ASR for Under-Resourced Languages from Probabilistic Transcription

Mark Hasegawa-Johnson[1], *Senior Member, IEEE* Preethi Jyothi[1], *Member, IEEE* Daniel McCloy[2], Majid Mirbagheri[2], Giovanni di Liberto[3], Amit Das[1], *Student Member, IEEE* Bradley Ekin[2], *Student Member, IEEE* Chunxi Liu[4], Vimal Manohar[4], *Student Member, IEEE* Hao Tang[5], Edmund C. Lalor[3], Nancy F. Chen[6], *Senior Member, IEEE* Paul Hager[7], Tyler Kekona[2], Rose Sloan[8], and Adrian KC Lee[2] *Member, IEEE*

1. University of Illinois, 2. University of Washington, 3. Trinity College, Dublin, 4. Johns Hopkins University, 5. Toyota Technological Institute Chicago, 6. Institute for Infocomm Research, 7. MIT, 8. Yale University

Abstract

In many under-resourced languages it is possible to find text, and it is possible to find speech, but transcribed speech suitable for training automatic speech recognition (ASR) is unavailable. In the absence of native transcripts, this paper proposes the use of a probabilistic transcript: a probability mass function over possible phonetic transcripts of the waveform. Three sources of probabilistic transcripts are demonstrated. First, self-training is a well-established semi-supervised learning technique, in which a cross-lingual ASR first labels unlabeled speech, and is then adapted using the same labels. Second, mismatched crowdsourcing is a recent technique in which non-speakers of the language are asked to write what they hear, and their nonsense transcripts are decoded using noisy channel models of second-language speech perception. Third, EEG distribution coding is a new technique in which non-speakers of the language listen to it, and their electrocortical response signals are interpreted to indicate probabilities. ASR was trained in four languages without native transcripts. Adaptation using mismatched crowdsourcing significantly outperformed self-training, and both significantly outperformed a cross-lingual baseline. EEG distribution coding and text-derived phone language models were both shown to improve the quality of probabilistic transcripts derived from mismatched crowdsourcing.

Index Terms

Automatic speech recognition, Under-resourced languages, Mismatched crowdsourcing, EEG

EDICS Category: SPE-RECO

ASR for Under-Resourced Languages from Probabilistic Transcription

I. Introduction

Automatic speech recognition (ASR) has the potential to provide database access, simultaneous translation, and text/voice messaging services to anybody, in any language, dramatically reducing linguistic barriers to economic success. To date, ASR has failed to achieve its potential, because successful ASR requires very large labeled corpora; the human transcribers must be computer-literate, and they must be native speakers of the language being transcribed. Large corpora are beyond the resources of most under-resourced language communities; we have found that transcribing even one hour of speech may be beyond the reach of communities that lack large-scale government funding. In order to create the databases reported in this paper, for example, we sought paid native transcribers, at a competitive wage, for the 68 languages in which we have untranscribed audio data. We found transcribers willing to work in only eleven of those languages, of which only seven finished the task.

Instead of recruiting native transcribers in search of a perfect reference transcript, this paper proposes the use of probabilistic transcripts. A probabilistic transcript is a probability mass function, $\rho_\Phi(\phi)$, specifying, as a real number between 0 and 1, the probability that any particular phonetic transcript ϕ is the correct transcript of the utterance. Prior to this work, machine learning has almost always assumed that the training dataset contains either deterministic transcripts ($\rho_{DT}(\phi) \in \{0, 1\}$, commonly called "supervised training") or completely untranscribed utterances (commonly called "unsupervised training," in which case we assume that $\rho_{LM}(\phi)$ is given by some *a priori* language model). This article proposes that, even in the absence of a deterministic transcript, there may be auxiliary sources of information that can be compiled to create a probabilistic transcript with entropy lower than that of the language model, and that machine learning methods applied to the probabilistic transcript are able to make use of its reduced entropy in order to learn a better ASR. In particular, this paper considers three useful auxiliary sources of information:

1) SELF-TRAINING: ASR pre-trained in other languages is used to transcribe unlabeled training data in the target language.
2) MISMATCHED CROWDSOURCING: Human crowd workers who don't speak the target language are asked to transcribe it as if it were a sequence of nonsense syllables.

3) EEG DISTRIBUTION CODING: Humans who do not speak the target language are asked to listen to its extracted syllables, and their EEG responses are interpreted as a probability mass function over possible phonetic transcripts.

II. Background

Suppose we require that, in order to develop speech technology, it is necessary first to have (1) some amount of recorded speech audio, and (2) some amount of text written in the target language. These two requirements can be met by at least several hundred languages: speech audio can be recorded from podcasts or radio broadcasts, and text can be acquired from Wikipedia, Bibles, and textbooks. Recorded speech is, however, not usually transcribed; and the requirement of native language transcription is beyond the economic capabilities of many minority-language communities.

A. ASR in Under-Resourced Languages

Krauwer [27] defined an under-resourced language to be one that lacks: stable orthography, significant presence on the internet, linguistic expertise, monolingual tagged corpora, bilingual electronic dictionaries, transcribed speech, pronunciation dictionaries, or other similar electronic resources. Berment [3] defined a rubric for tabulating the resources available in any given language, and proposed that a language should be called "under-resourced" if it scored lower than 10.0/20.0 on the proposed rubric. By these standards, technology for under-resourced languages is most often demonstrated on languages that are not really under-resourced: for example, ASR may be trained without transcribed speech, but the quality of the resulting ASR can only be proven by measuring its phone error rate (PER) or word error rate (WER) using transcribed speech. The intention, in most cases, is to create methods that can later be ported to truly under-resourced languages.

The International Phonetic Alphabet (IPA [21]) is a set of symbols representing speech sounds (phones) defined by the principle that, if two phones are used contrastively (i.e., they represent distinct phonemes) in any language, then those phones should have distinct symbolic representations in the IPA. This makes the IPA a natural choice for transcripts used to train cross-language ASR systems, and indeed ASR in a new language can be rapidly deployed using acoustic models trained to represent every distinct symbol in the IPA [39]. However, because IPA symbols are defined phonemically, there is no guarantee of cross-language equivalence in the acoustic properties of the phones they represent. This problem arises even between dialects of the same language: a monolingual Gaussian mixture model (GMM) trained on five hours of Levantine Arabic can be improved by adding ten hours of Standard Arabic data, but only if the log likelihood of cross-dialect data is scaled by 0.02 [18].

Better cross-language transfer of acoustic models can be achieved, but only by using structured transfer learning methods, including neural networks (NN) and subspace Gaussian mixture models (SGMM). SGMMs use language-dependent GMMs, each of which is the linear interpolation of language-independent mean and variance vectors [37], e.g., 16% relative WER reduction was achieved in Tamil by combining SGMM with an acoustic data normalization technique [32]. NN transfer learning can be categorized as tandem, bottleneck, pre-training, phone mapping, and multi-softmax methods. In a tandem system, outputs of the NN are Gaussianized, and used as features whose likelihood is computed with a GMM; in a bottleneck system, features are extracted from a hidden layer rather than the output layer. Both tandem [44] and bottleneck [47] features trained on other languages can be combined with GMMs [47] or SGMMs [20] trained on the target language in order to improve WER.

A hybrid ASR is a system in which the NN terminates in a softmax layer, whose outputs are interpreted as phone or senone [7] probabilities. Knowledge of the target language phone inventory is necessary to train a hybrid ASR, but it is possible to reduce WER by first pre-training the NN hidden layers with multilingual data [17], [45]. A hybrid ASR can be constructed using very little in-language speech data by adding a single phone-mapping layer [42] or senone-mapping layer [10] to the output of the multilingual NN. A multi-softmax system is a network with several different language-dependent softmax layers, each of which is the linear transform of a multilingual shared hidden layer [17], [38], [47].

B. Self-Training

Self-training is a class of semi-supervised learning techniques in which a classifier labels unlabeled data, and is then re-trained using its own labels as targets. Self-training is frequently used to adapt ASR from a well-resourced language to an under-resourced language [30], [5], or in some cases, to create target-language ASR by adapting several source-language ASRs [48]. A self-trained classifier tends to be too conservative, because the tails of the data distribution are truncated by the self-labeling process [40]; on the other hand, a self-trained classifier needs to be conservative, because the error rate of the learned classifier increases at a rate more than proportional to the error rate of the self-labeling process [19]. Self-training is therefore most useful when the in-language training data are filtered, to exclude frames with confidence below a threshold [46], and/or weighted, so that some frames are allowed to influence the learned parameters more than others [16]. Self-training of NN systems has been shown to be about 50% more effective (1.5 times the error rate reduction) as self-training of GMM systems [19].

C. Mismatched Crowdsourcing

In [24], a methodology was proposed that bypasses the need for native language transcription: mismatched crowdsourcing sends target language speech to crowd-worker transcribers who have no knowledge of the target language, then uses explicit mathematical models of second language phonetic perception to recover an equivalent phonetic transcript (Fig. 1). Majority voting is re-cast, in this paradigm, as a form of error-correcting code (redundancy coding), which effectively increases the capacity of the noisy channel; interpretation as a noisy channel permits us to explore more effective and efficient forms of error-correcting codes. Assume that cross-language phone misperception is a finite-memory process, and can therefore be modeled by a finite state transducer (FST). The complete sequence of representations from utterance language to annotation language can therefore be modeled as a noisy channel represented by the composition of up to three consecutive FSTs (Fig. 1): a pronunciation model, a misperception model, and an inverted grapheme-to-phoneme (G2P) transducer.

D. Electrophysiology of Speech Perception

The human auditory system is sensitive to within-category distinctions in speech sounds, but such pre-categorical perceptual distinctions may be lost in transcription tasks, where a listener must filter their percepts through the limited number of categorical representations available in their native language orthography. EEG distribution coding is a proposed new method that interprets the electrical evoked potentials of an untrained listener (measured by electroencephalography or EEG) as a probability distribution

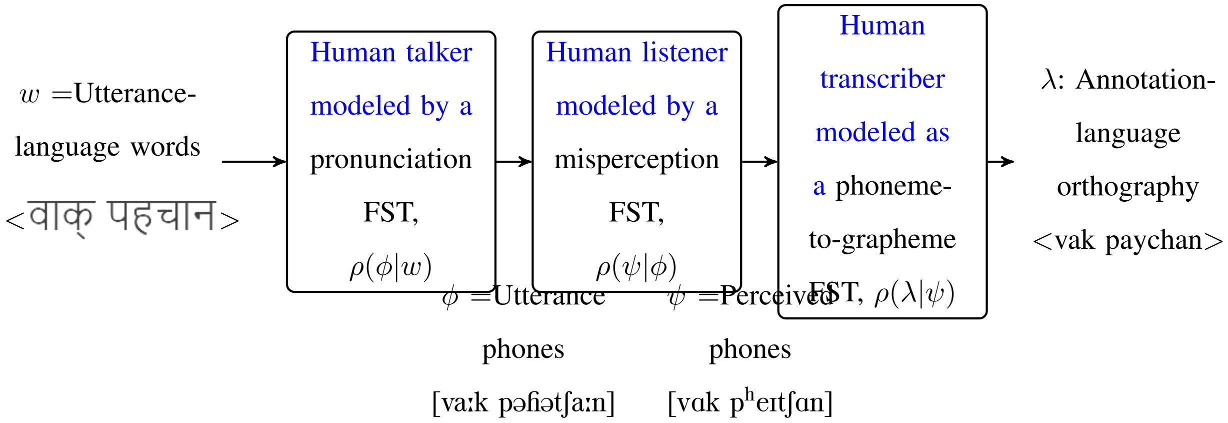

w =Utterance-language words
<वाक् पहचान>
Human talker modeled by a pronunciation FST, $\rho(\phi|w)$
Human listener modeled by a misperception FST, $\rho(\psi|\phi)$
Human transcriber modeled as a phoneme-to-grapheme FST, $\rho(\lambda|\psi)$
λ: Annotation-language orthography
<vak paychan>
ϕ =Utterance phones
[vaːk pəɦiətʃaːn]
ψ =Perceived phones
[vɑk pʰeɪtʃɑn]

Fig. 1. Mismatched Crowdsourcing: crowd workers on the web are asked to transcribe speech in a language they do not know. Annotation mistakes are modeled by a finite state transducer (FST) model of utterance-language pronunciation variability (reduction and coarticulation), composed with an FST model of non-native speech misperception (mapping utterance-language phones to annotation-language phones), composed with an inverted grapheme-to-phoneme (G2P) transducer.

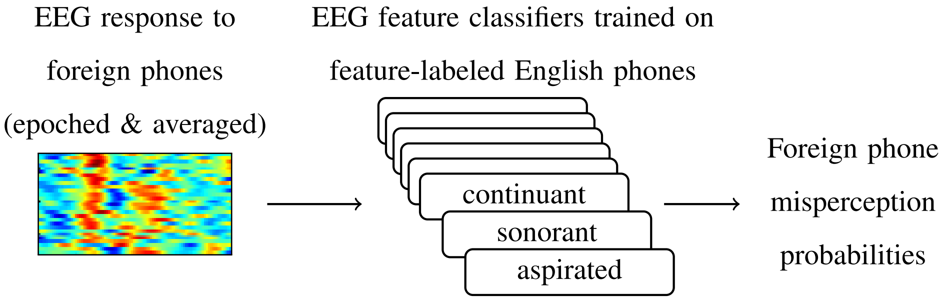


Fig. 2. EEG responses are recorded while the listener hears speech in his native language. A bank of distinctive feature classifiers are trained. The listener then hears speech in an unfamiliar language, and his EEG responses are classified, in order to estimate arc weights in the misperception FST.

over the phone set of the utterance language (Fig. 2). A transcriber, in this scenario, listens to speech in both his native language and an unfamiliar non-native target language, while his EEG responses are recorded. From his responses to English speech, an English-language EEG phone recognizer is trained [9]. Misperception probabilities $\rho(\psi|\phi)$ are then estimated: for each non-native phone ϕ, the classifier outputs are interpreted as an estimate of $\rho(\psi|\phi)$.

III. Algorithms that Induce a Probabilistic Transcript

A deterministic transcript is a sequence of phone symbols, $\phi^\ell = [\phi_1^\ell, \dots, \phi_M^\ell]$ where ϕ_m^ℓ is a symbol drawn from the phone set of the utterance language.

A probabilistic transcript is a probability mass function (pmf) over the set of deterministic transcripts. Capital letters denote random variables, lowercase denote instances: Φ_m^ℓ is a random variable whose instance is ϕ_m^ℓ. Denote the probability of transcript ϕ^ℓ as $\rho_{\Phi^\ell}(\phi^\ell)$, where ρ ("reference") means that $\rho_{\Phi^\ell}(\phi^\ell)$ is a reference distribution—a distribution specified by the probabilistic transcription process, and not dependent on ASR parameters during training. The distribution label Φ^ℓ is omitted when clear from the instance label, e.g., $\rho(\phi^\ell)$, but $\rho_{\Phi^\ell}(u)$. Superscript denotes waveform index, while subscript denotes frame or phone index. Absence of either superscript or subscript denotes a collection, thus $\Phi = \left\{\Phi^1, \dots, \Phi^L\right\}$ (with instance value $\phi = \left\{\phi^1, \dots, \phi^L\right\}$) is the random variable over all transcripts of the database. In all of the work described in this paper, the probabilistic transcript is represented as a confusion network [31], meaning that it is the product of independent symbol pmfs $\rho(\phi_m^\ell)$:

$$\rho(\phi) = \prod_{\ell=1}^{L} \rho(\phi^\ell) = \prod_{\ell=1}^{L} \prod_{m=1}^{M} \rho(\phi_m^\ell) \tag{1}$$

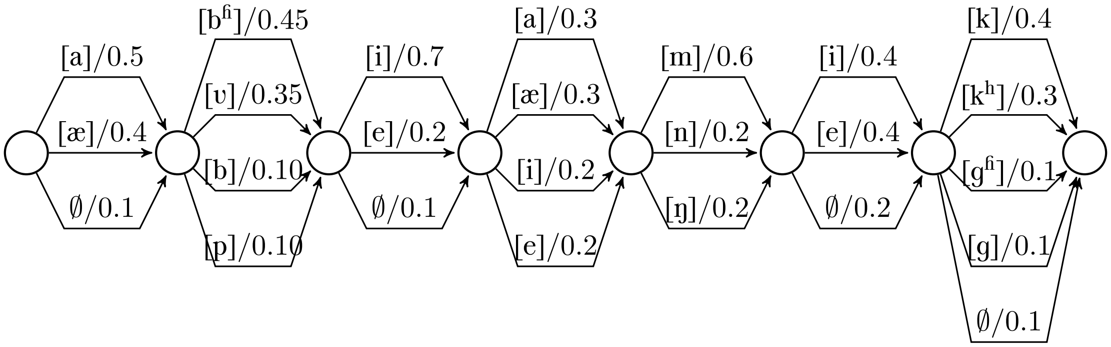


Fig. 3. A probabilistic transcript (PT) is a probability mass function (pmf) over candidate phonetic transcripts. All PTs considered in this paper can be expressed as confusion networks, thus, as sequential pmfs over the null-augmented space of IPA symbols. In this schematic example, ∅ is the null symbol, symbols in brackets are IPA, and numbers indicate probabilities.

The pmf $\rho(\phi^\ell)$ can be represented as a weighted finite state transducer (wFST) in which edges connect states in a strictly left-to-right fashion without skips, and in which the edges connecting state m to state $m+1$ are weighted according to the pmf $\rho(\phi^\ell_m)$ (Fig. 3).

Three different experimental sources were tested for the creation of a PT. Self-training is now well-established in the field of under-resourced ASR; we adopted the algorithm of Vesely, Hannemann and Burget [46]. Mismatched crowdsourcing used original annotations collected using published methods [25]. EEG was not used independently here, but rather, was used to learn a misperception model applicable to the interpretation of mismatched crowdsourcing.

A. Self-Training

The first set of PTs is computed using NN self-training. The Kaldi toolkit [36] is first used to train a cross-lingual baseline ASR, using training data drawn from six languages not including the target language. The goal of self-training, then, is to adapt the NN to a database containing L speech waveforms in the target language, each represented by acoustic feature matrix $x^\ell = [x^\ell_1, \ldots, x^\ell_T]$, where x^ℓ_t is an acoustic feature vector. The feature matrix x^ℓ represents an utterance of an unknown phone transcript $\phi^\ell = [\phi^\ell_1, \ldots, \phi^\ell_M]$ which, if known, would determine the sequence but not the durations of senones (HMM states) $s^\ell = [s^\ell_1, \ldots, s^\ell_T]$.

The feature matrix x^ℓ is decoded using the cross-lingual baseline ASR, generating a phone lattice output. Using scripts provided by previous experiments [46], the phone lattice is interpreted as a set of posterior senone probabilities $\rho(s^\ell_t|x^\ell_t)$ for each frame, and the senone posteriors serve as targets for re-estimating the NN weights. Experiments using other datasets found that self-training should use best-path alignment to specify a binary target for NN training [46], but, apparently because of differences in the adaptation set between our experiments and previous work, we achieve better performance using real-valued targets. As in previous work, senones with a posterior probability below 0.7 were removed from the training set, thus the training target was a number between 0.7 and 1.0.

B. *Mismatched Crowdsourcing*

The second set of PTs was computed by sending audio in the target language to non-speakers of the target language, and asking them to write what they hear. It would be preferable to recruit transcribers who speak a language with predictable orthography, but since transcribers in those languages were more expensive, this experiment instead recruited transcribers who speak American English. Denote using T the set of mismatched transcripts produced by these English-speaking crowd workers, which we wish to interpret as a pmf over target-language phone sequences, $\rho(\phi|T)$. As an intermediate step, prior work [25] developed techniques to merge texts into a confusion network $\rho(\lambda|T)$ over representative transcripts in the annotation-language orthography (Fig. 4).

Once transcripts have been aligned and filtered to create the orthographic confusion network $\rho(\lambda|T)$,

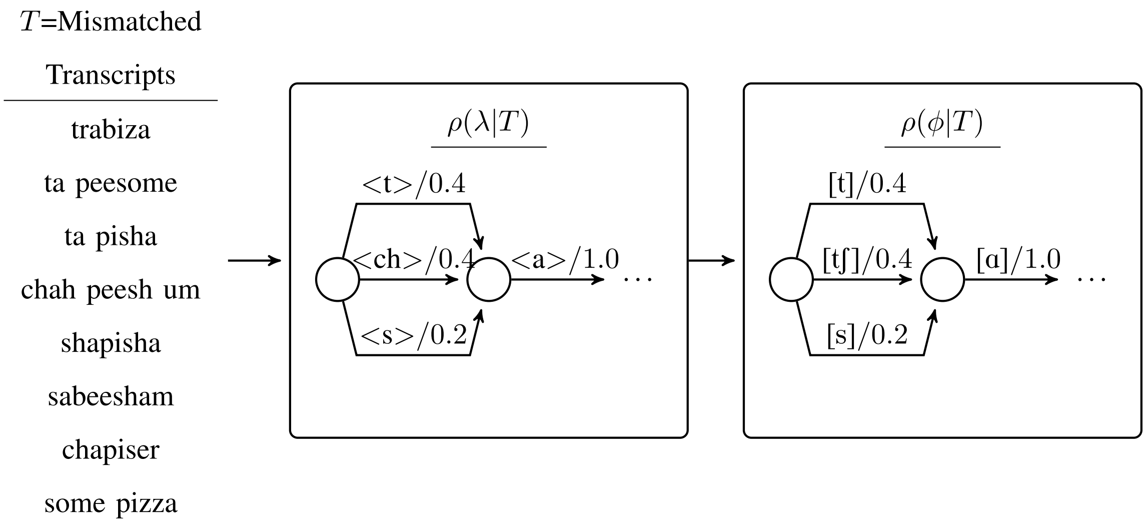


Fig. 4. Probabilistic transcription from mismatched crowdsourcing: Transcripts T are filtered to remove outliers, and merged to create a confusion network over orthographic symbols, $\rho(\lambda|T)$, from which the probabilistic transcript $\rho(\phi|T)$ is inferred. Example shown: Swahili speech, English-speaking transcribers. Symbols in <> are graphemes, symbols in [] are phones, numbers are probabilities.

they are then translated into a distribution over phone transcripts according to:

$$\begin{aligned}\rho(\phi|T) &\approx \max_{\lambda} \rho(\phi|\lambda)\rho(\lambda|T) \\ &= \max_{\lambda} \left(\frac{\rho(\lambda|\phi)}{\rho(\lambda)}\rho(\phi)\right)\rho(\lambda|T)\end{aligned} \tag{2}$$

The terms other than $\rho(\lambda|T)$ in Equation (2) are estimated as follows. $\rho(\lambda)$ is modeled using a unigram prior over the letter sequences in λ. $\rho(\phi)$ is modeled using either a cross-lingual phone unigram, a language-constrained cross-lingual unigram (the cross-lingual unigram, constrained to take values from the phone set of the target language), or a language-specific phone bigram $\rho(\phi) = \prod_{m=1}^{M} \rho(\phi_m|\phi_{m-1})$. Sec. IV-C describes an algorithm for training the phone bigram without using proscribed test-language resources; Sec. V lists the PT accuracies achieved using each of these three approaches. $\rho(\lambda|\phi)$ is called the misperception G2P, as it maps to graphemes in the annotation language, λ, fsrom phones in the utterance language, ϕ. Section III-C describes methods that decompose $\rho(\lambda|\phi)$ into separate misperception and G2P transducers, but it can also be trained directly using representative transcripts λ (and their corresponding native transcripts) for speech *in languages other than the target language*. The model learned in this way is essentially a machine translation model, which translates between graphemes in the annotation language (λ) to phonemes in any possible utterance language (ϕ). We assume that misperceptions depend more heavily on the annotation language than on the utterance language, and that therefore a model $\rho(\lambda|\phi)$ trained using a universal phone set for ϕ is also a good model of $\rho(\lambda|\phi)$ for the target language. Note that, while this assumption is not entirely accurate, it is necessitated by the requirement that no native transcripts in the target language can be used in building any part of our system.

C. *Estimating Misperceptions from Electrocortical Responses*

The misperception G2P described in Section III-B was estimated using a combination of mismatched and deterministic transcripts of non-target languages. However, with a small amount of transcribed data in the utterance language, it is possible to estimate the misperception G2P using electrocortical measurements of non-native speech perception. In this approach, the misperception G2P is decomposed into two separate transducers, a misperception transducer $\rho(\psi|\phi)$, and an annotation-language G2P $\rho(\lambda|\psi)$:

$$\rho(\lambda|\phi) \approx \sum_{\psi} \rho(\lambda|\psi)\rho(\psi|\phi) \tag{3}$$

where ϕ is a phone string in the utterance language, ψ is a phone string in the annotation language, and λ is an orthographic string in the annotation language. $\rho(\lambda|\psi)$ is an inverted G2P in the annotation language, e.g., trained on the CMU dictionary of American English pronunciations [28]. $\rho(\psi|\phi)$ is the mismatch transducer, specifying the probability that a phone string ϕ in the utterance language will be mis-heard as the annotation-language phone string ψ.

In principle, the mismatch transducer could be computed empirically from a phone confusion matrix, if experimental data on phone confusions were available for all phones in the target language, and those data were based on responses from a listener with the same language background as the crowd worker transcribers. These goals are hard to meet. An alternative is to use distinctive feature representations (originally proposed to characterize the perceptual and phonological natural classes of phonemes [22]) to predict misperceptions based on differences between the distinctive feature values of annotation- and utterance-language phones. Given the assumption that every distinctive feature shared by phones ϕ and ψ independently increases their confusion probability, their confusion probability can be expressed as

$$\rho(\psi|\phi) \propto \exp\left(-\sum_{k=1}^{K} w_k(\phi,\psi)\right) \tag{4}$$

where $w_k(\phi,\psi)$ is smaller if ϕ and ψ share the k^{th} distinctive feature. The assumption of independence is a simplifying assumption, given that many distinctive features have overlapping acoustic correlates. For example, the frequencies of the *two* lowest resonances of the vocal tract (the primary cues for vowel identity) are determined by articulatory gestures of the lips, jaw and tongue that are commonly represented by *three or more* distinctive features (e.g., height, backness, rounding, and advanced tongue root). Moreover, the weights w_k will probably also depend on properties of the speaker and listener (language, dialect, and idiolect), but data to train such a rich model do not exist.

However, a reasonable approximate model can be learned by assuming that w_k depend only on information about the listener, which can be incorporated via measurements of electrocortical activity. In particular, the weights w_k of the distinctive features can be set based on similarity of electrocortical responses (measured using EEG) as determined by a classifier trained to compute distinctive feature representations from electrocortical responses to the listener's native language phones. Thus, suppose a listener first hears phones $\phi = \psi$ in the native language, EEG response signals y are recorded, and a bank of binary classifiers $g_k(y)$ are trained to label the distinctive features $f_k(\phi)$ [9]. Second, the same listener hears phones $\phi \neq \psi$ in a new language, and EEG response signals y are recorded; then the contributions in Eq. (4) can be estimated as

$$w_k(\phi,\psi) = -\ln \Pr\left\{g_k(y) = f_k(\phi)\right\} \tag{5}$$

IV. Algorithms for Training ASR Using Probabilistic Transcription

An ASR is a parameterized pmf, $\pi(x, s|\phi, \theta)$, specifying the dependence of acoustic features, x, and senones, s, on the phone transcript ϕ and the parameter vector θ, where the notation $\pi(\cdot)$ denotes a pmf dependent on ASR parameters. Assume a hidden Markov model (HMM), therefore

$$\pi(x, s|\phi, \theta) = \prod_{\ell=1}^{L}\prod_{t=1}^{T} \pi(s_t^\ell|s_{t-1}^\ell, \phi^\ell, \theta)\pi(x_t^\ell|s_t^\ell, \theta)$$

A. Maximum Likelihood Training

Consider two observation-conditional sequence distributions $\pi(s, \phi|x, \theta)$ and $\pi(s, \phi|x, \theta')$, with parameter vectors θ and θ' respectively. The cross-entropy between these distributions is:

$$H\left(\theta\|\theta'\right) = -\sum_{s,\phi} \pi(s, \phi|x, \theta) \ln \pi(s, \phi|x, \theta') \tag{6}$$

$$= \mathcal{L}\left(\theta'\right) - Q\left(\theta, \theta'\right) \tag{7}$$

where the data log likelihood, $\mathcal{L}\left(\theta'\right)$, and the expectation maximization (EM) quality function, $Q\left(\theta, \theta'\right)$ [8], are

$$\mathcal{L}\left(\theta'\right) = \ln \pi(x|\theta') \tag{8}$$

$$Q\left(\theta, \theta'\right) = \sum_{s,\phi} \pi(s, \phi|x, \theta) \ln \pi(x, s, \phi|\theta') \tag{9}$$

Cross-entropy is bounded ($H\left(\theta\|\theta'\right) \geq H\left(\theta\|\theta\right)$), therefore

$$\mathcal{L}\left(\theta'\right) - \mathcal{L}\left(\theta\right) \geq Q\left(\theta, \theta'\right) - Q\left(\theta, \theta\right) \tag{10}$$

Given any initial parameter vector θ_n, the EM algorithm finds $\theta_{n+1} = \text{argmax}_{\theta'} Q(\theta_n, \theta')$, thereby maximizing the minimum increment in $\mathcal{L}(\theta)$. For GMM-HMMs, the quality function $Q\left(\theta, \theta'\right)$ is concave and can be analytically maximized; for NN-HMMs it is non-concave, but can be maximized using gradient ascent [2].

The probability $\pi(x, s, \phi|\theta)$ is computed by composing the following three weighted FSTs:

$$H : s^\ell \rightarrow s^\ell/\pi(x^\ell|s^\ell, \phi^\ell, \theta) \tag{11}$$

$$C : s^\ell \rightarrow \phi^\ell/\pi(s^\ell|\phi^\ell, \theta) \tag{12}$$

$$PT : \phi^\ell \rightarrow \phi^\ell/\rho(\phi^\ell) \tag{13}$$

where the notation has the following meaning. The probabilistic transcript, PT, is an FST that maps any phone string to itself. This mapping is deterministic and reflexive, but comes with a path cost determined by the transcription probability $\rho(\phi^\ell)$, as exemplified in Fig. 3. The context transducer, C, maps any senone sequence s^ℓ to a phone sequence ϕ^ℓ [33]. This mapping is stochastic, and the path cost is determined by the HMM transition weights

$$\pi(s^\ell|\phi,\theta) = \prod_{t=1}^{T} \pi(s_t^\ell|s_{t-1}^\ell,\phi^\ell,\theta) \tag{14}$$

The acoustic model, H, maps any senone sequence to itself. This mapping is deterministic and reflexive, but comes with a path cost determined by the acoustic modeling probability

$$\pi(x^\ell|s^\ell,\phi^\ell,\theta) = \prod_{t=1}^{T} \pi(x_t^\ell|s_t^\ell,\theta) \tag{15}$$

The posterior probability $\pi(s,\phi|x,\theta)$ is computed by composing the FSTs, pushing toward the initial state (normalizing so that probabilities sum to one), then finding the total cost of the path through $\text{PUSH}\,(H \circ C \circ PT)$ with input string s and output string ϕ. The analytical maximum of $Q\,(\theta,\theta')$ can be computed efficiently using the Baum-Welch algorithm, but experiments reported in this paper did not do so, for reasons described in the next subsection.

B. Segmental K-Means Training

The EM quality function, $Q(\theta,\theta')$, has properties that make it undesirable as an optimizer for $\mathcal{L}$. Suppose, as often happens, that there is a poor phone sequence, ϕ^p, that is highly unlikely given the correct parameter vector θ^*, meaning that $\pi(x,s,\phi^p|\theta^*)$ is very low. Suppose that the initial parameter vector, θ, is less discriminative, so that $\pi(x,s,\phi^p|\theta) > \pi(x,s,\phi^p|\theta^*)$. Indeed, the best speech recognizer is a parameter vector θ^* that completely rules out poor transcripts, setting $\pi(x,s,\phi^p|\theta^*) = 0$; but in this case $Q(\theta,\theta^*) = -\infty$. It is therefore not possible for the EM algorithm to start with parameters θ that allow ϕ^p, and to find parameters θ^* that rule out ϕ^p. With probabilistic transcription, this problem is quite common: if the human transcribers fail to rule out ϕ^p (e.g., because the correct and incorrect transcripts are perceptually indistinguishable in the language of the transcribers), then the EM algorithm will also never learn to rule out ϕ^p.

EM's inability to learn zero-valued probabilities can be ameliorated by using the segmental K-means algorithm [23], which bounds $\mathcal{L}(\theta')$ as $\mathcal{L}(\theta') \geq R(\theta,\theta')$:

$$R(\theta,\theta') = \ln \pi(x, s^*(\theta), \phi^*(\theta)|\theta') \tag{16}$$

$$s^*(\theta), \phi^*(\theta) = \underset{s,\phi}{\operatorname{argmax}}\, \pi(s,\phi|x,\theta) \tag{17}$$

Given an initial parameter vector θ, therefore, it is possible to find a new parameter vector θ' with higher likelihood by computing its maximum-likelihood senone sequence and phone sequence $s^*(\theta), \phi^*(\theta)$, and

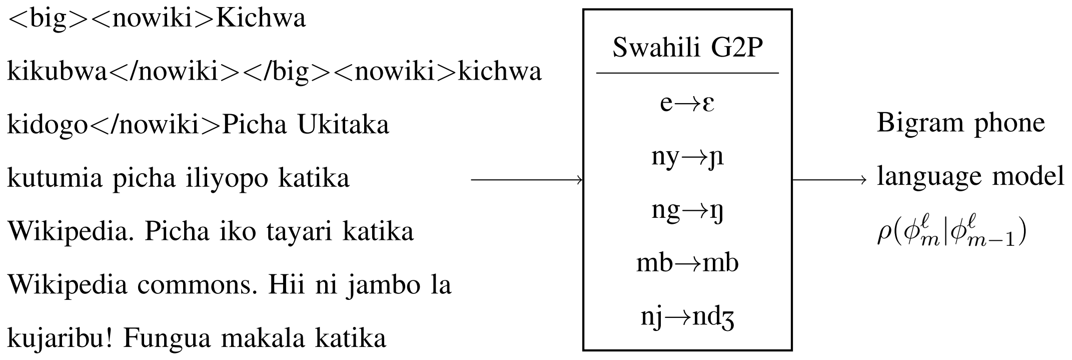

<big><nowiki>Kichwa
kikubwa</nowiki></big><nowiki>kichwa
kidogo</nowiki>Picha Ukitaka
kutumia picha iliyopo katika
Wikipedia. Picha iko tayari katika
Wikipedia commons. Hii ni jambo la
kujaribu! Fungua makala katika
Swahili G2P
e→ɛ
ny→ɲ
ng→ŋ
mb→mb
nj→ndʒ
Bigram phone
language model
$\rho(\phi^\ell_m|\phi^\ell_{m-1})$

Fig. 5. Bigram phone language model is trained using Wikipedia text (left) converted into phone strings using a zero-resource knowledge-based G2P (center).

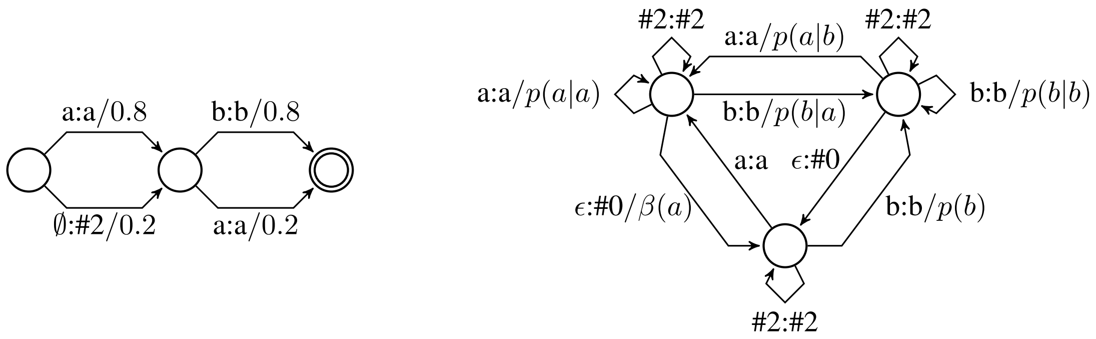
by maximizing θ' with respect to $s^*(\theta)$ and $\phi^*(\theta)$. Maximizing $R(\theta, \theta')$ rather than $Q(\theta, \theta')$ is useful for probabilistic transcription because it reduces the importance of poor phonetic transcripts.

C. Using a Language Model During Training

During segmental K-means, it is advantageous to incorporate as much information as possible about the utterance language. Define G to be an FST representing the modeled phone bigram probability $\pi(\phi^\ell|\theta) = \prod_{m=1}^{M} \pi(\phi^\ell_m|\phi^\ell_{m-1}, \theta)$. Training results can be improved by using $H \circ C \circ PT \circ G$ to compute segmental K-means.

By assumption, phone bigram information is not available from speech: we assume that there is no transcribed speech in the target language. A reasonable proxy, however, can be constructed from text. Fig. 5 shows text data downloaded from Wikipedia in Swahili, and a segment of a knowledge-based G2P for the Swahili language. Because this phone bigram will also be used in ASR testing, it is constructed using a knowledge-based method that requires zero test-language training data: The G2P is constructed by looking up “Swahili alphabet” on Wikipedia, downloading the resulting web page, and converting it by hand into an unweighted finite state transducer [15]. By passing the former through the latter, it is possible to generate synthetic phone sequences in the target language.

Composing $PT \circ G$ is complicated by the presence of null transitions in the PT. A null transition in the PT matches a non-event in the language model, for which normal FST notation has no representation. In order to compose the PT with the language model, therefore, it is necessary to introduce a special type of “non-event” symbol, here denoted “#2”, into the language model (Fig. 6). A language model “non-event” is a transition that leaves any state, and returns to the same state (a self-loop). Such self-loops, labeled with the special symbol “#2” on both input and output language, are added to every state in G (Fig. 6



(a) Probabilistic Transcript, PT

(b) Language Model, G

Fig. 6. Deletion edges in the probabilistic transcript (edges with the special null-phone symbol, $\emptyset$), required special handling in order to use information from a phone language model. As shown in (a), a new type of null symbol, "#2", was invented to represent the output for every PT edge with an $\emptyset$ input. Such edges were only allowed to match with state self-loops, newly added to the language model (b) in order to consume such non-events in the transcript. a,b: regular phone symbols, ϵ: null-string, $p(b|a)$: bigram probability, $\beta(a)$: language model backoff.

(b)). The probabilistic transcript, then, is augmented with the special symbol "#2" as the output-language symbol for every null-input edge (input symbol is $\phi_m^\ell = \emptyset$).

D. Maximum A Posteriori *Adaptation*

PT adaptation starts from a cross-lingual ASR, and adapts its parameters to PTs in the target language. The Bayesian framework for maximum *a posteriori* (MAP) estimation has been widely applied to GMM and HMM parameter estimation problems such as speaker adaptation [12]. Formally, for an unseen target language, denote its acoustic observations $x = (x_1^1, \ldots, x_T^L)$, and its acoustic model parameter set as θ, then the MAP parameters are defined as:

$$\theta_{\text{MAP}} = \underset{\theta}{\operatorname{argmax}}\, \pi(\theta|x) = \underset{\theta}{\operatorname{argmax}}\, \pi(x|\theta)\pi(\theta) \tag{18}$$

where $\pi(\theta)$ is the product of conjugate prior distributions, centered at the parameters of a cross-lingual baseline ASR. In a GMM-HMM, the acoustic model is computed by choosing a Gaussian component, G_t^ℓ, whose mixture weight is $c_{jk} = \pi_{G_t^\ell|S_t^\ell}(k|j)$, and whose mean vector and covariance matrix are μ_{jk} and Σ_{jk}. Maximum likelihood trains these parameters by computing $\gamma_t^\ell(j,k) = \pi_{S_t^\ell, G_t^\ell}(j,k|x^\ell,\theta)$, then accumulating weighted average acoustic frames with weights given by $\gamma_t^\ell(j,k)$. Segmental K-means quantizes $\pi_{S_t^\ell}(j|x^\ell,\theta) \rightarrow \{0,1\}$ using forced alignment, then proceeds identically. MAP adaptation assigns, to each parameter, a conjugate prior $\pi(\theta)$ with mode equal to $\bar{\theta}$ (the parameters of the cross-lingual baseline), and with a confidence hyperparameter τ_θ, resulting in re-estimation formulae that are linearly interpolated between the baseline parameters $\bar{\theta}$ and the statistics of the adaptation data, for example:

$$c'_{jk} = \frac{\tau_c \bar{c}_{jk} + \sum_{\ell,t} \gamma_t^\ell(j,k)}{\sum_n \left(\tau_c \bar{c}_{jn} + \sum_{\ell,t} \gamma_t^\ell(j,n) \right)} \tag{19}$$

E. Neural Networks

The NN acoustic model is $\pi_{X_t^\ell | S_t^\ell}(v|j,\theta) \propto y_t^\ell(j)$,

$$y_t^\ell(j) = \frac{1}{c_j} \frac{\exp\left(w_j^T h_t(v, w_{vh})\right)}{\sum_k \exp\left(w_k^T h_t(v, w_{vh})\right)} \tag{20}$$

whose parameters $\theta = \{c_j, w_j, w_{vh}\}$ include the senone priors c_j, the softmax weight vectors w_j, and the parameters defining the hidden nodes $h_t(v, w_{vh})$. NNs are trained by using a GMM-HMM to compute an initial senone posterior, $\pi_{S_t^\ell}(j|x^\ell, \theta)$, then minimizing the cross-entropy between the estimated senone posterior and the neural network output $y_t^\ell(j)$, using gradient descent in the direction

$$-\nabla_\theta H(S^\ell \| Y^\ell) = \sum_{t=1}^{T} \sum_j \frac{\pi_{S_t^\ell}(j|x^\ell, \theta)}{y_t^\ell(j)} \nabla_\theta y_t^\ell(j) \tag{21}$$

Preliminary experiments showed that forced alignment improves the accuracy of NNs trained from probabilistic transcripts: the best path through the PT, and the best alignment of the resulting senones to the waveform, were both computed using forced alignment. The resulting best senone string was used to train a NN using Eq. (21).

V. Audio Data and Mismatched Crowdsourcing

Speech data were extracted from publicly available podcasts [43] hosted in 68 different languages. In order to generate test corpora (in which it is possible to measure phone error rate), advertisements were posted at the University of Illinois seeking native speakers willing to transcribe speech in any of these 68 languages. Of the ten transcribers who responded, six people were each able to complete one hour of speech transcription (the other four dropped out). One additional language was transcribed by workers recruited at I^2R in Singapore, yielding a total of seven languages with native transcripts suitable for testing an ASR: Arabic (arb), Cantonese (yue), Dutch (nld), Hungarian (hun), Mandarin (cmn), Swahili (swh) and Urdu (urd). It is desirable to test the ideas in this paper with corpora larger than one hour per language, but larger corpora involve problems orthogonal to the purposes of this paper, e.g., the Babel corpora contain telephone speech, and therefore contain far more acoustic background noise than the podcast corpora used in this paper.

The podcasts contain utterances interspersed with segments of music and English. A GMM-based language identification system was developed in order to isolate regions that correspond mostly to the target language, which were then split into 5-second segments to enable easy labeling by the native transcribers. Native transcribers were asked to omit any 5-second clips that contained significant music, noise, English, or speech from multiple speakers. Resulting transcripts covered 45 minutes of speech in Urdu and 1 hour of speech in the remaining six languages. The orthographic transcripts for these clips were then converted into phonemic transcripts using language-specific dictionaries and G2P mappings. In order to make it possible to transfer ASR from training languages (which have native transcripts) to a test language (that has no native transcripts), the phone set must be standardized across all languages; for this purpose, the phone set was based on the international phonetic alphabet (IPA; [21]). Similarly, in order to transfer ASR from training languages to a test language, the training transcriptions must be converted to phonemes using a grapheme-to-phoneme transducer (G2P). G2Ps were therefore assumed to be available in all training languages, but not in the test language. Since these G2Ps are only used for training and not test languages, five of them (Arabic, Dutch, Hungarian, Cantonese and Mandarin) were trained using lexical resources, and only two (Urdu and Swahili) were constructed using the zero-resource knowledge-based method described in Sec. IV-C. English words in each transcript are identified and converted to phones with an English G2P trained using CMUdict [28], then other words are converted into phonetic transcripts using language-dependent dictionaries and G2Ps. The Arabic dictionary is from the Qatari Arabic Corpus [11], the Dutch dictionary is from CELEX v2 [1], the Hungarian dictionary was provided by BUT [14], the Cantonese dictionary is from I^2R, and the Mandarin dictionary is from CALLHOME [6]. For each language, we chose a random 40/10/10 minutes split into training, development and evaluation sets.

Mismatched transcripts were collected from annotators on Amazon Mechanical Turk. Each 5-sec speech segment was further split into 4 non-overlapping segments to make the non-native listening task easier. The crowdsourcing task was set up as described in [25]; briefly, the segments were played to annotators, who transcribed what they heard (typically in the form of nonsense syllables) using English orthography. Each segment was transcribed by 10 distinct annotators. More than 2500 annotators participated in these tasks, with roughly 30% of them claiming to know only English (Spanish, French, German, Japanese, Chinese were some of the other languages they reported knowing).

The quality of a probabilistic transcript derived from mismatched crowdsourcing is significantly improved by using a phone language model during the decoding process ($\rho(\phi)$ in Eq. (2)). Phone language models for each target language were computed from Wikipedia texts using the methods described in

Method	nld	cmn	urd	arb	hun	swh
Universal set	87.4	88.86	97.95	79.04	92.87	88.56
Target set	78.12	87.4	87.81	66.39	84.78	59.83
Phone bigram	70.43	70.88	64.67	65.29	63.98	50.45

TABLE I

Label phone error rate (LPER) of probabilistic transcripts for universal phone set, target-language phone set, text-based phone bigram.

Sec. IV-C. Label phone error rate (LPER) of the 1-best path through the resulting PTs are shown in Table I, computed with reference to a native transcript in each language. As shown, the use of a phone language model, derived from Wikipedia text, reduces LPER by about 10% absolute, in each language.

LPER of the 1-best path does not accurately reflect the extent of information in the PTs that can be leveraged during ASR adaptation. Consider, for example, the four Urdu phones [p,p$^{\text{h}}$,b,b$^{\text{ɦ}}$]. An attentive English-speaking transcriber must choose between the two letters <p,b> in order to represent any of these four phones. The misperception G2P therefore maps the letters <p,b> into a distribution over the phones [p,p$^{\text{h}}$,b,b$^{\text{ɦ}}$]. There is no reason to expect that the maximizer of $\rho(\phi|\lambda)$ is correct, but there is good reason to expect the correct answer to be a member of a short N-best list ($N \leq 4$ phones/grapheme). A fuller picture is therefore obtained by pruning the PT to a small number of paths, then searching for the most correct path in the pruned PT. One useful metric is entropy per segment, defined as $H^\ell(\Phi) = -\frac{1}{M}\sum_{m=1}^{M}\sum_u \log_2 \rho_{\Phi^\ell_m}(u)$, e.g., a PT in which every segment has two equally probable options would measure $H^\ell(\Phi) = 1$. Fig. 7 shows the trend of LPER (for three languages) obtained by pruning the PT at several different levels of $H^\ell(\Phi)$. LPER rates drop significantly across all languages within 1 bit of entropy per phone, illustrating the extent of information captured by the PTs.

VI. EEG Recording and Analysis

To compute distinctive feature weights for the misperception transducer shown in Eqs. (4) and (5), cortical activity in response to non-native phones was recorded by an EEG. Signals were acquired using a BrainVision actiCHamp system with 64 channels and 1000 Hz sampling frequency. All procedures were approved by the University of Washington Institutional Review Board.

Auditory stimuli were consonant-vowel (CV) syllables representing consonants of three languages: English, Dutch and Hindi. The inclusion of only two non-English languages was dictated by the relatively high number of repetitions needed for good signal-to-noise ratio from averaged EEG recordings. The choice of Dutch and Hindi was made based on language phonological similarity, defined as the number

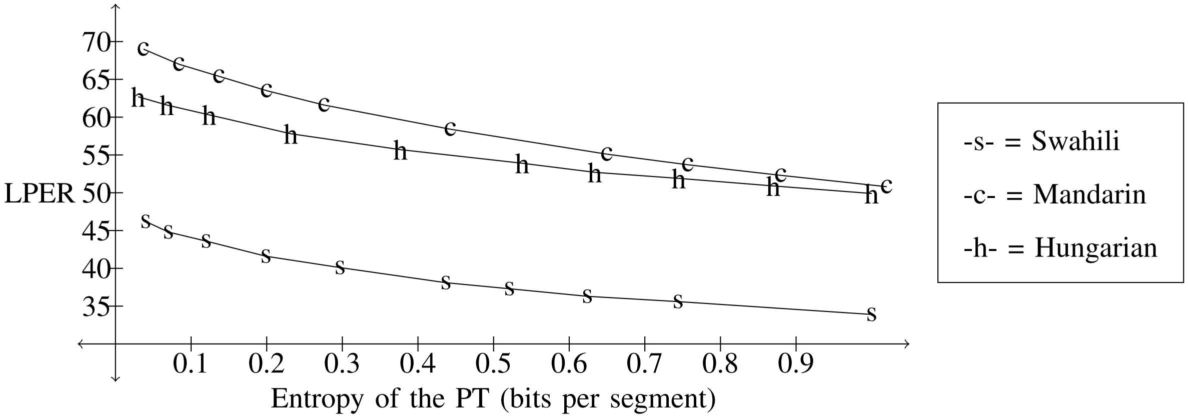


Fig. 7. LPER plotted against entropy rate estimates of phone sequences in three different languages.

of many-to-one mappings (N_{M2O}) between the English phoneme inventory and the non-English phoneme inventory. Many-to-one mappings are expected to pose a problem for the non-native transcription task being modeled by the misperception transducer, so to test the contribution of EEG we chose languages that differed greatly in this property. Using distinctive feature representations of the phonemes in each inventory from the PHOIBLE database [34], a many-to-one mapping was defined by finding, for each non-English phoneme ϕ, the English phoneme $\psi^*(\phi)$ to which it is most similar. The number of many-to-one collisions is then defined as

$$N_{M2O} = \frac{1}{|\Omega_\Psi|} \sum_{\phi_1 \neq \phi_2} [\psi^*(\phi_1) = \psi^*(\phi_2)] \tag{22}$$

where $|\Omega_\Psi|$ is the size of the English phoneme inventory, and $[\cdot]$ is the unit indicator function. The frequency of many-to-one mappings is listed in Table II for several languages. Hindi was chosen for having a large number of many-to-one mappings with English, while Dutch has relatively few. Note that, although Hindi podcasts were not included in the training data described in Section V, colloquial spoken Hindi and Urdu are extremely similar phonologically [26], and considering that the auditory stimuli for the EEG portion of this experiment are simple CV syllables, it is reasonable to consider Hindi and Urdu as equivalent for the purpose of computing feature weights for the misperception transducer.

To construct the auditory stimuli, two vowels and several consonants were selected from the phoneme inventory of each language (18 consonants for English, 17 for Dutch, and 19 for Hindi). Consonants were chosen to emphasize differences in the many-to-one relationships between English-Dutch and English-Hindi, while maintaining roughly equal numbers of consonants for each language. The consonants chosen for each language are given in Table III; the vowels chosen were the same for all three languages (/a/ and /e/).

Language	N_{M2O}	Language	N_{M2O}	Language	N_{M2O}
spa	0.862	yue	1.280	cmn	1.531
por	1.152	jpn	1.333	amh	1.844
nld	1.182	vie	1.393	hun	1.857
deu	1.258	kor	1.429	hin	2.848

TABLE II

FREQUENCY OF MANY-TO-ONE MAPPINGS N_{M2O} BETWEEN OTHER LANGUAGES' PHONEME INVENTORIES AND THE INVENTORY OF ENGLISH. LANGUAGES ARE REPRESENTED BY THEIR ISO 639-3 CODES.

	Consonants used in the EEG experiment									
eng	p	t	k	tʃ	v	ð	z	m	n	
nld	p	t	ɣ		v		z	m	n	
hin	p b	t̪ d̪ ʈ ɖ	k g		ʋ			m	n̪	ɳ
eng	pʰ	tʰ	kʰ	tʃʰ	f	θ	ʃ	l	ɹ	
nld	pʰ	tʰ	kʰ	tʃʰ	f		ʃ	l	ʀ	j
hin	bʱ	t̪ʰ ʈʰ d̪ʱ ɖʱ	kʰ gʱ							

TABLE III

CONSONANT PHONES USED IN THE EEG EXPERIMENT REPRESENTED USING IPA. VERTICAL ALIGNMENT OF CELLS SUGGESTS MANY-TO-ONE MAPPINGS EXPECTED BASED ON DISTINCTIVE FEATURE VALUES.

Two native speakers of each language (one male and one female) were recorded (44100 Hz sampling frequency, 16 bit depth) speaking multiple repetitions of the set of CV syllables for their language. Three tokens of each unique syllable were excised from the raw recordings, downsampled to 24414 Hz (for compatibility with the presentation hardware, Tucker Davis Technologies RP2.1), and RMS normalized. Recorded syllables had an average duration of 400 ms, and were presented via headphones to one monolingual American English listener. The stimuli were presented in 9 blocks of 15 minutes per block, for a total of 135 minutes. Syllables were presented in random order with an inter-stimulus interval of 350 ms. Twenty-one repetitions of each syllable were presented, for a grand total of 9072 syllable presentations.

EEG recordings were divided into 500 ms epochs. The epoched data were coded with a subset of distinctive features that minimally defined the phoneme contrasts of the English consonants. Where more than one choice of features was sufficient to define those contrasts, preference was given to features that reflect differences in temporal as opposed to spectral features of the consonants, due to the high fidelity

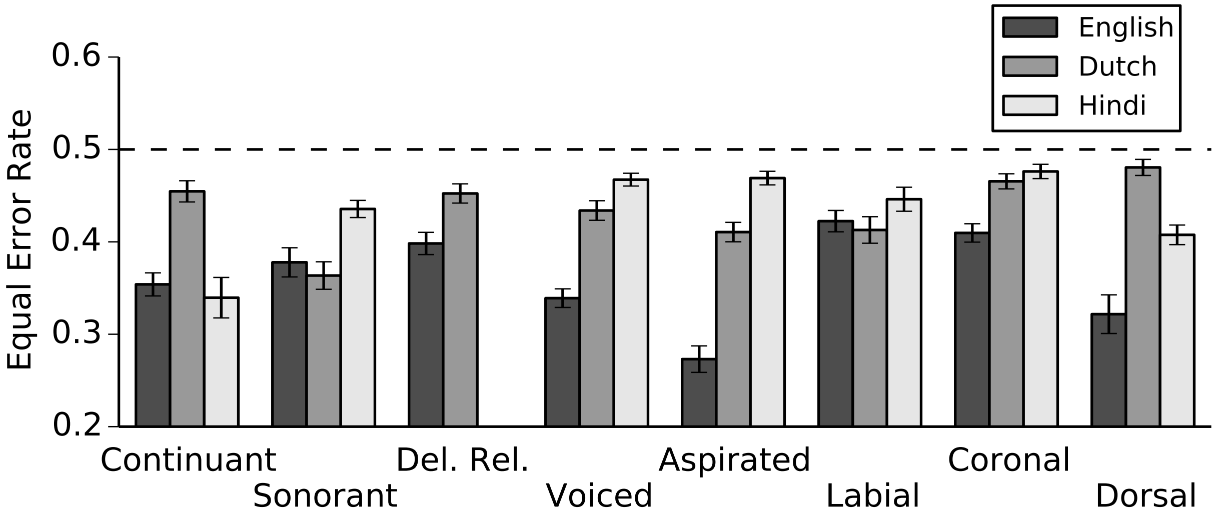


Fig. 8. Classifiers were trained to observe EEG signals, and to classify the distinctive features of the phone being heard. Equal error rates are shown for English (the language used in training; train and test data did not overlap), Dutch, and Hindi. Dashed line shows chance=50%.

of EEG at reflecting temporal envelope properties of speech [9]. The final set of features chosen was: continuant, sonorant, delayed release, voicing, aspiration, labial, coronal, and dorsal.

Epoched and feature-coded EEG data *for the English syllables only* were used to train a support vector machine classifier for each distinctive feature. The classifiers were then used (without re-training) to classify the EEG responses to the Dutch and Hindi syllables. Fig. 8 shows equal error rates of these classifiers when applied to the three languages. EER of the classifier when applied to English phones is comparable to those reported in [9], the only prior work to attempt a recognition of speech phonemes from EEG of the listener.

Eq. (4) defines a log-linear model of $\rho(\psi|\phi)$, the probability that a non-English phoneme ϕ will be perceived as English phoneme ψ. Denote by $\rho_U(\psi|\phi)$ the model of Eq. (4) with uniform binary weights for all distinctive features. Denote by $\rho_{EEG}(\psi|\phi)$ the same model, but with weights w_k derived from EEG measurements (Eq. (5)). Fig. 9 shows these two confusion matrices: $\rho_U(\psi|\phi)$ on the left, $\rho_{EEG}(\psi|\phi)$ on the right. The entropy of the binary weighting, $\rho_U(\psi|\phi)$, is too low: when a Dutch phoneme ϕ has a nearest-neighbor $\psi^*(\phi)$ in English, then few other phonemes are considered to be possible confusions. $\rho_{EEG}(\psi|\phi)$ has a very different problem: since distinctive feature classifiers have been trained for only a small set of distinctive features, there are large groups of phonemes whose confusion probabilities can not be distinguished (giving the figure its block-matrix structure). The faults of both models can be ameliorated by averaging them in some way, e.g., by computing the linear interpolation

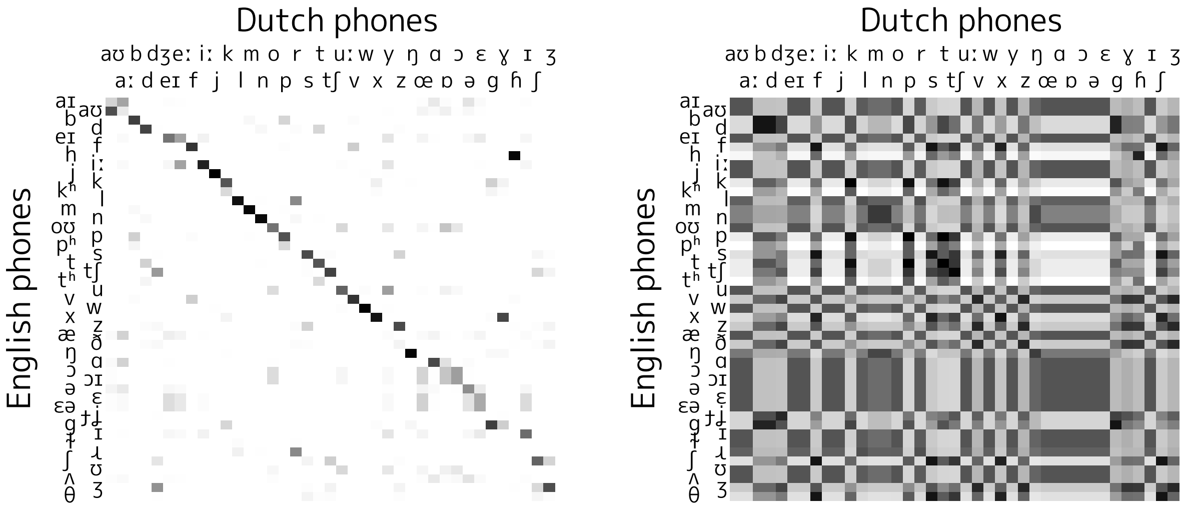


Fig. 9. Phone confusion probabilities between English and Dutch phones using models in which the negative log probability is proportional to unweighted or weighted distance between the corresponding distinctive feature vectors. Left: unweighted. Right: feature weights equal negative log confusion probability of EEG signal classifiers.

$\rho_I(\psi|\phi) = (1-\alpha)\rho_U(\psi|\phi) + \alpha\rho_{EEG}(\psi|\phi)$ for some constant $0 \leq \alpha \leq 1$.

In order to evaluate the effectiveness of the EEG-induced misperception transducer we looked at the LPER of mismatched crowdsourcing for Dutch when performed using 1) a multilingual misperception model $\rho(\lambda|\phi)$ (the machine translation model described in Sec. III-B), 2) feature-based misperception transducer computed using binary weighting, $\rho_U(\psi|\phi)$, or 3) EEG-induced transducer combined with the feature-based transducer, $\rho_I(\psi|\phi)$. Both method (2) and method (3) required the use of a G2P in order to compute $\rho(\lambda|\psi)$: the Dutch G2P was estimated using the CELEX database, while the Hindi G2P was estimated using the zero-resource knowledge-based method described in Sec. IV-C. The constant $\alpha = 0.29$ was chosen as the average of the values selected by all folds in a leave-one-out cross-validation. LPER of the multilingual model was 70.43% (as shown in Table I), of the feature-based model, 69.44%, and of the EEG-interpolated model, 68.61%.

VII. Automatic Speech Recognition

ASR was trained in four target languages in topline, baseline, and experimental conditions. Training methods are detailed in Sec. VII-A. Results are desribed in Sec. VII-B.

A. ASR Methods

Automatic speech recognition (ASR) systems were trained in four languages (hun=Hungarian, cmn=Mandarin, swa=Swahili, yue=Cantonese), using three different types of transcription. First, a topline MONOLINGUAL system was trained in each language using speech transcribed by a native speaker of that language. Second, a baseline CL (cross-lingual) system was trained using data from other languages, and tested in the target language. Third, the experimental PT-ADAPT system was created by adapting the cross-lingual system to probabilistic transcriptions in the target language. The MONOLINGAL topline system is trained using native transcripts, and converted to the phone set of the test language using the G2Ps described in Sec. V. These resoures were not available to the CL or PT-ADAPT systems, which were not permitted to use any natively transcribed training data in the test language.

Audio data, native transcripts, and probabilistic transcripts are as described in Sec. V. The MONOLINGUAL topline system was trained using 40 minutes of training data, then stream weights and insertion penalties were calculated using 10 minutes of development test data. Monolingual systems were trained using a maximum likelihood (ML) criterion using the 40 minute in-language training set: GMM parameters were initialized using a monophone system trained on the same 40 minutes, NN parameters were initialized using a restricted Boltzmann machine trained on five hours of unlabeled audio in the same language. The CL baseline systems were each trained using 40 minutes of training data in languages other than the test language. CL systems were trained using ML, maximum mutual information (MMI), minimum phone error rate (MPE), and state-based minimum Bayes risk (sMBR, [13]) training criteria. The PT-ADAPT system was initialized using the CL system (ML training), then adapted to the target language using PTs based on mismatched crowdsourcing (these transcripts are described in detail in Sec. V). Probabilistic transcripts based on EEG were not used to adapt ASR, because it is not yet possible to use EEG to generate probabilistic transcripts on a scale sufficient for ASR adaptation.

All systems were trained using the Kaldi [36] toolkit. Acoustic features consisted of MFCC (13 features), stacked ±3 frames ($13 \times 7 = 91$ features), reduced to 40 dimensions using LDA followed by fMLLR. GMM-HMM systems directly observed this 40-dimensional vector; NN-HMM systems computed fMLLR+d+dd stacked ±5 frames ($40 \times 3 \times 11 = 1320$ features/frame). All systems used tied triphone acoustic models, based on a decision tree with 1200 leaves. Each GMM-HMM used a library of 8000 Gaussians, shared among the 1200 leaves. Each NN-HMM used six hidden layers with logistic nonlinearities, and with 1024 nodes per hidden layer, followed by a softmax output layer with 1200 nodes.

The PT-ADAPT system was adapted using MAP adaptation (Sec. IV-D) computed over weighted finite state transducers in Kaldi [36]. In order to efficiently carry out the required operations on the cascade

$H \circ C \circ PT \circ G$, the cascade for PT includes an additional wFST restricting the number of consecutive deletions of phones and insertions of letters (to a maximum of 3). MAP adaptation for the acoustic model was carried out for a number of iterations (12 for yue & cmn, 14 for hun & swh, with a re-alignment stage in iteration 10).

B. ASR Results

Tables IV and V present phone error rates (PERs) for four different languages. The first column shows the phone error rate (PER) of monolingual topline systems: evaluation test results are followed by development test results in parentheses. The column titled CL lists cross-lingual baseline error rates. The column labeled ST lists the PERs of self-trained ASR systems. The column headed PT-ADAPT in Table IV lists PERs from CL ASR systems that have been adapted to PTs derived from mismatched crowdsourcing. Phone error rates are reported instead of word error rates because, in order to compute a word error rate, it is necessary to have either native transcriptions in the target language (thereby permitting the training of a grapheme-based recognizer) or a pronunciation lexicon in the target language. These resources are used by the monolingual topline, but not by any of the baseline or experimental systems.

The monolingual ASR is trained using only 40 minutes of audio and transcript data per language, but performs reasonably well (31.58% average PER, NN-HMM). The cross-lingual ASRs, however, perform poorly. Using a text-based phone bigram (denoted TEXT) gives significant improvement over a cross-language phone bigram (denoted CL), but significantly underperforms a system that has seen the test language during training. This is true even if the system has seen closely related languages during training: the Cantonese cross-lingual system has seen Mandarin during training, and the Mandarin system has seen Cantonese during training, but neither system is able to generalize well from its six training languages to its test language. Three different types of discriminative training were tested. MMI performs consistently worse than MPE and sMBR, and is therefore not listed in Table IV. Averaged across all languages and systems shown in Table IV, the development-test PERs of ML, MPE, and sMBR training are 73.43%, 73.04%, and 72.98% respectively; differences are not statistically significant, therefore only the ML system was tested on evaluation test data.

Evaluation test PER of each experimental system (columns ST and PT-ADAPT, 20 systems) was compared to evaluation test PER of the corresponding CL system (ML training, TEXT LM) using the MAPSSWE test of the `sc_stats` tool [35]. Each neural net PT-ADAPT system was also compared to the corresponding ST system (4 comparisons). There are therefore 24 independent statistical comparisons in Tables IV and V; the study-corrected significance level is $0.05/24 = 0.002$.

AM	Monolingual	Cross-Lingual (CL)						CL + PT adaptation (PT-ADAPT)		
LM	Transcript	CL			Text			Text		
Training	ML	ML	MPE	sMBR	ML	MPE	sMBR	ML	MPE	sMBR
yue	32.77 (34.61)	79.64 (79.83)	(79.49)	(79.48)	68.40 (68.35)	(68.02)	(66.94)	57.20** (56.57)	56.33** (55.70)	**55.97**** (55.51)
hun	39.58 (39.77)	77.13 (77.85)	(78.67)	(78.89)	68.62 (66.90)	(67.09)	(67.41)	**56.98**** (57.26)	57.05** (56.95)	57.05** (57.18)
cmn	32.21 (26.92)	83.28 (82.12)	(81.60)	(81.76)	71.30 (68.66)	(66.35)	(66.02)	58.21** (57.85)	55.17** (54.49)	**55.07**** (54.94)
swh	35.33 (46.51)	82.99 (81.86)	(81.80)	(82.32)	63.04 (64.73)	(63.48)	(62.99)	44.31** (48.88)	**42.80**** (46.77)	43.61** (46.55)

TABLE IV

Phone error rate of GMM-HMMs: evaluation test (development test). AM=Acoustic Model, LM=Language Model, ML=Maximum Likelihood, sMBR=State-based Minimum Bayes Risk. MAPSSWE w/respect to CL: ** denotes $p < 0.001$.

AM	Monolingual	Cross-Lingual (CL)						Self-training (ST)	CL + PT adaptation (PT-ADAPT)
LM	Transcript	CL			Text			Text	Text
Training	ML	ML	MPE	sMBR	ML	MPE	sMBR	ML	ML
yue	27.67 (28.88)	78.62 (77.58)	(77.86)	(77.92)	66.59 (65.28)	(65.76)	(65.76)	63.79ns (62.46)	**53.64**** (53.80)
hun	35.87 (36.58)	75.98 (76.44)	(77.56)	(77.61)	66.43 (66.58)	(65.52)	(65.67)	63.53ns (63.50)	**56.70**** (58.45)
cmn	27.80 (23.96)	81.86 (80.47)	(81.02)	(80.93)	65.77 (64.80)	(63.90)	(63.82)	64.90ns (64.00)	**54.07**** (53.13)
swh	34.98 (41.47)	82.30 (81.18)	(81.30)	(81.24)	65.30 (65.04)	(63.78)	(63.99)	58.76* (59.81)	**44.73**** (48.60)

TABLE V

PER of NN-HMMs: evaluation test (development test). ML=Maximum Likelihood, MPE=minimum phone error, sMBR=state-level Minimum Bayes Risk. MAPSSWE w/respect to CL AM w/ text-based LM: * means $p < 0.002$, ns=not significant. ** denotes a score lower than both CL and ST at $p < 0.001$.

Self-training was only performed using NN systems; no self-training of GMMs was performed, because previous studies [17] reported it to be less effective. The Swahili ST system was judged significantly better than CL at a level of $p = 0.002$ (denoted *); the Cantonese, Mandarin and Hungarian ST systems were not significantly better than CL at this level.

The relative reductions in PER of the PT-ADAPT system compared to both CL and ST baselines were all statistically significant at $p < 0.001$ (denoted **). This suggests that adaptation with PTs is providing more information than that obtained by model self-training alone.

PT-adapt GMM-HMM systems were trained using four different training criteria: ML, MMI, MPE and sMBR. MMI training consistently underperformed MPE and sMBR, and is therefore not shown. MPE training of PT-ADAPT systems improves their PER by a little more than 1% on average, comparable to the improvement provided to CL baseline systems.

PER improvements for Swahili are larger than for the other three languages. We conjecture this may be due to the relatively good mapping between Swahili's phone inventory and that of English. For example: all Swahili vowel qualities are also found in English, and the Swahili phonemes that would be unfamiliar to an English speaker (prenasalized stops, palatal consonants) have representations in English orthography that are fairly natural ("mb", "nd", etc. for prenasalized stops; "tya", "chya", "nya", etc. for palatals). In contrast: Mandarin, Cantonese, and Hungarian each have at least two vowel qualities not found in English; Mandarin and Cantonese have many diphthongs not found in English; and some of the consonant phonemes (e.g., Mandarin retroflexes) do not have representations in English orthography that are obvious or straightforward.

VIII. Discussion

Models of human neural processing systems have often been used to inspire improvements in machine-learning systems (for a catalog of such approaches and a warning, see [4]). These systems are often called neuromorphic, because the system is engineered to mimic the behavior of human neural systems. In contrast to that approach, our incorporation of EEG signals into ASR resonates with the Human Aided Computing approach used in computer vision [41], [49]. Together with our EEG work presented here, this class of approach represents a less explored direction for design of machine learning systems, whereby recorded neural data (rather than neuro-inspired models) are used as a source of prior information to improve system performance. Therefore, our work here suggests that, by thinking about the kinds of prior information required by a machine learning system, engineers and neuroscientists can work together to design specific neuroscience experiments that leverage human abilities and provide information that can be directly integrated into the system to solve an engineering problem.

NN-HMM outperforms the GMM-HMM in all baseline conditions, but not always when adapted using PTs. Table V shows that PT adaptation improves the NN-HMM, but the benefit to a NN-HMM is not as great as the benefit to a GMM-HMM; for this reason, the accuracy of the PT-adapted GMM-HMM catches up to that of the NN-HMM. Preliminary analysis suggests that the NN is more adversely affected than the GMM by label noise in the PTs. A NN is trained to match the senone posterior probabilities $\pi(s_t^\ell|x^\ell, \phi^\ell, \theta)$ computed by a first-pass GMM-HMM. Many papers have demonstrated that entropy in the senone posteriors is detrimental to NN training. In PT adaptation, however, entropy is unavoidable. Forced alignment is better than using soft alignment, but is not sufficient to make PT adaptation of the NN-HMM always better than that of the GMM-HMM. Table I showed that PTs computed using a text-based phone bigram language model only achieve LPER in the range 50.45-70.88%, depending on the language. These high error rates are, perhaps, incomprehensible to most speech technology experts, who are accustomed to think of human transcriptions as having 0.0% error rate, but there is good reason for this: the transcribers don't speak the target language, so they find some of its phone pairs to be perceptually indistinguishable. Future work will seek methods that can improve the robustness of NN training in the face of label noise.

The primary conclusion of this article is economic. In most of the languages of the world, it is impossible to recruit native transcribers on any verified on-line labor market (e.g., crowdsourcing). Without on-line verification, native transcriptions can only be acquired by in-person negotiation; in practice, this has meant that native transcriptions are acquired only for languages targeted by large government programs. Native transcription (NT) permits one to train an ASR with PER of 31.58% (average, first column of Table V). Self-training (ST), by contrast, costs very little, and benefits little: average PER is 62.75% (Table V). Probabilistic transcription (PT) is a point intermediate between NT and ST: average PER is 52.29%, cost is typically $500 per ten transcribers per hour of audio. PT is therefore a method within the budget of an individual researcher. We expect that an individual researcher with access to a native population will wish to combine NT (as many hours as she can convince her informants to provide) with PT (on perhaps a much larger scale); future research will study the best strategy for combining these sources of information if both are available.

IX. Conclusions

When a language lacks transcribed speech, other types of information about the speech signal may be used to train ASR. This paper proposes compiling the available information into a probabilistic transcript: a pmf over possible phone transcripts of each waveform. Three sources of information are discussed: self-training, mismatched crowdsourcing, and EEG distribution coding. Auxiliary information from EEG is used, together with text-based phone language models, to improve the decoding of transcripts from mismatched crowdsourcing. Self-trained ASR outperforms cross-lingual ASR in one of the four test languages (Swahili). ASR adapted using mismatched crowdsourcing outperforms both cross-lingual ASR and self-training in all four of the test languages.

X. Acknowledgments

This work was supported by JHU via grants from NSF (IIS), DARPA (LORELEI), Google, Microsoft, Amazon, Mitsubishi Electric, and MERL, and by NSF IIS 15-50145 to the University of Illinois. Parts of this work were previously published in [29].

REFERENCES

[1] R Baayen, R Piepenbrock, and L Gulikers. CELEX2. Technical Report LDC96L14, Linguistic Data Consortium, 1996.

[2] Y. Bengio, R. De Mori, G. Flammia, and R. Kompe. Global optimization of a neural network - hidden Markov model hybrid. *IEEE Trans. Neural Networks*, 3(2):252–259, 1992.

[3] V. Berment. *Méthodes pour informatiser des language des langues et des groupes de langues peu dotées*. PhD thesis, J. Fourier University, Grenoble, 2004.

[4] Hervé Bourlard, Hynek Hermansky, and Nelson Morgan. Towards increasing speech recognition error rates. *Speech Communication*, 18:205–231, 1996.

[5] Özgür Cetin. Unsupervised adaptive speech technology for limited resource languages: A case study for Tamil. In *Workshop on Spoken Language Technology for Under-Resourced Languages (SLTU)*, Hanoi, Vietnam, 2008.

[6] Linguistic Data Consortium. CALLHOME Mandarin Chinese. Technical Report LDC1996S34, 1996.

[7] George E. Dahl, Dong Yu, Li Deng, and Alex Acero. Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Trans. Audio, Speech and Language*, 20(1):30–42, 2012.

[8] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39:1–38, 1977.

[9] Giovanni di Liberto, James A. O'Sullivan, and Edmund C. Lalor. Low-frequency cortical entrainment to speech reflects phoneme-level processing. *Current Biology*, 25(19):2457–2465, 2015.

[10] Van Hai Do, Xiong Xiao, Eng Siong Chng, and Haizhou Li. Context dependant phone mapping for cross-lingual acoustic modeling. In *International Symposium on Chinese Spoken Language Processing (ISCSLP)*, pages 16–19, 2012.

[11] Mohamed Elmahdy, Mark Hasegawa-Johnson, and Eiman Mustafawi. Development of a TV broadcasts speech recognition system for Qatari Arabic. In *The 9th edition of the Language Resources and Evaluation Conference (LREC 2014)*, Reykjavik, Iceland, 2014.

[12] Jean-Luc Gauvain and Chin-Hui Lee. Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains. *IEEE Transactions on Speech and Audio Processing*, 2(2):291–298, 1994.

[13] M. Gibson and T. Hain. Hypothesis spaces for minimum bayes risk training in large vocabulary speech recognition. In *Proc. ICASSP*, 2006.

[14] František Grézl, Martin Karafiát, and Karel Veselý. Adaptation of multilingual stacked bottle-neck neural network structure for new language. In *Proc. ICASSP*, pages 7704–7708, 2014.

[15] Mark Hasegawa-Johnson. WS15 dictionary data. Downloaded 8/29/2016 from http://isle.illinois.edu/sst/data/g2ps/, 2015.

[16] Roger Hsiao, Tim Ng, František Grézl, Damianos Karakos, Stavros Tsakalidis, Long Nguyen, and Richard Schwartz. Discriminative semi-supervised training for keyword search in low resource languages. In *Proc. Interspeech*, pages 440–443, 2013.

[17] Jui-Ting Huang, Jing Li, Dong Yu, Li Deng, and Yifan Gong. Cross language knowledge transfer using multilingual deep neural network with shared hidden layers. In *Proc. ICASSP*, 2013.

[18] Po-Sen Huang and Mark Hasegawa-Johnson. Cross-dialectal data transferring for Gaussian mixture model training in Arabic speech recognition. In *International Conference on Arabic Language Processing (CITALA)*, pages 119–122, 2012.

[19] Yan Huang, Dong Yu, Yifan Gong, and Chaojun Liu. Semi-supervised GMM and DNN acoustic model training with multi-system combination and re-calibration. In *Proc. Interspeech*, pages 2360–2363, 2013.

[20] David Imseng, Petr Motlicek, Hervé Bourlard, and Philip N. Garner. Using out-of-language data to improve an under-resourced speech recognizer. *Speech Communication*, 56:142 – 151, 2014.

[21] International Phonetic Association (IPA). International phonetic alphabet, 1993.

[22] R. Jakobson, G. Fant, and M. Halle. Preliminaries to speech analysis. Technical Report 13, MIT Acoustics Laboratory, 1952.

[23] Bing-Hwang Juang and Lawrence Rabiner. The segmental K-means algorithm for estimating parameters of hidden Markov models. *IEEE Trans. Acoustics, Speech, and Signal Processing*, 38(9):1639–1641, 1990.

[24] Preethi Jyothi and Mark Hasegawa-Johnson. Acquiring speech transcriptions using mismatched crowdsourcing. In *Proceedings of AAAI*, 2015.

[25] Preethi Jyothi and Mark Hasegawa-Johnson. Transcribing continuous speech using mismatched crowdsourcing. In *Proceedings of Interspeech*, 2015.

[26] Yamuna Kachru. Hindi-Urdu. In Bernard Comrie, editor, *The world's major languages*, pages 399–416. Oxford University Press, New York, 1990.

[27] S. Krauwer. The basic language resource kit (BLARK) as the first milestone for the language resources roadmap. In *Proc. International Workshop Speech and Coputer (SPECOM-2003)*, pages 8–15, Moscow, Russia, 2003.

[28] Kevin Lenzo. The CMU pronouncing dictionary. Downloaded 1/31/2016 from http://www.speech.cs.cmu.edu/cgi-bin/cmudict, 1995.

[29] Chunxi Liu, Preethi Jyothi, Hao Tang, Vimal Manohar, Rose Sloan, Tyler Kekona, Mark Hasegawa-Johnson, and Sanjeev Khudanpur. Adapting ASR for under-resourced languages using mismatched transcriptions. In *Proc. ICASSP*, 2016.

[30] Jonas Lööf, Christian Gollan, and Hermann Ney. Cross-language bootstrapping for unsupervised acoustic model training: rapid development of a Polish speech recognition system. In *Proceedings of Interspeech*, 2009.

[31] L. Mangu, E. Brill, and A. Stolcke. Finding consensus in speech recognition: word error minimization and other applications of confusion networks. *Computer, Speech and Language*, 14(4):373–400, 2000.

[32] Aanchan Mohan, Richard Rose, Sina Hamidi Ghalehjegh, and S. Umesh. Acoustic modelling for speech recognition in Indian languages in an agricultural commodities task domain. *Speech Communication*, 56:167 –180, 2014.

[33] Mehryar Mohri, Fernando Pereira, and Michael Riley. Speech recognition with weighted finite-state transducers. In *Springer Handbook of Speech Processing*, pages 559–584. Springer, 2008.

[34] Steven Moran, Daniel R. McCloy, and Richard A. Wright. PHOIBLE: Phonetics Information Base and Lexicon Online, 2013.

[35] D.S. Pallet, W.M. Fisher, and J.G. Fiscus. Tools for the analysis of benchmark speech recognition tests. In *Proc. ICASSP*, volume 1, pages 97–100, 1990.

[36] D. Povey, A. Ghoshal, et al. The Kaldi speech recognition toolkit. *Proc. of ASRU*, 2011.

[37] Daniel Povey, Lukáš Burget, Mohit Agarwal, Pinar Akyazi, Feng Kai, Arnab Ghoshal, Ondrej Glembek, Nagendra Goel, Martin Karafiát, Ariya Rastrow, Richard C. Rose, Petr Schwarz, and Samuel Thomas. The subspace Gaussian mixture model—a structured model for speech recognition. *Computer Speech and Language*, 25(2):404–439, 2011.

[38] S. Scanzio, P. Laface, L. Fissore, R. Gemello, and F. Mana. On the use of a multilingual neural network front-end. In *Proc. Interspeech*, pages 2711–2714, 2008.

[39] Tanja Schultz and Alex Waibel. Experiments on cross-language acoustic modeling. In *INTERSPEECH*, 2001.

[40] H.J. Scudder. Probability of error of some adaptive pattern-recognition machines. *IEEE Trans. on Information Theory*, 11:363–371, 1965.

[41] Pradeep Shenoy and Desney S. Tan. Human-aided computing: utilizing implicit human processing to classify images. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 845–854, Florence, 2008. ACM Press.

[42] Ke Chai Sim and Haizhou Li. Context sensitive probabilistic phone mapping model for cross-lingual speech recognition. In *Proc. Interspeech*, pages 2175–2718, 2008.

[43] Special Broadcasting Services Australia. Podcasts in Your Language. http://www.sbs.com.au/podcasts/yourlanguage.

[44] A. Stolcke, F. Grezl, M.-Y. Hwang, X. Lei, N. Morgan, and D. Vergyri. Cross-domain and cross-lingual portability of acoustic features estimated by multilayer perceptrons. In *Proc. ICASSP*, 2006.

[45] P. Swietojanski, A. Ghoshal, and S. Renals. Unsupervised cross-lingual knowledge transfer in DNN-based LVCSR. In *Proc. IEEE Workshop on Spoken Language Technology (SLT)*, 2012.

[46] Karel Vesely, Mirko Hannemann, and Lukas Burget. Semi-supervised training of Deep Neural Networks. In *Proceedings of ASRU*, 2013.

[47] Karel Vesely, Martin Karafiát, Frantisek Grezl, Marcel Janda, and Ekaterina Egorova. The language-independent bottleneck features. In *Proceedings of SLT*, 2012.

[48] Ngoc Thang Vu, Franziska Kraus, and Tanja Schultz. Cross-language bootstrapping based on completely unsupervised training using multilingual A-stabil. In *Proceedings of ICASSP*, 2011.

[49] Jun Wang, Eric Pohlmeyer, Barbara Hanna, Yu-Gang Jiang, Paul Sajda, and Shih-Fu Chang. Brain state decoding for rapid image retrieval. In *Proceedings of the 17th ACM international conference on multimedia*, pages 945–954. ACM Press, 2009.