{K} N_k$ variables with one $N \times N$ covariance matrix
- Covariance matrix models correlations between all pairs of events across all multinomials
- Apply the logistic transformation to subvectors to get individual multinomials

Shared LN Distribution

- $N \times N$ size covariance matrix is expensive to create
- Instead of a single normal vector for all multinomials, use several normal vectors
- Partition normal vectors, use N normal experts to sample from multinomials, recombine parts of vectors and take average
- Result: $\theta \sim \mathrm{SLN}(\mu, \Sigma, \delta)$

SLN Example

$$
\left.
\begin{array}{lcll}
I_1 & = & \{1{:}2, 3{:}6, 7{:}9\} & = \{\ I_{1,1},\ I_{1,2},\ I_{1,L_1}\ \} \\
I_2 & = & \{1{:}2, 3{:}6\} & = \{\ I_{2,1},\ I_{2,L_2}\ \} \\
I_3 & = & \{1{:}4, 5{:}7\} & = \{\ I_{3,1},\ I_{3,L_3}\ \} \\
I_N & = & \{1{:}2\} & = \{\ I_{4,L_4}\ \} \\
 & & & \quad J_1 \quad J_2 \quad J_K
\end{array}
\right\} \text{prt. struct. } \mathcal{S}
$$

$$
\left.
\begin{array}{lcll}
\eta_1 & = & \langle \eta_{1,1}, \eta_{1,2},\ \eta_{1,3}, \eta_{1,4}, \eta_{1,5}, \eta_{1,6},\ \eta_{1,7}, \eta_{1,8}, \eta_{1,\ell_1} \rangle & \sim \text{Normal}(\mu_1, \Sigma_1) \\
\eta_2 & = & \langle \eta_{2,1}, \eta_{2,2},\ \eta_{2,3}, \eta_{2,4}, \eta_{2,5}, \eta_{2,\ell_2} \rangle & \sim \text{Normal}(\mu_2, \Sigma_2) \\
\eta_3 & = & \langle \eta_{3,1}, \eta_{3,2}, \eta_{3,3}, \eta_{3,4},\ \eta_{3,5}, \eta_{3,6}, \eta_{3,\ell_3} \rangle & \sim \text{Normal}(\mu_3, \Sigma_3) \\
\eta_4 & = & \langle \eta_{4,1}, \eta_{4,\ell_4} \rangle & \sim \text{Normal}(\mu_4, \Sigma_4)
\end{array}
\right\} \text{sample } \eta
$$

$$
\left.
\begin{array}{lcl}
\tilde{\eta}_1 & = & \frac{1}{3}\langle \eta_{1,1} + \eta_{2,1} + \eta_{4,1},\ \eta_{1,2} + \eta_{2,2} + \eta_{4,2} \rangle \\
\tilde{\eta}_2 & = & \frac{1}{3}\langle \eta_{1,3} + \eta_{2,3} + \eta_{3,1},\ \eta_{1,4} + \eta_{2,4} + \eta_{3,2},\ \eta_{1,5} + \eta_{2,5} + \eta_{3,3}, \\
 & & \eta_{1,6} + \eta_{2,6} + \eta_{3,4} \rangle \\
\tilde{\eta}_3 & = & \frac{1}{2}\langle \eta_{1,7} + \eta_{3,5},\ \eta_{1,8} + \eta_{3,6},\ \eta_{1,9} + \eta_{3,7} \rangle
\end{array}
\right\} \text{combine } \eta
$$

$$
\left.
\begin{array}{lcl}
\theta_1 & = & (\exp \tilde{\eta}_1) \Big/ \sum_{i'=1}^{N_1} \exp \tilde{\eta}_{1,i'} \\
\theta_2 & = & (\exp \tilde{\eta}_2) \Big/ \sum_{i'=1}^{N_2} \exp \tilde{\eta}_{2,i'} \\
\theta_3 & = & (\exp \tilde{\eta}_3) \Big/ \sum_{i'=1}^{N_3} \exp \tilde{\eta}_{3,i'}
\end{array}
\right\} \text{softmax}
$$

Bayesian Models over Grammars

- Use maximum *a posteriori* framework for learning with symmetric Dirichlet priors (Smith 2006):

$$\max_{\theta} p(\theta|\alpha, \mathbf{G}) \prod_{m=1}^{M} \sum_{\mathbf{y}} p(\mathbf{x}_m, \mathbf{y}|\theta, \mathbf{G})$$

- This model: treat θ as a hidden variable: integrate out θ in the probability of the data

$$p(\mathbf{x}_1, ..., \mathbf{x}_M|\alpha, \mathbf{G}) = \int p(\theta|\alpha, \mathbf{G}) \prod_{m=1}^{M} \sum_{y} p(\mathbf{x}_m, \mathbf{y}|\theta, \mathbf{G}) d\theta$$

  - Estimate α, the distribution over grammar parameters

Two Model Variations

Model I:

For $m \in \{1, \ldots, M\}$:

1. Draw θ_m from the prior $p(\theta \mid \mathbf{G}, \ldots)$.
2. Draw $(\mathbf{x}_m, \mathbf{y}_m)$ from $p(\mathbf{x}_m, \mathbf{y}_m \mid \theta_m, \mathbf{G})$.

Model II:

1. Draw θ from the prior $p(\theta \mid \mathbf{G}, \ldots)$.
2. For $m \in \{1, \ldots, M\}$:
   Draw $(\mathbf{x}_m, \mathbf{y}_m)$ from $p(\mathbf{x}_m, \mathbf{y}_m \mid \theta, \mathbf{G})$.

Model 1: grammar parameters θ drawn once per sentence

Model 2: grammar parameters θ drawn once for all sentences in corpus

Choosing the Prior Distribution

- Raiffa and Schaifer (1961) establish 3 necessary qualities for prior distributions
  1) Analytical tractability
  2) Richness
  3) Interpretability
- Most literature has focused on (1), using a Dirichlet prior because it is conjugate to the multinomial family
- What about (2) and (3)?

Dirichlet Priors

- Computationally, a good choice for prior because of analytic tractability
- May encourage sparse solutions (eliminating unnecessary grammar rules)
- However, no explicit covariance structure when drawing θ from a Dirichlet distribution

LN Priors

- Define one LN distribution for each multinomial
  - SLN covariance: define one normal expert for each single multinomial and other experts across related multinomials
- Prior over $\theta\downarrow k$ that allows covariance among $\langle \theta\downarrow\{k,1\},\ldots,\theta\downarrow\{k,N\downarrow k\}\rangle$
- For SLN, covariance among $\theta\downarrow\{k,i\}$ not directly defined
  - Normal experts $\eta\downarrow\{i,j\}$ define this relationship. Think of $\eta\downarrow\{i,j\}$ as weights associated with event probabilities.

Decoding

- How to choose an analysis (grammatical structure **y**) given the input
- Viterbi decoding: the most likely analysis

$$\mathbf{y}^* = \operatorname*{argmax}_{\mathbf{y}} p(\mathbf{y} \mid \mathbf{x}, \theta, \mathbf{G}) = \operatorname*{argmax}_{\mathbf{y}} p(\mathbf{x}, \mathbf{y} \mid \theta, \mathbf{G})$$

- Minimum Bayes risk decoding: the analysis that minimizes risk

$$\mathbf{y}^* = \operatorname*{argmax}_{\mathbf{y}} \mathbb{E}_{p(\cdot|\mathbf{x},\theta,\mathbf{G})} \text{cost}(\mathbf{y}, \cdot) = \operatorname*{argmax}_{\mathbf{y}} \sum_{\mathbf{y}'} p(\mathbf{y}' \mid \mathbf{x}, \theta, \mathbf{G}) \ \text{cost}(\mathbf{y}, \mathbf{y}')$$

  - $\text{cost}(\mathbf{y}, \mathbf{y}\uparrow^*)$ is the cost of choosing $\mathbf{y}$ when the correct analysis is $\mathbf{y}\uparrow^*$

3 Decoding Techniques

1) Viterbi decoding applied to point estimate of θ
2) MBR decoding applied to point estimate of θ
   - Loss function is dependency attachment error.
3) Committee decoding: randomly sample grammar weights, apply decoding, average results
   - Viterbi and MBR ignore covariance matrix Σ
   - This method has generalization error guarantees

Variational Inference

- Bound the log-likelihood and optimize with respect to approximate posterior $q(\theta, \boldsymbol{y})$
- Mean-field approximation: $q(\theta, \boldsymbol{y})$ is factorized and has form $q(\theta, \boldsymbol{y}) = q(\theta)q(\boldsymbol{y})$
- LN prior requires additional approximation because of lack of conjugacy
  - First-order Taylor approximation to log of normalization of LN distribution
  - Use inside-outside algorithm with weighted grammar for inference

Variational EM

Variational inference algorithm assumes that μ and Σ are fixed. To estimate these parameters, use variational EM.

- E-step: maximize bound with respect to variational parameters using coordinate ascent. Optimize each parameter separately.
- M-step: use maximum likelihood estimation to update values of μ and Σ from variational parameters.

Grammar Induction Experiments

1) LN distribution on English text
2) LN distribution on additional languages (Chinese, Portuguese, Turkish, Czech, Japanese)
3) SLN distribution tying parameters for coarse POS tags (English, Portuguese, Turkish)
4) SLN distribution with bilingual settings (English, Portuguese, Turkish)

Experiment: English Text

- Input: gold standard POS tagged text from Penn treebank
- Training restricted to sentences of ≤10 words
- Attachment accuracy: for what fraction of words did the predicted parent match the gold annotation?

Experiment: English Text

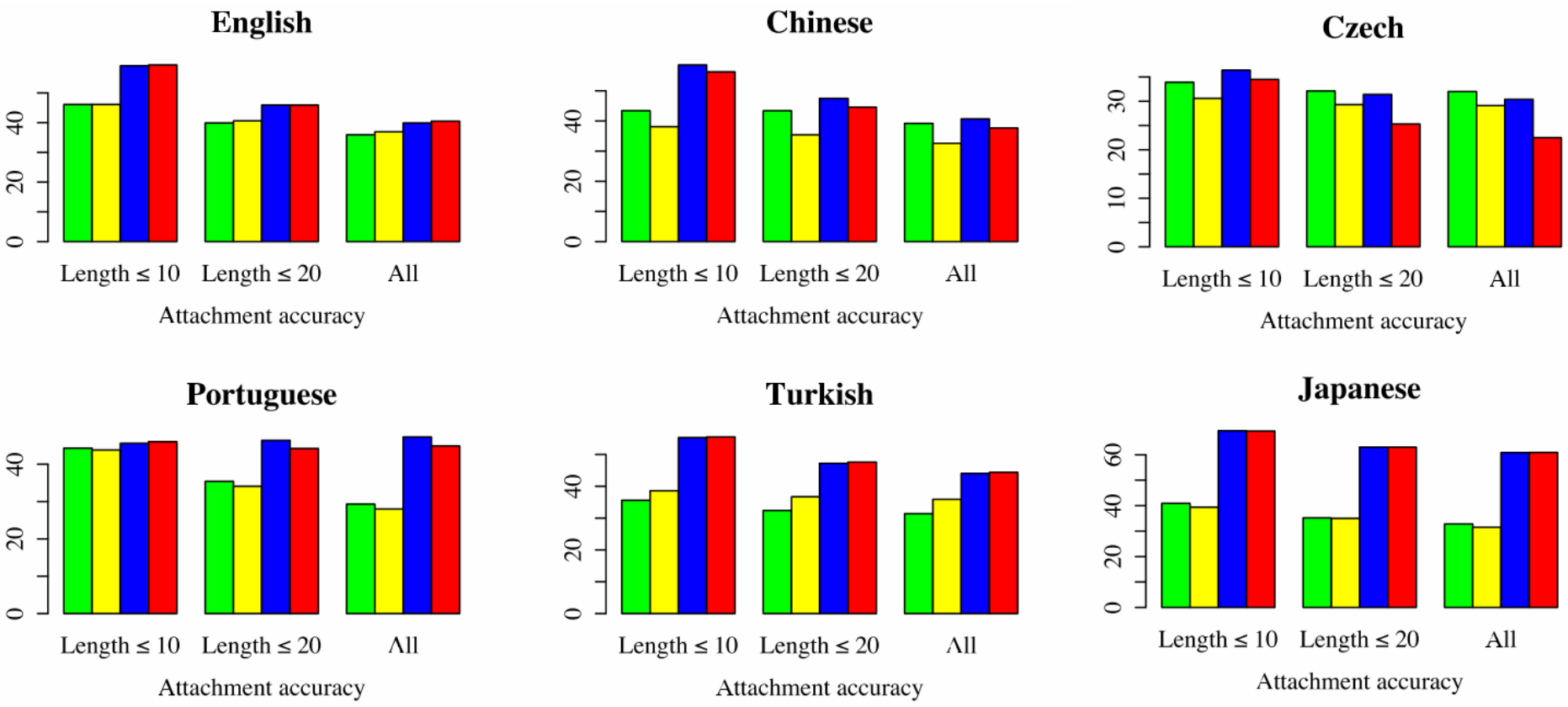
Covariance matrix initialization

1) $N\downarrow k \times N\downarrow k$ identity matrix
2) Use prior knowledge of POS tags
   - 12 disjoint tag families (coarse tags)
   - Covariance matrix has 1 on diagonal, 0.5 between tags in same family, 0 elsewhere

Results: English Text

	attachment accuracy (%)								
	Viterbi decoding			MBR decoding			Committee decoding		
	≤ 10	≤ 20	all	≤ 10	≤ 20	all	≤ 10	≤ 20	all
MLE	45.8	39.1	34.2	46.1	39.9	35.9		*	
Dirichlet-I	45.9	39.4	34.9	46.1	40.6	36.9		*	
LN-I, $\Sigma_k^{(0)} = \mathbf{I}$	56.5	42.9	36.6	58.4	45.2	39.5	**56.4**$_{\pm.001}$	**42.3**$_{\pm.001}$	**36.2**$_{\pm.001}$
LN-I, families	**59.3**	**45.1**	**39.0**	**59.4**	**45.9**	**40.5**	56.3$_{\pm.01}$	41.3$_{\pm.01}$	34.9$_{\pm.005}$
LN-II, $\Sigma_k^{(0)} = \mathbf{I}$	26.1	24.0	22.8	27.9	26.1	25.3	22.0$_{\pm.02}$	20.1$_{\pm.02}$	19.1$_{\pm.02}$
LN-II, families	24.9	21.0	19.2	26.3	22.8	21.5	26.6$_{\pm.003}$	22.7$_{\pm.003}$	20.8$_{\pm.0006}$

Results: Additional Languages



Green: MLE, yellow: Dirichlet-I, blue: LN-I, $\sum k\uparrow(0) \text{▒}=\mathbf{I}$, red: LN-I, families initializer

Experiment: SLN Priors

Add normal experts to tie probabilities of related parents (defined by coarse tags) for each direction

1) Verbal parents
2) Nominal parents
3) Verbs and nouns
4) Adjectival parents

Results: SLN Priors and Bilingual Data

			English			Portuguese			Turkish		
			≤ 10	≤ 20	all	≤ 10	≤ 20	all	≤ 10	≤ 20	all
		MLE	46.1	39.9	35.9	44.3	35.4	29.3	35.6	32.4	31.4
		Dirichlet-I	46.1	40.6	36.9	43.8	34.1	28.0	38.6	36.7	35.9
		$\Sigma_k^{(0)} = \mathbf{I}$	59.1	45.9	40.5	45.6	**45.9**	**46.5**	55.3	47.2	44.0
		families	59.4	45.9	40.5	45.9	44.0	44.4	55.5	47.6	44.4
Trained with	English	TIEV	60.2	46.2	40.0	45.4	43.7	44.5	† 56.5	**48.7**	45.5
		TIEN	60.2	46.7	40.9	45.7	44.3	45.0	51.1	43.7	41.2
		TIEV&N	61.3	47.4	41.4	46.3	44.6	45.1	55.9	48.2	45.2
		TIEA	59.9	45.8	39.6	45.4	43.8	44.6	49.8	43.2	40.8
	Portuguese	TIEV	62.1	48.1	42.2	45.2	42.3	42.3	**56.7**	† 48.6	45.1
		TIEN	60.7	46.9	40.9	45.7	42.8	42.9	33.2	29.8	28.7
		TIEV&N	61.4	47.8	42.0	46.3	44.6	45.1	**56.7**	49.2	**46.0**
		TIEA	62.1	47.8	41.8	45.2	42.7	42.7	31.5	28.4	27.5
	Turkish	TIEV	**62.5**	**48.3**	**42.4**	45.4	43.2	43.7	55.2	47.3	44.0
		TIEN	61.0	47.2	41.2	45.9	43.9	44.4	45.1	39.8	37.8
		TIEV&N	† 62.3	**48.3**	† 42.3	**46.7**	44.3	44.6	55.7	**48.7**	45.5
		TIEA	† 62.3	48.0	42.1	45.1	43.2	43.7	38.6	34.0	32.5

Experiment: Bilingual Data

- Merge models for 2 languages
- Normal experts
  - For each POS tag
  - For each language combining multinomials in coarse POS classes
  - For 2 languages together combining multinomials in coarse POS classes
- Non-parallel corpora

Results: SLN Priors and Bilingual Data

			English			Portuguese			Turkish		
			≤ 10	≤ 20	all	≤ 10	≤ 20	all	≤ 10	≤ 20	all
		MLE	46.1	39.9	35.9	44.3	35.4	29.3	35.6	32.4	31.4
		Dirichlet-I	46.1	40.6	36.9	43.8	34.1	28.0	38.6	36.7	35.9
		$\Sigma_k^{(0)} = \mathbf{I}$	59.1	45.9	40.5	45.6	**45.9**	**46.5**	55.3	47.2	44.0
		families	59.4	45.9	40.5	45.9	44.0	44.4	55.5	47.6	44.4
Trained with	English	TIEV	60.2	46.2	40.0	45.4	43.7	44.5	† 56.5	**48.7**	45.5
		TIEN	60.2	46.7	40.9	45.7	44.3	45.0	51.1	43.7	41.2
		TIEV&N	61.3	47.4	41.4	46.3	44.6	45.1	55.9	48.2	45.2
		TIEA	59.9	45.8	39.6	45.4	43.8	44.6	49.8	43.2	40.8
	Portuguese	TIEV	62.1	48.1	42.2	45.2	42.3	42.3	**56.7**	† 48.6	45.1
		TIEN	60.7	46.9	40.9	45.7	42.8	42.9	33.2	29.8	28.7
		TIEV&N	61.4	47.8	42.0	46.3	44.6	45.1	**56.7**	49.2	**46.0**
		TIEA	62.1	47.8	41.8	45.2	42.7	42.7	31.5	28.4	27.5
	Turkish	TIEV	**62.5**	**48.3**	**42.4**	45.4	43.2	43.7	55.2	47.3	44.0
		TIEN	61.0	47.2	41.2	45.9	43.9	44.4	45.1	39.8	37.8
		TIEV&N	† 62.3	**48.3**	† 42.3	**46.7**	44.3	44.6	55.7	**48.7**	45.5
		TIEA	† 62.3	48.0	42.1	45.1	43.2	43.7	38.6	34.0	32.5

Discussion

- Modeling covariance within and across the multinomials in a probabilistic grammar can improve performance in dependency grammar induction
- Some gains from training joint models on non-parallel corpera for multiple languages
- Is there a better way to represent prior linguistic knowledge besides covariance structure?

Conclusions

- Used logistic normal distribution to model covariance between parameters of probabilistic grammar
- Extended LN distribution to model covariance across multinomials in probabilistic grammar
- Introduced variational inference algorithm for probabilistic grammars that use LN priors
- Experiments in grammar induction on multiple languages