Detection and Identification of Rare Audiovisual Cues

Insperata accident magis saepe quam quae speres.
(Things you do not expect happen more often than
things you do expect)   Plautus (ca 200(B.C.)

Project no: 027787

DIRAC

Detection and Identification of Rare Audio-visual Cues

Integrated Project
IST - Priority 2

DELIVERABLE NO: D5.2
Combination of Nonlinear Classifier

Date of deliverable: 31.12.2007
Actual submission date: 23.01.2007

Start date of project: 01.01.2006 Duration: 60 months

Organization name of lead contractor for this deliverable: ex: **IDIAP Research Institute**

Revision [1]

Project co-funded by the European Commission within the Sixth Framework Program (2002-2006)		
Dissemination Level		
PU	Public	X
PP	Restricted to other program participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

DIRAC
Detection and Identification of Rare Audiovisual Cues

Insperata accident magis saepe quam quae speres.
(Things you do not expect happen more often than
things you do expect)   Plautus (ca 200(B.C.)

D5.2 Combination of Nonlinear Classifier

:

IDIAP Research Institute (IDIAP)

Abstract:

This work explores different methods for combining outputs from nonlinear classifiers that estimate posterior probabilities of context independent phonemes and are used for data driven feature extraction. The classifier combination techniques are evaluated on Automatic Speech Recognition (ASR). Input to the classifiers are spetro-temporal features that emphasize long term temporal and short term spectral aspects of the speech signal. Firs, some classical combination rules are investigated. Further, we investigate combination of neural net based classifiers using Dempster-Shafer Theory of Evidence. Under some assumptions, combination rule resembles a product of errors rule observed in human speech perception. Different combination are tested in ASR experiments both in matched and mismatched conditions and compared with more conventional probability combination rules. Proposed classifier combination techniques are particularly effective in mismatched conditions. Finally, we study a hierarchy of Neural Network classifiers used for data driven feature extraction. Two different hierarchical structures based on long and short temporal context are considered. Features are tested in a LVCSR task on meetings data and compared with classical speech features.

Table of Content

1 Introduction

1.1 Artificial Neural Networks as Posterior Probability Estimators and as Generators of Features for the HMM-based ASR

Neural network – based classifiers can be trained in such a way that their output estimate class posterior probabilities [12]. Generally, a three-layer network structure is used even though other topologies were also investigated. The output from the third layer is processed through a softmax static nonlinearity so that the outputs sum to one. In the automatic recognition of speech (ASR), basic speech sounds classes are phonemes of the language and the neural network can be trained to estimate their posterior probability distributions for any given time instant in the input speech signal. In that way, the speech signal can be turned into a two-dimensional representation that we call a *posteriogram* that represents a sequence of distributions of posterior probabilities. An example of the posteriogram is shown in Fig. 1. Ideally, a well trained neural network will activate at any given time that element of the posterior probability distribution, which corresponds to the phoneme represented by the input feature pattern. To use neural network estimated features in a conventional hidden markov model based ASR, the vector space of estimated posterior probability distributions can be gaussianized (e.g. through the logarithmic static nonlinearity) and subsequently decorrelated using a principal component transform, derived from some training data [13].

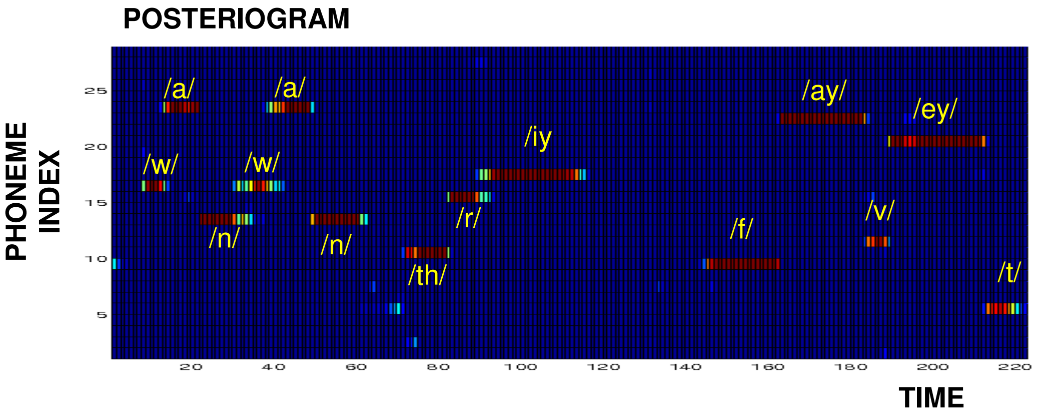


Fig. 1 An example of a posteriogram for the digit sentence "one-one-three-five-eight".

1.2 Multi-stream approaches to ASR

Multi-stream speech recognition approaches where individual information streams are formed by using evidence from different elements of the signal are becoming a norm in the ASR community (e.g. multi-band [1, 2], feature combinations [3], classifier combinations [17, 18, 19]). In this work, we study combinations of posterior probabilities of phonemes derived from different input speech representations. The probabilities are estimated by a multi-layer perceptron (MLP) trained on phoneme-labeled data. In literature, many papers have already addressed the problem (e.g. [4]) considering combination rules like sum, product, maximum and minimum rules. The major factors that influence the performance are the diversity in the classifier team and also the method employed for combining [20,21]. The diversity in the classifier team depends on the amount of complementary information present in the individual classifiers [10, 17, 18, 19, 1, 2, 20]. As evident, feature extraction methods inspired by auditory perception capture better relevant information in speech and hence results in improved performance [21]. Further, if we have different speech processing methods inspired by different aspects of auditory perception, then features extracted from them may exhibit complementary information.

1.3 The outline of the report

The present work is performed in the framework of automatic recognition of speech (ASR), in particular, on digit recognition task, using OGI-Stories and OGI-Numbers95 databases, and on a meeting task that contains much larger amounts of data. The databases are described in the Section 2.1. below. Different methods will then be explored for combining estimated posterior probabilities o and using them in ASR.

ASR deals with recognizing spoken message from input speech [22]. Since speech is redundant in nature, different speech processing methods have been developed for extracting relevant features [22, 23]. Among these, Perceptual Linear Prediction (PLP) and Mel-Frequency Cepstral Coefficients (MFCC), both based on auditory perception are the most commonly used ones [23, 24]. Most recently Multi-RASTA (MRASTA), also based on auditory perception, has been proposed and known to perform better compared to PLP [14]. PLP extracts features by analyzing speech over short segments of 10-30 ms [24]. Alternatively, MRASTA based features are obtained by analyzing speech over segments of 1 sec [14]. That is, PLP represents speech spectral band energy information over short segments of 10-30 ms and MRASTA represents temporal information over long segments of 1 sec. Thus MRASTA and PLP seem to exhibit different aspects of speech and hence may contain complementary information.

Further, we study a combination rule based on Dempster-Shafer theory of evidence ([5]) which can be considered an extension of Bayesian probability. Main advantage of this framework is the explicit representation of ignorance. DS theory has already been investigated in speech recognition (e.g. [6]) but this is probably the first attempt to use it for combination of information coming from different acoustic streams. Furthermore, under some assumption, DS combination rule is similar to what is known in the speech recognition community as the Fletcher's "product of errors" (see [7],[8]).

2. Overview of Databases and Relevant Techniques Applied in the Current Study

2.1 Speech Databases

Two speech corpora were considered for the present work, namely, OGI-Stories and OGI-Numbers95 [25, 26]. Both contain speech recorded over a telephone channel in similar recording conditions. OGI Stories contains spontaneous continuous speech with rather large vocabulary, OGI-Numbers95 contains strings of digits and numbers. The task involves recognition of digits from zero to nine as well as oh. This vocabulary of 11 words composes of 29 phonemes [14]. Three distinct data sets were created from these corpora:

MLP-Training Set: This set contains 208 files from OGI-Stories (about 2.8 hrs) and 2547 files from OGI-Numbers95 containing strings of 11 digits from zero to nine plus oh (about 1.3 hrs) transcribed on phoneme level by hand. This set will be used for training Multi Layer Perceptron (MLP) for estimating phoneme posterior probabilities, more commonly termed as posteriors.

HMM-Training Set: This set contains 2547 files from OGI-Numbers95 containing strings of 11digits (same 2547 files as used earlier). This set will be used for generating posteriors for training HMM based ASR.

HMM-Testing Set: This set contains 2169 files from OGI-Number95 (different from those used in the earlier two sets). This set will be used for generating posteriors for testing HMM based ASR.

Meetings Database is used in some experiments with hierarchical MLP. It consists in around 100 hours of meetings recorded on different sites (AMI, ICSI, NIST and ISL). Channel used is the independent headset microphone. Phoneme set is composed of 46 targets including silence. Training data are phonetically labeled using forced alignment by an LVCSR system.

2.2 PLP Features

The PLP analysis of speech involves the following steps [24]: (i) Convolving short term power spectrum of speech with a simulated critical-band masking pattern, (ii) resampling the critical-band spectrum at approximately 1-Bark intervals, (iii) pre-emphasis by a simulated fixed equal loudness curve, (iv) compression of resampled and preemphasized spectrum through the cubic-root non-linearity and (v) approximating the compressed spectrum by the all-pole model. In the present work, the all-pole model coefficients are further transformed into Cepstral Coefficients (CC). Speech signals are processed in frames of 25 ms with a shift of 10 ms. For each frame of 25 ms, 13 PLPCC, 13 _PLPCC and 13 __PLPCC are extracted by PLP analysis. Thus effectively for every 10 ms of speech a 39 dimension PLP based feature vector is computed.

2.3 MRASTA Features

Critical-band auditory spectrum is extracted for every frame of 25 ms with a shift of 10 ms. By filtering temporal trajectories of each critical band with a bank of N fixed length low pass Finite Impulse Response (FIR) filters representing Gaussian functions of several different widths and by subsequent computing of first and second differentials of the smoothed trajectories will yield a set of N × 2 modified spectra at every frame (Gaussian Features) [14]. The same filter bank is used for all bands. A bank of 16 filters consisting of first and second order derivatives of Gaussian functions is applied to all 15 temporal trajectories of critical-band spectral energies at all frequencies resulting in 240 (16 × 15) features per frame [14]. All temporal filters are zero-phase FIR filters that is, they are centered around the frame being processed. Length of all filters is fixed at 101 frames, corresponding to 1000 ms of signal and each frame for every 10 ms duration. The first frequency derivatives of the stream are derived by applying an FIR filter to output of each of the 16 filters, across frequencies [14]. Derivatives for the first and last critical bands are not defined and hence gives a feature set of size 208 (16×13) features per frame. The feature vector is formed by appending the first order frequency derivatives to the main feature stream, resulting in 448 features per frame.

2.4 TANDEM-based ASR

Phoneme posteriors estimated from MLP have been used in ASR for a number of years [12]. Hermansky et. al. [13] proposed a way of converting these posteriors to features appropriate for conventional HMM recognizers. This hierarchical classification technique of combining various information sources in deriving features for conventional HMM-based recognizers came to be known as TANDEM feature extraction. In TANDEM-based ASR, first the speech signals are processed using suitable signal processing methods like PLP or MRASTA to extract features for training MLP. MLP of suitable structure is then trained using extracted features as input vectors and vectors representing corresponding phoneme labels as targets.

The phoneme posteriors will then be estimated from the trained MLP. The estimated posteriors will be non-Gaussian in nature and to use them in HMM framework for ASR, they will be gaussianized by TANDEM operation which involves non-linear (e.g. log or inverse softmax function) operation followed by Principal Component Analysis (PCA) on the posteriors. The gaussianized and decorrelated posteriors from the training set will be used for generating HMM models. The same operation (with PCA basis derived on the training set) is done on the test set. The gaussianized and decorrelated posteriors from the test set will then be used for evaluating the performance of trained HMM models.

2.5 The Dempster-Shafer Theory of Evidence

The Dempster-Shafer (DS) Theory of Evidence (see [5]) allows representation and combination of different measures of evidence. It can be considered as a generalization of the Bayesian framework and permits the characterization of uncertainty and ignorance.

Let $\Theta = \{\theta_1, ..., \theta_K\}$ be a finite set of mutually exclusive and exhaustive hypotheses refereed as singletons. Θ is referred as *frame of discernment*. Let 2^Θ be the power set of Θ, i.e. the set of all subsets of Θ. A basic probability assignment (BPA) is a function m from 2^Θ to [0, 1] such that

$$m : 2^\Theta \rightarrow [0,1], \quad \sum_{A\subset\Theta} m(A) = 1 \text{ and } m(\oslash) = 0 \qquad (1)$$

$m(A)$ can be interpreted as the amount of belief that is assigned exactly to A and not to any of its subsets. In probability theory, a measure is assigned only to atomic hypothesis $m(\theta_i)$ while in DS Theory it can be assigned to a set A without any further commitment on the on the atomic hypothesis that compose A. The situation of total ignorance is represented by $m(\theta_t) = 1$. On the other hand, if we set $m(\theta_t\ )$for all θ_t and $m(A) = 0$ for all $A \neq \theta_t$ we recover the probability theory.

Let A be complementary set of A i.e. the set $\{\Theta - A\}$. In DS Theory, m(A) + m(A) < 1 (contrarily to probability theory), which means that we can consider an amount of belief that is not attributed to an hypothesis nor to its negation. In other words, “we don’t need to over-commit when we are ignorant”.

The function that assigns to each subset A, the sum of all basic probability numbers of its subset is called *belief function* or *credibility* of

Subset A for which $m(A) > 0$ are called focal elements and their union is called core. A belief function is defined as vacuous if it has only Θ as focal element. A belief function is defined as *simple support* function if it has only one focal element in addition to Θ and Bayesian if its focal elements are singleton.

In an analogous way, *Plausibility* of an hypothesis A is defined as:

$$Pl(A) = 1 - Bel(\bar{A}) = \sum_{B\cap A\neq 0} m(B) \qquad (3)$$

and it measures to what extent we fail to doubt in A. Another interesting point in DS Theory is how two different belief functions Bel_1 and Bel_2 over the same frame of discernment are combined into a single belief function. Dempster’s rule states that Bel_1 and Bel_2 must be

$$\textstyle\sum_{A\cap B=\oslash} m_1(A)m_2B < 1.$$

combinable i.e. their cores must not be disjoint. Given m_1 and m_2 BPAs associated with Bel1 and Bel2 this condition can be expressed as

. In this case m_1 and m_2 can be combined as:

$$m(\oslash) = 0, \;\; m(\theta) = \frac{\sum_{A \cap B = \theta} m_1(A) m_2(B)}{1 - \sum_{A \cap B = \oslash} m_1(A) m_2(B)} \tag{4}$$

and $m(\theta)$ is a BPA. The belief function given by m is called orthogonal sum of Bel_1 and Bel_2 denoted as $Bel_1 \oplus Bel_2$ (m as well is denoted as $m_1 \oplus m_2$). DS orthogonal sum is both associative and commutative. Given two belief functions Bel_1 and Bel_2, if Bel_1 is vacuous, then $Bel_1 \oplus Bel_2 = Bel_2$; if Bel_1 is Bayesian, then $Bel_1 \oplus Bel_2$ is also Bayesian. Let us consider now the case of orthogonal sum between two simple support belief functions Bel_1 and Bel_2 with focus
$A \neq \Theta$ i.e. $m_1(A) = s_1$, $m_1(\Theta) = 1 - s_1$, $m_2(A) = s_2$, $m_2(\Theta) = 1 - s_2$.

Applying DS orthogonal sum (4), we obtain:

$$m(\Theta) = (1 - s_1)(1 - s_2), \;\; m(A) = 1 - (1 - s_1)(1 - s_2) \tag{5}$$

In words, in case of simple support belief functions, the total ignorance is the product of ignorances of single belief. In next section, we draw a parallel with product of errors.

2.5.1. Product of Errors

Work of Fletcher ([7]) on human processing of speech suggests that humans process speech in different frequency sub-bands independently. Combination of processing from each sub-band is done in such a way that total error is equal to product of errors in different sub-bands. In other words, to recognize correctly a phoneme it is enough to recognize it correctly in one of the available sub-bands.

Those findings suggested as possible combination rule of classifiers based on different acoustic evidence, the product of errors (PoE). Let us denote with p1 and p2 the probability of correct recognition of a phoneme for two different acoustic streams, according to PoE, the combined probability of those classifiers should be $p = 1-(1-p1)(1-p2)$. It is evident the analogy in between previous expression and results from expression (5) with the difference that in theory of evidence we should talk about "product of ignorance" rather then "product of errors". Anyway, as we will verify in the experimental section, combination according to PoE does not provide results comparable to classical classifiers combination rules; on the other hand, "product of ignorances" gives good results compared to other rules.

2.5.2 From MLP output to Basic Probability Assignment

DS theory represents an interesting alternative to classical probability framework for combining different classifiers and it has already been largely studied in the machine learning community (e.g. see [9]). Main weakness of DS theory is the fact that results are strongly sensitive on the choice of the Basic Probability Function. Thus DS combination rule has a certain degree of heuristic depending on the type of classifier we aims at combining.

We will focus on combination of outputs from different Neural Networks. In [10] and [11], multiple neural nets outputs are combined using DS orthogonal sum for handwriting recognition applications. The main question is how to choose an effective BPA. Each output from the neural net is considered as a source of information (a belief) that induces a frame of discernment. If we denote with θ_i the $i - th$ output of the MLP, focal elements of the corresponding BPA will be $m_i(\theta_i)$ i.e. the belief we have in the hypothesis associated with the $i - th$ output, $m_i(\Theta_i)$ i.e. the belief we have in the complementary of this hypothesis and $m_i(\theta)$ i.e. the ignorance associated with this hypothesis. In [10], BPA are estimated respectively according to recognition rate, error rate and rejection rate of each Neural Net output while in [11], they are estimated according to different kind of distances between MLP outputs and some reference vectors.

We consider the output of a Neural Network trained in order to estimate posterior distributions for a target class (i.e. a phoneme posterior) [12]. Let us consider a phoneme set $\Theta = \{\theta_1, ..., \theta_k\}$ and a trained Neural Net that produces target posteriors $\{p_1 = p(\theta_1|X), ..., p_k = p(\theta_k|X)\}$ with

$$\sum_i p_i = 1$$

where X is an observation vector. First problem we have to deal with is how to transform the probabilistic output of the MLP into a BPA. With DS formalism, the probabilistic output can be represented by the following BPA

$$m(\theta_i) = p_i \ \forall i \text{ and } m(\Theta) = 0$$

i.e. all belief is attributed to atomic hypotheses (phonemes) and no belief to the ignorance. To quantify the degree of ignorance of the MLP output, a natural choice is the use of the entropy of the output

$$H = \sum_i^k p_i log(p_i).$$

Ignorance is supposed to be total (i.e. $m(\Theta) = 1$) when entropy of the output achieves its maximum value

$$H = \sum_i^k p_i log(p_i).$$

Under those considerations a possible choice for a BPA is represented by:

$$\begin{aligned} m_i(\theta_i) &= \alpha p_i \quad m_i(\Theta) = 1 - \alpha p_i = 1 - m_i(\theta_i) && (6) \\ \text{with } \alpha &= (1 - \frac{H}{H_{max}})^\gamma && (7) \end{aligned}$$

When the entropy H is zero, ignorance $m_i(\Theta)$ is equal to $1 - p_i$ while when entropy is maximum ignorance $m_i(\Theta) = 1$. Choice of function (7) is heuristic; exponent factor γ is supposed to better fit ignorance estimation to entropy measure because ignorance should may not be a linear function of the entropy. BPAs as defined in (6) are simple support functions and we refer to them as BPA1.

Anyway other BPAs can be defined in which we further add information on the complementary set Θ_i. For instance we could define a new BPA as

$$m_i(\theta_i) = \alpha p_i \qquad m_i(\neg\theta_i) = \alpha(\sum_{j\neq i} p_j) \tag{8}$$

$$m_i(\Theta) = 1 - m_i(\theta_i) - m_i(\neg\theta_i) \tag{9}$$

In this case each MLP output is supposed to provide information on both phoneme i and set of phonemes *Θ − i.* Contrarily to probability theory, they do not sum to one because a certain amount of belief is supposed to be assigned to all phoneme set *Θ*. We refer to BPAs (8-9) as BPA2.

Finally a third set of BPA can be directly derived from orthogonal sum of BPAs (6). In fact BPA from each MLP output as defined in (6) are combinable; applying $(\oplus_i m_i)$ orthogonal sum (4) a new set of BPA can be directly obtained

$$m(\theta_i) = m_i(\theta_i) \prod_{j\neq i}(1 - m_j(\theta_j))/Z \tag{10}$$

$$m(\neg\theta_i) = (1 - m_i(\theta_i)) \prod_{j\neq i}(1 - m_j(\theta_j))/Z \tag{11}$$

$$m(\Theta) = \prod_{j}(1 - m_j(\theta_j))/Z \tag{12}$$

$$Z = 1 - m_i(\theta_i)(1 - \prod_{i\neq j}(1 - m_j(\theta_j)) \tag{13}$$

We refer to set of BPAs (10-13) as BPA3. In this section, we described three different ways of associating a basic probability assignment on a frame of discernment induced by a MLP output. In next section we describe how to combine two different BPAs obtained trough two different Neural Networks.

2.5.3 DS Theory for Classifier Combination

Let us consider now the case in which we have two different Neural Networks and their corresponding BPA obtained in one of the three ways described in previous section. Those BPA can now be combined applying orthogonal sum (4). In case of simple support functions (i.e. BPA1), we must combine BPA with only one focal element. Given two MLP *a* and *b* and correspondent BPA $m_a(\theta_i) = s_a$, $m_a(\Theta) = 1 - s_a$,$m_b(\theta_i) = s_b$, $m_b(\Theta) = 1 - s_b$, orthogonal sum gives:

$$\begin{aligned} m(\Theta) &= m_a(\Theta)m_b(\Theta) = (1 - s_a)(1 - s_b) & (14)\\ m(\theta_i) &= m_a(\theta_i)m_b(\theta_i) + m_a(\theta_i)m_a(\Theta) + m_b(\theta_i)m_a(\Theta) & \\ &= 1 - (1 - s_a)(1 - s_b) & (15) \end{aligned}$$

Similarity of expressions (14 - 15) with product of errors rule are quite obvious with the difference that in this case combination rule consider product of "ignorance" instead of errors. In case of BPA2 and BPA3, combination rule must handle as well the set $m_i(\Theta_i)$; orthogonal sum gives:

$$\begin{aligned} m(\theta_i) &= \{m_a(\theta_i)m_b(\theta_i) + m_a(\theta_i)m_a(\Theta) + \\ &+ m_b(\theta_i)m_a(\Theta)\}/Z && (16) \\ m(\neg\theta_i) &= \{m_a(\Theta)m_b(\neg\theta_i) + m_b(\Theta)m_a(\neg\theta_i)\}/Z && (17) \\ m(\Theta) &= \{m_a(\Theta)m_b(\Theta)\}/Z && (18) \\ Z &= 1 - m_a(\neg\theta_i)m_b(\theta_i) - m_b(\neg\theta_i)m_a(\theta_i) && (19) \end{aligned}$$

Combination rules (14 - 15) and (16 - 19) show how to combine BPA from two different MLP into a single BPA. Those rules can be easily extended to more then two classifiers because they are associative.

2.6 Hierarchical Neural Networks

Neural Network architectures are an active field of study and several hierarchical structures have been proposed for applications in ASR [29, 30]. The hierarchy we study is based on the work reported in [30] and consists of the architecture where a cascade of MLPs is used and the second MLP uses as its input the MLP-derived features (i.e. after a gaussianizing and a PCA rotation) from the first (previously trained) MLP together with the original spectro-temporal pattern that was used for estimating the MLP-derived features. The basic hierarchical architecture is shown in Fig. 1.

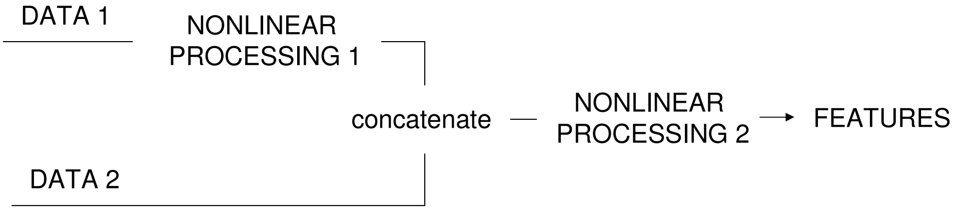


Fig. 1 Hierarchical classifier combination for extraction of data-driven features for speech recognition. Both datasets, the DATA 1 and the DATA 2, come from some (not necessarily the same) vicinity of the time instant where the posterior density is derived. The NONLINEAR MAPPING blocks represents different MLPs with their associated post-processing.

The intuition behind such an architecture is that the first net yields a particular output when it is activated by a particular input. For some more ambiguous inputs, the output might be in error. These errors may be between two or three competing phoneme classes. The second net (which is trained on both the output from the first net and on the pattern that activated the first net), may be able to correct the errors generated by these ambiguous inputs.

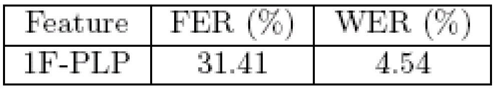
3 Experiments

3.1 ASR using PLP Features

3.1.1. ASR using Single Frame PLP (1F-PLP) Features

PLP features are extracted from MLP-Training Set. These features are used for training an MLP with 39 units in input layer, 10600 units in hidden layer and 29 units in output layer. The number of units in the input layer corresponds to the dimensionality of input feature vectors and the number of units in the output layer corresponds to the number of phonemes. The number of units in the hidden layer is mainly driven by the amount of information that need to be captured from the input features, and the computational resources available for training. With performance on a cross-validation set as a stopping criterion for training, more the number of units (up to a certain value, to be determined experimentally), better may be the performance, but more will be the need for computational resources. The PLP features from HMM-Training Set are applied to the trained MLP to estimate corresponding phoneme posteriors. The estimated posteriors are gaussianized as explained earlier and then used as features for HMM based ASR. One HMM model is trained for each of the phonemes. The HMM models were tri-phone models trained in context independent mode. Also, each model was represented by 32 Gaussian mixture components [14]. The gaussianized phoneme posteriors from HMM-Testing Set obtained in a similar fashion as that of HMM-Training Set are then applied to the trained HMM models to evaluate the performance. FER was defined on MLP posteriors as a ratio of frames with maximum posterior matching the underlying class label over the overall number of frames, considering MLP cross-validation set. In the MLP-Training Set approximately 10% of data were used for cross-validation. The FER obtained is given in Table 1. WER is defined as WER = (S+I+D)/N, where, I, S, D, N are counts of substitutions, insertions, deletions and all recognized words, respectively. The WER obtained for the HMM-Testing Set is given Table 1.

Table 1: Recognition performance of ASR using single frame PLP features.

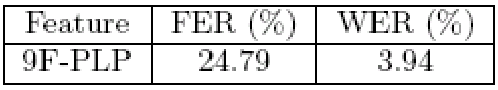


Feature	FER (%)	WER (%)
1F-PLP	31.41	4.54

3.1.2. ASR using Nine Frames of PLP (9F-PLP) Features

Instead of single frame PLP features, contiguous nine frames of PLP features have also been used as feature vectors for ASR. This leads to feature vectors of dimension 9 × 39 = 351 [12]. Such a combination has been shown to give better performance compared to single frame PLP features. An MLP is trained with 351 units in input layer, 1800 units in hidden layer and 29 units in output layer. The PLP features from HMM-Training Set are then applied to the trained MLP to estimate the corresponding phoneme posteriors. The estimated posteriors are gaussianized and then applied as features for HMM based ASR. The gaussianized phoneme posteriors from HMM-Testing Set are applied to the trained HMM models to evaluate the performance. The FER obtained for the cross validation data is given in Table 2. The WER obtained for HMM-Testing Set is also given in Table 2. Using contiguous nine frames of PLP features, the performance improves about 21% relative in case of FER and 13% in case of WER over the ASR using single frame PLP features.

Table 2: Recognition performance of ASR using nine frames of PLP features.



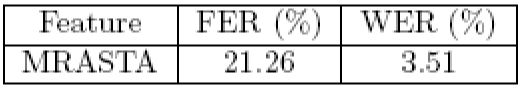
Feature	FER (%)	WER (%)
9F-PLP	24.79	3.94

3.2 ASR using MRASTA Features

3.2.1. ASR using MRASTA Features

MRASTA feature vectors are extracted from MLP-Training Set. These features are used for training an MLP with 448 units in input layer, 1800 units in hidden layer and 29 units in output layer. The MRASTA features from HMM-Training Set are applied to the trained MLP to estimate the corresponding phoneme posteriors. The estimated posteriors are gaussianized and then used as features for HMM based ASR system. The gaussianized phoneme posteriors from HMM-Testing Set are then applied to trained HMM models. The FER obtained for the cross validation data is given in Table 3. The WER obtained for the HMM-Testing Set is also given in Table 3. The performance is significantly better compared to single frame PLP features (please compare Tables 3 and 1). The performance is even better compared to nine frames of PLP features (please compare Tables 3 and 2). The ASR using MRASTA provides a relative improvement of about 32% in case of FER and 22% in case of WER compared to ASR using single frame PLP features.

Table 3: Recognition performance of ASR using MRASTA features.



Feature	FER (%)	WER (%)
MRASTA	21.26	3.51

Combining Posteriors from MRASTA and PLP

As discussed earlier, MRASTA and PLP features are different representations of speech and hence makes sense to combine the two. Some of the combination methods like product, sum, minimum and maximum are the obvious choice due to their simplicity [3p, 17p, 18p]. Therefore we explore the significance of each of these combination methods for combining posteriors from MRASTA and PLP.

3.2.2. Product of Posteriors

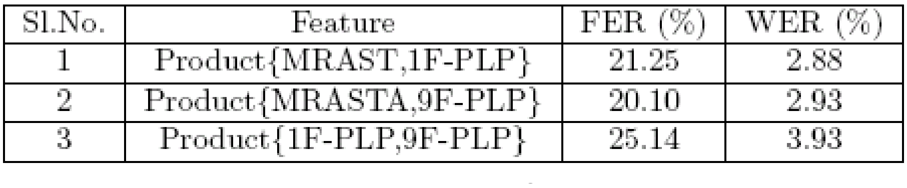
For a given frame of speech we have 29 phoneme posteriors derived using PLP and 29 phoneme posteriors derived using MRASTA. In product case, for each of the 29 phonemes, corresponding posteriors from MRASTA and PLP are multiplied and are normalized with respect to the total sum of multiplied posterior values.

Let P_{Mi}, $P_{P}i$ and P^{prod}_{Ci} , where i = 1, 2, . . . , 29 represent phoneme posteriors for MRASTA, PLP and product cases, respectively. The product of MRASTA and PLP posteriors is implemented as

$$P^{prod}_{Ci} = \frac{P_{Mi}P_{Pi}}{\sum_i P_{Mi}P_{Pi}} \qquad i = 1, 2, \ldots, 29$$

We are calling the resulting values as product posteriors. Since the product posteriors are also non-Gaussian in nature, they are processed by TANDEM feature extraction to make them Gaussian. The gaussianized product posteriors from HMM-Training Set are used for training the HMM models. The trained HMM models are tested using the gaussianized product posteriors from HMM-Testing Set. The FER and WER obtained for product posteriors in different combinations of MRASTA and PLP are given in Table 4.

Table 4: Recognition performance of ASR using product posteriors.



Sl.No.	Feature	FER (%)	WER (%)
1	Product{MRAST,1F-PLP}	21.25	2.88
2	Product{MRASTA,9F-PLP}	20.10	2.93
3	Product{1F-PLP,9F-PLP}	25.14	3.93

As it can be observed from Table 4, the performance of combined system using product posteriors is significantly better in terms of WER than MRASTA alone. Over MRASTA, the combined system using MRASTA and single frame PLP features shows a relative improvement of 18% in terms of WER. Even though the classifier using single frame PLP features gives poor performance on its own (Table 1), it combines well with MRASTA to significantly improve the performance. The near equal performance obtained by combining MRASTA with either single frame PLP or nine frames PLP infers that, both single frame and nine frames PLP features have same amount of complementary information. When single frame and nine frames PLP features are combined, it hardly gives any improvement. This fact emphasizes the need for careful selection of feature sets, which have complementary information, before combining them.

3.2.3. Sum of Posteriors

In sum case, for each of the 29 phonemes, corresponding posteriors from MRASTA and PLP are added and are normalized with respect to the total sum of added posterior values

Let P_{M}i, P_{Pi} and P_{sum}^{Ci} , where i = 1, 2, . . . , 29 represent phoneme posteriors for MRASTA, PLP and sum cases, respectively. The sum of MRASTA and PLP features is implemented as

$$P_{Ci}^{sum} = \frac{P_{Mi} + P_{Pi}}{\sum_i P_{Mi} + P_{Pi}} \qquad i = 1, 2, \ldots, 29$$

We call the resulting posteriors the sum posteriors. The sum posteriors are processed in a similar way as explained in the case of product posteriors. The ASR study is conducted using the gaussianized sum posteriors. FER obtained for the cross-validation data and WER obtained for HMM-Testing Set are given in Table 5.

Table 5: Recognition performance of ASR using sum posteriors.

Sl.No.	Feature	FER (%)	WER (%)
1	Sum{MRASTA,1F-PLP}	21.07	3.20
2	Sum{MRASTA,9F-PLP}	20.23	3.17
3	Sum{1F-PLP,9F-PLP}	24.96	3.99

In this case also, same trend as in the case of product posteriors is observed. However, the improvement is less than that using product posteriors. This fact emphasizes the complementary information present in MRASTA and PLP and is being best exploited by taking product [7p].

3.2.4. Minimum of Posteriors

In minimum case, for each of the 29 phonemes, minimum of corresponding posteriors from MRASTA and PLP is taken and are normalized with respect to the total sum of minimum posterior values. Let P_{Mi}, P_{P}i and P_{min}^{Ci} , where i = 1, 2, . . . , 29 represent phoneme posteriors for MRASTA, PLP and min cases, respectively.

The minimum of MRASTA and PLP features is implemented as

$$P_{Ci}^{min} = \frac{\min\{P_{Mi}, P_{Pi}\}}{\sum_i \min\{P_{Mi}, P_{Pi}\}} \qquad i = 1, 2, \ldots, 29$$

We call these posteriors the min posteriors. The performance of ASR system using min posteriors processed in a similar fashion as that of product and sum posteriors is given in Table 6

Table 6: Recognition performance of ASR using min posteriors.

Sl.No.	Feature	FER (%)	WER (%)
1	Min{MRASTA,1F-PLP}	21.85	3.02
2	Min{MRASTA,9F-PLP}	20.22	3.02
3	Min{1F-PLP,9F-PLP}	25.54	3.92

Same trend as in the earlier cases of product and sum posteriors is observed in this case also. However, the improvement is less than that obtained using product posteriors.

3.2.5. *Maximum of Posteriors*

In the maximum case, for each of the 29 phonemes, the maximum of corresponding posteriors from MRASTA and PLP is taken and are normalized with respect to the total sum of maximum posterior values.

Let P_{Mi}, P_{Pi} and P_{max}^{Ci} , where i = 1, 2, . . . , 29 represent phoneme posteriors for MRASTA, PLP and max cases, respectively. The maximum of MRASTA and PLP features is implemented as

$$P_{Ci}^{max} = \frac{\max\{P_{Mi}, P_{Pi}\}}{\sum_i \max\{P_{Mi}, P_{Pi}\}} \qquad i = 1, 2, \ldots, 29$$

We call these posteriors from eqn.(4) max posteriors. The performance of ASR system using max posteriors processed in a similar fashion as that of other posteriors is given in Table 7.

Table 7. Recognition performance of ASR system using max posteriors.

Sl.No.	Feature	FER (%)	WER (%)
1	Max{MRASTA,1F-PLP}	21.40	3.34
2	Max{MRASTA,9F-PLP}	20.49	3.24
3	Max{1F-PLP,9F-PLP}	25.14	3.79

Same trend as in the earlier cases of product, sum and min posteriors is observed in this case also. However, the performance of combined system is less than that obtained using product posteriors.

3.3 ASR On Noisy Channel

To emulate a stationary channel mismatch between training and testing data, we applied first order preemphasis filter with $\alpha = 0.97$ to HMM-Testing Set [14]. Such distorted test data were passed through existing systems and word error rates were evaluated. The performance of different systems is tabulated in Table 8.

Table 8: Recognition performance of ASR for preemphasized data from HMM-Testing Set.

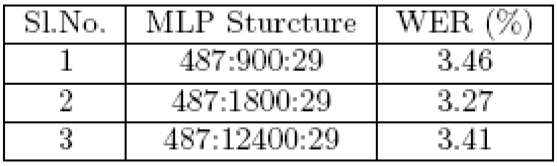
Sl.No.	Feature	WER (%)
1	Single frame PLP	15.61
2	Nine frames PLP	10.00
3	MRASTA	3.50
4	Product{MRASTA,1F-PLP}	3.80
5	Sum{MRASTA,1F-PLP}	4.98
6	Min{MRASTA,1F-PLP}	4.21
7	Max{MRASTA,1F-PLP}	5.54

Both single frame and nine frames PLP features based systems are very sensitive to these distortions, where as the system based on MRASTA features is quite resistant. The performance for the best performing combination scheme, that is, MRASTA and single frame PLP are also shown in the table. The product combination is relatively more resistant to degradation compared to others. That is, the effect of degradation is felt minimum in the combined system using product.

3.3.1. Combining MRASTA and PLP Features

Instead of combining posteriors from different classifiers trained using MRASTA and PLP, as discussed earlier, we also experimented combining MRASTA and PLP features and training one classifier for the combined features. For each frame of speech we have 39 PLP and 448 MRASTA features and padding them will give a feature vector of dimension 487. Initially an MLP with 487 units in input layer, 900 units in hidden layer and 29 units in output layer was trained using the combined features and the performance is given in Table 9. The performance is poor compared to that using combining posteriors. Later we tried with different number of hidden units and the performance for each case is given in Table 9. Even by increasing the number of units in hidden layer to a large value (12400), equivalent to sum of the units used in MLP trained using MRASTA and MLP trained using PLP, the performance is poor. This infers that instead of combining features and training a single classifier, it may be better to train separate classifiers for each of the features and combine their posteriors.

Table 9: Recognition performance of ASR for single MLP classifier trained using combined features.



Sl.No.	MLP Sturcture	WER (%)
1	487:900:29	3.46
2	487:1800:29	3.27
3	487:12400:29	3.41

3.4 ASR Experiments with Combinations Based on DS Theory of Evidence

It the following experiments, a slightly different techniques for feature extraction were used so the absolute numbers do not exactly correspond to the numerical results in the previous section. However, the experiments with the individual TANDEM and MRASTA techniques, as well as the experiments with product and sum of posteriors are repeated and the new numbers are give so that he reader can directly compare the improvements obtained using the Theory of Evidence.

We investigate the use of DS theory of evidence for combining output of Neural Networks in data- driven feature extractions for ASR. Results are compared with classical combination rules like product and sum.

Data driven feature extraction methods aims at estimating directly from data, features that are used in the recognition process. An effective and well established technique consists in estimating phoneme posteriors using a Neural Network (see [13]). Phoneme posteriors are further processed trough a logarithmic function and a Principal Component Analysis Transform (PCA) before using them as features in the classical HMM/ GMM framework.

Database we used for recognition experiments is the OGI-Numbers 95 while MLP is trained using 3 hours of hand-labeled speech from the OGI-Stories database. Phoneme set is constituted by 29 English phonemes. Two different posterior streams are considered: TANDEM-PLP posterior ([13]) and Multi-RASTA posterior ([14]).

In case of TANDEM-PLP posteriors, MLP input is a vector of 9 consecutive frames of PLP features. In case of Multi-RASTA posteriors, MLP input is a segment of one second critical band energies filtered through a set of multi resolution filters. Those two streams are supposed to capture short and long term dependencies in two different features set. We will consider combination of those different streams according to DS theory.

Multi-RASTA features are inherently robust to linear distortion of the signal [14]. On the other side, Tandem-PLP features are seriously affected by this distortion. To verify the effectiveness of the combination techniques, we study performances of combination when a first order preemphasis filter with $\alpha = 0.95$ is applied to the test data set.

Table 10 reports TANDEM-PLP and Multi-RASTA performances in terms of WER in case of matched and mismatched conditions. While Multi-RASTA features hold the performance even in mismatched conditions, TANDEM-PLP are seriously affected.

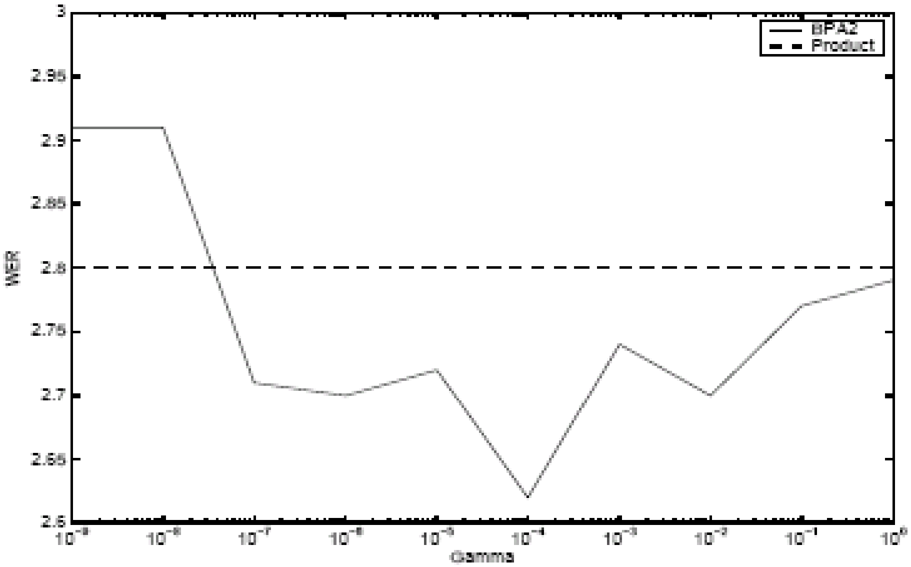
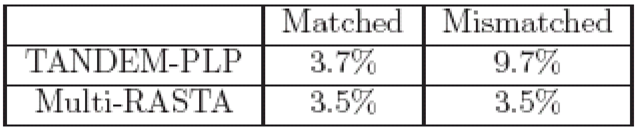


Figure 2: Performance of combination rule BPA2 function of the factor γ in matched conditions.

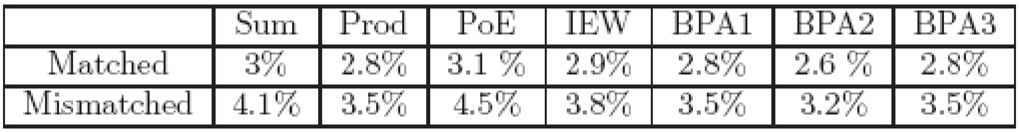
Table 10: WER for TANDEM-PLP and Multi-RASTA features in matched and mismatched conditions.



	Matched	Mismatched
TANDEM-PLP	3.7%	9.7%
Multi-RASTA	3.5%	3.5%

In the following, we study different combination rules for the two posterior stream. Combined posterior are converted into features using a logarithmic transform and then a KLT transform. Classical way of combining posterior are the sum rule and the product rule (e.g. [4],[15]). We also consider the product of errors rule, directly applied on posterior estimation and inverse entropy weighting (IEW)[16]. In addition to those, we consider combination trough DS theory. When DS theory of evidence is applied, posterior distributions are first transformed into BPA using rules BPA1, BPA2 and BPA3 as described in section 4. BPA from different posterior streams are then combined together using rules described in section 5: BPA1 is combined using rules (14 - 15) (for simple support functions) while BPA2 and BPA3 are combined using rules (16 - 19).

Table 11: WER for different combination rules in matched and mismatched conditions. Sum, Prod (product),PoE (product of errors), IEW (inverse entropy weighting).



	Sum	Prod	PoE	IEW	BPA1	BPA2	BPA3
Matched	3%	2.8%	3.1 %	2.9%	2.8%	2.6 %	2.8%
Mismatched	4.1%	3.5%	4.5%	3.8%	3.5%	3.2%	3.5%

Table 11 shows Word Error Rates for different combination techniques in matched and mismatched conditions. In clean conditions combination of posteriors gives always better results than each posterior stream independently.

Out of the combination rules based on traditional probability theory, product holds the best performance, while product of errors gives the higher error rate. In mismatched conditions, product rule gives same performance of the best feature stream, while sum and product of errors give inferior results.

Let us now consider results from DS combination rules. Out of the three proposed combination framework, the best performing BPA2 is giving 7% improvement in matched conditions and 9% improvement in mismatched conditions w.r.t. product rule.

BPA1 and BPA3 performances are similar to those obtained using product rule. Combination rules BPA1 and BPA3 give very similar results indicating that merging evidence from different outputs of the same MLP does not give any improvement in our experiments.

Many other approaches for combining MLP outputs according to entropy measures have been considered in the past (e.g. [16],[15]). Combination rules still are product rule or sum rule but they are weighted according to some functions of the entropy. In our approach entropy is used to determine the amount of belief from a given MLP output that must be discarded i.e. assigned to the ignorance hypothesis. DS orthogonal sum 4 in the general case cannot be re-conducted into any of those rules.

The most questionable part is the way we transform the output of a probabilistic classifier (i.e. a MLP) into Basic Probability Assignment. Our choices are somehow heuristic and must be further investigated. The use of the entropy is a natural way of representing ignorance but there is no reason for supposing that ignorance should be a linear function of the entropy. As solution to this problem, we choose the function (7) with a correction factor γ. This factor has actually an impact on the final performance of the combination. Figure 1 plots WER in matched conditions as a function of γ for BPA2. WER are sensitive to the value of γ and there are some intervals in which DS combination performs consistently better than sum or product rules.

3.5 ASR Experiments With Hierarchical Neural Nets

For hierarchical neural nets, we felt that the experiments on relatively small OGI digit data (which contains together only about three hours of speech) may not be appropriate since the number of free parameters in the net hierarchies is larger than in the single nets investigated earlier. Therefore we have switched to larger (about 100 hours of speech) database of speech from meetings, also described earlier in the report.

We trained hierarchical NN up to a level of hierarchy of 3. Table 12 report Frame Error Rate for single NN, two NN and three NN models. Input is 9 frames PLP features augmented with delta and delta-delta.

Table 12 Frame error rates computed considering all the speech and the silence frames, and only the speech frames while considering the classification correct if the output indicating the correct phoneme was the highest, among two highest and among top three highest outputs

	With silence	Without silence	2 best without silence	3 best without silence
Single MLP	34.6 %	43.0 %	32.3 %	26.3 %
Two stage hierarchy	29.2 %	35.9 %	27.5 %	22.7 %
Three stage hierarchy	27.9 %	33.0 %	25.5 %	21.1 %

In Table 12, we report frame error rates for the single MLP, and for hierarchy of two and tree MLPs. The overall frame error rate is reduced by 5.5 % absolute when the cascade of two MLP is used and by additional 2\% absolute for the cascade of three MLPs. It is also interesting to notice that the difference between the frame error rate of a single output and of three best outputs is progressively reduced. Detailed analysis of results(not shown here)shows that there is no frame error increase for any phoneme.

An interesting effect of such a hierarchy is that at each layer the acoustic context is progressively increased: if the first MLP has a temporal context of 9 frames, the second MLP will in effect use a temporal context of 9+8 frames and the third one a temporal context of 9+8+8. In next section we investigate the use of a hierarchy of NN directly using directly an acoustic context of one second.

Finally, Table 13 shows word error rates on large vocabulary continuous speech experiments obtained using PLP, Tandem, MRASTA and hierarchical features. Single net TANDEM and MRASTA features do not outperforms classical PLP front-end. There is a consistent drop in performance in VT data which are particularly noisy when NN features are used. When a second NN is used with TANDEM features an average improvement of 2.2 % absolute is obtained; this improvement is verified on all type of data. On the other hand, when a third NN is added, overall performance deteriorates by 0.6 %. This may be an effect due to the over-fitting of the net on a finite size training data.

Table 13 Word error rates in percent for several feature sets, including the hierarchical nonlinear classifier based features

	average	AMI	CMU	ICSI	NIST	VT
PLP	**42.4**	42.8	40.5	31.9	51.1	46.8
TANDEM	**46.6**	41.4	43.7	31.3	54.5	64.9
Two stage hierarchy	**44.4**	39.6	42.3	28.9	51.5	62.9
Three stage hierarchy	**45.0**	40.5	44.4	29.4	51.1	61.9
MRASTA	**45.9**	48.0	41.9	37.1	54.4	48.8
2 stage hierarchy with MRASTA	**39.4**	38.1	36.9	28.2	48.0	46.9

MRASTA features are designed to remove mean value in the modulation spectrum through the use of a multi-resolution band-pass filters on the modulation spectrum and are thus more robust to noise and distortions. Furthermore they use an acoustic context of one second. Overall performance of MRASTA is slightly better than Tandem features. It is interesting to notice the performance on VT data, where contrarily to Tandem they hold performance comparable to PLP. On the other hand Hierarchical MRASTA show an average improvement of 6 % over the single net MRASTA and 3 % improvement over PLP features.

4 Summary and Conclusions

The combination system using MRASTA and PLP gives a significant improvement over MRASTA alone. This infers that MRASTA and PLP carry complementary information. Among different combination methods examined, product of posteriors seems to give maximum benefit. If there is any degradation due to channel, then also the combined system using product offers more immunity compared to others. Finally, the poor performance of the single classifier trained using combined MRASTA and PLP features infers that it is better to train independent classifers on MRASTA and PLP and combine their posteriors.

In this work we have demonstrated by conducting different ASR experiments that MRASTA and PLP have complementary information and hence combine well to give improved performance. Several improvements are possible over this work. We can further subdivide MRASTA and PLP feature sets depending on some criterion, train classifiers for each of the subsets and then try to combine them.

Further, we attempted combining output from different neural networks based on Dempster-Shafer Theory of Evidence. Main appeal of this theory is the possibility of representing ignorance. Under certain assumptions, DS combination rule show analogies with what was found by Fletcher in his speech perception experiments.

Three different rules for transforming MLP outputs into belief are presented. DS combination rule is tested in recognition experiments and compared with classical combination rules (sum, product and product of errors) both in matched and mismatched conditions. In matched conditions, all combination rules outperforms individual feature streams. Best combination rule is BPA2 while PoE is the worst one. On the other side, product of "ignorance" (i.e. BPA1) shows similar results as the product rule. In mismatched conditions, we would like to have at least a performance equal to the performance of the best feature stream. In case of product rule and BPA1 this is achieved. BPA2 is still achieving error rate lower than the one provided by the best feature stream meaning that it is able to extract useful information from both streams. Sum and PoE rules are giving error rate higher than those achieved by the best feature stream.

5 References

[1] Bourlard H. and Dupont S., "A new asr approach based on independent processing and re-combination of partial frequency bands.," Proc. ICSLP 96.

[2] Hermansky H., Tibrewala S., and Pavel M., "Towards asr on partially corrupted speech," Proc. ICSLP 1996.

[3] Janin A., Ellis D., and Morgan N., "Multi-stream speech recognition: Ready for prime time," Proc Eurospeech-1999.

[4] Kirchhoff K. and Bilmes J., "Combination and joint training of acoustic classifiers for speech recognition." in ISCA ITRW Conference on Automatic Speech Recognition, Paris, 2000.

[5] Shafer G., A mathematical theory of evidence., Princeton, MIT Press, 1976.

[6] Kobayashi T., "An application of dempster and shafer's probability theory to speech recognition," The Journal of the Acoustical Society of America., vol. 100 (4), October 1996.

[7] Fletcher H., Speech and Hearing in Communication., Krieger, Hew York, 1953.

[8] Allen J.B., Articulation and Intelligibility, Morgan and Claypool, 2005.

[9] Mandler E.J. and Schurman J., "Combining the classification results of independent classifiers based on dempster/shafer theory of evidence.," Pattern Recognition and Artificial Intelligence, vol. X, pp. 381–393, 1988.

[10] Xu L., Kryzak A., and Suen C.Y., "Methods of combining multiple classifiers and their applications to handwriting recognition.," IEEE transactions on Systems, Man and Cybernetics, vol. 22(3), pp. 418–435, 1992.

[11] Galina L. R., "Combining the results of several neural network classifiers.," Neural Networks, vol. 7(5), pp. 777–781, 1994.

[12] Bourlard H. and Morgan N., Connectionist Speech Recognition - A Hybrid Approach., Kluwer Academic Publishers, 1994.

[13] Hermansky H., Ellis D., and Sharma S., "Connectionist feature extraction for conventional hmm systems.," Proceedings of ICASSP, 2000.

[14] Hermansky H. and Fousek P., "Multi-resolution rasta filtering for tandem-based asr.," in Proceedings of Interspeech 2005, 2005.

[15] Hagen A., Robust speech recognition based on multi-stream processing, Ph.D. thesis, ´ Ecole Polytechnique F´ed´erale de Lausanne, Lausanne, Switzerland, December 2001.

[16] Misra H., Bourlard H., and Tyagi V., "Entropy-based multi-stream combination," in Proceedings of ICASSP, 2000

[17] T. K. Ho, J. J. Hull, S. N. Srihari, Decision combination in multiple classifier systems, IEEE Trans. Pattern Analysis, Machine Intelligence 16 (1) (1994) 66–75.

[18] J. Kittler, M. Hatef, R. P. W. Duin, J. Matas, On combining classifiers, IEEE Trans. Pattern Analysis, Machine Intelligence 20 (3) (1998) 226–239.

[19] L. I. Kuncheva, A theoritical study on six classifier fusion strategies, IEEE Trans. Pattern Analysis, Machine Intelligence 24 (2) (2002) 281–286. Philadelphia, PA, USA, 1996, pp. 426–429.

[20] Pavel, H. Hermansky, Information fusion by human and machines, in: Proc. First European Conf. Signal Analysis, Prediction, Prague, Czech Republic, 1997, pp. 350–353.

[21] Hermansky, Should recognizers have ears?, Speech Communication 25 (1998) 3–27.

[22] R. Rabiner, B. H. Juang, Fundamentals of Speech Recognition, Prentice-Hall, Englewood Cliffs, New Jersy, 1993.

[23J. Picone, Signal modelling techniques in speech recognition, Proc. IEEE 81 (1993) 1215–1247.

[24] Hermansky, Perceptual linear predictive analysis of speech, J. Acoust. Soc. Amer. 87 (4) (1990) 1738–1752.

[25] R. A. Cole et. al., Telephone speech corpus development at CSLU, in: Proc. Int. Conf. Spoken Language Processing, Yokohama, Japan, 1994, pp. 1815–1818.

[26] R. A. Cole, M. Noel, T. Lander, T. Durham, New telephone speech corpus at CSLU, in: Proc. European Conf. Speech Processing, Technology (EUROSPEECH), Madrid, Spain, 1995, pp. 821– 824.

[27] F. M. Alkoot, J. Kittler, Experimental evaluation of expert fusion strategies, Pattern Recognition Letters 20 (1999) 1361–1369.

[28] F. M. Alkoot, J. Kittler, Modified product fusion, Pattern Recognition Letters 23 (2002) 957–965.

[29] Sivadas S. and Hermansky H., Hierarchical Tandem Feature Extraction, Proceedings of ICASSP-2002.

[30] Schwarz P., Matejka P., Cernocký J., Hierarchical structures of neural networks for phoneme recognition, Proceedings of ICASSP 2006.