Representational Bias in Unsupervised Learning of Syllable Structure

Representational Bias in Unsupervised Learning of Syllable Structure

Sharon Goldwater and **Mark Johnson**
Department of Cognitive and Linguistic Sciences
Brown University
Providence, RI 02912
{Sharon_Goldwater,Mark_Johnson}@brown.edu

Abstract

Unsupervised learning algorithms based on Expectation Maximization (EM) are often straightforward to implement and provably converge on a local likelihood maximum. However, these algorithms often do not perform well in practice. Common wisdom holds that they yield poor results because they are overly sensitive to initial parameter values and easily get stuck in local (but not global) maxima. We present a series of experiments indicating that for the task of learning syllable structure, the initial parameter weights are not crucial. Rather, it is the choice of model class itself that makes the difference between successful and unsuccessful learning. We use a language-universal rule-based algorithm to find a good set of parameters, and then train the parameter weights using EM. We achieve word accuracy of 95.9% on German and 97.1% on English, as compared to 97.4% and 98.1% respectively for supervised training.

1 Introduction

The use of statistical methods in computational linguistics has produced advances in tasks such as parsing, information retrieval, and machine translation. However, most of the successful work to date has used supervised learning techniques. Unsupervised algorithms that can learn from raw linguistic data, as humans can, remain a challenge. In a statistical framework, one method that can be used for unsupervised learning is to devise a probabilistic model of the data, and then choose the values for the model parameters that maximize the likelihood of the data under the model.

If the model contains hidden variables, there is often no closed-form expression for the maximum likelihood parameter values, and some iterative approximation method must be used. Expectation Maximization (EM) (Neal and Hinton, 1998) is one way to find parameter values that at least locally maximize the likelihood for models with hidden variables. EM is attractive because at each iteration, the likelihood of the data is guaranteed not to decrease. In addition, there are efficient dynamic-programming versions of the EM algorithm for several classes of models that are important in computational linguistics, such as the forward-backward algorithm for training Hidden Markov Models (HMMs) and the inside-outside algorithm for training Probabilistic Context-Free Grammars (PCFGs).

Despite the advantages of maximum likelihood estimation and its implementation via various instantiations of the EM algorithm, it is widely regarded as ineffective for unsupervised language learning. Merialdo (1994) showed that with only a tiny amount of tagged training data, supervised training of an HMM part-of-speech tagger outperformed unsupervised EM training. Later results (e.g. Brill (1995)) seemed to indicate that other methods of unsupervised learning could be more effective (although the work of Banko and Moore (2004) suggests that the difference may be far less than previ-

Proceedings of the 9th Conference on Computational Natural Language Learning (CoNLL),
pages 112–119, Ann Arbor, June 2005.

ously assumed). Klein and Manning (2001; 2002) recently achieved more encouraging results using an EM-like algorithm to induce syntactic constituent grammars, based on a deficient probability model.

It has been suggested that EM often yield poor results because it is overly sensitive to initial parameter values and tends to converge on likelihood maxima that are local, but not global (Carroll and Charniak, 1992). In this paper, we present a series of experiments indicating that for the task of learning a syllable structure grammar, the initial parameter weights are not crucial. Rather, it is the choice of the model class, i.e., the *representational bias*, that makes the difference between successful and unsuccessful learning.

In the remainder of this paper, we first describe the task itself and the structure of the two different classes of models we experimented with. We then present a deterministic algorithm for choosing a good set of parameters for this task. The algorithm is based on language-universal principles of syllabification, but produces different parameters for each language. We apply this algorithm to English and German data, and describe the results of experiments using EM to learn the parameter weights for the resulting models. We conclude with a discussion of the implications of our experiments.

2 Statistical Parsing of Syllable Structure

Knowledge of syllable structure is important for correct pronunciation of spoken words, since certain phonemes may be pronounced differently depending on their position in the syllable. A number of different supervised machine learning techniques have been applied to the task of automatic syllable boundary detection, including decision-tree classifiers (van den Bosch et al., 1998), weighted finite state transducers (Kiraz and Möbius, 1998), and PCFGs (Müller, 2001; Müller, 2002). The researchers presenting these systems have generally argued from the engineering standpoint that syllable boundary detection is useful for pronunciation of unknown words in text-to-speech systems. Our motivation is a more scientific one: we are interested in the kinds of procedures and representations that can lead to successful unsupervised language learning in both computers and humans.

Our work has some similarity to that of Müller, who trains a PCFG of syllable structure from a corpus of words with syllable boundaries marked. We, too, use a model defined by a grammar to describe syllable structure.[1] However, our work differs from Müller's in that it focuses on how to learn the model's parameters in an unsupervised manner. Several researchers have worked on unsupervised learning of phonotactic constraints and word segmentation (Elman, 2003; Brent, 1999; Venkataraman, 2001), but to our knowledge there is no previously published work on unsupervised learning of syllable structure.

In the work described here, we experimented with two different classes of models of syllable structure. Both of these model classes are presented as PCFGs. The first model class, described in Müller (2002), encodes information about the positions within a word or syllable in which each phoneme is likely to appear. In this *positional* model, each syllable is labeled as initial (I), medial (M), final (F), or as the one syllable in a monosyllabic word (O). Syllables are broken down into an optional onset (the initial consonant or consonant cluster) followed by a rhyme. The rhyme consists of a nucleus (the vowel) followed by an optional coda consonant or cluster. Each phoneme is labeled with a preterminal category of the form *CatPos.x.y*, where *Cat* $\in$ {*Ons, Nuc, Cod*}, *Pos* $\in$ {*I, M, F, O*}, x is the position of a consonant within its cluster, and y is the total number of consonants in the cluster. x and y are unused when *Cat = Nuc*, since all nuclei consist of a single vowel. See Fig. 1 for an example parse.

Rather than directly encoding positional information, the second model class we investigate (the *bigram* model) models statistical dependencies between adjacent phonemes and adjacent syllables. In particular, each onset or coda expands directly into one or more terminal phonemes, thus capturing the ordering dependencies between consonants in a cluster. Also, the shape of each syllable (whether it contains an onset or coda) depends on the shape of the previous syllable, so that the model can learn, for example, that syllables ending in a coda should be followed by syllables with an onset.[2] This kind

[1] We follow Müller in representing our models as PCFGs because this representation is easy to present. The languages generated by these PCFGs are in fact regular, and it is straightforward to transform the PCFGs into equivalent regular grammars.

[2] Many linguists believe that, cross-linguistically, a poten-

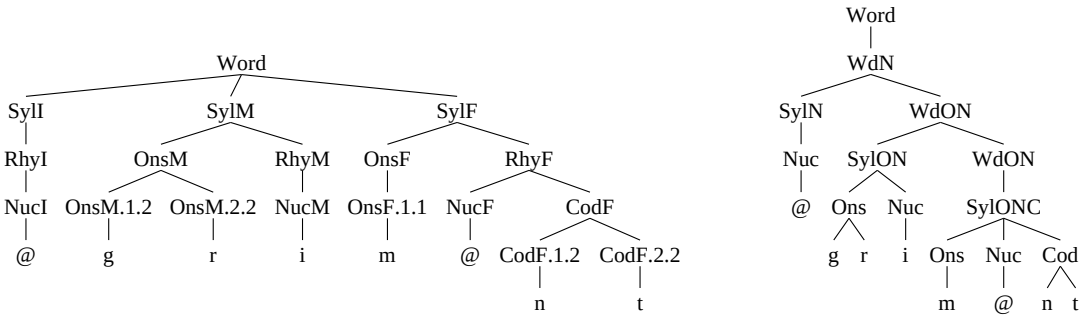

Figure 1: Positional analysis (left) and bigram analysis (right) of the word *agreement*. Groups of terminals dominated by a Syl* node constitute syllables. Terminals appear in the SAMPA encoding of IPA used by CELEX.

of bigram dependency between syllables is modeled using rules of the form WdX $\rightarrow$ Syl$X$ WdY, where X and Y are drawn from the set of possible combinations of onset, nucleus, and coda in a syllable: {N, ON, NC, ONC}. Each Syl$X$ category has only one expansion. See Fig. 1 for an example.

With respect to either of these two model classes, each way of assigning syllable boundaries to a word corresponds to exactly one parse of that word. This makes it simple to train the models from a corpus in which syllable boundaries are provided, as in Müller (2001). We used two different corpora for our experiments, one German (from the ECI corpus of newspaper text) and one English (from the Penn WSJ corpus). Each corpus was created by converting the orthographic forms in the original text into their phonemic transcriptions using the CELEX database (Baayen et al., 1995). CELEX includes syllable boundaries, which we used for supervised training and for evaluation. Any words in the original texts that were not listed in CELEX were discarded, since one of our goals is to compare supervised and unsupervised training.[3] From the resulting phonemic corpora, we created a training set of 20,000 tokens and a test set of 10,000 tokens. Using standard maximum likelihood supervised training procedures, we obtained similar results for models from the two model classes. In German, word accuracy (i.e. the percentage of words with no syllabification errors) was 97.4% for the bigram model and 97.2% for the positional model,[4] while in English it was 98.1% and 97.6% respectively. These results for English are in line with previous reported results using other supervised learning techniques, e.g. van den Bosch et al. (1998). Since many of the words in the data are monosyllabic (49.1% in German, 61.2% in English) and therefore contain no ambiguous syllable boundaries, we also calculated the multisyllabic word accuracy. This was 94.9% (bigram) and 94.5% (positional) in German, and 95.2% (bigram) and 93.8% (positional) in English.

3 Categorical Parsing of Syllable Structure

In the previous section, we described two different model classes and showed that the maximum likelihood estimates with supervised training data yield good models of syllable structure. In moving to unsupervised learning, however, there are two problems that need to be addressed: exactly what class of models do we want to consider (i.e., what kinds of rules should the model contain), and how should we select a particular model from that class (i.e., what weights should the rules have)? We take as our solution to the latter problem the most straightforward approach; namely, maximum likelihood estimation using EM. This leaves us with the question of how to choose a set of parameters in the first place. In this section, we describe an algorithm based on two fundamental phonological principles that, when given a set of data from a particular language, will produce a

tially ambiguous consonant, such as the *b* in *saber*, is always syllabified as the onset of the second syllable rather than the coda of the first. We discuss this point further in Section 3.

[3]Due to the nature of the corpora, the percentage of words discarded was fairly high: 35.6% of the English tokens (primarily proper nouns, acronyms, and numerals, with a smaller number of morphologically complex words) and 26.7% of the German tokens (with compound words making up a somewhat larger portion of these discards).

[4]Müller reports slightly lower results of 96.88% on German using the same positional model. We have no explanation for this discrepancy.

set of rules appropriate to that language. These rules can then be trained using EM.

Given a particular rule schema, it is not immediately clear which of the possible rules should actually be included in the model. For example, in the bigram model, should we start off with the rule *Ons* $\rightarrow$ *k n*? This rule is unnecessary for English, and could lead to incorrect parses of words such as *weakness*. But /kn/ is a legal onset in German, and since we want an algorithm that is prepared to learn any language, disallowing /kn/ as an onset out of hand is unacceptable. On the other hand, the set of all combinatorially possible consonant clusters is infinite, and even limiting ourselves to clusters actually seen in the data for a particular language yields extremely unlikely-sounding onsets like /lkj/ (*calculate*) and /bst/ (*substance*). Ideally, we should limit the set of rules to ones that are likely to actually be used in the language of interest.

The algorithm we have developed for producing a set of language-appropriate rules is essentially a simple categorical (i.e., non-statistical) syllable parser based on the principles of *onset maximization* and *sonority sequencing* (Blevins, 1995). Onset maximization is the idea that in word-medial consonant clusters, as many consonants as possible (given the phonotactics of the language) should be assigned to onset position. This idea is widely accepted and has been codified in Optimality Theory (Prince and Smolensky, 1993) by proposing the existence of a universal preference for syllables with onsets.[5]

In addition to onset maximization, our categorical parser follows the principle of sonority sequencing whenever possible. This principle states that, within a syllable, segments that are closer to the nucleus should be higher in sonority than segments that are further away. Vowels are considered to be the most sonorous segments, followed by glides (/j/, /w/), liquids (/l/, /r/), nasals (/n/, /m/, /ŋ/), fricatives (/v/, /s/, /θ/, ...), and stops (/b/, /t/, /k/, ...). Given a cluster of consonants between two syllable nuclei, sonority sequencing states that the syllable boundary should occur either just before or just after the consonant with lowest sonority. Combining this principle with onset maximization predicts that the boundary should fall before the lowest-sonority segment.

Predicting syllable boundaries in this way is not foolproof. In some cases, clusters that are predicted by sonority sequencing to be acceptable are in fact illegal in some languages. The illegal English onset cluster *kn* is a good example. In other cases, such as the English onset *str*, clusters are allowed despite violating sonority sequencing. These mismatches between universal principles and language-specific phonotactics lead to errors in the predictions of the categorical parser, such as *wea.kness* and *ins.tru.ment*. In addition, certain consonant clusters like *bst* (as in *substance*) may contain more than one minimum sonority point. To handle these cases, the categorical parser follows onset maximization by adding any consonants occurring between the two minima to the onset of the second syllable: *sub.stance*.

Not surprisingly, the categorical parser does not perform as well as the supervised statistical parser: only 92.7% of German words and 94.9% of English words (85.7% and 86.8%, respectively, of multisyllabic words) are syllabified correctly. However, a more important result of parsing the corpus using the categorical parser is that its output can be used to define a model class (i.e., a set of PCFG rules) from which a model can be learned using EM.

Specifically, our model class contains the set of rules that were proposed at least once by the categorical parser in its analysis of the training corpus; in the EM experiments described below, the rule probabilities are initialized to their frequency in the categorical parser's output. Due to the mistakes made by the categorical parser, there will be some rules, like *Ons* $\rightarrow$ *k n* in English, that are not present in the model trained on the true syllabification, but many possible but spurious rules, such as *Ons* $\rightarrow$ *b s t*, will be avoided. Although clusters that violate sonority sequencing tend to be avoided by the categorical parser, it does find examples of these types of clusters at the beginnings and endings of words, as well as occasionally word-medially (as in *sub.stance*). This means that many legal clusters that

[5] An important point, which we return to in Section 5, is that exceptions to onset maximization may occur at morpheme boundaries. Some linguists also believe that there are additional exceptions in certain languages (including English and German), where stressed syllables attract codas. Under this theory, the correct syllabification for *saber* would not be *sa.ber*, but rather *sab.er*, or possibly *sa[b]er*, where the *[b]* is ambisyllabic. Since the syllable annotations in the CELEX database follow simple onset maximization, we take that as our approach as well and do not consider stress when assigning syllable boundaries.

	Bigram		Positional	
	all	multi	all	multi
CP	92.7	85.7	92.7	85.7
CP + EM	95.9	91.9	91.8	84.0
CP-U + EM	95.9	91.9	92.0	84.4
supervised	97.4	94.9	97.2	94.5
SP + EM	71.6	44.3	94.4	89.1
SP-U + EM	71.6	44.3	94.4	89.0

Table 1: Results for German: % of all words (or multisyllabic words) correctly syllabified.

	Bigram		Positional	
	all	multi	all	multi
CP	94.9	86.8	94.9	86.8
CP + EM	97.1	92.6	94.1	84.9
CP-U + EM	97.1	92.6	94.1	84.9
supervised	98.1	95.2	97.6	93.8
SP + EM	86.0	64.0	96.5	90.9
SP-U + EM	86.0	64.0	67.6	16.5

Table 2: Results for English.

violate sonority sequencing will also be included in the set of rules found by this procedure, although their probabilities may be considerably lower than those of the supervised model. In the following section, we show that these differences in rule probabilities are unimportant; in fact, it is not the rule probabilities estimated from the categorical parser's output, but only the set of rules itself that matters for successful task performance.

4 Experiments

In this section, we present a series of experiments using EM to learn a model of syllable structure. All of our experiments use the same German and English 20,000-word training corpora and 10,000-word testing corpora as described in Section 2.[6]

For our first experiment, we ran the categorical parser on the training corpora and estimated a model from the parse trees it produced, as described in the previous section. This is essentially a single step of Viterbi EM training. We then continued to train the model by running (standard) EM to convergence. Results of this experiment with Categorical Parsing + EM (CP + EM) are shown in Tables 1 and 2. For both German and English, using this learning method with the bigram model yields performance that is much better than the categorical parser alone, though not quite as good as the fully supervised regime. On the other hand, training a positional model from the categorical parser's output and then running EM causes performance to degrade.

To determine whether the good performance of the bigram model was simply due to good initialization of the parameter weights, we performed a second experiment. Again starting with the set of rules output by the categorical parser, we initialized the rule weights to the uniform distribution. The results of this experiment (CP-U + EM) show that for the class of bigram models, the performance of the final model found by EM does not depend on the initial rule probabilities. Performance within the positional model framework does depend on the initial rule probabilities, since accuracy in German is different for the two experiments.

As we have pointed out, the rules found by the categorical parser are not exactly the same as the rules found using supervised training. This raises the question of whether the difference in performance between the unsupervised and supervised bigram models is due to differences in the rules. To address this question, we performed two additional experiments. First, we simply ran EM starting from the model estimated from supervised training data. Second, we kept the set of rules from the supervised training data, but reinitialized the probabilities to a uniform distribution before running EM. The results of these experiments are shown as SP + EM and SP-U + EM, respectively. Again, performance of the bigram model is invariant with respect to initial parameter values, while the performance of the positional model is not. Interestingly, the performance of the bigram model in these two experiments is far worse than in the CP experiments. This result is counterintuitive, since it would seem that the model rules found by the supervised system are the optimal rules for this task. In the following section, we explain why these rules are not, in fact, the optimal rules for unsupervised learning, as well as why we believe the bigram model performs so much better

[6] Of course, for unsupervised learning, it is not necessary to use a distinct testing corpus. We did so in order to use the same testing corpus for both supervised and unsupervised learning experiments, to ensure fair comparison of results.

than the positional model in the unsupervised learning situation.

5 Discussion

The results of our experiments raise two interesting questions. First, when starting from the categorical parser's output, why does the bigram model improve after EM training, while the positional model does not? And second, why does applying EM to the supervised bigram model lead to worse performance than applying it to the model induced from the categorical parser?

To answer the first question, notice that one difference between the bigram model and the positional model is that onsets and codas in the bigram model are modeled using the same set of parameters regardless of where in the word they occur. This means that the bigram model generalizes whatever it learns about clusters at word edges to word-medial clusters (and, of course, vice versa). Since the categorical parser only makes errors word-medially, incorrect clusters are only a small percentage of clusters overall, and the bigram model can overcome these errors by reanalyzing the word-medial clusters. The errors that are made after EM training are mostly due to overgeneralization from clusters that are very common at word edges, e.g. predicting le.gi.sla.tion instead of le.gis.la.tion.

In contrast to the bigram model, the positional model does not generalize over different positions of the word, which means that it learns and repeats the word-medial errors of the categorical parser. For example, this model predicts /ɛ.gzɛ.kju.tɪv/ for *executive*, just as the categorical parser does, although /gz/ is never attested in word-initial position. In addition, each segment in a cluster is generated independently, which means clusters like /tl/ may be placed together in an onset because /t/ is common as the first segment of an onset, and /l/ is common as the second. While this problem exists even in the supervised positional model, it is compounded in the unsupervised version because of the errors of the categorical parser.

The differences between these two models are an example of the bias-variance trade-off in probabilistic modeling (Geman et al., 1992): models with low bias will be able to fit a broad range of observations fairly closely, but slight changes in the observed data will cause relatively large changes in the induced model. On the other hand, models with high bias are less sensitive to changes in the observed data. Here, the bigram model induced from the categorical parser has a relatively high bias: regardless of the parameter weights, it will be a poor model of data where word-medial onsets and codas are very different from those at word edges, and it cannot model data with certain onsets such as /vp/ or /tz/ at all because the rules $Ons \rightarrow v\ p$ and $Ons \rightarrow t\ z$ are simply absent. The induced positional model can model both of these situations, and can fit the true parses more closely as well (as evidenced by the fact that the likelihood of the data under the supervised positional model is higher than the likelihood under the supervised bigram model). As a result, however, it is more sensitive to the initial parameter weights and learns to recreate the errors produced by the categorical parser. This sensitivity to initial parameter weights also explains the extremely poor performance of the positional model in the SP-U + EM experiment on English. Because the model is so unconstrained, in this case it finds a completely different local maximum (not the global maximum) which more or less follows coda maximization rather than onset maximization, yielding syllabifications like *synd.ic.ate* and *tent.at.ive.ly*.

The concept of representational bias can also explain why applying EM to the supervised bigram model performs so poorly. Examining the model induced from the categorical parser reveals that, not surprisingly, it contains more rules than the supervised bigram model. This is because the categorical parser produces a wider range of onsets and codas than there are in the true parses. However, the induced model is not a superset of the supervised model. There are four rules (three in English) that occur in the supervised model but not the induced model. These are the rules that allow words where one syllable contains a coda and the following syllable has no onset. These are never produced by the categorical parser because of its onset-maximization principle. However, it turns out that a very small percentage of words do follow this pattern (about .14% of English tokens and 1.1% of German tokens). In English, these examples seem to consist entirely of words where the unusual syllable boundary occurs at a morpheme boundary (e.g. *un.usually, dis.appoint,*

week.end, turn.over). In German, all but a handful of examples occur at morpheme boundaries as well.[7]

The fact that the induced bigram model is unable to model words with codas followed by no onset is a very strong bias, but these words are so infrequent that the model can still fit the data quite well. The missing rules have no effect on the accuracy of the parser, because in the supervised model the probabilities on the rules allowing these kinds of words are so low that they are never used in the Viterbi parses anyway. The problem is that if these rules are included in the model prior to running EM, they add several extra free parameters, and suddenly EM is able to reanalyze many of the words in the corpus to make better use of these parameters. It ends up preferring certain segments and clusters as onsets and others as codas, which raises the likelihood of the corpus but leads to very poor performance. Essentially, it seems that the presence of a certain kind of morpheme boundary is an additional parameter of the "true" model that the bigram model doesn't include. Trying to account for the few cases where this parameter matters requires introducing extra parameters that allow EM too much freedom of analysis. It is far better to constrain the model, disallowing certain rare analyses but enabling the model to learn successfully in a way that is robust to variations in initial conditions and idiosyncracies of the data.

6 Conclusion

We make no claims that our learning system embodies a complete model of syllabification. A full model would need to account for the effects of morphological boundaries, as well as the fact that some languages allow resyllabification over word boundaries. Nevertheless, we feel that the results presented here are significant. We have shown that, despite previous discouraging results (Carroll and Charniak, 1992; Merialdo, 1994), it is possible to achieve good results using EM to learn linguistic structures in an unsupervised way. However, the choice of model parameters is crucial for successful learning. Carroll and Charniak, for example, generated all possible rules within a particular framework and relied on EM to remove the "unnecessary" rules by letting their probabilities go to zero. We suggest that this procedure tends to yield models with low bias but high variance, so that they are extremely sensitive to the small variations in expected rule counts that occur with different initialization weights.

Our work suggests that using models with higher bias but lower variance may lead to much more successful results. In particular, we used universal phonological principles to induce a set of rules within a carefully chosen grammatical framework. We found that there were several factors that enabled our induced bigram model to learn successfully where the comparison positional model did not:

1. The bigram model encodes bigram dependencies of syllable shape and disallows onset-less syllables following syllables with codas.

2. The bigram model does not distinguish between different positions in a word, so it can generalize onset and coda sequences from word edges to word-medial position.

3. The bigram model learns specific sequences of legal clusters rather than information about which positions segments are likely to occur in.

Notice that each of these factors imposes a constraint on the kinds of data that can be modeled. We have already discussed the fact that item 1 rules out the correct syllabification of certain morphologically complex words, but since our system currently has no way to determine morpheme boundaries, it is better to do so than to introduce extra free parameters. One possible extension to this work would be to try to incorporate morphological boundary information (either annotated or induced) into the model.

A more interesting constraint is the one imposed by item 2, since in fact most languages do have some differences between the onsets and (especially) codas allowed at word edges and within words. However, the proper way to handle this fact is not by introducing completely independent parameters for initial, medial, and final positions, since this allows far too much freedom. It would be extremely surprising to find a language with one set of codas allowed word-internally, and a completely disjoint set

[7] The exceptions in our training data were *auserkoren 'chosen'*, *erobern 'capture'*, and forms of *erinnern 'remind'*, all of which were listed in CELEX as having a syllable boundary, but no morpheme boundary, after the first consonant. Our knowledge of German is not sufficient to determine whether there is some other factor that can explain these cases.

allowed word-finally. In fact, the usual situation is that word-internal onsets and codas are a subset of those allowed at word edges, and this is exactly why using word edges to induce our rules was successful.

Considering language more broadly, it is common to find patterns of linguistic phenomena with many similarities but some differences as well. For such cases, adding extra parameters to a supervised model often yields better performance, since the augmented model can capture both primary and secondary effects. But it seems that, at least for the current state of unsupervised learning, it is better to limit the number of parameters and focus on those that capture the main effects in the data. In our task of learning syllable structure, we were able to use just a few simple principles to constrain the model successfully. For more complex tasks such as syntactic parsing, the space of linguistically plausible models is much larger. We feel that a research program integrating results from the study of linguistic universals, human language acquisition, and computational modeling is likely to yield the most insight into the kinds of constraints that are needed for successful learning.

Ultimately, of course, we will want to be able to capture not only the main effects in the data, but some of the subtler effects as well. However, we believe that the way to do this is not by introducing completely free parameters, but by using a Bayesian prior that would enforce a degree of similarity between certain parameters. In the meantime, we have shown that employing linguistic universals to determine which set of parameters to include in a language model for syllable parsing allows us to use EM for learning the parameter weights in a successful and robust way.

Acknowledgments

We would like to thank Eugene Charniak and our colleagues in BLLIP for their support and helpful suggestions. This research was partially supported by NSF awards IGERT 9870676 and ITR 0085940 and NIMH award 1R0-IMH60922-01A2.

References

R. Baayen, R. Piepenbrock, and L. Gulikers. 1995. The CELEX lexical database (release 2) [cd-rom].

M. Banko and R. Moore. 2004. A study of unsupervised part-of-speech tagging. In *Proceedings of COLING '04*.

J. Blevins. 1995. The syllable in phonological theory. In J. Goldsmith, editor, *the Handbook of Phonological Theory*. Blackwell, Oxford.

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

E. Brill. 1995. Unsupervised learning of disambiguation rules for part of speech tagging. In *Proceedings of the 3rd Workshop on Very Large Corpora*, pages 1–13.

G. Carroll and E. Charniak. 1992. Two experiments on learning probabilistic dependency grammars from corpora. In *Proceedings of the AAAI Workshop on Statistically-Based Natural Language Processing Techniques*, San Jose, CA.

J. Elman. 2003. Generalization from sparse input. In *Proceedings of the 38th Annual Meeting of the Chicago Linguistic Society*.

S. Geman, E. Bienenstock, and R. Doursat. 1992. Neural networks and the bias/variance dilemma. *Neural Computation*, 4:1–58.

G. A. Kiraz and B. Möbius. 1998. Multilingual syllabification using weighted finite-state transducers. In *Proceedings of the Third European Speech Communication Association Workshop on Speech Synthesis*.

D. Klein and C. Manning. 2001. Distributional phrase structure induction. In *Proceedings of the Conference on Natural Language Learning*, pages 113–120.

D. Klein and C. Manning. 2002. A generative constituent-context model for improved grammar induction. In *Proceedings of the ACL*.

B. Merialdo. 1994. Tagging english text with a probabilistic model. *Computational Linguistics*, 20(2):155–172.

K. Müller. 2001. Automatic detection of syllable boundaries combining the advantages of treebank and bracketed corpora training. In *Proceedings of the ACL*.

K. Müller. 2002. Probabilistic context-free grammars for phonology. In *Proceedings of the Workshop on Morphological and Phonological Learning at ACL*.

R. Neal and G. Hinton, 1998. *A New View of the EM Algorithm That Justifies Incremental and Other Variants*, pages 355–368. Kluwer.

A. Prince and P. Smolensky. 1993. Optimality theory: Constraint interaction in generative grammar. Technical Report TR-2, Rutgers Center for Cognitive Science, Rutgers Univ.

A. van den Bosch, T. Weijters, and W. Daelemans. 1998. Modularity in inductively-learned word pronunciation systems. In *New Methods in Language Processing and Computational Language Learning (NeMLaP3/CoNLL98)*.

A. Venkataraman. 2001. A statistical model for word discovery in transcribed speech. *Computational Linguistics*, 27(3):351–372.