On the Utility of Curricula in Unsupervised Learning of Probabilistic Grammars (Supplementary Material)

Kewei Tu
Department of Computer Science
Iowa State University
Ames, IA 50011, USA
tukw@iastate.edu

Vasant Honavar
Department of Computer Science
Iowa State University
Ames, IA 50011, USA
honavar@iastate.edu

This technical report provides supplementary material for the paper "On the Utility of Curricula in Unsupervised Learning of Probabilistic Grammars" [18]. Section 1 provides the proofs of the theorems in Section 3 of the paper. Section 2 gives more details of the experimental settings and results. Section 3 discusses the related work.

1 Proofs of Theorems

We first prove Theorem 1 in Section 3 of the paper.

Theorem 1 *If a curriculum $\langle W_1, W_2, \ldots, W_n \rangle$ satisfies incremental construction (with either condition 3 or 3b), then for any i, j, k s.t. $1 \leq i < j < k \leq n$, we have*

$$\begin{aligned} d_1(\theta_i, \theta_k) &\geq d_1(\theta_j, \theta_k) \\ d_{TV}(G_i, G_k) &\geq d_{TV}(G_j, G_k) \end{aligned}$$

where $d_1(\cdot, \cdot)$ denotes the L_1 distance; $d_{TV}(G_i, G_j)$ represents the total variation distance between the two distributions of grammatical structures defined by G_i and G_j.

Proof: Here we assume condition 3b because it is more general than condition 3. There are two inequalities in the conclusion of the theorem. We first give the

proof of the inequality with the L_1 distance of parameter vectors. As defined in Section 3 of the paper, a parameter vector is the concatenation of a set of multinomial vectors, each of which is the vector of probabilities of grammar rules with a specific rule condition (left-hand side) of the target grammar. Denote $\theta_{i,p}$ as the multinomial vector of rule condition p in grammar G_i, and denote $\theta_{i,p\to q}$ as the probability of rule $p\to q$ in grammar G_i. Note that

$$d_1(\theta_i, \theta_j) = \sum_p d_1(\theta_{i,p}, \theta_{j,p})$$

So to prove the first inequality, it is sufficient to prove that

$$\forall p,\ d_1(\theta_{i,p}, \theta_{k,p}) \geq d_1(\theta_{j,p}, \theta_{k,p})$$

Because the L_1 norm of a multinomial vector is always 1, for any rule condition p we have

$$\begin{aligned}
d_1(\theta_{i,p}, \theta_{j,p}) &= \sum_{q:\theta_{i,p\to q} > \theta_{j,p\to q}} (\theta_{i,p\to q} - \theta_{j,p\to q}) + \sum_{q:\theta_{i,p\to q} \leq \theta_{j,p\to q}} (\theta_{j,p\to q} - \theta_{i,p\to q}) \\
&= \left(1 - \sum_{q:\theta_{i,p\to q} \leq \theta_{j,p\to q}} \theta_{i,p\to q}\right) - \left(1 - \sum_{q:\theta_{i,p\to q} \leq \theta_{j,p\to q}} \theta_{j,p\to q}\right) \\
&\quad + \sum_{q:\theta_{i,p\to q} \leq \theta_{j,p\to q}} (\theta_{j,p\to q} - \theta_{i,p\to q}) \\
&= 2 \times \sum_{q:\theta_{i,p\to q} \leq \theta_{j,p\to q}} (\theta_{j,p\to q} - \theta_{i,p\to q}) \\
&= 2 \times \sum_q (\theta_{j,p\to q} - \theta_{i,p\to q}) f_{i,j,p}(q) \qquad (1)
\end{aligned}$$

where $f_{i,j,p}(q)$ is defined as

$$f_{i,j,p}(q) = \begin{cases} 1 & \text{if } \theta_{i,p\to q} \leq \theta_{j,p\to q} \\ 0 & \text{if } \theta_{i,p\to q} > \theta_{j,p\to q} \end{cases}$$

According to Definition 1 (with condition 3b) of the paper, for any grammar rule $p\to q$ in the target grammar, with the increase of i, its probability $\theta_{i,p\to q}$ first remains 0, then shifts to a non-zero value in a certain intermediate grammar, and after that decreases monotonically. So for any $i < j < k$, there are three possibilities, which we consider in turn.

1. If $\theta_{i,p\to q} = \theta_{j,p\to q} = 0$ and $\theta_{k,p\to q} \geq 0$, then we have

$$(\theta_{k,p\to q} - \theta_{i,p\to q}) f_{i,k,p}(q) = (\theta_{k,p\to q} - \theta_{j,p\to q}) f_{j,k,p}(q)$$

2. If $\theta_{i,p\to q} = 0$ and $\theta_{j,p\to q} \geq \theta_{k,p\to q} > 0$, then we have

$$(\theta_{k,p\to q} - \theta_{i,p\to q}) f_{i,k,p}(q) > 0 = (\theta_{k,p\to q} - \theta_{j,p\to q}) f_{j,k,p}(q)$$

3. If $\theta_{i,p\to q} \geq \theta_{j,p\to q} \geq \theta_{k,p\to q} > 0$, then we have

$$(\theta_{k,p\to q} - \theta_{i,p\to q}) f_{i,k,p}(q) = (\theta_{k,p\to q} - \theta_{j,p\to q}) f_{j,k,p}(q) = 0$$

Therefore, we get

$$\sum_q (\theta_{k,p\to q} - \theta_{i,p\to q}) f_{i,k,p}(q) \geq \sum_q (\theta_{k,p\to q} - \theta_{j,p\to q}) f_{j,k,p}(q)$$

where equality holds if there exists no assignment of q that satisfies the second possibility. According to Eq.1, we have

$$d_1(\theta_{i,p}, \theta_{k,p}) \geq d_1(\theta_{j,p}, \theta_{k,p})$$

Therefore we have proved the first inequality.

Now we turn to the second inequality in the conclusion of the theorem and prove it in a similar fashion. Because the sum of probabilities over all grammatical structures is always 1, we have

$$\begin{aligned} d_{TV}(G_i, G_j) &= \frac{1}{2} \sum_s |P(s|G_i) - P(s|G_j)| \\ &= \sum_s (P(s|G_j) - P(s|G_i)) f_{i,j}(s) \end{aligned} \tag{2}$$

where $f_{i,j}(s)$ is defined as

$$f_{i,j}(s) = \begin{cases} 1 & \text{if } P(s|G_i) \leq P(s|G_j) \\ 0 & \text{if } P(s|G_i) > P(s|G_j) \end{cases}$$

The first equality of Eq.2 is the definition of total variation, and the second equality can be derived in a similar way as in Eq.1. According to Definition 1 (with condition 3b) of the paper, for any grammatical structure s that can be generated by the target grammar, with the increase of i, the probability $P(s|G_i)$ first remains 0 (when at least one grammar rule used in deriving s is absent from G_i), then shifts to a non-zero value (when all the grammar rules needed to derive s have non-zero probabilities), and after that decreases monotonically (because the probabilities of all the grammar rules used in deriving s are decreasing). Just as in the proof of the first inequality, for any $i < j < k$ there are three possibilities, and by analyzing the three possibilities in turn we can get

$$\sum_s (P(s|G_k) - P(s|G_i)) f_{i,k}(s) \geq \sum_s (P(s|G_k) - P(s|G_j)) f_{j,k}(s)$$

So according to Eq.2, we have

$$d_{TV}(G_i, G_k) \geq d_{TV}(G_j, G_k)$$

Therefore we have proved the second inequality. (End of Proof)

Now we give a proof sketch of Theorem 2.

Theorem 2 *If a curriculum $\langle W_1, W_2, \ldots, W_n \rangle$ satisfies the first two conditions in Definition 1 as well as a further relaxed version of the third condition:*

3c. for any grammar rules r, $P(r|G_i)$ first monotonically increases with i and then monotonically decreases with i.

then for any i, j, k s.t. $1 \leq i < j < k \leq n$, we have

$$d_1(\theta_i, \theta_k) \geq d_1(\theta_j, \theta_k)$$

Proof Sketch: The proof is the same as the proof of the first inequality of Theorem 1, except that the three possibilities are changed because of condition 3c. According to condition 3c, with the increase of i, the probability of a grammar rule $\theta_{i,p\to q}$ first increases monotonically and then decreases monotonically. So for any $i < j < k$, we have three new possibilities.

1. If $\theta_{i,p\to q} \leq \theta_{j,p\to q} \leq \theta_{k,p\to q}$, then we have

$$(\theta_{k,p\to q} - \theta_{i,p\to q}) f_{i,k,p}(q) \geq (\theta_{k,p\to q} - \theta_{j,p\to q}) f_{j,k,p}(q)$$

2. If $\theta_{i,p\to q} \leq \theta_{j,p\to q}$ and $\theta_{j,p\to q} \geq \theta_{k,p\to q}$, then we have

$$(\theta_{k,p\to q} - \theta_{i,p\to q}) f_{i,k,p}(q) \geq 0 = (\theta_{k,p\to q} - \theta_{j,p\to q}) f_{j,k,p}(q)$$

3. If $\theta_{i,p\to q} \geq \theta_{j,p\to q} \geq \theta_{k,p\to q}$, then we have

$$(\theta_{k,p\to q} - \theta_{i,p\to q}) f_{i,k,p}(q) = (\theta_{k,p\to q} - \theta_{j,p\to q}) f_{j,k,p}(q) = 0$$

So we can still get

$$\sum_q (\theta_{k,p\to q} - \theta_{i,p\to q}) f_{i,k,p}(q) \geq \sum_q (\theta_{k,p\to q} - \theta_{j,p\to q}) f_{j,k,p}(q)$$

and the rest of the proof is exactly the same as in the proof of the first inequality of Theorem 1. (End of Proof Sketch)

2 Experiments

In this section we provide more details of the experiments presented in Section 4 and 5 of the paper.

We adapted the DAGEEM software[1] to implement the expectation-maximization algorithm of the DMV grammar. We then implemented curriculum learning by using expectation-maximization as the base learner. In the experiments on synthetic data, expectation-maximization was initialized with a trivial grammar in which rules with the same left-hand side have equal probabilities; in the experiments on real data, we used an initial grammar provided in the DAGEEM

[1] `http://www.ark.cs.cmu.edu/DAGEEM/`

software which is created according to the heuristic approach described in [10]. As mentioned in the paper, we used a dynamic smoothing factor in the experiments on synthetic data to alleviate the overfitting problem discussed in Section 3.1 of the paper. The dynamic smoothing factor is computed from the size of the partial corpus that is hidden from the learner during curriculum learning. More specifically, for each training sentence that is hidden, we assume it is sampled from a uniform distribution over all possible sentences that have the same length as the hidden sentence, and we also assume a uniform grammar in which rules with the same left-hand side have equal probabilities, so we can easily compute the expected counts of each grammar rule r being used in parsing this sentence; then the dynamic smoothing factor for grammar rule r is the sum of the expected counts over all the training sentences that are hidden. We found that dynamic smoothing often improves the learning result in the experiments on synthetic data; however, in the experiments on real data, dynamic smoothing usually hurts learning.

We used the WSJ30 corpus (the set of sentences no longer than 30 in the Wall Street Journal corpus of the Penn Treebank) in our experiments. Because we used the DMV grammar formalism in our experiments, which is a type of dependency grammar, we converted the phrase structure annotations in the Penn Treebank to the dependency annotations by running the "ptbconv" software[2]. When generating the synthetic data, we found the dependency treebank grammar of WSJ30 tends to generate sentences much longer than the actual sentences in WSJ30, so we decreased by 30% the probabilities of grammar rules that determine if a new dependency should be generated under a certain condition.

We tested different values of the smoothing factor in the experiments on both synthetic data and real data. We found that although the value of the smoothing factor did affect the learning performance, the advantage of curriculum learning over the baseline was consistently observed.

In Section 5.2 of the paper, we mention that the change of rule probabilities during learning with a curriculum is similar to the change plotted in Figure 2(c) of the paper (which plots the change of rule probabilities in the sequence of intermediate grammars specified by the curriculum). Here we show the actual plot of the change during learning for VBD-headed grammar rules (Figure 1). It can be seen that the probabilities of most rules first rise and then drop, and rules are learned in a specific order according to the curriculum. However, we can also see that some rules behave differently than specified by the curriculum, which is due to the errors or alternative parses made by the unsupervised learner. For example, the unsupervised learner learns to assign DT (determiner) as the head of a noun phrase, so in Figure 1 we see a curve for the rule VBD→DT, which is not present in Figure 2(c) of the paper.

[2]Available at `http://www.jaist.ac.jp/~h-yamada/`

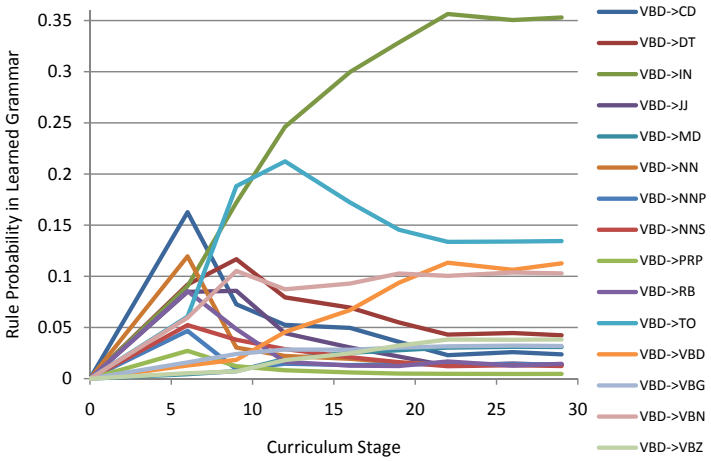

Figure 1: The change of probabilities of VBD-headed rules with the stages of the length-based curriculum during learning (best viewed in color). Rules with probabilities always below 0.025 are omitted.

3 Discussions

We first give a brief survey of existing work on unsupervised learning of probabilistic grammars. The inside-outside algorithm [1, 13] is one of the earliest algorithms for learning probabilistic context-free grammars and is a special case of the EM algorithm. It assumes a fixed, usually fully connected grammar structure and tries to maximize the likelihood of the grammar. A similar algorithm is proposed to learn dependency grammars, but with a heuristic initialization to mitigate the local minimum problem of EM [10]. More recent work has adopted the Bayesian framework to maximize the posterior of the learned grammar given the corpus. A variational inference method is derived in [11], while a Markov Chain Monte Carlo method is presented in [9]. A Dirichlet prior with less-than-one hyperparameters is usually used in the Bayesian framework, because it encourages smaller grammars to avoid over-fitting; other priors have also been used, e.g., the logistic normal prior that models the correlations between symbols [3, 4] and the hierarchical Dirichlet process prior that can accommodate an unbounded number of nonterminals [5, 14]. The linear-interpolation smoothing used in [8] can also be seen as a kind of prior. The methods mentioned so far all assume a fixed grammar structure and try to infer the parameters, but structure search can also be incorporated in learning [17, 2, 12]. There are also some methods that go beyond standard probabilistic inference. Structural annealing [15] controls the strength of two types of structural bias to guide the iterative learning. The approach of [6] uses posterior regularization to encode sparsity bias that cannot be easily expressed by priors, and applies a novel iterative algorithm for optimization. UNSEARN [7] adapts a supervised struc-

tured prediction algorithm for unsupervised use, and applies it to unsupervised dependency grammar learning. As we discussed in the paper, most of these existing methods start with all the training sentences and try to learn the whole grammar, with the exception of the Baby-step algorithm [16] which starts the learning with short sentences and then adds increasingly longer sentences into the training corpus.

Curriculum learning is related to boosting algorithms in that both learn from a weighted training set in a iterative fashion, with the weights being evolved from one iteration to the next. However, there are a few important differences between the two. First, boosting starts with a uniform weighting scheme and modifies the weights based on the learner's performance on the training data, whereas curriculum learning starts with a weighting scheme that favors easy samples and ends with a uniform weighting scheme. The easiness measure of training samples in curriculum learning is usually based on some external knowledge (e.g., the prior knowledge that shorter sentences are easier), which therefore introduces additional information into learning. In addition, in boosting we learn a set of base learners and then combine them by weighted voting, while in curriculum learning we continuously update a single learner.

The likelihood-based curriculum learning proposed in Section 5.2 of the paper is related to active learning, in that it introduces new training samples to the learner based on the grammar that has been learned. The likelihood-based curriculum learning also resembles some self-training approaches, in that it reweights training samples based on the probabilities of the samples given the learned grammar, and such probabilities reflect the confidence of the learner in parsing the training samples.

References

[1] J. K. Baker. Trainable grammars for speech recognition. In *Speech Communication Papers for the 97th Meeting of the Acoustical Society of America*, 1979.

[2] Stanley F. Chen. Bayesian grammar induction for language modeling. In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, 1995.

[3] Shay B. Cohen, Kevin Gimpel, and Noah A. Smith. Logistic normal priors for unsupervised probabilistic grammar induction. In *NIPS*, pages 321–328, 2008.

[4] Shay B. Cohen and Noah A. Smith. Shared logistic normal distributions for soft parameter tying in unsupervised grammar induction. In *HLT-NAACL*, pages 74–82, 2009.

[5] Jenny Rose Finkel, Trond Grenager, and Christopher D. Manning. The infinite tree. In *Proceedings of the 45th Annual Meeting of the Association of*

Computational Linguistics, pages 272–279. Association for Computational Linguistics, June 2007.

[6] Jennifer Gillenwater, Kuzman Ganchev, ao Graça, Jo Fernando Pereira, and Ben Taskar. Sparsity in dependency grammar induction. In *ACL '10: Proceedings of the ACL 2010 Conference Short Papers*, pages 194–199, Morristown, NJ, USA, 2010. Association for Computational Linguistics.

[7] Hal Daumé III. Unsupervised search-based structured prediction. In *ICML*, page 27, 2009.

[8] William P. Headden III, Mark Johnson, and David McClosky. Improving unsupervised dependency parsing with richer contexts and smoothing. In *HLT-NAACL*, pages 101–109, 2009.

[9] Mark Johnson, Thomas L. Griffiths, and Sharon Goldwater. Bayesian inference for pcfgs via markov chain monte carlo. In *HLT-NAACL*, pages 139–146, 2007.

[10] Dan Klein and Christopher D. Manning. Corpus-based induction of syntactic structure: Models of dependency and constituency. In *Proceedings of ACL*, 2004.

[11] Kenichi Kurihara and Taisuke Sato. An application of the variational Bayesian approach to probabilistic contextfree grammars. In *IJCNLP-04 Workshop beyond shallow analyses*. 2004.

[12] Kenichi Kurihara and Taisuke Sato. Variational Bayesian grammar induction for natural language. In *ICGI 2006*, volume 4201 of *LNAI*, pages 84–96, 2006.

[13] K. Lari and S. Young. The estimation of stochastic context-free grammars using the inside-outside algorithm. *Computer Speech and Language*, 4:35–36, 1990.

[14] Percy Liang, Slav Petrov, Michael I. Jordan, and Dan Klein. The infinite pcfg using hierarchical Dirichlet processes. In *Proceedings of EMNLP-CoNLL*, pages 688–697, 2007.

[15] Noah A. Smith and Jason Eisner. Annealing structural bias in multilingual weighted grammar induction. In *ACL*, 2006.

[16] Valentin I. Spitkovsky, Hiyan Alshawi, and Daniel Jurafsky. From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing. In *NAACL*, 2010.

[17] Andreas Stolcke and Stephen M. Omohundro. Inducing probabilistic grammars by Bayesian model merging. In *ICGI*, pages 106–118, 1994.

[18] Kewei Tu and Vasant Honavar. On the utility of curricula in unsupervised learning of probabilistic grammars. In *Proceedings of the Twenty-second International Joint Conference on Artificial Intelligence (IJCAI 2011)*, 2011.