Android Application for Language Identification

Pedro A. Torres-Carrasquillo , Robert A. Ford and Joel C. Acevedo-Aviles
Massachusetts Institute of Technology Lincoln Laboratory
{ptorres, bobford, joel}@ll.mit.edu

Abstract

In this presentation, details about the implementation of a language identification system on a number of Android based smartphones are discussed. A description about the tradeoffs between computational complexity and performance are presented across a number of phones for both a 5-language open-set identification task and for an extended task including about 50 languages. Additionally, details about extending the current capabilities to the task of speaker identification are shown. The presentation will be accompanied by a live demonstration running on a commercially available Android phone.

Introduction

Automatic language identification (LID) is the process of determining the language being spoken in a speech utterance without human intervention. In this project two different types of LID tasks are considered: closed-set ID and open-set ID. Closed-set id refers to determining which of *N* predetermined languages is being spoken. Open-set ID adds "none of the above" as a possible outcome for cases in which the spoken language was not modeled.

In general, some possible applications of LID are language-based data filtering, pre-processing for automated speech applications such as machine translation and speech recognition, and requesting human translators for emergency situations. The focus of this project is to develop a smartphone application that addresses the latter two scenarios.

Language identification system

The language identification system (LID) employed in this work is based on a combination of classifiers whose outputs are combined to produce an identification score. The classifiers used include a Gaussian mixture model and an SVM classifier. The system is modeled after the system described in [1]. The software package used is based on technology developed in the Human Language Technology Group at MIT Lincoln Laboratory over more than 15 years and it was implemented in C++. These algorithms have shown state of the art performance on a number of international evaluations.[1,2]

System Architecture

The architecture of the LID Android application is depicted in Figure 1. Audio is captured through the smartphone built-in microphone, and passed to the LID module for processing. The LID module returns confidence scores for each of the modeled languages, a decision is made by the system and the resulting language is displayed to the user. All modules, except for the LID one, are implemented using Android's Java SDK. The LID module was implemented in C++ for legacy and efficiency reasons. The communication between the Java components and the C^{++} component is achieved through the Android's Native Development Kit (NDK).

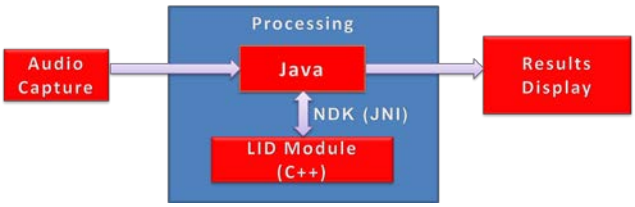

FIGURE 1. System architecture.

Performance

The initial task evaluated is a 5-class closed-identification task. The languages used for this task were: Arabic, Cantonese, English, Mandarin and Spanish. The performance of the system implemented has been measured at 4.9% classification error rate for the highest complexity system (2048-mixture order) to 15.4% for the lowest complexity one (16-mixture order), where the 16-mixture system is about an order of magnitude faster than the 2048-mixture one.

In terms of processing time, this technology has been evaluated on two different devices. The HTC Magic phone and the Samsung Nexus S. The tradeoff between model order and execution time is shown in Figure 2.

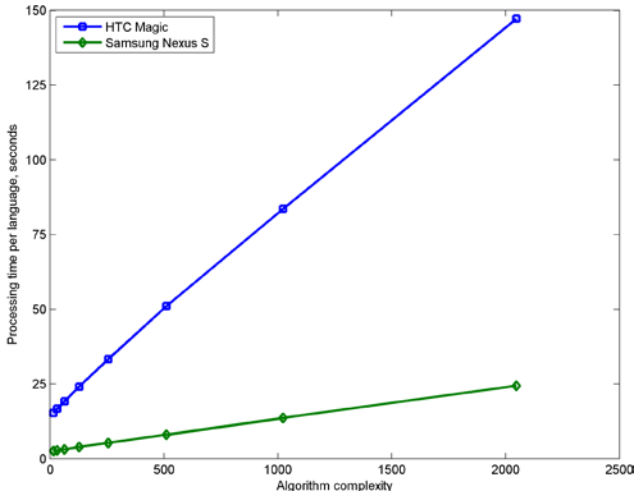

FIGURE 2. Average processing time (seconds) for different GMM systems on the HTC Magic and the Samsung Nexus S phones.

Next Steps

At the time of this writing, the following tasks have not been completed. However, they are scheduled for completion before the HPEC workshop.

- Extend task to open-set LID

- Extend the number of languages from 5 to 50
- Evaluate system performance with the addition of an SVM classifier
- Develop a GUI to support the target scenarios described in the *Introduction*
- Explore the possibility of leveraging the current implementation to the speaker identification problem.

References

[1] P. Torres-Carrasquillo, et.al, "The MITLL NIST LRE 2009 Language Recognition System", *Proceedings of International Conference on Acoustics, Speech and Signal Processing*, Dallas,TX USA, 2010.

[2] P. Torres-Carrasquillo, et.al, "The MITLL NIST LRE 2007 Language Recognition System", *Proceedings of the 10th Annual Conference of the International Speech Communication Association (InterSpeech)* , Brisbane, Australia, 2008.