Automatic speech recognition using probabilistic transcriptions in Swahili, Amharic, and Dinka

Amit Das⋆*, Preethi Jyothi*⋆*, Mark Hasegawa-Johnson*

Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign
Illinois, IL 61801, USA
{amitdas, pjyothi, jhasegaw}@illinois.edu

Abstract

In this study, we develop automatic speech recognition systems for three sub-Saharan African languages using probabilistic transcriptions collected from crowd workers who neither speak nor have any familiarity with the African languages. There is a language mismatch in this scenario. More specifically, utterances spoken in African languages were transcribed by crowd workers who were mostly native speakers of English. Due to this, such transcriptions are highly prone to inaccuracies in labels. The three African languages in consideration are Swahili, Amharic, and Dinka. First, we use a recently introduced technique called mismatched crowdsourcing which processes the raw crowd transcriptions through merging, contextual weighting, and ranking. Next, we adapt multilingual hybrid HMM-DNN systems using the probabilistic transcriptions of the African languages. We also explore the effect of adaptation using bottleneck features. Finally, we report the results using both deterministic and probabilistic phone error rates. Automatic speech recognition systems developed using this recipe are particularly useful for low resource languages where there is limited access to linguisitc resources and/or transcribers in the native language.

Index Terms: mismatched crowdsourcing, cross-lingual speech recognition, deep neural networks, African languages

1. Introduction

This work is focussed on knowledge transfer from multilingual data collected from a set of source (train) languages to a target (test) language which is not a part of the set of source languages. More specifically, we assume we have easy access to native transcripts in the source languages but not in the target language. However, non-native transcripts for the target language can easily be obtained from crowd workers available on online sources like Amazon's Mechanical Turk or Upwork. An automatic speech recognition (ASR) system trained using non-native transcipts in the target language is particularly useful for low-resourced languages in Africa where it is difficult to find native transcribers but relatively easier to find non-native crowd workers.

We explain some terms that will be frequently used in this paper. The term "deterministic transcript" (DT) means the transcript was collected from native speakers of a language and have accurate ground truth labels (letters or words). Since there is no ambiguity in such ground truth labels, the labels are deterministic in nature. As an example, the DT for the word "cat", after converting the labels to IPA phone symbols, can be represented as shown in Fig. 1 with each arc representing a symbol and a probability value. Here, each symbol occurs with probability 1.0. On the other hand, the term "probabilistic transcript" (PT) means that the transcript was probabilistic or ambiguous in nature. Such transcripts frequently occcur when collected from crowd workers. Usually a training audio clip (in some language L) is presented to a set of crowd workers who neither speak L nor have any familiarity with it. Thus, due to their lack of knowledge about L, the labels provided by such workers are inconsistent, i.e., a given segment of speech can be transcribed by a variety of labels. This inconsistency can be modeled as a probability mass function (pmf) over the set of labels transcribed by crowd workers. Such a pmf can be graphically represented by a confusion network as shown in Fig. 2. Unlike the DT in Fig. 1 which has a single sequence of symbols, the PT has 3×4×3×4 = 144 possible sequences one of which could be the right sequence. In this case, it is "k æ ∅ t".

Collecting and processing PTs for audio data in the target language L from crowd workers who do not understand L is called *mismatched crowdsourcing* [1]. The language L is the language we want to recognize using an automatic speech recognition (ASR) system trained using PTs. The objective of this study is to present a complete ASR training procedure to recognize African languages for which we have PTs but no DTs. The following five low resource conditions outine the nature of the data used in this study:

- PTs in Target Language: PTs in the target language L are collected from crowd workers who do not speak L.
- PTs are limited: The amount of PTs available from the crowd workers is limited to only 40 minutes of audio.
- Zero DT in Target Language: There are no DTs in L.
- DTs only in Source Languages: There are DTs from 5 other languages ($\neq L$).
- DTs are limited: The DTs are worth about 40 minutes of audio per language. Hence, the total amount of multilingual DTs available for training is 2 hours. (40 minutes/language × 5 languages = 200 minutes)
- Unsupervised data in Target Language: There are at least 5 hours of unlabeled data in L.

2. Sub-Saharan African Languages

2.1. Swahili

Swahili phonology details here

2.2. Amharic

Amharic phonology details here

[⋆] first authors

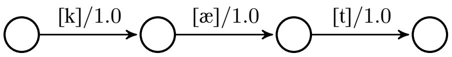

Figure 1: A deterministic transcription (DT) for the word *cat*.

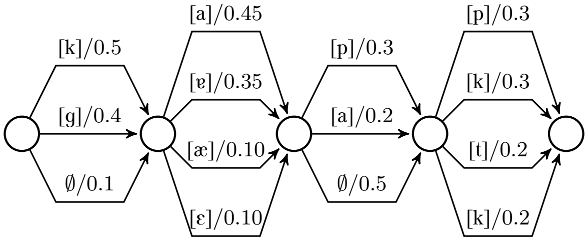

Figure 2: A probabilistic transcription (PT) for the word *cat*.

Table 1: Consonants in the Dinka language

Manner	Place				
	Lab	Den	Alv	Pal	Vel
Plosive	p b	t̪ d̪	t d	c ɟ	k g
Nasal	m	n̪	n	ɲ	ŋ
Trill			r		
Fricative					ɣ
Approx.	w			j	
Lat. Approx.			l		

2.3. Dinka

Dinka is a Western Nilotic language which is a member of the family of Nilo-Saharan languages. It is spoken by over 2 million people living in South Sudan. The four major dialects are Padang, Rek, Agar, and Bor of which the Rek dialect is considered the standard dialect of Dinka. This study is based on the Rek dialect. The Dinka orthography of Dinka closely follows its pronunciation. There are 33 alphabets in the Dinka orthography which are borrowed from a mixture of Latin and IPA alphabets [2]. Furthermore, 4 out of the 33 alphabets are digraphs. The Dinka phonolgy consists of 7 vowels and 20 consonants [3].

The set of vowels comprises of {/a/, /e/, /ɛ/, /i/, /o/, /ɔ/, /u}. Coincidentally, the orthographic symbols of these vowels are same as the phonemic symbols. The vowels often have a creaky quality. With the exception of /u/, these vowels could also have a breathy quality. For example, the breathy version of /a/ is /a̤/ and orthographically represented as ä. The breathy vowels are characterized by lower F1 values. In addition, there is relatively more energy at higher frequencies than at lower frequencies in the creaky vowels when compared with breathy vowels. Vowel lengths can be short or long. Orthographically, long vowels are usually indicated by repeating the letter twice. For example, the word *nëë* is pronounced as *ne:* .

The 20 Dinka consonants are given in Table 1. Voiced and voiceless plosives occur at five places of articulation gradually moving from external to internal portions of the mouth - labial, dental, alvelolar, palatal, and velar. Nasals follow a similar pattern. Interestingly, there is only one fricative. The 4 digraphs *dh*, *nh*, *th*, *ny* translate to /d̪/, /n̪/, /t̪/, /ɲ/ phonemes respectively.

3. ASR Steps

3.1. Data

Multilingual audio files were obtained from the Special Broadcasting Service (SBS) network which publishes multilingual radio podcasts in Australia. These data include over 1000 hours of speech in 68 languages. The following languages were used in our experiments - Swahili (SW), Amharic (AM), Dinka (DI), Hungarian (HG), Cantonese (CA), Mandarin (MD), Arabic (AR), Urdu (UR). However, only the sub-Saharan languages - SW, AM, DI - were considered as the target languages. The remaining languages were always considered as the source languages. The podcasts were not entirely homogeneous in the target language and contain utterances interspersed with segments of music and English. A simple HMM-based language identification system was used to isolate regions that correspond mostly to the target language. These long segments were then split into smaller 5-second utterances. The short length makes it easy for crowd workers to annotate the utterances since they did not understand the utterance language. More than 2500 Turkers participated in these tasks, with roughly 30% of them claiming to know only English. The remaining Turkers claimed knowing other languages such as Spanish, French, German, Japanese, and Chinese. Since English was the most common language among crowd workers, they were asked to annotate the sounds using English letters. The sequence of letters were not meant to be meaningful English words or sentences since this would be detrimental to the final performance. The important criterion was that the annotated letters represent sounds they heard from the utterances as if they were listening to nonsense syllables. PTs and DTs, worth about 1 hour of audio, were collected from crowd workers and native transcribers respectively. The training set consists of a) about 40 minutes of PTs in the target language and, b) about 40 minutes of DTs in other source languages which exclude the target language. The development and test sets were worth 10 minutes each.

To accumulate the PTs, each utterance was transcribed by 10 distinct Turkers. First the letters in the transcripts are converted to IPA symbols using a misperception G2P model learned from the source languages. More specifically, the misperceptions of the crowd workers can be approximately learned from the letter to phone mappings where the letter sequences were obtained from PTs and the phone sequences were obtained from corresponding DTs in the source languages. The target language is excluded while learning the misperception G2P model since the assumption is that there are no DTs in the target language. To remove the most erroneous transcripts, each symbol in a transcript was assigned a score which is the sum of context independent agreements and context dependent agreements with other transcrips. Following this, the multiple transcripts are merged using a ROVER technique applied on equivalence classes (symbols belonging to the same class). More details of these steps are given in [1].

To accumulate the DTs, the same set of utterances were labeled by native transcribers in the utterance language. This was necessary for comparing the ASR outputs against ground truth labels. For the DTs, the canonical pronunciation of a word was derived from a lexicon. If a lexicon was not available, a language specific G2P model was used.

Next, language dependent phones were merged to a compact multilingual phone set to enable data sharing across languages. Language specific diacritics such as tones and stress markers tend to make the phone symbols unique to a particular language. Therefore, diacritics were removed.

Swahili: phone merging goes here

Amharic: phone merging goes here

With a few exceptions for which we had to find approximate maps, most Dinka vowels and consonants were already a part of the multilingual set. Since breathy vowels are very specific to Dinka, all breathy vowels were mapped down to the regular vowels. For example, a̤ → a. The long vowels ɛː and oː were mapped by repeating the symbols twice: ɛː → ɛɛ, oː → oo. Finally, the dental nasal was mapped to the alveolar nasal: n̪ →

Table 2: SBS Multilingual Corpus.

Language	Utterances		Phones
	Train	Test	
Swahili (SW)	463	123	53
Amharic (AM)	516	127	37
Dinka (DI)	248	53	27
Hungarian (HG)	459	117	70
Cantonese (CA)	544	148	37
Mandarin (MD)	467	113	57
Arabic (AR)	468	112	51
Urdu (UR)	385	94	45
All	-	-	82

Table 3: PERs of monolingual HMM and DNN models. Dev set in parentheses.

Lang	PER (%)	
	HMM	DNN
SW	35.63 (47.00)	34.18 (39.49)
HG	?? (??)	?? (??)
MD	?? (??)	?? (??)

n.

Finally, phone based language models (LMs) for Swahili were built from the text in Wikipedia. For Amharic and Dinka, phone LMs were built from the DTs although these could also be built from the web. In all experiments, target language phone LMs are always used. The corpus is summarized in Table 2. Each utterance contains about 5 seconds of real speech data. Duration of pauses were not counted into 5 seconds.

3.2. Monolingual HMM and DNN

We first built the monolingual HMM and DNN models trained using DTs in the target language. This is an oracle baseline since it assumes DTs were available during training time. This baseline can be used to estimate the best possible (lower bound) PER. Context-dependent GMM-HMM acoustic models were trained using 39-dimensional MFCC features which include the delta and acceleration coefficients. Temporal context was included by splicing 7 successive 13-dimensional MFCC vectors (current +/- 3 frames) into a high dimensional supervector and then projecting the supervector to 40 dimensions using linear discriminant analysis (LDA). Using these features, a maximum likelihood linear transform (MLLT) [4] was computed to transform the means of the existing model. The forced alignments obtained from the LDA+MLLT model were further used for speaker adaptive training (SAT) by computing feature-space maximum likelihood linear regression (fMLLR) transforms [5] per subset of speakers. The LDA+MLLT+SAT model is the final HMM model that will be simply referred to as HMM in all experiments. The forced aligned senones obtained from the HMM were treated as the ground truth labels for DNN training.

For DNN training, we start with greedy layer-wise Restricted Boltzmann Machines (RBMs) unsupervised pre-training since this leads to better initialization [6]. Then the DNNs were fine-tuned using supervised cross-entropy training. All experiments were conducted using the Kaldi toolkit [7]. The monolingual PERs over a total of about 7K-8K phones are given in Table 3. This gives us an estimate about the approximate lower bound PERs.

Table 4: PERs of multilingual HMM and DNN models. Dev set in parentheses.

Lang	PER (%)		
	HMM	DNN	# Senones
SW	65.73 (67.58)	61.17 (??)	1003
AM	?? (??)	?? (??)	??
DI	?? (??)	?? (??)	??

Table 5: PERs of self-trained DNN models trained using STs. Dev set in parentheses.

Lang	PER %
SW	60.14 (62.07)
HG	?? (??)
MD	?? (??)

3.3. Multilingual HMM and DNN

DTs from the source languages were used to train multilingual HMMs and DNNs. Since we assume zero DTs in the target language during training, the multilingual DTs exclude the DTs in the target language. The total number of output nodes in the softmax layer representing multilingual senones was around 1000. The PERs are given in Table 4. Expectedly, due to lack of DTs in the target language, the PERs are much higher than the oracle monolingual case in Table 3. Hence, the PERs in Table 4 establish the upper bound of PERs.

In all subsequent experiments, our goal is to start from the upper bound of PERs in Table 4 and attempt to approach the lower bound PERs in Table 3.

3.4. Self Training

In this self-training experiment, the multilingual DNN decodes the audio in the target language and then uses a subset of decoded labels, with high confidences, to retrain itself in the target language [8], [9]. We will refer to the high confidence decoded labels as self-training labels or transcripts (STs) since they are used to retrain the system which generated the labels. The objective of this experiment is to evaluate the efficacy of the STs vs PTs. Since we are interested in generating STs from an ASR, we ignore the PTs from crowd workers and decode the 40 minutes of audio in the training set using the multilingual DNN from Section 3.3. The results are given in Table 5. Compared to the multilingual DNN in Table 4, the improvement due to self-training is in the range 1.01%-2.20%. We determined frame confidence thresholds as 0.5 or 0.6 from the development set.

4. References

[1] P. Jyothi and M. Hasegawa-Johnson, "Transcribing continuous speech using mismatched crowdsourcing," in *Interspeech*, 2015.

[2] Dinka omniglot. [Online]. Available: http://www.omniglot.com/writing/dinka.php

[3] B. Remijsen and C. A. Manyang, "Luanyjang dinka," *Journal of the International Phonetic Association*, vol. 39, no. 1, pp. 113–124, 2009.

[4] R. Gopinath, "Maximum likelihood modeling with Gaussian distributions for classification," in *ICASSP*, 1998, pp. 661–664.

[5] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language.*, vol. 12, pp. 75–98, 1997.

[6] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, "Greedy layer-wise training of deep networks," in *Adv. in Neural Information Processing Systems*, 2006.

[7] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlíček, Y. Qian, P. Schwarz, J. Silovský, G. Stemmer, and K. Veselý, "The Kaldi speech recognition toolkit," in *IEEE ASRU Workshop.*, 2011.

[8] K. Knill, M. J. F. Gales, A. Ragni, and S. Rath, "Language independent and unsupervised acoustic models for speech recognition and keyword spotting," in *Interspeech*, 2014.

[9] K. Vesely, M. Hannemann, and L. Burget, "Semi-supervised training of deep neural networks," in *IEEE ASRU Workshop.*, 2013, pp. 267–272.