Unsupervised Dependency Parsing with Acoustic Cues

John K Pate[†‡]
j.k.pate@sms.ed.ac.uk

Sharon Goldwater[†]
sgwater@inf.ed.ac.uk

[†]ILCC, School of Informatics
University of Edinburgh
Edinburgh, EH8 9AB, UK

[‡]Department of Computing
Macquarie University
Sydney, NSW 2109, Australia

Abstract

Unsupervised parsing is a difficult task that infants readily perform. Progress has been made on this task using text-based models, but few computational approaches have considered how infants might benefit from acoustic cues. This paper explores the hypothesis that *word duration* can help with learning syntax. We describe how duration information can be incorporated into an unsupervised Bayesian dependency parser whose only other source of information is the words themselves (without punctuation or parts of speech). Our results, evaluated on both adult-directed and child-directed utterances, show that using word duration can improve parse quality relative to words-only baselines. These results support the idea that acoustic cues provide useful evidence about syntactic structure for language-learning infants, and motivate the use of word duration cues in NLP tasks with speech.

1 Introduction

Unsupervised learning of syntax is difficult for NLP systems, yet infants perform this task routinely. Previous work in NLP has focused on using the implicit syntactic information available in part-of-speech (POS) tags (Klein and Manning, 2004), punctuation (Seginer, 2007; Spitkovsky et al., 2011b; Ponvert et al., 2011), and syntactic similarities between related languages (Cohen and Smith, 2009; Cohen et al., 2011). However, these approaches likely use the data in a very different way from children: neither POS tags nor punctuation are observed during language acquisition (although see Spitkovsky et al. (2011a) and Christodoulopoulos et al. (2012) for encouraging results using unsupervised POS tags), and many children learn in a broadly monolingual environment. This paper explores a possible source of information that NLP systems typically ignore: word duration, or the length of time taken to pronounce each word.

There are good reasons to think that word duration might be useful for learning syntax. First, the well-established *Prosodic Bootstrapping* hypothesis (Gleitman and Wanner, 1982) proposes that infants use acoustic-prosodic cues (such as word duration) to help them identify syntactic structure, because prosodic and syntactic structures sometimes coincide. More recently, we proposed (Pate and Goldwater, 2011) that infants might use word duration as a direct cue to syntactic structure (i.e., without requiring intermediate prosodic structure), because words in high-probability syntactic structures tend to be pronounced more quickly (Gahl and Garnsey, 2004; Gahl et al., 2006; Tily et al., 2009).

Like most recent work on unsupervised parsing, we focus on learning syntactic dependencies. Our work is based on Headden et al. (2009)'s Bayesian version of the Dependency Model with Valence (DMV) (Klein and Manning, 2004), using interpolated backoff techniques to incorporate multiple information sources per token. However, whereas Headden et al. used words and POS tags as input, we use words and word duration information, presenting three variants of their model that use this information in slightly different ways.[1]

[1] By using neither gold-standard nor learned POS tags as input, our work differs from nearly all previous work on unsupervised dependency parsing. While learned tags might be plausible

Transactions of the Association for Computational Linguistics, 1 (2013) 63–74. Action Editor: Brian Roark.
Submitted 9/2012; Published 3/2013.

To our knowledge, this is the first work to incorporate acoustic cues into an unsupervised system for learning full syntactic parses. The methods in this paper were inspired by our previous approach (Pate and Goldwater, 2011), which showed that word duration measurements could improve the performance of an unsupervised lexicalized syntactic chunker over a words-only baseline. However, that work was limited to HMM-like sequence models, tested on adult-directed speech (ADS) only, and none of the models outperformed uniform-branching baselines. Here, we extend our results to full dependency parsing, and experiment on transcripts of both spontaneous ADS and child-directed speech (CDS). Our models using word duration outperform words-only baselines, along with the Common Cover Link parser of Seginer (2007), and the Unsupervised Partial Parser of Ponvert et al. (2011), unsupervised lexicalized parsers that have obtained state-of-the-art results on standard newswire treebanks (though their performance here is worse, as our input lacks punctuation). We also outperform uniform-branching baselines.

2 Syntax and Word Duration

Before presenting our models and experiments, we first discuss why word duration might be a useful cue to syntax. This section reviews the two possible reasons mentioned above: duration as a cue to prosodic structure, or as a cue to predictability.

2.1 Prosodic Bootstrapping

Prosody is the structure of speech as conveyed by rhythm and intonation, which are, in turn, conveyed by such measurable phenomena as variation in fundamental frequency, word duration, and spectral tilt. Prosodic structure is typically analyzed as imposing a shallow, hierarchical grouping structure on speech, with the ends of prosodic phrases (constituents) being cued in part by lengthening the last word of the phrase (Beckman and Pierrehumbert, 1986).

The *Prosodic Bootstrapping* hypothesis (Gleitman and Wanner, 1982) points out that prosodic phrases are often also syntactic phrases, and proposes that language-acquiring infants exploit this correlation. Specifically, if infants can learn about prosodic phrase structure using word duration (and fundamental frequency), they may be able to identify syntactic phrases more easily using word strings and prosodic trees than using word strings alone.

Several behavioral experiments support the connection between prosody and syntax and the prosodic bootstrapping hypothesis specifically. For example, there is evidence that adults use prosodic information for syntactic disambiguation (Millotte et al., 2007; Price et al., 1991) and to help in learning the syntax of an artificial language (Morgan et al., 1987), while infants can use acoustic-prosodic cues for utterance-internal clause segmentation (Seidl, 2007).

On the computational side, we are aware of only our previous HMM-based chunkers (Pate and Goldwater, 2011), which learned shallow syntax from words, words and word durations, or words and hand-annotated prosody. Using these chunkers, we found that using words plus prosodic annotation worked better than just words, and words plus word duration worked even better. While these results are consistent with the prosodic bootstrapping hypothesis, we suggested that predictability bootstrapping (see below) might be a more plausible explanation.

Other computational work has combined prosody with syntax, but only in supervised systems, and typically using hand-annotated prosodic information. For example, Huang and Harper (2010) used annotated prosodic breaks as a kind of punctuation in a supervised PCFG, while prosodic breaks learned in a semi-supervised way have been used as features for parse reranking (Kahn et al., 2005) or PCFG state-splitting (Dreyer and Shafran, 2007). In contrast to these methods, our approach observes neither parse trees nor prosodic annotations.

2.2 Predictability Bootstrapping

On the basis of our HMM chunkers, we introduced the *predictability bootstrapping* hypothesis (Pate and Goldwater, 2011): the idea that word durations could be a useful cue to syntactic structure not (or not only) because they provide information about prosodic structure, but because they are a direct cue to syntactic predictability. It is well-established that talkers tend to pronounce words more quickly when they are more predictable, as measured by, e.g., word frequency, n-gram probability, or whether or not the word has been previously mentioned (Aylett and Turk, 2004; Bell et al., 2009). However, syntactic proba-

in a model of language acquisition, gold tags certainly are not.

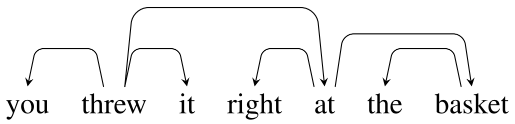

Figure 1: Example unlabeled dependency parse.

bility also seems to matter, with studies showing that verbs tend to be pronounced more quickly when they are in their preferred syntactic frame—transitive vs. intransitive or direct object vs. sentential complement (Gahl and Garnsey, 2004; Gahl et al., 2006; Tily et al., 2009). While this syntactic evidence is only for verbs, together with the evidence that effects of other notions of predictability, it suggests that such syntactic effects may also be widespread. If so, the duration of a word could give clues as to whether it is being used in a high-probability or low-probability structure, and thus what the correct structure is.

We found that our syntactic chunkers benefited more from duration information than prosodic annotations, providing some preliminary evidence in favor of predictability bootstrapping, but not ruling out prosodic bootstrapping. So, we are left with two plausible mechanisms by which word duration could help with learning syntax. Slow pronunciations may cue the end of a prosodic phrase, which is sometimes also the end of a syntactic phrase. Alternatively, slow pronunciations may indicate that the hidden syntactic structure is low probability, facilitating the induction of a probabilistic grammar. This paper will not seek to determine which mechanism is useful, instead taking the presence of two possible mechanisms as encouraging for the prospect of incorporating word duration into unsupervised parsing.

3 Models[2]

As mentioned, we will be incorporating word duration into unsupervised dependency parsing, producing analyses like the one in Figure 1. Each arc is between two words, with the *head* at the non-arrow end of the arc, and the *dependent* at the arrow end. One word, the *root*, depends on no word, and all other words depend on exactly one word. Following previous work on unsupervised dependency parsing, we will not label the arcs.

3.1 Dependency Model with Valence

All of our models are ultimately based on the Dependency Model with Valence (DMV) of Klein and Manning (2004), a generative, probabilistic model for projective (i.e. no crossing arcs), unlabeled dependency parses, such as the one in Figure 1.

The DMV generates dependency parses using three probability distributions, which together comprise model parameters θ. First, the root of the sentence is drawn from P_{root}. Second, we decide whether to stop generating dependents of the head h in direction $dir \in \{\text{left}, \text{right}\}$ with probability $P_{stop}(\cdot|h, dir, v)$, where v is T if h has a dir-ward dependent and F otherwise. If we decide to stop, then h takes no more dependents in the direction of dir. If we don't stop, we use the third probability distribution $P_{choose}(d|h, dir)$ to determine *which* dependent d to generate. The second and third step repeat for each generated word until all words have stopped generating in both directions.

The DMV was the first unsupervised parsing model to outperform a uniform-branching baseline on the Wall Street Journal corpus. It was trained using EM to obtain a maximum-likelihood estimate of the parameters θ, and learned from POS tags to avoid rare events. However, all work on syntactic predictability effects on word duration has been lexicalized (looking at, e.g., the transitivity bias of *particular* verbs). In addition, it is unlikely that children have access to the correct parts of speech when first learning syntactic structure. Thus, we want a DMV variant that learns from words rather than POS tags. We therefore adopt several extensions to the DMV due to Headden et al. (2009), described next.

3.2 The DMV with Backoff

Headden et al. (2009) sought to improve the DMV by incorporating lexical information in addition to POS tags. However, arcs between particular words are rare, so they modified the DMV in two ways to deal with this sparsity. First, they switched from MLE to a Bayesian approach, estimating a probability distribution over model parameters θ and dependency trees T given the training corpus C and a prior distribution $\boldsymbol{\alpha}$ over models: $P(T, \theta|C, \boldsymbol{\alpha})$.

Headden et al. avoided overestimating the probability of rare events that happen to occur in the train-

[2] The implementation of these models is available at http://github.com/jpate/predictabilityParsing

ing data by picking $\boldsymbol{\alpha}$ to assign low probability to models θ which give high probability to rare events. Accordingly, models that overcommit to rare events will contribute little to the final average over models. Specifically, Headden et al. use Dirichlet priors, with $\boldsymbol{\alpha}$ being the Dirichlet hyperparameters.

Headden et al.'s second innovation was to adapt interpolated backoff methods from language modeling with n-grams, where one can estimate the probability of word w_n given word w_{n-1} by interpolating between unigram and bigram probability estimates:

$$\hat{P}(w_n|w_{n-1}) = \lambda P(w_n|w_{n-1}) + (1-\lambda)P(w_n)$$

with $\lambda \in [0,1]$. Ideally, λ should be large when w_{n-1} is frequent, and small when w_{n-1} is rare. Headden et al. (2009) apply this method to the DMV by backing off from Choose and Stop distributions that condition on both head word and POS to distributions that condition on only the head POS.

In the equation above, λ is a scalar parameter. However, it actually specifies a probability distribution over the decision to back off ($\mathtt{B}$) or not back off ($\neg\mathtt{B}$), and we can use different notation to reflect this view. Specifically, $\lambda_{\mathit{stop}}(\cdot)$ and $\lambda_{\mathit{choose}}(\cdot)$ will represent our backoff distributions for the Stop and Choose decision, respectively. Using h_p and d_p to represent head and dependent POS tag and h_w and d_w to represent head and dependent word, one of the models Headden et al. explored estimates:

$$\begin{aligned} &\hat{P}_{\mathit{choose}}(d_p|h_w,h_p,\mathit{dir},\mathit{val}) = \\ &\quad \lambda_{\mathit{choose}}(\neg\mathtt{B}|h_w,h_p,\mathit{dir})P_{\mathit{choose}}(d_p|h_w,h_p,\mathit{dir}) \\ &\quad +\lambda_{\mathit{choose}}(\mathtt{B}|h_w,h_p,\mathit{dir})P_{\mathit{choose}}(d_p|h_p,\mathit{dir}) \end{aligned} \quad (1)$$

with an analogous backoff for P_{stop}. We can see from Equation 1 that $\hat{P}_{\mathit{choose}}$ backs off from a distribution that conditions on h_w to a distribution that marginalizes out h_w, and that the extent of backoff varies across h_w; we can use this to back off more when we have less evidence about h_w. This model only conditions on words; it does not generate them in the dependents. This means it is actually a conditional, rather than fully generative, model of observed POS tags and unobserved syntax conditioned on the observed words.

Since identifying the true posterior distribution $P(T,\theta|C,\boldsymbol{\alpha})$ is intractable, Headden et al. use Meanfield Variational Bayes (Kurihara and Sato, 2006; Johnson, 2007), which finds an approximation to the posterior using an iterative EM-like algorithm. In the E-step of VBEM, expected counts $E(r_i)$ are gathered for each latent variable using the Inside-Outside algorithm, exactly as in the E-step of traditional EM. The Maximization step differs from the M-Step of EM in two ways. First, the expected counts for each value of the latent variable r_i are incremented by the hyperparameter α_i. Second, the numerator and denominator are scaled by the function $\exp(\psi(\cdot))$, which reduces the probability of rare events. Specifically, the P_{choose} distribution is estimated using expectations for each arc $a_{d_p,h,\mathit{dir}}$ from head h to dependent POS tag d_p in direction dir, and the update equation for P_{choose} from iteration n to $n+1$ is:

$$\begin{aligned} &\hat{P}^{n+1}_{\mathit{choose}}(d_p|h,\mathit{dir}) = \\ &\qquad \frac{\exp(\psi(E^n(a_{d_p,h,\mathit{dir}}) + \alpha_{d_p,h,\mathit{dir}}))}{\exp(\psi(\sum_c (E^n(a_{c,h,\mathit{dir}}) + \alpha_{c,h,\mathit{dir}})))} \end{aligned} \quad (2)$$

where h is the head POS tag for the backoff distribution, and the head (word, POS) pair for the no backoff distribution. The update equation for P_{stop} is analogous.

Now consider the update equations for $\lambda_{\mathit{choose}}$:

$$\begin{aligned} &\hat{\lambda}^{n+1}_{\mathit{choose}}(\neg\mathtt{B}|h_w,h_p,\mathit{dir}) = \\ &\qquad \frac{\exp(\psi(\alpha_{\neg\mathtt{B}} + \sum_c (E^n(a_{c,h_w,h_p,\mathit{dir}}))))}{\exp(\psi(\alpha_{\mathtt{B}} + \alpha_{\neg\mathtt{B}} + \sum_c (E^n(a_{c,h_w,h_p,\mathit{dir}}))))} \\ &\hat{\lambda}^{n+1}_{\mathit{choose}}(\mathtt{B}|h_w,h_p,\mathit{dir}) = \\ &\qquad \frac{\exp(\psi(\alpha_{\mathtt{B}}))}{\exp(\psi(\alpha_{\mathtt{B}} + \alpha_{\neg\mathtt{B}} + \sum_c (E^n(a_{c,h_w,h_p,\mathit{dir}}))))} \end{aligned}$$

Only the $\neg\mathtt{B}$ numerator includes the expected counts, so as we see h_w in direction dir more often, the $\neg\mathtt{B}$ numerator will swamp the $\mathtt{B}$ numerator. By picking $\alpha_{\mathtt{B}}$ larger than $\alpha_{\neg\mathtt{B}}$, we can bias our λ distribution to prefer backing off until we expect at least $\alpha_{\mathtt{B}} - \alpha_{\neg\mathtt{B}}$ arcs out of h_w with tag h_p in the direction of dir.

To obtain good performance, Headden et al. replaced each word that appeared fewer than 100 times in the training data with the token "UNK." We will also use such an UNK cutoff.

3.3 DMV with Duration

We explore three models. One is a straightforward application of the DMV with Backoff to words and (quantized) word duration, and the other two are fully-generative variants. We also consider using words and POS tags as input to these models. Backoff models are given two streams of information, providing two of word identity, POS tag, or word duration for each observed token. We call one stream the "backoff" stream, and the other the "extra" stream. Backoff models learn a probability distribution conditioning on both streams, backing off to condition on only the backoff stream.

Our first words and duration model takes the duration as the extra stream and the word identity as the backoff stream, and, using h_a to represent the acoustic information for the head, defines:

$$\begin{aligned}\hat{P}_{choose}(d_w|h_w, h_a, dir) = \\ \lambda_{choose}(\neg\mathtt{B}|h_w, h_a, dir)P_{choose}(d_w|h_w, h_a, dir) \\ +\lambda_{choose}(\mathtt{B}|h_w, h_a, dir)P_{choose}(d_w|h_w, dir)\end{aligned} \quad (3)$$

with an analogous backoff scheme for P_{stop}. We will refer to this conditional model as "Cond." in our experiments. This equation is similar to Equation 1, except it uses words and duration instead of words and POS tags, and backs off *to*, not *away* from, words. We back off to the sparse words, rather than the less sparse duration, because duration provides almost no information about syntax in isolation.[3]

Directly modelling the extra stream among the dependents may allow us to capture selectional restrictions in POS and words models, or exploit effects of syntactic predictability on dependent duration. We therefore explore variants that generate both streams in the dependents. First, we examine a model ("Joint") that generates them jointly:

$$\begin{aligned}\hat{P}_{choose}(d_w, d_a|h_w, h_p, dir) = \\ \lambda_{choose}(\neg\mathtt{B}|h_w, h_a, dir) \\ P_{choose}(d_w, d_a|h_w, h_a, dir) \\ +\lambda_{choose}(\mathtt{B}|h_w, h_a, dir) \\ P_{choose}(d_w, d_a|h_w, dir)\end{aligned} \quad (4)$$

However, this joint model will have a very large state-space and may suffer from the same data sparsity, so we also explore a model ("Indep.") that generates the extra and backoff independently:

$$\begin{aligned}\hat{P}_{choose}(d_w, d_a|h_w, h_p, dir) = \\ \lambda_{choose}(\neg\mathtt{B}|h_w, h_a, dir) \\ P_{choose\_backoff}(d_w|h_w, h_a, dir) \\ P_{choose\_extra}(d_a|h_w, h_a, dir) \\ + \quad \lambda_{choose}(\mathtt{B}|h_w, h_a, dir) \\ P_{choose\_backoff}(d_w|h_w, dir) \\ P_{choose\_extra}(d_a|h_w, dir)\end{aligned} \quad (5)$$

We also modified the DMV with Backoff to handle heavily lexicalized models. In Headden et al. (2009), arcs between words that never appear in the same sentence are given probability mass only by virtue of the backoff distribution to POS tags, which all appear in the same sentence at least once. We want to avoid relying on POS tags, and we also want to use held-out development and test sets to avoid implicitly overfitting the data when exploring different model structures. To this end, we add one extra α_{UNK} hyperparameter to the Dirichlet prior of P_{choose} for each combination of conditioning events. This hyperparameter reserves probability mass for a head h to take a word d_w as a dependent if h and d_w never appeared together in the training data. The amount of probability mass reserved decreases as we see h_w more often. This is implemented in training by adding α_{UNK} to the denominator of the P_{choose} update equation for each h and dir. At test time, if a word d_w appears as an unseen dependent for head h, h takes d_w as a dependent with probability:

$$\hat{P}_{choose}(d_w|h, dir) = \quad (6)$$
$$\frac{\exp(\psi(\alpha_{\text{UNK}}))}{\exp(\psi(\alpha_{\text{UNK}} + \sum_c (E^{last}(r_{c,h,dir}) + \alpha_{c,h,dir})))}$$

Here, h may be a word, (word, POS) pair, or (word, duration) pair. Since this event by definition never occurs in the training data, α_{UNK} does not appear in the numerator during training. [4]

Finally, the conditional model ignores the extra stream in P_{root}, and the generative models estimate

[3] Preliminary dev-set experiments confirmed this intuition, as models that backed off to word duration performed poorly.

[4] Note also that α_{UNK} is different from a global UNK cutoff, which is imposed in preprocessing, and so effects every occurrence of an an UNK'd word in the model. α_{UNK} affects only dependents in P_{choose}, and treats a dependent as UNK iff it did not occur on that particular side of that particular head word in any sentence. We used both global UNK cutoffs (optimized on the dev set) and these α_{UNK} hyperparameters.

		Train	Dev	Test
wsj10	Word tokens	42,505	1,765	2,571
	Word types	7,804	818	1,134
	Sentences	6,007	233	357
swbdnxt10	Word tokens	24,998	2,980	3,052
	Word types	2,647	760	767
	Sentences	3,998	488	491
brent	Word tokens	20,954	2,127	2,206
	Word types	1,390	482	488
	Sentences	6,249	424	449

Table 1: Statistics for our three corpora.

P_{root} over both streams jointly and independently, respectively.

4 Experimental Setup

4.1 Datasets

We evaluate on three datasets: wsj10, sentences of length 10 or less from the Wall Street Journal portion of the Penn Treebank; swbdnxt10, sentences of length 10 or less from the Switchboard dataset of ADS used by Pate and Goldwater (2011); and brent, part of the Brent corpus of CDS (Brent and Siskind, 2001). Table 1 presents corpus statistics.

4.1.1 wsj10

We present a new evaluation of the DMV with Backoff on wsj10, which does not have any acoustic information, simply to verify that $\alpha_{\texttt{UNK}}$ performs sensibly on a standard corpus. Additionally, Headden et al. (2009) use an intensive initializer that relies on dozens of random restarts, and so, strictly speaking, only show that the backoff technology is useful for good initializations. Our new evaluation will show that the backoff technology provides a substantial benefit even for harmonic initialization.

wsj10 was created in the standard way; all punctuation and traces were removed, and sentences containing more than ten tokens were discarded. For our fully lexicalized version of wsj10, all words were lowercased, and numbers were replaced with the token "NUMBER."[5] Following standard practice, we used sections 2-21 for training, section 22 for development, and section 23 for test. wsj10 contains hand-annotated constituency parses, not dependency parses, so we used the standard "constituency-to-dependency" conversion tool of Johansson and Nugues (2007) to obtain high-quality CoNLL-style dependency parses.

4.1.2 swbdnxt10

Next, we evaluate on swbdnxt10, which contains all sentences up to length 10 from the same sections of the swbdnxt version of Switchboard used by Pate and Goldwater (2011). Short sentences are usually formulaic discourse responses (e.g. "oh ok"), so this dataset also excludes sentences shorter than three words. As our models successfully use word durations, this evaluation provides an important replication of the basic result from Pate and Goldwater (2011) with a different kind of syntactic model.

swbdnxt10 has a forced alignment of a dictionary-based phonetic transcription of each utterance to audio, providing our word duration information. As a very simple model of hyper-articulation and hypo-articulation, we classify a word as in the longest third duration, shortest third, or middle third. To minimize effects of word form, this classification was based on vowel count (counting a diphthong as one vowel): each word with n vowels is classified as in the shortest, longest, or middle tercile of duration among words with n vowels.

Like wsj10, swbdnxt10 is annotated only with constituency parses, so to provide approximate "gold-standard" dependencies, we used the same constituency-to-dependency conversion tool as for wsj10. We evaluated 200 randomly-selected sentences to check the accuracy of the conversion tool, which was designed for newspaper text. Excluding arcs involving words with no clear role in dependency structure (such as "um"), about 86% of the arcs were correct. While this rate is uncomfortably low, it is still much higher than unsupervised dependency parsers typically achieve, and so may provide a reasonable measure of *relative* dependency parse quality among competing systems.

4.1.3 brent

We also evaluated our models on the "Large Brent" dataset introduced in Rytting et al. (2010), a portion of the Brent corpus of child-directed speech (Brent and Siskind, 2001). We call this corpus brent. It consists of utterances from four of the mothers in Brent and Siskind's (2001) study, and, like

[5] Numbers were treated in this way only in wsj10.

swbdnxt10, has a forced alignment from which we obtain duration terciles. Rytting et al. (2010) used a 90%/10% train/test partition. We extracted every ninth utterance from the original training partition to create a dev set, producing an 80%/10%/10% partition. We also separated clitics from their base word. This dataset only has 186 sentences longer than ten words, with a maximum length of 22 words, so we discarded only sentences shorter than three words from the evaluation sets.

The Brent corpus is distributed via CHILDES (MacWhinney, 2000) with automatic dependency annotations. However, these are not hand-corrected, and rely on a different tokenization of the dataset than is present on the transcription tier. To produce a reliable gold-standard,[6] we annotated all sentences of length 2 or greater from the development and test sets with dependencies drawn from the Stanford Typed Dependency set (de Marneffe and Manning, 2008) using the annotation tool used for the Copenhagen Dependency Treebank (Kromann, 2003).

4.2 Parameters

In all experiments, hyperparameters for P_{root}, P_{stop}, and P_{choose} (and their backed-off distributions, and including α_{UNK}) were 1, α_{B} was 10, and $\alpha_{\neg\text{B}}$ was 1. VBEM was run on the training set until the data log-likelihood changed by less than 0.001%, and then the parameters were held fixed and used to obtain Viterbi parses for the evaluation sentences. Finally, we explored different global UNK cutoffs, replacing each word that appeared less than c times with the token UNK. We ran each model for each $c \in \{0, 1, 25, 50, 100\}$, and picked the best-scoring c on the development set for running on the test set and presentation here. We used a harmonic initializer similar to the one in Klein and Manning (2004).

4.3 Evaluation

In addition to evaluating the various incarnations of the DMV with backoff and input types, we compare to uniform branching baselines, the Common Cover Link (CCL) parser of Seginer (2007), and the Unsupervised Partial Parser (UPP) of Ponvert et al. (2011). The UPP produces a constituency parse from words and punctuation using a series of finite-state chunkers; we use the best-performing (Probabilistic Right Linear Grammar) version. The CCL parser produces a constituency parse using a novel "Cover Link" representation, scoring these links heuristically. Both CCL and UPP rely on punctuation (though according to Ponvert et al. (2011), UPP less so), which our input is missing. The left-headed "LH" (right-headed "RH") baseline assumes that each word takes the first word to its right (left) as a dependent, and corresponds to a uniform right-branching (left-branching) constituency baseline.

We evaluate the output of all models in terms of both constituency scores and dependency accuracy. Our wsj10 and swbdnxt10 corpora are originally annotated for constituency structure, with the dependency gold standard derived as described above, while our brent corpus is originally annotated for dependency structure, with the constituency gold standard derived by defining a constituent to span a head and each of its dependents (ignoring any one-word "constituents"). As the CCL and UPP parsers don't produce dependencies, only constituency scores are provided.

For constituency scores, we present the standard unlabeled Precision, Recall, and F-measure scores. For dependency scores, we present Directed attachment accuracy, Undirected attachment accuracy, and the "Neutral Edge Detection" (NED) score introduced by Schwartz et al. (2011). Directed attachment accuracy counts an arc as a true positive if it correctly identifies both a head and a dependent, whereas undirected attachment accuracy ignores arc direction in counting true positives. NED counts an arc as a true positive if it would be a true positive under the Undirected attachment score, or if the proposed head is the gold-standard grandparent of the proposed dependent. This avoids penalizing parses for flipping an arc, such as making determiners, rather than nouns, the head of noun phrases.

To assess statistical significance, we carried out stratified shuffling tests, with $10,000$ random shuffles, for all measures. Tables indicate significance differences between the backoff models and the most competitive baseline model on that measure, indicated by an italic score. A star $(^*)$ indicates $p < 0.05$, and a dagger $(^\dagger)$ indicates $p < 0.01$. To see the direction of a significant difference (i.e. whether the backoff model is better or worse than the baseline),

[6] Available at http://homepages.inf.ed.ac.uk/s0930006/brentDep/

		wsj10							swbdnxt10						
			Dependency			Constituency				Dependency			Constituency		
		UNK	Dir.	Undir.	NED	P	R	F	UNK	Dir.	Undir.	NED	P	R	F
EM	Wds	25	32.5	52.5	67.0	49.5	48.5	49.0	25	30.6	50.9	66.8	45.4	47.1	46.3
	POS	—	*46.4*	*63.8*	*78.1*	59.2	*58.1*	*58.6*	—	*53.0*	*65.0*	*76.8*	*52.5*	*52.9*	*52.7*
VB	Wds	25	29.4	52.4	70.5	51.3	52.6	52.0	25	36.1	54.9	72.7	49.0	50.0	49.5
	POS	—	43.5	61.9	77.3	*59.7*	57.1	58.4	—	51.3	62.5	74.3	47.1	46.6	46.8
Wds+POS	Cond.	50	49.9†	66.1†	79.6*	**64.2**†	**61.9**†	**63.0**†	100	45.5†	62.4†	77.8	58.4†	58.9†	58.7†
	Joint	50	46.0	63.7	79.0	62.0†	59.1	60.5*	1	49.4†	63.7	**79.6**†	60.0†	52.9	56.3†
	Indep.	25	**52.5**†	**68.0**†	**83.5**†	63.5†	61.5†	62.5†	100	**55.7**†	**65.8**	74.6†	**61.5**†	57.9†	59.6†
	LH	—	26.0	55.8	74.3	53.1	69.6	60.3	—	24.1	50.8	72.7	60.8	**82.5**	**70.0**
	RH	—	31.2	56.4	61.4	25.8	33.8	29.3	—	29.2	52.0	57.9	22.2	30.1	25.5
	CCL	—	—	—	—	50.8	40.7	45.2	—	—	—	—	53.6	47.4	50.3
	UPP	—	—	—	—	52.8	37.2	43.7	—	—	—	—	60.0	46.6	52.4

Table 2: Performance on wsj10 and swbdnxt10 for models using words and POS tags only. Bold scores indicate the best performance of all models and baselines on that measure.
† Significantly different from best non-uniform baseline (italics) by a stratified shuffling test, $p < 0.01$; *: $p < 0.05$.

look to the scores themselves.

5 Results

In all results, when a model sees only one kind of information, that is expressed by writing out the abbreviation for the relevant stream: "Wds" for words, "POS" for Part-Of-Speech, "Dur" for word duration. For baseline models that see two streams, the abbreviations are joined by a "×" symbol (as they treat input pairs as atoms drawn in the cross-product of the two streams' vocabulary). For the backoff models, the abbreviations are joined by a "+" symbol (as they combine the information sources with a weighted sum), with the "extra" stream name first.

5.1 Results: wsj10

The left half of Table 2 presents results on wsj10. For the baseline models, the first column with horizontal text indicates the *input*, while for the backoff (Wds+POS) models, the first column with horizontal text indicates whether and how the extra stream is modeled in dependents (as described in Section 3.3). The EM model with POS input is largely a replication of the original DMV, differing in the use of separate train, dev, and test sets, and possibly the details of the harmonic initializer. Our replication achieves an undirected attachment score of 63.8 on the test set, similar to the score of 64.5 reported by Klein and Manning (2004) when training and evaluating on all of wsj10. Cohen et al. (2008) use the same train/dev/test partition that we do, and report a directed attachment score of 45.8, similar to our directed attachment score of 46.4.

The VB model which learns from POS tags does not outperform the EM model which learns from POS tags, suggesting that data sparsity does not hurt the DMV when using POS tags. As expected, the words-only models perform much worse than both the POS input models and the uniform LH baseline. VB does improve the words-only constituency performance.

The Cond. and Indep. backoff models outperform the POS-only baseline on all measures, but the Joint backoff model does not demonstrate a clear advantage over the POS-only baseline on any measure. The success of the Indep. model indicates that modelling dependent word identity does provide enough information to justify the increase in sparsity. The failure of the Joint model to provide a further improvement indicates that the extra information in the full joint over dependents does not justify the large increase in parameters. We also see that several models outperform the LH baseline on dependencies, but the advantage is much less in F-Score, underscoring the loss of information in the conversion of dependencies to constituencies. Finally, all models outperform CCL and UPP on F-score, emphasizing their reliance on the punctuation we removed.

			Dependency			Constituency		
		UNK	Dir.	Undir.	NED	P	R	F
EM	Wds	25	30.6	50.9	66.8	45.4	47.1	46.3
	Wds×Dur	25	26.1	46.5	62.0	45.6	48.7	47.1
VB	Wds	25	*36.4*	*55.1*	*73.0*	49.1	50.0	49.6
	Wds×Dur	25	31.8	51.7	71.3	*49.2*	*55.9*	*52.3*
Dur+Wds	Cond.	25	32.6†	55.1	74.5†	59.1†	71.4†	64.7†
	Joint	50	31.8†	51.8†	70.8*	54.4†	60.5†	57.3†
	Indep.	50	**40.3**†	**59.1**†	**76.0**†	56.1†	61.7†	58.8†
	LH	—	24.1	50.8	72.7	**60.8**	**82.5**	**70.0**
	RH	—	29.2	52.0	57.9	22.2	30.1	25.5
	CCL	—	—	—	—	53.6	47.4	50.3
	UPP	—	—	—	—	60.0	46.6	52.4

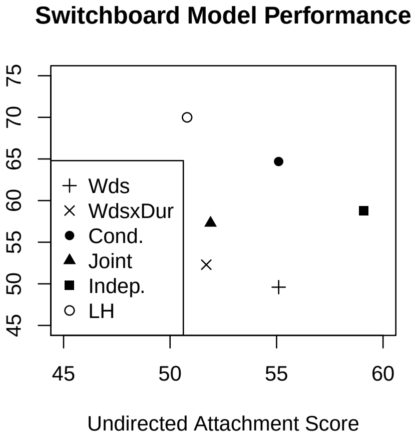

Table 3: Performance on swbdnxt10 for models using words and duration. The scatterplot includes a subset of the information in the table: F-score and undirected attachment accuracy for backoff models and VB and LH baseline. Bold, italics, and significance annotations as in Table 2.

5.2 Results: swbdnxt10

The right half of Table 2 presents performance figures on swbdnxt10 for input involving words and POS tags. As expected, the EM and VB baselines perform best when learning from gold-standard POS tags, and we again see no benefit for the VB POS-only model compared to the EM POS-only model. The POS-only baselines far outperform the uniform-attachment baselines on the dependency measures; to our knowledge this is the first demonstration outside the newspaper domain that the DMV outperforms a uniform branching strategy on these measures.

The other comparisons among systems listed in Table 2 are largely inconclusive. Models do comparatively well on *either* the constituency or dependency evaluation, but not both. The backoff models outperform the baseline POS-only models in the constituency evaluation, but underperform or match those same models in the dependency evaluation. Conversely, most models outperform the LH baseline in the dependency evaluation, but not in the constituency evaluation. There are probably two causes for the ambiguity in these results. First, the noise in the dependency gold-standard may have overwhelmed any advantage from backoff. Second, as we saw with wsj10, the conversion from dependencies to constituencies removes information, which may explain the failure of any model to outperform the LH baseline in the constituency evaluation.

Table 3 presents performance figures on swbdnxt10 for input involving words and duration, including a scatter-plot of Undirected attachment against constituency F-Score for the interesting comparisons. In the scatter-plot, models up and to the right performed better, and we see that the negative correlation between the dependency and constituency evaluations persists in words and duration input. VB substantially outperforms EM in the baselines, indicating that good smoothing is helpful when learning from words. Other comparisons are again ambiguous; the dependency evaluation is noisy, and backoff models outperform baseline models on the constituency evaluation but not the LH baseline. Still, the backoff models outperform all words-only baselines in constituency score, with two performing slightly worse in dependency score and one performing much better. So there is some evidence that word duration is useful, but we will find clearer evidence on the brent corpus.

5.3 Results: brent

Table 4 presents results on the brent dataset. VB is even more effective than in the other datasets for improving performance among baseline models, leading to double-digit improvements on some measures. Moreover, the best dev-set UNK cutoff drops to 1 for all VB models, indicating that, on this dataset, VB provides good smoothing even in models without backoff. This difference between datasets is likely related to differences in vocabulary diversity; the

			Dependency			Constituency		
		UNK	Dir.	Undir.	NED	P	R	F
EM	Wds	25	36.9	56.3	70.7	52.4	69.5	59.8
	Wds×Dur	25	31.3	51.1	66.9	50.7	64.7	56.9
VB	Wds	1	*51.2*	*64.2*	*77.3*	63.3	*68.1*	*66.0*
	Wds×Dur	1	47.0	60.5	74.0	***66.2***	64.9	65.5
Dur+Wds	Cond.	1	53.1*	65.5*	78.7*	65.4	68.6	**67.0***
	Joint	1	50.7	63.0	76.3	65.6	65.4†	65.5
	Indep.	1	**53.2**	**66.7**†	**79.6**†	61.5†	67.9	64.5
	LH	—	28.3	53.6	78.3	47.9	**85.6**	61.4
	RH	—	27.2	48.8	61.1	26.2	46.8	33.6
	CCL	—	—	—	—	41.7	58.8	48.8
	UPP	—	—	—	—	56.8	63.8	60.1

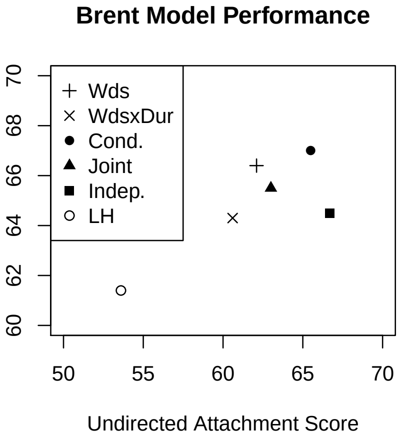

Table 4: Performance on brent for models using words and duration. The scatterplot includes a subset of the information in the table: F-score and undirected attachment accuracy for backoff models and VB and LH baseline. Bold, italics, and significance annotations as in Table 2.

type:token ratio in the brent training set is about 1:15, compared to 1:5 and 1:9 in the wsj10 and swbdnxt10 training sets, respectively.

More importantly for our main hypothesis, all three backoff models using words and duration outperform the words-only baselines (including CCL and UPP) on all dependency measures—the most accurate measures on this corpus, which has hand-annotated dependencies—and the Cond. model also wins on F-score.

6 Conclusion

In this paper, we showed how to use the DMV with Backoff and two fully-generative variants to explore the utility of word duration in fully lexicalized unsupervised dependency parsing. Although other researchers have incorporated features beyond words and POS tags into DMV-like models (e.g., semantics: Naseem and Barzilay (2011); morphology: Berg-Kirkpatrick et al. (2009)), we believe this is the first example based on Headden et al. (2009)'s backoff method. As far as we know, our work is also the first test of a DMV-based model on transcribed conversational speech and the first to outperform uniform-branching baselines without using either POS tags or punctuation in the input. Our results show that fully-lexicalized models can do well if they are smoothed properly and exploit multiple cues.

Our experiments also suggest that CDS is especially easy to learn from. Model performance on the brent dataset was generally higher than on swbdnxt10, with a much lower UNK threshold. This latter point, and the fact that brent has a much lower word type/token ratio than the other datasets, suggest that CDS provides more and clearer evidence about words' syntactic behavior.

Finally, our results provide more evidence, using a different, more powerful syntactic model than that of Pate and Goldwater (2011), that word duration is a useful cue for unsupervised parsing. We found that several ways of incorporating duration were useful, although the extra sparsity of Joint emissions was not justified in any of our investigations. Our results are consistent with both the prosodic and predictability bootstrapping hypotheses of language acquisition, providing the first computational support for these using a full syntactic parsing model and tested on child-directed speech. While our models do not provide a mechanistic account of how children might use duration information to help with learning syntax, they do show that this information is useful in principle, even without any knowledge of latent prosodic structure or its relationship to syntax. In addition, our results suggest it may be useful to explore using word duration to enrich NLP tasks in speech-related technologies, such as syntactically-inspired language models for text-to-speech generation. In the future, we also hope to investigate *why* duration is helpful, designing experiments to tease apart the role of prosody and predictability in learning syntax.

References

Matthew Aylett and Alice Turk. 2004. The smooth signal redundancy hypothesis: A functional explanation for relationships between redundancy, prosodic prominence, and duration in spontaneous speech. *Language and Speech*, 47(1):31–56.

Mary Beckman and Janet Pierrehumbert. 1986. Intonational structure in Japanese and English. *Phonology Yearbook*, 3:255–309.

Alan Bell, Jason M Brenier, Michelle Gregory, Cynthia Girand, and Dan Jurafsky. 2009. Predictability effects on durations of content and function words in conversational English. *Journal of Memory and Language*, 60:92–111.

Taylor Berg-Kirkpatrick, Alexandre Bouchard-Côté, John DeNero, and Dan Klein. 2009. Painless unsupervised learning with features. In *Proceedings of NAACL*.

Michael R Brent and Jeffrey M Siskind. 2001. The role of exposure to isolated words in early vocabulary development. *Cognition*, 81:31–44.

Christos Christodoulopoulos, Sharon Goldwater, and Mark Steedman. 2012. Turning the pipeline into a loop: Iterated unsupervised dependency parsing and PoS induction. In *Proceedings of the NAACL-HLT Workshop on the Induction of Linguistic Structure*, pages 96–99, Montréal, Canada, June. Association for Computational Linguistics.

Shay B Cohen and Noah A Smith. 2009. Shared logistic normal distributions for soft parameter tying in unsupervised grammar induction. In *Proceedings of NAACL*.

Shay B Cohen, Kevin Gimpel, and Noah A Smith. 2008. Logistic normal priors for unsupervised probabilistic grammar induction. In *Advances in Neural Information Processing Systems 22*.

Shay B Cohen, Dipanjan Das, and Noah A Smith. 2011. Unsupervised structure prediction with non-parallel multilingual guidance. In *Proceedings of EMNLP*.

Marie-Catherine de Marneffe and Christopher D Manning. 2008. Stanford typed dependencies manual. Technical report.

Markus Dreyer and Izhak Shafran. 2007. Exploiting prosody for PCFGs with latent annotations. In *Proceedings of Interspeech*, Antwerp, Belgium, August.

Susanne Gahl and Susan M Garnsey. 2004. Knowledge of grammar, knowledge of usage: Syntactic probabilities affect pronunciation variation. *Language*, 80:748–775.

Susanne Gahl, Susan M Garnsey, Cynthia Fisher, and Laura Matzen. 2006. “That sounds unlikely”: Syntactic probabilities affect pronunciation. In *Proceedings of the 27th meeting of the Cognitive Science Society*.

Lila Gleitman and Eric Wanner. 1982. Language acquisition: The state of the art. In Eric Wanner and Lila Gleitman, editors, *Language acquisition: The state of the art*, pages 3–48. Cambridge University Press, Cambridge, UK.

Will Headden, Mark Johnson, and David McClosky. 2009. Improved unsupervised dependency parsing with richer contexts and smoothing. In *Proceedings of NAACL-HLT*.

Zhongqiang Huang and Mary Harper. 2010. Appropriately handled prosodic breaks help PCFG parsing. In *Proceedings of NAACL-HLT*, pages 37–45, Los Angeles, California, June. Association for Computational Linguistics.

Richard Johansson and Pierre Nugues. 2007. Extended constituent-to-dependency conversion for English. In *Proceedings of NODALIDA 2007*.

Mark Johnson. 2007. Why doesn’t EM find good HMM POS-taggers. In *Proceedings of EMNLP-CoNLL*, pages 296–305.

Jeremy G Kahn, Matthew Lease, Eugene Charniak, Mark Johnson, and Mari Ostendorf. 2005. Effective use of prosody in parsing conversational speech. In *Proceedings of HLT-EMNLP*, pages 233–240.

Dan Klein and Christopher D. Manning. 2004. Corpus-based induction of syntactic structure: Models of dependency and constituency. In *Proceedings of ACL*, pages 479–486.

Matthias Trautner Kromann. 2003. The Danish Dependency Treebank and the DTAG treebank tool. In *Proceedings of the Second Workshop on Treebanks and Linguistic Theories*, pages 217–220.

Kenichi Kurihara and Taisuke Sato. 2006. Variational Bayesian grammar induction for natural language. In *Proceedings of the International Colloquium on Grammatical Inference*, pages 84–96.

Brian MacWhinney. 2000. *The CHILDES project: Tools for analyzing talk*. Lawrence Erlbaum Associates, Mahwah, NJ, third edition.

Séverine Millotte, Roger Wales, and Anne Christophe. 2007. Phrasal prosody disambiguates syntax. *Language and Cognitive Processes*, 22(6):898–909.

James L Morgan, Richard P Meier, and Elissa L Newport. 1987. Structural packaging in the input to language learning: contributions of prosodic and morphological marking of phrases to the acquisition of language. *Cognitive Psychology*, 19:498–550.

Tahira Naseem and Regina Barzilay. 2011. Using semantic cues to learn syntax. In *Proceedings of AAAI*.

John K Pate and Sharon Goldwater. 2011. Unsupervised syntactic chunking with acoustic cues: computational models for prosodic bootstrapping. In *Proceedings of the 2nd ACL workshop on Cognitive Modeling and Computational Linguistics*.

Elias Ponvert, Jason Baldridge, and Katrin Erk. 2011. Simple unsupervised grammar induction from raw text with cascaded finite state models. In *Proceedings of ACL-HLT*.

Patti J Price, Mari Ostendorf, Stefanie Shattuck-Hufnagel, and Cynthia Fong. 1991. The use of prosody in syntactic disambiguation. In *Proceedings of the HLT workshop on Speech and Natural Language*, pages 372–377, Morristown, NJ, USA. Association for Computational Linguistics.

C Anton Rytting, Chris Brew, and Eric Fosler-Lussier. 2010. Segmenting words from natural speech: subsegmental variation in segmental cues. *Journal of Child Language*, 37(3):513–543.

Roy Schwartz, Omri Abend, Roi Reichart, and Ari Rappoport1. 2011. Neutralizing linguistically problematic annotations in unsupervised dependency parsing evaluation. In *Proceedings of the 49th ACL*, pages 663–672.

Yoav Seginer. 2007. Fast unsupervised incremental parsing. In *Proceedings of ACL*.

Amanda Seidl. 2007. Infants' use and weighting of prosodic cues in clause segmentation. *Journal of Memory and Language*, 57(1):24–48.

Valentin I Spitkovsky, Hiyan Alshawi, Angel X Chang, and Daniel Jurafsky. 2011a. Unsupervised dependency parsing without gold part-of-speech tags. In *Proceedings of EMNLP*.

Valentin I Spitkovsky, Hiyan Alshawi, and Daniel Jurafsky. 2011b. Punctuation: Making a point in unsupervised dependency parsing. In *Proceedings of CoNLL*.

Harry Tily, Susanne Gahl, Inbal Arnon, Neal Snider, Anubha Kothari, and Joan Bresnan. 2009. Syntactic probabilities affect pronunciation variation in spontaneous speech. *Language and Cognition*, 1(2):147–165.