

Linguistic Knowledge and Empirical Methods in Speech Recognition

Andreas Stolcke

■ Automatic speech recognition is one of the fastest growing and commercially most promising applications of natural language technology. The technology has achieved a point where carefully designed systems for suitably constrained applications are a reality. Commercial systems are available today for such tasks as large-vocabulary dictation and voice control of medical equipment. This article reviews how state-of-the-art speech-recognition systems combine statistical modeling, linguistic knowledge, and machine learning to achieve their performance and points out some of the research issues in the field.

Automatic speech recognition (ASR) is one of the fastest growing and commercially most promising applications of natural language technology. Speech is the most natural communicative medium for humans in many situations, including applications such as giving dictation; querying database or information-retrieval systems; or generally giving commands to a computer or other device, especially in environments where keyboard input is awkward or impossible (for example, because one's hands are required for other tasks).

We are far from solving the ASR problem fully, and to the extent that human performance requires solving related AI-complete problems (such as humanlike natural language understanding), we might never fully achieve this goal. However, the technology has made steady progress over the past several decades and has achieved a point where carefully designed systems for suitably constrained applications are a reality. Commercial systems are available today for diverse tasks such as large-vocabulary dictation, voice control of medical equipment, and stock trading over the telephone.

Progress is not only evident in real-world applications. The technology has been constantly improving, as measured by various benchmarks administered by the U.S. government, in particular by the Defense Advanced Research Projects Agency (DARPA). The commonly used performance metric in these evaluations is the recognition *word error rate* (WER), defined as the ratio of the number of incorrectly recognized (or unrecognized) words to the total number of actually spoken words. The difficulty of an ASR task can broadly be characterized along a number of dimensions. The recognition task becomes increasingly more difficult along these dimensions, as given here:

First is the size of the vocabulary: small (2–100 words, for example, for a voice-menu system), medium (several 100s to 1,000s of words, for example, for a database-retrieval task), and large (several 10,000s of words, as in a general dictation task).

Second is the speaking style: read speech versus planned speech versus spontaneous speech.

Third is the language domain: task oriented and constrained (such as database query) versus open and unconstrained (nontask oriented, human-to-human conversation). Less constrained language tends to also have larger vocabulary and features more spontaneous speech.

Fourth is speaker specificity: speaker dependent (system trained on test speaker) versus speaker independent. (A related dimension is native versus nonnative speech.)

Fifth is channel quality: high bandwidth (encompassing the full frequency range of human speech) to low bandwidth (for example, telephone) as well as the amount of distortion.

Sixth is acoustic environment: less versus

Task	Vocabulary	Style	Channel	Acoustics	Word Error Rate (%)
Air-travel information system	2,000	Spontaneous, human to machine	High bandwidth	Clean	2.1
North American business news	60,000	Read	High Bandwidth	Clean	6.6
Broadcast news	60,000	Various	Various	Various	27.1
Switchboard	23,000	Spontaneous, conversational	Telephone	Clean	35.1

Table 1. State-of-the-Art Performance on Defense Advanced Research Projects Agency Speech-Recognition Tasks.

Performance is given as the word error rate (WER) achieved by the best system in the most recent evaluation of the tasks.

more background noise as well as the type of background noise (for example, stationary, nonhuman noise versus background speech and crosstalk by other speakers).

Table 1 summarizes recent results from DARPA evaluations on a number of tasks spanning several of the dimensions given here. All tasks involve speaker-independent recognition of American English. The Air Travel Information System (ATIS) task involves a human retrieving flight information from a database. The North American business news (NABN) corpus has speakers carefully reading articles from newspapers and wire services. In the broadcast news task, the goal is to automatically transcribe radio and television audio containing a mix of speaking styles, often interlaced with nonspeech. The switchboard (SWB) corpus contains spontaneous, casual speech from human-to-human telephone conversations. As shown in the table, word error rates increase as the speech becomes less constrained and acoustic conditions less controlled.

Interestingly, one dimension that does not seem to affect ASR performance inherently is the choice of the language itself. Although it is difficult to control the choice of language given the historical English bias in much ASR research, recent benchmarks on a range of non-English languages (Spanish, Arabic, Mandarin Chinese, Japanese) under similar conditions showed comparable performance (DARPA 1997). Although these languages dif-

fer widely in their phonology, morphology, and syntax, both from each other and from English, similar results were achieved using essentially identical technology (some of which is outlined in this article) simply by retraining the system on the appropriate training data. This result is encouraging because it demonstrates the power and flexibility of data-driven approaches.

A Statistical Paradigm for Speech Recognition

A state-of-the-art speech recognizer is best understood as a statistical pattern classifier (Duda and Hart 1973). Given an acoustic waveform A , its goal is to find the word sequence W that best matches A . *Best match* is defined in probabilistic terms; that is, the recognizer aims to find the words W that have the highest posterior probability $P(W|A)$ given A (Bahl, Jelinek, and Mercer 1983).

I gloss over the many algorithmic and engineering issues involved in building a working ASR system (especially concerning the search for the best hypothesis) and instead focus on some of the models used to compute the probabilities used to rank alternative recognition hypotheses. Virtually all models use a combination of linguistic knowledge and data-driven machine-learning techniques to achieve their goals. Speech recognition is therefore inherently empirical and corpus based in nature.

Finding the right combination of built-in structure and trainability is one of the keys to good performance and the feasibility of the overall system.

The standard procedure to compute the probability $P(W | A)$ is to use Bayes's law to decompose this posterior probability into a prior probability $P(W)$ for the word sequence W under consideration and an acoustic likelihood $P(A | W)$ plus a normalization term

$$P(W | A) = P(W)P(A | W) / P(A) .$$

The denominator does not depend on W and can therefore be ignored when comparing different hypotheses.

The model to compute $P(W)$ is called the *language model*; it is a probabilistic grammar expressing the prior probabilities of all possible word sequences that the recognizer can potentially recognize. The prior probability of an utterance depends not just on the language but also crucially on the application domain, the speaker, and the context of the utterance; a good language model should therefore be conditioned on all these. As a simple example, the word *you* has probability $\approx .024$ of occurring at any given position in the switchboard (telephone conversation) corpus, but in the NABN corpus, the probability is only $\approx .00086$.

The acoustic likelihood $P(A | W)$ characterizes the match between acoustic observation and hypothesized words and is computed by the *acoustic model*. The number of possible acoustic observations A and word sequences W is, for all purposes, infinite. Therefore, to be practical, an acoustic model must rely on a hierarchical decomposition into models of smaller units. This decomposition usually follows the levels of representation identified in linguistics. Thus, a typical acoustic model will first decompose a word sequence W into individual words, each with its own model, so that the same word occurring in different contexts is represented identically. Words are further decomposed into phones, and phones are, in turn, modeled by subphonetic states corresponding to onset, middle, and ends of their realizations. Later in this brief overview, we take a closer look at one of these modeling levels, namely, that of phone sequences. For lack of space, I do not touch on the lower-level components of acoustic modeling, except to say that they rely heavily on data-driven methods. For example, it turns out that the best approach to group subphonetic units into classes (something that is necessary to counteract the sparseness of training data) is through clustering algorithms driven by information-theoretic measures of model fit (Digalakis and Murveit 1994).

I now examine some of the problems of both language and acoustic models in more detail and outline how empirical methods can be brought to bear in each case. Naturally, I can only scratch the surface in this article; for a comprehensive account of speech-recognition methods, the reader is referred to Rabiner and Juang (1993). A recent book that focuses on the statistical and data-driven aspects of speech modeling is Jelinek (1997).

Language Modeling

As mentioned previously, the job of the language model in a speech recognizer is to assess the prior probabilities $P(W)$ of potential recognized word sequences. This section describes the most popular approaches to this problem as well as some of the research issues.

N-Gram Models

With the basic axioms of probability, the joint probability of the word sequence can be expressed as a product of word probabilities, each conditioned on all preceding words. The probability of "The cat is on the mat" becomes

$$\begin{aligned} P(\text{the cat is} \\ \text{on the mat}) &= P(\text{the} | \langle s \rangle) \times \\ &P(\text{cat} | \langle s \rangle \text{ the}) \times \\ &P(\text{is} | \langle s \rangle \text{ the cat}) \times \\ &P(\text{on} | \langle s \rangle \text{ the cat is}) \times \\ &P(\text{the} | \langle s \rangle \text{ the cat is on}) \times \\ &P(\text{mat} | \langle s \rangle \text{ the cat is on the}) \times \\ &P(\langle /s \rangle | \langle s \rangle \text{ the cat is} \\ &\text{on the mat}) . \end{aligned}$$

Here we use the tags $\langle s \rangle$ and $\langle /s \rangle$ to denote the beginning and end of sentences, respectively.

Although such a decomposition of the joint probability is exact, it introduces far too many model parameters (one for each conditional probability) to be practical. The solution used in the vast majority of current speech systems is to approximate the conditional word probabilities by truncating the history of each word to one or two tokens:

$$\begin{aligned} P(\text{the cat is on the mat}) &\approx P(\text{the} | \langle s \rangle) \times \\ &P(\text{cat} | \langle s \rangle \text{ the}) \times \\ &P(\text{is} | \text{the cat}) \times \\ &P(\text{on} | \text{cat is}) \times \\ &P(\text{the} | \text{is on}) \times \\ &P(\text{mat} | \text{on the}) \times \\ &P(\langle /s \rangle | \text{the mat}) . \end{aligned}$$

Such a model is called an *n-gram model* because it incorporates the statistics of N -tuples of words. The most commonly used versions are *bigram* ($N = 2$) and *trigram* ($N = 3$) models. N -gram models have a large number of parameters, essentially one conditional probability for

Automatic speech recognition (ASR) is one of the fastest growing and commercially most promising applications of natural language technology.

*Speech
recognition
is therefore
inherently
empirical and
corpus based
in nature.*

each N -tuple of words observed in the training corpus. N -gram model parameters are estimated by counting the occurrences of word N -tuples in a training corpus. A naive estimate for the probability $P(w | h)$ is the relative frequency of word w in the context, or “history,” h .

The relative frequency estimator suffers from the problem that even with large amounts of data, one cannot expect to see all word-history combinations in the training corpus, and all such combinations would receive probability zero. For example, after training on 2 million words of switchboard conversations, about 10 percent of the two-word combinations (bigrams) found in additional text from the same corpus remain novel. Fortunately, a number of effective *smoothing methods* exist that estimate nonzero probabilities for unseen word combinations in a principled, data-driven manner (Church and Gale 1991).

N -gram models disregard linguistic structure completely; for example, they do not try to capture syntactic long-distance relationships. In spite of this, they turn out to be hard to beat as statistical models (Jelinek 1991) because they capture local word cooccurrence constraints effectively. Their good performance, combined with practical advantages such as ease of training and computational simplicity, make n -grams the models of choice for most ASR systems. Still, there are problems with n -gram models that current research in language modeling is trying to overcome.

N -gram models are highly tuned to the sub-language they were trained on, which makes them hard to beat, assuming that sufficient training material matching the target domain of application is available. Conversely, n -gram statistics are not easily transferred or adapted from one domain to another, making it necessary to collect new training data for each new application.

There are some techniques to improve n -gram models in the absence of sufficient training data. For example, instead of modeling word n -grams, one can model the cooccurrence of word classes, which can be obtained from syntactic, semantic, or other criteria. Such *class n -gram language models* decompose a conditional word probability $P(w | h)$ into a product of the word-class probability $P(C(w) | h)$ and the probability $P(w | C(w))$ of the word given its class $C(w)$. Word histories h can be represented by the preceding words or their classes, and both types of probability can again be estimated by simple counting of training-set occurrences plus a suitable smoothing scheme.

To see the benefit of this approach, let’s assume that we have a word-class *fruit* and that

$C(\textit{banana}) = C(\textit{avocado}) = \textit{fruit}$. Based on the previous factorization, we can now infer the probability $P(\textit{avocado} | \textit{the ripe})$ even though we only observed *the ripe banana*. Because *banana* and *avocado* share the same class, observing one contributes to estimating the probability of the other using the shared probability $P(\textit{fruit} | \textit{the ripe})$.

The word-class-based model assumes that the words in a class are distributed identically once we condition on the class. This assumption is justified in restricted domains (for example, the ATIS task) where one can identify sets of words whose members are distributed almost identically (such as airline names, city names, days of the week). Alternatively, *automatic clustering algorithms* can be used to find word classes that optimize the overall language model quality as measured by its entropy (Brown et al. 1992). One drawback of automatic methods is that they only operate on words found in the training data, but hand-designed classes can increase the coverage of the language model to previously unseen words.

A general type of language model is obtained by using a decision tree as a predictor of $P(w | h)$ (Bahl et al. 1989). A decision tree is trained on data to predict the next word w based on a large number of features describing the word history h . The features include all the information found in standard language models, such as the identity and syntactic classes or words in the history. This approach gives the tree model the potential to find novel and more effective predictors based on the word history. Still, results to date have not shown a significant improvement over the standard word n -gram model.

Language Models Based on Phrase Structure

There have been various efforts to supplant n -gram models with more linguistically motivated language models. A considerable body of work is based on *probabilistic context-free grammars* (PCFGs), a generalization of context-free grammars that assigns probabilities to each rewrite rule (Lari and Young 1991; Jelinek, Lafferty, and Mercer 1990). A rule probability such as $P(NP \rightarrow Det N) = 0.78$ means that 78 percent of the noun phrases generated by the grammar consist of determiner-noun pairs. To train such grammars, one either needs a parsed corpus (a tree bank) or an iterative algorithm that repeatedly parses the training data and then reestimates the rule probabilities accordingly.

Unfortunately, no effective algorithms are known that learn realistic phrase structure grammars completely from scratch, that is,

without an initial set of grammar rules or tree-bank parses. A more fundamental problem with traditional PCFGs, as opposed to n -gram language models, is that the nonlexical, context-free structure of a grammar prevents it from capturing the local dependencies between words that predict most of their distribution. Most recent work aimed at leveraging phrase structure for language modeling tries to combine the advantages of standard n -gram models with some incremental benefit based on linguistic structure. For example, one can supplement the n -gram statistics from insufficient training data with n -gram statistics induced by a corpus-trained PCFG whose rules are written by hand (Jurafsky et al. 1995). A traditional robust parser can be used to parse a recognition hypothesis into phrases, after which both the phrase sequence and the word sequences in each phrase are evaluated according to n -gram models (Moore et al. 1995). Alternatively, word distributions can be modeled in terms of lexical cooccurrence along a syntactic dimension, such as subject noun-verb or verb-object noun, replacing or adding to the juxtaposed cooccurrences modeled in n -grams. This last approach can be illustrated by the example “the dog who chased the cat barked.” For predicting the word *barked*, we want to refer to its subject *dog* and not to the immediate predecessor *cat*, as an n -gram model would. In general, we want to find those words in the history that are in a predicate-argument relationship with the word to be predicted and then model their cooccurrence statistics. Although the idea is straightforward, the details are complex because the underlying syntactic structure is usually highly ambiguous, and the probabilistic formulation has to handle the multiple syntactic relationships each word is involved in (Chelba et al. 1997).

Unfortunately, none of these techniques based on linguistic phrase structure have to date been able to surpass the performance of the simple n -gram models in large vocabulary, open-domain ASR systems, although several of them give improvements on more constrained tasks.

Nonprobabilistic Knowledge Sources

All language-modeling approaches described to this point fit in the probabilistic framework outlined initially. These language models provide prior probabilities to be combined with acoustic model likelihoods for an estimate of the posterior probability of a recognition hypothesis.

A somewhat more general approach views the acoustic and language models as two of

possibly many knowledge sources that get to “vote” on the recognition hypotheses to determine a winner, or most likely hypothesis (Ostendorf et al. 1991). Each knowledge source contributes a score for each hypothesis, and the scores are weighted and added up to determine the overall winner. (Probabilistic models fit nicely into this scheme as a special case because the logarithms of probabilities can be used as scores, such that an additive voting scheme yields results that are equivalent to the probabilistic framework.) The weights of the voting function correspond to the relative importance of the various knowledge sources; they can be determined empirically by picking parameter values that give the best recognition accuracy on a held-out data set.

The *knowledge source combination approach* opens the door to a host of additional information sources that might help discriminate correct from incorrect recognition results. If the recognizer serves as the front end to a natural language understanding, translation, or information-extraction system, these back-end systems can contribute scores that correspond to the interpretability of hypotheses. To the extent that incorrect hypotheses are less likely to make sense to the back end, this approach will improve ASR performance. Coverage of the semantic component is usually not perfect, so it is important that interpretability not be a hard constraint on recognition. It has been shown that scoring hypotheses for their coverage by standard, nonprobabilistic natural language programming components can improve ASR accuracy (Rayner et al. 1994).

Pronunciation Modeling

As mentioned earlier, the acoustic model of an ASR system actually consists of a hierarchy of models of successively smaller temporal scope. The details of this hierarchy vary, but most systems use some form of intermediate representation corresponding to pronunciations represented as phone sequences. As before, let W be a word sequence and A a sequence of acoustic features corresponding to the waveform. We assume that each word corresponds to a sequence of phones, forming a joint phone sequence R . For example, the word sequence

the cat is on the mat

would correspond to a phone sequence

dh ax k ae t ih z aa n dh ax m ae t .

(We are using an ASCII encoding of the English phone set known as the *ARPAbet*. For example, *dh* denotes the voiced *th* sound.)

The complete acoustic likelihood $P(A | W)$ is

The solution used in the vast majority of current speech systems is to approximate the conditional word probabilities by truncating the history of each word to one or two tokens.

... trainability is the crucial element that makes ASR technology applicable to a variety of domains, languages, and environmental conditions.

now obtained as the product of a pronunciation probability $P(R | W)$ and the combined phone likelihood $P(A | R)$. The total phone likelihood is factored into the contributions of individual phones. Because the number of distinct phones is much smaller than the number of words, this decomposition greatly reduces the number of required acoustic model parameters (at the expense of some modeling accuracy because it is assumed that a phone sounds the same independent of the word it appears in).

This decomposition of word models into phone models also requires estimating the pronunciation probability $P(R | W)$. Most ASR systems today use a pronunciation dictionary to map each word to a few (mostly just one) possible pronunciations and assume that the pronunciation for the whole word sequence is the concatenation of the individual word pronunciations. Because pronunciation dictionaries are, for the most part, written by hand, the pronunciation model incorporates significant hand-coded linguistic knowledge. Here, too, research aims to develop automatic learning algorithms that replace hand-coded models with more accurate models induced from data. For example, in a first pass over the training data, the recognizer can be allowed to hypothesize multiple alternative phone sequences for each word. The observed pronunciation sequences for a word are then fed to a merging algorithm that induces probabilistic finite-state models, both pruning out unlikely variants and including likely, but unobserved, variants (Wooters and Stolcke 1994). A different approach involves building a decision tree that derives actual pronunciations from those found in the dictionary by predicting where

phones are changed, deleted, or inserted as a function of their phonetic context (Riley 1991). For example, in the earlier pronunciation example, the decision tree might learn that the *t* sound in *cat* when followed by an unstressed vowel can turn into the flapped consonant that sounds closer to a *d*. The tree would also learn the probability with which this and other changes apply, so that the overall pronunciation probability $P(R | W)$ can be computed.

Revisiting Model Decomposition

As we saw in the beginning, both language and acoustic models crucially rely on decomposing their event spaces (word sequences, acoustic observation sequences) into smaller units to cope with the large number of possible events and the relative sparseness of training data. The units on which model decomposition is based are largely informed by traditional linguistic concepts, such as words and phones. It is questionable whether such preconceived units are optimal. Yet another direction for ongoing research seeks to identify better units for modeling. One approach that has been tried is to search for collocations of multiple words that should be treated as a single unit. This approach obviously improves language models, but it also improves acoustic models because certain word combinations exhibit idiosyncratic pronunciations. For example, *going to* is more often than not pronounced *gonna* in casual speech, which is most easily captured by treating *going to* as a single word-level unit in the pronunciation model.

Conclusions and Future Directions

We have seen how modern speech-recognition systems model linguistic entities at multiple levels (sentences, words, phones, and so on) using various statistical techniques. The parameters of these models are usually trained on data, but their structure is largely determined by linguistic insight. Trainability is often achieved through severe simplifying assumptions about the statistical independence of the phenomena (word probabilities depend only on the last two words, phone realizations are independent of the words they occur in, and so on). However, trainability is the crucial element that makes ASR technology applicable to a variety of domains, languages, and environmental conditions.

Most current research on speech models is concerned with the right combination of built-

in structure and data-driven learning. In some areas, such as pronunciation modeling, the goal is to replace hand-coded models with more accurate ones patterned after the actually occurring data. In other cases, such as language modeling, the strategy is to inject the right structural constraints into existing unstructured models to make more efficient use of the available (and often scarce) training data. Success on both fronts will be crucial in making ASR technology ever more accurate, more robust, and more ubiquitous.

References

- Bahl, L. R.; Jelinek, F.; and Mercer, R. L. 1983. A Maximum Likelihood Approach to Continuous Speech Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 5(2): 179–190.
- Bahl, L. R.; Brown, P. F.; de Souza, P. V.; and Mercer, R. L. 1989. A Tree-Based Statistical Language Model for Natural Language Speech Recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 37(7): 1001–1008.
- Brown, P. F.; Della Pietra, V. J.; deSouza, P. V.; Lai, J. C.; and Mercer, R. L. 1992. Class-Based n -Gram Models of Natural Language. *Computational Linguistics* 18(4): 467–479.
- Chelba, C.; Engle, D.; Jelinek, F.; Jimenez, V.; Khudanpur, S.; Mangu, L.; Printz, H.; Ristad, E.; Rosenfeld, R.; Stolcke, A.; and Wu, D. 1997. Structure and Performance of a Dependency Language Model. In Proceedings of the Fifth European Conference on Speech Communication and Technology, Volume 5, 2775–2778. Grenoble, France: European Speech Communication Association.
- Church, K. W., and Gale, W. A. 1991. A Comparison of the Enhanced Good-Turing and Deleted Estimation Methods for Estimating Probabilities of English Bigrams. *Computer Speech and Language* 5:19–54.
- DARPA. 1997. Large Vocabulary Conversational Speech Recognition (LVCSR) Hub-5 Workshop. Washington, D.C.: Defense Advanced Research Projects Agency.
- Digalakis, V., and Murveit, H. 1994. GENOMES: An Algorithm for Optimizing the Degree of Tying in a Large-Vocabulary Hidden Markov Model-Based Speech Recognizer. In Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing, 537–540. Washington, D.C.: IEEE Computer Society.
- Duda, R. O., and Hart, P. E. 1973. *Pattern Classification and Scene Analysis*. New York: Wiley.
- Jelinek, F. 1997. *Statistical Methods for Speech Recognition*. Cambridge, Mass.: MIT Press.
- Jelinek, F. 1991. Up from Trigrams! The Struggle for Improved Language Models. In Proceedings of the Second European Conference on Speech Communication and Technology, 1037–1040. Grenoble, France: European Speech Communication Association.
- Jelinek, F.; Lafferty, J. D.; and Mercer, R. L. 1992. Basic Methods of Probabilistic Context-Free Grammars. In *Speech Recognition and Understanding. Recent Advances, Trends, and Applications, Volume F75*, eds. Pietro Laface and Renato De Mori, 345–360. NATO ASI Series. Berlin: Springer Verlag.
- Jurafsky, D.; Wooters, C.; Segal, J.; Stolcke, A.; Fosler, E.; Tajchman, G.; and Morgan, N. 1995. Using a Stochastic Context-Free Grammar as a Language Model for Speech Recognition. In Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing, Volume 1, 189–192. Washington, D.C.: IEEE Computer Society.
- Lari, K., and Young, S. J. 1991. Applications of Stochastic Context-Free Grammars Using the Inside-Outside Algorithm. *Computer Speech and Language* 5:237–257.
- Moore, R.; Appelt, D.; Dowding, J.; Gawron, J. M.; and Moran, D. 1995. Combining Linguistic and Statistical Knowledge Sources in Natural Language Processing for ATIS. In Proceedings of the ARPA Spoken-Language Systems Technology Workshop, 261–264. Washington, D.C.: Advanced Research Projects Agency.
- Ostendorf, M.; Kannan, A.; Austin, S.; Kimball, O.; Schwartz, R.; and Rohlicek, J. R. 1991. Integration of Diverse Recognition Methodologies through Reevaluation of n -Best Sentence Hypotheses. In Proceedings of the DARPA Speech and Natural Language Processing Workshop, 83–87. Washington, D.C.: Defense Advanced Research Projects Agency Information Science and Technology Office.
- Rabiner, L. R., and Juang, B.-H. 1993. *Fundamentals of Speech Recognition*. Englewood Cliffs, N.J.: Prentice Hall.
- Rayner, M.; Carter, D.; Digalakis, V.; and Price, P. 1994. Combining Knowledge Sources to Reorder n -Best Speech Hypothesis Lists. In Proceedings of the ARPA Workshop on Human Language Technology, 217–221. Washington, D.C.: Advanced Research Projects Agency.
- Riley, M. D. 1991. A Statistical Model for Generating Pronunciation Networks. In Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing, Volume 2, 737–740. Washington, D.C.: IEEE Computer Society.
- Wooters, C., and Stolcke, A. 1994. Multiple-Pronunciation Lexical Modeling in a Speaker-Independent Speech Understanding System. Paper presented at the International Conference on Spoken Language Processing, September, Yokohama, Japan.



Andreas Stolcke is a senior research engineer in the Speech Research and Technology Laboratory at SRI International in Menlo Park, California. He received a Ph.D. in computer science from the University of California at Berkeley and a Diplom degree in computer science from Technische Universität Munich. His research interests are in applying machine-learning techniques to speech modeling, especially language modeling for spontaneous conversational speech. His e-mail address is stolcke@speech.sri.com.