Priors in Bayesian Learning of Phonological Rules

Sharon Goldwater and Mark Johnson
Brown University

ACL SIGPHON'04
26 July, 2004

Why Morphology?

Understanding the morphology of a language can help us find lexical and syntactic relationships between words.

- walk ↔walked ↔walking
  - Stemming for information retrieval
  - Predicting subcategorization
  - Reducing sparse data for lexicalized parsing, MT, etc.
- walked ↔jumped ↔biked
  - Syntactic clustering/unsupervised POS tagging
  - Tagging unknown words

Why Phonology?

Morphological relationships may be obscured by phonological processes/spelling rules.

Without rules:

$$\begin{tikzcd}
\text{walk} \arrow[r, leftrightarrow] & \text{walk+ed} \arrow[r, leftrightarrow] \arrow[d, leftrightarrow] & \text{walk+ing} \arrow[d, leftrightarrow] \\
\text{bik+e} \arrow[r, leftrightarrow] & \text{bik+ed} \arrow[r, leftrightarrow] & \text{bik+ing}
\end{tikzcd}$$

With rule (*e* deletes before a vowel):

$$\begin{tikzcd}
\text{walk} \arrow[r, leftrightarrow] \arrow[d, leftrightarrow] & \text{walk+ed} \arrow[r, leftrightarrow] \arrow[d, leftrightarrow] & \text{walk+ing} \arrow[d, leftrightarrow] \\
\text{bike} \arrow[r, leftrightarrow] & \text{bike+ed} \arrow[r, leftrightarrow] & \text{bike+ing}
\end{tikzcd}$$

Goals of Current Work

- Learn phonological rules in order to improve and simplify an existing morphological analysis.
  - Complete system should be unsupervised: no annotated data or other resources.
  - Initial morphological analysis is obtained by running Linguistica (Goldsmith 2001) on raw text.
- Investigate the effects of different priors within a Bayesian framework.
  - Does straightforward Minimum Description Length work?

The Challenge of Unsupervised Learning

For our task, we want to consider models (grammars) with different numbers of parameters (stems, suffixes, rules).

- Models with different numbers of parameters are difficult to compare.
- How to balance model complexity against ability to fit the data?

Bayesian Learning

Given a dataset D, the goal of Bayesian learning is to find the model $\hat{M}$ where

$$
\begin{aligned}
\hat{M} &= \operatorname*{argmax}_{M} \overbrace{\Pr(M)}^{\text{prior}}\overbrace{\Pr(D|M)}^{\text{likelihood}} \\
&= \operatorname*{argmin}_{M} -\log\Pr(M) - \log\Pr(D|M)
\end{aligned}
$$

- Likelihood tells us how well the model fits the data.
- Prior can be used to induce a preference for simpler models.

Problem: How to specify such a prior?

Minimum Description Length

Idea: The more succinct a model (grammar) is, the more probable it is.

- $-\log \Pr(M)$ is the number of bits needed to specify M.
- $-\log \Pr(D|M)$ is the number of bits needed to encode the data using the specified grammar.
- So minimizing $-\log \Pr(M) - \log \Pr(D|M)$ is equivalent to minimizing the sum of the lengths of the grammar and the data.

MDL has been used successfully for various unsupervised learning tasks in morphology and phonology (Goldsmith 2001; Ellison 1993, 1994; de Marcken 1996; etc.).

Linguistica

Grammars considered by Linguistica consist of a set of signatures, which associate sets of stems and suffixes:

$$\left\{\begin{array}{c}\text{lift}\\ \text{jump}\\ \text{roll}\\ \text{walk}\\ \ldots\end{array}\right\} \times \left\{\begin{array}{c}\epsilon\\ -\text{s}\\ -\text{ed}\\ -\text{ing}\\ \ldots\end{array}\right\}$$

The probability of each word in the corpus is modeled using the probabilities of its signature, stem, and suffix:

$$\Pr(w = t + f) = \Pr(\sigma)\Pr(t|\sigma)\Pr(f|\sigma)$$

A Sample Linguistica Grammar

This toy grammar covers 34 words using 7 signatures and 12 stems:

$$
\begin{aligned}
\sigma_1 &= (\{\text{work, roll}\} \times \{\epsilon, \text{ed, ing, er}\}) \\
\sigma_2 &= (\{\text{bik, din}\} \times \{\text{e, ed, ing, er}\}) \\
\sigma_3 &= (\{\text{wait}\} \times \{\epsilon, \text{ed, er}\}) \\
\sigma_4 &= (\{\text{carr}\} \times \{\text{y, ied, ier}\}) \\
\sigma_5 &= (\{\text{carry}\} \times \{\epsilon, \text{ing}\}) \\
\sigma_6 &= (\{\text{bike, booth, worker}\} \times \{\epsilon, \text{s}\}) \\
\sigma_7 &= (\{\text{beach, match}\} \times \{\epsilon, \text{es}\})
\end{aligned}
$$

- Some relationships are missed (*bike* $\leftrightarrow$ *work*, *bikes* $\leftrightarrow$ *biked*)
- Some extra stems are proposed (*bik/bike*, *carr/carry*)

A Sample Linguistica Grammar

This toy grammar covers 34 words using 7 signatures and 12 stems:

$$\sigma_1 = (\{\text{work, roll}\}\times\{\epsilon, \text{ed, ing, er}\})$$
$$\sigma_2 = (\{\text{bik, din}\}\times\{\text{e, ed, ing, er}\})$$
$$\sigma_3 = (\{\text{wait}\}\times\{\epsilon, \text{ed, er}\})$$
$$\sigma_4 = (\{\text{carr}\}\times\{\text{y, ied, ier}\})$$
$$\sigma_5 = (\{\text{carry}\}\times\{\epsilon, \text{ing}\})$$
$$\sigma_6 = (\{\text{bike, booth, worker}\}\times\{\epsilon, \text{s}\})$$
$$\sigma_7 = (\{\text{beach, match}\}\times\{\epsilon, \text{es}\})$$

- Some relationships are missed (*bike* ↔ *work*, *bikes* ↔ *biked*)
- Some extra stems are proposed (*bik/bike*, *carr/carry*)

Adding Phonological Rules

Phonological rules have the form

$$a \rightarrow b \;/\; X_t y_t y_f X_f$$

- Transformation: a and b can be single characters or ϵ.
  - Ex: $e \;\rightarrow\; \epsilon$
- Context: $X_i \in \{C, V, \#\}, y_i$ a single character.
  - Ex: *bike* + *ing* has context *CeiC*.
- Transformation always occurs at the final stem position.
  - Ex: $e \rightarrow \epsilon \;/\; CeiC$ yields *bike* + *ing* → *biking*

Exceptions to Rules

When more than one rule applies in the same context, the most common rule is the default, other rules (including special **no-change** rule) are exceptions.

- Any stem requiring non-default rule must specify so in grammar.

- Exceptions add robustness against errors in initial morphology and allow for linguistic idiosyncracies (*debate/debatable* vs. *notice/noticeable*).

- Rules with too many exceptions will be rejected due to added grammar length.

A Sample Grammar with Rules

A grammar covering the same 34 words using 4 signatures, 10 stems, and 5 rules:

σ_1 = ({work, roll, dine, carry}×{ϵ, ed, er, ing})
σ_2 = ({bike}×{ϵ, ed, er, ing, s})
σ_3 = ({wait}×{ϵ, ed, er})
σ_4 = ({booth (r_5), worker, beach, match}×{ϵ, s})

r_1 = $e \rightarrow \epsilon \,/\, CeeC$
r_2 = $e \rightarrow \epsilon \,/\, CeiC$
r_3 = $y \rightarrow i \,/\, CyeC$
r_4 = $\epsilon \rightarrow e \,/\, Chs\#$
r_5 = **no-change** / $Chs\#$

A Sample Grammar with Rules

A grammar covering the same 34 words using 4 signatures, 10 stems, and 5 rules:

σ_1 = ($\{$work, roll, dine, carry$\}\times\{\epsilon$, ed, er, ing$\}$)
σ_2 = ($\{$bike$\}\times\{\epsilon$, ed, er, ing, s$\}$)
σ_3 = ($\{$wait$\}\times\{\epsilon$, ed, er$\}$)
σ_4 = ($\{$booth (r_5), worker, beach, match$\}\times\{\epsilon$, s$\}$)

r_1 = $e \rightarrow \epsilon \,/\, CeeC$
r_2 = $e \rightarrow \epsilon \,/\, CeiC$
r_3 = $y \rightarrow i \,/\, CyeC$
r_4 = $\epsilon \rightarrow e \,/\, Chs\#$
r_5 = **no-change** / $Chs\#$

A Sample Grammar with Rules

A grammar covering the same 34 words using 4 signatures, 10 stems, and 5 rules:

σ_1 = ({work, roll, dine, carry}$\times${ϵ, ed, er, ing})

σ_2 = ({bike}$\times${ϵ, ed, er, ing, s})

σ_3 = ({wait}$\times${ϵ, ed, er})

σ_4 = ({booth (r_5), worker, beach, match}$\times${ϵ, s})

r_1 = $e \rightarrow \epsilon \,/\, CeeC$

r_2 = $e \rightarrow \epsilon \,/\, CeiC$

r_3 = $y \rightarrow i \,/\, CyeC$

r_4 = $\epsilon \rightarrow e \,/\, Chs\#$

r_5 = **no-change** / $Chs\#$

Probabilistic Model

Given a stem and suffix, phonological rules are completely deterministic, so we still have

$$\Pr(w = t + f) = \Pr(\sigma)\Pr(t|\sigma)\Pr(f|\sigma).$$

Therefore the calculation of the likelihood term doesn't change.

But, how do we know which rules to add to the grammar?

Search Procedure

Goal: Explore a range of grammars similar to the initial grammar but with various phonological rules added, and choose the best one (according to some objective function).

1. Identify signatures with similar suffixes to infer possible transformations and contexts.
   - Ex: $\langle$*e.ed.ing*$\rangle$/$\langle\epsilon$*.ed.ing*$\rangle$ suggests $e \rightarrow \epsilon$ with contexts *XeeC* and *XeiC*
2. Try rules one at a time, evaluating and accepting or rejecting each.
   - Collapse signatures, introducing rule exceptions where necessary
   - Collapse stems such as *bik/bike* where necessary

Experiments

Ran our algorithm on text from the Penn WSJ Treebank, filtered to remove numbers, punctuation, acronyms, etc.

Initial morphological analysis produced by Linguistica on two different sized portions of text:

	Small	Large
Tokens	100k	888k
Types	11313	35631
Signatures	435	1634
Stems	8255	24529
Non-ϵ Stems	2363	7673

MDL Prior

First experiment: use the MDL prior described in Goldsmith (2001), modified to include phonological rules.

This prior is the number of bits needed to describe the grammar as follows:

- List suffixes and phonological rules and define a pointer to each.
- List stems and define a pointer to each. Each stem may also require a pointer to a phonological rule.
- List signatures, each containing
  - a list of pointers to suffixes.
  - a list of pointers to stems.

Results: MDL Prior

Algorithm considers many possible transformations, but only accepts a single one ($e \rightarrow \epsilon$). Why?

Consider results of adding $y \rightarrow i$ rules to grammar for large corpus:

	Initial Grammar	Change
# Signatures	1617	-10
# Stems	24374	-17
Likelihood:	6478490	+166
Grammar Length:	1335425	+520
Total:	7813915	+686

Results: MDL Prior

Adding $y \rightarrow i$ rules reduces the number of stems, but increases the length devoted to them!

	Initial Grammar	Change
# Signatures	1617	-10
# Stems	24374	-17
Likelihood:	6478490	+166
Grammar Length:	1335425	+520
Signatures, Suffixes	53933	-253
Stems	1280617	+493
Phonology	875	+279
Total:	7813915	+686

Results: MDL Prior

Adding $y \rightarrow i$ rules reduces the number of stems, but increases the length devoted to them:

$$\left\{\begin{array}{c}\text{alla}\\ \text{certif}\\ \text{dignif}\\ \text{disqualif}\\ \text{embod}\\ \text{empt}\\ \ldots\end{array}\right\} \times \left\{\begin{array}{c}-\text{y}\\ -\text{ied}\end{array}\right\} \Rightarrow \left\{\begin{array}{c}\text{allay}\\ \text{certify}\\ \text{dignify}\\ \text{disqualify}\\ \text{embody}\\ \text{empty}\\ \ldots\end{array}\right\} \times \left\{\begin{array}{c}\epsilon\\ -\text{ed}\end{array}\right\}$$

Results: MDL Prior

Even without added stem length, added length in phonology is more than reduction in overhead of signatures and suffixes:

	Initial Grammar	Change
# Signatures	1617	-10
# Stems	24374	-17
Likelihood:	6478490	+166
Grammar Length:	1335425	+520
Signatures, Suffixes	53933	-253
Stems	1280617	+493
Phonology	875	+279
Total:	7813915	+686

Encoding of signatures is simply too efficient.

Modified Prior

Using these insights, we redesigned the prior.

- Use a fixed cost for all stems.
- Increase the cost for signatures.
  - Now proportional to the sum of the lengths of the suffixes in the signature.
  - Collapsing signatures with many suffixes is better than collapsing signatures with few suffixes.

Rules Learned

In the large corpus, 22 rules with three types of transformations ($e \rightarrow \epsilon$, $\epsilon \rightarrow e$, and $y \rightarrow i$) were learned.

- 8 rules contained no exceptions.
  - $\epsilon \longrightarrow e \,/\, Vxs\#$ (*index + s* → *indexes*)
  - $y \longrightarrow i \,/\, CyeC$ (*certify + ed* → *certified*)

- 6 rules contained correctly analyzed exceptions.
  - *worthy + ness* → *worthiness* and *happy + ness* → *happiness*, but *dry + ness* → *dryness*)
  - Default for *Cos#* is **no-change** but some words require $\epsilon \rightarrow e$ (*potato + s* → *potatoes*).

Rules Learned

- 8 rules contained some misanalyzed exceptions.
  - $e \rightarrow \epsilon \, / \, CeeC$ lists an exception for *overse + er* $\rightarrow$ *overseer* (Should be reanalyzed as *oversee + er*).

In the small corpus, no $y \rightarrow i$ rules were learned due to the fact that no similar signatures attesting to these rules were found.

Final Results

7-10% of all non-ϵ stems were reanalyzed, reducing the number of signatures and stems in the grammars (which means more relationships between words).

	100k Corpus			888k Corpus		
	Morph	Phon	Diff	Morph	Phon	Diff
Signatures	435	404	-31	1634	1594	-57
Stems	8255	8186	-69	24529	24379	-150
Non-ϵ Stems	2363	2286	-77	7673	7494	-179
Rules		16	+16		22	+22

Lessons Learned

- MDL is an intuitive way to trade off data fit (likelihood) vs. generalization ability (prior), but presents difficulties for complex linguistic tasks.
  - Obvious encodings may not be linguistically appropriate (e.g. longer stems are worse than shorter stems).
  - When two different kinds of generalizations are possible (e.g. more signatures vs. fewer signatures and more rules), obvious encodings may not balance them correctly.
- Correcting for these problems allowed us to learn several major spelling rules of English and simplify the morphological analysis.

Future Work

We are currently implementing an integrated morphophonological learner with which we plan to

- Improve the search to consider more possible grammars.
- Expand the range of rules types allowed.
- Experiment with other languages.
- Try using phonological transcriptions.
- Investigate how to encode linguistic intuitions into a statistical prior.