

This is some MSU-contributed verbiage to a recent NSF proposal jointly submitted with the Center for Speech & Lang. Research at U. Colorado. You might be interested in the yellow topic below. The paper referenced in footnote 2 follows this page.

Watermarking. Extensive research into the robust watermarking of audio data has been undertaken in connection with the NGSW project at Michigan State University (MSU).¹ The SpeechFind project will serve as a further testbed for deployment of developed watermarking strategies to further verify the methods in an array of challenging acoustic signal conditions. The ability to protect intellectual property is of strategic economic importance to U.S. interests while the ability to authenticate audio streams has both economic and security implications.

Language Modeling. The principal focus of the Speech Processing Laboratory's efforts will be on various aspects of language modeling for the SpeechFind system. In a recognition problems of the scale proposed in this work, language models (LMs) are clearly essential, but existing research, which has focused on relatively constrained data sets, offers little guidance concerning the effective use of diverse corpora, partitioned corpora, adaptive learning, model fusion, and similar questions. A primary objective will be to develop predictive assessment tools for evaluating the performance effects of changes in a LM without the need for extensive hands-on experimentation with the altered LMs. These predictive tools will be based upon recent work on a detection-theoretic approach to LM assessment and accompanying software developments.² Experimental verification of all assessment tools will take place using the vast and diverse resources of the Vincent Voice Library at Michigan State.

Of particular interest in applications with large search spaces is research into the effectiveness of topically-trained LMs. A second key objective of the Michigan State effort will be to research the effectiveness of such models and the ability to adapt existing models in real time in response to topics inferred from user queries. Such adaptation strategies would allow a more general LM to effectively be partitioned into subtopical LMs with results from various partitions weighted and fused in proportion to the strength of evidence provided by the query. An adaptive LM of this type could have wide-ranging application from search of data bases for scholarly research to security systems designed to monitor multiple speakers.

A third objective of the MSU group will be to research the *a posteriori* use of evolving recognition results from very-large-scale LMs in the construction and adaptation of smaller, more specific LMs. A decoding model similar to that used in modeling a communication channel with burst errors will be explored as a model for decoding models created from "dirty transcripts." Related to this issue is the question of performance enhancement of topical LMs by using such large-scale LMs as a front-end preprocessor for the weighting and cross-referencing of topics.

Speech Processing Laboratory at MSU. The primary engineering research site at Michigan State University (MSU) will be the Speech Processing Laboratory (SPL), a cognate lab in the Communications and Signal Processing Laboratories Consortium in the Department of Electrical and Computer Engineering. Much of the major equipment necessary for this work is already in place in the SPL and in the general computing environment in the College of Engineering. Some of the local facilities in the SPL include an array of networked (Windows NT 4) Pentium-III- and IV-class personal computers, SUN Ultra 10 and SUN Sparc 10/30 workstations (Solaris 7), and

¹ For example, see A. Gurijala, J.R. Deller, Jr., M.S. Seadle, and J.H.L. Hansen, "Speech watermarking through parametric modeling," *Proc. 2002 Int'l Conf. Spoken Language Processing*, Denver, Sep. 2002 (published on CD-ROM).

² J.R. Deller and Y.P. Yang, "A detection-theoretic approach to the assessment of language models used in speech recognition," *Natural Language Engineering* (in review)

*A decision-theoretic framework
for the evaluation of language models
used in speech recognizers*

J.R. DELLER, Jr.[†] and Y.P. YANG

Michigan State University

Department of Electrical & Computer Engineering / 2120 EB

East Lansing, MI 48824-1226 USA

deller@msu.edu

(Received 9 May 2003)

Abstract

An analytical method for design and performance analysis of language models (LM) is described, and an example interactive software tool based on the technique is demonstrated. The LM performance analysis does not require on-line simulation or experimentation with the recognition system in which the LM is to be employed. By exploiting parallels with signal detection theory, a profile of the LM as a function of the design parameters is given in a set of curves analogous to a receiver-operating-characteristic display.

[†] JD was supported in part by the National Science Foundation of the United States under Cooperative Agreement No. IBIS-9817485. Opinions, findings, or recommendations expressed are those of the authors and do not necessarily reflect the views of the NSF.

Contents

List of symbols used in the manuscript	iii
1 Introduction	1
2 Formulation	2
2.1 General recognition problem	2
2.2 Language detection problem	4
2.3 Integrating the LD and recognition problems	6
2.4 Estimating the probability densities	8
3 ROC-LM design tool	12
4 Applications and examples	14
4.1 Speech material, acoustic-level analysis, and task languages	14
4.2 LM design case studies	17
5 Summary and conclusions	25
References	26
Appendix: Artificial digit-string language	27
Endnotes	28

List of symbols used in the manuscript

ℓ	script lower-case ‘ell’
$\mathcal{E}$	script (calligraphic) upper-case E
$\mathcal{P}$	script (calligraphic) upper-case P
δ	lower-case delta (Greek)
η	lower-case eta (Greek)
Ω	upper-case Omega (Greek)
ω	lower-case omega (Greek)
ϕ	lower-case phi (Greek)
ρ	lower-case rho (Greek)
ξ	lower-case xi (Greek)

1 Introduction

Language models¹ (LMs) are widely used in contemporary speech recognition systems to structure the decoding process and to improve recognition performance [e.g. (Deller *et al.*, 2000; Rabiner and Juang, 1993)]. While many classes of LMs, including adaptive and hybrid languages [e.g. (Torres, *et al.*, 2002; Iyer *et al.*, 1997a; Kneser and Peters, 1997; Hwang *et al.*, 1994; Lau *et al.*, 1993; Meeter and Rohlick, 1993; Della Pietra *et al.*, 1992)], have been researched, a few simple models, notably the N -gram statistical models, prevail. One of most serious impediments to the development and testing of new LMs is the inability to assess their performance without time-consuming simulation or experimentation. This difficulty is directly and indirectly alluded to in many ways in a compendium of modern speech recognition challenges (Cole *et al.*, 1995). The paper by Iyer *et al.* (1997b) addresses this problem for the N -gram models by seeking global and local model features that are reliable predictors of LM performance.

In the present paper, we develop a new LM design / evaluation tool, the *ROC-LM design tool*, which does not require experimentation with the recognition system in which the LM is to be deployed. The general formulation underlying the ROC-LM tool exploits a parallel between the theory of binary signal detection and the LM problem. The LM must be ‘large’ in order to appropriately ‘cover’ the content of the task language (TL). A ‘large’ LM assures proper ‘detection’ of a valid sentence from the TL. Conversely, a large LM tends to ‘over-generate’ out-of-TL hypotheses, risking ‘false alarms’ (out-of-TL utterances accepted as grammatical). For a fixed figure of merit (ordinarily, some measure of recognition rate), analogous to signal-to-noise ratio parameterization in detection theory, a trace of ‘over-generation / coverage’ pairs may be mapped in a two-dimensional *receiver-operating-characteristic*-like (ROC) plane (Poor, 1994). More generally, the designer may simply use the tool to monitor performance in the plane as certain design features are varied in the LM. Unlike the binary detection problem, design ‘toward the northwest’ of the ‘ROC’ plane is (surprisingly) not always desirable in LM design (although implicitly sought in practical recognizers), and the ROC-LM tool offers a convenient method of tracking performance in light of this fact.

The purpose of the ROC-LM design tool is to provide a meaningful ‘first pass’ evaluation of a LM, perhaps as part of an interactive design procedure. Assessment of LMs is a complex problem, and no claim is made that the present developments provide definitive, uniquely objective, evaluations of general LMs. However, the ROC-LM tool, given reliable acoustic models and adherence of the speech material to certain (tested) assumptions, can provide useful comparisons among different topologies and statistical models, and guidance for parameter-setting in LM design. Moreover, the formal mathematical framework underlying the tool will serve as a basis for further development of the tools and related research. For example, while the present tool focuses on hidden-Markov-model (HMM) based acoustic analysis with concomitant unimodal Gaussian scoring, the general ideas can be applied to the Gaussian-mixture-model based evaluation and inherent LM suggested in the recent paper by Torres-Carrasquilla *et al.* (2002). It should also be noted that the

ROC-LM tool and related developments are quite different, even complementary, to the work by Iyer *et al.* (1997b) noted above. In contrast, the present strategy is not specific to the use of N -gram models, and it does not represent any specific attempt to modify or account for *local* features in a LM (e.g., reduction of confusability by path restructuring). On the contrary, the ROC-LM tool focuses exclusively on the process of *evaluating* the efficacy of a LM in the context of a given acoustic framework, and, in fact, may be combined with techniques suggested in Iyer’s work in the design or enhancement of a LM.

2 Formulation

2.1 General recognition problem

The developments in this paper are described in terms of word models for simplicity, but they are readily adapted to LMs with other symbolic outputs, for example phones or triphones, that would more commonly be used in large vocabulary, continuous-speech recognition systems.

We structure the speech-recognition problem into the signal detection framework of Fig. 1 (Poor, 1994). The *vocabulary*, V , is an exhaustive set of $2^{|V|}$ words (or another fundamental acoustic unit, say $V = \{w_1, \dots, w_{|V|}\}$, including a ‘null’ word, that comprises the TL. If the maximum sentence length is $S_{\max}$, there are on the order of $|L| = |V|^{S_{\max}}$ possible sentences (word strings), say, $L = \{s_1, \dots, s_{|L|}\}$. For each s_i , there will be a (assumed finite) number of acoustic representations (‘pronunciations’). In a speech recognizer, each of these is invariably represented by some string of acoustic features (often vector-valued) extracted from the speech that is to be recognized. Let a_j denote a *complete* string of features (i.e., *not* a single feature in time). Then we denote by $A = \{a_1, a_2, \dots, a_{|A|}\}$ the exhaustive set of acoustic strings used to utter the sentences in L . The set $A_i = \{a_{i_1}, a_{i_2}, \dots, a_{i_{|A_i|}}\} \subset A$, on the other hand, is used to denote the acoustic strings associated with sentence s_i . The pronunciations and acoustic strings are in one-to-one correspondence and need not be distinguished.

The underlying probability space in the problem is $(\Omega, 2^\Omega, P)$, where P reflects the ‘true’ (associated with the TL) distribution of pairs $\omega_{ij} = (s_i, a_j)$, the sample points of Ω . Formally, when a sentence s_i is ‘pronounced as’ a_j , then $\omega_{ij} \in \Omega$ is selected; this occurs with probability $P(\omega_{ij})$. Outcome ω_{ij} is then analyzed by the ‘transmitter’ using a *source classifier*, $\underline{c}_R(\omega_{ij}) = i$, which, The underscored symbol $\underline{c}_R$ indicates a random variable, and, although a r.v. should strictly take a numerical value, for clarity, we henceforth write as this and similar expressions as $\underline{c}_R(\omega_{ij}) = s_i$. At the ‘receiver’ (recognizer), the *observation* $\underline{o}(\omega_{ij}) = a_j$ (strictly, $\underline{o}(\omega_{ij}) = j$) is made. Based on $\underline{o}$, the recognition *decision rule* is

$$(1) \quad \underline{\delta}_R(a_j) = \arg \min_{i'} \ell(s_{i'}, a_j) = s_{i^*} \quad (\text{strictly } = i^*)$$

in which ℓ denotes the decision statistic (likelihood)

$$\ell(s_i, a_j) = \left[-|a_j|^{-1} \ln Q(a_j, s_i) \right] + \eta_g \left[-|s_i|^{-1} \ln Q(s_i) \right]$$

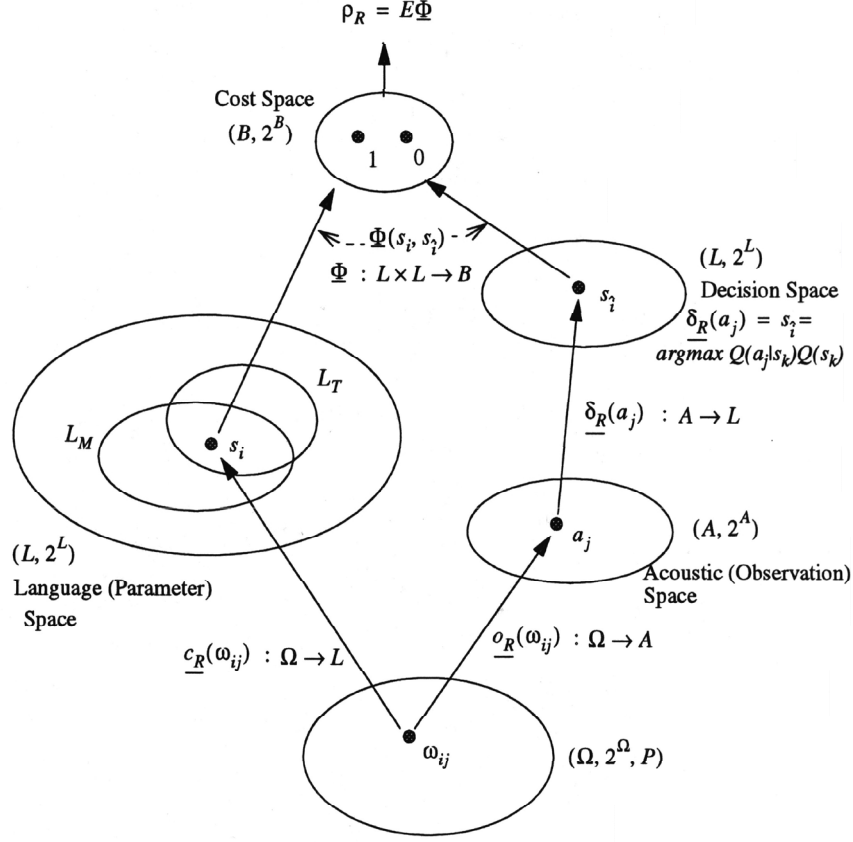


Fig. 1. Speech recognition structured as a conventional signal detection problem.

$$(2) \quad \stackrel{\text{def}}{=} \ell_a(a_j, s_i) + \eta_g \ell_g(s_i)$$

where $Q(\cdot)$ is a (probability) measure defined below. (Since all sentences are evaluated with respect to the same string, a_j , there is no apparent need to include the factor $|a_j|$ in (2). This normalization will be important below.) In (2), ℓ is decomposed into negative log likelihoods associated with the acoustic and language models (subscript ‘ g ’ on ‘ ℓ_g ’ to denote ‘grammar’). For generality, we also include a *weight*, η_g , to scale the relative importance of the grammar and acoustic likelihoods in the decision. Finally, the *cost function*, ϕ_R , determines the correctness of the outcome: $\phi_R(s_i, s_{i^*}) = \{1, \text{ if } i = i^*; 0, \text{ if } i \neq i^*\}$. The *recognition rate* for the system is given by

$$(3) \quad \rho_R = \mathcal{E} \phi_R$$

in which $\mathcal{E}$ denotes the statistical expectation. This rate, or further cost criteria, may be used to adjust the recognition models.

The distribution Q may be viewed as a ‘noisy probability measure’ over Ω ,

an estimate of the ‘true’ distribution P learned through the training process. In the language space L , there are two critical subsets associated with the distributions P and Q . Formally, the *task language* (TL), $L_T \subseteq L$, is defined as the subset $L_T = \{s_i | P(s_i) \neq 0; s_i \in L\}$, whereas the *language model* (LM), $L_M \subseteq L$, is $L_M = \{s_i | Q(s_i) \neq 0; s_i \in L\}$. Therefore $P(L_T, A) = 1$ and $Q(L_M, A) = 1$. To avoid tedious arguments, we henceforth assume that P and Q are approximately uniform over their support sets with respect to the sentence argument. Generalizations are discussed in (Yang, 1996).

2.2 Language detection problem

The heart of the speech recognition system is the decision rule $\underline{\delta}_R$. In turn, the ‘performance’ of the rule $\underline{\delta}_R$ reflects the quality of the training which is inherent in the distribution Q . The recognition performance can be characterized in terms of the question ‘How well does Q approximate P over the space $\Omega = L \times A$?’. In this work, however, we are interested in the quality of the LM, and this part of the training is reflected in the distribution $Q(s_i)$ over the set L and the corresponding term $Q(s_i)$, or $\ell_g(s_i)$, in $\underline{\delta}_R$. The *language detection* (LD) problem is structured similarly to the speech recognition problem (see Fig. 2) and is a *formal construct only*, which allows the quantified study of the LM as reflected in $Q(s_i)$ and $\underline{\delta}_R$.

In the LD problem, the initial probability space is $(L, 2^L, R)$ with L defined as above, and R is uniform over all elements in L . R is defined such that the distributions P and Q induced on L in the recognition problem (Fig. 1) satisfy $P(s_i, A) = R(s_i | L_T)$ and $Q(s_i, A) = R(s_i | L_M)$ for any $s_i \in L$, with the subsets L_T and L_M defined as above. For convenience, we also define subsets (see Fig. 3)

- (4) $L_{MT} \stackrel{\text{def}}{=} L_M \cap L_T$ (sentences grammatically correct in TL and LM),
- (5) $L_{M-} \stackrel{\text{def}}{=} L_M - L_{MT}$ (sentences grammatically correct in LM only).

In the LD problem, a test sentence s_i can be selected, with probability $R(s_i)$, from *anywhere* in L to evaluate the LM. In detection theory terms, the test sentence s_i is then analyzed by the ‘transmitter’ using a *source classifier*: $\underline{c}_L(s_i) = \underline{I}_T(s_i) = \{1, s_i \in L_T; 0, s_i \notin L_T\}$, where $\underline{I}_T$ is the indicator mapping for L_T . At the ‘receiver’ (detector), the *observation* is the sample sentence itself: $\underline{o}_L(s_i) = s_i$. The *decision rule* is $\underline{\delta}_L(s_i) = \underline{I}_M(s_i) = \{1, s_i \in L_M; 0, s_i \notin L_M\}$. For completeness (although the ‘output step’ is not used in the following developments): The *cost function* is the product $\underline{\phi}_L(c_L, \delta_L) = c_L \times \delta_L$, and the expectation of $\underline{\phi}_L$ with respect to the distribution R is the *LD rate*, $\hat{\rho}_L = \hat{\mathcal{E}}\underline{\phi}_L = R(L_{MT})$.

The LD problem is analogous to a classical binary channel problem [e.g., (Poor, 1994)] with the following hypothesis structure (letting $s_i \in L$ be a test sentence in a given trial):

$$(6) \quad H_0 : \underline{c}_L = 0 \quad (s_i \notin L_T) \quad \text{and} \quad H_1 : \underline{c}_L = 1 \quad (s_i \in L_T),$$

and the following decision structure:

$$(7) \quad D_0 : \underline{\delta}_L = 0 \quad (s_i \notin L_M) \quad \text{and} \quad D_1 : \underline{\delta}_L = 1 \quad (s_i \in L_M).$$

The ‘channel transition’ probabilities associated with this problem are: $R(D_1 | H_1) = R(\underline{\delta}_L = 1 | \underline{c}_L = 1)$ [conditional LD rate (hit-rate), related to LM ‘coverage’ described

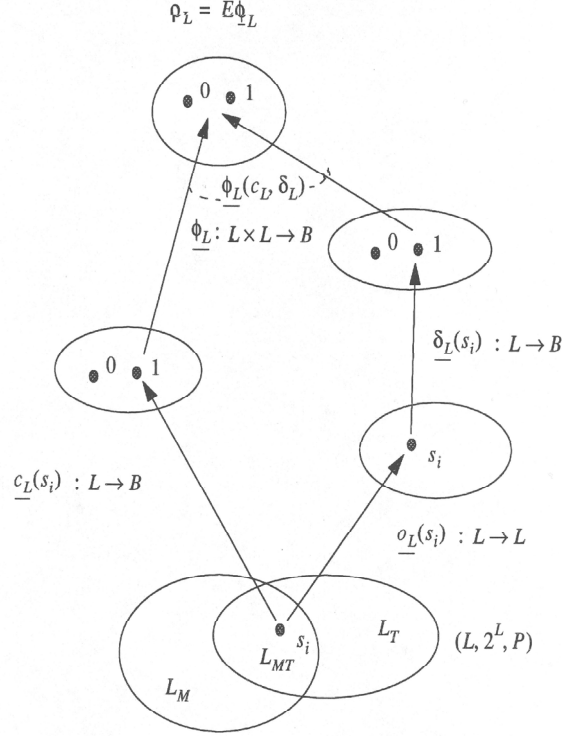


Fig. 2. The ‘LD problem’ viewed in a conventional signal detection framework.

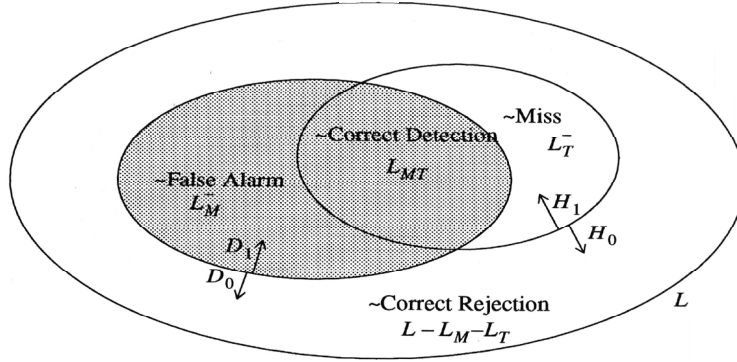


Fig. 3. Illustration of the four ‘hypothesis / decision’ conditions associated with the LD problem: detection, false-alarm, miss and correct rejection. It is important to recognize that the ‘LD problem’ is a conceptual framework for evaluating the LM. In the *use* of the LM in practice, the condition corresponding to neither H_0 (an utterance outside L_T is spoken) nor D_0 (a sentence outside the LM is recognized) can occur.

below]; $R(D_1|H_0) = R(\underline{\delta}_L = 1|\underline{c}_L = 0)$ (conditional false-alarm rate, related to LM ‘over-generation’ described below); $R(D_0|H_1) = R(\underline{\delta}_L = 0|\underline{c}_L = 1)$ (conditional miss rate); and $R(D_0|H_0) = R(\underline{\delta}_L = 0|\underline{c}_L = 0)$ (conditional correct rejection rate). Figure 3 illustrates the set relationships among these four probabilities in the language space L . Note that these channel probabilities capture the ‘performance’ of the distribution Q with respect to the LM, since it is upon Q (via $\underline{\delta}_R$) that the decisions are based.

The analysis of the LM model is based on an ROC-like characterization of the LD problem. The impetus for using ROC analysis in LM design derives from a design trade-off, analogous to the ‘detection / false-alarm’ phenomenon that occurs in signal detection. A LM needs to be ‘flexible’ (L_M sufficiently large) so that the complete set of TL sentences can be hypothesized during recognition. However, if the model is so flexible ($|L_M| \gg |L_T|$) that many sentences outside of L_T are hypothesized, the computation cost is increased and the recognition rate is decreased.

In these heuristic terms, the performance of a LM is mathematically described by two parameters, ‘coverage’ and ‘over-generation.’ The *coverage*, C_M , of the LM is exactly equivalent to the conditional hit rate of the LD problem and it reflects the probability that a sentence in the TL will be hypothesized (‘covered’ in the sense of Fig. 3) by the LM. Note, therefore, that

$$(8) \quad C_M \stackrel{\text{def}}{=} R(D_1|H_1) = R(\underline{\delta}_L = 1|\underline{c}_L = 1) = R(L_M|L_T) = P(L_M) = P(L_{MT}).$$

The *over-generation*, O_M , is defined to be an *a posteriori* false alarm rate

$$(9) \quad O_M \stackrel{\text{def}}{=} R(\underline{c}_L = 0|\underline{\delta}_L = 1) = R(H_0|D_1).$$

O_M differs from the proper false alarm rate $R(D_1|H_0)$ only by a scale factor, which, in turn, avoids the occurrence of very small numbers in the analysis.

Having defined the key measures O_M and C_M , we abandon the use of distribution R and return to the exclusive use of P and its ‘estimate’ Q . The next task is to integrate the LD and general recognition problems.

2.3 Integrating the LD and recognition problems

It is not difficult, beginning with (3) and using definition (8), to show that

$$(10) \quad \rho_R = C_M \sum_{s_i \in L_{MT}} P(\underline{\delta}_R = s_i|\underline{c}_R = s_i)P(\underline{c}_R = s_i|L_{MT})$$

Practical computation of this rate, and further integration of the LD results above, requires further characterization of the decision statistic (likelihood) introduced in (2).

Experimental evidence (Yang, 1996) suggests a simple model for the likelihood $\ell(s_i, a_j)$ that is amenable to the present purpose. We avoid unnecessary formality and simply assert that the likelihood values over the sentences in the LM can be modeled as outcomes of one of three r.v.s, the selection of which depends on certain conditions. It is important to understand that these r.v. models are for analysis and design purposes only. In particular, knowledge of the conditions under which they apply would be unknown in the practical deployment of the LM. The r.v. models

are:

$$(11) \quad \underline{\ell}_{1,1} \sim f_{1,1}(\cdot) \quad (\text{models } \ell(s_i, a_j) \text{ when } s_i \in L_{MT} \text{ and } a_j \in A_i);$$

$$(12) \quad \underline{\ell}_{1,0} \sim f_{1,0}(\cdot) \quad (\text{models } \ell(s_i, a_j) \text{ when } s_i \in L_{MT} \text{ and } a_j \in \bar{A}_i);$$

$$(13) \quad \underline{\ell}_{0,0} \sim f_{0,0}(\cdot) \quad (\text{models } \ell(s_i, a_j) \text{ when } s_i \in L_{M-} \text{ and } a_j \in \bar{A}_i).$$

The notation $\underline{\ell}_{1,1} \sim f_{1,1}$, for example, means that r.v. $\underline{\ell}_{1,1}$ generates values governed by probability density function (p.d.f.) $f_{1,1}$. Note that either (12) or (13) is used when the sentence hypothesized is an element of L_{MT} . The significance is that s_i is grammatical in *both* the LM and the TL (s_i is a ‘covered’ element of the TL), hence represents a sentence that is conceivably uttered. On the other hand, (13) is the appropriate model when s_i is an element of L_{M-} (the LM *only*), corresponding to an ‘overgenerated’ sentence that is theoretically never uttered. In each case, the initial subscript on the r.v. indicates whether the hypothesized sentence is grammatical in the TL, that is, whether $s_i \in L_{MT}$ (subscript 1), or not, ($s_i \in L_{M-}$, subscript 0). These conditions correspond to hypotheses H_1 and H_0 of (6), respectively. Further recall that A_i denotes the set of acoustic strings corresponding to hypothesized sentence s_i , $A_i \stackrel{\text{def}}{=} \{a_j | P(a_j, s_i) > 0\}$. We similarly define $\bar{A}_i$ appearing in (12) and (13) to be the complementary set of acoustic strings *not* corresponding to sentence s_i , $\bar{A}_i \stackrel{\text{def}}{=} \{a_j | P(a_j, s_i) = 0\}$. The second of the two subscripts on the r.v.s in (11) – (13) indicates whether the acoustic string being evaluated corresponds to the hypothesized sentence, that is, whether $a_j \in A_i$ (subscript 1), or not ($a_j \in \bar{A}_i$, subscript 0). It is worth noting that a sentence that is grammatically proper *in the TL only* (L_{T-}) would not be evaluated since it would *not* be hypothesized by the LM.

The simple conditional r.v. model of the total likelihood is the consequence of the appropriateness of similar models of the language and acoustic evaluations. In the ‘grammar-level’ evaluation of an hypothesized sentence, the likelihood outcomes, according to empirical results, can be modeled by two r.v.s:

$$(14) \quad \underline{\ell}_{g,1} \sim f_{g,1}(\cdot) \quad (\text{models grammar evaluations when } s_i \in L_{MT});$$

$$(15) \quad \underline{\ell}_{g,0} \sim f_{g,0}(\cdot) \quad (\text{models grammar evaluations when } s_i \in L_{M-}).$$

The latter is appropriate when the sentence being hypothesized cannot be generated by the TL (note that it *must* be generated by the LM), corresponding to H_0 in (6); the former is conditioned upon knowledge, corresponding to H_1 in (6), that the evaluated sentence is TL-grammatical.

Finally, and not surprisingly, the acoustical evaluations depend on the correctness of the hypothesized sentence rather than any specific grammatical considerations.³ For present purposes, two r.v. models are required

$$(16) \quad \underline{\ell}_{a,1} \sim f_{a,1}(\cdot) \quad (\text{models acoustic evaluations when } a_j \in A_i);$$

$$(17) \quad \underline{\ell}_{a,0} \sim f_{a,0}(\cdot) \quad (\text{models acoustic evaluations when } a_j \in \bar{A}_i).$$

The former is appropriate when the a_j represents an utterance of the sentence being hypothesized; the latter when the hypothesized sentence is incorrect. Note that the latter must always be used when $s_i \in L_{M-}$.

Now, assuming that the acoustic and language information can be used to find the relevant p.d.f.s for the total likelihoods, it is shown in (Yang, 1996) that these

p.d.f.s, the i.i.d. assumptions on the component r.v.s, and the assumption of approximate uniformity of sentence occurrence, lead to the following expression for the recognition rate:

$$\begin{aligned}
 \rho_R &= P(L_{MT}) \int_0^\infty f_{1,1}(\xi) [F_{00}(\xi)]^{|L_M|-1} [F_{1,0}(\xi)]^{|L_{MT}|-1} d\xi \\
 (18) \quad &= C_M \int_0^\infty f_{1,1}(\xi) [F_{00}(\xi)]^{|L_M|-1} [F_{1,0}(\xi)]^{|L_{MT}|-1} d\xi
 \end{aligned}$$

where

$$(19) \quad F_{0,0}(\xi) \stackrel{\text{def}}{=} \left[\int_\xi^\infty f_{0,0}(\xi') d\xi' \right] \quad \text{and} \quad F_{1,0}(\xi) \stackrel{\text{def}}{=} \int_\xi^\infty f_{1,0}(\xi'') d\xi''.$$

With equally-likely sentences, we may write that $|L'| \approx Q(L')|L_M|$ for any subset $L' \subseteq L_M$, and that $|L'| \approx P(L')|L_T|$ for any subset $L' \subseteq L_T$. In these terms,

$$(20) \quad |L_{MT}| = (1 - O_M) \cdot |L_M| = C_M \cdot |L_T|,$$

$$(21) \quad |L_M| = \frac{C_M}{1 - O_M} |L_T|,$$

$$(22) \quad \text{and } O_M = 1 - \frac{C_M}{|L_M|} |L_T|.$$

Similarly,

$$(23) \quad |L_{M-}| = |L_M| - |L_{MT}| = \frac{C_M}{1 - O_M} |L_T| - C_M \cdot |L_T| = \frac{C_M O_M}{1 - O_M} |L_T|.$$

Consequently (18) can be expressed in terms of the over-generation, coverage, and size of the TL by writing the exponents $|L_{M-}|$ and $|L_{MT}| - 1$ in terms of (23) and (21), respectively.

ρ_R is the performance measure used in assessing the quality of a LM below. Note that ρ_R is a *very conservative measure* in that it expresses the fraction of *perfectly recognized* sentences. More liberal measures, such as rate of sentences with one word substitutions, can be derived in a similar manner. We now turn to the task of estimating the density functions necessary to compute ρ_R .

2.4 Estimating the probability densities

2.4.1 Acoustic likelihood

This subsection further characterizes the acoustic likelihoods, (16) and (17), and explains how their density functions are estimated in the ROC-LM tool.

The performance of any recognition system depends critically upon the quality of the acoustic-level interface with the speech. In assessing a LM with the ROC-LM tool, the training of acoustic models is assumed to be fixed and characterized parametrically as described below. In this work, discrete-symbol, whole-word, HMMs are assumed to model the speech at the acoustic level, but this represents experimental convenience rather than a restriction of the method. In the following discussion, therefore, we speak in terms of word-level HMMs. The analysis remains fundamentally the same if a model for other unit such as a phoneme is employed.

There are seven important acoustic parameters ‘at the word level,’ and four related ‘sentence-level’ parameters, that are central to the acoustic partition of the

tool. We denote these seven and four numbers as elements of *acoustic parameter sets* $\mathcal{P}_a^w$ and $\mathcal{P}_a$, respectively,

$$(24) \quad \mathcal{P}_a^w \stackrel{\text{def}}{=} \{p_{a,0}^w, p_{a,1}^w, \mu_{a,0}^w, \sigma_{a,0}^w, \mu_{a,1}^w, \sigma_{a,1}^w, \epsilon\} \quad \text{and} \quad \mathcal{P}_a \stackrel{\text{def}}{=} \{\mu_{a,0}, \sigma_{a,0}, \mu_{a,1}, \sigma_{a,1}\}$$

The superscript w is a reminder that these quantities pertain to *word-level* models. The sets $\mathcal{P}_a^w$ and $\mathcal{P}_a$ are assumed fixed once LM design or evaluation begins.

To understand the meaning of these acoustic parameters, let us suppose that a (unknown) sentence represented by acoustic string a_j is determined, according to decision rule (1), to contain the word w_i . Let a_j^i denote the substring of a_j that is appropriated to the HMM for w_i in the maximum-likelihood decoding. The number $p_{a,1}^w$ is the probability that the acoustic model for w_i is capable of actually generating a_j^i given that w_i is the correct word, while $p_{a,0}^w$ is the similar quantity when w_i represents a word substitution. $p_{a,1}^w$ reflects the quality of the training in the sense that $1 - p_{a,1}^w$ is essentially the probability of occurrence of one or more untrained symbols in the HMM for w_i . For a sufficiently large vocabulary, good training will result in $p_{a,1}^w \gg p_{a,0}^w$. The *per-symbol* acoustic-level likelihood, say

$$(25) \quad \ell_a(a_j^i | w_i) = -|a_j^i|^{-1} \ln P(a_j^i | w_i),$$

has been found empirically to be relatively independent of w_i except for the dependency upon the correctness of the word (Yang, 1996). In the following we denote by $\ell_{a,1}^w$ and $\ell_{a,0}^w$ the r.v.s which model the word-level likelihood, conditioned upon w_i being correct and incorrect, respectively. When w_i is correct (and the HMM *can* produce a_j^i), $\ell_{a,1}^w$ is approximately Gaussian distributed with mean $\mu_{a,1}^w$ and variance $[\sigma_{a,1}^w]^2$, whereas when w_i represents a word substitution (and the HMM can produce a_j^i), then $\ell_{a,0}^w$ is Gaussian with mean $\mu_{a,0}^w \gg \mu_{a,1}^w$ and variance $[\sigma_{a,0}^w]^2 > [\sigma_{a,1}^w]^2$. Consequently, the p.d.f.s for $\ell_{a,0}^w$ and $\ell_{a,1}^w$ are,

$$(26) \quad f_{a,0}^w(\xi) \approx (1 - p_{a,0}^w) \delta(\xi - \epsilon) + p_{a,0}^w \mathcal{N}(\xi; \mu_{a,0}^w, \sigma_{a,0}^w)$$

$$(27) \quad f_{a,1}^w(\xi) \approx (1 - p_{a,1}^w) \delta(\xi - \epsilon) + p_{a,1}^w \mathcal{N}(\xi; \mu_{a,1}^w, \sigma_{a,1}^w)$$

respectively, in which $\mathcal{N}$ designates the Gaussian (normal) density with the parameters specified as arguments. The Dirac impulse function, δ , appears in each density to account for the discrete probability [either $(1 - p_{a,0}^w)$ or $(1 - p_{a,1}^w)$] that the string a_j^i cannot be produced by the HMM. In this case a ‘penalty’ likelihood, ϵ (which is theoretically infinite), is assigned to this part of the path.

In the ROC-LM system, the linguistic unit to be recognized is a sentence rather than a word. The derivation of an approximate statistical model for the sentence-level *acoustic* likelihood, $\ell_a(s_i, a_j)$ [see (2)], based on the word-level models above, is a straightforward but tedious exercise in combinatorics. Let $L_M^{(d)}$ indicate the set of all sentences in the LM that differ from the spoken sentence by exactly d words. For a sentence of length S words, the resulting conditional density for $\ell_a(a_j | s_i)$, is given by (Yang, 1996),

$$(28) \quad f_a(\xi | L_M^{(d)}) = \begin{cases} \sum_{k=0}^S \mathcal{N}(\xi; k\epsilon + S_k \mu_{a,1}^w, \sqrt{S_k} \sigma_{a,1}^w) \beta_k^S [1 - p_{a,1}^w]^k [p_{a,1}^w]^{S_k} & , d = 0 \\ \sum_{k_0=0}^d \sum_{k_1=0}^{S-d} \mathcal{N}(\xi; \mu_{a,0}^w(d), \sigma_{a,0}^w(d)) \epsilon_{k_0}(d) \epsilon_{k_1}(d) & , d \neq 0 \end{cases}$$

for any $d \in [0, S]$, where, $k_0 \in [0, d]$ ($k_1 \in [0, S - d]$) is the number of times $\ell_{a,0}$

$(\ell_{a,1})$ takes the value ϵ [see (26) and (27)], and

$$\begin{aligned}
S_k &= S - k \\
\mu_{a,0}(d) &= (k_0 + k_1)\epsilon + (d - k_0)\mu_{a,0}^w + (S - d - k_1)\mu_{a,1}^w \\
\sigma_{a,0}(d) &= \sqrt{(d - k_0)[\sigma_{a,0}^w]^2 + (S - d - k_1)[\sigma_{a,1}^w]^2} \\
\epsilon_{k_0}(d) &= \beta_{k_0}^d [1 - p_{a,0}^w]^{k_0} [p_{a,0}^w]^{d-k_0} \\
&\quad \text{(probability of } k_0 \text{ occurrences of } \ell_{a,0}^w = \epsilon \text{ in } d \text{ outcomes)} \\
\epsilon_{k_1}(d) &= \beta_{k_1}^{S-d} [1 - p_{a,1}^w]^{k_1} [p_{a,1}^w]^{S-d-k_1} \\
&\quad \text{(probability of } k_1 \text{ occurrences of } \ell_{a,1}^w = \epsilon \text{ in } S - d \text{ outcomes)} \\
\beta_k^S &= \frac{S!}{k!(S-k)!} \quad \text{(binomial coefficient).}
\end{aligned}$$

It follows that the p.d.f.s for the (sentence-level) acoustic likelihoods as defined in (16) and (17) are given by (all quantities defined immediately above)

$$(29) \ell_{a,0} \sim f_{a,0}(\xi) = \sum_{d=1}^S Q(L_M^{(d)}) \sum_{k_0=0}^d \sum_{k_1=0}^{S-d} \mathcal{N}(\xi; \mu_{a,0}(d), \sigma_{a,0}(d)) \epsilon_{k_0}(d) \epsilon_{k_1}(d)$$

$$(30) \ell_{a,1} \sim f_{a,1}(\xi) = \sum_{k=0}^S \mathcal{N}(\xi; k\epsilon + S_k \mu_{a,1}^w, \sqrt{S_k} \sigma_{a,1}^w) \beta_k^S [1 - p_{a,1}^w]^k [p_{a,1}^w]^{S-k}.$$

The probabilities $Q_M(L_M^{(d)})$ can be estimated in practice using the average branching factor for the LM, say, $B \approx |L_M|^{1/S}$, from which we may write

$$(31) \quad Q(L_M^{(d)}) \approx \beta_d^S \frac{(B-1)^d}{B^S} \approx \beta_d^S \frac{(|L_M|^{1/S} - 1)^d}{|L_M|}$$

(Note: $\sum_{d=0}^S Q(L_M^{(d)}) = 1$.)

As implied by these r.v. models, in practice, the acoustic likelihoods tend to be relatively independent of the sentence evaluated (except, of course, for the conditions of correct and incorrect hypotheses). Further, each of the two r.v.s used to model $\ell_a(s_i, a_j)$ is approximately Gaussian distributed, a result that is not surprising in light of (29) and (30) and the fact that the likelihood outcomes for various values of d would tend to be uncorrelated. Therefore a straightforward way to obtain $f_{a,0}(\cdot)$ and $f_{a,1}(\cdot)$ is to estimate the parameters of the set $\mathcal{P}_a$ directly by experimentation.

2.4.2 Grammar likelihood

O_M and C_M are defined as conditional probabilities in (8) and (9). C_M is based on the hypothesis structure, H_1 and H_0 , [see (6)], while O_M is based on decision structure, D_1 and D_0 [see (7)]. Therefore, characterization of a LM using O_M and C_M requires two sets of design sentences. Let X_T denote the set containing TL-grammatically-correct sentences which are randomly chosen from the TL, and X_M denote a set of sentences randomly generated by the LM.

To obtain C_M , X_T is input to the LM to be evaluated. Suppose that there are N_0 sentences in X_T that are not elements of L_M or that do not have finite likelihoods.

Then C_M is estimated as

$$(32) \quad C_M \approx 1 - (N_0/|X_T|).$$

To obtain O_M , sentences from the set X_M are fed into an observer which can detect whether they are elements of L_T . If N_1 sentences cannot be generated by the TL, then, in principle, O_M can be approximated by

$$(33) \quad O_M \approx N_1/|X_M|.$$

This simple method, however, is unrealistic in many speech recognition problems since the grammar associated with the TL may not be explicitly known. Equation (22) provides an alternative method for estimating O_M and $f_{0,0}$, given knowledge of C_M , $|L_T|$, and $|L_M|$. Let us assume that $|L_T|$ can be estimated by the designer, and that C_M is estimated as above. Then the problem of estimating O_M becomes one of estimating $|L_M|$. For a small LM, $|L_M|$ can be obtained by simply enumerating all possible sentences. For a large LM, $|L_M|$ can be estimated by averaging the log probabilities of sentences in X_M ,

$$(34) \quad |L_M| \approx \exp \left\{ -|X_M|^{-1} \sum_{i=1}^{|X_M|} \ln Q(s_i) \right\}.$$

Now let us consider the grammar likelihood models $\ell_{g,1}$ and $\ell_{g,0}$ and corresponding p.d.f.s. The estimation of p.d.f.s in this case is straightforward. In addition to the two r.v.s $\ell_{g,1}$ and $\ell_{g,0}$, we will have a brief need for a composite grammar likelihood model,

$$(35) \quad \ell_{g,M}(s_i) = -|s_i|^{-1} \ln Q(s_i) \sim f_{g,M}(\xi) = Q(L_{M-})f_{g,0}(\xi) + Q(L_{MT})f_{g,1}(\xi).$$

The p.d.f. $f_{g,1}$ is determined experimentally by creating a likelihood histogram for the $|X_T| - N_0$ sentences favorably evaluated in estimating C_M in (32) above. A normal p.d.f. is fit to this histogram, inherently producing the statistics $\mu_{g,1}$ and $\sigma_{g,1}$. $f_{g,M}$ can be determined similarly. Then $f_{g,0}$ is estimated using (35):

$$(36) \quad f_{g,0}(\xi) = \frac{f_{g,M}(\xi) - (1 - O_M)f_{g,1}(\xi)}{O_M} \approx \mathcal{N}(\xi; \mu_{g,0}, \sigma_{g,0}).$$

where the approximation indicates fitting to a Gaussian p.d.f. Because of the Gaussian assumptions, the grammar likelihoods are characterized by the simple parameter set

$$(37) \quad \mathcal{P}_g = \{\mu_{g,0}, \sigma_{g,0}, \mu_{g,1}, \sigma_{g,1}\}.$$

2.4.3 Total likelihood and related density functions

According to (2), the total likelihood associated with a pair (s_i, a_j) is the sum of the acoustic and grammar likelihoods. Because of the assumed independence of the two evaluations, we may write, for example,

$$(38) \quad \ell_{1,1} = \ell_{a,1} + \ell_{g,1} \sim f_{1,1}(\xi) = f_{1,1}(\xi_a + \eta_g \ell_g) = \int_0^\infty f_{a,1}(\xi_a) f_{g,1}\left(\frac{\xi - \xi_a}{\eta_g}\right) d\xi_a,$$

and similar expressions for $f_{1,0}(\cdot)$ and $f_{0,0}(\cdot)$. The Gaussian (plus impulses) forms of the p.d.f.s permit simple manipulations of means and variances to produce the desired results. Once the appropriate density functions are estimated, the recognition rate (or similar figure of merit) can be obtained using (18).

The outcome rate computation (18) is clearly dependent upon the weight value η_g employed in (2), and also the sentence length ‘ S ’ assumed in deriving the acoustic densities. The choice of η_g reflects the reliability of the grammar information in the LM relative to that of the acoustic knowledge. When sentences are truly equally-likely in L_M , then η_g should theoretically be set to zero as the product $\eta_g \ell_g$ otherwise produces a bias that contributes nothing but additional computation. The uniform sentence distribution, however, is usually a necessary, but reasonable, assumption in the absence of good modeling information, rather than reflective of precise knowledge about the grammar. In this case, the η_g can be used to maximize the benefit of the grammar likelihood. Concerning the dependence on ‘ S ,’ the acoustic density derivation can be modified to accommodate statistical models or precise knowledge of variability in the sentence lengths in the TL. However, such knowledge is usually not available. It is reasonable to assume that an ‘average’ value for S will provide meaningful relative performance measures across TL configurations, especially as the sizes of L_M and L_T increase. Limited experimental evidence affirms the validity of this assumption, and future research and deployment of the design tool will produce further evidence on this point and perhaps ways to more flexibly integrate the sentence-length information into the analysis.

3 ROC-LM design tool

This section will overview the salient features and operation of the example ROC-LM tool. Details are found in (Yang, 1996). The analysis performed by the ROC-LM design tool in evaluating a family of LMs is summarized on the display referred to as the ‘ROC-LM plane.’ Each point in the ROC-LM plane represents a (O_M, C_M) pair, hence a nominal LM with respect to a nominal TL.

Performance measures [in the present case, ρ_R using (18)] are computed for each of grid points on the ROC-LM plane, accounting for the given acoustic environments. Contour plots parameterized by performance measures are then plotted in the ROC-LM plane. Any specific LM to be employed with the given acoustic models can then be evaluated by computing its (O_M, C_M) pair and plotting it on the plane. The designer can easily observe the ‘motion’ of the model when its parameters are modified, and ‘move’ the LM to approach a design goal.

The complete ROC-LM design tool package consists of several software modules which are diagramed in Fig. 4. The top block includes the two-dimensional ROC-LM plane with coverage, C_M , as its vertical axis and over-generation, O_M , as its horizontal axis. The contour plots and design points are plotted in this plane as described above. For present purposes, an X-window version of the ROC-LM design tool was implemented with MATLAB graphics support.

As an important research instrument, the language portion of the ROC-LM design tool is equipped with a module that can train and evaluate N -gram models for $N = 2$ and 3 (bigram and trigram), and hybrids of bi- and tri-gram models (‘bi-tri-gram’). This subroutine is further described in the context of ‘Case II’ in Section 4.2.

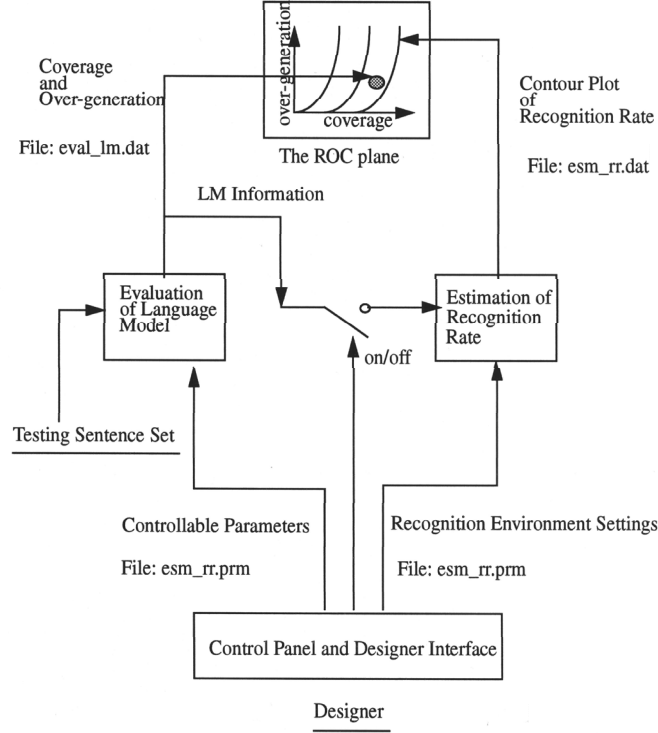


Fig. 4. Four blocks of the ROC-LM design tool: Display window, Recognition rate evaluator, LM evaluator, and the Control panel & designer interface.

Based on the developments above, the basic operation of the ROC-LM design tool is summarized as follows:

Initial parameter estimation: The parameters characterizing the acoustic environment [set $\mathcal{P}_a$ in (24)] are experimentally or otherwise determined. The means and variances of random variables $\ell_{a,0}$ and $\ell_{a,1}$ are obtained from histograms of acoustic evaluations. The size of the TL, $|L_T|$, and the average sentence length in the TL are measured or estimated. The relative weighting, η_g , of the language likelihood is specified (may be determined experimentally).

Analysis of changing LM topologies with invariant likelihood statistics: In this case, it is assumed that the underlying model of the language likelihoods (in particular $\mu_{g,0}$, $\sigma_{g,0}$, $\mu_{g,1}$, and $\sigma_{g,1}$) do not change significantly with changes in the LM structure. An $n \times n$ grid is conceptually superposed on the two-dimensional ROC-LM plane (e.g., $n = 25$). Each grid point represents a over-generation-coverage pair (O_M, C_M) . For a given LM (or set of LMs to be ‘averaged’), assumed to be statistically typical of the LMs to be tested in this analysis, the means and variances of random variables $\ell_{g,0}$ and $\ell_{g,1}$ are obtained from histograms of language evaluations. With these data, the recognition rate is calculated at each grid point according to (18). Contour plots of constant recognition rates are plotted on the

ROC chart for reference. For a LM to be evaluated, O_M and C_M are measured and the corresponding point is located on the ROC plane, thus yielding an immediate performance measure for that model. As various parameters and features of the LM are changed, a trajectory is made by connecting the locations of resulting (O_M, C_M) pairs on the ROC-LM plane.

Analysis of constant LM topologies with varying likelihood statistics: In this case, the nominal task is to ascertain the effects of varying statistical models of the language likelihoods (in particular $\mu_{g,0}$, $\sigma_{g,0}$, $\mu_{g,1}$, and $\sigma_{g,1}$), on the performance of a *fixed* LM topologies (in particular with fixed values of O_M and C_M). In principle, for each likelihood model to be studied, a set of contours parameterized by recognition rates is plotted in the ROC-LM plane. (For simplicity, consider comparing two likelihood models, each generating its own set of contours as illustrated in Fig. 5.) A ‘fixed’ LM topology, corresponding to a single (O_M, C_M) pair, will fall on a different contour for each likelihood model, indicating different performance measures for each model. For example, in Fig. 5, the language model labeled ‘LM₂’ corresponding to a *fixed* (O_M, C_M) pair (lighter gray marker), falls on the 93 percent rate contour (labeled ‘old 93’ – solid contour curve) under the ‘old’ likelihood model, and simultaneously the 97 percent rate contour (labeled ‘new 97’ – dotted contour) under the ‘new’ likelihood model. Clearly, the ‘new’ likelihood model is more discriminative since better performance is contained from the same LM structure. The improvement in recognition rate is not as large for the model labeled ‘LM₁’ [appearing on ‘old 93’ (solid) and ‘new 95’ (dotted) contours] underscoring the highly nonlinear relationships among the coverage, overgeneration, and likelihood structures of the model. It is further possible to assess the *apparent* change in the LM topology that is actually caused by the change in the likelihood model. For example, if we locate the nearest LM to LM₂ on the ‘new 93’ contour (because the recognition rate prior to the likelihood change is 93 percent), we see that a significant improvement in coverage and a small decrease in overgeneration are necessary to effect a change from 93 to 97 percent recognition rate in the new likelihood context. The ‘relative motion’ of model LM₁ in the ROC-LM plane is not as great. In general, the most favorable improvements are those that cause the most significant movements toward the ‘northwest’ corner of the ROC-LM plane ($O_M = 0$, $C_M = 1$), assuming that such movement is in the direction of improving recognition rate.

4 Applications and examples

4.1 Speech material, acoustic-level analysis, and task languages

Some design studies are presented to illustrate the use of the ROC-LM tool. In every example, the acoustic-level models are whole-word, six-state, discrete-symbol, HMMs with a Bakis topology. Feature analysis is based on eight mel-cepstral parameters and their differences, power, and differenced power. Features are computed on 256-point frames which overlap by 128 points. Twenty total mel-cepstral coefficients covering the 8 kHz Nyquist range are computed for each frame. The sym-

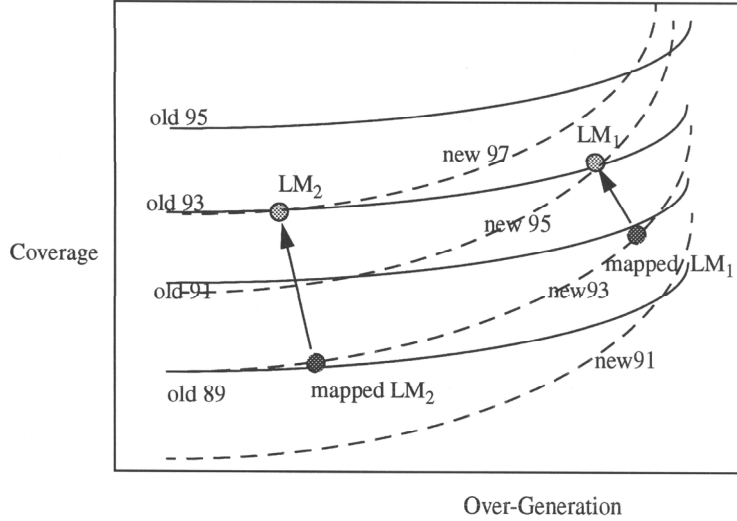


Fig. 5. An illustration of a ‘relatively’ changing contour plot with the topology of the LM unchanged, but probabilistic structure of the LM varied. The label ‘old 95,’ for example, represents the old contour curve for recognition rate 95 percent, while ‘new 95’ represents the performance with a change of the LM with new probabilities. Although LM_2 , for example, does not ‘move’ in the plane due to a change in probabilities, it can be viewed as having undergone a relative motion. In the new probability structure, LM_2 achieves 97 percent recognition rate, but only 93 percent before the change. The relative motion of LM_2 is shown as the mapping from 93 percent to 97 percent in the new probability setting. The relative motion is highly dependent upon location in the ROC-LM plane.

bols are drawn from a 128-point codebook according to customary methods. The Baum-Welch reestimation is used to train each HMM. For details on these standard acoustic-modeling techniques are found, for example, in (Deller *et al.*, 2000; Rabiner and Juang, 1993). Data for the experiments are taken from the TI-46 (Doddington and Schalk, 1981) and the TIMIT databases (NIST, 1988).

The acoustic parameters [see (24)] for the TI-46 and TIMIT databases are estimated using extensive experiments with material from the respective databases. In these experiments, known word boundaries (provided in TI-46, and inferred from transcription records with TIMIT) were used to estimate the density parameters for both ‘0’ and ‘1’ conditioning information. Not surprisingly, the TI-46 results reflect more consistent acoustic performance. The seven word-level acoustic parameters for TI-46 and TIMIT are:

$$(39) \quad \mathcal{P}_{a, \text{TI-46}}^w = \{5.2, 1.8, 0.01, 0.85, 1.1, 0.97, 20.1\} \text{ and}$$

$$(40) \quad \mathcal{P}_{a, \text{TIMIT}}^w = \{4.8, 2.6, 0.03, 0.85, 0.92, 0.93, 6.8\}.$$

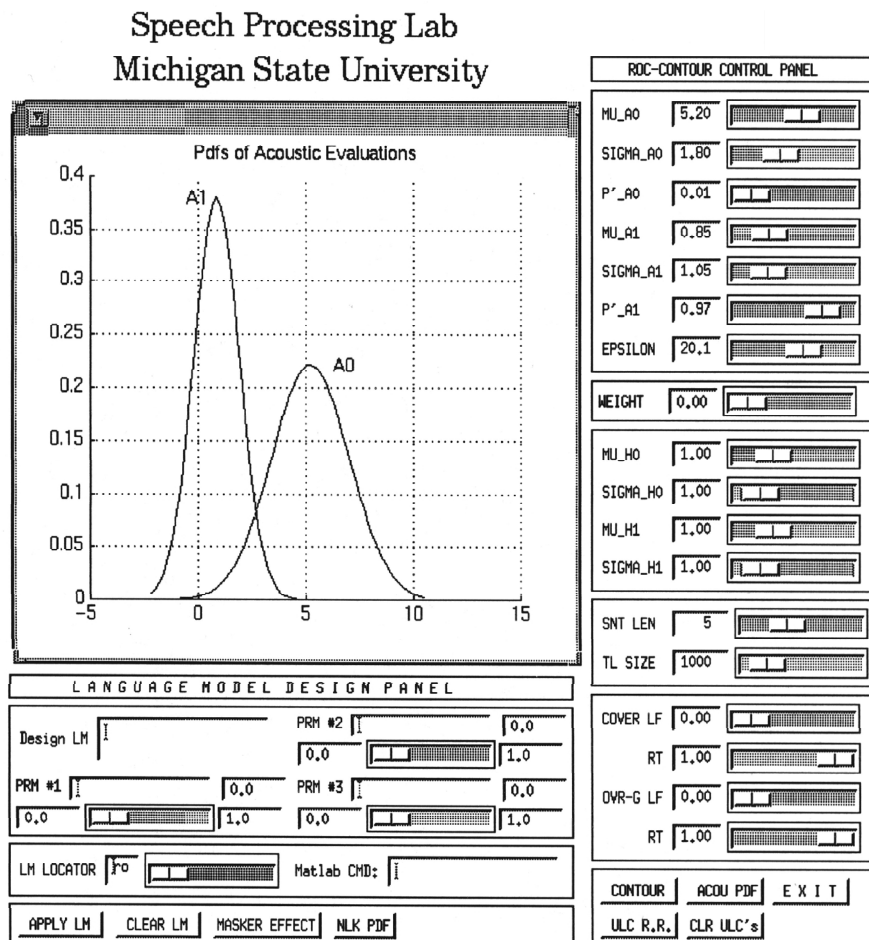


Fig. 6. By clicking on the button ACOU PDF, the two Gaussian p.d.f.s associated with the acoustic-level evaluations are shown on the LM-ROC WINDOW.

Figure 6 shows the experimental TI-46 density functions for the random variables $\mathcal{L}_{a,0}^w$ and $\mathcal{L}_{a,1}^w$. Initially, the penalty likelihood, ϵ , is set to 10.

Two TLs are employed in these experiments. The first involves an oral *spelling* task in which letters play the role of ‘words’ in the language and a complete spelling of a word comprises a ‘sentence.’ The grammar of the spelling TL is the set of rules for creating English words from alphabet symbols. Words (sentences) used for training and testing the LM were taken from the sports column of the *San Jose Mercury News*.

The second TL is an *artificial digit-string* (ADS) language described in the Appendix. The ADS language appears on page 621 of (Deller *et al.*, 2000) as a pedagogical illustration of the structure of a formal language. It is used here to demonstrate how the ROC-LM can be employed to find the optimal (or sub-optimal) settings for

an adaptive LM. The ADS language is useful in the assessment of the design tool because it is quantitatively well-defined from the lexical through pragmatic levels. Accordingly, the size and generation rules at each level can be exactly calculated by the ROC-LM tool, providing a precise TL against which to quantify the performance of any LM. For the spelling TL, the size and rules of the language are less-well specified, although an *ad hoc* procedure such as the use of a dictionary can be employed. The problem of accurately estimating the complete structure and size of the underlying TL becomes more difficult as the task becomes more complex.

4.2 LM design case studies

4.2.1 Introduction

Three LM design case studies are presented, each of which illustrates a different feature or use of the ROC-LM tool. The first case study involves the design of N -gram ($N = 2$ or 3) spelling LMs and explores the effects of the amount of training on the performance of the LM. The second case study extends the spelling LM problem to the design of a hybrid bi- / tri-gram language and considers issues of computational complexity. The third case study is concerned with the design of an adaptive LM.

These case studies will demonstrate the flexibility of the ROC-LM tool as a design aid. At the same time, it will be appreciated that the tool is an exploratory device to be used in an *ad hoc* manner. There are 21 adjustable parameters on the ROC-LM control panel [see Fig. 6], and there is no specific design scheme that assures the discovery of ‘optimal’ parameter settings. Frequently, it is true that the most effective parameter changes (in the sense of improving recognition rate) are those which move the LM (real or relative motion) toward the ‘northwest’ corner of the ROC-LM plane. This is generally in the direction normal to the local contour curves. This is not always the case, however, and the ROC-LM tool provides guidance in assessing the various tradeoffs in the process.

4.2.2 Case I: N -gram spelling LM (Effects of training set size)

We seek an N -gram ($N=2$ or $N=3$) LM and an assessment of the appropriate number of training sentences for an alphabet recognizer for the spelling TL. In spite of the small vocabulary size in this case (26 letters of the alphabet), recognition of individual alphabetic characters is often difficult due to the acoustic confusability of letters [e.g., Ch. 10 in (Deller *et al.*, 2000)]. An N -gram LM is often employed to approximate spelling rules and decrease entropy. The amount of training material for such a LM represents a trade-off for the designer. Too few training sentences result in insufficiency of ‘word’ concatenation information and causes low TL coverage, C_M . Meanwhile, too many training sentences result in generation of many extraneous letter combinations thus increasing overgeneration, O_M .

Bigram and trigram LMs are evaluated and compared on the ROC-LM plane.

$\mathcal{P}_{a, \text{TI-46}}^w$ [see (24) and (39)] is used to model the acoustic context. The penalty and weight values, ϵ and η_g , are initialized to zero. Since $\eta_g = 0$, the statistics $\mu_{g,0}$, $\sigma_{g,0}$, $\mu_{g,1}$ and $\sigma_{g,1}$ do not effect recognition rates and are arbitrarily initialized. The ‘sentence’ length S is set to the average word length in the training material. The TL size is assumed to be 1,000. The initial axes over which O_M and C_M are plotted are both set to view the full zero to one range. Once the acoustic information is initialized, we click on the button ACOU PDF to show the two conditional Gaussian density functions for the acoustic likelihoods $\ell_{a,0}^w$ and $\ell_{a,1}^w$ (see Fig. 6).

Following initialization, the embedded routine *bigram_train* is invoked. This routine conveys the acoustic parameter settings, the number of training sentences, and a random integer seed, from the ROC-LM control panel to the LM design module, then prepares a training context for the LM evaluation program. By clicking on the button APPLY LM, a circle at $(O_M, C_M) = (0.994, 0.482)$ appears on the ROC plane. To center the plot, we change the range of ROC plane by moving slider switches labeled COVER LF (coverage, ‘left’ axis extreme) to 0.44, and OVR-G LF to 0.992.

To test the effect of the number of training sentences on LM quality, the slider PRM #1, representing the number of training sentences, is moved through the sequence, 100, 190, 280, 370, 460, 560. APPLY LM is clicked each time PRM #1 is changed. The O_M, C_M trajectory appears in Fig. 7(a).

To test sensitivity of the random seed, PRM #2 is moved from 177 to 1,997, and LM LOCATOR is switched from ‘red circle’ to ‘red $\times$.’ The trajectory marked ‘ $\times$ ’ in Fig. 7(a) results as PRM #1 is moved backward through the sequence above.

Next, similar procedures are followed to explore a trigram LM using the routine *trigram_train*, with the exception that the initial value of PRM #1 is changed from 100 to 1000, and all subsequent values are increased by a factor 10. The results appear in Fig. 7(b).

Resulting bigram and trigram performances are plotted simultaneously in Fig. 7(c). It is observed that the trigram trace appears to the ‘west’ (smaller overgeneration) and initially slightly ‘southward’ (worse coverage) of that for the bigram LM. If the size of training set is limited, for instance at 550, the bigram LM achieves a better recognition rate than the trigram due to the bigram’s larger C_M . The trigram LM always achieves a lower O_M , which is benefits recognition rates if the training material is unlimited so that the trigram’s C_M achieves a comparable value to the that of the bigram. However, in this case, the prospect of achieving similar coverage with the trigram does not appear promising because of the flat contour plot, which implies the C_M dominates the recognition rate. If however, the trigram contour continues to rotate counter-clockwise with more training, the trigram LM may outperform the bigram in both measures of O_M and C_M as the bigram appears to have reached a point of diminishing returns with increased training.

The random seed represents different sets of training material. Not surprisingly, the smaller size of the training set, the more sensitive the result to the selection of the training material. For example, there is a 4.5 percent difference in recognition rate for a bigram LM when the two different sets of training material are of size 100. When the size is increased to 370, the difference is less than 1 percent.

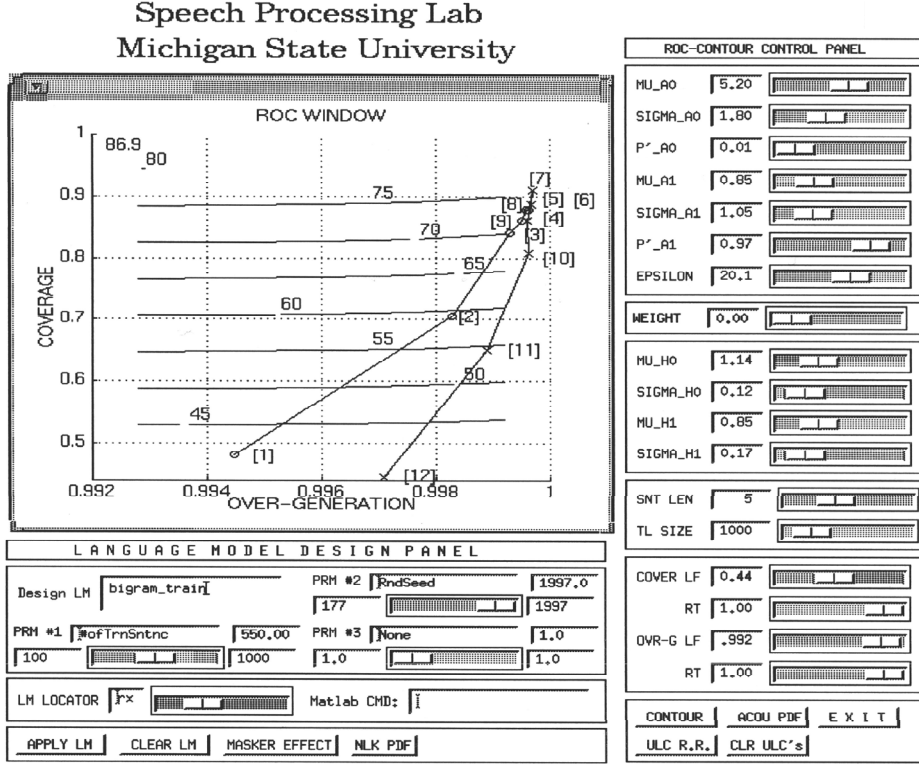


Fig. 7. Design Case I results. (a) Bigram performance when the slider PRM #1, representing the number of sentences in training context, is moved from 100, 190, through 550 with PRM #2 (random seed) at 177. A trajectory marked by circles is formed. The random seed is then changed to 1997, and PRM #1 moves backward from 550 to 100. Another trajectory marked by '5' is traced on the panel.

4.2.3 Case II: Hybrid bigram/trigram spelling LM (Complexity issues)

The task is to use the *bi-tri-gram_train* routine embedded in the LM-ROC tool to design an hybrid LM for the spelling task. There are three parameters in the *bi-tri-gram* module that control the p.d.f. of the likelihood measure, and the O_M and C_M of the LM. These three parameters are the *weight*, *penalty* and *pruning threshold*. The weight, η_{bt} (subscript 'bt' denoting 'bi-tri-(gram)'), controls the relative importance of the bi- and trigram outcomes. If ℓ_b and ℓ_t represent the grammar-level likelihoods using bigram and trigram LMs, respectively, then the hybrid likelihood, ℓ_g , is

$$(41) \quad \ell_g = (1 - \eta_{bt})\ell_b + \eta_{bt}\ell_t, \quad \text{with } 0 \leq \eta_{bt} \leq 1.$$

If a bigram representation of a message appears in the LM, but the trigram does not, then ℓ_t is set to the penalty value, ϵ_{bt} (theoretically infinite). Finally, if ℓ_g

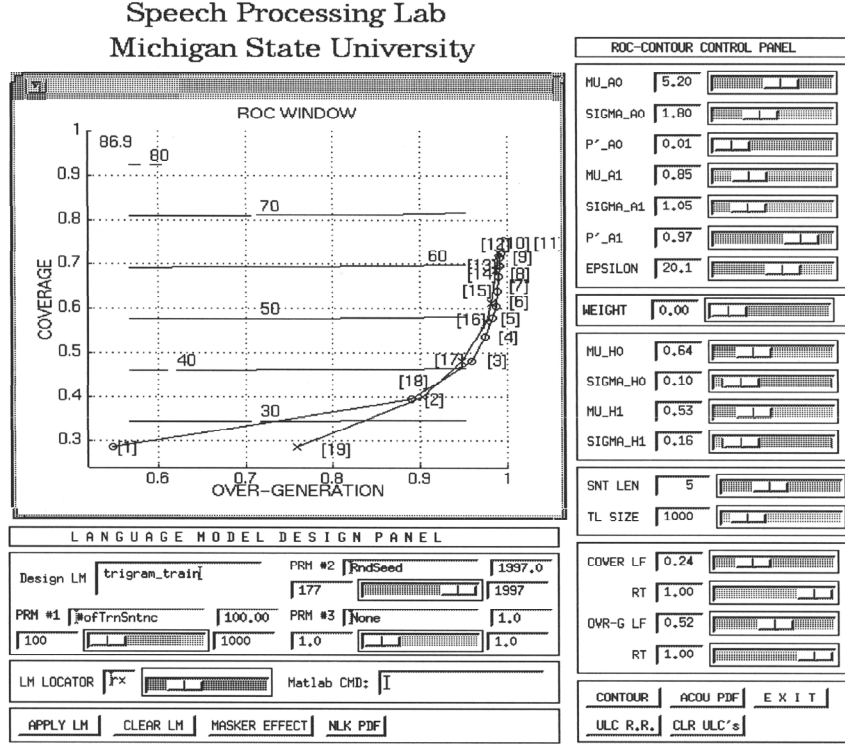


Fig. 7. [continued] (b) Similar experiment for the trigram model except that PRM #1 ranges from 100 to 1000.

is above the predefined pruning threshold, t_{bt} , then the hypothesized sentence is pruned.

Computational cost is an important consideration in many applications. If the stack size is infinite, then the computational cost of searching a LM is proportional to $|L_M|$. The threshold parameter t_{bt} employed in the routine *bi-tri-gram* is used to control the effective size of the LM, as we demonstrate below. Note that if $t_{bt} \equiv \infty$ (no pruning takes place), size of the hybrid LM is identical to that of the bigram, and the C_M values of the models are also equal. The same C_M and the same LM size imply the same O_M . Consequently, when the threshold parameter is infinite, the O_M and C_M values remain unchanged, and the LM remains at the same location in the ROC-LM plane, regardless of the weight η_{bt} and the penalty value ϵ_{bt} . Nevertheless, when both bigram and trigram models are present in the training context ($\eta_{bt} > 0$), the likelihood of a sentence evaluated against the bi-tri-gram LM is the linear combination of bigram and trigram evaluations as in (41). Although the O_M and C_M remain the same, the recognition rate contour plot is changed, a trajectory can be made by transforming the previous locations of the LMs to the

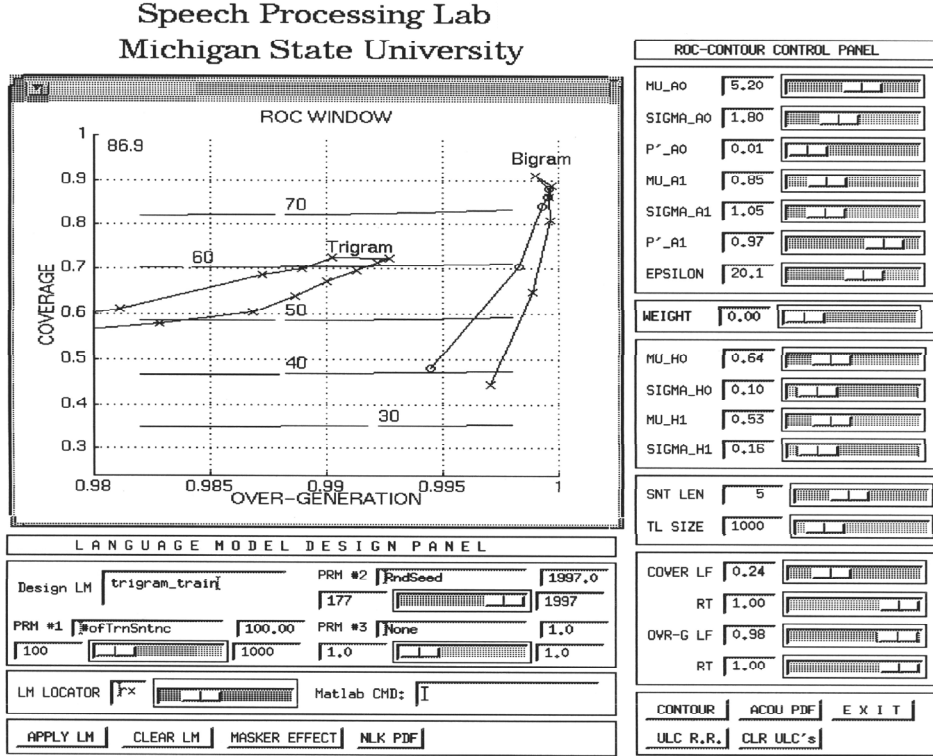


Fig. 7. [continued] (c) Comparison of bigram and trigram performance for upper range of overgeneration.

relative new locations for the new contours. The issue of ‘relative motion’ of the LM under such conditions is discussed in general in Section 3.

The basic setup and acoustic initialization procedures and parameters are identical to those in Case I. A pure trigram model ($\eta_{bt} = 1$) is initially employed which has a (O_M, C_M) coordinates at location ‘[1]’ in Fig. 8(a), and grammar statistics [recall (37)], $\mathcal{P}_g = \{5.61, 1.44, 1.70, 1.71\}$ which appear on the appropriate fields on the design panel. The pruning threshold is set to ‘infinity,’ $t_{bt} = 100$. A bigram model is then incorporated by sliding PRM #1, programmed to represent η_{bt} , from 1.0 to 0.66. After clicking on APPLY LM, the grammar statistics set is changed to $\mathcal{P}_g = \{4.17, 0.97, 1.46, 1.16\}$ but the O_M and C_M remain the same (see discussion above). To illustrate the ‘relative motion’ (see Section 3) of the LM under the changing likelihood structure, the contour plot is reestimated by clicking on CONTOUR. The newly-estimated contours are shown as the heavy lines on Fig. 8(a). Simultaneously, the ‘old’ LM is transformed to a ‘relative position’ with respect to the new contour plot, location ‘[2]’ on the panel in Fig. 8(a). The change is seen to be a detrimental one, since the ‘new’ recognition rate of about 85 percent can be

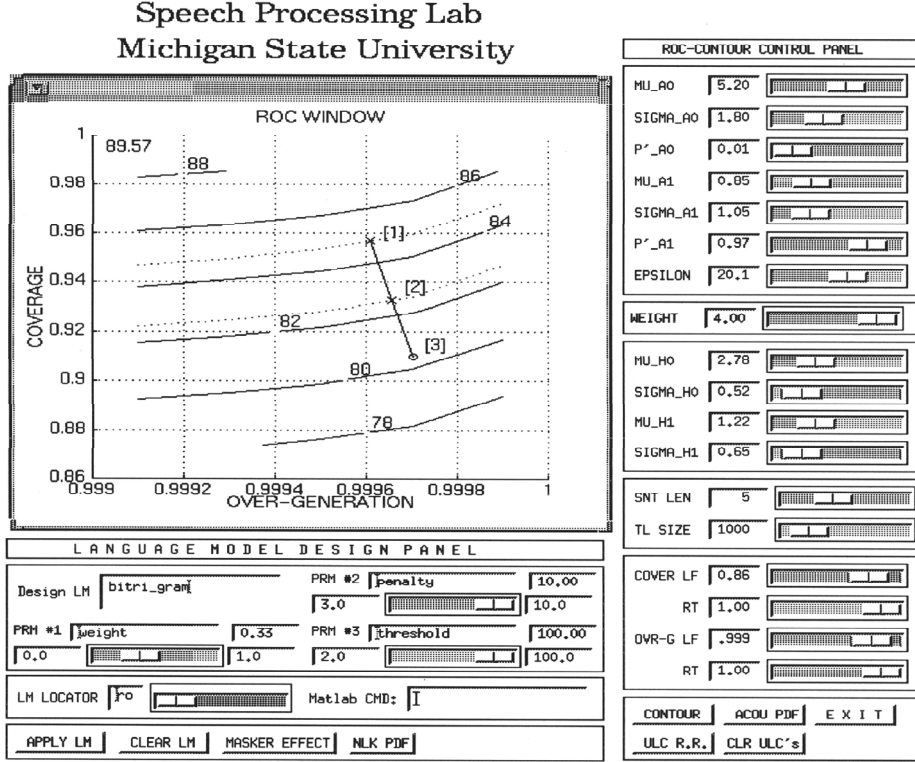


Fig. 8. Design Case II results. (a) The ‘relative motion’ (indicated by locations [1] and [2]) experienced by the LM when the bigram model is incorporated with the trigram by changing the mixture weight w_{bt} from unity to 0.66. Location [3] in on the panel is related to further experiments and should be ignored for present purposes.

achieved in the with the previous LM with worse overgeneration and coverage. In other words, the model moves relatively ‘southeast’ in the ROC-LM plane, counter to good design. Further, it is experimentally determined, by varying η_{bt} from one to zero, that a pure trigram model ($\eta_{bt} = 1$) provides the best recognition rate results for the present conditions. It is also determined experimentally that the penalty value $\varepsilon_{bt} = 14.0$ maximizes the recognition rate, so this value is chosen for further analysis. Location ‘[3]’ in Fig. 8(a) is related to further experiments and should be ignored for present purposes.

Next, in order to assess computational loads, a set of curves parameterized by LM size, $|L_M|$, is entered on the ROC plane (lightly dotted traces) in Fig. 8(b). These curves are easily derived using (21). An initial bi-tri-gram LM is employed with $\eta_{bt}=1.0$, $p_{bt}=14.0$, $t_{bt}=100$ (representing ∞), represented by point ‘[1]’ in Fig. 8(b). Now, suppose that there is an upper bound, say $2 \times 10^6 |L_T|$ for computational cost. The initial bi-tri-gram LM (labeled ‘[1]’) does not satisfy this requirement. Now as

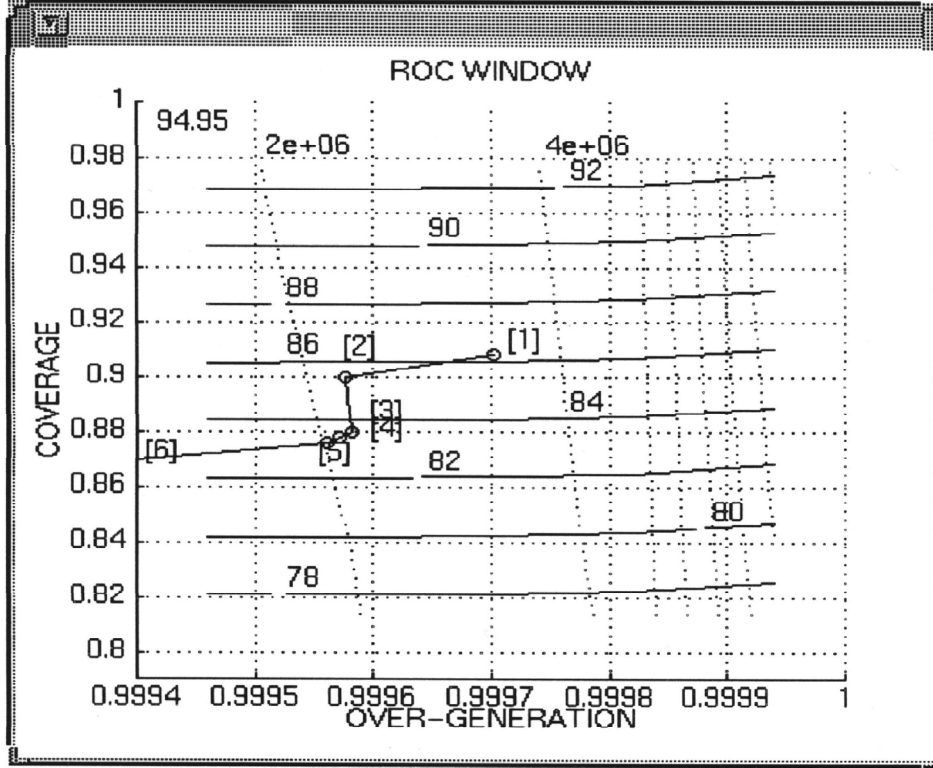


Fig. 8. [continued] (b) Trajectory of the LM as the pruning threshold is changed from 100 to 6.2 in five steps. The objective is to reduce the computational load by reducing the effective size of the LM. LM size is indicated by dotted contours on the panel.

the threshold t_{bt} is moved through the values 16.0 (result [2] in Fig. 8(b)), 13.2 ([3]), 10.4 ([4]), 7.6 ([5]) through 6.2 ([6]) to reduce the O_M , the bi-tri-gram LM moves toward the criterion line and crosses it at $t_{bt} = 7.6$. Thus the derived model satisfies the computational cost requirement, but at the cost of a decrement in recognition rate from 86.2 to 83.4 percent. Note in Fig. 8(b) that there is a discontinuity in the trajectory from the LM labeled '[2]' to LM '[3]'. Over this change, the direction of motion of the LM almost normal to the contour curves, implying that recognition rate is sacrificed with little improvement of computational cost.

4.2.4 Case III : Adaptive language modeling

In one formulation of the adaptive LM problem, the set of utterances to be recognized, say $X_{loc} \subset L_T$, are restricted, at any particular time, to a dynamically changing 'local' (partial) subset of the global task language. The training set, say $X_1 \subset X_{loc}$, available to represent X_{loc} at any time is generally insufficient to ro-

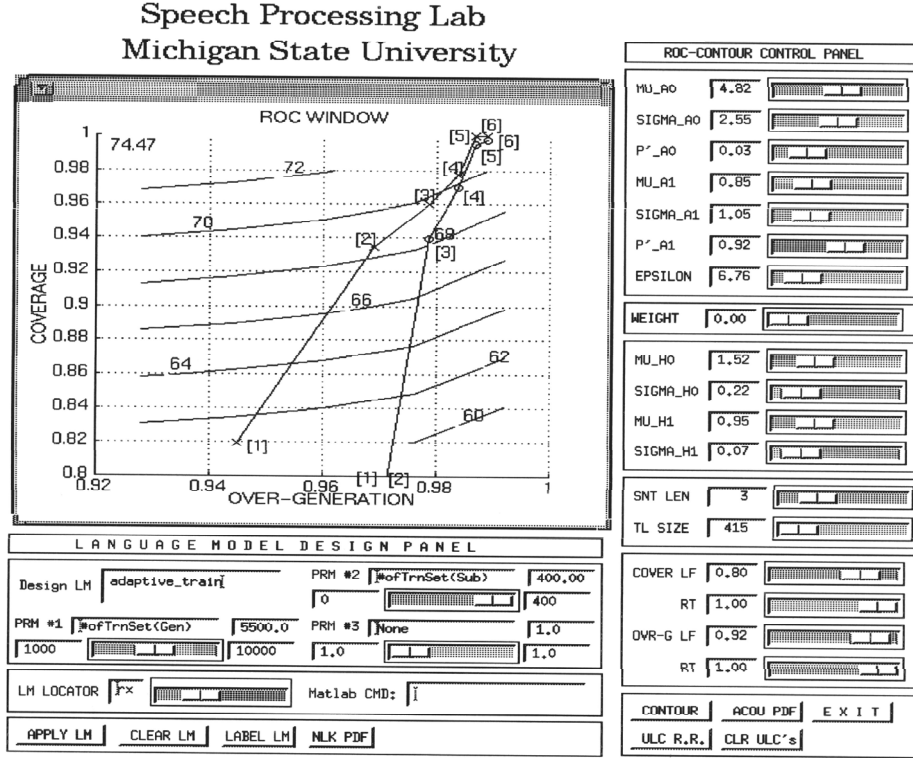


Fig. 9. Results of Design Case III. PRM #2, the number of X_{loc} sentences, is initially set to 0. Then, PRM #1, the number of X_{st} sentences, is varied in six steps: 1,000; 1,900; 2,800; 3,700; 4,600; 5,500. Then, PRM #2 is changed to 400. The PRM #1 steps are repeated. The second trajectory appears to the left side of the previous one. On the second trajectory, the recognition rate reaches its maximum when 3,700 X_{st} sentences are used.

bustly and dynamically train the LM. Therefore, a larger ‘static’ (or ‘global’) corpus $X_{st} \subseteq L_T$ is used to persistently supply more information. Let the training set drawn from X_{st} be denoted X_2 . The sizes of the training sets, $|X_1|$ and $|X_2|$ are parameters of the LM design that must be optimized with respect to the performance goal. Increasing $|X_2|$ will increase the C_M of a given LM while also increasing its O_M .

The set $\mathcal{P}_{a,TIMIT}^w$ [recall (40)] is employed as the acoustic model. The underlying TL in the present example is the *artificial digit-string* (ADS) language [see Appendix]. Pragmatic knowledge prescribes a subset of the ADS that can occur in conversations involving children. The ‘children’s’ subset comprises the ‘local’ language to be recognized, X_{loc} . An adaptive hybrid bi-tri-gram for X_{loc} is to be trained.

X_{st} is nominally the entire TL, and sentences generated for training from X_{st} , say, training set X_1 , are generated without consideration of pragmatic knowledge. Training sentences from X_{loc} , X_2 , adhere to the pragmatic rules for children’s speech in the ADS. The bigram portion of the LM is trained using both X_1 and X_2 , while and the trigram model is trained using X_2 only.

The sentence length (SNT LEN on the ROC-LM panel) is set to three, and, by enumeration of sentences according to pragmatic knowledge, the size of the ‘local TL,’ X_{loc} , is determined to be 415. The set-size parameters $|X_1|$ and $|X_2|$ are called ‘PRM #1’ and ‘PRM #2’ on the ROC-LM panel. Experimental results are displayed in Fig. 9. Each of the two design traces moving toward the ‘northeast’ in the ROC-LM plane represents increasing amounts of X_{st} (PRM #1 increasing) for a fixed X_{loc} size (PRM #2 = 0 in the rightmost trace, 400 in the other). (Details are found in the figure caption.) With either X_{loc} condition, adding training data from X_{st} improves C_M and decrements O_M , but with a nearly continual improvement in overall recognition performance. The entire display is computed in near-real time without the need for real recognition experiments.

With the information provided by X_2 (i.e., when PRM #2 is switched to 400), the ROC-LM trajectory is uniformly shifted toward the ‘northwest.’ Both O_M and C_M are improved in each case, thus improving the recognition rate as well as the computational cost.

5 Summary and conclusions

The design of LMs is a challenging problem requiring the careful consideration of many problem-dependent factors such as the amount of available training data, inherent TL constraints, computational complexity and storage requirements, and integration of the LM with other knowledge sources. The LM design problem is confounded by the need to test each design through laborious experimentation. Accordingly, changes of LM parameters or structure are not easily assessed, and LM ‘design’ is often relegated to the deployment of variations a few simple, widely-accepted methods (notably N -gram models) with few means for ‘optimizing’ or ‘informed tweaking.’

The ROC-LM design tool presented here is not a panacea to this general design problem. Rather, based on a formulation of LM the design problem in well-understood detection-theoretic terms, the tool represents a step toward the development of a formalized assessment methods. Experimental evidence suggests that the tool offers a powerful means for obtaining an initial assessment of a proposed LM, particularly in a *relative* sense compared with other LMs that differ in certain structural ways subject to parameterization. Nevertheless, the current developments involve several assumptions and models that are justified by limited experimental evidence, but which have the potential, if incorrect, to erode performance of the tool in general cases. These assumptions principally center on aggregate Gaussianity, but also (in the present experiments) on the use of ‘average’ sentence lengths. With varying degrees of difficulty, each of the simplifying assumptions can be re-

laxed in the ROC-LM developments given sufficient computing power and training data. The work presented here offers the framework for an interesting assessment tool with apparent potential for quick and meaningful off-line evaluation, and also a basis for further generalization and experimentation.

References

- Deller, J.R. Jr., Hansen, J.H.L. and Proakis J.G. (2000) *Discrete-Time Processing of Speech Signals* (2d ed.). New York: IEEE Press.
- Rabiner, L.R. and Juang, B.H. (1993) *Fundamentals of Speech Recognition*. Upper Saddle River, NJ: Prentice-Hall.
- Torres-Carrasquilla, P.A., Reynolds, D.A., and Deller, J.R. Jr. (2002) Language identification using Gaussian mixture model tokenization. *Proc. IEEE Intl. Conf. Acoustics, Speech, and Signal Proc.*, Orlando. (Published on CD-ROM)
- Iyer R., Ostendorf, M. and Gish, H. (1997a) Using out-of-domain data to improve in-domain language models. *IEEE Signal Proc. Letters*. 4:221–23.
- Kneser, R. and Peters, J. (1997) Semantic clustering for adaptive language modeling. *Proc. IEEE Intl. Conf. Acoustics, Speech, and Signal Proc.*, Munich. 2:779–82.
- Hwang, M., Rosenfeld, R., Theyer, E. *et al.* (1994) Improving speech recognition performance via phone-dependent VQ codebooks and adaptive language models in Sphinx-II. *Proc. IEEE Intl. Conf. Acoustics, Speech, and Signal Proc.*, Adelaide. 1:549–52 (1994).
- Lau, R., Rosenfeld, R., and Roucos, S. (1993) Trigger-based language models: A maximum entropy approach. *Proc. IEEE Intl. Conf. Acoustics, Speech, and Signal Proc.*, Minneapolis. 2:45–48.
- Meeter, M. and Rohlick, J.R. (1993) Statistical language modeling combining N -gram and context-free grammars. in *Proc. IEEE Intl. Conf. Acoustics, Speech, and Signal Proc.*, Minneapolis. 2:37–40.
- Della Pietra, S., Della Pietra, V., Mercer, R.L. and Roucos, S. (1992) Adaptive language modeling using minimum discriminant information. *Proc. IEEE Intl. Conf. Acoustics, Speech, and Signal Proc.*, San Francisco. 1:633–36.
- Cole, R., Hirschman, L., Atlas, L. *et al.* (1995) The challenge of spoken language systems: Research directions for the nineties. *IEEE Trans. Speech and Audio Proc.* 3:1–21.
- Iyer, R., Ostendorf, M. and Meeter, M. (1997b) Analyzing and predicting language model

improvements. *Proc. 1997 IEEE Workshop on Automatic Speech Recognition and Understanding*. pp. 254–61.

Poor, H.V. (1994) *An Introduction to Signal Detection and Estimation* (2d ed.). New York: Springer-Verlag.

Yang, Y.P. (1996) *Automated Evaluation of Language Models Based on Receiver-Operator-Characteristic Analysis* (Ph.D. dissertation). East Lansing: Michigan State U.

Doddington, G.R. and Schalk, T.B. (1981) Speech recognition: Turning theory into practice. *IEEE Spectrum*. pp. 26–32 (1981).

National Institute of Standards and Technology (1988) Getting started with the DARPA TIMIT CD-ROM: An acoustic-phonetic continuous speech database. Gaithersburg, MD.

Appendix: Artificial digit-string language (Deller *et al.* 2000, p. 621)

Consider an artificial language for which the symbols are 0, 1, 2, $\dots$, 9, +, -. The formal components of the language are as follows:

Lexical: A “word” consists of any number of digits concatenated by any sequence of + and - signs in any order. The first or last symbol in a word is never a sign. The total numerical value of a word must be nine. For example, “4+5-0” is a word in the language.

Syntactic: A “sentence” is any sequence of words such that any word ending in an odd digit must be followed by a word beginning in an even digit, and vice versa. For example, “3+6+4/7+3-1/2+0-5+8+4” is a proper sentence.

Semantic: To be “meaningful,” a sentence must contain words that become increasingly longer. For example, “9/2+7+1-1/4+5-5+5+0-0+2-1-1” is a meaningful sentence.

Pragmatic: When “speaking” to a child, each word in a sentence is usually no more than five digits long. There are no constraints for adult listeners.

Endnotes

1. The term *language model* is used in this paper to refer to the sources of knowledge in a speech recognizer that are ordinarily associated with the *grammatical* structure of natural language. In general, we include all structure (usually amenable, in principle, to objective and quantifiable modeling) that is ‘above’ the elemental acoustic representation, but ‘below’ the more difficult-to-model levels of the speech code having to do with meaning, for example. In short, a LM, as we use the term here, may model *lexical* and *syntactic* structures, and this knowledge may take a variety of forms depending principally upon the representations at the lowest levels of the hierarchy (e.g., triphone, monophone, or even whole-world, models may comprise the first super-acoustic representation in the LM).
2. Throughout, the notation $|X|$ indicates the number of elements in (or, ‘size of’) a set X . Similarly, if x is a sequence of objects (e.g., words, features), then $|x|$ indicates the number of objects in (or ‘length of’) the string.
3. Herein lies an opportunity for more detailed modeling. The present focus on *perfect* recognition as a figure of merit results in the use of only two categories of acoustic outcomes. However, a more detailed description of performance (e.g., number of word substitutions) could lead to a number of more refined categories of acoustic likelihood outcomes (e.g., acoustic likelihood values for hypotheses with d word substitutions). The description of likelihood p.d.f. construction in Section 2.4.1 below provides information that can be used in such generalizations of the acoustic outcome model.

extensive software support for algorithm development and hardware / software support for audio signal acquisition and preprocessing. In addition to appropriate facilities for signal processing, support facilities for graphics, word-processing, scanning, image editing, etc., are also in place. Laboratory desk space is available to all research personnel associated with the project.

Additionally, the Division of Engineering Computing Services (DECS) at MSU supports a rich environment for computing research and design. This center features a sophisticated network of more than 100 Unix Workstations and over 130 state-of-the-art PCs (Windows XP Pro), with 4 multi-user login servers and 11 printers. This network serves the College of Engineering in its six buildings, including the Research Complex for Engineering. DECS facilities are available to faculty offices, and to student labs, via high-speed ethernet and dial-in connections. The college supports numerous networked clusters of state-of-the-art workstations and personal computers, and an extensive catalog of software support for engineering research and teaching. Researchers at the University of Colorado will connect seamlessly to MSU facilities via guest accounts on the college network.

MSU is a member of the Linguistic Data Consortium which catalogs extensive databases for speech research. These resources will be available to the proposed project.

Bio Sketch. *Sent as pdf and LaTeX files.*