lex4all: A language-independent tool for building and evaluating pronunciation lexicons for small-vocabulary speech recognition

Anjana Vakil and Max Paulus
Department of Computational Linguistics, University of Saarland

INTRODUCTION

This poster presents *lex4all,* an easy-to-use PC application that allows even non-expert users to quickly and easily create pronunciation lexicons for words in any low-resource language (LRL), using:

- a small number of audio recordings
- a pre-existing recognition engine in a high-resource language (HRL)

The resulting lexicon can then be used to add small-vocabulary speech recognition functionality to applications in the LRL.

BACKGROUND & GOAL

Speech recognition interfaces can be extremely beneficial for applications in the developing world, particularly in communities where literacy rates are low or where PCs/internet connections are not always available [1, 2].

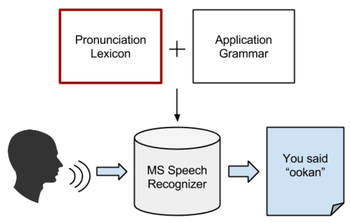
Unfortunately, large speech corpora are simply not available for the LRLs spoken in such communities. Such data is, however, essential for the training of acoustic models in speech recognizers.

However, for small-vocabulary applications (requiring recognition of a few dozen terms), we can use an existing recognizer trained for a HRL (e.g. the American English recognition engine of the Microsoft Speech Platform [3]) to accomplish recognition.

The idea is to feed the engine with a pronunciation lexicon mapping each term in the target vocabulary to one or more sequences of phonemes in the HRL, i.e. phonemes which the recognizer can model.

The recognition task requires:

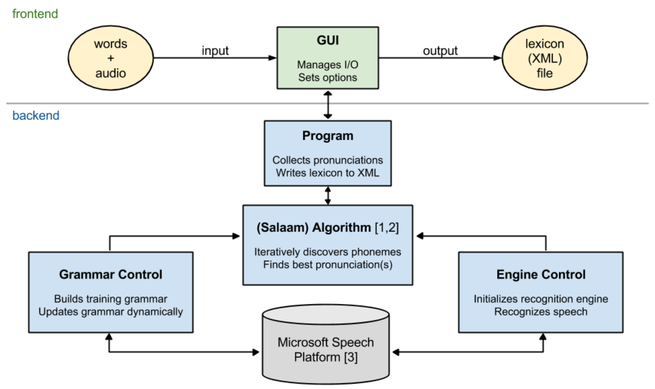
- audio input in the target LRL
- application-specific grammar
- lexicon for terms in grammar
- recognition engine for the HRL



Given a tool for automatically creating such lexicons, small-scale developers could add speech interfaces to applications in any language without large audio collections or expertise in speech technology.

This is the motivation behind lex4all, a desktop application for Windows based on the Microsoft Speech Platform [3] and the Salaam algorithm for pronunciation mapping [1, 2] (see "Algorithm").

SYSTEM OVERVIEW



ALGORITHM

We use the Salaam method [1, 2] for the automatic discovery of the best pronunciation sequence for each word in the target vocabulary.

The Salaam method [1, 2]:

- *"Super-wildcard" grammar:*

  Instructs the recognizer to treat each audio sample as a "phrase" consisting of 0-10 "words", where each "word" is a sequence of 1-3 source-language phonemes, i.e.:

  $$\{*|**|***\}_0^{10}$$

  where * represents a single phoneme of the source language.

- *Iterative training algorithm*

  Uses this grammar and the HRL recognizer to discover the best pronunciation sequence(s) for each word in the target vocabulary, one phoneme at a time

- Yields more accurate recognition than expert-written pronunciations[1]

ADDITIONAL FEATURES

- **Discriminative training** [2]

  An additional training step removes pronunciations in the lexicon that may reduce recognition accuracy by matching multiple words in the vocabulary

- **Evaluation module**

  Facilitates research by automatically simulating recognition on a test set of audio samples. Reports recognition accuracy rates and confusion matrix.

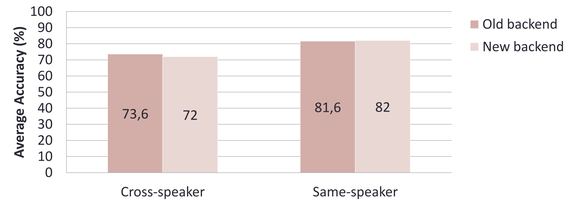
- **Built-in audio recorder**

CHALLENGE: RUNNING TIME

The main challenge we faced in engineering a user-friendly application based on the Salaam algorithm (see above) was the long training time due to the large "super-wildcard" grammar required by the algorithm.

- Original backend: 1-3 phonemes per sub-word
  - 40 phonemes (English) → 64,000 possible combinations
  - Training time (25 words, 5 samples/word): approx. 60-120 minutes
- New backend: only 1 phoneme per sub-word
  - 40 phonemes → 40-line wildcard
  - Training time (25 words, 5 samples/word): approx 2-5 minutes (≈20x faster)
- Evaluation
  - Tested on Yoruba data (25 words, 2 speakers, 5 samples/word/speaker)
  - Result: no significant drop in recognition accuracy (see Figure 1)

Figure 1. Evaluation of Word Recognition Accuracy



CONCLUSION & FUTURE WORK

The lex4all tool enables the rapid and automatic creation of pronunciation lexicons in any LRL, using an out-of-the-box commercial recognizer [3] for a HRL (English) and an existing algorithm for cross-language pronunciation mapping [1, 2].

We hope that this tool will help developers create speech interfaces for applications in LRL, as well as facilitate research in small-vocabulary speech recognition for such languages.

Possible future extensions of the project include:

- **Online lexicon repository**

  Adding an option for users to upload created lexicons to an online repository would allow sharing and re-use of lexicons across languages/language families.

- **Additional source-language recognizers**

  Microsoft offers recognizers in over 20 languages [3]. Using a source language that is more similar to the target language could improve recognition accuracy.

References

[1] Fang Qiao, Jahanzeb Sherwani, and Roni Rosenfeld, 2010. "Small-vocabulary speech recognition for resource- scarce languages," in *Proceedings of the First ACM Symposium on Computing for Development (ACM DEV '10).* ACM, New York, NY, USA, pp. 3:1–3:8.
[2] Hao Yee Chan and Roni Rosenfeld, 2012. "Discriminative pronunciation learning for speech recognition for resource scarce languages," in *Proceedings of the 2nd ACM Symposium on Computing for Development (ACM DEV '12).* ACM, New York, NY, USA, pp. 12:1–12:6.
[3] Microsoft, 2012. Microsoft Speech Platform SDK 11 Documentation. http://msdn.microsoft.com/en-us/library/dd266409

Acknowledgments

Many thanks to Roni Rosenfeld, Hao Yee Chan, and Mark Qiao for generously sharing their data and providing valuable advice on implementing the Salaam method.