The Meta-Pi Network: Building Distributed Knowledge Representations for Robust Multisource Pattern Recognition

John B. Hampshire II, *Student Member, IEEE*, and Alex Waibel, *Member, IEEE*

***Abstract*— We present a multinetwork connectionist classifier that forms distributed low-level knowledge representations for robust pattern recognition, given random feature vectors generated by multiple statistically distinct sources. The architecture comprises a number of source-dependent modules (i.e., each module is trained to classify patterns from one particular source) that are linked by a combinational superstructure.**

The superstructure adapts to the source being processed, integrating source-dependent classifications based on its internal assessment of the source model or combination of source models most likely to classify the input signal correctly. To train this combinational network, we have developed a new form of multiplicative connection, which we call the "Meta-Pi" connection; its function is closely aligned with predecessors described in [3], [29], and [31].

We illustrate how the Meta-Pi paradigm implements an adaptive Bayesian maximum *a posteriori* (MAP) classifier. We demonstrate its performance in the context of multispeaker phoneme recognition. In this task, the Meta-Pi superstructure combines speaker-dependent time-delay neural network (TDNN) modules to perform multispeaker /b,d,g/ phoneme recognition with speaker-*de*pendent error rates (2%).

Finally, we apply the Meta-Pi architecture to a limited source-independent recognition task, illustrating its discrimination of a novel source. We demonstrate that it can adapt to the novel source (speaker), given five adaptation examples of each of the three phonemes; the resulting error rate of 7% is approximately three times that of a typical source-dependent classifier. Longer term adaptation yields discrimination that is comparable with a speaker-dependent classifier of the novel source. We conclude with an assessment of our experimental results and their implications for larger real-world multisource and source-independent pattern recognition systems.

***Index Terms*—Bayesian discriminant function, class-conditional density, connectionism, Meta-Pi network, mixture density, multisource, phoneme recognition, speech recognition, time-delay neural network (TDNN).**

Manuscript received October 20, 1989; revised February 14, 1992. This work was funded by grants from Bell Communications Research, ATR Interpreting Telephony Research Laboratories, and the National Science Foundation (NSF grant EET-8716324). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of Bellcore, ATR, NSF, or the U.S. Government. Recommended for acceptance by Associate Editor C. Dyer.

J. B. Hampshire is with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213-3890

A. Waibel is with the School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213-3890.

I. Introduction

MANY PATTERN recognition tasks involve complex and highly conditional mappings[1] of stochastic inputs to finite-state classification outputs. The patterns to be recognized are often generated by multiple sources, each with unique statistical properties. We refer to these sources as "heterogeneous sources" [21]. When a classifier is robust for all possible sources (i.e., over the entire *ensemble* of sources), it is said to be "source-independent"; when it is robust for some subset of the ensemble, it is said to be a "multisource" classifier; when it is robust for only one source, it is said to be "source-dependent."

We present the Meta-Pi paradigm as a modular connectionist classifier for multisource pattern recognition and discuss how it might be used for source-independent pattern recognition.

A. Background and Summary

The notion of using modular connectionist systems to realize complex nonlinear transfer (or mapping) functions was discussed at least as far back as the mid 1980's by Barto and Hinton. Jacobs developed a taxonomy for a class of modular hierarchical connectionist models [15] based on Pollack's cascaded backpropagation architecture [29]. In so doing, he first articulated many of the key issues of modular connectionist design. Kämmerer and Küpper were among the first to build a hierarchical connectionist pattern classifier for the task of isolated word recognition [18]—one of a number of efforts at building modular connectionist classifiers for speech recognition (e.g., [24], [41], [44], [27]).

The Meta-Pi architecture [8] is a multisource connectionist pattern classifier that comprises a number of source-dependent subnetworks (or "modules") that are integrated by a combinational superstructure. We refer to the latter as the Meta-Pi combinational superstructure, owing to the multiplicative function that its output units perform. This function serves to combine the outputs of the modules (independently trained to classify inputs from specific sources) in order to form a global classification that is independent of the conditional statistical nature of the input source. For our particular application, the Meta-Pi superstructure is a time-

[1] i.e., mapping functions that vary widely according to statistical "environmental" conditions

delay neural network (TDNN), but the paradigm applies to any multilayer perceptron (MLP) classifier.[2]

We show that the output units of the Meta-Pi combinational superstructure are—as in the cascaded backpropagation model—continuous-valued weights. In the cascaded backpropagation model, the outputs of the superstructure (which Pollack calls the "supervisory network") constitute the weights of a module (which Pollack calls a "subordinate network"). Thus, the superstructure alters the mapping function of the module(s). In contrast, the outputs of the Meta-Pi model's combinational superstructure represent the degree to which a corresponding module contributes to the *global* classification decision; the superstructure does not alter the mapping function of the module(s).

We provide both a probabilistic rationale and a probabilistic framework for the Meta-Pi paradigm. In the latter, we show how the Meta-Pi paradigm implements a maximum *a posteriori* (MAP) Bayesian classifier that learns to compute optimal source mixtures in order to achieve robust multisource classification. We then describe how the Meta-Pi network's error signal is backpropagated through the combinational superstructure in an indirect way, altering the superstructure's output in order to optimize global discrimination.

We demonstrate the Meta-Pi network's discrimination performance on the multispeaker /b,d,g/ phoneme recognition task. At a very coarse level, the vocal tract characteristics of a particular speaker are unique to that individual; at another level, a number of conditions (e.g., the health of the individual, his/her emotional state, etc.) alter the acoustic-phonetic signature of that individual's speech. As a result, each speaker represents a statistically unique source, and multispeaker /b,d,g/ phoneme recognition provides a good heterogeneous-source pattern recognition task with which to test the Meta-Pi concept. We show that the Meta-Pi network's recognition performance on the multisource (six speaker /b,d,g/) task is comparable to the average performance of the source-dependent modules.

Following the multisource experiments, we demonstrate the Meta-Pi network's discrimination performance on a limited source-independent task. Specifically, we remove one source (speaker) from the Meta-Pi training procedure, reducing the number of training sources from six to five. We train the classifier, and then test its performance on the sixth (novel) source. We find that Meta-Pi network's error rate[3] on the novel speaker is approximately one order of magnitude higher than its error rate on known sources. However, we find that the Meta-Pi combinational superstructure can adapt to the novel source with a small number of examples (five of each phoneme), yielding an error rate that approaches the error rate for the known sources.

We conclude with a discussion of our experimental results and their implications for larger real-world multisource and source-independent pattern recognition systems.

[2]We use the term "multilayer perceptron" to describe a backpropagation network using any continuous sigmoidal nonlinearity.

[3]Throughout this paper, we use the term "error rate" in reference to the classifier's *estimated* probability of error.

B. Outline

We address the issues described above in the following sequence:

1. Section II: Classifier design issues—probabilistic and connectionist
2. Section III: Multisource pattern recognition using a modular hierarchical connectionist structure
3. Section IV: The Meta-Pi network
4. Section V: Multispeaker phoneme recognition using the Meta-Pi architecture
5. Section VI: A limited speaker-independent phoneme recognition experiment using the Meta-Pi architecture
6. Section VII: Discussion—properties of the Meta-Pi network and related architectures, significance of results, and remaining questions
7. Section VIII: Conclusion
8. Appendix: A formal statement of the acuity/generality tradeoff introduced in Section II-A.

Authors' Note: Since we first introduced the Meta-Pi paradigm in [8], a number of other researchers (who were working on very similar ideas independently) have published their work [44], [16], [26], [25], [17], [27]. We encourage the interested reader to review each of them in detail.

C. Experimental Data

The experimental data for this research are detailed in [40]. Japanese speech from six professional announcers (two female and four male) was sampled at 12 kHz, parsed for the /b, d, g/ phonemes, and hamming windowed; from this windowed data, 256-point DFT's were computed at 5-ms intervals. The DFT's were used to generate 16 Melscale coefficient spectra at 10-ms intervals. These spectra were normalized to produce suitable input levels for the TDNN's. Training tokens for individual speakers were shuffled randomly and interleaved to produce successive /b, d, g/ tokens (approximately 250 training and 250 testing tokens per phoneme, per speaker). Training tokens for the Meta-Pi combinational superstructure comprised a complete mixture of the tokens used to train the speaker-dependent (i.e., source-dependent) modules. The superstructures were also provided with the output states of each of the fully trained source-dependent modules for *all* training tokens (please see Section IV-B for a full discussion of the Meta-Pi training procedure).

II. Connectionist Classifier Design Issues

The issues of connectionist classifier design that we address are fundamentally pragmatic in nature. Although abstract issues such as theoretical questions of convergence are vitally important to the more general question of whether MLP classifiers are *provably* workable, we do not address them. Instead, we make assumptions regarding convergence and learnability (detailed in [7]) and focus on probabilistic and architectural factors that lead to the Meta-Pi paradigm.

The original motivation for the Meta-Pi model was twofold [8]:

1. There was a desire to build a connectionist system that

could recognize the phonemic speech of multiple speakers with error rates comparable to a speaker-dependent system designed to perform the same task on the speech of a single individual.
2. There was a desire to make the system highly modular by integrating speaker-dependent modules to perform the overall task of multispeaker recognition.

Both goals have probabilistic implications, and the second goal has architectural/computational implications. We discuss these implications in the following two sections.

A. Probablistic Design Issues: The Acuity/Generality Tradeoff

One perspective of modular connectionist systems is that by dividing the input-to-output functional mapping into submappings over the input space, they decompose a task into subtasks [15]. In contrast, we view the task of classifying patterns generated by multiple heterogeneous sources in a probabilistic context. Instead of a *task* that is decomposable into subtasks, we envisage a *probabilistic model* that can be estimated by adaptive combinations (or "mixtures" [10]) of other independent models. Nowlan takes a similar probabilistic view in [25] and [27]. This concept can be embodied in a modular connectionist system that utilizes a number of source-dependent statistical models to implement a multisource classifier—which is appropriate for speech recognition, as an example, wherein one can view each speaker as a source with unique statistical properties.

Early speaker-independent speech recognition systems used fixed probabilistic models of the atomic (acoustic phonetic) units of speech derived from large populations of speakers. Once computed, these fixed "mixture densities" [33] were then used in a Bayesian classification scheme to determine the atomic unit of speech associated with the speech signal at each point of analysis in time (e.g., see [32]). Such a scheme for building source-independent models of speech seems quite sensible, but it invariably leads to a discrimination tradeoff: although multisource and source-independent systems based on fixed mixture densities are significantly better than source-dependent systems at recognizing the speech of a novel speaker, they are significantly worse than source-dependent systems at recognizing the speech of any one particular individual.

In fact, there is a simple explanation for this paradox. It is grounded in the rudiments of Bayesian classification theory and provides a probabilistic perspective of multisource (and source-independent) pattern recognition:

> *There is a tradeoff between the* acuity *of a single classifier (i.e., its ability to discriminate among classes with accuracy) and its* generality *(i.e., its applicability to a large ensemble of heterogeneous sources). In short, one can do* moderately *well classifying inputs from* all *sources with one statistical model, or* very *well classifying inputs from* each *source with its own model, but one can't do very well classifying all sources with a single model.*

It is relatively straightforward to prove this acuity/generality tradeoff (see the Appendix). Such a proof leads to the following assertion:

> *If the input patterns to be classified are drawn from heterogeneous sources, the only way to achieve (optimal) Bayesian discrimination for all sources is to have an accurate statistical model of each source.*

When the upper bound on the number of possible sources is small (multisource task), it is practical to model each source explicitly. However, when this upper bound is large (large multisource or source-independent task), one must resort to an alternative strategy. One strategy is to have a single statistical model that is optimal for a fixed collection of sources and adapt that model to each particular source using some form of real-time learning scheme (see, for example, [39], chapter 7 of [20], [34], and [14]).

An alternative scheme is to have a statistically representative collection of source-dependent models and some mechanism for combining (i.e., adaptively mixing) these models in order to estimate the *a posteriori* class distributions of the particular source being recognized. The Meta-Pi paradigm employs this alternative scheme to achieve robust multisource pattern classification. There are implementational issues that stem from this probabilistic perspective; we raise them under the general auspice of connectionist classifier design.

B. Connectionist Design Issues: Modularity and Scaling

Waibel *et al.* [41] illustrate the effectiveness and computational efficiency of modular connectionist systems for recognizing all the consonants of an individual speaker. A single monolithic network trained to accomplish this all-consonant task from a "tabula rasa" state would be large (on the order of 10^5 connections) and would require a long training time, owing to the requisite size of the training data set. Indeed, monolithic connectionist approaches to pattern recognition, in general, seem tractable only for relatively rudimentary tasks. The constraints of network size and training set size (ergo, training time) are joined by an additional constraint—the acuity/generality tradeoff—when the patterns are generated by multiple heterogeneous sources.

Given this tradeoff, a connectionist approach to pattern recognition might employ some form of modularity whereby source-dependent recognition processes are integrated into a global (multisource) framework. In order to implement such a structure, we consider a number of properties of the module integration scheme.

1. **The size of the modular structure:** In particular, what is the size of the global structure, what is the size of a typical module, and what is the size of the largest distinct network (whether it is a module or part of the integrated structure)? We seek to minimize the size of all the components of the global structure, and we wish to keep the relative magnitude of all components roughly equivalent so that no single component of the modular structure constitutes a computational bottleneck.
2. **The size of the training set required:** What size training set is typically required for a module? What size training set is required for the integrated structure? We seek to minimize the training set sizes while maintaining robust powers of generalization in the overall modular structure.

3. **The total training time required for the modular structure:** What is the typical training time for a module? What is the training time for the integrated structure? Are the training of the modules and the integrated structure independent or dependent? If they are dependent, what is the degree to which they are dependent? We seek to minimize the training time for all components of the modular structure. We also seek to minimize the interdependence among the various training phases in order to maximize the modularity of the overall structure.
4. **Ease with which the modular structure is modified:** If the integrated structure combines a number of modules trained on specific sources, then one might want to add a new source-dependent module to the existing global structure at a later time. Does the architecture lend itself to such an addition with a minimum of retraining, or must the entire structure be dismantled and (in effect) retrained to incorporate the new module? We seek a modular structure that is easily modified. Ideally, we would like to train a source-dependent module independent of the overall modular structure and subsequently integrate this new module into the existing structure with a relatively simple structural alteration and retraining.
5. **Extensibility to novel sources:** Is the modular structure intrinsically extensible to novel sources? This issue can be viewed as one of generalization on a more global scale. A classifier is said to generalize well if its classification performance on a disjoint test set is comparable with its performance on the data set with which it was trained. Typically, our view of generalization assumes that these disjoint training and test sets are drawn from the same source or ensemble of sources. In this context, the source-dependent modules of our modular structure can generalize very well on test data that are disjoint from the training data but drawn from the same specific source without necessarily forming a global network that classifies patterns from novel sources accurately. We define a novel source as one with statistical properties that differ substantially from those of the sources modeled in our structure. Thus, good generalization at the module level is not a sufficient condition for extensibility to novel sources because an accurate probabilistic model of one source does not guarantee an accurate model of another source. Given this definition of a novel source, we characterize a modular network as being intrinsically extensible to novel sources if it a) can correctly classify a novel source without modification or b) can rapidly *adapt* to and correctly classify the novel source using a dynamic combination of known source-dependent models embodied in its existing set of modules. Note that this adaptive procedure occurs rapidly and does *not* involve training a new source-dependent module for the novel source. We seek a modular structure that is intrinsically extensible to novel sources.
6. **Error propagation and fault tolerance:** How does the integrated structure handle errors made at the module level? Do these lower level errors propagate to the final output of the modular structure, or does the architecture have some means of correcting—or at least suppressing—lower level errors. In addition, can the structure withstand the failure of a small percentage of its modules without necessarily yielding a statistically significant increase in its error rate? In short, is the modular structure fault tolerant?

One concept suggested by both the probabilistic and the connectionist issues that we have raised is a structure comprising a number of source-dependent modules linked by a combinational superstructure. We describe such a structure in the next section. By evaluating its properties and considering its probabilistic interpretation, we then describe a change in its training procedure that leads to the Meta-Pi paradigm.

III. The Source Identification Network: A Modular Hierarchical Connectionist Model for Recognizing Patterns from Multiple Sources

The source identification (SID) network is architecturally very similar to the integrated neural network (INN) [24]. The INN is used to integrate modules trained to recognize disjoint subsets of phonemes. The SID network is used to integrate modules trained to recognize different speakers; the specific recognition task is the same for all modules (in our application, recognizing the /b,d,g/ phonemes). We first describe the SID architecture and then offer a probabilistic interpretation of it.

A. Connectionist Architecture

The SID modular architecture shown in Fig. 1 classifies input signals from K different sources by using a source identification combinational superstructure to select the appropriate source-dependent module for classifying the input signal. In order to keep the figure compact and visually clean, the output units of each module and the global outputs have been aligned vertically. The C connections linking the C outputs $\{\rho_{k,1}, \rho_{k,2}, \ldots, \rho_{k,C}\}$ of each of the K modules to the C global outputs $\{O_1, O_2, \ldots, O_C\}$ via their respective SID combinational superstructure unit S_k are shown as single arrows. Each module in the overall structure is trained on data from a single source. The SID combinational superstructure is trained on the same data used to train the source-dependent modules; however, the data for each source are combined into a global training data set with which the SID superstructure is trained to identify the source generating the input signal. The inset of Fig. 1 illustrates the training procedures for the SID architecture's modules and combinational superstructure. The training of each element (e.g., the modules and combinational superstructure) can be viewed as a search on the parameter space $\boldsymbol{\theta}$ of that element; a measure of the element's discrimination over the set of training samples is evaluated (denoted by the comparator symbols in the inset), and the element's parameters are adjusted in order to optimize this measure. After training, recognition is performed by using the combinational superstructure to form a global (multisource) classification from the constituent modules. Note that the training of the superstructure is independent of the module

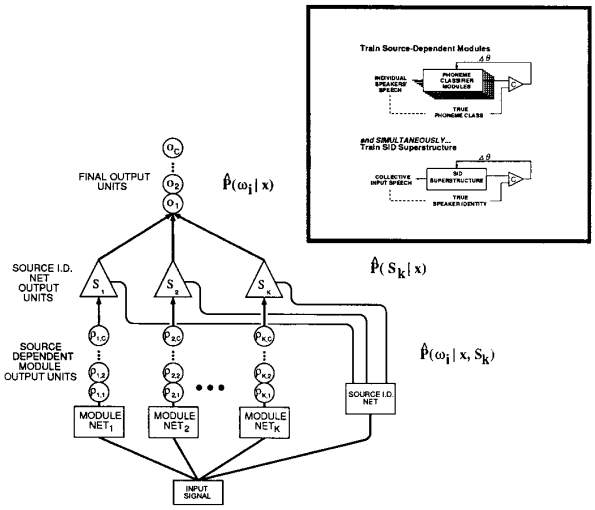


Fig. 1. Source identification (SID) modular network architecture and its training procedure.

training; therefore, all components of the SID architecture can be trained in parallel.

Once all of the source-dependent modules and the SID superstructure have been trained, the global classification decision is obtained (Fig. 1) by using the superstructure's output state to combine the outputs of the various modules. We employ one of two combinational schemes: a "winner-take-all" scheme in which

$$O_i = \rho_{k,i} \tag{1}$$

where k is identified by $\sup_k S_k$ (*sup* denotes the supremum operator) and a proportional scheme in which

$$O_i = \sum_{k=1}^{K} S_k \cdot \rho_{k,i} \tag{2}$$

Fig. 2 illustrates the SID architecture applied to the task of multispeaker recognition of the voiced-stop phonemes /b,d,g/. The basic building block of this modular structure is the TDNN. Three TDNN's for each of the six speakers share the same input layer along with the SID combinational superstructure (which is itself a TDNN with a first hidden layer (not shown) containing 12 units versus eight for the "standard" TDNN). For each speaker, one TDNN is trained with the mean-squared-error (MSE) [3] objective function, one is trained with the cross-entropy (CE) [12] objective function, and one is trained with the classification figure-of-merit (CFM) [9] objective function. The outputs of the three networks trained on a given individual's speech are combined using the three-way summation form of conflict arbitration to produce the final speaker-dependent, "three-way arbitrated," outputs[4] of each module shown in Fig. 2. The box in the upper right

[4]Three-way summation arbitration forms a final classification by taking the average output activations of the three TDNN's trained with the three different objective functions. Each of these TDNN's is viewed as an independent estimator of the Bayesian discriminant function (see Section III-B) for the /b,d,g/ task. Hampshire and Waibel [9] describes this arbitration procedure in detail and shows that it reduces speaker-dependent /b,d,g/ classification errors by 30%.

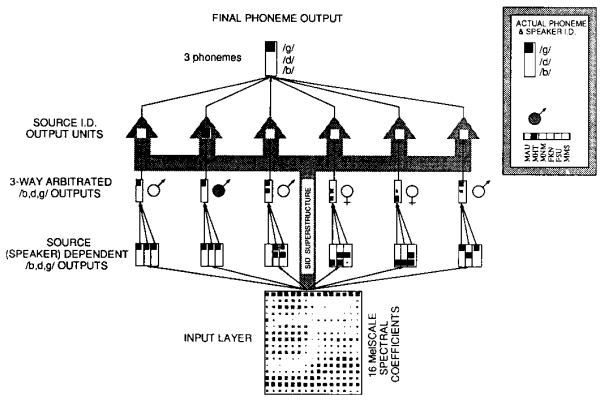


Fig. 2. SID architecture performing multispeaker phoneme recognition (/b,d,g/ task).

TABLE I
A COMPARISON OF SIX-SPEAKER /b, d, g/ ERROR RATES FOR ACTUAL AND ARTIFICIAL COMBINATIONAL SCHEMES USED BY THE SID NETWORK.

	Actual			Artificial	
	Winner-take-all	Proportional	Normalized Gross Sum	Gender known	SID known
3-Way Arbitrated SID Superstructure	1.9 (+0.5/-0.4)%	1.7 (+/- 0.4)%	21.2 (+/- 1.3)%	5.5 (+/- 0.7)%	1.3 (+/- 0.4)%
Single CFM-trained SID superstructure	2.3 (+/- 0.5)%	2.1 (+/- 0.5)%			

corner of Fig. 2 shows the actual phoneme spoken and the true identity of the speaker. The true identity of the speaker is also marked at the module level of the structure by a darkened gender symbol. Thus, speaker MHT is uttering the phoneme /g/ in this sample of speech; the SID superstructure correctly identifies the speaker and passes the MHT-specific phoneme classification to the global output of the network, yielding a correct recognition result.

Table I compares the error rates for the combinational strategies of (1) and (2) with the results one would obtain if one were to use the crude strategy of taking the normalized sum of all of the speaker-dependent module outputs as the global network output. Results are also shown for this same strategy when the gender of the speaker is explicitly given and used to limit the scope of the normalized sum (e.g., if the speaker is known to be female, only the outputs of the female modules are summed to form the global classification output). Finally, we show the error rate for the ideal case in which the speaker's identity is explicitly known (corresponding to perfect discrimination in the SID superstructure using the winner-take-all combinational scheme). The error rates for these three schemes (the latter two of which are artificial) are provided for comparison with the "winner-take-all" and proportional schemes.

Table I shows that the ideal case in which speaker identification is perfect yields a multispeaker error rate of 1.3 (+/- 0.4)% —the average of the individual speaker-dependent recognition rates.[5] The "winner-take-all" strategy yields an

[5]Numerical error rates are given with +/- deviations that represent the upper and lower bounds of a 95% confidence interval. This interval is computed under the assumption that the error rate is binomially distributed [11]. We judge two error rates to be statistically equivalent if their confidence bounds overlap. Please see Section VII for more details on the statistical significance of our results.

error rate of 1.9 (+0.5/-0.4)% , and the proportional strategy yields an error rate of 1.7 (+/- 0.4)% when a three-way arbitrated combinational superstructure is used. Both of these error rates are statistically equivalent to the ideal speaker-dependent rate of 1.3 (+/- 0.4)% . When we use a single TDNN trained with the CFM objective function (rather than a three-way arbitrated TDNN) to implement the combinational superstructure, the "winner-take-all" strategy yields an error rate of 2.3 (+/- 0.5)% , and the proportional strategy yields an error rate of 2.1 (+/- 0.5)% . Thus, there is no statistically significant difference between the two superstructure designs. We choose the single CFM-trained TDNN superstructure since it is one third the size and requires one third the training of the three-way arbitrated superstructure. Note that both the winner-take-all and proportional schemes outperform the normalized sum (21.2 (+/- 1.3)%) by a substantial margin and the gender-specific normalized sum (5.5 (+/- 0.7)%) by a statistically significant margin.

B. Probablistic Rationale

To facilitate the SID network's probabilistic description, we view it in the context of the multispeaker phoneme recognition task wherein each speaker is a unique source. Let us consider the structure in Fig. 1 as follows: The $\mathcal{C}$ global network outputs $O_1, \ldots, O_{\mathcal{C}}$ correspond to $\mathcal{C}$ possible phonemes to be recognized from the input speech signal (which we will call $\boldsymbol{x}$).[6] The K modules in the structure represent K source-dependent (i.e., speaker-dependent) networks trained on the same phoneme recognition task required of the global structure. The source-dependent outputs $\rho_{k,1}, \ldots, \rho_{k,\mathcal{C}}$ correspond to their global counterparts $O_1, \ldots, O_{\mathcal{C}}$. Let us assume for a moment that each global output O_i represents an estimate of the continuous-valued *a posteriori* probability of the phoneme ω_i, given the input signal $\boldsymbol{x}$:[7]

$$O_i = \hat{\mathrm{P}}(\omega_i|\boldsymbol{x}) \tag{3}$$

(where $\hat{\mathrm{P}}(\cdot)$ denotes an estimated probability). In reality, the *a posteriori* $\mathrm{P}(\omega_i|\boldsymbol{x})$ given in (3) is conditional in nature—principally affected by the dialectal characteristics, vocal tract properties, physical and emotional states, etc. of the speaker actually uttering the input signal $\boldsymbol{x}$. We bind all of these probabilistic conditions into the state variable $\boldsymbol{S}$ (denoting source) so that (3) is more precisely expressed as

$$O_i = \hat{\mathrm{P}}(\omega_i|\boldsymbol{x}, \boldsymbol{S}). \tag{4}$$

Clearly, the SID network yields optimal discrimination if O_i in (4) is indeed an accurate estimate of the Bayesian discriminant function $\mathrm{P}(\omega_i|\boldsymbol{x}, \boldsymbol{S})$ (see pp. 16–23 [5]) for all $\mathcal{C}$ sources. Having pointed out the conditional nature of the Bayesian discriminant function, we revert to the notation of (3).

[6] Strictly speaking, $\boldsymbol{x}$ is a random vector *sequence* (see Section I-C).

[7] References [5], [1], [45], [22], [6], [35], [2], [42], and [37] prove this assumption for classifiers trained with the mean-squared error objective function. Reference [7] extends the proof to two broad families of objective functions; proofs therein are given for all objective functions used in each "conflict arbitrated" module and/or combinational superstructure described in Sections III-A and IV-B.

If we consider $\rho_{k,i}$ (the ith output of the kth source-dependent module) in the same probabilistic light in which we view O_i, then

$$\rho_{k,i} = \hat{\mathrm{P}}(\omega_i|\boldsymbol{x}, \boldsymbol{S}_k) \tag{5}$$

which is the probability of phoneme ω_i, given that the input signal is $\boldsymbol{x}$, and the k th source's characteristics are accurately modeled by the k th source-dependent module.

It is possible to form an arbitrary probability density function (PDF) by the principle of linear superposition (i.e., one can form an arbitrary PDF by taking a normalized composite of a sufficiently large number of independent PDF's).[8] Thus, we can express the *a posteriori* probability $\mathrm{P}(\omega_i|\boldsymbol{x})$ as linear combination of the source-dependent *a posterioris*:

$$\begin{aligned}\mathrm{P}(\omega_i|\boldsymbol{x}) &\approx \sum_K \mathrm{P}(\omega_i|\boldsymbol{x}, \boldsymbol{S}_k) \cdot \mathrm{P}(\boldsymbol{S}_k|\boldsymbol{x}) \\ &= \sum_K \frac{\mathrm{P}(\omega_i, \boldsymbol{x}, \boldsymbol{S}_k) \cdot \mathrm{P}(\boldsymbol{x}, \boldsymbol{S}_k)}{\mathrm{P}(\boldsymbol{x}, \boldsymbol{S}_k) \cdot \mathrm{P}(\boldsymbol{x})} \\ &= \frac{\hat{\mathrm{P}}(\omega_i, \boldsymbol{x})}{\mathrm{P}(\boldsymbol{x})} \\ &= \hat{\mathrm{P}}(\omega_i|\boldsymbol{x}).\end{aligned} \tag{6}$$

Note, (6) assumes that K modules are sufficient to model any $\mathrm{P}(\omega_i|\boldsymbol{x})$. This assumption is valid for a multisource recognition task, wherein the number of sources is relatively small and the test and training sources are the same. Section VII-C considers the necessary conditions for the assumption to hold in a source-independent recognition task.

Using the proportional combinational scheme of (2) leads to the following interpretation of the SID network's outputs (see [7])[9]:

$$S_k = \hat{\mathrm{P}}(\boldsymbol{S}_k|\boldsymbol{x}). \tag{7}$$

Thus, in the Bayesian context, the SID network estimates the multisource Bayesian discriminant function by learning to compute estimates of the likelihoods $\mathrm{P}(\boldsymbol{S}_k|\boldsymbol{x})\ \forall k$ so that it can form a robust approximation of the *a posteriori* class distributions $\mathrm{P}(\omega_i|\boldsymbol{x})\ \forall i$ in (3) via the relationship of (6). The reader should note that Nowlan presents a similar probabilistic rationale for his connectionist approach to associative mixture models in [25] and [27].

The winner-take-all combinational strategy of (1) allows only one nonzero term in the classifier's estimate of the mixture in (6), whereas the proportional scheme of (2) allows all K terms to be nonzero. It is interesting to note that the gross summation scheme in Table I is equivalent to a proportional scheme in which all the source likelihood estimates $\hat{\mathrm{P}}(\boldsymbol{S}_k|\boldsymbol{x})$ in (7) are equal. This uninformed mixture model performs adequately (if indeed a 21.2 (+/- 1.3)% error rate can be considered adequate) over the six sources but at a cost in terms of its discrimination on any particular source—an exaggerated

[8] Note that [38] and [23] employ the principle of linear superposition of independent Gaussian PDF's in a connectionist framework to estimate the (arbitrary) PDF of a random vector. Nowlan uses a probabilistic mixture model for a connectionist approach to regression in [26].

[9] Note that S_k denotes the kth output of the SID combinational superstructure, whereas $\boldsymbol{S}_k$ denotes the kth source.

yet clear example of the acuity/generality tradeoff described in Section II-A.

IV. The Meta-Pi Paradigm

It would be helpful if the SID combinational superstructure could somehow link its module selection process with a performance assessment for the modules themselves, that is, if the combinational superstructure could somehow know when a source-dependent module provided an ambiguous or erroneous output, it could avoid using that module to recognize the input pattern in question, thereby avoiding (or at least suppressing) errors made at the module level. In training the SID superstructure to recognize a source's identity, we give the network no information regarding the *global* objective of accurate pattern recognition. As a result, the SID superstructure's training is completely independent of the global pattern recognition objective; there is no linkage between module discrimination and module selection. A review of the SID architecture's global discrimination on the multispeaker phoneme recognition task shows that it is very common for more than one source-dependent module to classify any given input correctly. Given the probabilistic relationships of Section III-B, this is not surprising. The phenomenon leads us to envisage an SID-like combinational superstructure that is no longer trained explicitly to perform source identification; instead, it is trained to use *any* combination of source-dependent modules that will classify a given phone correctly. In fact, it is possible to use the SID network's global phoneme recognition performance to *learn* this optimal combinational function. This change leads to the Meta-Pi paradigm. Architecturally, the Meta-Pi network is identical to the SID network (Fig. 1), but its training procedure is different. The difference has two elements: a difference of probabilistic interpretation and a resulting procedural difference, both of which pertain to the combinational superstructure.

A. Probablistic Rationale

In the SID network, each of the combinational superstructure's outputs S_k is an estimate of the probability that the input pattern $\boldsymbol{x}$ has been *generated* by the kth source $\boldsymbol{S}_k$ (see (7)). In the Meta-Pi network, each of the superstructure's outputs M_{π_k} is viewed as the probability that the kth source-dependent model $\boldsymbol{M}_k$ is *relevant*[10] to classifying the input pattern correctly. The distinction here is subtle but important.

Again, let us assume that each global output O_i represents an estimate of the continuous-valued probability of the phoneme ω_i, given the input signal $\boldsymbol{x}$ (again, see [7]):

$$O_i = \hat{\mathrm{P}}(\omega_i | \boldsymbol{x}) \tag{8}$$

However, we now consider $\rho_{k,i}$ (the ith output of the kth source-dependent module) as the probability of phoneme ω_i, given that the input signal is $\boldsymbol{x}$, and the kth source-dependent module is being used to classify $\boldsymbol{x}$:

$$\rho_{k,i} = \hat{\mathrm{P}}(\omega_i | \boldsymbol{x}, \boldsymbol{M}_k) \tag{9}$$

[10]The term "relevance"—as it pertains to modular connectionist systems—was coined by Hinton.

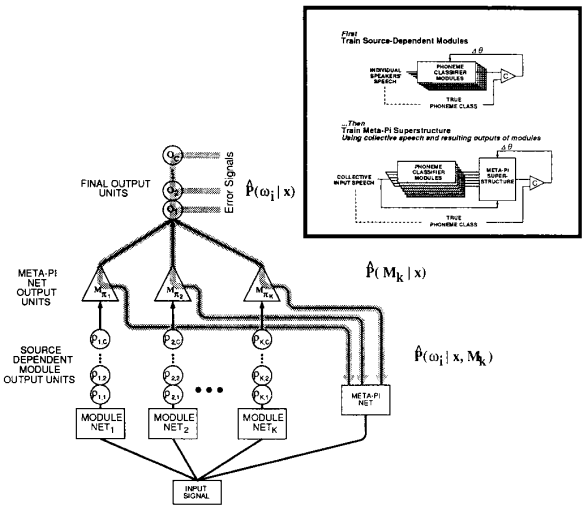


Fig. 3. Meta-Pi network and its training procedure.

Thus, we can express the *a posteriori* probability $\mathrm{P}(\omega_i | \boldsymbol{x})$ as a linear combination of the source-dependent *a posterioris*, just as we did in (6):

$$\mathrm{P}(\omega_i | \boldsymbol{x}) \approx \frac{1}{\mu} \sum_K \mathrm{P}(\omega_i | \boldsymbol{x}, \boldsymbol{M}_k) \cdot \mathrm{P}(\boldsymbol{M}_k | \boldsymbol{x}) \tag{10}$$

Note that the kth output of the Meta-Pi combinational superstructure M_{π_k} (see Fig. 3) is a measure of the relevance of the kth source-dependent module, given $\boldsymbol{x}$. In the probabilistic context, this kth superstructure output is an estimate of the probability that the kth source-dependent module will correctly classify the input:

$$M_{\pi_k} = \hat{\mathrm{P}}(\boldsymbol{M}_k | \boldsymbol{x}) \tag{11}$$

Because it is possible (albeit unlikely) for all K source-dependent modules to be relevant for a given input, we normalize the outputs of the Meta-Pi combinational superstructure by $\frac{1}{\mu}$, where

$$\mu \triangleq \sum_K \mathrm{P}(\boldsymbol{M}_k | \boldsymbol{x}) \tag{12}$$
$$\mu > 0$$
$$0 \leq \mathrm{P}(\boldsymbol{M}_k | \boldsymbol{x}) \leq 1 \quad \forall k$$

in (10), and

$$\sum_K M_{\pi_k} = \hat{\mu}. \tag{13}$$

This ensures that the normalized aggregate relevance of the sources $\frac{1}{\mu} \sum_K \mathrm{P}(\boldsymbol{M}_k | \boldsymbol{x})$ is unity, as required by the Bayesian formalism of (10).[11] Note—as with (6)—that (10) assumes that K modules are sufficient to model any $\mathrm{P}(\omega_i | \boldsymbol{x})$. Again, Section VII-C considers the necessary conditions for this assumption to hold in a source-independent recognition task.

[11]The scaling factor $\frac{1}{\mu}$ is not merely a theoretical nicety; it is a practical necessity. Omitting it leads to poor discrimination with Meta-Pi backpropagation.

Thus, the Meta-Pi network approximates the Bayesian discriminant function by learning to estimate the relevance $P(M_{\pi_k}|\boldsymbol{x})\ \forall k$ so that it can form a robust approximation of the *a posteriori* class distributions $P(\omega_i|\boldsymbol{x})\ \forall i$ in (8) via the relationship of (10). It is important to note that the Meta-Pi network learns to compute these relevance coefficients without any explicit knowledge of the input source identity. The Meta-Pi combinational superstructure's parameters are adjusted solely on the basis of how well the global network performs the global classification task—this is consistent with the Bayesian maximum *a posteriori* (MAP) description of the Meta-Pi paradigm presented in Section IV-B. As a result, there is only indirect supervision of the Meta-Pi network as it learns its combinational function.

B. Meta-Pi Training

As mentioned earlier, the difference of probabilistic interpretation between the Meta-Pi and SID training procedures leads to a procedural difference in the combinational superstructure's training.

1) Procedural Differences between Meta-Pi and SID Training: A comparison of Figs. 1 and 3 illustrates the procedural difference between SID and Meta-Pi training when the task is speech recognition. The SID training procedure involves training a set of source-dependent classifiers on the desired task and *independently* training a combinational superstructure using a training set comprising all of the source-dependent training data (Fig. 1 inset). For the Meta-Pi training procedure, source-dependent modules are trained the same way they are for the SID procedure *before* the combinational superstructure is trained (Fig. 3 inset). The combinational superstructure is then trained. Initially, its parameters are in an arbitrary random state. The superstructure performs its combinational function on the outputs of the source-dependent modules—each module processes each training sample and presents a classification output to the Meta-Pi superstructure. The superstructure processes the same training sample and produces a global classification output by forming a linear combination of the module outputs. As the combinational superstructure's parameters are initially random, so is the linear combination forming the global output. Training the superstructure therefore involves searching its parameter space to optimize the metric used to evaluate the *global* performance of the structure. In this way, the parameters of the combinational superstructure are altered so that the linear combination of (10) yields a valid classification for each training token. Thus, the *a posteriori* probability associated with the correct phoneme in (10) is maximized, and the Meta-Pi paradigm constitutes a connectionist MAP learning procedure.

Since we use TDNN MLP classifiers for the modules and the combinational superstructure of the Meta-Pi architecture, the training procedure requires an alteration of the backpropagation algorithm.

2) Meta-Pi Backpropagation: Fig. 3 illustrates the Meta-Pi modular structure used to combine the C outputs of K source-dependent modules trained to perform the same classification task. The layout of this figure is analogous to that of Fig. 1.[12] Thus

$$O_i = \frac{1}{\hat{u}} \sum_K \rho_{k,i} \cdot M_{\pi_k} \tag{14}$$

where, again, $\hat{\mu}$ is given in (12) and (13).

The continuous-valued Meta-Pi output unit $0 \leq M_{\pi_k} \leq 1$ modulates or gates the continuous-valued output $\rho_{k,i}$ ($0 \leq \rho_{k,i} \leq 1$) to form the global output O_i. Much as the Meta-generalized delta rule [31] uses one connection (or synapse) to modulate the value of another connection, the Meta-Pi network uses its continuous output state as a connection that modulates the output state of another *network*. Owing to the probabilistic motivation of Section IV-A, the output unit O_i of the global structure does *not* perform a thresholding or "squashing" function. O_i is a linearly scaled version of $\rho_{k,i}$—the output of a connectionist structure previously trained to perform the same classification task demanded of the global network. Intuitively, one would expect the Meta-Pi network to learn to pass the output $\rho_{k,i}$ through to the global output O_i when $\rho_{k,i}$ represents a correct classification of the input and to withhold $\rho_{k,i}$ from the global output when it represents an incorrect classification.

For the case in which one is using an error measure objective function [7] E, based on the global outputs $O_1, \ldots, O_C$ (such as the mean-squared-error (MSE) or cross-entropy (CE) objective function)

$$\frac{\partial E}{\partial M_{\pi_k}} = \frac{\partial E}{\partial \overline{O}} \cdot \frac{\partial \overline{O}}{\partial M_{\pi_k}} = [\nabla_O E]^T \cdot \nabla_{M_{\pi k}} \overline{O}. \tag{15}$$

One uses the expression $\frac{\partial E}{\partial M_{\pi_k}}$ and the backpropagation chain rule to determine[13] $\nabla_w E$. One then adjusts the parameters ("weights" or "connections") of the Meta-Pi network to optimize the global output O_i. Thus, Meta-Pi backpropagation is quite similar in form to cascaded backpropagation [29].

From (14) and (15)

$$\frac{\partial O_i}{\partial M_{\pi_k}} = \frac{1}{\hat{u}} [\rho_{k,i} - O_i]. \tag{16}$$

If $D_1, \ldots, D_C$ represent the target global output state of the Meta-Pi structure in Fig. 3, and one uses the modified MSE expression $\mathcal{E}$ [36], [3] where

$$\mathcal{E} \triangleq \frac{1}{2} \sum_{i=1}^{C} (O_i - D_i)^2 = \frac{C}{2} MSE \tag{17}$$

then

$$\frac{\partial E}{\partial O_i} = O_i - D_i \tag{18}$$

and (15), (16), and (18) combine to yield

$$\frac{\partial E}{\partial M_{\pi_k}} = \frac{1}{\hat{u}} \cdot \sum_{i=1}^{C} \{(O_i - D_i) \cdot [\rho_{k,i} - O_i]\}. \tag{19}$$

[12] A more tutorial description of Meta-Pi backpropagation can be found in [8].

[13] $\nabla_w E$ denotes the Meta-Pi network's parameter-space gradient of the error function E.

$$CFM_{MF} \triangleq \begin{cases} \frac{-\alpha}{\mathcal{C}-1} \sum_{\substack{i=1 \\ i \neq \tau}}^{\mathcal{C}} \log_e \left[1 + (\zeta - \Delta_i)^{2\beta}\right] & \zeta - \Delta_i < 0 \\ 0 & \zeta - \Delta_i \geq 0 \end{cases} \tag{32}$$

$$\frac{\partial CFM_{MF}}{\partial \Delta_i} = \begin{cases} \frac{2\alpha\beta}{\mathcal{C}-1} \frac{(\zeta - \Delta_i)^{2\beta-1}}{1+(\zeta-\Delta_i)^{2\beta}} & \zeta - \Delta_i < 0 \\ 0 & \zeta - \Delta_i \geq 0 \end{cases} \quad \forall\, i \neq \tau. \tag{33}$$

For the cross entropy objective function (e.g., [12])

$$CE = -\sum_{i=1}^{\mathcal{C}} \{D_i \log(O_i) + (1 - D_i) \log(1 - O_i)\} \tag{20}$$

$$\frac{\partial CE}{\partial O_i} = -\sum_{i=1}^{\mathcal{C}} \left\{ \frac{D_i}{O_i} - \frac{1 - D_i}{1 - O_i} \right\} \tag{21}$$

and (15), (16), and (21) yield

$$\frac{\partial CE}{\partial M_{\pi_k}} = -\frac{1}{\hat{u}} \sum_{i=1}^{\mathcal{C}} \left\{ \left[\frac{D_i}{O_i} - \frac{1 - D_i}{1 - O_i} \right] \cdot (\rho_{k,i} - O_i) \right\}. \tag{22}$$

For objective functions $\mathcal{M}$ based on differences between the global outputs $\{\Delta_1, \ldots, \Delta_{\mathcal{C}-1}\}$ such as the CFM objective function [9] for which

$$\Delta_i \triangleq O_\tau - O_i \quad i \neq \tau \tag{23}$$

$O_\tau \equiv$ the global output representing the correct classification of the input

$O_i \equiv$ the global output representing the ith *in*correct classification of the input signal (24)

we find

$$\frac{\partial \mathcal{M}}{\partial M_{\pi_k}} = \frac{\partial \mathcal{M}}{\partial \overline{\Delta}} \cdot \frac{\partial \overline{\Delta}}{\partial M_{\pi_k}} = \left[\nabla_{\overline{\Delta}} \mathcal{M}\right]^T \cdot \nabla_{M_{\pi_k}} \overline{\Delta} \tag{25}$$

Note that

$$\Delta_i = \frac{1}{\hat{u}} \sum_{K} (\rho_{k,\tau} - \rho_{k,i}) \cdot M_{\pi_k} \quad \forall\, i \neq \tau \tag{26}$$

$$\frac{\partial \Delta_i}{\partial M_{\pi_k}} = \frac{1}{\hat{u}} \left[(\rho_{k,\tau} - \rho_{k,i}) - \Delta_i\right] \quad \forall\, i \neq \tau. \tag{27}$$

The standard (sigmoidal) CFM objective function is given by [9][14]

$$CFM_\sigma \triangleq \frac{1}{\mathcal{C}-1} \sum_{\substack{i=1 \\ i \neq \tau}}^{\mathcal{C}} \frac{\alpha}{1 + e^{-(\beta\Delta_i + \zeta)}} \tag{28}$$

$$\frac{\partial CFM_\sigma}{\partial \Delta_i} = \frac{1}{\mathcal{C}-1} \cdot \alpha\beta \cdot y_i(1 - y_i) \quad \forall\, i \neq \tau \tag{29}$$

where

$$y_i \triangleq \frac{1}{1 + e^{-(\beta\Delta_i + \zeta)}} \quad \forall\, i \neq \tau. \tag{30}$$

From (23), (25), (27), (29), and (30)

$$\frac{\partial CFM_\sigma}{\partial M_{\pi_k}} = \frac{1}{\mathcal{C}-1} \cdot \frac{\alpha\beta}{u} \cdot \sum_{\substack{k\,reli=1 \\ i \neq \tau}}^{\mathcal{C}} \{y_i(1 - y_i) \cdot [(\rho_{k,\tau} - \rho_{k,i}) - \Delta_i]\}. \tag{31}$$

The maximally flat CFM objective function is given by [9].[15] (See (32) and (33) at the top of this page.) From (23), (25), (27), (32), and (33)

$$\frac{\partial CFM_{MF}}{\partial M_{\pi_k}} = \frac{1}{\mathcal{C}-1} \frac{2\alpha\beta}{\hat{u}} \cdot \sum_{\substack{i=1 \\ i \neq \tau \\ \zeta - \Delta_i < 0}}^{\mathcal{C}} \left\{ \frac{(\zeta - \Delta_i)^{2\beta-1}}{1 + (\zeta - \Delta_i)^{2\beta}} \cdot [(\rho_{k,\tau} - \rho_{k,i}) - \Delta_i] \right\}. \tag{34}$$

Equations (15) and (25) are general expressions describing the global error signal's propagation from the global output back to the Meta-Pi combinational superstructure's output stage (note the gray arrows in Fig. 3). As mentioned before, the standard backpropagation chain rule governs the propagation of the error signal back through the Meta-Pi network from this point; from its output units $M_{\pi_1}, \ldots, M_{\pi_K}$ back to its input stage, the Meta-Pi network is a standard MLP—in our case, a TDNN.

V. Multispeaker Phoneme Recognition Using the Meta-Pi Architecture

Table II compares the error rates of the Meta-Pi and SID networks. There is no significant difference between the discrimination of three-way arbitrated and single CFM-trained Meta-Pi combinational superstructures. For this reason, we limit our analysis to the single CFM-trained superstructure since it is one third the size of and requires one third the training time of the three-way arbitrated superstructure.

[14] One can omit the scaling factor of $\frac{1}{\mathcal{C}-1}$ for computational efficiency, making a commensurate adjustment in the objective function's derivatives.

[15] One can omit the scaling factor of $\frac{1}{\mathcal{C}-1}$ for computational efficiency, making a commensurate adjustment in the objective function's derivatives.

TABLE II
A COMPARISON OF SIX-SPEAKER /b,d,g/ ERROR RATES FOR THE META-PI AND SID COMBINATIONAL SUPERSTRUCTURES.

	Actual Combinational Schemes				Artificial Combinational Schemes	
	Meta-Pi	SID Proportional	SID Winner-take-all	Normalized Gross Sum	Gender known	SID known
3-Way Arbitrated Superstructure						
	1.7 (+/- 0.4)%	1.7 (+/- 0.4)%	1.9 (+0.5/-0.4)%	21.2 (+/- 1.3)%	5.5 (+/- 0.7)%	1.3 (+/- 0.4)%
Single CFM-trained Superstructure						
with MHT module	1.9 (+0.5/-0.4)%	2.1 (+/- 0.5)%	2.3 (+/- 0.5)%	6 speakers MHT other 5 speakers		
	0.8 (+0.8/-0.7)%	0.6 (+0.8/-0.6)%	0.6 (+0.8/-0.6)%			
	2.1 (+/- 0.5)%	2.4 (+0.6/-0.5)%	2.7 (+/- 0.6)%			
without MHT module	2.6 (+/- 0.5)%	8.0 (+/- 0.9)%		6 speakers MHT other 5 speakers		
	5.8 (+2.0/-1.8)%	37.2 (+3.9/-3.8)%				
	1.9 (+/- 0.5)%	2.3 (+/- 0.5)%				

The Meta-Pi network's error rate of 1.9 (+0.5/-0.4)% is statistically equivalent to the SID network's 2.1 (+/- 0.5)% ; both of these rates are statistically equivalent to the 1.3 (+/- 0.4)% average error rate of the speaker-dependent modules, which has been obtained for the ideal case in which the speaker's identity is explicitly known ("SID known" column).

The Meta-Pi combinational superstructure learns early during the training phase that gender plays a critical role in accurate phoneme recognition. As a result, it learns—without direct supervision—to group speakers by gender. Fig. 4 illustrates this phenomenon. The figure shows the unique connections between the input layer of the Meta-Pi combinational superstructure and its first hidden layer.[16] Twelve groups of connections depict the weights linking three 16-coefficient spectra (48 units) of the input layer to each of 12 first-hidden-layer units. Positive connections are white, negative connections are black, and the magnitude of each connection is proportional to the size of its corresponding rectangle. It is clear from Fig. 4 that the input to first-hidden-layer connections are block like (i.e., positive and negative connections to each first-hidden-layer unit are clustered in regular blocks). These blocks tend to correspond to formant[17] locations for the various speakers. The figure illustrates two formant features ($F2/F3$ separation and the presence of low-frequency (LF) energy) that the Meta-Pi combinational superstructure has learned to use for detecting male speech. Likewise, the superstructure has learned that a relatively high-frequency third formant ($F3$) indicates a female's speech. It has learned to rely on formant characteristics in order to minimize its global error rate. Fig. 5 illustrates the connection strengths and state formed by the SID combinational superstructure described in Section III for the same utterance shown in Fig. 4. There are notable similarities between Figs. 4 and 5 despite the fact that the Meta-Pi combinational superstructure was *not* explicitly trained to perform speaker identification, whereas the SID superstructure was. An interesting feature that distinguishes the SID combinational superstructure from the Meta-Pi superstructure is the finer detail of its connections. Where the Meta-Pi connections are block like, suggesting that the superstructure relies on relatively gross formant features to perform its combinational function, the SID superstructure's connections are less regular. We surmise that this is because the SID superstructure has learned a number of speaker-dependent spectral characteristics that are more detailed than the gross formant features learned by the Meta-Pi superstructure.

[16] See [19], [40] for details of the TDNN's temporally constrained connection topology.

[17] Formants are resonances in the vocal tract of a speaker. They are abbreviated Fi, where $F1$ denotes the formant with the lowest center frequency [28].

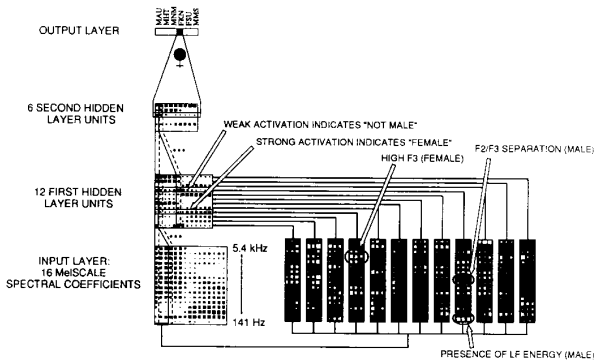


Fig. 4. Meta-Pi network's input-to-hidden-layer connections use gross formant features of the speech input signal to form an estimate of the *a posteriori* class probabilities of (10).

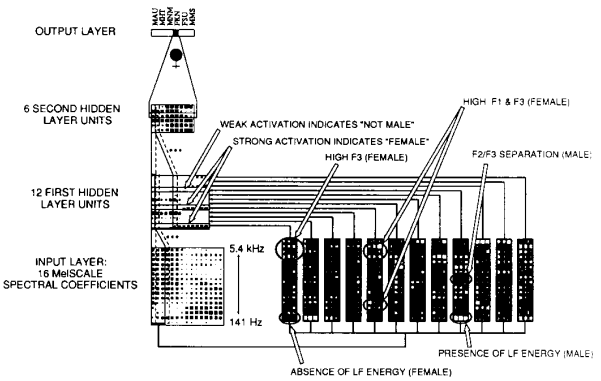


Fig. 5. SID network's input-to-hidden-layer connections. These connections evolved from explicitly learning the task of speaker identification. Note the similarities with the Meta-Pi network's connections in Fig. 4.

Fig. 4 illustrates that the Meta-Pi combinational superstructure is capable of specific speaker identification (only the output associated with speaker FKN is active). The percentage of utterances for which the superstructure identifies a single module for the global recognition task is actually low—less than 30%. Fig. 6 illustrates a much more typical mode of operation. In this figure, the speech token is actually uttered by female FSU, but the superstructure associates the input signal with both female modules and produces a correct global recognition result. Apparently, the combinational superstructure has learned to perform explicit speaker identification when the input signal possesses features that are unique to a particular individual. In addition, it has learned that many utterances are prototypical of a *group* of speakers (e.g., males or females). In cases where the utterance is prototypical, the Meta-Pi combinational superstructure attributes the speech to the collection of speakers it associates with the prototype.

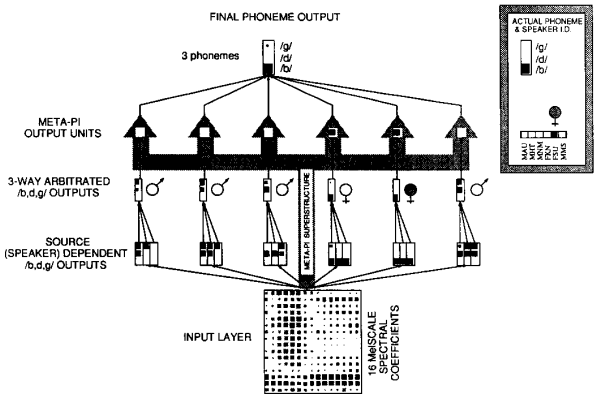


Fig. 6. Meta-Pi superstructure attributes a /b/ phone to both female speakers.

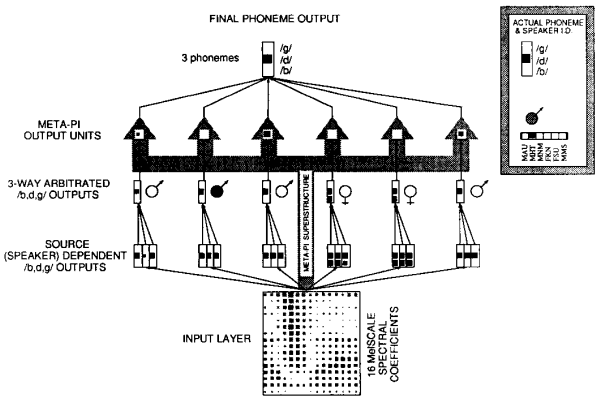


Fig. 7. Meta-Pi superstructure recognizes the speech of male MHT using a dynamic combination of *other* male speaker modules.

Figs. 4 through 6 give empirical evidence that the Meta-Pi network does indeed constitute a Bayesian MAP classifier capable of robust multisource pattern recognition. Furthermore, the figures suggest that the combinational superstructure uses a more general, less specific model of source (speaker) *type* when the input signal is prototypical and a more specific model of the source when the input signal is unique. These observations are consistent with the probabilistic differences between (6) and (10). The question remains: "Does the more general Meta-Pi combinational function produce a global system that is intrinsically extensible to novel sources?" There is evidence that suggests the answer to this question is affirmative. Fig. 7 illustrates the Meta-Pi network processing the voiced stop /d/ uttered by speaker MHT. A notable aspect of this figure is that the Meta-Pi combinational superstructure does not use the MHT module to recognize this utterance. In fact, when fully trained, the superstructure *never* uses the MHT module to classify speech tokens from any of the other five speakers, and it rarely uses the MHT module to classify tokens from speaker MHT. Indeed, it learns to model speaker MHT most often with a dynamic combination of *other* male speakers and, in so doing, still achieves a 0.8 (+0.8/-0.7)% error rate on the speech of MHT. An analysis of this phenomenon leads to the following explanation.

Speaker MHT is a very clear speaker. Using a single TDNN, we can recognize all but 0.2 (+0.5/-0.2)% of this male's speech [9]. This suggests that the *a posteriori* distributions of his voiced-stop phonemes are nearly separable (thus, our "clear" characterization). However, a TDNN trained on MHT but used to recognize the speech of *another* speaker (MAU) does quite poorly (21.9 (+3.3/-3.2)%). Conversely, a TDNN trained on the speech of MAU and used to recognize the speech of MHT achieves a 9.4 (+2.4/-2.3)% error rate. During the training phase, the Meta-Pi combinational superstructure finds the following:

1. It can usually model MHT's speech using combinations of other male modules.
2. There is no utility in using the MHT module to recognize speech tokens from any of the other speakers. Although the MHT module's recognition rate is high for one sixth of the training data (MHT's speech), it is low for the remaining five sixths of the training data.

As a result, the Meta-Pi combinational superstructure learns that there is no marginal utility in using the MHT module. In effect, it removes the MHT module from its "module database." Indeed, Table II shows that if we physically remove the MHT module from the trained Meta-Pi network, we find that its error rate for MHT speech increases moderately to 5.8 (+2.0/-1.8)% , whereas the error rate for the other five speakers remains unchanged (the decrease from 2.1 (+/- 0.5)% to 1.9 (+/- 0.5)% is not statistically significant). Note that the SID network's error rate for MHT increases to 37.2 (+3.9/-3.8)% when the MHT module is removed. This result has implications both for source-independent pattern recognition and for fault tolerance, which we consider in the following two sections.

VI. A Limited Speaker-Independent Experiment with the Meta-Pi Architecture

Since the six-speaker Meta-Pi network usually models speaker MHT with a combination of other source-dependent modules, it is logical to ask whether a five-speaker Meta-Pi network—trained on the speech of all the speakers except MHT—can recognize MHT's speech in a limited source-independent experiment. We trained such a five-speaker Meta-Pi network along with a comparable SID network and a single TDNN. Fig. 8 shows a comparison of the speaker-dependent modules ("speaker-dependent averages"), the six- and five-speaker TDNN's, and the six- and five-speaker Meta-Pi and SID networks. Separate error rates are shown for MHT and the other five speakers. The speaker-dependent module error rates are shown because they approximate the best performance achievable. The TDNN error rates are shown for control purposes since the single TDNN is the monolithic classifier against which we are comparing the SID and Meta-Pi architectures. Note that for the multispeaker experiment (i.e., the case in which each classifier is trained with the speech of all six speakers), the TDNN, SID, and Meta-Pi classifiers all yield MHT error rates that are statistically equivalent to the MHT-dependent error rate. However, the SID and Meta-

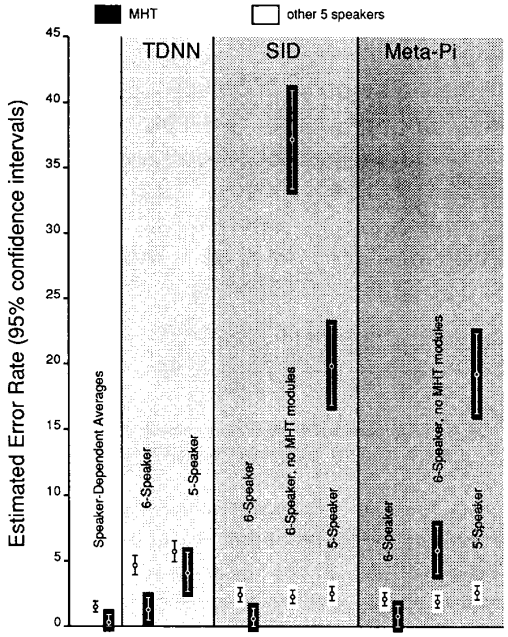


Fig. 8 Comparison of /b,d,g/ error rates for speaker-dependent modules, TDNN, SID, and Meta-Pi classifiers.

Pi classifiers yield error rates for the other five speakers that are statistically equivalent to the 1.3 (+/- 0.4)% speaker-dependent average, whereas the TDNN's error rate for this group of speakers is significantly higher at 4.7 (+0.8/-0.7)% .

As mentioned in the previous section, we see that when the MHT module is physically removed from the SID and Meta-Pi networks, the SID classifier's MHT error rate increases to 37.2 (+3.9/-3.8)% , whereas the Meta-Pi classifier's error rate increases to 5.8 (+2.0/-1.8)% . Note that the SID and Meta-Pi error rates for the other five speakers remain essentially unchanged at 2.3 (+/- 0.5)% and 1.9 (+/- 0.5)% , respectively, when the MHT module is removed.

For the limited source-independent experiment, we see that the five-speaker TDNN exhibits an error rate of 4.1 (+1.7/-1.6)% on the novel source (MHT) and an error rate of 5.7 (+/- 0.8)% on the known sources (i.e., the other five speakers). In contrast, the SID network exhibits an error rate of 19.8 (+3.2/-3.1)% on the novel source and an error rate of 2.5 (+0.6/-0.5)% on the known sources. Finally, the Meta-Pi network's error rates are virtually identical to those for the SID network: 19.2 (+3.2/-3.1)% for the novel source and 2.6 (+/- 0.6)% for the known sources. We see for this limited source-independent experiment that the TDNN's error rate on the novel source is comparable with its error rate on the known sources; both are significantly higher than the source-dependent error rates for this group of speakers. Although the SID and Meta-Pi classifiers yield source-dependent error rates for the known sources, their error rates for the novel source are approximately one order of magnitude higher. These results are consistent with the acuity/generality tradeoff described in Section II-A; we discuss their significance further in Section VII-C.

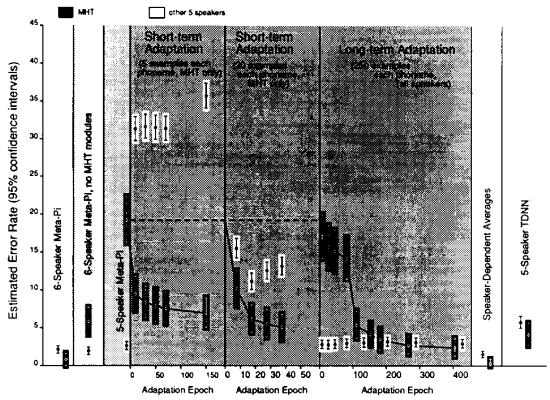


Fig. 9. Short- and long-term adaptation of the five-speaker Meta-Pi network to the novel speaker MHT. Short-term adaptation uses 5 and 20 training examples of each phoneme /b,d,g/ for speaker MHT *only*; long-term adaptation uses the combined training sets of all six speakers.

It is clear from Fig. 8—at least for this particular scenario—that neither the Meta-Pi nor the SID classifier is intrinsically extensible to the novel source without any modification to the combinational superstructure and/or the source-dependent modules. Because the SID network is explicitly trained to identify sources, we would have to add an MHT module to the five-speaker SID network, expand the combinational superstructure to handle this new model, and retrain the superstructure on the speech of all six speakers. This is tantamount to building the original six-speaker SID network. We refer to this as *very long term* adaptation. Because the Meta-Pi combinational superstructure is not explicitly trained to identify sources, it is possible (at least in principle) to adapt the five-speaker Meta-Pi network to the novel source by modifying the parameterization of the combinational superstructure alone—something that cannot be done with the SID architecture.[18] Fig. 9 illustrates the results of this adaptation.

In the "short-term adaptation—5 examples" portion of this figure, we adapt the fully trained five-speaker Meta-Pi combinational superstructure to five examples of each of the three /b,d,g/ phonemes spoken by MHT—a process we call *short-term adaptation*. After 150 training epochs (i.e., 150 iterations of the backpropagation algorithm on these 15 adaptation examples), the Meta-Pi network's error rate on MHT's speech drops from 19.2 (+3.2/-3.1)% to 6.9 (+2.1/-2.0)% . The reparameterized superstructure's error rate on the other five speakers jumps to 35.7 (+/- 1.7)% , but this is not problematic because we use the reparameterized superstructure only to recognize MHT. We use the original fully trained five-speaker superstructure to recognize the known sources. As we see in the "short-term adaptation—20 examples" portion of Fig. 9, if we adapt the fully trained five-speaker Meta-Pi combinational superstructure to 20 examples of each of the three phonemes, the MHT error rate is 5.0 (+1.8/-1.7)% —not

[18] Granted, there are ways in which the SID superstructure could adapt to a novel source. For example, one could determine which (if any) of the source-dependent modules correctly classifies each input from the novel source and train the SID to attribute the input to the module that best classifies the input. However, such a technique is a crude form of Meta-Pi learning, and therefore, the resulting structure is no longer an SID architecture.

significantly better than the "5–example adaptation" result. It is interesting to compare the impacts of the smaller and larger short-term adaptation sets on the reparameterized classifier's error rate for the other five speakers (note the black error bars on white backgrounds).

Finally, if we adapt the fully trained five-speaker Meta-Pi combinational superstructure to all of the MHT training examples, along with all the training examples for the other five speakers—a process that we call *long-term adaptation*—we achieve an error rate of 2.4 (+1.3/-1.2)% on MHT and an error rate of 2.9 (+/- 0.6)% on the other five sources. In all these adaptation experiments, the error rate for MHT is comparable to the five-speaker TDNN's rate (the adaptation result is insignificantly lower) but somewhat higher than the original six-speaker Meta-Pi's rate as shown in Fig. 9. In the long-term adaptation experiment, the 2.9 (+/- 0.6)% error rate for the other five speakers is higher than the 1.5 (+0.5/-0.4)% speaker-dependent average but is comparable to the six-speaker Meta-Pi network's 2.1 (+/- 0.5)% rate and lower than the five-speaker TDNN's 5.7 (+/- 0.8)% rate.

Thus, for this particular scenario, both long- and short-term adaptation (involving reparameterization of the combinational superstructure without the addition of a source-dependent module) allow the Meta-Pi network to yield error rates on the novel source that are comparable to those of a monolithic classifier (i.e., the five-speaker TDNN control) and slightly higher than those of a source-dependent classifier. This is achieved without degrading the source-dependent error rates obtained on the other five known sources. We discuss these adaptation results further in Section VII-C.

VII. Discussion

Our discussion of the Meta-Pi and SID paradigms has three components: a review of the architectures' connectionist properties, a discussion of the significance of our experimental findings, and a review of questions that need to be answered in order for the Meta-Pi paradigm to be used in real-world pattern recognition systems.

A. Connectionist Properties

In Section II-B, we outlined a number of properties that we thought desirable for a modular connectionist classifier. We now review and compare the SID and Meta-Pi architectures in the context of these properties.

1. **Size of the Meta-Pi and SID Modular Nets:** The Meta-Pi and SID modular networks use three TDNN's for each speaker-dependent module and one somewhat larger TDNN for the combinational superstructure—a total of 19 TDNN's (or approximately 123 000 connections). At face value, an increase of almost 1.3 orders of magnitude in the number of connections for the overall six-source task of Section V to achieve a 52% error reduction seems inequitable. We believe the most immediate way to reduce the size of the Meta-Pi structure studied herein is to eliminate the redundant networks used for conflict arbitration at the module level. Ongoing research associated with [9] suggests that we can achieve the same source-dependent error rates without conflict arbitration. Such a six-speaker Meta-Pi structure for the /b,d,g/ task would have only seven TDNN's with 48 400 connections. Of these connections, only about 10% would be unique, owing to the architectural structure of the TDNN. New algorithms for training TDNN's allow one to achieve convergence rates proportional to the number of unique connections in the TDNN when training is done on continuous (rather than isolated) speech [43]. Thus, we can expect to implement the six-speaker Meta-Pi structure with—in effect—fewer than 5000 connections in the future.
2. **Training Set Required:** Each of the speaker-dependent modules' TDNN's requires roughly 250 training tokens for each of the three phonemes it must learn to classify ($\sim$ 750 tokens). The SID superstructure uses all the tokens used to train the six speaker-dependent modules ($\sim$ 4500) to learn its speaker-identification task. The training set for the Meta-Pi network is virtually identical to that for the SID network. The theoretical results of [7] along with empirical studies in [9] suggest that it is *not* possible to reduce the size of the source-dependent modules' training sets and/or the SID combinational superstructure's training set without an appreciable increase in error rate. However, the adaptation results of Section VI suggest that it may be possible to train the combinational superstructure of the Meta-Pi classifier with fewer examples than the number obtained by combining all the source-dependent training sets. Further research is necessary to make a conclusive statement on this point.
3. **Training Time:** The speaker-dependent module TDNN's train relatively quickly; therefore, it is possible to train all the modules in a matter of hours on the WARP systolic array, which has a maximum sustained rate of 17 million connections/s [30]. Because the SID/Meta-Pi combinational superstructures are somewhat larger than the module TDNN's and their training set size is six times that of a single module's TDNN's, their total training time is comparable with the time required to train all of the module subnets. Thus, the overall training time for the modular structure is proportional to the number of modules combined by the superstructure—this represents a reasonable degree of computational efficiency for this size task.[19] Since the training of the SID combinational superstructure and its modules is completely independent, it is possible to train modules and the superstructure simultaneously on different processors (Fig. 1). Because the Meta-Pi superstructure requires the output states of its composite modules as part of its training data, it cannot be trained independent of its modules. The modules must be trained first; only then can the combinational superstructure be trained (Fig. 3). This is a significant difference since it is, in principle, possible to train the SID modules and

[19] Waibel *et al.* [41] compare the computational efficiency of various monolithic and modular connectionist approaches to consonant recognition tasks; this comparison serves as the basis for our assertion.

combinational superstructure simultaneously on different processors; such a scheme would not be possible for the Meta-Pi architecture.

During training, the Meta-Pi superstructure takes approximately three times as long as its SID counterpart to converge to its optimal parameterization (the six-speaker SID combinational superstructure converges in approximately 150 epochs; the equivalent Meta-Pi superstructure converges in approximately 450 epochs). This disparity can be attributed to the fact that the error signal propagated back through the SID network is based on a unique input-to-output mapping specified by the explicit source identification objective. For the Meta-Pi network, there is usually a manifold of input-to-output mappings that yield the correct global classification output for any given input. Since most training tokens have a nonunique optimal Meta-Pi superstructure state, the point in the Meta-Pi superstructure's parameter space that is optimal for *all* training tokens is in a broad region of near-optimal parameterizations. This leads to small gradient terms in Meta-Pi backpropagation using gradient descent, which in turn account for the slower convergence rate.

4. **Ease of Modification:** The SID and Meta-Pi modular structures are easily modified in very long-term adaptation. A new speaker-dependent module can be trained independently and then added to the module layer; the combinational superstructure can then be expanded to account for the new module—beginning with the previously learned connections corresponding to the original set of modules—and retrained on the expanded global SID training set. Thus, the computational cost of a very long-term modification to both modular architectures is equal to the cost of training one source-dependent module and one combinational superstructure. As the number of modules increases, this cost becomes a proportionally smaller fraction of the cost of retraining a monolithic classifier. It seems unlikely, however, that the discrimination of the SID superstructure would remain robust if the number of modules it combines were to grow large; the speaker identification task would probably become confounding and result in a degradation of global system discrimination. Additionally, the limited extensibility of the SID structure (see below) does not bode well for the architecture's discrimination in a speaker-independent mode.

   We expect that problems would also arise with a comparably large Meta-Pi network for the simple reason that the combinational superstructure would eventually suffer from its own form of the acuity/generality tradeoff described in Section II-A. Specifically, the same sound can represent different phonemes for different speakers. Using the techniques outlined in (10) and [7], it is relatively straightforward to show that a single Meta-Pi superstructure trained on the speech from a large body of heterogeneous speakers would yield ambiguous phoneme classification outputs for an input that could be associated with different phonemes for different speakers.

5. **Extensibility to Novel Sources:** We review the three levels of adaptation described in Section VI based on the assumption that the classifier comprises a collection of source-dependent modules linked by a combinational superstructure; listed in order of increasing computational complexity and data requirements, they include the following:

   a. **Short-term adaptation**: This adaptation involves a *temporary* reparameterization of the combinational superstructure in order to recognize the novel source. No source-dependent modules are added to the classifier, and the amount of training data for the adaptation is limited. The adaptation process begins with the superstructure's parameters for the known sources. Once the classifier has finished processing the novel source, the superstructure's parameters are reset to their values before the adaptation; as a result, the adaptation is temporary.

   b. **Long-term adaptation**: This adaptation is similar to short-term adaptation, except that the amount of training data for the adaptation is large—as much data as one would need to build a source-dependent classifier of the source. This data is combined with the training data for all the known sources, and the combinational superstructure is reparameterized so that it learns to recognize the novel source using existing models of known sources. The reparameterized superstructure then forms a classifier that recognizes all the original known sources as well as the new source without using a module trained on the speech of the new source. This adaptation is *permanent.*

   c. **Very long-term adaptation**: This adaptation involves adding a source-dependent module for the new source, expanding the combinational superstructure, and retraining it to form a new, expanded multisource classifier. It is *permanent.*

6. The Meta-Pi architecture can perform all three forms of adaptation, whereas the SID architecture performs only very long-term adaptation. As illustrated by Section VI, short-term adaptation can, in some cases, yield acceptable error rates with few examples of the novel source. It is also clear from Section VI that some sources can be modeled effectively with long-term adaptation. We see this performance characteristic as having the potential for self-maintaining pattern recognition systems—systems that can develop and maintain their own database of known sources, adapting to new sources when possible, spawning new source-dependent learning processes when necessary, and eliminating redundant or obsolete source-dependent modules when appropriate.
7. **Error Propagation and Fault Tolerance:** Because the SID superstructure is trained independent of the speaker-dependent modules that it combines, it simply selects the module to decode the given utterance, independent of whether or not the module correctly classifies the input

signal. In contrast, the Meta-Pi superstructure learns to form its final classification decision from only those source-dependent modules that correctly classify the input signal. This difference appears to be significant if the modular structure is damaged by the failure of a source-dependent module. We found that the Meta-Pi classifier could still recognize speaker MHT reasonably well when the MHT module was removed, whereas the SID network had a relatively large increase in its MHT error rate. We do not claim that the Meta-Pi network will maintain low error rates for every known source whose module is removed from the structure; to the contrary, we think this unlikely. However, we do feel the MHT results show that the Meta-Pi structure is more tolerant of such failures than its SID counterpart.

B. Significance of Results

As mentioned early on, the error rates we quote throughout this paper are estimates based on each classifier's recognition of a random test sample that is disjoint from the sample used to train the classifier. We assume that the estimated error rate is binomially distributed and quote the upper and lower bounds of a 95% confidence interval based on this assumption [11]. Two error rates are judged statistically equivalent if their confidence intervals overlap. This gives the reader a picture of the statistical significance of the results for each classifier, given the particular *a posteriori* class distributions on the input RV $\boldsymbol{x}$ and a particular parameterization for the classifier. However, it gives no assessment of the classifier's expected discrimination given a training set with any *a posteriori* class distributions on $\boldsymbol{x}$ and any parameterization judged to be best for that training set.[20] That is, perhaps the *a posteriori* class distributions on $\boldsymbol{x}$ happen to be unusually easy (or hard) to model, or perhaps we have been particularly lucky (or particularly unlucky) in training the classifier.

Distribution-independent confidence bounds on the deviation of empirical from true best error rates for MLP classifiers based on training sample size are still being debated (see the works of Baum, Cover, DeVroye, Haussler, and Vapnik-Chervonenkis). Therefore, rather than compute theoretical bounds, we refer the reader to [9]. Based on experiments detailed therein, we doubt that the error rates estimated for source-dependent modules in this paper deviate from the true best error rates by more than 1.5%. It is less clear what the maximum deviation of the empirical from true best error rates are for the overall Meta-Pi and SID classifiers. In the limited number of six- and five-speaker superstructure training runs we have made, the error rates on the five known sources for the overall classifier have been tightly clustered (+0.6/-0.3%) around 2.0% (eight trials).

Beyond the statistical significance of our results is their importance to larger real-world pattern recognition systems. We caution the reader that our experiments to date are not sufficiently complex, nor do they involve a sufficiently large number of heterogeneous sources, to be viewed as typical of the Meta-Pi classifier's discrimination for the general multi-source and source-independent pattern recognition tasks. Instead, they should be viewed as a feasibility study—a set of experiments to determine if further research aimed at implementing the Meta-Pi paradigm for a large-scale multi-source/source-independent pattern recognition tasks is worthwhile. Based on our results, we conclude that further study is warranted; there are many issues to be explored.

[20] Devroye makes an important distinction between the *best* error rate a particular family of classifiers can achieve and the *optimal* error rate achieved by the Bayesian discriminant function [4].

C. Remaining Questions

The questions that must be answered in order to evaluate the Meta-Pi classifier for large real-world multisource/source independent tasks include the following:

Are the multisource and source-independent results achieved in this study representative of the Meta-Pi network's characteristics for much larger multisource and source-independent tasks? Our opinion is that the multisource and source-independent results for speaker MHT are likely better than those for the typical case, owing to the nice statistical properties of the subject's speech. Speaker MHT was the only speaker among the six we studied whom we could model using only other source-dependent modules. Attempting this with any of the other speakers yielded unacceptably high error rates. How well the Meta-Pi network does in classifying patterns from a novel source undoubtedly depends on the number of source-dependent modules it has.

Does the disparity between a monolithic and modular classifiers' error rates change for larger tasks? If so, does this change argue for or against the modular approach? The operative equation governing the comparison of a modular and monolithic classifiers' error rates is

$$\begin{aligned} \mathrm{P}(\mathrm{Error}) &= \mathrm{P}(\mathrm{Error} \mid \mathrm{Known\ source})\,\mathrm{P}(\mathrm{Known\ source}) \\ &\quad + \mathrm{P}(\mathrm{Error} \mid \mathrm{Unknown\ source})\,\mathrm{P}(\mathrm{Unknown\ source}) \\ &= [\mathrm{P}(\mathrm{Error} \mid \mathrm{Unknown\ source}) \\ &\quad - \mathrm{P}(\mathrm{Error} \mid \mathrm{Known\ source})]\,\mathrm{P}(\mathrm{Unknown\ source}) \\ &\quad + \mathrm{P}(\mathrm{Error} \mid \mathrm{Known\ source}). \end{aligned} \tag{35}$$

This error rate is not only a function of each classifier's error rate for known and unknown sources, it is also a function of the probabilities of the two types of sources. Clearly, if the probability of encountering an unknown source is zero (as would be the case in a multisource task), (35) reduces to $\mathrm{P}(\mathrm{Error}) = \mathrm{P}(\mathrm{Error} \mid \mathrm{Known\ source})$. Thus, if the Meta-Pi paradigm scales well to large multisource classification tasks, the acuity/generality tradeoff introduced in Section II-A and detailed in the Appendix assures that it will yield better discrimination than a monolithic classifier. This is because the acuity/generality tradeoff shows that the monolithic classifier's error rate for known sources will increase to the level dictated by the multisource *a posteriori* distributions derived from all the sources used to train the classifier. As the collection of training sources grows large and representative of the ensemble (i.e., all possible sources), the monolithic classifier's error rates for known and unknown sources will converge to the same value. For heterogeneous sources, this value

will be significantly higher than the average source-dependent error rate for the ensemble, as shown in the example of the Appendix.

Using (35), one can show that the necessary condition for a modular classifier to yield (on average) better discrimination than a monolithic classifier is

$$\begin{aligned} &\text{P(Unknown source)} < \\ &\quad \Big[\text{P(Error} \mid \text{Known source)}_{\text{monolithic}} \\ &\quad - \text{P(Error} \mid \text{Known source)}_{\text{modular}}\Big] \\ &\quad \cdot \Big[[\text{P(Error} \mid \text{Unknown source)} \\ &\quad - \text{P(Error} \mid \text{Known source)}]_{\text{modular}} \\ &\quad - [\text{P(Error} \mid \text{Unknown source)} \\ &\quad - \text{P(Error} \mid \text{Known source)}]_{\text{monolithic}}\Big]^{-1}. \end{aligned} \tag{36}$$

Using (36) and the data depicted in Fig. 8, it is straightforward to show that the five-speaker SID and Meta-Pi classifiers—without any adaptation—will yield a lower overall error rate than the five-speaker TDNN for the five known and one novel speaker, as long as the probability that the novel speaker is talking does not exceed approximately 0.15—or about one sixth. If we allow the five-speaker Meta-Pi network to adapt to five examples of each phoneme from the novel speaker, it yields better overall discrimination than the five-speaker TDNN as long as the probability that the novel speaker is talking does not exceed 0.52—a high probability for an unknown source. Presumably, sources with high probabilities are *known*, and unknown (i.e., novel) sources have low probabilities. As a result, we would expect that if the Meta-Pi paradigm scales well to large multisource classification tasks (modeling the known sources is such a task) and maintains its ability to adapt rapidly, it will yield better discrimination than the monolithic classifier by (36).

How many source-dependent modules are necessary for robust source-independent recognition? Peterson and Barney's seminal paper on the formant locations of English vowels serves as a crude indicator of how many sources are necessary for a robust speaker-independent Meta-Pi phoneme classifier. Their speaker population numbered 76, and graphical representations of the first two formant locations for the vowels suggest that one would require $\mathcal{O}$rder this number of sources to build a robust speaker-independent vowel classifier. This estimate is corroborated by the SPHINX speech recognition system; its developers cite 80–100 as the minimum number of training sources necessary for robust speech recognition [13].

Can a large Meta-Pi structure model some novel sources without any adaptation phase?

How can a large Meta-Pi structure be designed so that the combinational superstructure(s) avoid the type of acuity/generality tradeoff outlined in Section VII-B? We think it unlikely that robust speaker-independent recognition of all phonemes can be performed by a single Meta-Pi combinational superstructure operating on $\mathcal{O}$[100] all-phoneme speaker-dependent modules. If the Meta-Pi paradigm does prove to be scalable to such a large task, it is likely to take a more complex hierarchical form.

What quantitative factors should be employed to decide between using long-term and very long-term adaptation to incorporate a particular novel source into the collection of known sources?

How can the computational complexity of the Meta-Pi structure be reduced as it begins to recognize a particular source (i.e., how can the number of modules actually used to model the source be reduced during recognition to minimize computational requirements)?

Does the Meta-Pi architecture lend itself to a particular computer architecture for large scale implementations?

VIII. Conclusion

In this paper, we have presented the Meta-Pi network—a modular connectionist classifier for multisource pattern recognition. We have shown how Meta-Pi learning constitutes a Bayesian MAP training procedure and demonstrated the paradigm on a multisource (six-speaker) and limited source-independent (five known speakers, one unknown speaker) /b,d,g/ phoneme recognition task.

The Meta-Pi classifier yields a source-dependent error rate of 1.9 (+0.5/-0.4)% for the multisource task, whereas a comparable monolithic classifier trained for the same task exhibits an error rate of 4.1 (+0.7/-0.6)% . Beyond this, the Meta-Pi architecture can model one of the six speakers using only the modules of other speakers—an attribute that has possibilities for pattern recognition systems that are both fault tolerant and capable of maintaining their own database of relevant source-dependent models of the input RV.

In the limited source-independent experiment, the Meta-Pi classifier yields a 19.2 (+3.2/-3.1)% error rate on the speech of a novel speaker—five times the 4.1 (+1.7/-1.6)% error rate of a comparable monolithic classifier. However, this Meta-Pi classifier's 2.6 (+/- 0.6)% error rate for five known sources is half the monolithic classifier's 5.5 (+/- 0.7)% error rate for the known sources. Furthermore, the Meta-Pi architecture can adapt to the novel source with five examples of each of the three phonemes, yielding a 6.9 (+2.1/-2.0)% error rate on the novel source that is statistically equivalent to the monolithic classifier's rate. Longer-term adaptation allows the Meta-Pi classifier to model the novel source at source-dependent error rates.

These results indicate that further study of the Meta-Pi paradigm is warranted to determine if it can be a useful component of robust real-world multisource and source-independent pattern classifiers—classifiers that model the source being recognized with an adaptive combination of known source-dependent models.

Appendix
A Formal Statement of the Acuity/Generality Tradeoff

A formal description of the acuity/generality tradeoff is best begun with a review of Bayesian discrimination. Simply stated, the Bayesian discriminant function associates the random

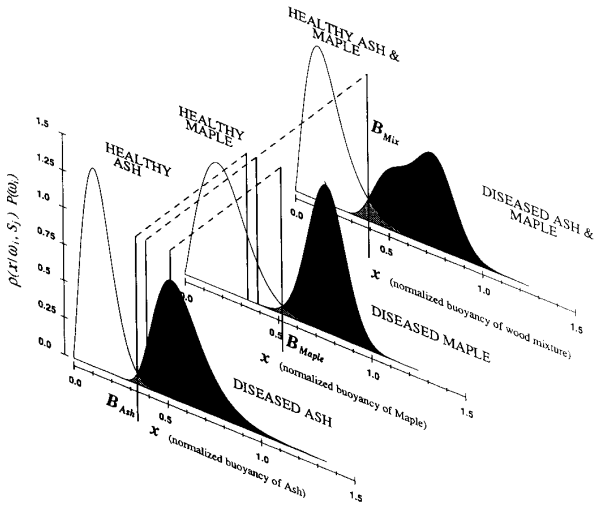


Fig. 10. PDF's for the buoyancy of ash, maple, and a mixture of the two.

feature vector (RV) $\boldsymbol{x}$ with the "best"—by which we mean the most likely—class ω_b.

Consider a two-class real-valued scalar random variable problem for which the class-conditional densities $\rho(x\,|\,\omega_i)$ are unimodal, and there exists only one optimal boundary between the two classes. We assume that all values of x to the left of the class boundary on the real number line belong to class one (ω_1), and all values of x to the right of the class boundary belong to class two (ω_2). Fig. 10 illustrates three such problems; the density associated with ω_1 is shown in white, the density associated with ω_2 is shown in dark gray, and the region on x, where the two densities overlap, is shown light gray.

Given this discrimination task, one can determine the probability of making a classification error, given some value of the class boundary x_B:

$$\mathrm{P}(\mathrm{Error}|x_B) = \int_{x_B}^{\infty} \rho(x|\omega_1)\,\mathrm{P}(\omega_1)\,dx + \int_{-\infty}^{x_B} \rho(x|\omega_2)\,\mathrm{P}(\omega_2)\,dx. \quad (37)$$

One can use Leibnitz's rule to differentiate (37) with respect to x_B in order to determine the value of x_B that yields the *minimum* error rate. This minimum error rate is that yielded by the Bayesian class boundary B, which is achieved when

$$B = x \quad \text{s.t.} \quad \rho(x|\omega_1)\,\mathrm{P}(\omega_1) = \rho(x|\omega_2)\,\mathrm{P}(\omega_2). \quad (38)$$

For the C-class problem wherein $\boldsymbol{x}$ is a random vector with dimensionality N (i.e., $\boldsymbol{x} \in \Re^N$), the result of (38) can be generalized. The Bayesian class boundaries for the ith class in such a problem are described by a set of N-dimensional hypersurfaces[21] for which

$$\rho(\boldsymbol{x}|\omega_i)\,\mathrm{P}(\omega_i) = \sup_{j \neq i} \rho(\boldsymbol{x}|\omega_j)\,\mathrm{P}(\omega_j) \quad (39)$$

[21] The ith set of such hypersurfaces may have no members, a finite number of members, or an infinite number of members. For the case in which the set is empty, the Bayes classifier will never associate $\boldsymbol{x}$ with the ith class.

where *sup* denotes the "supremum" operator. When $N = 2$, these "hypersurfaces" are curves on $\boldsymbol{x}$; when $N = 3$, they are surfaces, etc.

Inside each hypersurface in the ith set satisfying (39)

$$\rho(\boldsymbol{x}|\omega_i)\,\mathrm{P}(\omega_i) > \rho(\boldsymbol{x}|\omega_j)\,\mathrm{P}(\omega_j) \quad \forall\, j \neq i \quad (40)$$

and the Bayes classifier associates $\boldsymbol{x}$ with the ith class ω_i. Any classifier forming class boundaries that deviate from those described above will yield an error rate that is higher than that of the Bayesian discriminant function. A classifier that forms class boundaries as described above is said to implement the Bayesian discriminant function. Some form of these results can be found in all reference texts on pattern classification (e.g., Section 2.5 of [5]).

Now, let us apply these results to the case in which $\boldsymbol{x}$ is generated by any one of a number of statistically heterogeneous sources. To simplify the analysis, we will revert to an example of the simple two-class scalar-input problem describe above. Consider for a moment a hypothetical classification task in which a lumber mill must distinguish between healthy and diseased wood,[22] based on the buoyancy of logs being floated downriver from the point of harvest. Healthy logs have relatively low buoyancy (x), whereas diseased logs (e.g., those affected by wood-boring insects) have relatively high buoyancy. Using appropriate probabilistic models, the mill foreman computes a profile of healthy and diseased samples of the hardwoods ash and maple, as shown in Fig. 10 (front and middle images, respectively). The foreman notices that B in (38) is different for each species of wood. He recognizes that each species of wood represents a *source* with unique statistical properties. He therefore denotes source with the variable $\boldsymbol{S}$ and expresses the probability of misclassifying samples from a particular source $\boldsymbol{S}_k$, given a class boundary x_B, as

$$\mathrm{P}(\mathrm{Error}|x_B, \boldsymbol{S}_k) = \int_{x_B}^{\infty} \rho(x|\omega_1, \boldsymbol{S}_k)\,\mathrm{P}(\omega_1)\,dx + \int_{-\infty}^{x_B} \rho(x|\omega_2, \boldsymbol{S}_k)\,\mathrm{P}(\omega_2)\,dx. \quad (41)$$

Thus, Bayes error rate for the jth source is yielded by the Bayesian class boundary B_k:

$$B_k = x \quad \text{s.t.} \quad \rho(x|\omega_1, \boldsymbol{S}_k)\,\mathrm{P}(\omega_1) = \rho(x|\omega_2, \boldsymbol{S}_k)\,\mathrm{P}(\omega_2). \quad (42)$$

If the foreman classifies ash with the maple threshold (B_{maple}) or maple with the ash threshold (B_{ash}), he finds that his percentage of misclassifications rises sharply:

$$\mathrm{P}(\mathrm{Error}|B_{\mathrm{maple}}, S_{\mathrm{ash}}) \gg \mathrm{P}(\mathrm{Error}|B_{\mathrm{ash}}, S_{\mathrm{ash}}) \quad (43)$$

$$\mathrm{P}(\mathrm{Error}|B_{\mathrm{ash}}, S_{\mathrm{maple}}) \gg \mathrm{P}(\mathrm{Error}|B_{\mathrm{maple}}, S_{\mathrm{maple}}). \quad (44)$$

This is illustrated in Fig. 10, where B_{maple} is superimposed on the ash model (front image); the dark gray area to the left of B_{maple} represents the increase in error rate when the maple model is used to classify ash. Likewise, the white area to the right of B_{ash} (middle image) represents the increase in error rate when the ash model is used to classify maple.

[22] A variation on the theme by Duda and Hart (see Section 2.1 of [5].

$$\begin{bmatrix} \mathrm{P}(\mathrm{Error}|B_{\mathrm{maple}}, S_{\mathrm{ash}}) \\ + \\ \mathrm{P}(\mathrm{Error}|B_{\mathrm{ash}}, S_{\mathrm{maple}}) \end{bmatrix} \gg \begin{bmatrix} \mathrm{P}(\mathrm{Error}|B_{\mathrm{mix}}, S_{\mathrm{ash}}) \\ + \\ \mathrm{P}(\mathrm{Error}|B_{\mathrm{mix}}, S_{\mathrm{maple}}) \end{bmatrix} > \begin{bmatrix} \mathrm{P}(\mathrm{Error}|B_{\mathrm{ash}}, S_{\mathrm{ash}}) \\ + \\ \mathrm{P}(\mathrm{Error}|B_{\mathrm{maple}}, S_{\mathrm{maple}}) \end{bmatrix} \quad (45)$$

(worst case) (best case).

The foreman decides to build a single probabilistic model of buoyancy for ash *and* maple, hoping that this model will be adequate for both species of wood. The resulting ash/maple *mixture densities* are shown in the back image of Fig. 10. When the foreman classifies ash and maple with this mixture density-based threshold (B_{mix} is superimposed on the ash and maple models in Fig. 10), he finds that his percentage of misclassifications is lower than when he uses the wrong species-dependent model to classify a log. However, for both species of wood, the mixture density-based threshold still yields more misclassifications than the species-dependent thresholds originally computed (note the position of B_{mix} relative to the source-dependent optimal boundaries on the ash and maple models; see (45) at the top of this page). Indeed, the foreman realizes that there is a tradeoff between the *acuity* of a single classifier (i.e., its ability to discriminate between healthy and diseased wood with accuracy) and its *generality* (i.e., its applicability to a wide variety of wood species). In short, he can do reasonably well classifying both species with one model or very well classifying each species with its own model, but he cannot do very well classifying both species with a single model. Thus, the only way to achieve optimal discrimination between diseased and healthy wood across all species is to first determine the species of the log and then apply the appropriate species-dependent classification model.

Our hypothetical mill foreman has discovered an optimal strategy for classifying random samples obtained from multiple heterogeneous sources (species of wood, in this example). An understanding of this acuity/generality tradeoff is directly applicable to speech recognition and other pattern recognition processes in which the probabilistic nature of the feature vector is source dependent. However, there is an important caveat: It is not always possible or practical to model each possible source of the input RV. In such cases, we attempt to *approximate* Bayesian discrimination by using some method of *adapting* our classifier to the statistical properties of the (unknown) source that generates $\boldsymbol{x}$.

X. Acknowledgment

A number of people have made suggestions that have improved this paper: Bellcore's D. Kahn and C. Kamm; Siemens' S. Hanson; CMU's B. V. K. Vijaya Kumar, D. Pomerleau, and R. Stern; Yale's B. Pearlmutter; and the anonymous reviewers—we thank them all. We also thank CMU's Warp/iWarp[23] group for their support of our computational requirements.

[23] iWarp is a registered trademark of Intel Corporation.

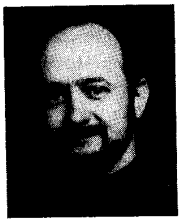
References

[1] A. Barron, "Statistical learning networks: A unifying view," presented at the Symp. Interface: Stat. Comput. Sci., 1988.
[2] H. Bourlard and C. Wellekens, "Links between Markov models and multilayer perceptrons," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 12, no. 12, pp. 1167–1178, Dec. 1990.
[3] D. E. Rumelhart *et al.*, *Parallel Distributed Processing*. Cambridge, MA: MIT Press, 1987.
[4] L. Devroye, "Automatic pattern recognition: A study of the probability of error," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 10, no. 4, pp. 530–543, July 1988.
[5] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 1973.
[6] H. Gish, "A probabilistic approach to the understanding and training of neural network classifiers," in *Proc. 1990 IEEE Int. Conf. Acoustics Speech Signal Processing*, Apr. 1990, pp. 1361–1364, vol. 3.
[7] J. B. Hampshire, II, B. A. Pearlmutter, and B. V. K. Vijaya Kumar, "General equivalence proofs for supervised pattern classifiers and the Bayesian discriminant function," to be published; this is a revised and extended version of "Equivalence proofs for multi-layer perceptron clasifiers and the Bayesian discriminant function," in *Proc. 1990 Connectionist Models Summer School*, 1991, pp. 159–172.
[8] J. B. Hampshire, II and A. H. Waibel, "The Meta-Pi network: Building distributed knowledge representations for robust pattern recognition, Tech. Rep. CMU-CS-89-166-R, Carnegie Mellon Univ., School Comput. Sci., Aug. 1989.
[9] ——, "A novel objective function for improved phoneme recognition using time-delay neural networks, *IEEE Trans. Neural Networks*, vol. 1, no. 2, pp. 216–228, June 1990; this is a revised and extended version of work first presented at the 1989 Int. Joint Conf. Neural Networks, pp. 235–241, vol. I.
[10] V. Hasselblad, "Estimation of parameters for a mixture of normal distributions," *Technometrics*, vol. 8, pp. 431–444, 1966.
[11] W. H. Highleyman, "The design and analysis of pattern recognition experiments," *Bell Syst. Tech. J.*, vol. 41, pp. 723–744, Mar. 1962.
[12] G. E. Hinton, "Connectionist learning procedures," in *Machine Learning: Paradigms and Methods* (J. G. Carbonell, Ed.). Cambridge, MA: MIT Press, 1990, pp. 185–234.
[13] H. W. Hon, personal communication regarding the SPHINX speech recognition system, Apr. 1990.
[14] X. D. Huang, K. F. Lee, and H. W. Hon, "On semicontinuous hidden Markov modeling," in *Proc. IEEE 1990 Conf. Acoustics Speech Signal Processing*, 1990, vol. S_2 VA, pp. 689–692.
[15] R. Jacobs, "Initial experiments on constructing domains of expertise and hierarchies in connectionist systems," in *Proc. 1988 Connectionist Models Summer School* (San Mateo, CA), 1988, pp. 144–153.
[16] R. Jacobs, "Task decomposition through competition in a modular connectionist architecture," Tech. Rep. COINS-90-27, Univ. of Mass., Dept. of Comput. Inform. Sci., Mar. 1990.
[17] R. Jacobs, M. Jordan, S. Nowlan, and G. Hinton, "Adaptive mixtures of local experts," *Neural Computation*, vol. 3, no. 1, Jan. 1991.
[18] B. Kämmerer and W. Küper, "Design of hierarchical perceptron structures and their application to the task of isolated word recognition," in *IEEE Proc. 1989 Int. Joint Conf. Neural Networks* (Washington, DC), June 1989, pp. 243–249.
[19] K. J. Lang, "A time-delay neural network architecture for speech recognition," Ph.D. thesis, Carnegie Mellon Univ., School of Comput. Sci., July 1989.
[20] K. F. Lee, "Large vocabulary speaker-independent continuous speech recognition: The SPHINX system," Tech. Rep. CMU-CS-88-148, Carnegie Mellon Univ., School of Comput. Sci., Apr. 1988.
[21] H. C. Leung and V. W. Zue, "Applications of error back-propagation to phonetic classification," in *Advances in Neural Information Processing Systems, vol. 1* (D. S. Touretzky, Ed.). San Mateo, CA: Morgan Kauffman, 1989, pp. 206–214.

[22] R. Lippmann, "Pattern classification using neural networks," *IEEE Commun. Mag.*, vol. 27, no. 11, 1989.
[23] P. Maloney and D. Specht, "The use of probabilistic neural networks to improve solution times for hull-to-emitter correlation problems," in *Proc. IEEE 1989 Int. Conf. Neural Networks*, June 1989, pp. 289–294, vol. 1.
[24] T. Matsuoka, H. Hamada, and R. Nakatsu, "Syllable recognition using integrated neural networks," in *Proc. IEEE 1989 Int. Joint Conf. Neural Networks*, June 1989, pp. 251–258, vol. 1.
[25] S. J. Nowlan, "Competing experts: An experimental investigation of associative mixture models," Tech. Rep. CRG-TR-90-5, Univ. of Toronto, Dept. of Comput. Sci., Sept. 1990.
[26] ——, "Maximum likelihood competitive learning," in *Advances in Neural Information Processing Systems, vol. 2* (D. S. Touretzky, Ed.). San Mateo, CA: Morgan Kauffman, 1990, pp. 574–582.
[27] ——, "Soft competitive adaptation: Neural network learning algorithms based on fitting statistical mixtures," Ph.D. thesis CMU-CS-91-126, Carnegie Mellon Univ., Pittsburgh, PA, Apr. 1991.
[28] D. O'Shaugnessy, *Speech Communication: Human and Machine*. Reading, MA: Addison-Wesley, 1987.
[29] J. Pollack, "Cascaded back-propagation on dynamic connectionist networks," in *Proc. Ninth Ann. Conf. Cognitive Sci. Soc.*, 1987, pp. 391–404, 1987.
[30] D. Pomerleau, G. Gusciora, D. Touretzky, and H. T. Kung, "Neural network simulation at WARP speed: How we got 17 million connections per second," in *Proc. IEEE 1988 Int. Conf. Neural Networks*, July 1988, pp. 143–150.
[31] D. A. Pomerleau, "The meta-generalized delta rule: A new algorithm for learning in connectionist networks," Tech. Rep. CMU-CS-87-165, Carnegie Mellon Univ., School of Comput. Sci., Sept. 1987.
[32] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, pp. 257–286, Feb. 1989.
[33] R. A. Redner and H. F. Walker, "Mixture densities, maximum likelihood and the EM algorithm," *SIAM Rev.*, vol. 26, pp. 195–239, 1984.
[34] D. Rtischev, "Speaker adaptation in a large vocabulary speech recognition system," Master's thesis, Mass. Inst. of Technol., Jan. 1989.
[35] D. W. Ruck, S. K. Rogers, M. Kabrinsky, M. E. Oxley, and B. W. Sutter, "The multilayer perceptron as an approximation to a Bayes optimal discriminant function," *IEEE Trans. Neural Networks*, vol. 1, no. 4, pp. 296–298, Dec. 1990.
[36] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by backpropagation errors," *Nature*, vol. 323, pp. 533–536, Oct. 1986.
[37] P. A. Shoemaker, "A Note on least-squares learning procedures and classification by neural network models," *IEEE Trans. Neural Networks*, vol. 2, no. 1, pp. 158–160, Jan. 1991.
[38] D. Specht, "Probabilistic neural networks for classification, mapping, and associative memory, in *Proc. IEEE 1988 Int. Conf. Neural Networks*,July 1988, pp. 525–532, vol. 1
[39] R. M. Stern and M. J. Lasry, "Dynamic Speaker Adaptation for Feature-Based Isolated Word Recognition," *IEEE Trans. Acoustics Speech Signal Processing*, vol. ASSP-35, pp. 751–763, June 1987.
[40] A. Waibel, T. Hanazawa, G. Hinton, K. Shikano, and K. Lang, "Phoneme recognition using time-delay neural networks," *IEEE Trans. Acoustics Speech Signal Processing*, vol. ASSP-37, pp. 328–339, Mar. 1989.
[41] A. Waibel, H. Sawai, and K. Shikano, "Modularity and scaling in large phonemic neural networks," *IEEE Trans. Acoustics Speech Signal Processing*, vol. ASSP-37, pp. 1888–1898, Dec. 1989.
[42] E. A. Wan, "Neural network classification: A Bayesian interpretation," *IEEE Trans. Neural Networks*, vol. 1, no. 4, pp. 303–305, Dec. 1990.
[43] ——, "Temporal backpropagation: An efficient algorithm for finite impulse response neural networks," in *Proc. 1990 Connectionist Models Summer School* (San Mateo, CA), 1991, pp. 159–172.
[44] R. Watrous, "Context-modulated discrimination of similar vowels using second-order connectionist networks," Tech. Rep. CRG-TR-89-5, Univ. of Toronto, Dept. of Comput. Sci., Dec. 1989.
[45] H. White, "Learning in artificial neural networks: A statistical perspective, *Neural Computation*, vol. 1, no. 4, pp. 425–464, Winter 1989.

John B. Hampshire, II (S'86) was born in Natick, MA, in July 1958. He received the B.S.E.E. degree from the U.S. Naval Academy, Annapolis, MD, in 1980 and the M.S. degree in electrical and biomedical engineering sciences from Thayer School of Engineering, Dartmouth College, Hanover, NH, in 1988.

Following commisioning into the Navy, he spent three years at sea two years as an admiral's executive assistant in Washington, DC. He resigned his commission in 1986 and returned to school. He is currently completing his Ph.D. dissertation at Carnegie Mellon University's Department of Electrical and Computer Engineeering. His Ph.D. thesis research is entitled "A Differential Theory of Learning for Statistical Pattern Recognition with Connectionist Models." His general research interests are in connectionist approaches to statistical pattern recognition, detection and estimation, information theory, and computational learning theory.



Alex Waibel (M'86) received the B.S. degree from the Massachusetts Institute of Technology, Cambridge, in 1979 and the M.S. degree in electrical engineering and computer science and Ph.D. degree in computer science from Carnegie Mellon University, Pittsburgh, PA, in 1980 and 1986, respectively.

From 1980 to 1985, he was a member of the Computer Science Research Staff at Carnegie Mellon University. In 1986, he joined the faculty of the Computer Science Department as a Research Associate and in 1988 as a Research Computer Scientist. Since 1991, he has been a Senior Research Computer Scientist at Carnegie Mellon and University Professor of Informatik at Karlsruhe University, Karlsruhe, Germany. At Carnegie Mellon, he holds joint faculty appointments at the Center for Machine Translation and the Computational Linguistics Program. His research interests include speech recognition and synthesis, neurocomputing, machine learning, and machine translation. From May 1987 to July 1988, he has worked as Invited Research Scientist at the ATR Interpreting Telephony Research Laboratories, Osaka, Japan, where he initiated research in neural computation and its application to speech translation. He now directs a large research effort in speech translation and neural computation at Carnegie Mellon and at Karlsruhe University. He has published and lectured extensively in the area of speech processing and neural computation. He served as consultant to several corporations and research labs in several countries.

Dr. Waibel has chaired a number of conferences and workshops, served on various program committees, and has given many invited talks and tutorials. He has served on several government steering committees in the United States and Germany. He has served as a referee on a number of journals, conferences, and granting agencies and is currently a member of the Technical Committee of the IEEE Accoustics, Speech, and Signal Processing Society. He is a member of the IEEE Signal Processing and Computer Societies, the Computer Society of America, the Association for Computational Linguistics, and the International Neural Network Society. In 1979, he was awarded the MIT Guillemin Award. His 1989 paper on Time-Delay Neural Networks was awarded the IEEE Signal Processing Society's Senior Best Paper Award in 1991 and the ATR Paper Award in 1990.