Baby Steps: How “Less is More” in Unsupervised Dependency Parsing

Valentin I. Spitkovsky
Computer Science Department
Stanford University and Google Inc.
valentin@cs.stanford.edu
valentin@google.com

Hiyan Alshawi
Google Inc.
1600 Amphitheatre Parkway
Mountain View, CA, 94043, USA
hiyan@google.com

Daniel Jurafsky
Departments of Linguistics and Computer Science
Stanford University, Stanford, CA, 94305, USA
jurafsky@stanford.edu

Abstract

We present an empirical study of two very simple approaches to unsupervised grammar induction. Both are based on Klein and Manning’s Dependency Model with Valence. The first, *Baby Steps*, requires no initialization and bootstraps itself via iterated learning of increasingly longer sentences. This method substantially exceeds Klein and Manning’s published numbers and achieves 39.4% accuracy on Section 23 of the Wall Street Journal corpus — a result that is already competitive with the recent state-of-the-art. The second, *Less is More*, is based on the observation that there is sometimes a trade-off between the quantity and complexity of training data. Using the standard linguistically-informed prior but training at the “sweet spot” — sentences up to length 15, it attains 44.1% accuracy, beating state-of-the-art. Both results generalize to the Brown corpus and shed light on opportunities in the present state of unsupervised dependency parsing.

1 Introduction

Unsupervised learning of hierarchical syntactic structure from free-form natural language text is a hard problem whose eventual solution promises to benefit applications ranging from question answering to speech recognition and machine translation. In recent years, a restricted version of the problem (which assumes partial annotation, in the form of sentence boundaries, tokenization and typically even part-of-speech tagging) has received much attention, eliciting a diverse array of techniques [1, 2, 3, 4, 5, 6, 7]. In 2004, Klein and Manning’s Dependency Model with Valence (DMV) became the first to outperform a very simple parsing heuristic — the right-branching baseline [1]. Today, in 2009, state-of-the-art systems are still rooted in the DMV [7, 6].

Despite recent advances, unsupervised parsers still lag far behind their supervised counterparts. Nevertheless, extreme cost and limited coverage of manually-annotated corpora strongly motivate unsupervised learning in general [8] and unsupervised parsing in particular [9]. Large amounts of unlabeled data have been shown to improve semi-supervised parsing [10], yet the best unsupervised systems use even less data than is available for supervised training, relying on complex models instead [7, 6]. Headden III et al.’s Extended Valence Grammar (EVG) combats data sparsity with smoothing alone, training on the same small subset of the tree-bank as the original DMV [7]. Cohen and Smith use more complicated algorithms (variational EM and MBR decoding) and stronger linguistic hints (tying related parts of speech and syntactically similar bilingual data) [6].

In this work, we adopt a contrarian stance and ask what can be achieved through judicious use of data and simple, scalable techniques. Our first approach iterates over a series of training sets that gradually increase in size and complexity, forming an initialization-independent scaffolding for learning a grammar. It works with Klein and Manning's simple model (the original DMV) and training algorithm (classic EM) but eliminates their crucial dependence on manually-tuned linguistically-biased priors [1]. Our second approach builds on the observation that learning is most successful within a narrow band of the size-complexity spectrum. Both insights generalize beyond the DMV and could be applied to more intricate models and advanced algorithms.

2 Baby Steps

Global non-convex optimization is hard [11] and initialization is important to the success of any local search procedure [1]. We offer Baby Steps as a meta-heuristic for finding approximate solutions without the guess-work. Its underlying assumption is that a good solution to a closely related problem should be a fine starting point for the actual problem of interest. The main idea of Baby Steps is to decompose a difficult problem into a sequence of approximations that begins with an easy case and extends it to the problem we care about. The intuition is that if we found a way to increase complexity very gradually, taking tiny steps in the problem space, then there may be hope for preserving continuity in the solution space as well.

When instances of a problem themselves suffer from a combinatorially-exploding solution space, the size of that space presents a natural proxy for complexity. In the case of parsing, the number of possible syntactic trees grows exponentially with the length of the sentence. Consequently, for longer sentences, the unsupervised optimization problem becomes severely under-constrained, whereas for shorter sentences, it is still tightly reined in by data. In the extreme case of a single-word sentence, there is no choice but to parse it correctly. For two-word sentences, the chance of correctly guessing the head and its dependent is still high, at 50%. But as sentences grow in length, the accuracy of even educated guessing rapidly plummets, suggesting that longer sentences are more difficult.

Baby Steps works with a series of nested subsets of increasingly longer sentences that culminates in the complete data set. The base case — sentences of length one — has a trivial solution that requires no initialization or search and reveals something about sentence heads. The next step — sentences of length one and two — refines the initial impression of heads, introduces dependents, and exposes their identities and positions relative to the heads. Although short sentences are not representative of the full complexity of the grammar, they capture enough information to paint most of the picture needed by slightly longer sentences. This sets up an easier, incremental subsequent learning task. Step $k+1$ augments the training set with sentences of length $k+1$, now including lengths $1, 2, \ldots, k, k+1$ of the full data set, and executes local search starting from the grammar estimated by step k. This truly is grammar induction.

3 Experimental Setup: Data Sets and Metrics

Klein and Manning [1] both trained and tested the DMV on the same customized subset (WSJ10) of Penn English Treebank's Wall Street Journal portion [12]. Its 49,208 annotated parse trees were pruned[1] down to 7,422 sentences of at most 10 terminals, spanning 35 unique part-of-speech (POS) tags. Following standard practice, automatic "head-percolation" rules [13] were used to convert the remaining trees into dependencies. Forced to produce a single "best" parse, their algorithm was judged on accuracy: its *directed score* was the fraction of correct dependencies; a more flattering[2] *undirected score* was also used. We employ the same metrics but emphasize the directed scores. Generalizing Klein and Manning's setup, we let WSJx be the subset of pre-processed sentences with at most x terminals. Our experiments focus on $x \in \{1, \ldots, 45\}$, but we also test on WSJ100 and Section 23 of WSJ$^\infty$ (the entire WSJ). In addition, we test on the held-out Brown100 (similarly derived from the Brown corpus [14]). See Figure 1 for sentence and token counts of these corpora.

[1] Stripped of all empty sub-trees, punctuation, and terminals (tagged # and $) not pronounced where they appear, those sentences still containing more than ten tokens were thrown out.

[2] Ignoring polarity of parent-child relations partially obscured effects of alternate analyses (systematic choices between modals and main verbs for heads of sentences, determiners for noun phrases, etc.) and facilitated comparison with prior work.

Corpus	*Sentences*	*POS Tokens*
WSJ1	159	159
WSJ2	499	839
WSJ3	876	1,970
WSJ4	1,394	4,042
WSJ5	2,008	7,112
WSJ6	2,745	11,534
WSJ7	3,623	17,680
WSJ8	4,730	26,536
WSJ9	5,938	37,408
WSJ10	7,422	52,248
WSJ11	8,856	68,022
WSJ12	10,500	87,750

Corpus	*Sentences*	*POS Tokens*
WSJ13	12,270	110,760
WSJ14	14,095	136,310
WSJ15	15,922	163,715
WSJ20	25,523	336,555
WSJ25	34,431	540,895
WSJ30	41,227	730,099
WSJ35	45,191	860,053
WSJ40	47,385	942,801
WSJ45	48,418	986,830
WSJ100	49,206	1,028,054
Section 23	2,353	48,201
Brown100	24,208	391,796

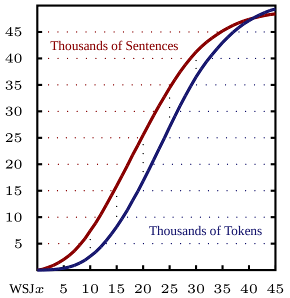
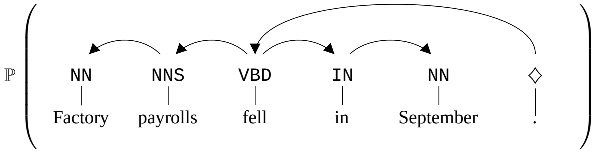


Figure 1: Sizes of WSJ$\{1, \dots, 45, 100\}$, Section 23 of WSJ$^\infty$ and Brown100.

4 Experimental Methods: New Algorithms for the Classic Model

We ground our experimental design in the DMV's simple generative process [1]. Operating over lexical word classes $\{c_w\}$ — POS tags, its generative story for a sub-tree rooted at a head (of class c_h) rests on the independence assumptions inherent in the following decisions: (i) initial direction $dir \in \{\texttt{L}, \texttt{R}\}$ in which to attach children, via probability $\mathbb{P}_{\texttt{ORDER}}(c_h)$; (ii) whether to seal dir, stopping with probability $\mathbb{P}_{\texttt{STOP}}(c_h, dir, adj)$, conditioned on $adj \in \{\texttt{T}, \texttt{F}\}$ (true iff considering dir's first or *adjacent* child); and (iii) attachment (of class c_a), according to $\mathbb{P}_{\texttt{ATTACH}}(c_h, dir, c_a)$. This produces only projective trees, disallowing crossing dependencies (in contrast to methods like spanning tree algorithms [15] that do not use charts). A special token $\diamondsuit$ generates the head of the sentence as its left (and only) child. See Figure 2 for a simple example that ignores (sums out) $\mathbb{P}_{\texttt{ORDER}}$.



$$\begin{array}{llllllll}
= & (1-\overbrace{\mathbb{P}_{\texttt{STOP}}(\diamond,\texttt{L},\texttt{T})}^{0}) & \times & \mathbb{P}_{\texttt{ATTACH}}(\diamond,\texttt{L},\texttt{VBD}) & \times & \overbrace{\mathbb{P}_{\texttt{STOP}}(\diamond,\texttt{L},\texttt{F})}^{1} & \times & \overbrace{\mathbb{P}_{\texttt{STOP}}(\diamond,\texttt{R},\texttt{T})}^{1} \\
\times & (1-\mathbb{P}_{\texttt{STOP}}(\texttt{VBD},\texttt{L},\texttt{T})) & \times & \mathbb{P}_{\texttt{ATTACH}}(\texttt{VBD},\texttt{L},\texttt{NNS}) & \times & \mathbb{P}_{\texttt{STOP}}(\texttt{VBD},\texttt{L},\texttt{F}) & & \\
\times & (1-\mathbb{P}_{\texttt{STOP}}(\texttt{VBD},\texttt{R},\texttt{T})) & \times & \mathbb{P}_{\texttt{ATTACH}}(\texttt{VBD},\texttt{R},\texttt{IN}) & \times & \mathbb{P}_{\texttt{STOP}}(\texttt{VBD},\texttt{R},\texttt{F}) & & \\
\times & (1-\mathbb{P}_{\texttt{STOP}}(\texttt{NNS},\texttt{L},\texttt{T})) & \times & \mathbb{P}_{\texttt{ATTACH}}(\texttt{NNS},\texttt{L},\texttt{NN}) & \times & \mathbb{P}_{\texttt{STOP}}(\texttt{NNS},\texttt{L},\texttt{F}) & \times & \mathbb{P}_{\texttt{STOP}}(\texttt{NNS},\texttt{R},\texttt{T}) \\
\times & (1-\mathbb{P}_{\texttt{STOP}}(\texttt{IN},\texttt{R},\texttt{T})) & \times & \mathbb{P}_{\texttt{ATTACH}}(\texttt{IN},\texttt{R},\texttt{NN}) & \times & \mathbb{P}_{\texttt{STOP}}(\texttt{IN},\texttt{R},\texttt{F}) & \times & \mathbb{P}_{\texttt{STOP}}(\texttt{IN},\texttt{L},\texttt{T}) \\
\times & \mathbb{P}^2_{\texttt{STOP}}(\texttt{NN},\texttt{L},\texttt{T}) & \times & \mathbb{P}^2_{\texttt{STOP}}(\texttt{NN},\texttt{R},\texttt{T}). & & & &
\end{array}$$

Figure 2: A dependency structure for a short sentence and its probability according to the DMV.

The DMV lends itself to unsupervised learning via EM and inside-outside re-estimation [16]. Klein and Manning estimated all probabilities without smoothing. Lacking a global optimization routine, they guarded EM's sensitivity to initial conditions by starting with an "ad-hoc harmonic" completion: non-root heads took the same number of dependents, attaching them in inverse proportion to (a constant plus) their distance; $\diamondsuit$ took its single dependent uniformly at random. This non-distributional heuristic nudged EM towards typical linguistic dependency structures.

4.1 New Algorithm #1: Ad-Hoc* — A Variation on the Original Ad-Hoc Initialization

With crucial implementation details of the original initialization parameters and termination conditions absent from the literature [1, 17], reimplementing the DMV required a fair bit of improvisation. Everything in this section is a guess and likely does not match Klein and Manning's actual choices.

We use the following ad-hoc harmonic scores (for all tokens other than $\diamondsuit$):

$$\mathbb{S}_{\texttt{ORDER}} \equiv \frac{1}{2}, \quad \mathbb{S}_{\texttt{STOP}} \equiv \frac{1}{d_s + k_s} = \frac{1}{d_s + 3} \text{ and } \mathbb{S}_{\texttt{ATTACH}} \equiv \frac{1}{d_a + k_a} = \frac{1}{d_a + 2},$$

with $d_s \geq 0$ a head's distance to the stopping boundary and $d_a \geq 1$ its distance to a child. The integer constants k_s and k_a come from related code in Stanford's JavaNLP Project [18]. Note that

k_s is one higher than is strictly necessary to avoid both division by zero and determinism, whereas k_a could have been safely zeroed out entirely, since we never compute $1 - \mathbb{P}_{\texttt{ATTACH}}$ (see Figure 2).

Using these scoring functions, we initialize training by producing the best-scoring parses of all input sentences and converting them into proper probability distributions $\mathbb{P}_{\texttt{STOP}}$ and $\mathbb{P}_{\texttt{ATTACH}}$ via maximum-likelihood estimation (a single step of Viterbi training [19]). Since left children are independent of those on the right, we drop $\mathbb{P}_{\texttt{ORDER}}$ altogether, making "headedness" deterministic. Our parser is careful to randomize tie-breaking, so that all parse trees of a particular sentence that have the same score have an equal shot at being selected (both during initialization and evaluation).

Finally, we terminate EM when successive changes in per-token perplexity drop below 2^{-20} bits.

4.2 New Algorithm #2: Baby Steps — An Initialization-Independent Scaffolding

We eliminate the need for initialization by first training on a trivial subset of the data — WSJ1; this works, since there is only one (the correct) way to parse a single-token sentence. The resulting model is then used to initialize training on WSJ2 (sentences of up to two tokens), and so forth, building up to WSJ45's 48,418 sentences (these cover 94.4% of all sentences in WSJ; the longest of the missing 790 has length 171). This algorithm is otherwise identical to Ad-Hoc*, with the exception that it re-estimates each model using Laplace smoothing, so that earlier solutions could be passed to next levels (which sometimes contain previously unseen dependent and head POS tags).

4.3 Baselines: Uninformed, Oracle and Previously Published State-of-the-Art Results

To better appreciate the problem space, we consider two extreme initialization strategies. The uninformed uniform prior serves as a fair "zero-knowledge" baseline for comparing uninitialized models. The maximum-likelihood "oracle" prior, computed from reference parses, serves as a "skyline" — a bound for how an algorithm that stumbled on the true solution would fare at EM's convergence.

In addition to citing Klein and Manning's Ad-Hoc's numbers [1], we compare our results on Section 23 of WSJ$^\infty$ to other past baselines (see Table 2). Headden III et al.'s lexicalized results are by far the strongest on short sentences, but they unfortunately do not report the EVG's performance for the more complex and realistic test sets [7]; to the best of our knowledge, Cohen and Smith's are the highest reported numbers for longer sentences [6]. In addition to these two state-of-the-art systems, we include revealing intermediate results [5] that preceded the parameter-tying approach [6]. These include Bayesian models with Dirichlet [5] and various log-normal [5] priors, coupled with both Viterbi and minimum Bayes-risk (MBR) decoding [5, 6].

4.4 Hypothesis: "Less is More" — An Anticipated Size-Complexity Trade-Off

Having conjectured that sentence length is a good proxy for complexity, we suspect that the very long sentences may present too much ambiguity (see Section 2) to the under-constrained learning problem. But the simpler short sentences are few and may not capture the full richness of the grammar. This suggests the possibility of a "sweet spot" at WSJx, for x not too high (excluding the truly daunting training examples) and not too low (including enough moderately accessible information).

5 Experimental Result #1: Baby Steps

We traced out a curve of Baby Steps' performance when trained and tested on WSJ$(x + 1)$, using its solution to WSJx as initialization, for $x < 45$ (see Figure 3). Klein and Manning's published results, 43.2% (63.7%) [1], appear as dots at WSJ10, where Baby Steps achieves 53.0% (65.7%); trained and tested on WSJ45, Baby Steps scores 39.7% (54.3%). Classic EM learns very little about directed dependencies without the benefit of a linguistically-biased prior for the DMV: it improves slightly, e.g. from 17.3% (34.2%) to 19.1% (46.5%) on WSJ45, learning a little of the structure (as evidenced by its undirected scores), but actually gets worse on shorter sentences, where its initial guessing rate is high. And while we expected EM to walk away from supervised solutions [20, 21], the extent of its drop there is truly shocking, e.g. from 69.8% (72.2%) to 50.6% (59.5%) on WSJ45. Not surprisingly, Baby Steps' scores do not change much from one step to the next; but where its changes are big, they are always positive.

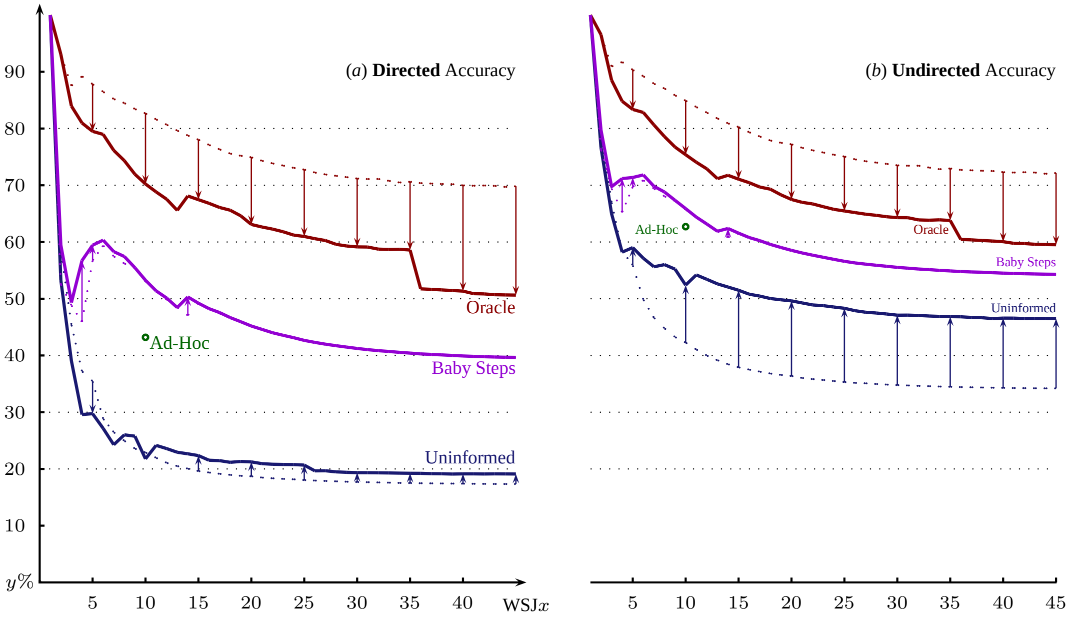


Figure 3: Directed and undirected accuracy scores attained by the DMV, when trained and tested on the same gradation of WSJ, for several different initialization strategies. The two green circles represent Klein and Manning's published numbers [1]; from top to bottom, red, violet and blue curves represent the supervised (maximum-likelihood oracle) initialization, Baby Steps, and the uninformed uniform prior. Dotted curves reflect starting performance, solid curves register performance at EM's convergence, and the arrows connecting them emphasize the impact of learning.

We also explored how Klein and Manning's initializer may have fared at different gradations of WSJ, by tracing out a similar curve for Ad-Hoc* (see Figure 4). Somewhat surprisingly, our implementation performs significantly better than their published numbers at WSJ10: 54.5% (68.3%), scoring slightly higher than Baby Steps; nevertheless, given enough data (from WSJ22 onwards), Baby Steps outperforms Ad-Hoc*, whose ability to learn takes a serious dive once the data set becomes sufficiently complex (at WSJ23) and never recovers. Note that the linguistically-biased prior peaks early (at WSJ6), eventually falling below the guessing rate (by WSJ24), but nevertheless remains well-positioned to climb beyond the uninformed uniform prior's performance.

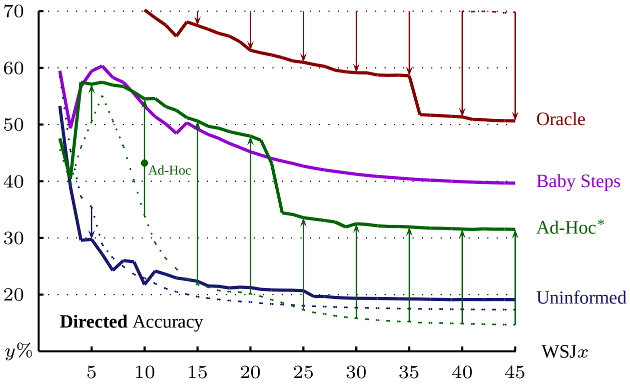
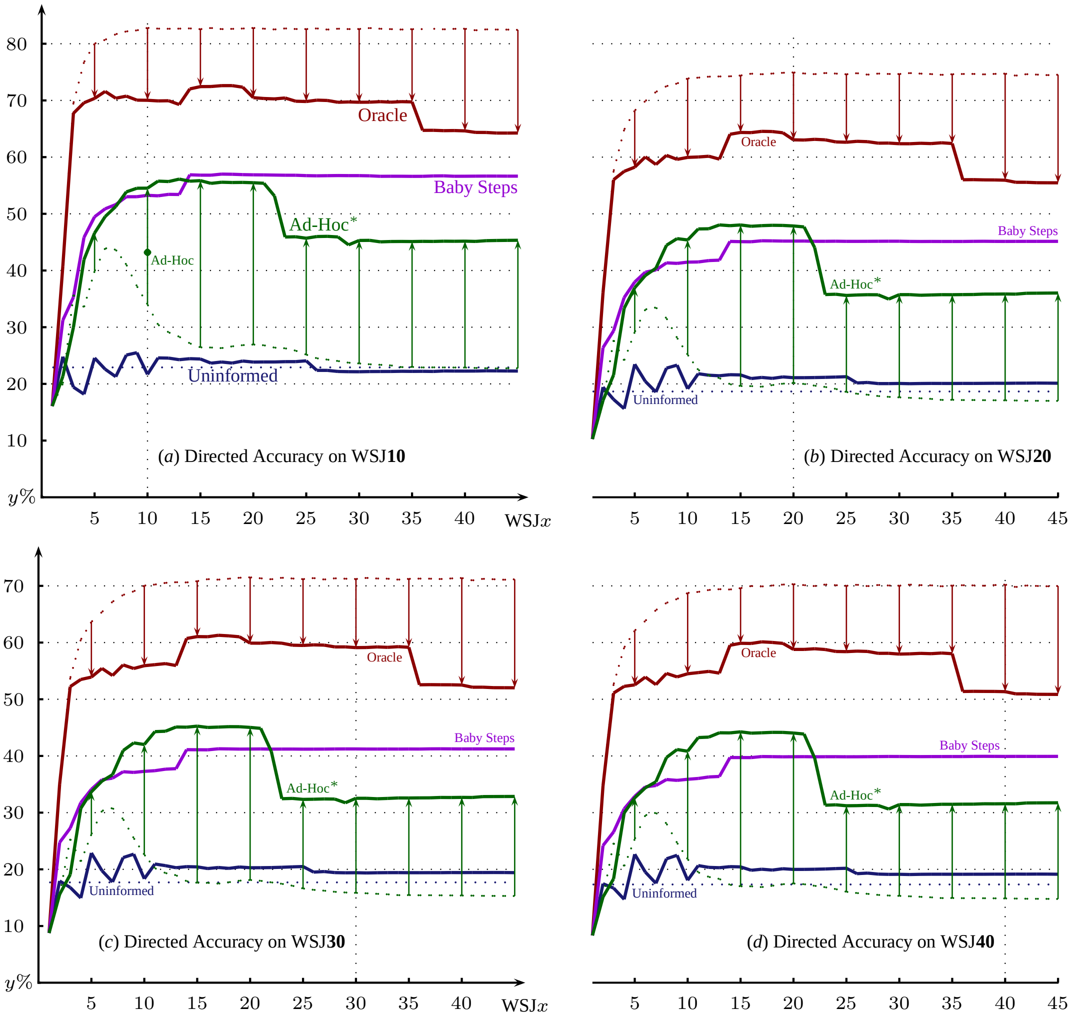


Figure 4: Directed accuracy for the DMV using Ad-Hoc*, shown in green, when trained and tested on the same gradation of WSJ. As in Figure 3, the green circle corresponds to Klein and Manning's published score [1]; red, violet and blue curves represent supervised, Baby Steps, and uniform initialization strategies. Dotted curves reflect starting performance, solid curves register performance at EM's convergence, and the arrows connecting them emphasize the impact of learning.

6 Experimental Result #2: “Less is More”

The graphs in the previous section (Figures 3 and 4) could be misleading, as they do not tell the whole story of how learning scales with more (complex) data. They are difficult to interpret because, on the one hand, as the data set increases in size, the training algorithm gets access to more information; on the other, since in this unsupervised setting the training and test sets are the same, additional longer sentences make for significantly more challenging evaluation. To better understand these dynamics, we applied Laplace smoothing to all models (other than Baby Steps, which does its own smoothing) and re-plotted their performance, holding several test sets fixed (see Figure 5).



(a) Directed Accuracy on WSJ**10**

(b) Directed Accuracy on WSJ**20**

(c) Directed Accuracy on WSJ**30**

(d) Directed Accuracy on WSJ**40**

Figure 5: Directed accuracies attained by the DMV, trained at various gradations of WSJ but tested against four fixed evaluation sets — WSJ{10,20,30,40}, for four different initialization strategies. As in Figure 4, the green circle corresponds to Klein and Manning’s published score [1]; red, violet, green and blue curves represent supervised, Baby Steps, Ad-Hoc* and uniform initialization strategies. Dotted curves reflect starting performance, solid curves register performance at EM’s convergence, and the arrows connecting them emphasize the impact of learning.

The new graphs show that Baby Steps scales best with more (complex) data — its curves are the only ones that do not trend downwards. However, a good initialization induces a sweet spot at WSJ15, where the DMV is learned best using Ad-Hoc*. We call this mode “Less is More.” Curiously, even oracle training exhibits a bump here: once sentences get long enough (at WSJ36), its performance degrades below training with virtually no supervision (at the hardly representative WSJ3).

7 Experimental Result #3: Generalization

Our main findings carry over to the larger WSJ100, Section 23 of WSJ$^\infty$, as well as the independent Brown100 (see Table 1). Built up to WSJ45, Baby Steps performs best, scoring 39.4% (54.1%) on WSJ100, compared to Ad-Hoc*'s 31.3% (53.8%). Trained at the sweet spot, Ad-Hoc* is the undisputed champion, scoring 44.1% (58.8%), compared to Baby Steps' 39.2% (53.8%) if stopped there. Although Ad-Hoc* trained on WSJ15 generalizes well enough to reign on Brown100 as well, its score drops slightly, to 43.3% (59.2%). In contrast, Baby Steps trained up to WSJ15 actually scores higher on Brown100 than on WSJ100, though still lower than Ad-Hoc* — 42.3% (55.1%), suggesting that its iterative approach leads to better generalization, consistent with our expectations [22, 23].

Table 1: Directed and undirected accuracies on Section 23 of WSJ$^\infty$, WSJ100 and Brown100 for Ad-Hoc* and Baby Steps, trained at WSJ15 and WSJ45.

	@15		@45	
	*Ad-Hoc**	*Baby Steps*	*Ad-Hoc**	*Baby Steps*
Section 23	**44.1** (**58.8**)	39.2 (53.8)	31.5 (51.6)	39.4 (54.0)
WSJ100	**43.8** (**58.6**)	39.2 (53.8)	31.3 (51.5)	39.4 (54.1)
Brown100	**43.3** (**59.2**)	42.3 (55.1)	32.0 (52.4)	42.5 (55.5)

Results on Section 23 show, unexpectedly, that Baby Steps would have been state-of-the-art in 2008, whereas "Less is More" (Ad-Hoc* trained at WSJ15) already outperforms state-of-the-art in 2009 on longer sentences (see Table 2). Baby Steps is competitive with the log-normal families technique [5] of 2008, scoring slightly better on longer sentences against Viterbi decoding, though worse against MBR. "Less is More" outperforms the current best system by close to 2% on longer sentences.

Table 2: Directed accuracies on Section 23 of WSJ$\{10, 20, \infty\}$ for several baselines and recent state-of-the-art systems (adapted from [5], [6] and [7]).

			Year	*Decoding*	WSJ10	WSJ20	WSJ$^\infty$
	Attach-Right	[1]	2004	—	38.4	33.4	31.7
DMV	Ad-Hoc	[1]	2004	Viterbi	45.8	39.1	34.2
	Dirichlet	[5]	2008	Viterbi	45.9	39.4	34.9
	Ad-Hoc	[5]	2008	MBR	46.1	39.9	35.9
	Dirichlet	[5]	2008	MBR	46.1	40.6	36.9
	Log-Normal Families	[5]	2008	Viterbi	59.3	45.1	39.0
	Baby Steps @15		*2009*	*Viterbi*	*55.5*	*44.3*	*39.2*
	Baby Steps @45		*2009*	*Viterbi*	*55.1*	*44.4*	*39.4*
	Log-Normal Families	[5]	2008	MBR	59.4	45.9	40.5
	Shared Log-Normals, Tying Verbs and Nouns	[6]	2009	MBR	61.3	47.4	41.4
	Bilingual Shared Log-Normals, Tying Verbs and Nouns	[6]	2009	MBR	62.0	48.0	42.2
	Less is More (Ad-Hoc* @15)		*2009*	*Viterbi*	*56.2*	**48.2**	**44.1**
EVG	Smoothed (skip-val)	[7]	2009	Viterbi	62.1		
	Smoothed (skip-head)	[7]	2009	Viterbi	65.0		
	Smoothed (skip-head), Lexicalized	[7]	2009	Viterbi	**68.8**		

8 A Brief Historical Overview and Discussion of Related Work

Originating in behavioral psychology [24], the idea of "starting small" [25] stirred controversy [26] within cognitive science. Elman [25] claimed that artificial neural networks could succeed in learning a pseudo-natural language only under conditions of restricted memory or input, guided by a scaffolding for either model or data complexity. His networks failed to recognize a complex grammar when trained with the full "adult" language from the outset, but mastered it when the data were binned into grades of difficulty and presented in order of increasing complexity. We observed a similar effect with Ad-Hoc* and Baby Steps at WSJ45, but Rohde and Plaut's attempts to replicate Elman's exact study showed that limiting input in fact hinders language acquisition [26]. As they made Elman's grammar more English-like, by introducing and strengthening semantic constraints, the already significant advantage for "starting large" increased. Noting Rohde and Plaut's concern that Elman's simulations did not allow the networks exposed exclusively to complex inputs sufficient training time warranted by their initial random weights, we used a generous, low termination threshold for EM. Still, Baby Steps should be re-tested with a lexicalized model (such as the EVG), since its current POS tag-based approach is purely syntactic.

Elman reported equally good results with the learning mechanism itself undergoing “maturational changes” during training, holding the input constant (instead of gradually complicating the environment) — an observation consistent with the “less is more” proposal [27, 28]. Networks that started with severe memory limitations effectively restricted the range of data to which they were exposed in the early phases, imitating the increase in working memory and attention span that occurs over time in children [29]. Elman explained the paradoxical effect — that learning could be improved under conditions of limited capacity — by suggesting how restricted capabilities could neatly compensate for specific shortcomings of their learning mechanisms, making a long period of development play a positive role in the acquisition of a behavior. Baby Steps appeared to patiently improve local search by tweaking simplified training landscapes, repeatedly taking advantage of EM’s initial progress. Elman’s effect of early learning also seemed to filter the input, constraining the solution space by presenting the network with just the right data (simple sentences that permitted it to learn the basic representational categories) at just the right time (early on, when its plasticity was greatest).

Despite Rohde and Plaut’s failure to replicate Elman’s results with simple recurrent networks, many other machine learning techniques have been shown to benefit from scaffolded model complexity on a variety of language tasks. In word-alignment, Brown et al. [19] used IBM Models 1-4 as “stepping stones” to the training of Model 5 — a procedure that to this day serves as a corner-stone of statistical machine translation. Other prominent examples include “coarse-to-fine” approaches to parsing [30, 31, 32], translation [33, 32] and speech recognition [32], as well as a recent application to unsupervised POS tagging [34]. The first model is typically either trivial or particularly simple, so that both learning and inference are cheap. Each refinement on the way to the full model introduces only limited complexity, enabling incrementality. Brown et al.’s Model 1 had a global optimum that could be computed exactly, so that, as with Baby Steps, no parameters depended on initialization.

Examples of scaffolded data complexity are rare, although ideas for gradually making the learning task more difficult have been explored in robotics (typically in the context of navigation), in association with reinforcement learning [35, 36, 37, 38, 39, 40]. The year 2009 saw a renewed interest in shaping — a method of instruction in which the teacher decomposes a complete task into subcomponents, providing an easier path to learning [22, 23]. When Skinner [24] first coined the term, he described it as a “method of successive approximations.” Krueger and Dayan [22] showed that shaping speeds up language learning and leads to better generalization. Bengio et al. [23] confirmed this using simple multi-stage curriculum strategies, for language and vision tasks, and conjectured that a well-chosen sequence of training criteria, each associated with a different set of weights on the examples, could act as a continuation method [41], helping[3] to find better local optima of a non-convex training criterion. They also noted that at any point during learning, some examples could be considered “too easy” (not helping to improve the current model), while others “too difficult” (not captured by any small change to the model). Perhaps if Baby Steps focused on “interesting” examples — those near the frontier of its knowledge and abilities (neither too easy nor too hard), as Bengio et al. suggest, then it would not flat-line quite so early in its development (see Figure 5).

9 Conclusion

We have presented two ideas for unsupervised dependency parsing. “Less is More” is the paradoxical result that better performance can be attained by training with less data — even when removing samples from the true distribution. Taking advantage of the sweet spot at WSJ15, small tweaks to Klein and Manning’s approach of 2004 break through the 2009 state-of-the-art on longer sentences.

The second, Baby Steps, is a simple and elegant meta-heuristic for optimizing a non-convex training criterion. This idea eliminates the need for (and strongly outperforms) a linguistically-biased manually-tuned initialization when the location of the sweet spot is not known, scaling gracefully with more (complex) data. It should easily carry over to more powerful models and algorithms.

Future work could explore unifying these techniques. We see lots of opportunities for improvement, considering the poor performance of the oracle models relative to the supervised state-of-the-art, and in turn the poor performance of the unsupervised state-of-the-art relative to these oracle models.

[3] The basic idea of continuation methods is to first optimize a smoothed objective, then gradually consider less smoothing, with the intuition that smoothed versions of the problem reveal the global picture [23].

Acknowledgments

First author supported by the Fannie & John Hertz Foundation Fellowship. Our research, partially funded by the NSF via award IIS-0811974, began as the first author's final project in Prof. Christopher D. Manning's course on Natural Language Processing (CS.224N) and Prof. Andrew Y. Ng's class on Machine Learning (CS.229), at Stanford University. The authors would also like to thank Angel X. Chang and Christopher D. Manning for their comments on the early versions of this draft.

References

[1] D. Klein and C. D. Manning, "Corpus-based induction of syntactic structure: Models of dependency and constituency," in *Proc. of ACL*, 2004.

[2] N. A. Smith and J. Eisner, "Guiding unsupervised grammar induction using contrastive estimation," in *Proc. of the IJCAI Workshop on Grammatical Inference Applications*, 2005.

[3] R. Bod, "An all-subtrees approach to unsupervised parsing," in *Proc. of COLING-ACL*, 2006.

[4] Y. Seginer, "Fast unsupervised incremental parsing," in *Proc. of ACL*, 2007.

[5] S. B. Cohen, K. Gimpel, and N. A. Smith, "Logistic normal priors for unsupervised probabilistic grammar induction," in *NIPS*, 2008.

[6] S. B. Cohen and N. A. Smith, "Shared logistic normal distributions for soft parameter tying in unsupervised grammar induction," in *Proc. of NAACL-HLT*, 2009.

[7] W. P. Headden III, M. Johnson, and D. McClosky, "Improving unsupervised dependency parsing with richer contexts and smoothing," in *Proc. of NAACL-HLT*, 2009.

[8] A. Halevy, P. Norvig, and F. Pereira, "The unreasonable effectiveness of data," *IEEE Intelligent Systems*, vol. 24, no. 2, pp. 8–12, 2009.

[9] G. Druck, G. Mann, and A. McCallum, "Semi-supervised learning of dependency parsers using generalized expectation criteria," in *Proc. of ACL-IJCNLP*, 2009.

[10] J. Suzuki, H. Isozaki, X. Carreras, and M. Collins, "An empirical study of semi-supervised structured conditional models for dependency parsing," in *Proc. of EMNLP*, 2009.

[11] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[12] M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz, "Building a large annotated corpus of English: The Penn Treebank," *Computational Linguistics*, vol. 19, no. 2, pp. 313–330, 1993.

[13] M. Collins, *Head-Driven Statistical Models for Natural Language Parsing*. PhD thesis, University of Pennsylvania, 1999.

[14] W. N. Francis and H. Kucera, *Manual of Information to Accompany a Standard Corpus of Present-Day Edited American English, for use with Digital Computers*. Department of Linguistic, Brown University, 1964/1979.

[15] R. McDonald, F. Pereira, K. Ribarov, and J. Hajic, "Non-projective dependency parsing using spanning tree algorithms," in *Proc. of HLT-EMNLP*, 2005.

[16] J. K. Baker, "Trainable grammars for speech recognition," in *Speech Communication Papers for the 97th Meeting of the Acoustical Society of America*, pp. 547–550, 1979.

[17] D. Klein, *The Unsupervised Learning of Natural Language Structure*. PhD thesis, Stanford University, 2005.

[18] *Stanford JavaNLP Project*, 2009. `http://www-nlp.stanford.edu/javanlp/`.

[19] P. F. Brown, V. J. Della Pietra, S. A. Della Pietra, and R. L. Mercer, "The mathematics of statistical machine translation: Parameter estimation," *Computational Linguistics*, vol. 19, pp. 263–311, 1993.

[20] D. Elworthy, "Does Baum-Welch re-estimation help taggers?," in *Proc. of ANLP*, 1994.

[21] P. Liang and D. Klein, "Analyzing the errors of unsupervised learning," in *Proc. of HLT-ACL*, 2008.

[22] K. A. Krueger and P. Dayan, "Flexible shaping: How learning in small steps helps," *Cognition*, vol. 110, pp. 380–394, 2009.

[23] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *ICML*, 2009.

[24] B. F. Skinner, *The behavior of organisms: An experimental analysis*. Appleton-Century-Crofts, 1938.

[25] J. L. Elman, "Learning and development in neural networks: The importance of starting small," *Cognition*, vol. 48, pp. 781–799, 1993.

[26] D. L. T. Rohde and D. C. Plaut, "Language acquisition in the absence of explicit negative evidence: How important is starting small?," *Cognition*, vol. 72, no. 1, pp. 67–109, 1999.

[27] E. L. Newport, “Constraints on learning and their role in language acquisition: Studies of the acquisition of American Sign Language,” *Language Sciences*, vol. 10, no. 1, pp. 147–172, 1988.

[28] E. L. Newport, “Maturational constraints on language learning,” *Cognitive Science*, vol. 14, no. 1, pp. 11–28, 1990.

[29] R. Kail, *The development of memory in children*. W. H. Freeman and Company, 2nd ed., 1984.

[30] E. Charniak and M. Johnson, “Coarse-to-fine n-best parsing and MaxEnt discriminative reranking,” in *Proc. of ACL*, 2005.

[31] E. Charniak, M. Johnson, M. Elsner, J. Austerweil, D. Ellis, I. Haxton, C. Hill, R. Shrivaths, J. Moore, M. Pozar, and T. Vu, “Multilevel coarse-to-fine PCFG parsing,” in *HLT-NAACL*, 2006.

[32] S. O. Petrov, *Coarse-to-Fine Natural Language Processing*. PhD thesis, University of California, Berkeley, 2009.

[33] S. Petrov, A. Haghighi, and D. Klein, “Coarse-to-fine syntactic machine translation using language projections,” in *Proc. of EMNLP*, 2008.

[34] S. Ravi and K. Knight, “Minimized models for unsupervised part-of-speech tagging,” in *Proc. of ACL-IJCNLP*, 2009.

[35] S. P. Singh, “Transfer of learning by composing solutions of elemental squential tasks,” *Machine Learning*, vol. 8, pp. 323–339, 1992.

[36] T. D. Sanger, “Neural network learning control of robot manipulators using gradually increasing task difficulty,” *IEEE Trans. on Robotics and Automation*, vol. 10, 1994.

[37] L. M. Saksida, S. M. Raymond, and D. S. Touretzky, “Shaping robot behavior using principles from instrumental conditioning,” *Robotics and Autonomous Systems*, vol. 22, no. 3, pp. 231–249, 1997.

[38] M. Dorigo and M. Colombetti, *Robot Shaping: An Experiment in Behavior Engineering*. MIT Press/Bradford Books, 1998.

[39] T. Savage, “Shaping: The link between rats and robots,” *Connection Science*, vol. 10, no. 3, pp. 321–340, 1998.

[40] T. Savage, “Shaping: A multiple contingencies analysis and its relevance to behaviour-based robotics,” *Connection Science*, vol. 13, no. 3, pp. 199–234, 2001.

[41] E. L. Allgower and K. Georg, *Numerical Continuation Methods: An Introduction*. Springer-Verlag, 1990.