

# Bayesian Learning of Probabilistic Language Models

by

Andreas Stolcke

## Abstract

The general topic of this thesis is the probabilistic modeling of language, in particular natural language. In probabilistic language modeling, one characterizes the strings of phonemes, words, etc. of a certain domain in terms of a probability distribution over all possible strings within the domain. Probabilistic language modeling has been applied to a wide range of problems in recent years, from the traditional uses in speech recognition to more recent applications in biological sequence modeling.

The main contribution of this thesis is a particular approach to the learning problem for probabilistic language models, known as *Bayesian model merging*. This approach can be characterize as follows.

- Models are built either in batch mode or incrementally from samples, by *incorporating* individual samples into a working model
- A uniform, small number of simple operators works to gradually transform an instance-based model to a generalized model that abstracts from the data.
- Instance-based parts of a model can coexist with generalized ones, depending on the degree of similarity among the observed samples, allowing the model to adapt to non-uniform coverage of the sample space.
- The generalization process is driven and controlled by a uniform, probabilistic metric: the Bayesian posterior probability of a model, integrating both criteria of goodness-of-fit with respect to the data and a notion of model simplicity ('Occam's Razor').

The Bayesian model merging framework is instantiated for three different classes of probabilistic models: Hidden Markov Models (HMMs), stochastic context-free grammars (SCFGs), and simple probabilistic attribute grammars (PAGs). Along with the theoretical background, various applications and case studies are presented, including the induction of multiple-pronunciation word models for speech recognition (with HMMs), data-driven learning of syntactic structures (with SCFGs), and the learning of simple sentence-meaning associations from examples (with PAGs).

Apart from language learning issues, a number of related computational problems involving probabilistic context-free grammars are discussed. A version of Earley's parser is presented that solves the standard problems associated with SCFGs efficiently, including the computation of sentence probabilities and sentence prefix probabilities, finding most likely parses, and the estimation of grammar parameters.

Finally, we describe an algorithm that computes  $n$ -gram statistics from a given SCFG, based on solving linear systems derived from the grammar. This method can be an effective tool to transform part of

the probabilistic knowledge from a structured language model into an unstructured low-level form for use in applications such as speech decoding. We show how this problem is just an instance of a larger class of related ones (such as average sentence length or derivation entropy) that are all solvable with the same computational technique.

An introductory chapter tries to present a unified view of the various model types and algorithms found in the literature, as well as issues of model learning and estimation.

---

Prof. Jerome A. Feldman, Dissertation Chair

# Contents

<b>List of Figures</b>	<b>vii</b>
<b>List of Tables</b>	<b>viii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Overview	1
1.2 Structural Learning of Probabilistic Grammars	2
1.2.1 Probabilistic finite-state models	3
1.2.2 The Miniature Language Learning ( $L_0$ ) Task	3
1.3 Miscellaneous topics	6
1.4 Bibliographical Note	6
<b>2 Foundations</b>	<b>7</b>
2.1 Preliminaries	7
2.2 Probabilistic Language Models	7
2.2.1 Interpretation of probabilities	8
2.2.2 An example: $n$ -gram models	8
2.2.3 Probabilistic grammars as random string generators	9
2.2.4 Multinomial distributions	9
2.2.5 Parameter estimation	10
2.2.6 Likelihood and cross-entropy	11
2.3 Grammars with hidden variables	12
2.3.1 Mixture grammars	13
2.3.2 Expectation-Maximization	13
2.3.3 Viterbi derivations and approximate EM	14
2.3.4 Hidden Markov Models	15
2.3.5 Stochastic Context-free Grammars	16
2.4 Levels of Learning and Model Merging	17
2.4.1 Beyond parameter estimation	17
2.4.2 Model merging	17
2.4.3 A curve-fitting example	18
2.4.4 Knowing when to stop	18
2.5 Bayesian Model Inference	20
2.5.1 The need for inductive bias	20
2.5.2 Posterior probabilities	21
2.5.3 Bayesian Model Merging	21
2.5.4 Minimum Description Length	22
2.5.5 Structure vs. parameter priors	23
2.5.6 Description Length priors	24

2.5.7	Posteriors for grammar structures	26
<b>3</b>	<b>Hidden Markov Models</b>	<b>27</b>
3.1	Introduction and Overview	27
3.2	Hidden Markov Models	28
3.2.1	Definitions	28
3.2.2	HMM estimation	29
3.2.3	Viterbi approximation	30
3.3	HMM Merging	31
3.3.1	Likelihood-based HMM merging	31
3.3.2	An example	32
3.3.3	Priors for Hidden Markov Models	34
3.3.4	Why are smaller HMMs preferred?	37
3.3.5	The algorithm	39
3.4	Implementation Issues	40
3.4.1	Efficient sample incorporation	40
3.4.2	Computing candidate merges	41
3.4.3	Model evaluation using Viterbi paths	41
3.4.4	Global prior weighting	45
3.4.5	Search issues	45
3.5	Related Work	46
3.5.1	Non-probabilistic finite-state models	47
3.5.2	Bayesian approaches	47
3.5.3	State splitting algorithms	47
3.5.4	Other probabilistic approaches	48
3.6	Evaluation	49
3.6.1	Case studies of finite-state language induction	49
3.6.2	Phonetic word models from labeled speech	63
3.6.3	Multiple pronunciation word models for speech recognition	72
3.7	Conclusions and Further Research	74
<b>4</b>	<b>Stochastic Context-free Grammars</b>	<b>75</b>
4.1	Introduction and Overview	75
4.2	Stochastic Context-free Grammars	76
4.2.1	Definitions	76
4.2.2	SCFG estimation	78
4.2.3	Viterbi parses	79
4.3	SCFG Merging	79
4.3.1	Sample incorporation and merging operators	79
4.3.2	An example	83
4.3.3	Bracketed samples	85
4.3.4	SCFG priors	86
4.3.5	Search strategies	88
4.3.6	Miscellaneous	90
4.4	Related Work	91
4.4.1	Bayesian grammar learning by enumeration	91
4.4.2	Merging and chunking based approaches	91
4.4.3	Cook's Grammatical Inference by Hill Climbing	92
4.5	Evaluation	93
4.5.1	Formal language benchmarks	93
4.5.2	Natural language syntax	96
4.5.3	Sample ordering	101

4.5.4	Summary and Discussion	103
<b>5</b>	<b>Probabilistic Attribute Grammars</b>	<b>104</b>
5.1	Introduction	104
5.2	Probabilistic Attribute Grammars	104
5.2.1	Definitions	105
5.2.2	An example	107
5.2.3	PAG estimation	108
5.3	PAG Merging	109
5.3.1	Sample incorporation	109
5.3.2	Nonterminal merging and chunking	110
5.3.3	Feature operators	111
5.3.4	Efficient search for feature operations	112
5.3.5	PAG Priors	114
5.4	Experiments	115
5.4.1	$L_0$ examples	115
5.4.2	Syntactic constraints imposed by attributes	116
5.5	Limitations and Extensions	118
5.5.1	More expressive feature constraints	118
5.5.2	Hierarchical features	119
5.5.3	Trade-offs between context-free and feature descriptions	120
5.6	Summary	120
<b>6</b>	<b>Efficient parsing with Stochastic Context-free Grammars</b>	<b>122</b>
6.1	Introduction	122
6.2	Overview	124
6.3	Earley Parsing	124
6.4	Probabilistic Earley Parsing	127
6.4.1	Stochastic context-free grammars	127
6.4.2	Earley paths and their probabilities	129
6.4.3	Forward and inner probabilities	131
6.4.4	Computing forward and inner probabilities	133
6.4.5	Coping with recursion	134
6.4.6	An example	139
6.4.7	Null productions	139
6.4.8	Existence of $R_L$ and $R_U$	143
6.4.9	Complexity issues	145
6.4.10	Summary	147
6.5	Extensions	147
6.5.1	Viterbi parses	147
6.5.2	Rule probability estimation	149
6.5.3	Parsing bracketed inputs	152
6.5.4	Robust parsing	154
6.6	Implementation Issues	158
6.6.1	Prediction	158
6.6.2	Completion	158
6.6.3	Efficient parsing with large sparse grammars	159
6.7	Discussion	160
6.7.1	Relation to finite-state models	160
6.7.2	Online pruning	161
6.7.3	Relation to probabilistic LR parsing	162
6.7.4	Other related work	163

6.7.5	A simple typology of SCFG algorithms . . . . .	164
6.8	Summary . . . . .	165
6.9	Appendix: LR item probabilities as conditional forward probabilities . . . . .	166
<b>7</b>	<b><i>N</i>-grams from Stochastic Context-free Grammars</b>	<b>168</b>
7.1	Introduction . . . . .	168
7.2	Background and Motivation . . . . .	169
7.3	The Algorithm . . . . .	171
7.3.1	Normal form for SCFGs . . . . .	171
7.3.2	Probabilities from expectations . . . . .	171
7.3.3	Computing expectations . . . . .	172
7.3.4	Computing prefix and suffix probabilities . . . . .	173
7.3.5	<i>N</i> -grams containing string boundaries . . . . .	174
7.4	Efficiency and Complexity Issues . . . . .	174
7.5	Consistency of SCFGs . . . . .	175
7.6	Experiments . . . . .	176
7.7	Summary . . . . .	178
7.8	Appendix: Related Problems . . . . .	179
7.8.1	Expected string length . . . . .	179
7.8.2	Derivation entropy . . . . .	179
7.8.3	Expected number of nonterminal occurrences . . . . .	180
7.8.4	Other grammar types . . . . .	180
<b>8</b>	<b>Future directions</b>	<b>181</b>
8.1	Formal characterization of learning dynamics . . . . .	181
8.2	Noisy samples . . . . .	182
8.3	More informative search heuristics and biases . . . . .	182
8.4	Induction by model specialization . . . . .	182
8.5	New applications . . . . .	183
8.6	New types of probabilistic models . . . . .	183
	<b>Bibliography</b>	<b>184</b>