2015

Sign Constraints on Feature Weights Improve a Joint Model of Word Segmentation and Phnology

Joe Pater
Mark Johnson, *Macquarie University*
Robert Staubs
Emmanuel Dupoux, *Ecole des Hautes Etudes en Sciences Sociales*

Sign constraints on feature weights improve a joint model of word segmentation and phonology

Mark Johnson
Macquarie University
Sydney, Australia
Mark Johnson@MQ.edu.au

Joe Pater
University of Massachusetts, Amherst
Amherst, MA, USA
pater@linguist.umass.edu

Robert Staubs
University of Massachusetts, Amherst
Amherst, MA, USA
rstaubs@linguist.umass.edu

Emmanuel Dupoux
École des Hautes Etudes
en Sciences Sociales, ENS, CNRS,
Paris, France
emmanuel.dupoux@gmail.com

Abstract

This paper describes a joint model of word segmentation and phonological alternations, which takes unsegmented utterances as input and infers word segmentations and underlying phonological representations. The model is a Maximum Entropy or log-linear model, which can express a probabilistic version of Optimality Theory (OT; Prince and Smolensky (2004)), a standard phonological framework. The features in our model are inspired by OT's Markedness and Faithfulness constraints. Following the OT principle that such features indicate "violations", we require their weights to be non-positive. We apply our model to a modified version of the Buckeye corpus (Pitt et al., 2007) in which the only phonological alternations are deletions of word-final /d/ and /t/ segments. The model sets a new state-of-the-art for this corpus for word segmentation, identification of underlying forms, and identification of /d/ and /t/ deletions. We also show that the OT-inspired sign constraints on feature weights are crucial for accurate identification of deleted /d/s; without them our model posits approximately 10 times more deleted underlying /d/s than appear in the manually annotated data.

1 Introduction

This paper unifies two different strands of research on word segmentation and phonological rule induction. The *word segmentation task* is the task of segmenting utterances represented as sequences of phones into sequences of words. This is an idealisation of the lexicon induction problem, since the resulting words are phonological forms for lexical entries.

In its simplest form, the data for a word segmentation task is obtained by looking up the words of an orthographic transcript (of, say, child-directed speech) in a pronouncing dictionary and concatenating the results. However, this formulation significantly oversimplifies the problem because it assumes that each token of a word type is pronounced identically in the form specified by the pronouncing dictionary (usually its citation form). In reality there is usually a significant amount of pronunciation variation from token to token.

The Buckeye corpus, on which we base our experiments here, contains manually-annotated surface phonetic representations of each word as well as the corresponding underlying form (Pitt et al., 2007). For example, a token of the word "lived" has the underlying form /l.ih.v.d/ and could have the surface form [l.ah.v] (we follow standard phonological convention by writing underlying forms with slashes and surface forms with square brackets, and use the Buckeye transcription format).

There is a large body of work in the phonological literature on inferring phonological rules mapping underlying forms to their surface realisations. While most of this work assumes that the underlying forms are available to the inference procedure, there is work that induces underlying forms as well as the phonological processes that map them to sur-

Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL, pages 303–313,
Denver, Colorado, May 31 – June 5, 2015. ©2015 Association for Computational Linguistics

face forms (Eisenstat, 2009; Pater et al., 2012).

We present a model that takes a corpus of unsegmented surface representations of sentences and infers a word segmentation and underlying forms for each hypothesised word. We test this model on data derived from the Buckeye corpus where the only phonological variation consists of word-final /d/ and /t/ deletions, and show that it outperforms a state-of-the-art model that only handles word-final /t/ deletions.

Our model is a MaxEnt or log-linear model, which means that it is formally equivalent to a Harmonic Grammar, which is a continuous version of Optimality Theory (OT) (Smolensky and Legendre, 2005). We use features inspired by OT, and show that sign constraints on feature weights result in models that recover underlying /d/s significantly more accurately than models that don't include such contraints. We present results suggesting that these constraints simplify the search problem that the learner faces.

The rest of this paper is structured as follows. The next section describes related work, including previous work that this paper builds on. Section 3 describes our model, while section 4 explains how we prepared the data, presents our experimental results and investigates the effects of design choices on model performance. Section 5 concludes the paper and discusses possible future directions.

2 Background and related work

The *word segmentation task* is the task of segmenting utterances represented as sequences of phones into sequences of words. Elman (1990) introduced the word segmentation task as a simplified form of lexical acquisition, and Brent and Cartwright (1996) and Brent (1999) introduced the *unigram model of word segmentation*, which forms the basis of the model used here. Goldwater et al. (2009) described a non-parametric Bayesian model of word segmentation, and highlighted the importance of contextual dependencies. Johnson (2008) and Johnson and Goldwater (2009) showed that word segmentation accuracy improves when phonotactic constraints on word shapes are incorporated into the model. That model has been extended to also exploit stress cues (Börschinger and Johnson, 2014), the "topics" present in the non-linguistic context (Johnson et al., 2010) and the special properties of function words (Johnson et al., 2014).

Liang and Klein (2009) proposed a simple unigram model of word segmentation much like the original Brent unigram model, and introduced a "word length penalty" to avoid under-segmentation that we also use here. (As Liang et al note, without this the maximum likelihood solution is not to segment utterances at all, but to analyse each utterance as a single word). Berg-Kirkpatrick et al. (2010) extended this model by defining the unigram distribution with a MaxEnt model. The MaxEnt features can capture phonotactic generalisations about possible word shapes, and their model achieves a state-of-the-art word segmentation f-score.

The *phonological learning* task is to learn the phonological mapping from underlying forms to surface forms. Johnson (1984) and Johnson (1992) describe a search procedure for identifying underlying forms and the phonological rules that map them to surface forms given surface forms organised into inflectional paradigms. Goldwater and Johnson (2003) and Goldwater and Johnson (2004) showed how Harmonic Grammar phonological constraint weights (Smolensky and Legendre, 2005) can be learnt using a Maximum Entropy parameter estimation procedure given data consisting of underlying and surface word form pairs. There is now a significant body of work using Maximum Entropy techniques to learn phonological constraint weights (see esp. Hayes and Wilson (2008), as well as the review in Coetzee and Pater (2011)).

Recently there has been work attempting to integrate these two approaches. The word segmentation work generally ignores pronunciation variation by assuming that the input to the learner consists of sequences of citation forms of words, which is highly unrealistic. The phonology learning work has generally assumed that the learner has access to the underlying forms of words, which is also unrealistic.

In the word segmentation area, Elsner et al. (2012) and Elsner et al. (2013) generalise the Goldwater bigram model by assuming that the bigram model generates underlying forms, which a finite state transducer maps to surface forms. While this is an extremely general model, inference in such a model is very challenging, and they restrict attention to transducers where the underlying to surface mapping consists of simple substitutions, so their model cannot handle the deletion phenomena studied here. Börschinger et al. (2013) also generalise the Goldwater bigram model by including an underlying-to-surface mapping, but their mapping only allows word-final underlying /t/ to be deleted, which enables them to use a straight-forward generalisation of Goldwater's Gibbs sampling inference procedure.

In phonology, Eisenstat (2009) and Pater et al. (2012) showed how to generalise a MaxEnt model so it also learns underlying forms as well as MaxEnt phonological constraint weights given surface forms in paradigm format. The vast sociolinguistic literature on /t/-/d/-deletion is surveyed in Coetzee and Pater (2011), together with prior OT and MaxEnt analyses of the phenomena.

2.1 The Berg-Kirkpatrick et al. model

This section contains a more technical description of the Berg-Kirkpatrick et al. (2010) MaxEnt unigram model of word segmentation, which our model directly builds on. Our model integrates the MaxEnt unigram word segmentation model of Berg-Kirkpatrick et al. with the MaxEnt phonology models developed by Goldwater and Johnson (2003) and Goldwater and Johnson (2004). Because both kinds of models are MaxEnt models, this integration is fairly easy, and the inference procedure requires optimisation of a fairly straight-forward objective function. We use a customised version of the OWLQN-LBFGS procedure (Andrew and Gao, 2007) that allows us to impose sign constraints on individual feature weights.

As is standard in the word-segmentation literature, the model's input is a sequence of utterances $D = (w_1, \ldots, w_n)$, where each utterance $w_i = (w_{i,1}, \ldots, w_{i,m_i})$ is a sequence of (surface) phones. The Berg-Kirkpatrick et al model is a unigram model, so it defines a probability distribution over possible words s, where s is also a sequence of phones. The probability of an utterance w is the sum of the probability of all word sequences that generate it:

$$\mathrm{P}(w \mid \theta) \quad = \quad \sum_{\substack{s_1 \ldots s_\ell \\ \text{s.t.} s_1 \ldots s_\ell = w}} \prod_{j=1}^{\ell} \mathrm{P}(s_j \mid \theta)$$

Berg-Kirkpatrick et al's model of word probabilities $\mathrm{P}(s \mid \theta)$ is a MaxEnt model with parameters θ, where the features $f(s)$ of surface form s are chosen to encourage the model to generalise appropriately over word shapes. While they don't describe their features in complete detail, they include features for each word s, features for the prefix and suffix of s and features for the CV skeleton of the prefix and suffix of s.

In more detail, $\mathrm{P}(s \mid \theta)$ is a MaxEnt model as follows:

$$\begin{aligned} \mathrm{P}(s \mid \theta) &= \frac{1}{Z} \exp(\theta \cdot f(s)), \text{ where:} \\ Z &= \sum_{s' \in \mathcal{S}} \exp(\theta \cdot f(s')) \end{aligned}$$

The set of possible surface word forms $\mathcal{S}$ is the set of substrings (i.e., sequences of phones) occuring in the training data D that are shorter than a user-specified length bound. We follow Berg-Kirkpatrick in imposing a length bound on possible words; for the Brent corpus the maximum word length is 10 phones, while for the Buckeye corpus the maximum word length is 15 phones (reflecting the fact that words are longer in this adult-directed corpus).

While restricting the set of possible word forms $\mathcal{S}$ to the substrings appearing in D is reasonable for a simple multinomial model like the one in Liang and Klein (2009), it's interesting that this produces good results with a MaxEnt model like Berg-Kirkpatrick et al's, since one might expect such a model would have to learn generalisations about *impossible word shapes* in order to perform well. Because $\mathcal{S}$ only contains a small fraction of the possible phone strings, one might worry that the model would not see enough "impossible words" to learn to distinguish possible words from impossible ones, but the model's good performance suggests this is not the case.[1]

[1] The non-parametric Bayesian approach of Goldwater et al. (2009) and Johnson (2008) can be viewed as setting $\mathcal{S}$ to the set of all possible phone strings (i.e., a possible word can be any string of phones, whether or not it appears in D). The success of Berg-Kirkpatrick et al's approach suggests that these non-parametric methods might not be necessary here, i.e., the set of substrings actually occuring in D is "large enough" to enable the model to learn "implicit negative evidence" generalisations about impossible word shapes.

Berg-Kirkpatrick et al follow Liang et al in using maximum likelihood estimation to estimate their model's parameters (Berg-Kirkpatrick et al actually use L_2-regularised maximum likelihood estimates). As Liang et al note, it's easy to show that the maximum likelihood segmentation leaves each utterance unsegmented, i.e., each utterance is analysed as a single word. To avoid this, Berg-Kirkpatrick et al follow Liang et al by multiplying the word probabilities by a *word length penalty* term. Thus the likelihood L_D they actually maximise is as shown below:

$$\begin{aligned} L_D(\theta) &= \prod_{i=1}^{n} \mathrm{P}(w_i \mid \theta) \\ \mathrm{P}(w \mid \theta) &= \sum_{\substack{s_1 \ldots s_\ell \\ \text{s.t.} s_1 \ldots s_\ell = w}} \prod_{j=1}^{\ell} \mathrm{P}(s_j \mid \theta) \exp(-|s_i|^d) \end{aligned}$$

where d is a constant chosen to optimise segmentation performance. This means that the model is deficient, i.e., $\sum_{s \in \mathcal{S}} \mathrm{P}(s \mid \theta) < 1$. (Because our model uses a word length penalty in the same way, it too is deficient).

As Figure 1 shows, performance is very sensitive to the word length penalty parameter d: the best word segmentation on the Brent corpus is obtained when $d \approx 1.6$, while the best segmentation on the Buckeye corpus is obtained when $d \approx 1.5$. As far as we know there is no principled way to set d in an unsupervised fashion, so this sensitivity to d is perhaps the greatest weakness of this kind of model.

Even so, it's interesting that a unigram model without the kind of inter-word dependencies that Goldwater et al. (2009) argues for can do so well. It's possible that the improvement that Goldwater et al found with the bigram model is because modelling individual bigram dependencies splits the data in a way that reduces overlearning (Börschinger et al., 2012).

3 A MaxEnt unigram model of word segmentation and word-final /d/ and /t/ deletion

This section explains how we extend the Berg-Kirkpatrick et al. (2010) model to handle a set $\mathcal{P}$ of phonological processes, where a phonological process $p \in \mathcal{P}$ is a partial, non-deterministic function mapping underlying forms to surface forms. For example, word-final /t/ deletion is the function mapping underlying underlying forms ending in /t/ to surface forms lacking that final segment.

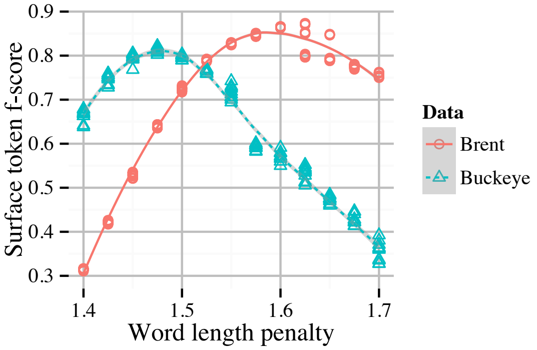

Figure 1: Sensitivity of surface token f-score to word length penalty factor d for the Brent and Buckeye corpora on data with no /d/ or /t/ deletions. Performance is sensitive to the value of the word length penalty d, and the optimal value of d depends on the corpus.

Our model is also a unigram model, but it defines a distribution over pairs (s, u) of surface/underlying form pairs, where s is a surface form and u is an underlying form. Below we allow this distribution to condition on phonological properties of the neighbouring surface forms.

The set $\mathcal{X}$ of possible (s, u) surface/underlying form pairs is defined as follows. For each surface form $s \in \mathcal{S}$ (the set of length-bounded phone substrings of the data D), $(s, s) \in \mathcal{X}$. In addition, if $u \in \mathcal{S}$ and some phonological alternation $p \in \mathcal{P}$ maps u to a surface form $s \in p(u) \in \mathcal{S}$, then $(s, u) \in \mathcal{X}$. That is, we require that potential underlying forms appear as surface substrings somewhere in the data D (which means this model cannot handle e.g., absolute neutralisation).

In the experiments below, we let $\mathcal{P}$ be phonological processes that delete word-final /d/ and /t/ phonemes. Given the Buckeye data, ([l.ih.v], /l.ih.v/), ([l.ih.v], /l.ih.v.d/) and ([l.ih.v], /l.ih.v.t/) are all members of $\mathcal{X}$ (i.e., candidate (s, u) pairs), corresponding to "live", "lived" and the non-word "livet" respectively, where the latter two surface forms are generated by final /d/ and /t/ deletion respectively.

Word-final /d/ and /t/ deletion depends on various aspects of the phonological context, such as whether the following word begins with a consonant or a vowel. Our model handles this dependency by learning a conditional model over surface/underlying form pairs $(s, u) \in \mathcal{X}$ that depends on the phonological context c:

$$\begin{aligned} \mathrm{P}(s, u \mid c, \theta) &= \frac{1}{Z_c} \exp(\theta \cdot f(s, u, c)), \text{ where:} \\ Z_c &= \sum_{(s,u) \in \mathcal{X}} \exp(\theta \cdot f(s, u, c)) \end{aligned}$$

In our experiments below, the set of possible contexts is $\mathcal{C} = \{\mathrm{C}, \mathrm{V}, \#\}$, encoding whether the following word begins with a consonant, a vowel or is the end of the utterance respectively. We leave for future research the exploration of other sorts of contextual conditioning. Note that the set $\mathcal{X}$ is the same for all contexts c; we show below that restricting attention to just those surface/underlying pairs appearing in the context c degrades the model's performance. In other words, the model benefits from the implicit negative evidence provided by underlying/surface pairs that do not occur in a given context.

We define the probability of a surface form $s \in \mathcal{S}$ in a context $c \in \mathcal{C}$ by marginalising out the underlying form:

$$\mathrm{P}(s \mid c, \theta) = \sum_{u:(s,u) \in \mathcal{X}} \mathrm{P}(s, u \mid c, \theta)$$

We optimise a penalised log likelihood $Q_D(\theta)$, with the word length penalty term d applied to the underlying form u.

$$\begin{aligned} Q(s \mid c, \theta) &= \sum_{u:(s,u) \in \mathcal{X}} \mathrm{P}(s, u \mid c, \theta) \exp(-|u|^d) \\ Q(w \mid \theta) &= \sum_{\substack{s_1 \ldots s_\ell \\ \text{s.t.} s_1 \ldots s_\ell = w}} \prod_{j=1}^{\ell} Q(s_j \mid c, \theta) \\ Q_D(\theta) &= \sum_{i=1}^{n} \log Q(w_i \mid \theta) - \lambda \, \|\theta\|_1 \end{aligned}$$

We are somewhat cavalier about the conditional contexts c here: in our model below the context c for a word is determined by the following word, so one can view our model as a generative model that generates the words in an utterance from right to left.

Because our model is a MaxEnt model, we have considerable freedom in the choice of features, and as Berg-Kirkpatrick et al. (2010) emphasise, the choice of features directly determines the kinds of generalisations the model can learn. The features $f(s, u, c)$ of a surface form s, underlying form u and context c we use here are inspired by OT. We describe our features using an example where $s =$ [l.ih.v], $u =$ /l.ih.v.t/ and $c =$ C (i.e., the word is followed by a consonant).

Underlying form lexical features: A feature for each underlying form u. In our example, the feature is `<U l ih v t>`. These features enable the model to learn language-specific lexical entries. There are 4,803,734 underlying form lexical features (one for each possible substring in the training data).

Surface markedness features: The length of the surface string (`<#L 3>`), the number of vowels (`<#V 1>`) (this is a rough indication of the number of syllables), the surface suffix (`<Suffix v>`), the surface prefix and suffix CV shape (`<CVPrefix CV>` and `<CVSuffix VC>`), and suffix+context CV shape (`<CVContext _C>` and `<CVContext C _C>`). There are 108 surface markedness features.

Faithfulness features: A feature for each divergence between underlying and surface forms (in this case, `<*F t>`). There are two faithfulness features.

We used L_1 regularisation here, rather than the L_2 regularisation used by Berg-Kirkpatrick et al. (2010), in the hope that its sparsity-inducing "feature selection" capabilities would enable it to "learn" lexical entries for the language, as well as precisely which markedness features are required to account for the data. However, we found that the choice of L_1 versus L_2 regression makes little difference, and the model is insensitive to the value of the regulariser constant λ (we set to $\lambda = 1$ in the experiments below).

We developed a specially modified version of the LBFGS-OWLQN optimisation procedure for optimising L_1-regularised loss functions (Andrew and

Gao, 2007) that allows us to constrain certain feature weights θ_k to have a particular sign. This is a natural extension of the LBFGS-OWLQN procedure since it performs orthant-constrained line searches in any case. We describe experiments below where we require the feature weights for the markedness and faithfulness features to be non-positive, and where the underlying lexical form features are required to be non-negative. The requirement that the lexical form features are positive, combined with the sparsity induced by the L_1 regulariser, was intended to force the model to learn an explicit lexicon encoded by the underlying form features with positive weights (although our results below suggest that it did not in fact do this).

The inspiration for the requirement that markedness and faithfulness features are non-positive comes from OT, which claims that the presence of such features can only reduce the "harmony", i.e., the well-formedness, of an (s, u) pair. Versions of Harmonic Grammar that aim to produce OT-like behavior with weighted constraints often bound weights at zero (see e.g. Pater (2009)). The results below are the first to show that these constraints matter for word segmentation.

4 Experimental results

This section describes the experiments we performed to evaluate the model just described. We first describe how we prepared the data on which the model is trained and evaluated, and then we describe the performance of that model. Finally we perform an analysis of how the model's performance varies as parameters of the model are changed.

We ran this model on data extracted from the Buckeye corpus of conversational speech (Pitt et al., 2007) which was modified so the only alternations it contained are final /d/ and /t/ deletions. The Buckeye corpus gives a surface realisation and an underlying form for each word token, and following Börschinger et al. (2013), we prepared the data as follows. We used the Buckeye underlying forms as our underlying forms. Our surface forms were also identical to the Buckeye underlying forms, except when the underlying form ends in either a /d/ or a /t/. In this case, if the Buckeye surface form does not end in an allophonic variant of that segment, then our surface form consists of the Buckeye underlying form with that final segment deleted. Thus the only phonological variation in our data are deletions of word-final /d/ and /t/ appearing in the Buckeye corpus, otherwise our surface forms are identical to Buckeye underlying forms.

For example, consider a token whose Buckeye underlying form is /l.ih.v.d/ "lived". If the Buckeye surface form is [l.ah.v] then our surface form would be [l.ih.v], while if the Buckeye surface form is [l.ah.v.d] then our surface form would be [l.ih.v.d].

We now present some descriptive statistics on our data. The data contains 48,796 sentences and 890,597 segments. The longest sentence has 187 segments. The "gold" data has the following properties. There are 236,996 word boundaries, 285,792 word tokens, and 9,353 underlying word types. The longest word has 17 segments. Of the 41,186 /d/s and 73,392 /t/s in the underlying forms, 24,524 /d/s and 40,720 /t/s are word final, and of these 13,457 /d/s and 11,727 /t/s are deleted (i.e., do not appear on the surface).

Our model considers all possible substrings of length 15 or less as a possible surface form of a word, yielding 4,803,734 possible word types and 5,292,040 possible surface/underlying word type pairs. Taking the 3 contexts derived from the following word into account, there are 4,969,718 possible word+context types. When all possible surface/underlying pairs are considered in all possible contexts there are 15,876,120 possible surface/underlying/context triples.

Table 1 summarises the major experimental results for this model, and compares them to the results of Börschinger et al. (2013). Note that their model only recovers word-final /t/ deletions and was run on data without word-final /d/ deletions, so it is solving a simpler problem than the one studied here. Even so, our model achieves higher overall accuracies.

We also conducted experiments on several of the design choices in our model. Figure 2 shows the effect of the sign constraints on feature weights discussed above. This plot shows that the contraints on the weights of markedness and faithfulness features seems essential for good word segmentation performance. Interestingly, we found that the weight constraints make very little difference if the data does

	Börschinger et al. 2013	Our model
Surface token f-score	0.72	**0.76** (0.01)
Underlying type f-score	—	0.37 (0.02)
Deleted /t/ f-score	0.56	**0.58** (0.03)
Deleted /d/ f-score	—	0.62 (0.19)

Table 1: Results summary for our model compared to that of the Börschinger et al. (2013) model. Surface token f-score is the standard token f-score, while underlying type or "lexicon" f-score measures the accuracy with which the underlying word types are recovered. Deleted /t/ and /d/ f-scores measure the accuracy with which the model recovers segments that don't appear in the surface. These results are averaged over 40 runs (standard deviations in parentheses) with the word length penalty $d = 1.525$ applied to underlying forms; standard deviations are given in parentheses.

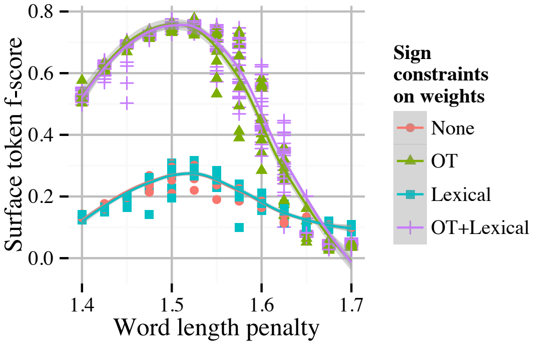

Figure 2: The effect of constraints on feature weights on surface token f-score. "OT" indicates that the markedness and faithfulness features are required to be non-positive, while "Lexical" indicates that the underlying lexical features are required to be non-negative.

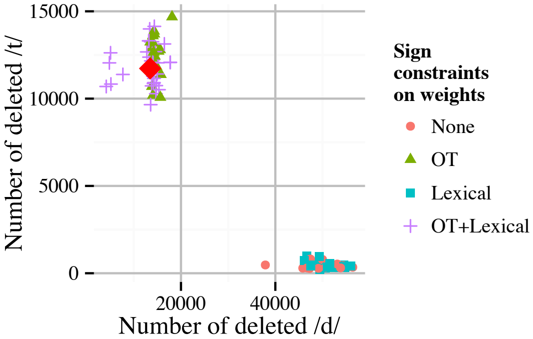

Figure 3: The effect of constraints feature weights on the number of deleted underlying /d/ and /t/ segments posited by the model (d = 1.525). The red diamond indicates the 13,457 deleted underlying /d/ and 11,727 deleted underlying /t/ in the "gold" data.

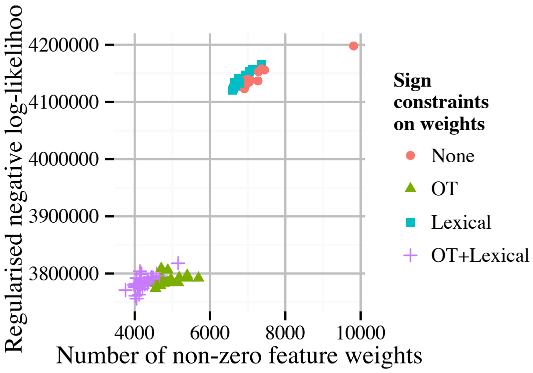

Figure 4: The regularised log-likelihood as a function of the number of non-zero weights for different constraints on feature weights ($d = 1.525$).

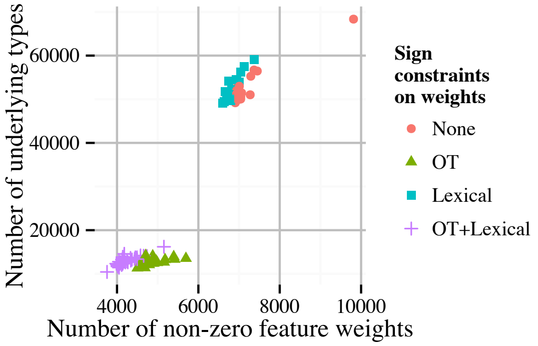

Figure 5: The number of underlying types proposed by the model as a function of the number of non-zero weights, for different constraints on feature weights ($d = 1.525$). There are 9,353 underlying types in the "gold" data.

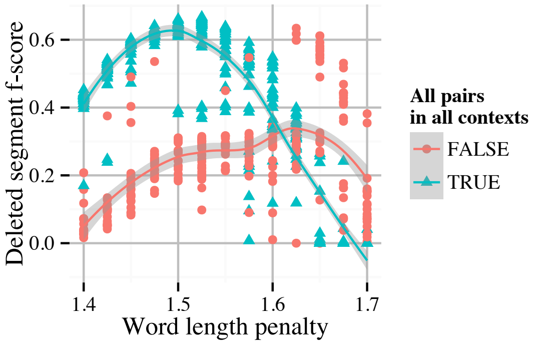

Figure 6: F-score for deleted /d/ and /t/ recovery as a function of word length penalty d and whether all surface/underlying pairs $\mathcal{X}$ are included in all contexts $\mathcal{C}$ ($d = 1.525$).

not any /t/ or /d/ deletions (i.e., the case that Berg-Kirkpatrick et al. (2010) studied).

Investigating this further, we found that the weight constraints on the markedness and faithfulness features has a dramatic effect on the recovery of underlying segments, particularly underlying /d/s. Figure 3 shows that with these constraints the model recovers approximately the correct number of deleted underlying segments, while without this constraint the model posits far too many underlying /d/s. Figure 4 shows that these constraints help the model find higher regularised likelihood sets of feature weights with fewer non-zero feature weights.

We examined how the number of non-zero feature weights (most of which are for underlying type features) relate to the number of underlying types posited by the model. Figure 5 shows that the weight constraints on markedness and faithfulness constraints have great impact on the number of non-zero feature weights and on the number of underlying forms the model posits. In all cases, the model recovers far more underlying forms than it finds non-zero weights.

The lexicon weight constraints have much less impact than the OT weight constraints. As Figure 3 shows, without the OT weight constraints the models posit too many deleted /d/ and essentially no deleted /t/. Figure 4 shows that OT weight constraints enable the model to find higher likelihood solutions, i.e., the OT weight constraints help search. Inspired by a reviewer's comments, we studied type-token ratios and the number of boundaries our models posit. We found that the models without OT weight constraints posit far too few word boundaries compared to the gold data, so the number of surface tokens is too low, so the words are too long, and the number of underlying types is too high. This is consistent with Figures 4–5.

We also examined whether it is necessary to consider all surface/underlying pairs $\mathcal{X}$ in each context $\mathcal{C}$, or whether it is possible to restrict attention to the much smaller sets $\mathcal{X}_c$ that occur in each $c \in \mathcal{C}$ (this dramatically reduces the amount of memory required and speeds the computation). Figure 6 shows that working with the smaller, context-specific sets dramatically decreases the model's ability to recover deleted segments.

5 Conclusions and future work

The MaxEnt unigram model of word segmentation developed by Berg-Kirkpatrick et al. (2010) integrates straight-forwardly with the MaxEnt phonology models of Goldwater and Johnson (2003) to produce a MaxEnt model that jointly models word segmentation and the mapping from underlying to surface forms.

We tested our model on data derived from the manually-annotated Buckeye corpus of conversational speech (Pitt et al., 2007) in which the only phonological alternations are deletions of word-final /d/ and /t/ segments. We demonstrated that our model improves on the state-of-the-art for word segmentation, recovery of underlying forms and recovery of deleted segments for this corpus.

Our model is a MaxEnt or log-linear unigram model over the set of possible surface/underlying form pairs. Inspired by the work of Berg-Kirkpatrick et al. (2010), the set of surface/underlying form pairs our model calculates the partition function over is restricted to those actually appearing in the training data, and doesn't include all logically possible pairs. We found that even with this restriction, the model produces good results.

Because our model is a Maximum Entropy or log-linear model, it is formally an instance of a Harmonic Grammar (Smolensky and Legendre, 2005), so we investigated features inspired by OT, which is a discretised version of Harmonic Grammar that has been extensively developed in the linguistics literature. The features our model uses consist of underlying form features (one for each possible underlying form), together with markedness and faithfulness phonological features inspired by OT phonological analyses. According to OT, these markedness and faithfulness features should always have negative weights (i.e., when such a feature "fires", it should always make the analysis less probable). We found that constraining feature weights in this way dramatically improves the model's accuracy, apparently helping to find higher likelihood solutions.

Looking forwards, a major drawback of the MaxEnt approaches to word segmentation are their sensitivity to the *word length penalty* parameter, which this model shares with the models of Berg-Kirkpatrick et al. (2010) and (Liang and Klein, 2009) on which it is based. It would be very desirable to have a principled way to set this parameter in an unsupervised manner.

Because our goal was to explore the MaxEnt approach to joint segmenation and alternation, we deliberately used a minimal feature set here. As the reviewers pointed out, we did not include any morphological features, which could have a major impact on the model. Investigating the impact of richer feature sets, including a combination of phonotactic and morphological features, would be an excellent topic for future work.

It would be interesting to extend this approach to a wider range of phonological processes in addition to the word-final /t/ and /d/ deletion studied here. Because this model enumerates the possible surface/underlying/context triples before beginning to search for potential surface and underlying words, its memory requirements would grow dramatically if the set of possible surface/underlying alternations were increased. (The fact that we only considered word final /d/ and /t/ deletions means that there are only three possible underlying word forms for each surface word forms). Perhaps there is a way of identifying potential underlying forms that avoids enumerating them. For example, it might be possible to sample possible underlying word forms during the learning process rather than enumerating them ahead of time, perhaps by adapting non-parametric Bayesian approaches (Goldwater et al., 2009; Johnson and Goldwater, 2009; Börschinger et al., 2013).

Acknowledgments

This research was supported under the Australian Research Council's *Discovery Projects* funding scheme (project numbers DP110102506 and DP110102593), by the Mairie de Paris, the fondation Pierre Gilles de Gennes, the École des Hautes Etudes en Sciences Sociales, the École Normale Supérieure, the Region Ile de France, by the US National Science Foundation under Grant No. S121000000211 to the third author and Grant BCS-424077 to the University of Massachusetts, and by grants from the European Research Council (ERC-2011-AdG-295810 BOOTPHON) and the Agence Nationale pour la Recherche (ANR-10-LABX-0087 IEC, ANR-10-IDEX-0001-02 PSL*). We'd also like to thank the three anonymous reviewers for helpful comments and suggestions.

References

Galen Andrew and Jianfeng Gao. 2007. Scalable training of l1-regularized log-linear models. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pages 33–40, New York, New York. ACM.

Taylor Berg-Kirkpatrick, Alexandre Bouchard-Côté, John DeNero, and Dan Klein. 2010. Painless unsupervised learning with features. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 582–590. Association for Computational Linguistics.

Benjamin Börschinger and Mark Johnson. 2014. Exploring the role of stress in Bayesian word segmentation using adaptor grammars. *Transactions of the Association for Computational Linguistics*, 2(1):93–104.

Benjamin Börschinger, Katherine Demuth, and Mark Johnson. 2012. Studying the effect of input size for Bayesian word segmentation on the Providence corpus. In *Proceedings of the 24th International Conference on Computational Linguistics (Coling 2012)*, pages 325–340, Mumbai, India. Coling 2012 Organizing Committee.

Benjamin Börschinger, Mark Johnson, and Katherine Demuth. 2013. A joint model of word segmentation and phonological variation for English word-final /t/-deletion. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1508–1516, Sofia, Bulgaria. Association for Computational Linguistics.

M. Brent and T. Cartwright. 1996. Distributional regularity and phonotactic constraints are useful for segmentation. *Cognition*, 61:93–125.

M. Brent. 1999. An efficient, probabilistically sound algorithm for segmentation and word discovery. *Machine Learning*, 34:71–105.

Andries Coetzee and Joe Pater. 2011. The place of variation in phonological theory. In John Goldsmith, Jason Riggle, and Alan Yu, editors, *The Handbook of Phonological Theory*, pages 401–431. Blackwell, 2nd edition.

Sarah Eisenstat. 2009. Learning underlying forms with MaxEnt. Master's thesis, Brown University.

Jeffrey Elman. 1990. Finding structure in time. *Cognitive Science*, 14:197–211.

Micha Elsner, Sharon Goldwater, and Jacob Eisenstein. 2012. Bootstrapping a unified model of lexical and phonetic acquisition. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 184–193, Jeju Island, Korea. Association for Computational Linguistics.

Micha Elsner, Sharon Goldwater, Naomi Feldman, and Frank Wood. 2013. A joint learning model of word segmentation, lexical acquisition, and phonetic variability. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 42–54, Seattle, Washington, USA, October. Association for Computational Linguistics.

Sharon Goldwater and Mark Johnson. 2003. Learning OT constraint rankings using a Maximum Entropy model. In J. Spenader, A. Eriksson, and Osten Dahl, editors, *Proceedings of the Stockholm Workshop on Variation within Optimality Theory*, pages 111–120, Stockholm. Stockholm University.

Sharon Goldwater and Mark Johnson. 2004. Priors in Bayesian learning of phonological rules. In *Proceedings of the Seventh Meeting Meeting of the ACL Special Interest Group on Computational Phonology: SIGPHON 2004*.

Sharon Goldwater, Thomas L. Griffiths, and Mark Johnson. 2009. A Bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, 112(1):21–54.

Bruce Hayes and Colin Wilson. 2008. A Maximum Entropy model of phonotactics and phonotactic learning. *Linguistic Inquiry*, 39(3):379–440.

Mark Johnson and Sharon Goldwater. 2009. Improving nonparameteric Bayesian inference: experiments on unsupervised word segmentation with adaptor grammars. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 317–325, Boulder, Colorado, June. Association for Computational Linguistics.

Mark Johnson, Katherine Demuth, Michael Frank, and Bevan Jones. 2010. Synergies in learning words and their referents. In J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 1018–1026.

Mark Johnson, Anne Christophe, Emmanuel Dupoux, and Katherine Demuth. 2014. Modelling function words improves unsupervised word segmentation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 282–292. Association for Computational Linguistics, June.

Mark Johnson. 1984. A discovery procedure for certain phonological rules. In *10th International Conference on Computational Linguistics and 22nd Annual Meeting of the Association for Computational Linguistics*.

Mark Johnson. 1992. Identifying a rule's context from data. In *The Proceedings of the 11th West Coast Conference on Formal Linguistics*, pages 289–297, Stanford, CA. Stanford Linguistics Association.

Mark Johnson. 2008. Using Adaptor Grammars to identify synergies in the unsupervised acquisition of linguistic structure. In *Proceedings of the 46th Annual Meeting of the Association of Computational Linguistics*, pages 398–406, Columbus, Ohio. Association for Computational Linguistics.

Percy Liang and Dan Klein. 2009. Online EM for unsupervised models. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 611–619, Boulder, Colorado, June. Association for Computational Linguistics.

Joe Pater, Robert Staubs, Karen Jesney, and Brian Smith. 2012. Learning probabilities over underlying representations. In *Proceedings of the Twelfth Meeting of the ACL-SIGMORPHON: Computational Research in Phonetics, Phonology, and Morphology*, pages 62–71.

Joe Pater. 2009. Weighted constraints in generative linguistics. *Cognitive Science*, 33:999–1035.

Mark A. Pitt, Laura Dilley, Keith Johnson, Scott Kiesling, William Raymond, Elizabeth Hume, and Eric Fosler-Lussier. 2007. Buckeye corpus of conversational speech.

Alan Prince and Paul Smolensky. 2004. *Optimality Theory: Constraint Interaction in Generative Grammar*. Blackwell.

Paul Smolensky and Géraldine Legendre. 2005. *The Harmonic Mind: From Neural Computation To Optimality-Theoretic Grammar*. The MIT Press.