Discriminative Product-of-Expert Acoustic Mapping for Cross-lingual Phone Recognition

Khe Chai Sim *

# National University of Singapore, Singapore
Computing 1, 13 Computing Drive, Singapore 117417
sinkc@comp.nus.edu.sg

* Institute for Infocomm Research, Singapore
1 Fusionopolis Way, #21-01 Connexis, Singapore 138632
kcsim@i2r.a-star.edu.sg

Abstract—This paper presents a Product-of-Expert framework to perform probabilistic acoustic mapping for cross-lingual phone recognition. Under this framework, the posterior probabilities of the target HMM states are modelled as the weighted product of experts, where the experts or their weights are modelled as functions of the posterior probabilities of the source HMM states generated by a foreign phone recogniser. Careful choice of these functions leads to the Product-of-Posterior and Posterior Weighted Product-of-Expert models, which can be conveniently represented as 2-layer and 3-layer feed-forward neural networks respectively. Therefore, the commonly used error back-propagation method can be used to discriminatively train the model parameters. Experimental results are presented on the NTIMIT database using the Czech, Hungarian and Russian hybrid NN/HMM recognisers as the foreign phone recognisers to recognise English phones. With only about 15.6 minutes of training data, the best acoustic mapping model achieved 46.00% phone error rate, which is not far behind the 43.55% of training data, the best acoustic mapping model achieved to recognise English phones. With only about 31.31 hours of data.

I. INTRODUCTION

Cross-lingual speech recognition is an important technique which allows a well-trained recognition system of one language to be quickly adapted to perform speech recognition in another language. Cross-lingual speech recognition operates under the condition that there is limited amount of training data available in the target language to reliably build a system from scratch. This is particularly true for the under-resourced languages. Cross-lingual speech recognition techniques are also useful for multilingual speech recognition where well-trained systems can be easily adapted to multiple languages. A speech recognition system comprises two major components: 1) acoustic model and 2) language model. To perform cross-lingual speech recognition, it is necessary to map the acoustic and language models from one language to another. This paper focuses on the problem of acoustic mapping. Some of the previous work on cross-lingual speech recognition can be found in [1], [2], [3].

One simple approach to acoustic mapping is the probabilistic mapping of acoustic symbols. These symbols can be the phones or phone states which are language dependent. For example, work on phone mapping has been investigated in [4], [2]. Recently, Probabilistic Phone Mapping (PPM) models [5], [6] were formulated as discrete Hidden Markov Models (HMMs) to map a foreign phone sequence to a target phone sequence.

Hybrid NN/HMM phone recognisers have been shown to yield superior performance and widely used in multilingual speech processing such as the phonotactic tokenisers for language recognition [7]. The NN/HMM uses feed-forward neural networks to generate HMM state posterior probabilities, which are then used as the HMM state emission probabilities to perform Viterbi decoding. In order to perform cross-lingual phone recognition using the NN/HMM phone recogniser, a Product-of-Expert (PoE) framework is proposed to map the state posterior probabilities generated by a foreign NN/HMM phone recogniser to the state posterior probabilities of the target language. Under this framework, the target state posterior probabilities are expressed as the weighted product of experts, where the experts or the corresponding expert weights are modelled as functions of the state posterior probabilities of the foreign language. Under certain conditions, models that fit within this framework can be represented as a feed-forward neural network using softmax activation. These neural networks take in log posterior probabilities of the source states and produce the posterior probabilities of the target states. This paper presents two probabilistic acoustic mapping models, Product-of-Posterior (PoP) and Posterior Weighted PoP (PWPoP), which fit within the proposed PoE framework.

The remaining of this paper is organised as follows. Section II describes the hybrid NN/HMM phone recognition system. Section III formulates cross-lingual phone recognition as a probabilistic acoustic mapping (PAM) problem. Section IV presents the proposed Product-of-Expert (PoE) framework for PAM and introduces two models that fit into this framework. Experimental results are reported in Section V.

II. HYBRID NN/HMM PHONE RECOGNISER

The phone recognition problem is to convert speech waveform into a corresponding sequence of phones. Typically, a sequence of $T$ observation vectors, $O = \{o_1, o_2, \ldots, o_T\}$, are extracted from the speech waveform. Given a set of phones, $Y$, a phone recogniser finds the best phone sequence, $Y^* = \{y_1, y_2, \ldots, y_N\}$, where $y_i \in Y$ for $1 \leq i \leq N$. Under
the probabilistic framework, this can be formulated as:

\[ Y^* = \arg \max_Y P(Y|O) = \arg \max_Y P(O|Y)P(Y) \tag{1} \]

where \( P(Y|O) \) is the posterior probability of the phone sequence, \( Y \), given the observation sequence, \( O \). By applying the Bayes’ theorem, the posterior probability can be decomposed into the product of likelihood, \( P(O|Y) \) and prior probability, \( P(Y) \). These are commonly known as the acoustic and language model probabilities respectively. The evidence, \( P(O) \), has been dropped from the above equation as it is independent of \( Y \) and therefore does not affect the maximisation problem.

Hidden Markov Models (HMMs) [7] are widely used to model the phone units for speech recognition. Each HMM phone model is typically represented by a 3-state left-to-right topology. The observation probability associated with each HMM state is commonly represented as a Gaussian Mixture Model (GMM) distribution. This form of HMM models are referred to as the GMM/HMM models, where the state observation probability distribution is given by:

\[ p(o_t|j) = \sum_{m=1}^{M} c_{jm} N(\mu_{jm}, \sigma^2_{jm}) \tag{2} \]

where \( p(o_t|j) \) is the likelihood of state \( j \) generating the observation vector \( o_t \). \( N(\cdot) \) denotes the normal distribution function. \( c_{jm}, \mu_{jm} \text{ and } \sigma^2_{jm} \) are respectively the prior, mean and variance of the \( m \)th Gaussian component of state \( j \). These parameters can be estimated efficiently using the Baum-Welch training algorithm to maximise the conventional Maximum Likelihood (ML) estimation criterion [8]. Alternatively, discriminative training paradigms, such as Maximum Mutual Information (MMI) [9] and Minimum Phone Error (MPE) [10], have been found to yield superior performance.

Instead of using discriminative training paradigms to estimate the GMM/HMM model parameters, discriminative classifiers, such as Neural Network (NN), can also be combined with HMM to yield discriminative acoustic models [7]. This yields a hybrid NN/HMM phone recogniser. Given the input feature vector, \( f_t \), at time \( t \), NN is used to generate the HMM state posterior probabilities, \( P(j|f_t) \). Typically, \( f_t \) is obtained by stacking multiple observation vectors around time \( t \), i.e.

\[ f_t = [o_{t-w}^{T} \cdots o_{t}^{T} \cdots o_{t+w}^{T}] \tag{3} \]

\( w \) is the number of preceding and succeeding observation vectors to be included in \( f_t \) to capture longer temporal information. The effective window length is therefore \( 2w + 1 \). For the NN output vectors to satisfy the probability axioms:

\[ P(j|f_t) \geq 0 \quad \text{and} \quad \sum_{j} P(j|f_t) = 1 \]

a softmax activation function is used at the output layer:

\[ v_i = \frac{\exp(u_i)}{\sum_{j} \exp(u_j)} \tag{4} \]

To train the NN, the training sample pairs, \( \{(f_t, \hat{P}(j|t)); 1 \leq t \leq T\} \) are used. In this paper, hard target values are used to train the NN, i.e.,

\[ \hat{P}(j|f_t) = \begin{cases} 1 & \text{if state } j \text{ is aligned to time } t \\ 0 & \text{otherwise} \end{cases} \tag{5} \]

The state alignment information can be obtained using Viterbi forced-alignment using an HMM model.

The hybrid NN/HMM system uses the state posterior probabilities, \( P(j|f_t) \), as the state likelihood probabilities, \( p(o_t|j) \), and perform the standard Viterbi decoding to obtain the output phone sequence. The hybrid NN/HMM phone recognisers have been found to outperform the classic ML trained GMM/HMM systems [7]. This paper also shows that the NN/HMM phone recognisers outperform the MMI trained GMM/HMM systems (c.f. Section V-A).

III. CROSS-LINGUAL PHONE RECOGNITION

Under certain circumstances, the acoustic model that is needed to obtain \( P(O|Y) \) may not be available. This may be due to the lack of training data or the time constraint which prohibits the complete training of the acoustic models from scratch. In such cases, it may be preferable to utilise one or more phone recognisers that are well-trained on other languages to perform the recognition. Such a task is known as a cross-lingual phone recognition task. Formally, a cross-lingual phone recognition problem can be formulated by rewriting Equation (1) as follows:

\[ Y^* = \arg \max_Y P(O|Y)P(Y) \]

\[ = \arg \max_Y \left( \sum_{X} P(O|X)P(X|Y) \right) P(Y) \tag{6} \]

where \( P(O|X) \) is the likelihood of the phone sequence, \( X \), generating the observations, \( O \). \( P(X|Y) \) is the probability of mapping the phone sequence \( Y \) to \( X \). Since \( P(O|X) \) is generated by a foreign phone recogniser, all is left is to estimate the phone sequence mapping probabilities, \( P(X|Y) \), so that a foreign phone recogniser can be used to perform recognition in the target language. To further simplify the problem, Equation (6) can be approximated as:

\[ Y^* \approx \arg \max_Y P(X^*|Y)P(Y) \tag{7} \]

\[ X^* = \arg \max_X P(O|X) \tag{8} \]

The approximation was made by assuming that the summation over \( X \) is dominated by the term \( P(O|X^*) \). Therefore, the cross-lingual phone recognition problem is decoupled into two stages: 1) perform phone recognition using the foreign recogniser to obtain the best foreign phone sequence, \( X^* \), using Equation (8); 2) perform probabilistic phone mapping to convert \( X^* \) to \( Y^* \) using Equation (7). It was reported in [5] that the latter can be formulated as a discrete HMM model. This is known as the Probabilistic Phone Mapping (PPM) model. Spatial and temporal context-sensitive phone mapping can be easily incorporated into the PPM model to
further improve the phone recognition performance [11], [6]. However, the assumption made in Equation (7) suffers from great information loss as speech is ‘quantised’ into phone tokens in the first stage. Solely relying on the PPM models to recover the target phone sequences leads to great performance degradation. Therefore, better mapping models that uses richer acoustic information is needed. In this paper, Probabilistic Acoustic Mapping (PAM) is proposed to extend the PPM models.

The aim of the Probabilistic Acoustic Mapping (PAM) model is to transfer as much acoustic information as possible from the foreign phone recogniser to the target phone recogniser. As previously described in Section II, the essence of a high quality NN/HMM is the discriminatively estimated state posterior probabilities using a feed-forward neural network. Hence, the objective of PAM is to map the state posterior probabilities of one phone recogniser to another. Let \( \mathcal{X} \) and \( \mathcal{Y} \) be the set of states of the source and target phone recognisers respectively. The total number of states in these recognisers are \(|\mathcal{X}| = M\) and \(|\mathcal{Y}| = N\) respectively. The state posterior probabilities at time \( t \) are given by

\[
\begin{align*}
    p_{j}^{(x)}(t) &= P(x_{j}|O, t) \quad \text{for} \quad 1 \leq j \leq M, \\
    p_{j}^{(y)}(t) &= P(y_{j}|O, t) \quad \text{for} \quad 1 \leq j \leq N
\end{align*}
\]

(9)

For convenience, the posterior probability vectors for the source and target phone recognisers are written as

\[
\begin{bmatrix}
    p_{1}^{(x)}(t) \\
    p_{2}^{(x)}(t) \\
    \vdots \\
    p_{M}^{(x)}(t)
\end{bmatrix} \quad \text{and} \quad
\begin{bmatrix}
    p_{1}^{(y)}(t) \\
    p_{2}^{(y)}(t) \\
    \vdots \\
    p_{N}^{(y)}(t)
\end{bmatrix}
\]

Hence, the PAM model can be expressed using the following equation:

\[
p_{j}^{(y)}(t) = \mathcal{F}_{j}(p^{(x)}(t), \lambda)
\]

(10)

where \( \mathcal{F}_{j}(.) \) is the PAM function for \( p_{j}^{(y)}(t) \) with parameters \( \lambda \). In the next section, a Product-of-Expert (PoE) framework will be used to model \( \mathcal{F}_{j}(.) \).

IV. PRODUCT-OF-EXPERT FRAMEWORK

The Product-of-Expert (PoE) framework for the mapping function, \( \mathcal{F}_{j}(.) \), can be expressed as:

\[
p_{j}^{(y)}(t) = \mathcal{F}_{j}(p^{(x)}(t), \lambda) = \frac{1}{Z} \prod_{s=1}^{S} \mathcal{E}_{js}(p^{(x)}(t), \lambda_{js}^{(E)}) \mathcal{H}_{js}(p^{(x)}(t), \lambda_{js}^{(H)})
\]

(11)

where \( \mathcal{E}_{js}(.) \) is the \( s \)th expert for the \( p_{j}^{(y)}(t) \) with parameters \( \lambda_{js}^{(E)} \) and \( \mathcal{H}_{js}(.) \) is the corresponding expert weight function with parameters \( \lambda_{js}^{(H)} \). \( Z \), the normalisation constant that ensures \( \sum_{j=1}^{N} p_{j}^{(y)}(t) = 1 \), is given by

\[
Z = \sum_{j=1}^{N} \prod_{s=1}^{S} \mathcal{E}_{js}(p^{(x)}(t), \lambda_{js}^{(E)}) \mathcal{H}_{js}(p^{(x)}(t), \lambda_{js}^{(H)})
\]

(12)

By letting \( \mathcal{G}_{js}(.) = \log \mathcal{E}_{js}(.) \), Equation (11) can be rewritten in the form of a softmax function as follows:

\[
p_{j}^{(y)}(t) = \frac{\exp \left\{ K_{j}(p^{(x)}(t), \lambda_{j}) \right\}}{\sum_{k=1}^{N} \exp \left\{ K_{k}(p^{(x)}(t), \lambda_{k}) \right\}}
\]

(13)

where

\[
K_{j}(p^{(x)}(t), \lambda_{j}) = \sum_{s=1}^{S} \mathcal{H}_{js}(p^{(x)}(t), \lambda_{js}^{(H)}) \mathcal{G}_{js}(p^{(x)}(t), \lambda_{js}^{(E)})
\]

and \( \lambda_{j} = \{ \lambda_{js}^{(E)}, \lambda_{js}^{(H)} : 1 \leq s \leq S \} \) denotes the set of all the parameters associated with the experts for generating the \( j \)th posterior probability, \( p_{j}^{(y)}(t) \). Note that if either \( \mathcal{G}_{js}(.) \) or \( \mathcal{H}_{js}(.) \) is independent of \( p^{(x)}(t) \), \( K_{j}(.) \) can then be expressed as an inner product of two vectors, one of which is independent of \( p^{(x)}(t) \). In other words, \( K_{j}(.) \) is linear w.r.t. \( p^{(x)}(t) \) and Equation (13) becomes a 2-layer feed-forward neural network.

In the following, two PAM models that fits within this PoE framework will be described. These models differ by the forms of which \( \mathcal{G}_{js}(.) \) and \( \mathcal{H}_{js}(.) \) assume.

A. The Product-of-Posterior (PoP) model

The Product-of-Posterior (PoP) model uses the following expert and weight functions:

\[
\begin{align*}
    \mathcal{E}_{js}(p^{(x)}(t), \lambda_{js}^{(E)}) &= p_{s}^{(x)}(t) \quad (14) \\
    \mathcal{H}_{js}(p^{(x)}(t), \lambda_{js}^{(H)}) &= \gamma_{js} \quad (15)
\end{align*}
\]

Each expert is given by the state posterior probabilities produced by the foreign phone recogniser, \( p_{j}^{(x)}(t) \). No additional parameters are used for the experts, i.e. \( \lambda_{js}^{(E)} = \emptyset \). On the other hand, the expert weights are simply modelled as scalar values, \( \gamma_{js} \), independent of \( p^{(x)}(t) \). By substituting Equation (14) and Equation (15) into Equation (11), the expression of the posterior probabilities of the target states becomes:

\[
p_{j}^{(y)}(t) = \frac{1}{Z} \prod_{s=1}^{S} p_{s}^{(x)}(t) \gamma_{js}
\]

(16)

Therefore, the posterior probability of the \( j \)th target state is modelled as the weighted product of the posterior probabilities of the source states, \( s \). If \( \gamma_{js} \) is constrained to be either 0 or 1, the above expression becomes the joint posterior probability of those source states that correspond to \( \gamma_{js} = 1 \). One may view this model as ‘synthesising’ the target states by joining several source states, such that the original observation vector at time \( t \) will generate a high target state posterior probabilities only when the posterior probabilities of the corresponding source states are jointly high. It is interesting to note that when only one of the \( \gamma_{js} \) is nonzero for each target state \( j \), it becomes a many-to-one mapping, i.e. each source state is mapped to one and only one target state. In this case, each source state is treated as an independent expert. In general, the PoP model can be regarded as a 2-layer neural network such that \( \gamma_{js} \) corresponds to the NN weights. The parameters can then be estimated through standard NN training via error back-propagation.
B. The Mixture-of-PoP (MoPoP) model

One limitation of the PoP model is that it can only ‘synthesize’ one target model for each target state. This limitation can be circumvented by using a Mixture-of-PoP (MoPoP) model:

\[
p_j^{(y)}(t) = \sum_{m=1}^{M} p_j^{(y)m}(t)
\]

(17)

\[
p_j^{(y)m}(t) = \frac{1}{Z_m} \prod_{s=1}^{S} p_s^{(x)}(t)^{\gamma_{jsm}}
\]

(18)

where \(p_j^{(y)}(t)\), the posterior probability of component \(m\) in state \(j\) at time \(t\), is given by a PoP model. This allows each target state, \(j\), to be associated with more than one PoP models. The hard targets used to train the PoP model in Equation (18) is now the Gaussian components instead of the HMM states. These hard targets can be obtained by performing a component-level forced-alignment using a GMM/HMM model with the desired number of mixtures. The component-level posterior probabilities within a state are then summed together using Equation (17) to obtain the state-level posterior probabilities. As the number of mixture components increases, it becomes increasingly more difficult to estimate the component-level posterior probabilities. The results shown in Section V-D reveals that the MoPoP performance is the best when \(M = 2\).

C. The Posterior Weighted Product-of-Expert (PWPoE) model

A more advanced model that can be modelled by the PoE framework is the Posterior Weighted Product-of-Expert (PWPoE) model. This model’s expert and weight functions are given by:

\[
E_{js}\left(p^{(x)}(t), \lambda^{(E)}_{js}\right) = \omega_{js}
\]

(19)

\[
\mathcal{H}_{js}\left(p^{(x)}(t), \lambda^{(H)}_{js}\right) = \frac{1}{Z} \prod_{r=1}^{R} p_r^{(x)}(t)^{\gamma_{sr}}
\]

(20)

where

\[
Z = 1 + \prod_{r=1}^{R} p_r^{(x)}(t)^{\gamma_{sr}}
\]

(21)

Each expert, \(E_{js}()\), is modelled as constant parameters independent of the state posterior probabilities generated by the foreign phone recognisers. They represent information specific to the target language which will be learnt from the training data. On the other hand, the expert weights, \(\mathcal{H}_{js}\left(p^{(x)}(t), \lambda^{(H)}_{js}\right)\), are modelled as a PoP model such that the weights are non-negative posterior values. Note that the expert weights are shared by all the target states, i.e. they are independent of \(j\). This leads to a cascade of two PoE models which can be conveniently modelled as a 3-layer feed-forward neural network with a sigmoid hidden layer activation, (c.f. Equation (20)). The weights of the synapses connecting the input and hidden layers are used to predict the expert weights, while those connecting the hidden and output layers are themselves the experts for generating the target state posterior probabilities. The number of hidden units corresponds to the number of experts in the system. This is a variable that can be adjusted for model complexity control.

V. EXPERIMENTAL RESULTS

This section will report experimental results of cross-lingual phone recognition using the Discriminative Product-of-Expert Probabilistic Acoustic Mapping models. The NTIMIT\(^1\) database was used to train and evaluate the PAM models. The NTIMIT database is obtained by passing the TIMIT database through the telephone channel. There were a total of 3696 training utterances (\(\sim 3.13\) hours) and 1344 testing utterances (\(\sim 1.14\) hours). A subset of the training data (312 utterances, \(\sim 0.26\) hours) was used to train the PAM models. This allows the study of the robustness of the PAM models in adapting to a new language using only limited amount of data. These training sets will be referred to as fullset and subset.

For cross-lingual phone recognition, three foreign phone recognisers were used to recognise the NTIMIT test set. These recognisers were the hybrid NN/HMM phone recognisers [13] train on the Czech (CZ), Hungarian (HU)and Russian (RU) telephone speech data from the SpeechDat(E) database [12]. These phone recognisers use the TRAP features to train the neural networks to produce the state posterior probabilities. The details of these phone recognisers are shown in Table I.

<table>
<thead>
<tr>
<th>Phone Recogniser</th>
<th># of phones</th>
<th>PER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CZ</td>
<td>46</td>
<td>24.24</td>
</tr>
<tr>
<td>HU</td>
<td>62</td>
<td>33.32</td>
</tr>
<tr>
<td>RU</td>
<td>53</td>
<td>39.27</td>
</tr>
</tbody>
</table>

A. Baseline Monolingual Phone Recognition

Firstly, a set of baseline performance results are reported for monolingual phone recognition. A set of baseline GMM/HMM systems were built using the 39-dimensional MFCC vectors comprising 12 static coefficients, the \(C0\) energy term and the first two differential parameters. Both ML and MMI trained GMM/HMM systems up to 32-components were built using the subset and fullset training data. The Phone

<table>
<thead>
<tr>
<th>Training Paradigm</th>
<th># of Gaussian Components</th>
<th>PER (%)</th>
<th>subset</th>
<th>fullset</th>
</tr>
</thead>
<tbody>
<tr>
<td>ML</td>
<td>16</td>
<td>58.64</td>
<td>56.44</td>
<td></td>
</tr>
<tr>
<td></td>
<td>32</td>
<td>59.20</td>
<td>53.96</td>
<td></td>
</tr>
<tr>
<td>MMI</td>
<td>16</td>
<td>57.73</td>
<td></td>
<td>51.17</td>
</tr>
</tbody>
</table>

\(^1\)LDC catalogue number: LDC93S2

Error Rate (PER) performance of these systems are tabulated...
in Table II. It was found that increasing the number of Gaussian components from 1 to 16 consistently improves the performance of the systems trained on both training sets. Furthermore, larger performance improvements were obtained on the fullset data. However, while the 32-component system trained on fullset continues to show improvement (with no signs of saturation), the performance of the 32-component system trained on the subset degraded slightly. This clearly shows the over-fitting problem on the subset data. The ML GMM/HMM systems gave the best PER performance of 58.64% and 53.96% on the subset and fullset data sets respectively. MMI training further improved the performance to 57.73% and 51.17% on these training sets.

Next, the same performance comparisons were made for the hybrid NN/HMM systems. The neural networks are trained on the same 39-dimensional MFCC vectors with the hard target labels obtained by performing forced-alignment using the MMI-trained GMM/HMM systems. The neural networks, consist of 3 layers with 1000 hidden units, were trained using quicknet. The state posterior probabilities produced by the neural networks were used as the HMM state likelihood probabilities to perform the Viterbi decoding. The results are summarised in Table III. The window lengths of 1, 5 and 9 were investigated. With a window length of 1, the NN/HMM system trained on the fullset gives a PER performance of 50.45%, approximately 0.7% absolute improvement over the MMI-trained GMM/HMM system. However, using only the subset, the NN/HMM system only achieves 60.33% PER, about 2.6% behind the MMI-trained GMM/HMM system. A consistent improvement was observed as the window length was increased to 9. The best NN/HMM performance obtained were 53.43% and 43.55% on the subset and fullset respectively. These will be used as the benchmark for the subsequent cross-lingual phone recognition performance.

### Table III

<table>
<thead>
<tr>
<th>Window Length</th>
<th>PER (%)</th>
<th>subset</th>
<th>Fullset</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>60.33</td>
<td>50.45</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>55.46</td>
<td>44.74</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>53.43</td>
<td>43.55</td>
<td></td>
</tr>
</tbody>
</table>

B. Probabilistic Phone Mapping

The first set of experiments are based on the Probabilistic Phone Mapping models. The three foreign phone recognisers were used to first transcribe the speech data into the phone sequences in their respective languages. During training, the decoded phone sequence and the reference phone sequence pairs were used to train the discrete HMM models to perform the probabilistic mapping. During testing, PPM models were used to map the decoded foreign phone sequences to the target phone sequences. The performance of the PPM models are given in Table IV. For each foreign phone recogniser, two PPM models were trained, one at the phone level and the other at the state level. Performing PPM on each individual phone recogniser yields PERs of 73.39–76.55% and 72.70–75.66% with the phone and state level mappings respectively. Using state-level mapping consistently outperforms phone-level mapping. Combining the individual systems as a multiple-stream discrete HMM system yields a spatial context-sensitive PPM model, denoted by CZ⊗HU⊗RU. This system gave a large improvement of 6.31% absolute over the HU system for phone-level mapping. However, only a marginal improvement of 1.09% is obtained for state-level mapping.

### Table IV

<table>
<thead>
<tr>
<th>Phone Recogniser</th>
<th>Phone</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>CZ</td>
<td>76.56</td>
<td>75.66</td>
</tr>
<tr>
<td>HU</td>
<td>73.39</td>
<td>72.70</td>
</tr>
<tr>
<td>RU</td>
<td>76.32</td>
<td>73.81</td>
</tr>
<tr>
<td>CZ⊗HU⊗RU</td>
<td>67.08</td>
<td>71.61</td>
</tr>
</tbody>
</table>

C. Tandem Acoustic Mapping

One major disadvantage of the PPM models is that a hard decision is made about the state at every frame. Therefore, much of the information about acoustic feature is lost. A simple way of making use of the state posterior probabilities generated by the foreign recognisers is to treat these as input features and train a GMM/HMM acoustic model. This approach is known as the tandem acoustic mapping. It was found that using the log of the posterior features leads to a more Gaussian-like distribution. Table V shows the performance of the tandem GMM/HMM systems with different number of Gaussian components per state.

### Table V

<table>
<thead>
<tr>
<th># of Gaussian Components</th>
<th>CZ</th>
<th>HU</th>
<th>RU</th>
<th>CZ⊗HU⊗RU</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>75.48</td>
<td>75.48</td>
<td>77.37</td>
<td>71.81</td>
</tr>
<tr>
<td>2</td>
<td>67.07</td>
<td>66.17</td>
<td>68.42</td>
<td>63.05</td>
</tr>
<tr>
<td>4</td>
<td>62.38</td>
<td>61.61</td>
<td>62.82</td>
<td>58.31</td>
</tr>
<tr>
<td>8</td>
<td>59.26</td>
<td>58.66</td>
<td>59.24</td>
<td>55.02</td>
</tr>
<tr>
<td>16</td>
<td>56.77</td>
<td>56.08</td>
<td>56.82</td>
<td>52.51</td>
</tr>
<tr>
<td>32</td>
<td>55.86</td>
<td>55.04</td>
<td>55.10</td>
<td>51.32</td>
</tr>
</tbody>
</table>
obtained by combining the 32-component individual systems to yield 51.52% PER. Note that this is also better than the baseline GMM/HMM trained on the fullset training data.

D. Product-of-Posterior (PoP) Models

Next, discriminative training of the PoP models were investigated. As previously described in Section IV, the PoP models can be formulated as a feed-forward neural network using the softmax activation function. Since neural network is naturally a discriminative classifier, the resulting PoP models can also be viewed as discriminatively trained models. The results of the MoPoP models are shown in Table VI. When there is only one component (a PoP model), the PER performance for the CZ, HU and RU recognisers are 55.19%, 54.02% and 53.52% respectively. As described in Section IV-B, it is also possible to produce a component level posterior probabilities. With 2 components per state, it was found that absolute improvements of 2.15–2.89% were obtained. These systems were also found to be better than the individual tandem models. However, further increasing the number of components to 4 and 8 resulted in significant performance degradation. This may be caused by the lack of training data to obtain a reliable estimation of the component level posterior probabilities as the number of components increases.

E. Posterior Weighted Product-of-Expert (PWPoE)

TABLE VI

<table>
<thead>
<tr>
<th>Phone Recogniser</th>
<th># of Components</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>CZ</td>
<td>55.19</td>
</tr>
<tr>
<td>HU</td>
<td>54.02</td>
</tr>
<tr>
<td>RU</td>
<td>53.52</td>
</tr>
</tbody>
</table>

Finally, Table VII show that the results obtained from the PWPoE models were significantly improved. At the optimum number of hidden units of 800/900, mapping the output from the CZ, HU and RU recognisers yields PERs of 48.29%, 47.15% and 48.03% respectively. These results are only 2.60–4.74% behind the best monolingual NN/HMM systems trained on the fullset NTIMIT training data. Finally, the CZ@HU@RU takes the product of the state posterior probabilities generated by the 3 individual systems to obtain a new set of state posterior probabilities. This yields a further 1.02–1.73% absolute over the best individual system, giving the lowest PER of 46.00% among all the PAM models.

VI. CONCLUSIONS

This paper has proposed a Product-of-Expert framework for probabilistic acoustic mapping with application to cross-lingual phone recognition. According to this framework, the posterior probabilities of the target states can be modelled as a weighted product of experts, where either the experts or their weights are modelled as functions of the posterior probabilities of the source states, generated by the foreign phone recognisers. Two models that fit within the proposed framework are described in this paper, namely the Product-of-Posterior (PoP) and Posterior Weighted Product-of-Expert (PWPoE) models. These models can be formulated as 2-layer and 3-layer feed-forward neural networks respectively. Therefore, the parameters of these models can be estimated discriminatively using the error back-propagation technique. Experimental results on the NTIMIT database show that using only 15.6 minutes of training data to train the probabilistic acoustic mapping models achieves a cross-lingual phone recognition errors of 46.00%. A NN/HMM phone recogniser trained directly on the same amount of data gave only 53.43%.

REFERENCES