Knowledge-Rich Morphological Priors for Bayesian Language Models

Victor Chahuneau **Noah A. Smith** **Chris Dyer**
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA 15213, USA
{vchahune,nasmith,cdyer}@cs.cmu.edu

Abstract

We present a morphology-aware nonparametric Bayesian model of language whose prior distribution uses manually constructed finite-state transducers to capture the word formation processes of particular languages. This relaxes the word independence assumption and enables sharing of statistical strength across, for example, stems or inflectional paradigms in different contexts. Our model can be used in virtually any scenario where multinomial distributions over words would be used. We obtain state-of-the-art results in language modeling, word alignment, and unsupervised morphological disambiguation for a variety of morphologically rich languages.

1 Introduction

Despite morphological phenomena's salience in most human languages, many NLP systems treat fully inflected forms as the atomic units of language. By assuming independence of lexical stems' various surface forms, this *avoidance approach* exacerbates the problem of data sparseness. If it is employed at all, morphological analysis of text tends to be treated as a preprocessing step to other NLP modules. While this latter *disambiguation approach* helps address data sparsity concerns, it has substantial drawbacks: it requires supervised learning from expert-annotated corpora, and determining the optimal morphological granularity is labor-intensive (Habash and Sadat, 2006).

Neither approach fully exploits the finite-state transducer (FST) technology that has been so successful for modeling the mapping between surface forms and their morphological analyses (Karttunen and Beesley, 2005), and the mature collections of high quality transducers that already exist for many languages (e.g., Turkish, Russian, Arabic). Much linguistic knowledge is encoded in such FSTs.

In this paper, we develop morphology-aware nonparametric Bayesian language models that bring together hand-written FSTs with statistical modeling and require no token-level annotation. The sparsity issue discussed above is addressed by hierarchical priors that share statistical strength across different inflections of the same stem by backing off to word formation models that piece together morphemes using FSTs. Furthermore, because of the nonparametric formulation of our models, the regular morphological patterns found in the long tail of word types will rely more heavily on deeper analysis, while frequent and idiosyncratically behaved forms are modeled opaquely.

Our prior can be used in virtually any generative model of language as a replacement for multinomial distributions over words, bringing morphological awareness to numerous applications. For various morphologically rich languages, we show that:

- our model can provide rudimentary unsupervised disambiguation for a highly ambiguous analyzer;
- integrating morphology into n-gram language models allows better generalization to unseen words and can improve the performance of applications that are truly open vocabulary; and
- bilingual word alignment models also benefit greatly from sharing translation information

across stems.

We are particularly interested in low-resource scenarios, where one has to make the most of the small quantity of available data, and overcoming data sparseness is crucial. If analyzers exist in such settings, they tend to be highly ambiguous, and annotated data for learning to disambiguate are also likely to be scarce or non-existent. Therefore, in our experiments with Russian, we compare two analyzers: a rapidly-developed *guesser*, which models regular inflectional paradigms but contains no lexicon or irregular forms, and a high-quality analyzer.

2 Word Models with Morphology

In this section, we describe a generative model of word formation based on Pitman-Yor processes that generates word types using a finite-state morphological generator. At a high level, the process first produces lexicons of stems and inflectional patterns; then it generates a lexicon of inflected forms using the finite-state generator. Finally, the inflected forms are used to generate observed data. Different independence assumptions can be made at each of these levels to encode beliefs about where stems, inflections, and surface forms should share statistical strength.

2.1 Pitman-Yor Processes

Our work relies extensively on Pitman-Yor processes, which provide a flexible framework for expressing backoff and interpolation relationships and extending standard models with richer word distributions (Pitman and Yor, 1997). They have been shown to match the performance of state-of-the-art language models and to give estimates that follow appropriate power laws (Teh, 2006).

A draw from a Pitman-Yor process (PYP), denoted $G \sim \mathrm{PY}(d, \theta, G_0)$, is a discrete distribution over a (possibly infinite) set of events, which we denote abstractly $\mathcal{E}$. The process is parameterized by a discount parameter $0 \leq d < 1$, a strength parameter $\theta > -d$, and a base distribution G_0 over the event space $\mathcal{E}$.

In this work, our focus is on the base distribution G_0. We place vague priors on the hyperparameters $d \sim \mathbb{U}([0, 1])$ and $(\theta + d) \sim \mathrm{Gamma}(1, 1)$. Inference in PYPs is discussed below.

2.2 Unigram Morphology Model

The most basic expression of our model is a unigram model of text. So far, we only assume that each word can be analyzed into a stem and a sequence of morphemes forming an inflection pattern. Let G_s be a distribution over stems, G_p be a distribution over inflectional patterns, and let $\textsc{Generate}$ be a deterministic mapping from $\langle \mathrm{stem}, \mathrm{pattern} \rangle$ pairs to inflected word forms.[1] An inflected word *type* is generated with the following process, which we designate $\mathrm{MP}(G_s, G_d, \textsc{Generate})$:

$$\begin{aligned}
\mathrm{stem} &\sim G_s \\
\mathrm{pattern} &\sim G_p \\
\mathrm{word} &= \textsc{Generate}(\mathrm{stem}, \mathrm{pattern})
\end{aligned}$$

For example, in Russian, we might sample stem = прочий,[2] pattern = $\textsc{Stem}$`+Adj+Pl+Dat`, and obtain word = прочим.

This model could be used directly to generate observed tokens. However, we have said nothing about G_s and G_p, and the assumption that stems and patterns are independent is clearly unsatisfying. We therefore assume that both the stem and the pattern distributions are generated from PY processes, and that $\mathrm{MP}(G_s, G_p, \textsc{Generate})$ is itself the base distribution of a PYP.

$$\begin{aligned}
G_s &\sim \mathrm{PY}(d_s, \theta_s, G_s^0) \\
G_p &\sim \mathrm{PY}(d_p, \theta_p, G_p^0) \\
G_w &\sim \mathrm{PY}(d, \theta, \mathrm{MP}(G_s, G_p, \textsc{Generate}))
\end{aligned}$$

A draw G_w from this PYP is a unigram distribution over tokens.

2.3 Base Stem Model G_s^0

In general there are an unbounded number of stems possible in any language, so we set G_s^0 to be character trigram model, which we statically estimate, with Kneser-Ney smoothing, from a large corpus of word types in the language being modeled. While using fixed parameters estimated to maximize likelihood is

[1] The assumption of determinism is only inappropriate in cases of inflectional spelling variants (e.g., *modeled* vs. *modelled*) or pronunciation variants (e.g., reduced forms in certain environments).

[2] прочий (pronounced [prɐtɕij]) = *other*

questionable from the perspective of Bayesian learning, it is tremendously beneficial for computational reasons. For some applications (e.g., word alignment), the set of possible stems for a corpus S can be precomputed, so we will also experiment with using a uniform stem distribution based on this set.

2.4 Base Pattern Model G_p^0

Several choices are possible for the base pattern distribution:

MP$_0$ We can assume a uniform G_p^0 when the number of patterns is small.

MP$_1$ To be able to generalize to new patterns, we can draw the length of the pattern from a Poisson distribution and generate morphemes one by one from a uniform distribution.

MP$_2$ A more informative prior is a Markov chain of morphemes, where each morpheme is generated conditional on the preceding morpheme.

The choice of the base pattern distribution could depend on the complexity of the inflectional patterns produced by the morphological analyzer, reflecting the type of morphological phenomena present in a given language. For example, the number of possible patterns can practically be considered finite in Russian, but this assumption is not valid for languages with more extensive derivational morphology like Turkish.

2.5 Posterior Inference

For most applications, rather than directly generating from a model using the processes outlined above, we seek to infer posterior distributions over latent parameters and structures, given a sample of data.

Although there is no known analytic form of the PYP density, it is possible to marginalize the draws from it and to work directly with observations. This marginalization produces the classical Chinese restaurant process representation (Teh, 2006). When working with the morphology models we are proposing, we also need to marginalize the different latent forms (stems s and patterns p) that may have given rise to a given word w. Thus, we require that the inverse relation of GENERATE is available to compute the marginal base word distribution:

$$p(w \mid G_w^0) = \sum_{\textsc{Generate}(s,p)=w} p(s \mid G_s)\, p(p \mid G_p)$$

Since our approach encodes morphology using FSTs, which are invertible, this poses no problem.

To illustrate, consider the Russian word прочим, which may be analyzed in several ways:

прочий	`+Adj +Sg +Neut +Instr`
прочий	`+Adj +Sg +Masc +Instr`
прочий	`+Adj +Pl +Dat`
прочить	`+Verb +Pl +1P`
прочее	`+Pro +Sg +Ins`

Because the set of possible analyses is in general small, marginalization is fast and complex blocked sampling is not necessary.

Finally, to infer hyperparameter values $(d, \theta, \ldots)$, a Metropolis-Hastings update is interleaved with Gibbs sampling steps for the rest of the hidden variables.[3]

Having described a model for generating words, we now show its usage in several contexts.

3 Unsupervised Morphological Disambiguation

Given a rule-based morphological analyzer encoded as an unweighted FST and a corpus on which the analyzer has been run – possibly generating multiple analyses for each token – we can use our unigram model to learn a probabilistic model of disambiguation in an unsupervised setting (i.e., without annotated examples). The corpus is assumed to be generated from the unigram distribution G_w, and the base stem model is set to a fixed character trigram model.[4] After learning the parameters of the model, we can find for each word in the vocabulary its most likely analysis and use this as a crude disambiguation step.

[3] The proposal distribution for Metropolis-Hastings is a Beta distribution (d) or a Gamma distribution ($\theta + d$) centered on the previous parameter values.

[4] Experiments suggest that this is important to constrain the model to realistic stems.

3.1 Morphological Guessers

Finite-state morphological analyzers are usually specified in three parts: a **stem lexicon**, which defines the words in the language and classifies them into several categories according to their grammatical function and their morphological properties; a set of **prefixes and suffixes** that can be applied to each category to form surface words; and possibly **alternation rules** that can encode exceptions and spelling variations. The combination of these parts provides a powerful framework for defining a generative model of words. Such models can be reversed to obtain an analyzer. However, while the two latter parts can be relatively easy to specify, enumerating a comprehensive stem lexicon is a time consuming and necessarily incomplete process, as some categories are truly open-class.

To allow unknown words to be analyzed, one can use a *guesser* that attempts to analyze words missing in the lexicon. Can we eliminate the stem lexicon completely and use only the guesser? This is what we try to do by designing a lexicon-free analyzer for Russian. A guesser was developed in three hours; it is prone to over-generation and produces ambiguous analyses for most words but covers a large number of morphological phenomena (gender, case, tense, etc.). For example, the word иврите[5] can be correctly analyzed as иврит+Noun+Masc+Prep+Sg but also as the incorrect forms: ивритъ+Verb+Pres+2P+Pl, иврита+Noun+Fem+Dat+Sg, ивритя+Noun+Fem+Prep+Sg, and more.

3.2 Disambiguation Experiments

We train the unigram model on a 1.7M-word corpus of TED talks transcriptions translated into Russian (Cettolo et al., 2012) and evaluate our analyzer against a test set consisting of 1,500 gold-standard analyses obtained from the morphology disambiguation task of the DIALOG 2010 conference (Lyaševskaya et al., 2010).[6]

Each analysis is composed of a lemma (иврит), a part of speech (Noun), and a sequence of additional functional morphemes (Masc,Prep,Sg). We consider only open-class categories: nouns, adjectives, adverbs and verbs, and evaluate the output of our model with three metrics: the lemma accuracy, the part-of-speech accuracy, and the morphology F-measure.[7]

As a baseline, we consider picking a random analysis from output of the analyzer or choosing the most frequent lemma and the most frequent morphological pattern.[8] Then, we use our model with each of the three versions of the pattern model described in §2.2. Finally, as an upper bound, we use the gold standard to select one of the analyses produced by the guesser.

Since our evaluation is not directly comparable to the standard for this task, we use for reference a high-quality analyzer from Xerox[9] disambiguated with the MP$_0$ model (all of the models have very close accuracy in this case).

Model	Lemma	POS	Morph.
Random	29.8	70.9	50.2
Frequency	31.1	74.4	48.8
Guesser MP$_0$	60.0	82.2	66.3
Guesser MP$_1$	58.9	82.5	69.5
Guesser MP$_2$	59.9	82.4	65.5
Guesser oracle	68.4	84.9	83.0
Xerox MP$_0$	83.6	96.4	78.1

Table 1: Russian morphological disambiguation.

Considering the amount of effort put in developing the guesser, the baseline POS tagging accuracy is relatively good. However, the disambiguation is largely improved by using our unigram model with respect to all the evaluation categories. We are still far from the performance of a high-quality analyzer but, in absence of such a resource, our technique might be a sensible option. We also note that there is no clear winner in terms of pattern model, and conclude that this choice is task-specific.

[5] иврите = *Hebrew* (masculine noun, prepositional case)

[6] http://ru-eval.ru

[7] F-measure computed for the set of additional morphemes and averaged over the words in the corpus.

[8] We estimate these frequencies by assuming each analysis of each token is uniformly likely, then summing fractional counts.

[9] http://open.xerox.com/Services/fst-nlp-tools/Pages/morphology

4 Open Vocabulary Language Models

We now integrate our unigram model in a hierarchical Pitman-Yor n-gram language model (Fig. 1). The training corpus words are assumed to be generated from a distribution G_w^n drawn from $\mathrm{PY}(d_n, \theta_n, G_w^{n-1})$, where G_w^{n-1} is defined recursively down to the base model G_w^0. Previous work Teh (2006) simply used $G_w^0 = \mathbb{U}(V)$ where V is the word vocabulary, but in our case G_w^0 is the MP defined in §2.2.

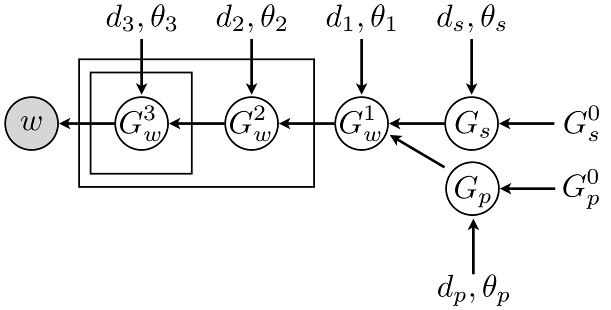

Figure 1: The trigram version of our language model represented as a graphical model. G_w^1 is the unigram model of §2.2.

We are interested in evaluating our model in an open vocabulary scenario where the ability to explain new unseen words matters. We expect our model to be able to generalize better thanks to the combination of a morphological analyzer and a stem distribution which is less sparse than the word distribution (for example, for the 1.6M word Turkish corpus, $|V| \approx 3.5|S| \approx 140\text{k}$).

To integrate out-of-vocabulary words in our evaluation, we use infinite base distributions: G_w^0 (in the baseline model) or G_s^0 (in the MP) are character trigram models. We define perplexity of a held-out test corpus in the standard way:

$$\mathrm{ppl} = \exp\left(-\frac{1}{N}\sum_{i=1}^{N} \log p\left(w_i \mid w_{i-n+1} \cdots w_{i-1}\right)\right)$$

but compared to the common practice, we do not need to discount OOVs from this sum since the model vocabulary is infinite. Note that we also marginalize by summing over all the possible analyses for a given word when computing its base probability according to the MP.

4.1 Language Modeling Experiments

We train several trigram models on the Russian TED talks corpus used in the previous section. Our baseline is a hierarchical PY trigram model with a trigram character model as the base word distribution. We compare it with our model using the same character model for the base *stem* distribution. Both of the morphological analyzers described in the previous section help obtaining perplexity reductions (Table 2). We ran a similar experiment on the Turkish version of this corpus (1.6M words) with a high-quality analyzer (Oflazer, 1994) and obtain even larger gains (Table 3).

Model	ppl
PY-character LM	563
Guesser MP$_2$	530
Xerox MP$_2$	522

Table 2: Evaluation of the Russian n-gram model.

Model	ppl
PY-character LM	1,640
Oflazer MP$_2$	1,292

Table 3: Evaluation of the Turkish n-gram model.

These results can partly be attributed to the high OOV rate in these conditions: 4% for the Russian corpus and 6% for the Turkish corpus.

4.2 Predictive Text Input

It is difficult to know whether a decrease in perplexity, as measured in the previous section, will result in a performance improvement in downstream applications. As a confirmation that correctly modeling new words matters, we consider a predictive task with a truly open vocabulary and that requires *only* a language model: predictive text input.

Given some text, we encode it using a lossy deterministic character mapping, and try to recover the original content by computing the most likely word sequence. This task is inspired by predictive text input systems available on cellphones with a 9-key keypad. For example, the string `gave me a cup` is encoded as `4283 63 2 287`, which could also be decoded as: `hate of a bus`.

Silfverberg et al. (2012) describe a system designed for this task in Finnish, which is composed of a weighted finite-state morphological analyzer trained on IRC logs. However, their system is restricted to words that are encoded in the analyzer's lexicon and does not use context for disambiguation.

In our experiments, we use the same Turkish TED talks corpus as the previous section. As a baseline, we use a trigram character language model. We produce a character lattice which encodes all the possible interpretations for a word and compose it with a finite-state representation of the character LM using OpenFST (Allauzen et al., 2007). Alternatively, we can use a unigram word model to decode this lattice, backing off to the character language model if no solution is found. Finally, to be able to make use of word context, we can extract the k most likely paths according to the character LM and produce a word lattice, which is in turn decoded with a language model defined over the extracted vocabulary.

Model	WER	CER
Character LM	48.37	16.72
1-gram + character LM	8.50	3.28
1-gram MP$_2$	**6.46**	**2.37**
3-gram + character LM	7.86	3.07
3-gram MP$_2$	**5.73**	**2.15**

Table 4: Evaluation of Turkish predictive text input.

We measure word and character error rate (WER, CER) on the predicted word sequence and observe large improvements in both of these metrics by modeling morphology, both at the unigram level and when context is used (Table 4).

Preliminary experiments with a corpus of 1.6M Turkish tweets, an arguably more appropriate domain this task, show smaller but consistent improving: the trigram word error rate is reduced from 26% to 24% when our model is used.

4.3 Limitations

While our model is an important step forward in practical modeling of OOVs using morphological processes, we have made the linguistically naive assumption that morphology applies inside the language's lexicon but has no effect on the process that put inflected lexemes together into sentences. In this regard, our model is a minor variant on traditional n-gram models that work with "opaque" word forms. How to best relax this assumption in a computationally tractable way is an important open question left for future work.

5 Word Alignment Model

Monolingual models of language are not the only models that can benefit from taking into account morphology. In fact, alignment models are a good candidate for using richer word distributions: they assume a target word distribution conditioned on each source word. When the target language is morphologically rich, classic independence assumptions produce very weak models unless some kind of preprocessing is applied to one side of the corpus. An alternative is to use our unigram model as a word translation distribution for each source word in the corpus.

Our alignment model is based on a simple variant of IBM Model 2 where the alignment distribution is only controlled by two parameters, λ and p_0 (Dyer et al., 2013). p_0 is the probability of the null alignment. For a source sentence f of length n, a target sentence e of length m and a latent alignment a, we define the following alignment link probabilities ($j \neq 0$):

$$p(a_i = j \mid n, m) \propto (1 - p_0) \exp\left(-\lambda \left|\frac{i}{m} - \frac{j}{n}\right|\right)$$

λ controls the flatness of this distribution: larger values make the probabilities more peaked around the diagonal of the alignment matrix.

Each target word is then generated given a source word and a latent alignment link from the word translation distribution $p(e_i \mid f_{a_i}, G_w)$. Note that this is effectively a unigram distribution over target words, albeit conditioned on the source word f_j. Here is where our model differs from classic alignment models: the unigram distribution G_w is assumed be generated from a PY process. There are two choices for the base word distribution:

- As a baseline, we use a uniform base distribution over the target vocabulary: $G^0_w = \mathbb{U}(V)$.
- We define a stem distribution $G_s[f]$ for each source word f, a shared pattern distribution G_p, and set $G^0_w[f] = \text{MP}(G_s[f], G_p)$. In this case,

we obtain the model depicted in Fig. 2. The stem and the pattern models are also given PY priors with uniform base distribution ($G_s^0 = \mathbb{U}(S)$).

Finally, we put uninformative priors on the alignment distribution parameters: $p_0 \sim \text{Beta}(\alpha, \beta)$ is collapsed and $\lambda \sim \text{Gamma}(k, \theta)$ is inferred using Metropolis-Hastings.

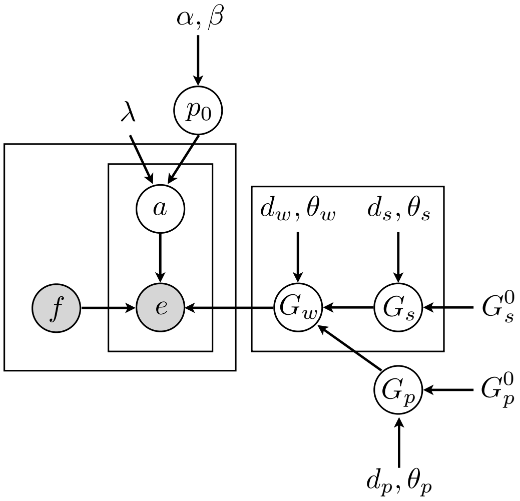

Figure 2: Our alignment model, represented as a graphical model.

Experiments

We evaluate the alignment error rate of our models for two language pairs with rich morphology on the target side. We compare to alignments inferred using IBM Model 4 trained with EM (Brown et al., 1993),[10] a version of our baseline model (described above) without PY priors (learned using EM), and the PY-based baseline. We consider two language pairs.

English-Turkish We use a 2.8M word cleaned version of the South-East European Times corpus (Tyers and Alperen, 2010) and gold-standard alignments from Çakmak et al. (2012). Our morphological analyzer is identical to the one used in the previous sections.

English-Czech We use the 1.3M word News Commentary corpus and gold-standard alignments from Bojar and Prokopová (2006). The morphological analyzer is provided by Xerox.

Results Results are shown in Table 5. Our lightly parameterized model performs much better than IBM Model 4 in these small-data conditions. With an identical model, we find PY priors outperform traditional multinomial distributions. Adding morphology further reduced the alignment error rate, for both languages.

	AER	
Model	en-tr	en-cs
Model 4	52.1	34.5
EM	43.8	28.9
PY-$\mathbb{U}(V)$	39.2	25.7
PY-$\mathbb{U}(S)$	**33.8**	**24.8**

Table 5: Word alignment experiments on English-Turkish (en-tr) and English-Czech (en-cs) data.

As an example of how our model generalizes better, consider the sentence pair in Fig. 3, taken from the evaluation data. The two words composing the Turkish sentence are not found elsewhere in the corpus, but several related inflections occur.[11] It is therefore trivial for the stem-base model to find the correct alignment (marked in black), while all the other models have no evidence for it and choose an arbitrary alignment (gray points).

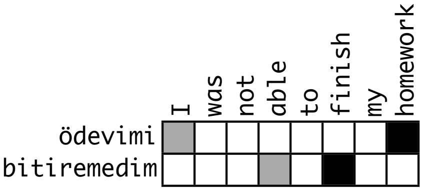

Figure 3: A complex Turkish-English word alignment (alignment points in gray: EM/PY-$\mathbb{U}(V)$; black: PY-$\mathbb{U}(S)$).

6 Related Work

Computational morphology has received considerable attention in NLP since the early work on *two-level morphology* (Koskenniemi, 1984; Kaplan and

[10] We use the default GIZA++ stage training scheme: Model 1 + HMM + Model 3 + Model 4.

[11] ödevinin, ödevini, ödevleri; bitmez, bitirileceğinden, bitmesiyle, ...

Kay, 1994). It is now widely accepted that finite-state transducers have sufficient power to express nearly all morphological phenomena, and the XFST toolkit (Beesley and Karttunen, 2003) has contributed to the practical adoption of this modeling approach. Recently, open-source tools have been released: in this paper, we used Foma (Hulden, 2009) to develop the Russian guesser.

Since some inflected forms have several possible analyses, there has been a great deal of work on selecting the intended one in context (Hakkani-Tür et al., 2000; Hajič et al., 2001; Habash and Rambow, 2005; Smith et al., 2005; Habash et al., 2009). Our disambiguation model is closely related to generative models used for this purpose (Hakkani-Tür et al., 2000).

Rule-based analysis is not the only approach to modeling morphology, and many unsupervised models have been proposed.[12] Heuristic segmentation approaches based on the minimum description length principle (Goldsmith, 2001; Creutz and Lagus, 2007; de Marcken, 1996; Brent et al., 1995) have been shown to be effective, and Bayesian model-based versions have been proposed as well (Goldwater et al., 2011; Snyder and Barzilay, 2008; Snover and Brent, 2001). In §3, we suggested a third way between rule-based approaches and fully unsupervised learning that combines the best of both worlds.

Morphological analysis or segmentation is crucial to the performance of several applications: machine translation (Goldwater and McClosky, 2005; Al-Haj and Lavie, 2010; Oflazer and El-Kahlout, 2007; Minkov et al., 2007; Habash and Sadat, 2006, *inter alia*), automatic speech recognition (Creutz et al., 2007), and syntactic parsing (Tsarfaty et al., 2010). Several methods have been proposed to integrate morphology into n-gram language models, including factored language models (Bilmes and Kirchhoff, 2003), discriminative language modeling (Arısoy et al., 2008), and more heuristic approaches (Monz, 2011).

Despite the fundamentally open nature of the lexicon (Heaps, 1978), there has been distressingly little attention to the general problem of open vocabulary language modeling problem (most applications make a closed-vocabulary assumption). The classic exploration of open vocabulary language modeling is Brown et al. (1992), which proposed the strategy of interpolating between word- and character-based models. Character-based language models are reviewed by Carpenter (2005). So-called *hybrid* models that model both words and sublexical units have become popular in speech recognition (Shaik et al., 2012; Parada et al., 2011; Bazzi, 2002). Open-vocabulary language language modeling has also recently been explored in the context of assistive technologies (Roark, 2009).

Finally, Pitman-Yor processes (PYPs) have become widespread in natural language processing since they are natural power-law generators. It has been shown that the widely used modified Kneser-Ney estimator (Chen and Goodman, 1998) for n-gram language models is an approximation of the posterior predictive distribution of a language model with hierarchical PYP priors (Goldwater et al., 2011; Teh, 2006).

7 Conclusion

We described a generative model which makes use of morphological analyzers to produce richer word distributions through sharing of statistical strength between stems. We have shown how it can be integrated into several models central to NLP applications and have empirically validated the effectiveness of these changes. Although this paper mostly focused on languages that are well studied and for which high-quality analyzers are available, our models are especially relevant in low-resource scenarios because they do not require disambiguated analyses. In future work, we plan to apply these techniques to languages such as Kinyarwanda, a resource-poor but morphologically rich language spoken in Rwanda. It is our belief that knowledge-rich models can help bridge the gap between low- and high-resource languages.

Acknowledgments

We thank Kemal Oflazer for making his Turkish language morphological analyzer available to us and Brendan O'Connor for gathering the Turkish tweets used in

[12] Developing a high-coverage analyzer can be a time-consuming process even with the simplicity of modern toolkits, and unsupervised morphology learning is an attractive problem for computational cognitive science.

the predictive text experiments. This work was sponsored by the U. S. Army Research Laboratory and the U. S. Army Research Office under contract/grant number W911NF-10-1-0533.

References

H. Al-Haj and A. Lavie. 2010. The impact of Arabic morphological segmentation on broad-coverage English-to-Arabic statistical machine translation. *Proc. of AMTA*.

Cyril Allauzen, Michael Riley, Johan Schalkwyk, Wojciech Skut, and Mehryar Mohri. 2007. OpenFst: A general and efficient weighted finite-state transducer library. In *Implementation and Application of Automata*, pages 11–23.

Ebru Arısoy, Brian Roark, Izhak Shafran, and Murat Saraçlar. 2008. Discriminative n-gram language modeling for Turkish. In *Proc. of Interspeech*.

Issam Bazzi. 2002. *Modelling out-of-vocabulary words for robust speech recognition*. Ph.D. thesis, MIT.

K.R. Beesley and L. Karttunen. 2003. *Finite-state morphology: Xerox tools and techniques*. CSLI, Stanford.

Jeff A. Bilmes and Katrin Kirchhoff. 2003. Factored language models and generalized parallel backoff. In *Proc. of NAACL*.

Ondřej Bojar and Magdalena Prokopová. 2006. Czech-English word alignment. In *Proc. of LREC*.

Michael R. Brent, Sreerama K. Murthy, and Andrew Lundberg. 1995. Discovering morphemic suffixes: A case study in MDL induction. In *Proceedings of the Fifth International Workshop on Artificial Intelligence and Statistics*.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, Robert L. Mercer, and Jennifer C. Lai. 1992. An estimate of an upper bound for the entropy of English. *Computational Linguistics*, 18(1):31–40.

P. F. Brown, V. J. D. Pietra, S. A. D. Pietra, and R. L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Bob Carpenter. 2005. Scaling high-order character language models to gigabytes. In *Proceedings of the ACL Workshop on Software*.

Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. WIT3: Web inventory of transcribed and translated talks. In *Proc. of EAMT*.

Stanley F. Chen and Joshua Goodman. 1998. An empirical study of smoothing techniques for language modeling. Technical Report TR-10-98, Harvard University.

Mathias Creutz and Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. *ACM Transactions on Speech and Language Processing*, 4(1).

M. Creutz, T. Hirsimäki, M. Kurimo, A. Puurula, J. Pylkkönen, V. Siivola, M. Varjokallio, E. Arisoy, M. Saraçlar, and A. Stolcke. 2007. Morph-based speech recognition and modeling of out-of-vocabulary words across languages. *ACM Transactions on Speech and Language Processing*, 5(1):3.

Carl G. de Marcken. 1996. *Unsupervised Language Acquisition*. Ph.D. thesis, MIT.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM Model 2. In *Proc. of NAACL*.

J. Goldsmith. 2001. Unsupervised learning of the morphology of a natural language. *Computational Linguistics*, 27(2):153–198.

S. Goldwater and D. McClosky. 2005. Improving statistical MT through morphological analysis. In *Proc. of EMNLP*.

Sharon Goldwater, Thomas L. Griffiths, and Mark Johnson. 2011. Producing power-law distributions and damping word frequencies with two-stage language models. *Journal of Machine Learning Research*, 12:2335–2382.

Nizar Habash and Owen Rambow. 2005. Arabic tokenization, part-of-speech tagging, and morphological disambiguation in one fell swoop. In *Proc. of ACL*.

Nizar Habash and Fatiha Sadat. 2006. Arabic preprocessing schemes for statistical machine translation. In *Proc. of NAACL*.

Nizar Habash, Owen Rambow, and Ryan Roth. 2009. MADA+TOKAN: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, POS tagging, stemming and lemmatization. In *Proceedings of the Second International Conference on Arabic Language Resources and Tools*.

Jan Hajič, P. Krbec, P. Květoň, K. Oliva, and V. Petrovič. 2001. Serial combination of rules and statistics. In *Proc. of ACL*.

D. Z. Hakkani-Tür, Kemal Oflazer, and G. Tür. 2000. Statistical morphological disambiguation for agglutinative languages. In *Proc. of COLING*.

Harold Stanley Heaps. 1978. *Information Retrieval: Computational and Theoretical Aspects*. Academic Press.

M. Hulden. 2009. Foma: a finite-state compiler and library. In *Proc. of EACL*.

Ronald M. Kaplan and Martin Kay. 1994. Regular models of phonological rule systems. *Computational Linguistics*, 20(3):331–378.

Lauri Karttunen and Kenneth R. Beesley. 2005. Twenty-five years of finite-state morphology. In *Inquiries into Words, Constraints and Contexts*, pages 71–83. CSLI.

Kimmo Koskenniemi. 1984. A general computational model for word-form recognition and production. In *Proc. of ACL-COLING*.

O. Lyaševskaya, I. Astaf'yeva, A. Bonch-Osmolovskaya, A. Garejšina, Y. Grišina, V. D'yačkov, M. Ionov, A. Koroleva, M. Kudrinskij, A. Lityagina, Y. Lučina, Y. Sidorova, S. Toldova, S. Savčuk, and S. Koval'. 2010. Ocenka metodov avtomatičeskogo analiza teksta: morfologičeskie parseri russkogo yazyka. *Komp'juternaya lingvistika i intellektual'nye texnologii (Computational linguistics and intellectual technologies)*.

Einat Minkov, Kristina Toutanova, and Hisami Suzuki. 2007. Generating complex morphology for machine translation. In *Proc. of ACL*.

Christof Monz. 2011. Statistical machine translation with local language models. In *Proc. of EMNLP*.

Kemal Oflazer and İlknur Durgar El-Kahlout. 2007. Exploring different representational units in English-to-Turkish statistical machine translation. In *Proc. of StatMT*.

K. Oflazer. 1994. Two-level description of Turkish morphology. *Literary and Linguistic Computing*, 9(2):137–148.

Carolina Parada, Mark Dredze, Abhinav Sethy, and Ariya Rastrow. 2011. Learning sub-word units for open vocabulary speech recognition. In *Proc. of ACL*.

Jim Pitman and Marc Yor. 1997. The two-parameter Poisson-Dirichlet distribution derived from a stable subordinator. *Annals of Probability*, 25(2):855–90.

Brian Roark. 2009. Open vocabulary language modeling for binary response typing interfaces. Technical Report CSLU-09-001, Oregon Health & Science University.

M. Ali Basha Shaik, David Rybach, Stefan Hahn, Ralf Schluüter, and Hermann Ney. 2012. Hierarchical hybrid language models for open vocabulary continuous speech recognition using wfst. In *Proc. of SAPA*.

M. Silfverberg, K. Lindén, and M. Hyvärinen. 2012. Predictive text entry for agglutinative languages using unsupervised morphological segmentation. In *Proc. of Computational Linguistics and Intelligent Text Processing*.

Noah A. Smith, David A. Smith, and Roy W. Tromble. 2005. Context-based morphological disambiguation with random fields. In *Proc. of EMNLP*.

Matt G. Snover and Michael R. Brent. 2001. A Bayesian model for morpheme and paradigm identification. In *Proc. of ACL*.

Benjamin Snyder and Regina Barzilay. 2008. Unsupervised multilingual learning for morphological segmentation. In *Proc. of ACL*.

Yee Whye Teh. 2006. A hierarchical Bayesian language model based on Pitman-Yor processes. In *Proc. of ACL*.

Reut Tsarfaty, Djamé Seddah, Yoav Goldberg, Sandra Kübler, Marie Candito, Jennifer Foster, Yannick Versley, Ines Rehbein, and Lamia Tounsi. 2010. Statistical parsing of morphologically rich languages: What, how and whither. In *Proc. of Workshop on Statistical Parsing of Morphologically Rich Languages*.

F. Tyers and M.S. Alperen. 2010. South-east european times: A parallel corpus of Balkan languages. In *Proceedings of the LREC workshop on Exploitation of multilingual resources and tools for Central and (South) Eastern European Languages*.

M. Talha Çakmak, Süleyman Acar, and Gülşen Eryiğit. 2012. Word alignment for English-Turkish language pair. In *Proc. of LREC*.