

# Theory and Practice of Acoustic Confusability

Harry Printz and Peder Olsen  
IBM T J Watson Research Center  
Yorktown Heights, NY 10598 USA  
{printz,pederao}@us.ibm.com

## ABSTRACT

In this paper we define two alternatives to the familiar perplexity statistic (hereafter *lexical perplexity*), which is widely applied both as a measure-of-goodness and as an objective function for training language models. These alternatives, respectively *acoustic perplexity* and the *synthetic acoustic word error rate*, fuse information from both the language model and the acoustic model. We show how to compute these statistics by effectively synthesizing a large acoustic corpus, demonstrate their superiority to lexical perplexity as predictors of language model performance, and investigate their use as objective functions for training language models. We present results from a simple speech recognition experiment that demonstrate a small reduction in word error rate.

## 1. Introduction

Let  $P_\theta$  be a language model, where  $P_\theta(w_0 \dots w_S)$  is the probability that the model assigns to word sequence  $w_0 \dots w_S$ , and  $\theta$  is a (typically very large) set of parameters, which determine the numerical value of this probability. One popular method of assigning values to the elements of  $\theta$  is to obtain a corpus  $\mathcal{C} = w_0 \dots w_{N-1}$  of naturally generated text, also very large, and adjust  $\theta$  to maximize  $P_\theta(\mathcal{C})$ , the modeled probability of the corpus. This is an instance of the well-established principle of maximum-likelihood estimation [8, Section 8.2.3].

Maximizing  $P_\theta(\mathcal{C})$  is of course equivalent to minimizing the familiar quantity

$$Y_L(P_\theta, \mathcal{C}) = P_\theta(\mathcal{C})^{-1/|\mathcal{C}|}, \quad (1)$$

called the perplexity [2].  $Y_L(P_\theta, \mathcal{C})$ , or the same statistic  $Y_L(P_\theta, \mathcal{T})$ , computed on an independent test corpus  $\mathcal{T}$ , is commonly reported in papers on language modeling, as evidence of the value of the author's new insight or technique. Yet it is widely acknowledged that lexical perplexity has serious shortcomings, both as a measure-of-goodness and as a training principle for language models. In particular, it is possible to achieve substantial reductions in perplexity, yet increase the word error rate [5]. The search for a substitute is the subject of previous research [3, 6, 9, 10].

In this paper, we define and investigate a statistic, called *acoustic perplexity*, which we propose to use both as a measure-of-goodness for evaluating language models, and as a computational principle for constructing them. Acoustic perplexity differs fundamentally from most other proposed measures, in that it incorporates the characteristics of the acoustic channel. Specifically, acoustic perplexity, and the related quantity *synthetic acoustic word error rate*, which we also define below, measure how well a language model will function when used as a component of a speech recognition system.

In our development, we show how acoustic perplexity—which is closely related to the statistics discussed in [6]—is a natural extension of the existing notion of perplexity (hereafter called *lexical perplexity* when confusion may arise). We analyze acoustic perplexity and the synthetic acoustic word error rate, and define both in terms of two other quantities we introduce, the *acoustic encoding probability* and the *acoustic confusability* of word pairs.

By manipulating the hidden Markov models that are standard in automatic speech recognition, we develop rigorous computational methods for determining all of these quantities, and show how they may be applied in the evaluation and training of language models.

## 2. Definition of Acoustic Perplexity

Let  $\mathcal{C}$  be a large textual corpus, and let  $\mathcal{A}$  be an acoustic realization of this corpus—that is, an audio recording of  $\mathcal{C}$  spoken aloud. The essence of our idea is that if our aim is to train a language model to decode text  $\mathcal{C}$  from acoustics  $\mathcal{A}$ , we should adjust the model's parameters  $\theta$  according to  $\hat{\theta} = \operatorname{argmax}_\theta P_\theta(\mathcal{C} | \mathcal{A})$ . Here  $P_\theta(\mathcal{C} | \mathcal{A})$  is the reverse channel model constructed as detailed below.

Note that while training by lexical perplexity requires only a large text corpus  $\mathcal{C}$ , training by acoustic perplexity requires both text  $\mathcal{C}$  and acoustics  $\mathcal{A}$ , where the latter is a spoken version of the former. We will refer to the pair  $(\mathcal{C}, \mathcal{A})$  as a *joint corpus*. The need for a large joint corpus is a practical obstacle that we address later.

As an exact analog of lexical perplexity, we define

$$Y_A(P_\theta, \mathcal{C}, \mathcal{A}) = P_\theta(\mathcal{C} | \mathcal{A})^{-1/|\mathcal{C}|}, \quad (2)$$

the *acoustic perplexity* of the model  $P_\theta(\mathcal{C} | \mathcal{A})$ , evaluated on lexical corpus  $\mathcal{C}$  and its acoustic realization  $\mathcal{A}$ . Moreover, in the same way as  $P_\theta(\mathcal{C})$  is decomposed into a product of individual-word probabilities, for use in computing  $Y_L$ , so too may  $P_\theta(\mathcal{C} | \mathcal{A})$  be decomposed.

To express this decomposition, we adopt the following notational conventions. The word we're decoding, at the current position  $i$  of the corpus, is  $w_i$ . Its acoustic realization—that is to say, the sound of someone speaking this word—is written  $\alpha(w_i)$ . The sequence of all words preceding  $w_i$ , which is  $w_0 w_1 \dots w_{i-1}$ , is denoted  $h_i$ ; its acoustic realization is  $\alpha(h_i)$ . Likewise the sequence of all words following  $w_i$  is written  $r_i$ , with acoustics denoted  $\alpha(r_i)$ . (Here the letter  $r$  is used to suggest right context.) By elementary probability theory [4, Section 5.2, Proposition 1], we have the familiar decomposition

$$P_\theta(\mathcal{C}) = P_\theta(w_0 w_1 \dots w_{|\mathcal{C}|-1}) = \prod_{i \in \mathcal{C}} p_\theta(w_i | h_i), \quad (3)$$

where “ $i \in \mathcal{C}$ ” denotes a product that advances through every position in the corpus. In view of (3), we will identify the language model  $P_\theta$  with the family of conditional distributions  $\{p_\theta(w | h)\}$  that underlies it, and speak of the family in the singular as “a language model.” Writing  $\mathcal{C} = w_0 w_1 \dots w_{|\mathcal{C}|-1}$  and  $\mathcal{A} = \alpha(w_0 w_1 \dots w_{|\mathcal{C}|-1})$ , as in (3) we have

$$P_\theta(\mathcal{C} | \mathcal{A}) = \prod_{i \in \mathcal{C}} p_\theta(w_i | h_i \mathcal{A}) = \prod_{i \in \mathcal{C}} p_\theta(w_i | h_i \alpha(h_i w_i r_i)), \quad (4)$$

where the rightmost expression is a purely notational variant of the middle one. (This is so because even as  $i$  varies through  $\mathcal{C}$ ,  $\alpha(h_i w_i r_i)$  continues to denote the entire acoustic signal  $\mathcal{A}$ .)

Next we show how the language model probability enters explicitly into this expression. Consider any one factor in (4). Suppressing the  $i$  subscript for readability, by Bayes' theorem we may write

$$p_{\theta}(w|h) p_{\theta}(a(hwr)|wh) = \frac{p(a(hwr)|wh) p_{\theta}(w|h)}{\sum_{x \in V} p(a(hwr)|xh) p_{\theta}(x|h)}. \quad (5)$$

Here  $p_{\theta}(w|h)$  and  $p_{\theta}(x|h)$  are regular language model probabilities and  $V$  is the lexical vocabulary. For the case of discrete speech, this expression takes the form

$$p_{\theta}(w|h) p_{\theta}(a(w)|xh) = \frac{p(a(w)|wh) p_{\theta}(w|h)}{\sum_{x \in V} p(a(w)|xh) p_{\theta}(x|h)}. \quad (6)$$

We will refer to the family of conditional distributions  $\{p(a(hwr)|xh)\}$ , or just  $\{p(a(w)|xh)\}$  for the discrete case, as an *acoustic encoding model*.

### 3. Words and Pronunciations

As a precursor to computing an acoustic encoding probability  $p(a(w)|xh)$ , we need to establish a point of contact between words and their acoustic realizations. This contact is made through one or more *pronunciations* of  $x$ , which is to say models of how  $x$  sounds when spoken.

For simplicity we ignore  $h$  for the moment and consider just  $p(a(w)|x)$ . Here  $a(w)$  is an acoustic event, which is to say a signal that exists in the physical world. By comparison  $x$  is an element of an abstract space of words, drawn from a finite set  $V$ , the vocabulary. This  $x$  is just a placeholder, to determine which model to use when computing  $p(a(w)|x)$ . If there were only one model for  $x$ , then  $p(a(w)|x)$  would be the probability that this model assigns to the observation  $a(w)$ . But in general a given word  $x$  has many pronunciations. We will refer to each one as a *lexeme* or *baseform*—we use these words interchangeably—and write

$$x = \{l^1(x), l^2(x), \dots, l^{n_x}(x)\}, \quad (7)$$

where  $n_x$  is the number of distinct pronunciations we recognize for  $x$ . Carrying this notation a little further, we will write  $l(x) \in x$  for an arbitrary lexeme  $l(x)$  associated with the word  $x$ , and  $\sum_{l(x) \in x}$  for a sum in which  $l(x)$  varies over the lexeme set for  $x$ . More generally, we write  $B$  for the list of all baseforms corresponding to the word vocabulary  $V$ .

We now state without proof a simple lemma that allows us to decompose  $p(a(w)|x)$  in terms of multiple lexemes.

**Lemma 1** *Let  $X, X_1, X_2 \subset \Omega_X$ , with  $X = X_1 \cup X_2$ , and  $X_1 \cap X_2 = \emptyset$ . Let  $Y \subset \Omega_Y$ . Then if  $P(X_1) > 0$  and  $P(X_2) > 0$ ,*

$$P(Y|X) = P(Y|X_1)P(X_1|X) + P(Y|X_2)P(X_2|X).$$

This of course generalizes to an arbitrary finite list of pairwise disjoint alternates  $X_1, \dots, X_M$ . Taking  $a(w)$  as  $Y$ , the set  $x = \{l^1(x), \dots, l^{n_x}(x)\}$  as  $X$ , each singleton  $\{l^i(x)\}$  as  $X_i$ , we have  $p(a(w)|x) = \sum_{l(x) \in x} p(a(w)|l(x)) \cdot p(l(x)|x)$ . This really expresses a trivality, which is that when there is a choice among pronunciation models, the probability of an acoustic event is the sum of the individual pronunciation probabilities, weighted by the prior probability of the pronunciation. By purely formal manipulations this extends to arbitrary conditioning  $h$ , and so we have  $p(a(w)|xh) = \sum_{l(x) \in x} p(a(w)|l(x)h) \cdot p(l(x)|xh)$ . From this point on we will assume that the prior probability of any given pronunciation  $p(l(x)|xh)$  is known, for instance by frequency counting, or just taking a uniform model. Our attention will now focus on the quantity  $p(a(w)|l(x)h)$ .

## 4. Computational Methods

Thus far we have made a case for training a language model by maximization of  $P_{\theta}(\mathcal{C}|\mathcal{A})$ , rather than  $P_{\theta}(\mathcal{C})$ . We now come to grips with two key obstacles to carrying out this program. First, we must devise a scheme for computing the all-important acoustic encoding probabilities,  $p(a(hwr)|hx)$ , for arbitrary  $h, w, r$  and  $x$ . Second, we need a way to cope with the great paucity of joint corpora  $(\mathcal{C}, \mathcal{A})$ . A typical language model training corpus may contain on the order of one billion ( $10^9$ ) words of text. But the largest joint corpus that we know of contains a measly three million ( $3 \times 10^6$ ) or so words of pronounced text. While this may (or may not) suffice to train an acoustic model for a speech recognition system, it surely is not enough to train a language model.

These two obstacles, seemingly unrelated, have a single solution that dovetails them neatly together: we propose to synthesize the information we would obtain from the desired large joint corpus. To do so we proceed in two steps. First we use our existing (small) joint corpus to build acoustic models of each word. This is just acoustic training as we presently understand it. Then we train the language model on a full-sized textual corpus, using synthetic approximations to the required acoustic encoding probabilities. Via a technique that we will explain, these synthetic approximations can be computed analytically from the just-trained acoustic models.

### 4.1 Acoustic Events and Their Models

We begin by discussing the meaning of the acoustic event  $a(w)$ . In a roundabout way, this is actually determined by the model  $p(\cdot|l(x)h)$ . Literally  $a(w)$  is a radially expanding pressure wave, emanating from the speaker's mouth when the word  $w$  is pronounced. But the computational model  $p(\cdot|l(x)h)$  does not operate upon anything like this. The pressure wave is transduced to an analog electrical signal, which is then quantized in time and amplitude, and ultimately processed into a finite sequence  $\langle \bar{a}_w^0 \dots \bar{a}_w^{T-1} \rangle$ , for which we will also write  $\langle \bar{a}_w^i \rangle$ , of  $d$ -dimensional feature vectors—that is, each  $\bar{a}_w^i$  is an element of  $\mathbf{R}^d$ . Thus  $\langle \bar{a}_w^i \rangle$  constitutes the *observation sequence*, the likelihood of which we desire.

In this paper, we will assume that the model  $p(\cdot|l(x)h)$  is a continuous-density hidden Markov model [11, Chapter 2]. We remind the reader that such a model consists of a set of states  $Q$  with identified initial and final states, an output or observation probability density function  $\delta_{qd}$  for each allowed state-to-state transition, and a matrix  $\tau$  of transition probabilities. The likelihood of a sequence of observations  $p(\langle \bar{a}_w^i \rangle|l(x)h)$  is then taken as the sum, over all paths from the initial to final state, of the joint path and individual-observation probabilities. Since such a model serves to evaluate the likelihood of an observation sequence, we will refer to it as a *valuation model*.

Unfortunately, as noted above we are decidedly short of joint corpora. For this reason, we will adopt the strategy of *synthesizing* observation sequences corresponding to  $a(w)$ . To do so we use precisely the same model we would apply to evaluate the likelihood of an observation sequence, but operating with the model's densities and transition probabilities to *generate* data points. Though it has exactly the same form as an evaluation model, we will refer to such a model when used in this way as a *synthesizer model*. We are not the first to propose the use of synthetic data in speech recognition, and we note especially the contributions of [12].

### 4.2 Computing Acoustic Confusability

We now present an algorithm for computing acoustic confusability. Since it relies upon the hidden Markov model formalism, it necessarily involves summation over sequences of states. Our method comprises an exact computation over all state sequences of all lengths, and yields a closed-form expression for  $p_{\theta}(l(w)|l(x)h)$ , for a given model  $l(w)$  of  $a(w)$ . It can be ap-

plied to hidden Markov models of arbitrary size and topology, subject only to practical limits of processor speed and memory size.

#### 4.2.1 Confusability of Densities

Let's first consider a radically simplified version of the computation: suppose that for every acoustic event  $a(w)$ , the associated sequence  $\langle \bar{a}_w^i \rangle$  has length 1, and that the dimension  $d$  of this single vector is also 1. In other words,  $a(w)$  is identified with a single real number; call it  $a_w$ . Likewise suppose that the valuation model  $p(\cdot | l(x) h)$  has a single transition, with associated density  $\delta_{l(x)h}$ , hereafter abbreviated  $\delta_x$ . Hence if  $\mathcal{A}_w = \{a_{w1} \dots a_{wN}\}$  is a corpus of one-dimensional, length-1 observations corresponding to  $N$  true pronounced instances of word  $w$ , then the likelihood of these observations according to the valuation model is

$$L(\mathcal{A}_w | \delta_x) = \delta_x(a_{w1}) \cdots \delta_x(a_{wN}). \quad (8)$$

Now we replace true observations with synthetic ones. Assume for a moment that word  $w$  has a single pronunciation  $l(w)$ , and consider a synthesized observation corpus  $\hat{\mathcal{A}}_w = \{\hat{a}_{w1} \dots \hat{a}_{wN}\}$ , where the elements are iid random variables, distributed according to density  $\delta_{l(w)h}(\cdot)$ , hereafter abbreviated  $\delta_w$ . Fix some finite interval  $[-r, r]$ , and imagine that it is divided into  $N$  subintervals  $J_i = [\nu_i, \nu_i + \Delta\nu]$ , where  $\nu_i = -r + i\Delta\nu$  and  $\Delta\nu = 2r/N$ , where  $i$  runs from 0 to  $N - 1$ . The expected number of elements of  $\hat{\mathcal{A}}_w$  falling into  $J_i$  therefore goes as  $\delta_w(\nu_i) \cdot \Delta\nu \cdot N$ . We define the *synthetic likelihood* of this sequence as

$$L_{rN}(\hat{\mathcal{A}}_w | \delta_x) = \prod_{i=0}^{N-1} \delta_x(\nu_i)^{\delta_w(\nu_i) \cdot \Delta\nu \cdot N}. \quad (9)$$

Hence the *per-event synthetic log likelihood* is

$$\begin{aligned} S_{rN}(\hat{\mathcal{A}}_w | \delta_x) &= \frac{1}{N} \log L_{rN}(\hat{\mathcal{A}}_w | \delta_x) \\ &= \sum_{i=0}^{N-1} \delta_w(\nu_i) \cdot \log \delta_x(\nu_i) \Delta\nu \end{aligned} \quad (10)$$

This is a Riemann-Stieltjes sum, as developed in [1, Chapter 9]. If  $\delta_w$  and  $\delta_x$  are both mixtures of Gaussians it can be shown that  $\lim_{N \rightarrow \infty} S_{rN}(\hat{\mathcal{A}}_w | \delta_x)$  exists and converges to  $\int_{-r}^r \delta_w(\nu) \log \delta_x(\nu) d\nu$ . Taking the limit as  $r \rightarrow \infty$  we define

$$\rho(\delta_w | \delta_x) = \int \delta_w \log \delta_x, \quad (11)$$

the *synthetic log likelihood of  $\delta_w$  given  $\delta_x$* . Exponentiating this quantity we define

$$\kappa(\delta_w | \delta_x) = \exp \rho(\delta_w | \delta_x), \quad (12)$$

the *synthetic likelihood of  $\delta_w$  given  $\delta_x$* . We will treat this quantity as if it were a true likelihood, and operate with it accordingly without further justification. Indeed, this substitution of synthetic for true likelihoods lies at the heart of our program to circumvent our impoverished corpora with synthesized data.

#### 4.2.2 Paths and Densities

In this section we show how paths, densities and synthetic likelihoods are related. In general, the valuation (synthesis) of a sequence depends upon a particular path in the valuation (synthesis) model. This dependency expresses itself in two ways: through the probability of the path, and through the sequence of densities associated with the path.

To begin we review the hidden Markov model formalism, and establish notation as follows:

$Q = \{q_m\}$	a set of states
$q_I, q_F \in Q$	respectively initial, final states
$\tau = \{\tau_{mn}\}$	transition probabilities $q_m \rightarrow q_n$
$\delta = \{\delta_{mn}\}$	a collection of densities, where $\delta_{mn}$ is the density for transition $q_m \rightarrow q_n$

We will refer to the collection  $\langle Q, q_I, q_F, \tau, \delta \rangle$  as a *hidden Markov model*  $H$ . To distinguish between different models, say corresponding to lexemes  $l(x)$  and  $l(w)$ , we will attach a subscript, and refer thus to  $H_x$ , its state set  $Q_x$ , a transition probability  $\tau_{xmn}$ , and so on.

For a given length- $T$  observation sequence  $\langle \bar{a}_w^i \rangle$ , the likelihood of this sequence according to the model  $H$  is

$$\begin{aligned} L(\langle \bar{a}_w^i \rangle | H) &= \sum_{\pi} p(\langle \bar{a}_w^i \rangle | \pi) p(\pi) \\ &= \sum_{\pi} \delta_{\pi^0 \pi^1}(\bar{a}_w^0) \cdots \delta_{\pi^{T-1} \pi^T}(\bar{a}_w^{T-1}) \times \\ &\quad \tau_{\pi^0 \pi^1} \cdots \tau_{\pi^{T-1} \pi^T}. \end{aligned} \quad (13)$$

Here  $\pi$  is a sequence of  $T + 1$  states  $\langle \pi^0, \pi^1, \dots, \pi^T \rangle$ , also called a *path*;  $p(\pi)$  is the probability of the path  $\pi$  according to  $\tau$ ; and  $p(\langle \bar{a}_w^i \rangle | \pi)$  is the likelihood of the observation sequence according to the densities associated with  $\pi$ . Since an observation is associated with a transition and not a state, the likelihood of a sequence of  $T$  observations is evaluated with respect to a path comprising  $T$  transitions and therefore  $T + 1$  states. We say a path is *valid* if  $\pi^0 = q_I$  and  $\pi^T = q_F$ . The sums in equations (13) and (14) run over all valid paths in  $Q^{T+1}$ , the  $(T + 1)$ -fold Cartesian product of  $Q$  with itself.

To develop an intuition for the interaction of paths and densities, consider a restricted model  $H_x$  in which every transition is forced. That is,  $Q_x = \{q_{x0} q_{x1}, \dots, q_{xT}\}$ , where  $q_{x0}$  is  $q_{xI}$  and  $q_{xT}$  is  $q_{xF}$ , and suppose the only transitions with non-zero probability are  $q_{xm} \rightarrow q_{xm+1}$ . Then there is only one valid path through this model, which is  $\pi_x = q_{x0} \dots q_{xT}$ , and its probability is unity. Thus the sum over paths in equation (14) collapses, and we have

$$L(\langle \bar{a}_w^i \rangle | H_x) = \prod_{i=1}^T \delta_{x_{i-1} i}(\bar{a}_w^{i-1}) \quad (15)$$

for the length- $T$  observation sequence  $\langle \bar{a}_w^i \rangle$ .

Suppose now that we are synthesizing an observation sequence  $\langle \hat{a}_w^i \rangle$  according to an identically restricted model  $H_w$ , so that there is only one valid path  $\pi_w = q_{w0} \dots q_{wT}$  through this model. An event synthesized by this model is a sequence of  $T$  synthetic observations, with sample  $\hat{a}_w^0 \sim \delta_{w01}$ , sample  $\hat{a}_w^1 \sim \delta_{w12}$ , and so on. Conversely valuation model  $H_x$  concentrates all its probability mass on observation sequences of length  $T$ , evaluating the likelihood of the first observation according to  $\delta_{x01}$ , and so on. Thus we write

$$L(H_w | H_x \pi_w \pi_x) = \prod_{i=1}^T \kappa(\delta_{w_{i-1} i} | \delta_{x_{i-1} i}) \quad (16)$$

for the synthetic likelihood of  $H_w$  according to  $H_x$ .

The essential point is that the paths  $\pi_x$  and  $\pi_w$  determine which  $\kappa$  values appear in the product. A given  $\kappa(\delta_{w_{mn}} | \delta_{x_{rs}})$  is present precisely when the corresponding transitions  $q_{wm} \rightarrow q_{wn}$  and  $q_{xr} \rightarrow q_{xs}$  are traversed in the same discrete time interval in  $H_w$  and  $H_x$  respectively. In this special case, due to the restrictive transition structure of each model, the path probabilities are both unity, and so do not appear explicitly in (16).

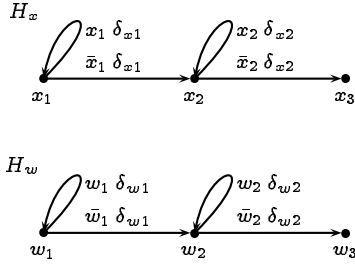


Figure 1: Models  $H_x$  (top) and  $H_w$  (bottom).  $H_x$  has states  $Q_x = \{x_1, x_2, x_3\}$ , transition probabilities  $x_1, \bar{x}_1, x_2$  and  $\bar{x}_2$ , and densities  $\delta_{x_1}, \delta_{x_2}, \delta_{x_3}$ . Likewise for  $H_w$ .

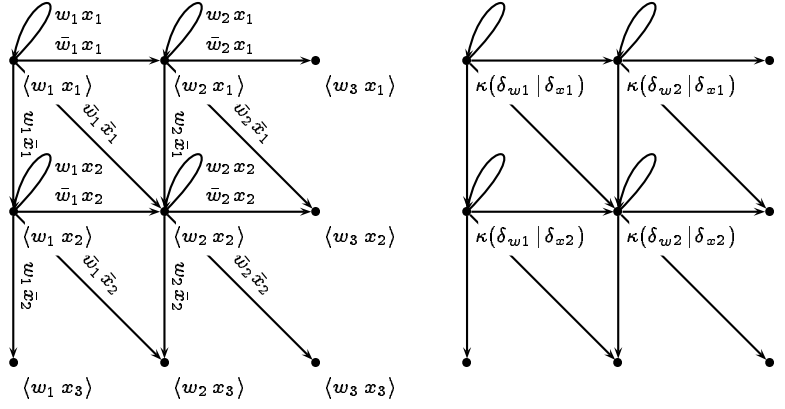


Figure 2: States, Transitions and Synthetic Likelihoods of  $H_w|x$ . Left: States and transition probabilities. Right: Synthetic likelihoods. The value  $\kappa(\delta_{w_m} | \delta_{x_r})$  is shared by all transitions that emanate from state  $\langle w_m x_r \rangle$ . For graphic clarity states are unlabeled in the righthand panel.

### 4.2.3 Confusability of Hidden Markov Models

We now remove the restrictive assumptions on valid paths in  $H_w$  and  $H_x$ , and develop a construction that defines a confusability measure between arbitrary hidden Markov models. This measure comprises observation sequences synthesized over all valid paths of all lengths, and yields an efficient algorithm that gives an exact result.

In what follows,  $H_x$  is the valuation model, and  $H_w$  is the synthesizer model. For simplicity, we will assume that both  $H_x$  and  $H_w$  are three-state models, with the topology, densities and transition probabilities as depicted in Figure 1. However the construction is entirely general, and there is no need to restrict the size or topology of either model.

For economy of expression, we use slightly different notation than given above. We use  $x_1$  both as the name of a state in the model  $H_x$ , and also as the probability of transition  $x_1 \rightarrow x_1$ . We write  $\bar{x}_1$  for the complement of this probability; that is,  $\bar{x}_1$  is the probability of the transition  $x_1 \rightarrow x_2$ .

From these two hidden Markov models, we define the product machine  $H_w|x$  as follows. We begin by setting out some notation and definitions

$$\begin{aligned}
 Q_{w|x} &= Q_w \times Q_x && \text{a set of states} \\
 q_{w|x} I &= \langle q_w I, q_x I \rangle && \text{an initial state} \\
 q_{w|x} F &= \langle q_w F, q_x F \rangle && \text{a final state} \\
 \{\tau_{w|x} \langle w_m, x_r \rangle \langle w_n, x_s \rangle\} &= \{\tau_w m n \tau_x r s\} && \text{a set of transition probabilities.}
 \end{aligned}$$

The states and transitions of this machine are depicted in Figure 2. Although superficially  $H_w|x$  shares many of the characteristics of a hidden Markov model, it is not in fact a model of anything. In particular the arcs are not labeled with densities, from which observation likelihoods may be computed. Instead, we label an arc  $\langle w_m, x_r \rangle \rightarrow \langle w_n, x_s \rangle$  with  $\kappa(\delta_w m n | \delta_x r s)$ , and treat this quantity as the likelihood, according to  $\delta_x r s$ , of observing a sample generated according to  $\delta_w m n$ .

Observe that any path taken through the state diagram of  $H_w|x$  is a sequence  $\langle w^0 x^0 \rangle, \langle w^1 x^1 \rangle \dots$  of pairs of states of the original machines,  $H_w$  and  $H_x$ . There is a natural bijection between sequences  $\pi_{w|x}$  of state pairs, and pairs of state sequences  $(\pi_w \pi_x)$ . Moreover, every pair  $\langle \pi_w \pi_x \rangle$ , of valid paths of identical lengths in  $H_w$  and  $H_x$  respectively, corresponds to a valid path in  $H_w|x$ , and conversely. Thus a computation that traverses all valid paths in  $H_w|x$  comprises all pairs of same-length valid paths in the synthesizer and valuation models.

We proceed to construct a trellis for the state-transition graph of Figure 2, and to write appropriate forward trellis equations, with synthetic likelihoods in place of true observation probabilities. The left panel of Figure 3 shows two successive time slices in the trellis. The arcs drawn correspond to the allowed state transitions of  $H_w|x$ , as the reader is encouraged to confirm.

Next we derive the forward trellis equation for state  $\langle w_1 x_2 \rangle$ , as pictured in the righthand panel of the same figure. Our aim is to obtain an expression for  $\alpha_{\langle w_1 x_2 \rangle}^{t+1}$ , the likelihood of arriving at this state at time  $t+1$  by any path, having observed  $t$  frames of synthetic data for  $w$ , as evaluated by the densities of  $x$ . It is apparent from the diagram that there are only two ways that this can happen: via a transition from  $\langle w_1 x_1 \rangle$ , and via a transition from  $\langle w_1 x_2 \rangle$ .

Let us suppose that the synthetic likelihood of arriving in state  $\langle w_1 x_1 \rangle$  at time  $t$  by all paths is  $\alpha_{\langle w_1 x_1 \rangle}^t$ . The probability of traversing both transition  $w_1 \rightarrow w_1$  in  $H_w$  and transition  $x_1 \rightarrow x_2$  in  $H_x$  is  $w_1 \bar{x}_1$ , and the synthetic likelihood of the data corresponding to this transition pair is  $\kappa(\delta_{w_1} | \delta_{x_1})$ . Thus the contribution to  $\alpha_{\langle w_1 x_2 \rangle}^{t+1}$  of all paths passing through  $\langle w_1 x_1 \rangle$  at  $t$  is  $\kappa(\delta_{w_1} | \delta_{x_1}) w_1 \bar{x}_1 \alpha_{\langle w_1 x_1 \rangle}^t$ . Likewise the contribution from paths passing through  $\langle w_1 x_2 \rangle$  at  $t$  is  $\kappa(\delta_{w_1} | \delta_{x_2}) w_1 x_2 \alpha_{\langle w_1 x_2 \rangle}^t$ . Since these paths pass through different states at time  $t$  they are distinct, so their probabilities add and we have

$$\alpha_{\langle w_1 x_2 \rangle}^{t+1} = \kappa(\delta_{w_1} | \delta_{x_1}) w_1 \bar{x}_1 \alpha_{\langle w_1 x_1 \rangle}^t + \kappa(\delta_{w_1} | \delta_{x_2}) w_1 x_2 \alpha_{\langle w_1 x_2 \rangle}^t, \quad (17)$$

the forward trellis equation for state  $\langle w_1 x_2 \rangle$ . In a straightforward way, we can write such an equation for every state of  $H_w|x$ .

Now we make a crucial observation. Let us write  $\bar{\alpha}^t$  for the distribution of probability mass across all nine states of  $Q_{w|x}$  at time  $t$ , thus

$$\bar{\alpha}^t = \langle \alpha_{\langle w_1 x_1 \rangle}^t \alpha_{\langle w_2 x_1 \rangle}^t \alpha_{\langle w_3 x_1 \rangle}^t \alpha_{\langle w_1 x_2 \rangle}^t \alpha_{\langle w_2 x_2 \rangle}^t \alpha_{\langle w_3 x_2 \rangle}^t \alpha_{\langle w_1 x_3 \rangle}^t \alpha_{\langle w_2 x_3 \rangle}^t \alpha_{\langle w_3 x_3 \rangle}^t \rangle^T \quad (18)$$

and likewise  $\bar{\alpha}^{t+1}$  for the same vector one timestep later. The complete family of trellis equations can be expressed as  $\bar{\alpha}^{t+1} = M \bar{\alpha}^t$ . Here  $M$  is a square matrix of dimension  $m \times m$ , where  $m = |Q_{w|x}| = |Q_w| \cdot |Q_x|$ . Note that the elements of  $M$  do not depend upon  $t$ .

By assumption, at time 0 all the probability mass in  $\bar{\alpha}^0$  is concentrated on the initial state  $\langle w_1 x_1 \rangle$ ; thus  $\bar{\alpha}^0 = \langle 1 \dots 0 \rangle^T$ . By iteration of  $\bar{\alpha}^{t+1} = M \bar{\alpha}^t$  we obtain the sequence of distributions

$$\bar{\alpha}^1 = M \bar{\alpha}^0, \quad \bar{\alpha}^2 = M \bar{\alpha}^1 = M^2 \bar{\alpha}^0, \dots \quad (19)$$

or in general  $\bar{\alpha}^t = M^t \bar{\alpha}^0$ . Now let us ask, what is the total probability, over all time, of arriving in the final state  $\langle w_3 x_3 \rangle$  of  $H_{w|x}$ ? We will write  $\xi_{w|x}$  for this quantity. By the set of equations in (19) we have

$$\xi_{w|x} = [\bar{\alpha}^0]_{\langle w_3 x_3 \rangle} + [\bar{\alpha}^1]_{\langle w_3 x_3 \rangle} + \dots \quad (20)$$

$$= [\bar{\alpha}^0 + M^1 \bar{\alpha}^0 + M^2 \bar{\alpha}^0 + \dots]_{\langle w_3 x_3 \rangle} \quad (21)$$

$$= [(I + M + M^2 + M^3 + \dots) \bar{\alpha}^0]_{\langle w_3 x_3 \rangle} \quad (22)$$

where the notation  $[\ ]_{\langle w_3 x_3 \rangle}$  denotes the extraction of element  $\langle w_3 x_3 \rangle$  of the vector enclosed in brackets. It remains to evaluate the sum  $S = I + M + M^2 + M^3 + \dots$ . We will say that the matrix  $M$  is *convergent* if each of the individual sequences corresponding to the matrix elements of  $S$  converges. It can be proved that if  $M$  is convergent then  $(I - M)^{-1}$  exists and  $S = (I - M)^{-1}$ . A sufficient condition for  $M$  to be convergent is that each eigenvalue  $\lambda$  of  $M$  satisfy  $|\lambda| < 1$ .

Returning to the main line of development, if  $M$  is convergent then

$$\xi_{w|x} = [(I - M)^{-1} \bar{\alpha}^0]_{\langle w_3 x_3 \rangle}. \quad (23)$$

But now observe that the vector  $(I - M)^{-1} \bar{\alpha}^0$  is just the  $\langle w_1 x_1 \rangle$  column of the matrix  $(I - M)^{-1}$ , and we seek the  $\langle w_3 x_3 \rangle$  element of this vector. More generally, if  $\bar{u}_I$  is  $\bar{\alpha}^0$ , which is to say an  $m$ -element vector with a 1 in the position corresponding to the initial state of  $H_{w|x}$ , and with 0s everywhere else, and if  $\bar{u}_F$  is defined likewise, except with a 1 in the position corresponding to the final state of  $H_{w|x}$ , then

$$\xi_{w|x} = \bar{u}_F^T (I - M)^{-1} \bar{u}_I. \quad (24)$$

We take this as the fundamental definition of the confusability of  $H_w$  given  $H_x$ . It is our algorithm's estimate of the likelihood, according to model  $H_x$ , of observing acoustics synthesized according to  $H_w$ .

We have treated  $H_w$  and  $H_x$  abstractly, but it should be clear that we intend for each one to represent a lexeme. Thus  $H_w$  is the hidden Markov model for some lexeme  $l(w)$ , and likewise  $H_x$  for  $l(x)$ . To exhibit this explicitly we will change notation and write

$$\xi(l(w)|l(x)h) = \bar{u}_F^T (I - M(l(w)|l(x)h))^{-1} \bar{u}_I. \quad (25)$$

The reader may be wondering why we introduced the new quantity  $\xi$ , rather than writing  $p(l(w)|l(x)h)$  outright on the lefthand side of (25). The answer is that experience has shown us that (25) yields exceedingly small values. Much of the likelihood in acoustic space belongs to non-speech acoustic events, or so our models declare. Only a small amount is left to spread over the legitimate word sounds enumerated in the baseform list  $B$ .

For this reason we renormalize the results of (25) via

$$p_\lambda(l(w)|l(x)h) = \frac{\xi^{1+\lambda}(l(w)|l(x)h)}{\sum_{l(z) \in B} \xi^{1+\lambda}(l(z)|l(x)h)}. \quad (26)$$

The presence of the exponent  $\lambda$  is due to the extreme sharpness of the distributions obtained from renormalization of raw  $\xi$  values. The need to smooth these distributions is discussed in Section 5.2 below.

### 4.3 Multiple Pronunciations

Our work so far has given us a closed-form analytic expression for  $p(l(w)|l(x)h)$ ; by our discussion in Section 3 we may combine results for the various  $l(x) \in x$  to yield  $p(l(w)|xh)$ . However word  $w$  itself may admit several pronunciations.

To incorporate this elaboration we will declare that  $a(w)$  is a set comprised of all the  $l(w) \in w$ , and furthermore treat the various  $l(w)$  as non-overlapping. It then follows that  $p(a(w)|xh) = \sum_{l(w) \in w} p(l(w)|xh)$ . This argument is admittedly open to debate, and we are investigating alternatives. But for the moment we adopt this expression as a working definition.

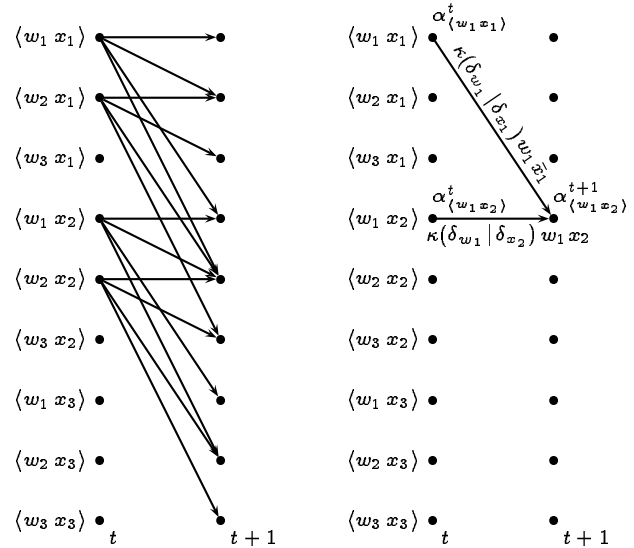


Figure 3: Trellis of  $H_{w|x}$ . The lefthand panel shows two successive slices, and legal transitions, of the trellis corresponding to  $H_{w|x}$ . Each bullet is a node of the trellis, and corresponds to the indicated state of  $H_{w|x}$ . For clarity of the diagram, we display names of the lefthand column of nodes only. The righthand panel shows the derivation of the forward trellis equation for state  $\langle w_1 x_2 \rangle$ .

## 5. Performance Statistics

We begin by introducing a new statistic, the synthetic acoustic word error rate (hereafter sawer). We discuss the adjustment of the exponent  $\lambda$ , which appears in the acoustic encoding probabilities that underlie acoustic perplexity. Finally we present experimental evidence comparing lexical perplexity, acoustic perplexity and sawer as measures of language model performance. In our experiments, sawer was the most reliable of the three statistics.

### 5.1 Synthetic Acoustic Word Error Rate

As argued above, the acoustic encoding probability  $p(a(w)|xh)$  is the probability, according to the acoustic models for word  $x$  in context  $h$ , of observing acoustic data  $a(w)$ . Thus it is natural to speak of  $p_\theta(w|a(w)h)$ , obtained by Bayes' theorem via (6), as the *acoustic decoding probability*.

In fact this number does not represent the probabilistic operation of any real decoder that we know of. It is the probability, according to the models appearing in (6), that word  $w$  was spoken, given lexical history  $h$  and acoustics  $a(w)$ . However, let us for the moment treat it as an estimate of the probability of decoding word  $w$ , and ask what conclusions we can draw.

We proceed to define a random variable  $X_i$ , associated with position  $i$  of the joint corpus  $\langle \mathcal{L}, \mathcal{A} \rangle$ . Suppose for the moment that  $\mathcal{A}$  contains true and not synthesized acoustics, and that we have decoded the entire corpus. The sequence  $\langle X_i \rangle$  will represent the outcome of this decoding experiment, as follows:  $X_i$  is 0 if acoustics  $a(w_i)$  are decoded correctly as  $w_i$ , and  $X_i$  is 1 otherwise. Let

$N = |\mathcal{C}|$  be the size of the corpus. Then for any assignment of 0s and 1s to  $\langle X_i \rangle$ , corresponding to some actual decoding of acoustics  $\mathcal{A}$ , the quantity  $\sum_{i \in \mathcal{C}} X_i/N$  is the *true word error rate*, ignoring insertions and deletions.

We now consider this statistic for the case of synthesized acoustics, with the behavior of the decoder modeled by the quantity  $p_\theta(w_i | a(w_i) h_i)$ . If this is nominally the probability of correctly decoding  $w_i$  from acoustics  $a(w_i)$ , then its complement  $1 - p_\theta(w_i | a(w_i) h_i)$  is the probability of decoding  $a(w_i)$  incorrectly. Thus the *expected word error rate* according to this model is

$$S_A(P_\theta, \mathcal{C}, \mathcal{A}) = E_{p_\theta}[\sum_{i \in \mathcal{C}} X_i/N] = \sum_{i \in \mathcal{C}} E_{p_\theta}[X_i]/N \quad (27)$$

$$= \sum_{i \in \mathcal{C}} (1 - p_\theta(w_i | a(w_i) h_i))/N, \quad (28)$$

where we have used  $E_{p_\theta}[X_i] = 0 \cdot p_\theta(w_i | a(w_i) h_i) + 1 \cdot (1 - p_\theta(w_i | a(w_i) h_i))$ . We refer to  $S_A$  as the *synthetic acoustic word error rate*, or sawer.

## 5.2 Adjustment of Smoothing Parameter $\lambda$

In equation (26) above we introduced the parameter  $\lambda$  on grounds that the raw confusabilities  $\xi$  are too sharp. At the time we presented no justification for this claim. We do so now, and also explain how we propose to determine  $\lambda$ .

For justification, we need look no further than the raw confusabilities of words that are frequently decoded wrongly. Consider for instance the word *Boston*. Table 1 shows the five most confusable lexemes of each word, computed from (26) both without ( $\lambda = 0.0$ ) and with ( $\lambda = -0.86$ ) smoothing.

$l(x) = \text{B AO S T AX N}$	$\log_{10} p_\lambda(l(w) l(x))$		
$w$	$l(w)$	$\lambda = 0.00$	$\lambda = -0.86$
<i>Boston</i>	B AO S T AX N	-0.00	-0.57
<i>Austin</i>	AO S T AX N	-5.92	-1.40
<i>Baden</i>	B AO DX AX N	-10.59	-2.05
<i>busted</i>	B AH S T IX DD	-10.73	-2.07
<i>bossed</i>	B AO S TD	-11.54	-2.19

Table 1: Unsmoothed and Smoothed Confusabilities.

Inspecting the unsmoothed logprobs ( $\lambda = 0.0$ ) reported in the table, we see a gap between  $l(w) = \text{B AO S T AX N}$  and the next most confusable lexeme  $l(w) = \text{AO S T AX N}$  of almost six orders of magnitude. In other words, the estimated probability of decoding *Austin* when *Boston* was said is about 1,000,000 times smaller than decoding the word correctly. But in fact, *Austin* is a frequent misdecoding of *Boston*.

We believe this excessive sharpness arises because of a well-known weakness of hidden Markov models, which is that they treat successive acoustic observations as independent events. Since these observations are of course well-correlated, this results in a severe underestimate of the likelihood of true observation sequences. Moreover, since our synthesis scheme generates observations independently as well, our method exhibits this weakness in spades.

The solution we adopted was to introduce the smoothing parameter  $\lambda$  in (26). We then decoded an independent corpus  $\mathcal{H}$  to obtain a true word error rate, computed the sawer  $S_A(P_{\theta\lambda}, \mathcal{H}, \mathcal{A})$  on the same corpus, and adjusted  $\lambda$  to match  $S_A$  to the true word error rate. This yielded  $\lambda = -0.86$ , for an exponent of  $1 + \lambda = 0.14$ . Some of the resulting smoothed lexeme confusabilities, which are much more plausible, are exhibited in the righthand column of Table 1. We use this value of  $\lambda$  for the experiments reported below.

## 5.3 Predictive Power

We now exhibit results for three empirical measures of language model performance: lexical perplexity  $Y_L$ , acoustic perplexity  $Y_A$  and synthetic acoustic word error rate  $S_A$ . We tested each measure on three independent test corpora, respectively SOB (11180 total words, 10 speakers, office dictation), NRR (9060 total words, 5 speakers, IBM ViaVoice product consumer data) and SPT (568141 words, 10 speakers, spontaneous speech). For each test corpus, we evaluated the same five language models. An evaluation consisted of determining the true word error rate, by decoding with each language model, and also computing values of the three measures, via (1), (2) or (28), using the true text and synthesized acoustics for each corpus.

Figure 4 displays the results of these tests. Each vertical axis gives the true word error rate, and each horizontal axis gives one of the three statistics listed above. A statistic that correlates well with word error rate will have a graph that slopes more or less directly from upper right to lower left. It is clear from appearances alone that the sawer statistic is a much better predictor of model performance than either of the other two. Moreover Table 2, listing the sample correlation coefficient [8, Section 7.1] for each statistic against word error rate, for each data set, provides empirical confirmation of this claim.

Corpus	$Y_L(P_\theta, \mathcal{T}, \mathcal{A})$	$Y_A(P_\theta, \mathcal{T}, \mathcal{A})$	$S_A(P_\theta, \mathcal{T}, \mathcal{A})$
SOB	0.918	0.979	0.989
NRR	-0.562	-0.402	0.934
SPT	-0.995	-0.921	0.951

Table 2: Sample Correlation Coefficient  $r$  for Test Data Sets.

## 6. Training of Language Models

We have shown that acoustic perplexity and sawer—especially the latter—are better predictors of language model performance than lexical perplexity. This has led us to investigate the adoption of one or the other as the objective function to be minimized in the training of language models.

Unfortunately there is no direct analytic expression for the global minimum of either of our measures. But in this section we give an *iterative* algorithm for training a language model by optimization of the synthetic acoustic word error rate. Though we will not develop the relationship in detail, in fact the same algorithm, with appropriate changes, can be used to optimize the acoustic perplexity.

For clarity we treat the case of discrete speech; however, our methods apply to continuous speech as well. Our starting point is equation (28). We seek the language model family  $P_\theta = \{p_\theta(w | h)\}$ , that minimizes  $S_A(P_\theta, \mathcal{C}, \mathcal{A})$ , where the parameters  $\theta$  are raw language model probabilities  $p_\theta(w|h) = \theta_{wh}$ .

Our first observation is that minimizing  $S_A(P_\theta, \mathcal{C}, \mathcal{A})$  is equivalent to maximizing  $1 - S_A(P_\theta, \mathcal{C}, \mathcal{A})$ . Moreover by collecting terms with identical history  $h$ , the expression for  $1 - S_A(P_\theta, \mathcal{C}, \mathcal{A})$  can be brought to the form

$$\frac{1}{|\mathcal{C}|} \sum_{h \in \mathcal{C}} \left( \sum_{w \in \mathcal{V}} c(w, h) \frac{p(a(w)|w h) \theta_{wh}}{\sum_{x \in \mathcal{V}} p(a(w)|x h) \theta_{xh}} \right), \quad (29)$$

where the outer sum runs over distinct *histories* in  $\mathcal{C}$ . Because the probability distributions  $p_\theta(w|h) = \theta_{wh}$  are independent for distinct  $h$ , it suffices to maximize the parenthesized sum in (29) separately for each value of  $h$ . We proceed to do this now.

Fix the history  $h$  to some definite value, and define the function

$$f(\theta) = \sum_{w \in \mathcal{V}} c_w \frac{a_w \theta_w}{\sum_{x \in \mathcal{V}} b_{wx} \theta_x}. \quad (30)$$

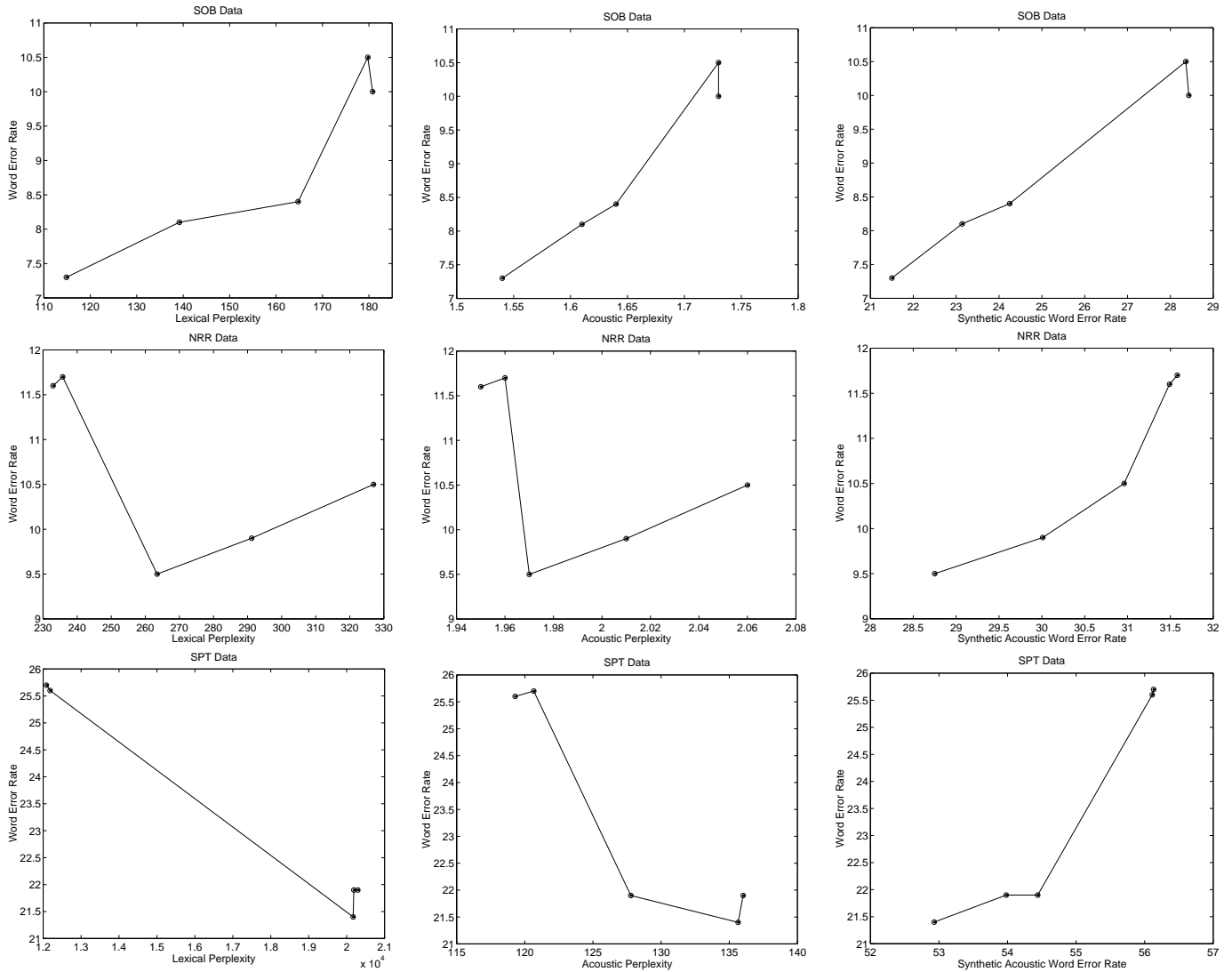


Figure 4: Comparison of Lexical Perplexity, Acoustic Perplexity and Synthetic Acoustic Word Error Rate. The lefthand column of graphs shows the relation between lexical perplexity and true word error rate, for each of five different language models, as computed on three test corpora. The middle column shows this relation for acoustic perplexity, and the righthand column for synthetic acoustic word error rate, both against true word error rate, for the same five language models, and the same three test corpora. Top row: SOB. Middle row: NRR. Bottom row: SPT.

Here  $c_w = c(w, h)$ ,  $\alpha_w = p(a(w) | w h)$  and  $b_{w_x} = p(a(w) | x h)$ ; moreover  $\theta$  is just the vector  $\langle \theta_w \rangle = \langle \theta_{w h} \rangle$  for fixed  $h$ , as  $w$  runs through  $V$ . Comparison shows that  $f(\theta)$  is the parenthesized sum in (29) for the given  $h$ . Note that  $f(\theta)$  satisfies  $f(t\theta) = f(\theta)$  for all  $t > 0$ . Writing  $m = |V|$ , we proceed to establish the following lemma.

**Lemma 2** Let  $f : \mathbf{R}^m \rightarrow \mathbf{R}$  be a  $C^1(\mathbf{R}^m)$  function satisfying  $f(t\theta) = f(\theta)$  for all  $t > 0$ . Then

$$\sum_{i \in V} \theta_i \frac{\partial f}{\partial \theta_i}(\theta) = 0 \quad \forall \theta \in \mathbf{R}^m \quad (31)$$

**Proof:** Define the function  $g(t) = f(t\theta)$ . Since  $g(t)$  is a constant function we have  $0 = \frac{dg}{dt}|_{t=1} = \frac{d}{dt} \{f(t\theta)\}|_{t=1} = \sum_{i \in V} \theta_i \frac{\partial f}{\partial \theta_i}(\theta)$ . ■

A geometric proof of the lemma follows from the observation that  $f$  being constant along the direction  $\theta$  implies that the gradient  $\nabla f(\theta)$  must be orthogonal to  $\theta$ .

Lemma 2 gives us a method to locate incrementally larger values for  $f(\theta)$ . The following theorem specifies a direction in which we are guaranteed to find a better value, unless we are at a boundary point, or a point where  $\nabla f = 0$ .

**Theorem 1** Let  $f : \mathbf{R}^m \rightarrow \mathbf{R}$  be a  $C^2(\mathbf{R}^m)$  function such that  $f(t\theta) = f(\theta)$ . Consider  $\theta$  satisfying  $\sum_{i \in V} \theta_i = 1$  and  $\theta_i \geq 0 \forall i$ . Suppose as well that for some  $i \in V$ , both  $\frac{\partial f}{\partial \theta_i}(\theta) \neq 0$  and  $0 < \theta_i < 1$  hold. Define  $\hat{\theta}_i^\epsilon = \theta_i + \epsilon \theta_i \frac{\partial f}{\partial \theta_i}(\theta)$ . Then there exists  $\epsilon > 0$  such that the following three properties hold

$$\sum_{i \in V} \hat{\theta}_i^\epsilon = 1, \quad (32)$$

$$\hat{\theta}_i^\epsilon \geq 0 \quad \forall i \in V \quad (33)$$

and

$$f(\hat{\theta}^\epsilon) > f(\theta). \quad (34)$$

**Proof:** The proof of (32) follows from Lemma 2: we have  $\sum_{i \in V} \hat{\theta}_i^\epsilon = \sum_{i \in V} \theta_i + \epsilon \sum_{i \in V} \theta_i \frac{\partial f}{\partial \theta_i}(\theta) = 1 + 0 = 1$ .

For (33), observe that since  $\hat{\theta}_i^\epsilon = \theta_i(1 + \epsilon \frac{\partial f}{\partial \theta_i})$  and  $\theta_i \geq 0$ , it suffices that the parenthesized quantity be non-negative. If  $\partial f / \partial \theta_i \geq 0$  this is immediate. If  $\partial f / \partial \theta_i < 0$ , it suffices that  $\epsilon < -1 / (\partial f / \partial \theta_i)$ . Hence by choosing  $\epsilon$  sufficiently small all the inequalities of (33) will be satisfied.

Finally to establish (34), by Taylor’s theorem [1, Theorem 6-22]

$$f(\hat{\theta}^\epsilon) = f(\theta) + \sum_{i \in V} (\hat{\theta}_i^\epsilon - \theta_i) \frac{\partial f}{\partial \theta_i}(\theta) + \sum_{i \in V} \sum_{j \in V} (\hat{\theta}_i^\epsilon - \theta_i)(\hat{\theta}_j^\epsilon - \theta_j) \frac{\partial^2 f}{\partial \theta_i \partial \theta_j}(\theta^*), \quad (35)$$

for some  $\theta^*$  in the closed interval bounded by  $\theta_i$  and  $\hat{\theta}_i^\epsilon$ . Substituting in the formula for  $\hat{\theta}^\epsilon$  and collecting terms in order of powers of  $\epsilon$  we get

$$f(\hat{\theta}^\epsilon) = f(\theta) + \epsilon \sum_{i=1}^m \theta_i \left( \frac{\partial f}{\partial \theta_i}(\theta) \right)^2 + O(\epsilon^2). \quad (36)$$

The expression  $\sum_{i \in V} \theta_i \left( \frac{\partial f}{\partial \theta_i}(\theta) \right)^2$  is always strictly positive under the assumptions made in the theorem. The  $O(\epsilon^2)$  can therefore always be dominated for small enough  $\epsilon$ , thus proving (34). ■

The proof of Theorem 1 tells us only that some suitable  $\epsilon > 0$  exists, and does not provide us with an  $\epsilon$  for which the theorem holds. Such a value may in fact be found using theory developed in [7]. Unfortunately this value is of no practical use, as it is far too small to yield an efficient update rule. However, conducting a line search along the direction  $\langle \theta_i (\partial f / \partial \theta_i) \rangle$  is efficient and effective.

## 7. Decoding Results

We experimented with a total of five language models (no relation to those in Section 5.3 above). No model made any use of history; we made this choice to keep the sawer training computation tractable. We began with two base language models, the uniform model  $p_0(w)$  and the unigram model  $p_1(w)$ , respectively defined as  $p_0(w) = 1/|V|$  and  $p_1(w) = c(w)/N$ , where  $c(w)$  is the count of  $w$  in the training corpus, and  $N$  is its size.

We used  $p_0$  and  $p_1$  as starting points for two models, respectively  $s_0$  and  $s_1$ , trained by iterative improvement of the sawer objective function, using the methods of the preceding section. The training data for  $s_0$  was a synthetic corpus in which each word  $w \in V$  occurred exactly once. The training data for  $s_1$  was likewise synthetic, in which each word  $w \in V$  appeared exactly  $c(w)$  times. Thus  $s_0$  and  $s_1$  had access to no additional lexical data, other than that available to  $p_0$  and  $p_1$  respectively. The fifth model we experimented with was the linear mixture  $\bar{p} = 0.1s_0 + 0.1p_0 + 0.4s_1 + 0.4p_1$ .

Our decoding experiments consisted of decoding the three test corpora used above to create Figure 4. Our decoder had a vocabulary of 63389 words, and used acoustic models trained on about 250 hours of acoustic data. The results are summarized in Table 3 below. We observe a small improvement in the word error rate for the mixture model. Note that this model is formally a list of  $|V|$  unigram probabilities, and thus contains exactly the same number of parameters as  $p_1$  (or  $s_1$ ). Thus we have improved performance, over either  $p_1$  or  $s_1$  individually, without increasing the model size.

## 8. Summary

In this paper we explored the theory and practice of acoustic confusability. We defined and motivated the statistics *acoustic perplexity* ( $Y_A$ ) and *synthetic acoustic word error rate* ( $S_A$ ). We showed how these depend upon the *acoustic encoding probability*  $p(a(w) | x h)$ , and introduced an algorithm for computing this

	SOB	NRR	SPT	ALL
$p_0$	78.8%	76.2%	87.4%	84.9%
$p_1$	40.9%	47.3%	59.7%	55.5%
$s_0$	85.0%	84.4%	90.1%	88.7%
$s_1$	51.1%	58.3%	66.1%	63.0%
$\bar{p}$	40.5%	44.8%	59.8%	55.2%

Table 3: WER Results for Base ( $p_0, p_1$ ), Sawer ( $s_0, s_1$ ) and Mixture ( $\bar{p}$ ) Models.

quantity synthetically. We demonstrated the superiority of  $Y_A$  and  $S_A$  as predictors of language model performance, and then showed how these statistics may be used as objective functions for language model training. We presented results from a simple speech recognition experiment, showing that a mixture model trained with  $S_A$  as an objective function of some components can yield a small reduction in word error rate. We are extending these ideas to more sophisticated recognition experiments, and to such diverse tasks as vocabulary selection for speech recognition systems, and the ranking of features for maximum entropy language models.

## REFERENCES

- [1] Tom M. Apostol. *Mathematical Analysis*. Addison-Wesley, Reading, MA, 1957.
- [2] L.R. Bahl, J.K. Baker, F. Jelinek, and R.L. Mercer. Perplexity - a measure of the difficulty of speech recognition tasks. *Program of the 94th Meeting of the Acoustical Society of America J. Acoust. Soc. Am.*, 62:S63, 1977. Suppl. no. 1.
- [3] Stanley Chen, Douglas Beeferman, and Ronald Rosenfeld. Evaluation metrics for language models. In *Proceedings of the Broadcast News Transcription and Understanding Workshop*, pages 275–280, Lansdowne, Virginia, February 1998.
- [4] Kai Lai Chung. *Elementary Probability Theory with Stochastic Processes*. Springer-Verlag, New York, NY, 1979.
- [5] Philip Clarkson and Tony Robinson. The applicability of adaptive language modelling for the broadcast news task. In *Proceedings of the Fifth International Conference on Spoken Language Processing*, Sydney, Australia, November 1998.
- [6] Marco Ferretti, Giulio Maltese, and Stefano Scarci. Measuring information provided by language model and acoustic model in probabilistic speech recognition: Theory and experimental results. *Speech Communication*, 9:531–539, 1990.
- [7] P. S. Gopalakrishnan, D. Kanevsky, A. Nádas, and D. Nahamoo. An inequality for rational functions with applications to some statistical estimation problems. *IEEE Transactions on Information Theory*, 37(1):107–113, 1991.
- [8] Paul G. Hoel. *Introduction to Mathematical Statistics*. John Wiley and Sons, New York, NY, 5th edition, 1984.
- [9] Akinori Ito, Masaki Kohda, and Mari Ostendorf. A new metric for stochastic language model evaluation. In *Proceedings of the Sixth European Conference on Speech Communication and Technology*, volume 4, pages 1591–1594, Budapest, Hungary, September 1999.
- [10] Rukmini Iyer, Mari Ostendorf, and Marie Meteer. Analyzing and predicting language model improvements. In *IEEE Workshop on Automatic Speech Recognition and Understanding*, 1997.
- [11] Frederick Jelinek. *Statistical Methods for Speech Recognition*. The MIT Press, Cambridge, MA, 1997.
- [12] Don McAllaster, Larry Gillick, Francesco Scattone, and Mike Newman. Fabricating conversational speech data with acoustic models: A program to examine model–data mismatch. In *Proceedings of ICSLP*, Sydney, Australia, November 1998. Paper 986.