Layperson-Trained Speech Recognition for Resource Scarce Languages

Student	Advisor	Co-Advisor
Fang Qiao	**Roni Rosenfeld**	**Jahanzeb Sherwani**
School of Computer Science	School of Computer Science	School of Computer Science
Carnegie Mellon University	Carnegie Mellon University	Carnegie Mellon University
Pittsburgh, PA 15213	Pittsburgh, PA 15213	Pittsburgh, PA 15213
fqiao@andrew.cmu.edu	*roni@cs.cmu.edu*	*jahanzeb@gmail.com*

The following is an extended abstract of the thesis on developing a technique to build speech recognizers for target languages with limited training data.

1. INTRODUCTION

Among the approximately 7000 living languages used in the world today, only a tiny fraction has been incorporated into commercial and open source speech engines, due to factors such as small potential market of the products and low availability of resources to develop them. Commercial packages like the Microsoft Speech Server (MSS) provides high quality recognizers for the world's most commonly used languages and dialects. Open source recognition engines like the CMU Sphinx allow the training language models in any languages, but with the premise that the trainer has much knowledge about how speech technology works.

Studies in recent years have pointed to both the significance and potential benefits of developing speech technologies for areas of the developing world [4, 5 and 6]. At the same time, a growing understanding of the work has lead to identifications of major road blocks in our effort. In particular, high-quality automatic speech recognition (ASR) is an essential part of voice user interfaces (VUI) such as ones in various spoken dialogue systems (SDS). But a good ASR for any language requires resources such as audio data, linguists, and technical experts; and the process of building the ASR takes time and money – all of which are unreasonable requirements in rural areas where much of the populations are low-literate.

In this study I will focus on developing practical methods for creating speech recognizers for the resource-scarce languages where many of the speakers are illiterate. Specifically, our design will focus on achieving following goals in building high-quality speech recognizers:

- The technique should work for any language.
- The technologies utilized must be cost-efficient.
- The resulting speech recognizers should be fit for use by low-literate people.
- The speech recognizers should have high accuracy over small vocabularies
- The speech recognizer built with an implementation of the technique should not require either linguistic or technical expertise.

2. RELATED WORK

We find solutions to similar problems in known research, but few provide a comprehensive solution for developing low-cost technologies with limited resources. The Globalphone project [7] aimed at building a large database of acoustic models of languages for which we do have sufficient data, thus making it possible to compensate for the lack of data in the target languages. However, to develop a speech

recognizer using the products of Globalphone project we will need linguistic and technical expertise for a target language. On the other hand, there is a method using a joint Viterbi algorithm to automatically decode audios of the target language and determine pronunciations [9] but it requires low-level access to a speech engine, which excludes the prospects of utilizing high quality commercial products such as MSS as the baseline for new recognizers. Currently the working solution that addresses the problems most comprehensively is the Salaam approach, discussed in section 2.

Our method is a strongly improved version of the Speech-based Automated Learning of Accent and Articulation Mapping (Salaam) by Sherwani [2], which is in turn first introduced as the "Poor Man's Speech Recognizer" in previous studies by Sherwani et al. on speech technology for low literate user [8].

The published Salaam approach is aimed at building small-vocabulary recognizers by transcribing the pronunciation a word in the target language into phonemes in the source language, using a well-trained speech recognizer as an underlying recognizer. Specifically, cross-language phoneme mapping using existing recognizers is used as the work around to avoid training acoustic models relying on bountiful data. Moreover, baseline recognizer was used to semi-automatically decode training data of the target language to help with obtaining more accurate pronunciations, which will improve upon that provided by a human expert.

Several existing research has tested all or parts of the Salaam approach. Sherwani's own test of the method at an international conference yielded less than 10% word error rate (WER) on various languages, with vocabulary sizes from 3 to 10 words. A comparative study on voice interfaces in Rural India [1] has attained less than 6% WER over a small domain with a system using recognizers trained on another language. Another project conducted by the Meraka Institued in African test a few recognizers build atop recognizers trained on other languages [3], and received varied results from as high as over 90% to as low as just over 50% recognition rate. Both studies showed that the Salaam approach can reach promising performance but understandably falls short compared to recognizers trained directly using resources from the target language.

3. METHOD DESIGN

3.1 Cross-Language Phoneme Mapping

We incorporate a cross-language phoneme mapping approach as described in the published Salaam method without much variation.

Using an existing speech recognition system, cross-language phoneme mapping can be done by defining the word or phrase using a sequence of phonemes that are defined in the system, which in turn represents an entry in a lexicon file. For example, the standard North American pronunciation of the word "long" in phonemes of MSS U.S. English recognizer would be similar to "L AO NG."

A clear drawback of this approach is of course that the set of phonemes in the source language and target language are most likely not the same. For instance, the Hebrew word for one, "אחד", has a uvular fricative phoneme that sounds like a mix between an "H" sound and the "K" sound in English. In such cases, we pick the one that our baseline recognizer agrees with the most given the training samples. So with the MSS U.S. English recognizer, the resulting pronunciation would be similar to "E H AA D" or "E K AA D", or both if the implementation accepts multiple pronunciations.

3.2 Data-driven Approach

Incorporating a data-driven approach aims to help humans with the task generate a pronunciation for new words, i.e. the aforementioned cross-language transcription. The idea is largely reliant on the scoring of recognition results returned by the baseline recognizer. It follows that, if the recognizer is given a large set of potential phoneme sequences, it would pick out the ones that matches the audio input, and provide acoustic score and/or confidence score we can then use them to pick out the best pronunciations for the training data. But, trying to match even a 5 phoneme sequence creates the search space of 37^5 distinct sequences on a recognizer with 37 phonemes, making the task computationally impractical.

THE PUBLISHED SALAAM METHOD

The design described by the published Salaam methods is a semi-automatic pronunciation technique attempted to address computational complexity issue by having a linguistic expert fix down a number of phonemes that are more certain – e.g. the consonants – and then create arbitrary word boundaries in the word. The former part of this design endeavors to relieve the recognizer of problems human experts can solve; and the latter effectively makes the resulting words produced by the boundaries "separate problems." In practice, if we have a word where there are 2 phonemes the expert is uncertain of, one can place the word boundary somewhere between the two phonemes, method will match each separate word with set sequences whose size is equal to or less than the all that is in the recognizer. So in general, if there are N phonemes in the recognizer and there are n uncertain phonemes, the complexity of the search can be effectively reduced to $O(nN)$.

To eliminate the need for human linguistic experts, the original Salaam method suggested using a speech recognizer's letter-to-sound rules to generate the initial pronunciations with the help of a foreign word expressed in the English alphabets (much like typing on instant messaging or SMS text messages).

IMPROVING ON THE SALAAM METHOD

One of the goals of our work on the data-driven approach is to make it fully automatic and thus implicitly eliminate any need for human expertise. Furthermore, we have improved the existing method in the following areas:

1) The published method's reliance/assumptions on the phonemes fixed by the expert, and on the total number of phonemes in the target pronunciation.
2) The reliance on word boundaries to reduce computational complexity; which may have unaddressed negative effects on the recognition results from the baseline recognizer, and hence on the pronunciation produced.

Removing all human expertise implies that the baseline recognizer must be used to generate the entire phoneme sequence. To do this, we must look at some subsets of all possible phoneme sequences/decoding, and take the ones that the recognizer agrees with the most given the audio data of the target word. But as pointed out before, the set of potential phoneme sequences cannot be too large. So due to computing limitation we still need to use word boundaries, albeit in a different manner, to cut down on the size of the search space.

We designed an iterative algorithm that progressively generates phonemes resulting in a sequence that has been given a relatively high score by the underlying recognizer. To start, we explicitly enumerate all possible phoneme sequences for the first 1 through 3 phonemes for the recognizer to match on the training audio, and then try to match a number of single phonemes after the initial 1 to 3, with word boundaries separating them. A graphical representation of this would be:

"* / * / * / * / *..." or "* * / * / * / * / *..." or "* * * / * / * / * / *..."

Where the "*" denotes any phoneme, and "/" denotes the word boundaries. For the sake of easy representation, we write this setup as "Phx[1-3] / Ph[n]", where "Phx[1-3]" is the explicitly enumerated 1 through 3 phoneme sequences, and "Ph[n]" is from 0 up to n word boundary separated phonemes we match at the end. As we can see, when the recognizer decodes the audio data by evaluating one of the three cases: the first phoneme by itself (separated by a word boundary from the rest), or along with another phoneme, or along with two more phonemes; in the latter two cases, the resulting first phoneme recognized should be less influenced by the first word boundary. We accept the very first phonemes from recognition results as the potential first phoneme in the final pronunciation. Note we do not just take the phonemes with the highest scores because there are still boundaries in the target sequence; so a phoneme from a recognition result with low score may in fact be a part of a high-score pronunciation that is without word boundaries.

In the second iteration, we concatenate each phoneme obtained from iteration one to the beginning of our "Phx[1-3] / Ph[n]" rule. So if the first iteration gave produced the set P of m phonemes, we will have m "X Phx[1-3] / Ph[n]" sequences to match in this step for each phoneme "X" in P. With the same logic as the first iteration, we can now accept the very first 2 phonemes from each recognition result as the potential first and second phonemes in the final pronunciation.

Likewise, at iteration i, we should have a set P_{i-1} of phoneme sequences each consisting of $i-1$ phonemes from iteration $i-1$. For each phoneme sequence "X[$i-1$]" in P_{i-1}, we will match "X[$i-1$] Phx[1-3] / Ph[n]" in the current iteration. The algorithm stops when on an iteration j, we no longer produce any more phonemes from Phx[1-3] / Ph[n], i.e. the recognition results are no longer than $j-1$ phonemes. Then we can accept the "X[$j-1$]" phoneme sequences that have the highest scores given by the recognizer as the final pronunciations.

The complexity in the search space for each sequence in the set P_{i-1}, using a recognizer with N built-in phonemes, is $N+N^2+N^3+nN$. We can further limit size of P_{i-1} to be M, and upper-bound the length of the final pronunciation of any word to be L, then the overall complexity of pronunciation generated for that word is $O(N^3ML+nNML)$.

4. EXPERIMENTATION AND RESULTS

To test the new Salaam method, we have collected data for various languages from different speakers. The tests are then divided up into training sets and testing sets to evaluate the performance of our system.

4.1 Data Collection

We have compiled a list of 50 words, consisting of numbers, short commands to computers or SDS, disease names etc. A voice contributor (usually the first contributor for a particular language) is asked to translate the words into the language whose data he/she is to provide. If multiple translations are possible, the contributor was asked to take the most commonly used expression. When the word in the respective language is exactly the same as in English (e.g. AIDS), the contributor is asked to either say the word in English or skip it altogether.

The recording is done at a quiet location, either through landline telephone or cell phones, for those are the medium we expect the recognizers to be implemented for in developing countries. For this purpose we have developed a SDS in VoiceXML using Voxel for the contributors to call in and input their voice.

Each contributor is asked to readout the translated words in order, for 5 iterations, this is done so each repetition of the same word has minimum effect on the next time it is spoken. Hence we obtain 5 audio samples per word per speaker.

4.2 Evaluation and Results

We first looked at single-speaker results with varying vocabulary sizes. We have the 5-fold cross-validate results for 41 words of Yoruba and 50 words of Hebrew. Furthermore, the resulting recognition accuracy are compared with the recognition results for the same word type using pronunciations manually written down by the contributor of that language. (See Figure 4.1)

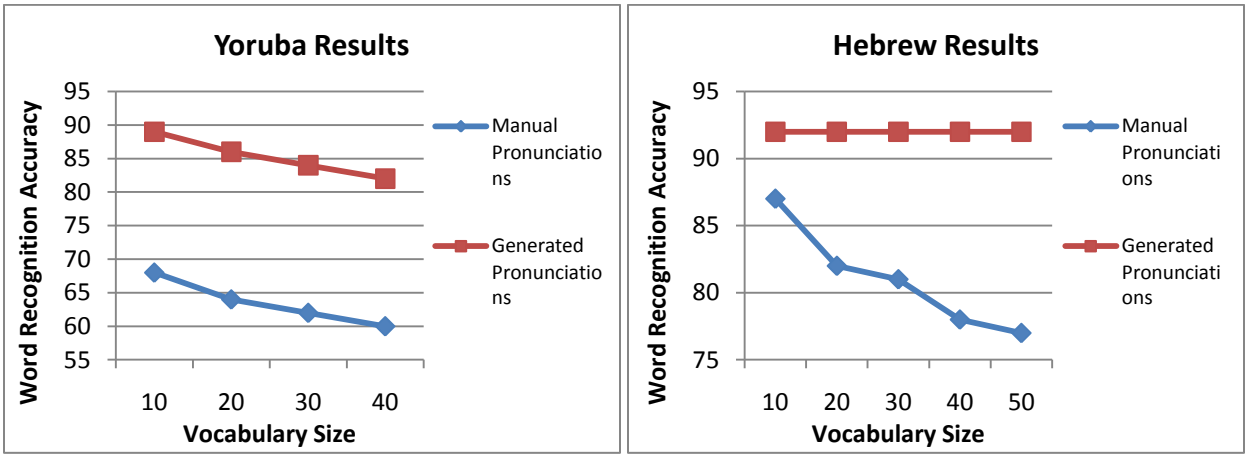

Figure 4.1 Recognition result of Yoruba and Hebrew for both manual and automatically generated pronunciations with varying vocabulary size.

For results from data sets of both Yoruba and Hebrew, we see that pronunciations generated by our approach readily beat the pronunciations provided by human.

Currently, we are looking at results from cross-speaker tests of the pronunciations generated by our method from three sets of Hebrew data. The pronunciation trained from each speaker is tested on the two others, and plotted cross different vocabulary sizes.

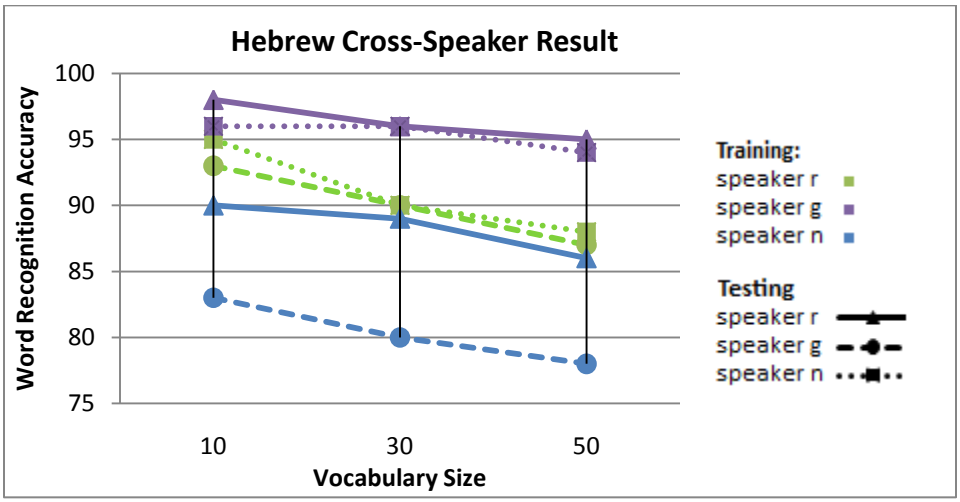

Figure 4.2 Cross-Speaker results for Hebrew with varying vocabulary size.

One surprising observation in the cross-speaker results is the correlation of the source of training data with recognition accuracy. While the pronunciations trained on speaker g worked fantastically, those trained with data from speaker r and n did not do very well, with the performance pronunciation from speaker n varying greatly between the two test data. We look forward to see cross-speaker results by training from two speakers and testing on one.

CONCLUSION

Current results present empirical confirmation that our method can achieve high recognition accuracy over a small vocabulary for a language without any involvement of human experts or reliance on sizable language resources. Pronunciations in the source language generated by our algorithm consistently outperform those provided by linguistic experts, hence proving our method a potent way to apply cross-language phoneme mapping when training data in a target language is lacking.

Although we only have results from two different languages, these languages come from two different areas and belong to two distinct language families, the Afroasiatic languages (Hebrew) and the Niger-Congo languages (Yoruba); and the method yielded satisfactory results for both. Currently the tests on other languages are being conducted, and results will be presented in the future.

As per our description of the method's design in section 3.2, implementation of our method should not entail low-level modifications to a recognizer of the source language – our design could potentially be implemented using any recognizer, even closed-sourced ones. It could be an interesting research in the future to test the method's effectiveness/performance when implemented with different base-line recognizers.

REFERENCES

[1] Patel, N., Agarwal, S., Rajput, N., Nanavati, A., Dave, P. & Parikh, T. *A Comparative Study of Speech and Dialed Input Voice Interfaces in Rural India.* ACM CHI 2009.

[2] J. Sherwani. 2009. Speech Interface for Information Access by Low Literate Users. Thesis. Docter of Philosophy (PhD). Department of Computer Science. Carnegie Mellon University.

[3] Van Heerden C., Barnard E. and Davel M., *Basic speech recognition for spoken dialogues*, In Proc. of the 10th Annual Conference of the International Speech Communication Association (Interspeech 2009), pp 3003-3006, September 2009.

[4] J. Sherwani, N. Ali, S. Mirza, A. Fatma, Y. Memon, M. Karim, R. Tongia, and R. Rosenfeld. *Healthline: Speech-based Access to Health Information by Low-literate Users.* In Proc. IEEE/ACM Int'l Conference on Information and Communication Technologies and Development, pp131-139, Bangalore, India, December 2007.

[5] Plauche, M., Nallasamy, U., Pal, J., Wooters, C., & Ramachandran, D. (2006). *Speech Recognition for Illiterate Access to Information and Technology.* Proc. 115 International Conference on Information and Communications Technologies and Development, 2006.

[6] Measuring the Information Society: The ICT Development Index. http://www.itu.int/ITU-D/ict/publications/idi/2009/index.html. Accessed March 25, 2009.

[7] Schultz, T. & Waibel, A. (1998). Adaptation of Pronunciation Dictionaries for Recognition of Unseen Languages. Workshop on Speech and Communication (SPECOM-1998), pp 207-210, St. Petersburg, Russia, October 1998.

[8] J. Sherwani S. Palijo, S. Miraza, T. Ahmed, N. Ali, and R. Rosenfeld. *Speech vs. Touch-tone: Telephony Interface for Information Accesses by Low Literate Users.* In Proc. IEEE Int. Conf. n ICTD, pp.447-457, Doha, Qatar, 2009.

[9] Bansal, D., Nair, N., Singh, R. & Raj, B. *A Joint Decoding Algorithm for Multiple-Example-Based addition of Words to a Pronunciation Lexicon*, Proc. ICASSP 2009.