ADAPTING ASR FOR UNDER-RESOURCED LANGUAGES USING MISMATCHED TRANSCRIPTIONS

Chunxi Liu[*†], *Preethi Jyothi*[*§], *Hao Tang*[‡], *Vimal Manohar*[†], *Mark Hasegawa-Johnson*[§], *Sanjeev Khudanpur*[†]

[†] Center for Language and Speech Processing, Johns Hopkins University, MD USA
[§]Beckman Institute, University of Illinois at Urbana-Champaign, IL USA
[‡]Toyota Technological Institute at Chicago, IL USA

ABSTRACT

Mismatched transcriptions of speech in a target language refers to transcriptions provided by people unfamiliar with the language, using English letter sequences. In this work, we demonstrate the value of such transcriptions in building an ASR system for the target language. For different languages, we use less than an hour of mismatched transcriptions to successfully adapt baseline multilingual models built with no access to native transcriptions in the target language. The adapted models provide up to 25% relative improvement in phone error rates on an unseen evaluation set.

Index Terms— mismatched transcriptions, ASR adaptation, ASR for under-resourced languages

1. INTRODUCTION

Speech and language technologies are unavailable to a large majority of the world's languages. Most languages are *under-resourced* in terms of the technological resources needed to build speech recognition systems (see [1] for a detailed survey on speech processing for under-resourced languages). A first step towards building an ASR system for a new language typically involves collecting sufficient amounts of transcribed speech data. Crowdsourcing has been explored as an innovative way in which transcriptions for speech data are solicited from large numbers of crowd workers who are native speakers of the language [2, 3]. This technique, however, would be constrained to languages for which it is possible to find native speakers online. To circumvent the need for native transcribers in a language, mismatched crowdsourcing has been introduced as a technique [4, 5] that makes use of transcriptions in the form of English syllables from crowd workers unfamiliar with the target language (henceforth referred to as "mismatched transcriptions").

Mismatched transcriptions have been demonstrated to produce reasonably accurate transcriptions in both isolated word and continuous speech tasks. Using mismatched transcriptions, isolated Hindi words from a medium vocabulary task were recovered with a 1-best accuracy of over 85% [4] and transcriptions for short continuous speech segments in Hindi were labeled with an accuracy of over 55% on a large vocabulary task. A logical question that follows is: what is the impact of mismatched transcriptions on ASR performance? This is the main question that is tackled in this work.

There has been a lot of prior work on building speech technologies for under-resourced languages using multilingual models from high-resource languages and applying unsupervised adaptation techniques (e.g. [6] for Polish, [7] for Vietnamese and [8] for Czech). Our work is the first attempt to explore the use of mismatched transcriptions as adaptation data with multilingual models. We also demonstrate significant performance improvements from using mismatched transcriptions beyond that obtained from adaptation with untranscribed speech data in the target language.

2. PROBLEM SETUP

Our goal is to train a phone recognition system for a given target language in which no native transcriptions are available. We assume that we have access to unspoken texts and to untranscribed audio in the target language, but not to transcribed audio. Baseline multilingual systems are trained using native transcriptions from several different languages (not including the target language). Section 4 details multilingual GMM-HMM and DNN-based ASR systems with language-specific grammar models and Section 5 describes a semi-supervised baseline that uses unlabeled data from the target language. Next, we adapt the parameters of the acoustic model of the above system using only probabilistic phone transcriptions in the target language derived from mismatched transcriptions. The construction of probabilistic phone transcriptions is described in Section 3 and the acoustic model adaptation is detailed in Section 6.

2.1. Task Details

Our speech data were extracted from publicly available Special Broadcasting Service Australia (SBS) radio podcasts [9] hosted in 68 different languages. We restricted our experiments to seven of these languages for which we could find a native transcriber willing to provide orthographic transcriptions for roughly 1 hour of speech: Arabic (AR), Cantonese (CA), Dutch (DT), Hungarian (HG), Mandarin (MD), Swahili (SW) and Urdu (UR).[1]

The SBS radio podcasts are not entirely homogeneous in the target language and contain utterances interspersed with segments of music and English. A simple GMM-based language identification system was developed as a first pass over the podcasts in order to isolate regions that correspond mostly to the target language. These long segments were then split into smaller ≈ 5-second segments. This was to enable easy labeling by the native transcribers, and more importantly to allow for the collection of mismatched transcriptions that required the speech segments to be short (see below for more details). To further check that only speech clips in the target language were retained, the native transcribers were asked to omit

[*] Joint first author

[1] CA transcriptions were provided by Nancy Chen at I^2R in Singapore, as part of a collaborative research project. The native transcribers for the other six languages were paid student volunteers at the University of Illinois at Urbana-Champaign.

Language Code	Dev set (1-best)	Eval set (1-best)
AR	65.8	66.2
CA	66.4	67.8
DT	68.9	70.9
HG	63.7	63.5
MD	70.9	69.6
SW	47.6	50.3
UR	67.2	70.5

Table 1. 1-best probabilistic phone transcription error rates on the development and evaluation sets.

any 5-sec clips that contained music, significant amounts of noise, English speech or speech from multiple speakers. The resulting transcribed speech clips roughly amounted to 45 minutes of speech in Urdu and 1 hour of speech in the remaining seven languages. The orthographic transcriptions for these clips were then converted into phonemic transcriptions using language-specific dictionaries and grapheme-to-phoneme mappings (these resources are detailed in Section 4). For each language, we chose a random 40/10/10 minutes split into training, development and evaluation sets.

Mismatched transcriptions. Mismatched transcriptions were collected from crowd workers (Turkers) on Amazon Mechanical Turk (MTurk) [10]. The 5-sec speech segments described above were further split into 4 non-overlapping segments; shorter segments made the listening task easier for the Turkers. The crowdsourcing task was set up as described in [5]. The Turkers were asked to listen to speech segments in a language they were unfamiliar with and write down English text (typically in the form of nonsense syllables) closest to what they think they heard. Each speech segment was transcribed by 10 distinct Turkers. More than 2500 Turkers 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 listed by the Turkers.)

3. PROBABILISTIC TRANSCRIPTIONS

Our goal in this section is to compute a distribution over phone sequences π in the target language (referred to as probabilistic transcripts or PTs), given a set of mismatched transcripts, T. As an intermediate step towards this goal, prior work [5] has developed techniques to merge the transcripts in T into a distribution $\Pr(\lambda|T)$ over "representative transcripts" denoted by λ. Then, we write:

$$\begin{aligned}\Pr(\pi|T) &= \sum_{\lambda} \Pr(\pi, \lambda|T) = \sum_{\lambda} \Pr(\pi|\lambda, T) \Pr(\lambda|T) \\ &\approx \max_{\lambda} \Pr(\pi|\lambda) \Pr(\lambda|T) \\ &= \max_{\lambda} \left(\frac{\Pr(\lambda|\pi)}{\Pr(\lambda)} \Pr(\pi) \right) \Pr(\lambda|T) \qquad (1)\end{aligned}$$

The terms other than $\Pr(\lambda|T)$ in Equation 1 are estimated as follows.

- $\Pr(\lambda)$ is modeled using a simple context-free prior over the letter sequences in λ.
- $\Pr(\pi)$ is modeled using a bigram phone language model, trained on a corpus of Wikipedia text in the target language, converted into phone sequences as described in Section 4.
- $\Pr(\lambda|\pi)$ is trained using the Carmel toolkit [11] as a probabilistic finite state transducer (FST) mapping phones to letters. We also allow this FST to delete phones and insert letters. The training uses representative transcripts λ and their corresponding phone transcripts π (derived from orthographic transcripts, as described in Section 4), for speech in six languages *other than the target language*. We assume that such a model approximates $\Pr(\lambda|\pi)$ for the target language. Note that, while this assumption is not entirely accurate, it is necessitated by the requirement that no native transcriptions in the target language can be used in building any part of our system.

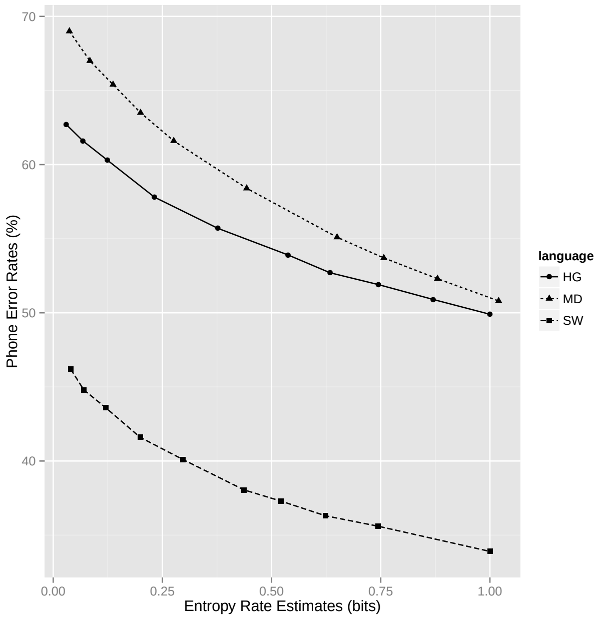

Fig. 1. Phone error rates plotted against entropy rate estimates of phone sequences in three different languages.

A crude measure of the quality of the PTs is given by the phone error rate between $\pi^* = \text{argmax}_{\pi} \Pr(\pi|T)$ and the reference phone sequences. Table 1 lists these 1-best error rates on the SBS development and evaluation sets, for all seven languages. However, the 1-best error rates do not accurately reflect the extent of information in the PTs that can be leveraged during ASR adaptation. A fuller picture is obtained by considering a collection of sequences Π that are almost as probable as π^* according to our model. Figure 1 shows the trend of phone error rates (for three languages) obtained by using collections Π of increasing size, plotted against an entropy estimate of Π. This estimate measures the average entropy of phones in the sequences in Π; e.g., 1 bit of entropy allows two equally probable choices for each phone in π. We note that the phone error rates significantly drop across all languages, staying within 1 bit of entropy per phone, illustrating the extent of information captured by the PTs.

4. MULTILINGUAL BASELINES

The goal of building a multilingual system is two-fold. One is to setup a baseline for generalizing to an unseen language without any labeled audio corpus. The other is have the baseline serve as a starting point for adaptation.

The dataset consists of 40 minutes of labeled audio for training, 10 minutes for development, 10 minutes for testing for each language. The orthographic transcriptions are converted into phonemic transcriptions in the following steps. Beginning with a list of the IPA symbols used in canonical descriptions of all seven languages, symbols appearing in only one language were each merged with a symbols differing in only one distinctive feature; this process proceeded until each phone in the universal set is represented in at least two languages. English words are identified and converted to phonemes with an English G2P trained using the CMUdict [12]. We take the canonical pronunciation of a word if the word appears in a lexicon, otherwise estimate the word's pronunciation using a G2P. The Arabic dictionary is from the Qatari Arabic Corpus [13], the Dutch dictionary is from CELEX v2 [14], the Hungarian dictionary was provided by BUT [15], the Cantonese dictionary is from I^2R, the Mandarin dictionary is from CALLHOME [16], and the Urdu and Swahili G2Ps were compiled from simple rule-based descriptions of the orthographic systems in those two languages [17].

We train a standard HMM with training data from six languages, fine-tune hyperparameters on the development set of the seventh language, and test the model on the test set of the seventh language. We assume that the lexicon of the target language is unknown, but that we are allowed to restrict the universal phone set at test time to output only phones in the target language. We also assume we have access to texts of the target language, so that we can train a phone language model on the phone sequences converted from texts. The texts are collected from Wikipedia articles linked from the main page of each language crawled once per day over four months. Results are shown in Table 2 where we compare results using universal phone set and phone language model to those obtained using language-dependent phone set and phone language model. Without a language specific phone set and phoneme language model, it is hard for a multilingual system to generalize to an unseen language. This is true even if the system has seen closely related languages such as Mandarin when tested on Cantonese.

As an oracle experiment, we also train language dependent HMMs for each individual language with 40 minutes of labeled audio. Results are shown in Table 2. It is encouraging to see significant improvement across all languages when equipped with DNN even if there are only 40 minutes of data to train the DNN.

5. SEMI-SUPERVISED BASELINES

The goal of building semi-supervised trained systems is to see how well an ASR system can measure up to the task of generating target-language transcriptions as against the crowd workers. By semi-supervised training, we mean that we train an acoustic model using a mix of a relatively large amount of untranscribed data from the target language and a small amount of transcribed data in the non-target training languages. The acoustic model is trained from scratch as opposed to performing an unsupervised adaptation to the target language.

The setup for semi-supervised baselines is the same as that described in Section 4 for multilingual training, but with an additional 5-6hrs untranscribed audio in the target language. The target language uses the same universal IPA phones as the non-target languages.

The method used for semi-supervised training is a modification of the self-training approach described in [18]. The multilingual DNN trained in Section 4 is used as a seed model to decode the unlabelled audio. The phone language model used during decoding was trained on target-language text. Lattice posteriors are used as confidence measures and only frames that have a posterior of at least 0.7 on the best path of the lattice are selected. We empirically found it better to use the posteriors as soft-targets in frame cross-entropy training. This is different from the approach in [18], which uses the best path alignment as the target. Additionally, we scaled the amount of transcribed data by 2 to create a good balance between transcribed and untranscribed data as suggested in the original work.

The results on using this DNN is shown in Table 3. Although, semi-supervised training improves PER performance over multilingual DNN, it still falls short of adaptation to probabilistic transcriptions (described in Section 6). This is in spite of the untranscribed audio data being several times larger than the probabilistic transcription data. Thus, we show that mismatch transcripts can be more effective than ASR transcription for training acoustic models.

6. ASR ADAPTATION USING PTS

As can be seen from Table 2, the multilingual baseline systems appear not to generalize well to an unseen target language. This section will detail how we improve the generalization capability of these multilingual systems to an unseen target language using mismatched probabilistic transcriptions (described in Section 3).

Our ASR framework is based on weighted finite-state transducers (WFSTs) as outlined in [19]. In this framework, the acoustic model is specified by a probabilistic mapping from acoustic signals to a sequence of discrete symbols, and a WFST H mapping these symbol sequences to triphone sequences. The other WFSTs in the framework are C which maps down triphone sequences to monophone sequences, a pronunciation model L and a language model G. Since our tasks involve phone recognition, L is essentially an identity mapping and G is a phone N-gram model.

To describe the adaptation process, it will be helpful to compare the following two cases.

- In training the parameters of the baseline acoustic model, for each training utterance, we work with the cascade $H \circ C \circ L \circ T$, where T is a linear chain FST representing the training transcript. The multilingual baselines described in Section 4 are trained in this manner using training data from languages other than the target language.
- During adaptation, for each training utterance (in the target language), we work with the cascade $H \circ C \circ L \circ PT$, where PT is a WFST representing the probabilistic transcript, obtained as in Section 3.

As noted in Figure 1, a PT contains significant amount of information beyond any single transcript extracted from the PT. Motivated by this, the statistics for the MAP estimation are accumulated from a lattice derived from the cascade $H \circ C \circ L \circ PT$.

6.1. MAP estimation of the acoustic model

The Bayesian framework for maximum a posteriori (MAP) estimation has been widely applied to GMM and HMM parameter estimation problems such as parameter smoothing and speaker adaptation [20].

Formally, for an unseen target language, we denote its acoustic observations $\mathbf{x} = (x_1, \ldots, x_T)$, and its acoustic model parameter set as λ, then the MAP parameters are defined as:

$$\lambda_{\text{MAP}} = \operatorname*{argmax}_{\lambda} \Pr(\lambda|\mathbf{x}) = \operatorname*{argmax}_{\lambda} \Pr(\mathbf{x}|\lambda) \Pr(\lambda) \qquad (2)$$

where we use multilingual baseline GMM-HMM parameters to assign the conjugate prior hyperparameters in $p(\lambda)$, and take the modes of the prior distributions as the initial model parameter estimates. Using suitable models for these distributions, [20] derive update

target language	CA	HG	MD	SW
multilingual GMM-HMM (universal)	79.64 (79.83)	77.13 (77.85)	83.28 (82.12)	82.99 (81.86)
multilingual DNN-HMM (universal)	78.62 (77.58)	75.98 (76.44)	81.86 (80.47)	82.30 (81.18)
multilingual GMM-HMM (language specific)	68.40 (68.35)	68.62 (66.90)	71.30 (68.66)	63.04 (64.73)
multilingual DNN-HMM (language specific)	66.54 (65.28)	66.08 (66.58)	65.77 (64.80)	64.75 (65.04)
monolingual GMM-HMM	32.77 (34.61)	39.58 (39.77)	32.21 (26.92)	35.33 (46.51)
monolingual DNN-HMM	27.67 (28.88)	35.87 (36.58)	27.80 (23.96)	34.98 (41.47)

Table 2. PERs of unadapted multilingual systems on the evaluation sets along with monolingual systems. PERs on the development sets are in parentheses.

Language	Multilingual	Semi-supervised	Mult-L + PT adaptation		
Code	(MULT-L)	(SS)	(PT-ADAPT)	% Rel. redn over MULT-L	% Rel. redn over SS
CA	68.40 (68.35)	63.79 (62.46)	**57.20 (56.57)**	16.4 (17.1)	10.3 (9.2)
HG	68.62 (66.90)	63.53 (63.50)	**56.98 (57.26)**	16.9 (14.3)	10.2 (9.9)
MD	71.30 (68.66)	64.90 (64.00)	**58.21 (57.85)**	18.4 (15.7)	10.3 (9.7)
SW	63.04 (64.73)	58.76 (59.81)	**44.31 (48.88)**	29.6 (24.6)	24.7 (18.4)

Table 3. PERs on the evaluation and development sets (latter within parentheses) before and after adaptation with PTs.

rules in an EM algorithm for computing λ_{MAP}. For example, the mean μ_{ik} of the GMM mixture component k associated with HMM state i is updated as:

$$\tilde{\mu}_{ik} = \frac{\tau_{ik}\mu_{ik} + \alpha_{ik}\hat{\mu}_{ik}}{\tau_{ik} + \alpha_{ik}} \tag{3}$$

$$\alpha_{ik} = \sum_{t=1}^{T} c_{ikt} \qquad \hat{\mu}_{ik} = \frac{\sum_{t=1}^{T} c_{ikt} x_t}{\sum_{t=1}^{T} c_{ikt}}$$

where τ_{ik} is a hyperparameter in the prior density for the mixture component k of state i and c_{ikt} denotes the probability of the HMM being in state i with mixture component k given observation x_t (estimated using statistics accumulated from the cascade $H \circ C \circ L \circ PT$). In our setting, the initial value of μ_{ik} is obtained from the multilingual baseline model, and $\tilde{\mu}_{ik}$ eventually converges to a model for the target language data.

6.2. Implementation details

The baseline and the adapted models were implemented using Kaldi [21]. In order to efficiently carry out the required operations on the cascade $H \circ C \circ L \circ PT$, we carefully design PT. PT is an acceptor defined as $\mathrm{proj}_{\mathrm{input}}(\widehat{PT})$ where $\widehat{PT}$ is a WFST mapping phone sequences to English letter sequences obtained as a cascade of WFSTs modeling the distributions shown in Equation 1 and $\mathrm{proj}_{\mathrm{input}}$ refers to projecting onto the input labels. For the purposes of computational efficiency, the cascade for $\widehat{PT}$ includes an additional WFST restricting the number of consecutive deletions of phones and insertions of letters (to a maximum of 3 in our experiments). We use two additional disambiguation symbols [19], apart from the ones used in typical Kaldi recipes, to determinize these insertions and deletions in $\widehat{PT}$. MAP adaptation for acoustic model was carried out for a number of iterations (12 for CA & MD, 14 for HG & SW, with a re-alignment stage in iteration 10).

6.3. Experimental results

Table 3 presents phone error rates (PERs) on the evaluation (and development) sets for four different languages. MULT-L corresponds to the multilingual GMM-HMM baseline error rates reproduced from Table 2 and SS refers to the DNN-HMM multilingual baselines adapted with untranscribed audio in the target language. We observe a consistent drop in error rates moving from MULT-L to SS.

PT-ADAPT corresponds to PERs from the multilingual GMM-HMM systems adapted to mismatched transcriptions from the target language. We observe substantial PER improvements using PT-ADAPT over MULT-L across all four languages. We also find that PT adaptation consistently outperforms the SS systems for all four languages. (The relative reductions in PER compared to both baselines are listed in the last two columns.) This suggests that adaptation with PTs is providing more information than that obtained by model self-training alone. It is also interesting that we obtain significantly larger PER improvements with PTs for Swahili compared to the other three languages. We conjecture this may be partly because Swahili's orthography is based on the Roman alphabet unlike the other three languages. Since the mismatched transcripts also used the Roman alphabet, the PTs derived from them may more closely resemble the native Swahili transcriptions (from which the phonetic transcriptions are derived).

7. CONCLUSIONS

In this work, we demonstrate the utility of mismatched transcriptions in significantly improving ASR systems for different target languages. A relative reduction of phone error rate of up to 25% (for Swahili) is observed on adapting baseline ASR systems using mismatched transcriptions. Similar impact is shown for languages from different language families and containing sounds with distinctive phonological properties.

8. ACKNOWLEDGMENTS

The work reported here was started at JSALT 2015 in UW, Seattle, and was supported by JHU via grants from NSF (IIS), DARPA (LORELEI), Google, Microsoft, Amazon, Mitsubishi Electric, and MERL. The authors also thank Amit Das at the University of Illinois for his help with preprocessing the SBS audio podcasts.

9. REFERENCES

[1] Laurent Besacier, Etienne Barnard, Alexey Karpov, and Tanja Schultz, "Automatic speech recognition for under-resourced languages: A survey," *Speech Communication*, vol. 56, pp. 85–100, 2014.

[2] Scott Novotney and Chris Callison-Burch, "Cheap, fast and good enough: Automatic speech recognition with non-expert transcription," in *Proceedings of NAACL HLT*, 2010.

[3] Maxine Eskenazi, Gina-Anne Levow, Helen Meng, Gabriel Parent, and David Suendermann, *Crowdsourcing for Speech Processing: Applications to Data Collection, Transcription and Assessment*, John Wiley & Sons, 2013.

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

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

[6] 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.

[7] Ngoc Thang Vu, Franziska Kraus, and Tanja Schultz, "Rapid Building of an ASR System for Under-Resourced Languages Based on Multilingual Unsupervised Training," in *Proceedings of Interspeech*. Citeseer, 2011.

[8] 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.

[9] "Special Broadcasting Services Australia," `http://www.sbs.com.au/yourlanguage`.

[10] "Amazon Mechanical Turk," `http://www.mturk.com`.

[11] "Carmel Finite-State Toolkit," `http://www.isi.edu/licensed-sw/carmel/`.

[12] Kevin Lenzo, "The CMU pronouncing dictionary," Downloaded 9/24/2015 from http://www.speech.cs.cmu.edu/cgi-bin/cmudict, 2015.

[13] 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.

[14] R Baayen, R Piepenbrock, and L Gulikers, "CELEX2," Tech. Rep. LDC96L14, Linguistic Data Consortium, 1996.

[15] František Grézl, Martin Karafiaát, and Karel Veselý, "Adaptation of multilingual stacked bottle-neck neural network structure for new language," in *Proc. ICASSP*, 2014, pp. 7704–7708.

[16] Alexandra Canavan and George Zipperlen, "CALLFRIEND egyptian arabic," Tech. Rep. LDC96S49, Linguistic Data Consortium, 1996.

[17] Mark Hasegawa-Johnson, "WS15 dictionary data," Downloaded 9/24/2015 from http://isle.illinois.edu/sst/data/dict, 2015.

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

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

[20] 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*, vol. 2, no. 2, pp. 291–298, 1994.

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