Bayesian Adaptation in Speech Recognition

Peter F. Brown, Chin-Hui Lee, James C. Spohrer.

Verbex Corporation

Two Oak Park

Bedford, MA 01730

Abstract

In order to achieve state-of-the-art performance in a speaker-dependent speech recognition task, it is necessary to collect a large number of acoustic data samples during the training process. Providing these samples to the system can be a long and tedious process for users. One way to attack this problem is to make use of extra information from a data bank representing a large population of speakers. In this paper we demonstrate that by using Bayesian techniques, prior knowledge derived from speaker-independent data can be combined with speaker-dependent training data to improve system performance.

Introduction

In the training phase of speech recognition a set of parameters are estimated which characterize a statistical model from which we assume a speaker's acoustic data is generated. The performance of the system depends on both the model and on the precision of the model parameters. For the model to have a high potential of being accurate, it is usually necessary that it be cast in a large parameter space. For the model to realize its potential accuracy, it is necessary that the estimates of the parameters be precise, so that the probability estimates computed from these parameters will be precise. Unfortunately, it can be shown that the amount of training data needed to achieve a given level of precision in the probability estimates increases significantly with the size of the parameter space.[3] This means that a high performance speaker-dependent speech recognition system usually requires a large number of training samples, and therefore that a user of such a system is usually required to go through a long and tedious enrollment process. One approach to this problem would be to reduce the dimensionality of the parameter space, with some technique such as principal component analysis. Another approach is to make use of prior information. In this paper we shall discuss the latter approach.

Bayesian Parameter Estimation using Dynamic Programming

We shall assume the formalism of hidden Markov modeling of speech.[1] That is, we assume that words are modeled by probabilistic finite-state machines in which each arc is labeled with a pattern number. Associated with each pattern is a vector of statistics describing a probability distribution of acoustic observations.

For notational convenience, let X[1:m] represent a sequence, or a set, of m elements, and let X(i) represent the ith element in X[1:m]. Suppose that we are given a sequence of vectors of training data, T[1:n]. That is, we are given a sequence of n acoustic data vectors and a script describing the sequence of words spoken. Let T(i) be the acoustic observation vector associated with the ith time frame in the utterences with the given script. While recognizing speech, we want to calculate the probability of some sequence of words given the input acoustic data to be recognized, and the previously collected training data and script. We can approximate such a probability with the probability of that sequence of words given the input acoustic data to be recognized, and a set of estimates of our model parameters. The task in the training, or enrollment, phase is to calculate this set of parameter estimates. One very reasonable technique is to make the estimates that are most probably true, given the training data and the assumptions of the model. In other words, it is reasonable to solve for the estimates that maximize $Pr\{S[1:r]|T[1:n]\}$. By Bayes' rule we have:

$$Pr\{S[1:r]|T[1:n]\} = \frac{Pr\{T[1:n]|S[1:r]\}Pr\{S[1:r]\}}{Pr\{T[1:n]\}} \quad (1)$$

Maximizing (1) with respect to S[1:r] is the same as solving (2) since $Pr\{T[1:n]\}$ is constant with respect to S[1:r].

$$MAX_{S[1:r]}\ Pr\{T[1:n]|S[1:r]\}Pr\{S[1:r]\} \quad (2)$$

The first factor, $Pr\{T[1:n]|S[1:r]\}$, is the probability of the model, characterized by S[1:r], the probabilistic finite-state machines, and the training script, generating the training data T[1:n]. To calculate this factor we sum over all paths, P, associating patterns with time frames, allowed by our word models and training scripts. We also make use of an implicit assumption of hidden Markov models, that the probability of a path, P, is independent of the pattern statistics, S[1:r]. (2) then becomes

$$MAX_{S[1:r]}\ \Sigma_P\ Pr\{T[1:n]|S[1:r],P\}Pr\{S[1:r]\}Pr\{P\}. \quad (3)$$

We can approximate (3) by (4), in which the sum is replaced by a maximum.

CH1841-6/83/0000-0761 $1.00 © 1983 IEEE

$$MAX_{S[1:r],\,P}\ Pr\{T[1:n]|S[1:r],P\}Pr\{S[1:r]\}Pr\{P\}. \quad (4)$$

We can solve approximately for the S[1:r] and P which maximize (4) in two stages. In the first stage, we assume some preliminary estimates, $S_0[1:r]$, and calculate the optimal path, P_0, which solves

$$MAX_P\ Pr\{T[1:n]|S_0[1:r],P\}Pr\{P\}. \quad (5)$$

To calculate Pr(P) we use the Markov assumption, and assume that the probability of making a transition from one state to another in, or between, word models depends only on the origin state of that transition. Pr{P}, then, is just the product of the transition probabilities along the path P. After also using the assumptions in equations (10) and (11) below to reformulate $Pr\{T[1:n]|S_0[1:r],P\}$, we can efficiently solve for the path, P_0, which maximizes (5) with dynamic programming, as is described in [1] and [4].

This path, P_0, determines a mapping of acoustic data vectors onto patterns. In the second stage we use this mapping to estimate the statistics, $S_0[1:r]$, that solve

$$MAX_{S[1:r]}\ Pr\{T[1:n]|S[1:r],P_0\}Pr\{S[1:r]\}. \quad (6)$$

The transition probability matrix of the probabilistic finite-state machine can also be estimated from the mapping determined by P_0. In this paper, however, we will assume that the transition probabilities are known a priori.

We can then plug the new set of statistics, $S_0[1:r]$, that maximizes (6), into (5), and repeat the first step to determine a new path P_0. We can then use this new P_0 and maximize (6) again, and so on, repeatedly substituting the P_0 which maximizes (5) into (6), and the $S_0[1:r]$ which maximizes (6) into (5). Since the P_0 which maximizes (5) also maximizes

$$Pr\{T[1:n]|S_0[1:r],P\}Pr\{S_0[1:r]\}Pr\{P\}, \quad (7)$$

and since the $S_0[1:r]$ which maximizes (6) also maximizes

$$Pr\{T[1:n]|S[1:r],P_0\}Pr\{S[1:r]\}Pr\{P_0\}, \quad (8)$$

at any point in the iterative solving of (5) and (6), we our guarenteed that

$$Pr\{T[1:n]|S_0[1:r],P_0\}Pr\{S_0[1:r]\}Pr\{P_0\} \quad (9)$$

evaluated at the current P_0 and $S_0[1:r]$ will be at least as great as it would have been when evaluated at a previous P_0 and $S_0[1:r]$. Since probabilities are bounded, by iterating these two maximization steps over the same training data, we will converge to a set of statistics which locally maximizes an approximation of (2), (see the Acknowledgments). While this decision-directed learning procedure only derives an approximation of the desired statistics, empirically we have found that the procedure works well as long as the initial estimates are not wildly implausible. We have also found that convergence normally occurs within a few iterations.

While conducting the experiments described below, when we solved for the the path, P_0, that maximizes (5), and the set of statistics, $S_0[1:r]$, that maximizes (6), we made a number of assumptions. First, we assumed that

$$\log(Pr\{T[1:n]|S[1:r],P\}) = \sum_{j=1}^{n} \log(Pr\{T(j)|S[1:r],P\})/k. \quad (10)$$

Parameters are correlated across time frames, but it is difficult to both properly model this correlation, and use dynamic programming to solve (5), so we treated them as independent and then realized that the resultant probabilities are too small and took the kth root to approximate a correction. This factor, k, can be thought of as the number of time frames per independent frame. The particular value of k that works best, depends on the acoustic parameters used, as well as on the duration of the time frames, be they of fixed or variable duration.

Secondly, we calculated the terms in (10) by assuming that the acoustic observations in the jth frame depend only on the pattern associated with the jth arc in a path through the network of finite-state word models. If we let q(j) be the pattern associated with the jth arc in the path, P, and S(q(j)) the statistics associated with this pattern, then this assumption can be expressed as

$$Pr\{T(j)|S[1:r],P\} = Pr\{T(j)|S(q(j))\}. \quad (11)$$

Thirdly, we assumed that the parameters within a time frame are independent. That is, if there are p parameters per acoustic observation vector, and t(i,j) represents the jth parameter in T(i), we assumed

$$Pr\{T(i)|S(q(i))\} = \prod_{j=1}^{p} Pr\{t(i,j)|S(q(i))\}. \quad (12)$$

This assumption is usually not true but, if the parameter space is chosen carefully, it can be a tolerable assumption.

In calculating Pr{S[1:r]}, the second factor in (6), we assumed that the prior probabilities of statistics in different patterns are independent. That is, we assumed that

$$Pr\{S[1:r]\} = \prod_{i=1}^{r} Pr\{S(i)\}, \quad (13)$$

where S(i) is the vector of statistics associated with pattern i. Empirically, this assumption is very poor. It amounts to assuming

that what a particular speaker's voice sounds like while he is producing one sound, has nothing to do with what it sounds like while he is producing another. With this assumption it would be impossible for the prior statistics to model such things as southern accents or nasalized voices, for example. We made it in the experiments below purely for computionaly simplicity. In the future it will be very important to investigate models in which this assumption is relaxed. One reason for this is that only by relaxing this assumption will it be possible for a system to adequately estimate speaker-dependent statistics for acoustic patterns which are not found in models corresponding to words in the training script.

Finally, we assumed that all the statistics in a given reference pattern are independent of one another. That is, if S(i) is a vector of dimension d, and s(i,j) represents the jth statistic in S(i), then we assumed that

$$\Pr\{S(i)\} = \prod_{j=1}^{d} \Pr\{s(i,j)\}. \quad (14)$$

For example, if the ith pattern happened to be a p dimensional Gaussian, then because of (13) all off-diagonal covariance terms would be 0, and S(i) might consist of p means and p variances. Then (14) would mean that each mean is independent of all the other means and all the variances, and that each variance is independent of all the means and all the other variances. The validity of (14) depends very much on the acoustic parameters and on the distributions used to model those parameters.

Each prior probability, Pr{s(i,j)}, is calculated from a prior probability distribution. Maximum likelihood training procedures use nonimformative prior densities. That is, when making Maximum Likelihood estimates, it is assumed that the members of s(i,j) are distributed uniformly. Bayesian procedures assume prior distributions with parameters derived from speaker-independent data. We determined the class of prior distributions by using conjugate priors.[2] The conjugate prior is defined such that given the observations, T[1:n], the posterior distribution, Pr{s(i,j)|T[1:n]}, is of the same form as the prior distribution, Pr{s(i,j)}. We assumed that we were estimating statistics of Gaussian distributions. The sample mean of a Gaussian variable is distributed as a Gaussian, and the sample variance as Chi-Squared, a subclass of the Gamma distributions. We therefore assumed that the posterior distributions of our means and variances are Gaussians and Gammas, respectively, and used Gaussian priors for the means and Gamma priors for the variances.

Performance Results

We have run a number of experiments comparing Bayesian estimation with informative priors to Maximum Likelihood estimation. To make sure that the patterns corresponded to the same sounds across speakers, we first trained speaker-independent patterns, and then used these speaker-independent patterns as seeds in the speaker-dependent training of each of 26 speakers in the design group, 13 males, and 13 females. The speaker-dependent patterns from these speakers were then used to form prior distributions to be used in estimating statistics for a different group of 24 speakers, 12 males and 12 females. The results listed below are averages over the results for each of these 24 speakers. The data for each speaker consisted of 50 isolated digits, 50 digit doublets, triplets, quadruplets, quintuplets, sextuplets, and septuplets, for a total of 1400 digits per speaker. In addition, there was a similar, but separate, design set of 1400 digits for each speaker. All experiments were run using predetermined word models. A grammar, which accepts utterances of any number of digits was used in the experiments shown in Table 1. In Table 2 a grammar that specifies the number of digits actually occuring in the spoken utterences was used.

The Exp column contains the experiment number. The Init column indicates the type of initialization used. The training procedure was either initialized with speaker-independent patterns, SI, or with templates derived from speaker-dependent isolated words, IW. In all experiments involving estimates with informative priors, speaker-independent inital patterns were used in order to make sure that the prior distributions corresponded to the patterns being trained. In the Train column the set of training data is specified. 3's for example stands for the set of all 50 digit triplets. The Up column indicates the type of template update. ML stands for Maximum Likelihood update, and MA stands for Maximum A Posteriori update. The scores are listed as the number of insertions, in the Ins column, deletions, in the Del column, substitutions, in the Sub column, and total errors, in the Tot column, per 1000 digits spoken.

Table 1 below shows both the effect of training sample size, and the effect of speaker-independent, as compared to speaker-dependent isolated-word, initialization. Note that even after training on 50 digit triplets, we are still undertrained. This can be seen by comparing experiments 8, in which the training data consists of 50 triplets, and 10, in which the training data consists of all 1400 digits. Notice also, that with less training data the improvement of Bayesian estimates over Maximum Likelihood estimates is more significant. This is expected since as the amount of training data goes to infinity the Bayesian estimation formulas converge to the Maximum Likelihood estimation formulas.

Observe that while the substitution rate is always improved by using prior information, the insertion rate increases when prior information is used. Since the noise pattern is treated like any other pattern, this result is unexpected, and we currently have no good explanation for the phenomena.

Table 1 also shows that using speaker-independent patterns to initialize the training procedure is better than using a few speaker-dependent isolated words for initialization.

Exp	Init	Train	Up	Ins	Del	Sub	Tot
1	IW	1's	ML	8.73	9.00	50.5	68.2
2	SI	1's	ML	6.82	6.21	41.2	54.2
3	SI	1's	MA	15.6	3.12	17.7	36.4
4	IW	2's	ML	12.6	3.32	20.7	36.6
5	SI	2's	ML	11.4	2.69	15.0	29.1
6	SI	2's	MA	15.7	1.94	9.64	27.3
7	IW	3's	ML	8.79	2.22	13.6	24.6
8	SI	3's	ML	10.9	2.06	10.1	23.0
9	SI	3's	MA	14.5	1.27	7.48	23.3
10	SI	1-7's	ML	11.0	1.01	5.57	17.6

Table 1

In Table 2, we can see that using a syntax which specifies the number of digits in an utterence accentuates the differences between Maximum A Posteriori estimation, and Maximum Likelihood estimation. The fact that the improvments are greater with a more restrictive syntax clearly is related to the problem with insertions observed in Table 1.

Exp	Init	Train	Up	Ins	Del	Sub	Tot
1	IW	1's	ML	1.52	5.44	52.4	59.4
2	SI	1's	MA	1.50	1.50	18.2	21.2
3	IW	2's	ML	1.29	1.37	21.2	23.8
4	SI	2's	MA	1.92	1.92	9.67	13.5
5	IW	3's	ML	0.87	0.87	14.2	15.9
6	SI	3's	MA	1.12	1.12	7.42	9.67

Table 2

Conclusion

The results above indicate that overall the use of prior information improves recognition accuracy in two ways. Using informative prior densities in the update step improves performance, and performance is also improved simply by using speaker-independent preliminary estimates of templates in our decision-directed training scheme. The latter effect may be due to the fact that isolated words do not contain the articulatory effects found in continuous speech, and therefore initialing with isolated words will not be as good as initializing with data derived from continuous speech.

Acknowledgements

The work described in this paper was done under the guidance of Larry Bahler, Janet Baker, and Jim Baker, We first learned of the Maximum Likelihood version of the decision-directed training procedure described in formulas (5) and (6) from Jim Baker.

References

[1] Baker, James K., "Stochastic Modeling for Automatic Speech Understanding", in Speech Recognition, D.R. Reddy(ed.), Academic Press, 1975.

[2] DeGroot, M. H., Optimal Statistical Decisions, McGraw Hill, New York, 1970.

[3] Hughes, G.H. "On the Mean Accuracy of Statistical Pattern Recognizers", IEEE Trans. Info. Theory, IT-14 pp 55-63, 1968.

[4] Spohrer, Brown, Hochschild, and Baker, "Partial Traceback in Continuous Speech Recognition", in Proceedings of IEEE International Conference on Cybernetics and Society, 1980.