Short-burst Language Learning using Speech Recognition and Crowdsourcing

Supervisor: Prof. Philip J. Guo, University of Rochester, NY *
Student: Davide Berdin, Uppsala University †

Problem Description

The major aim with this project is to develop a software tool for improving the learning methodology of a second language through the use of automated speech recognition. In particular, we aim to develop a mobile application on Android platform where the user will be able to test his/her pronunciation. The user will receive a score based on how good was the pronunciation since all the testers will compete among each other. The key point of the competition is to stimulate people to improve themselves. Psychological studies have demonstrated that competitions improve endurance performance because of an increased level of enjoyment and effort [1].

Major Goal of the study

The major goal of this project is to develop an application on Android platform. In particular, we want to exploit the "Unlock Screen" page of a smartphone in such a way that the user will test his/her pronunciation. Our main goal is to take advantage of all those moments where a user is on idle. The Unlock screen phase is only one of several possibilities where we can deliver our application; for instance, after a phone call, after he/she sent either an email or a text or even when the user is simply scrolling the set of available applications on the phone just for wasting some time. All these moments are potential frames were we can ask the user to test the pronunciation. Smartphones are not the only platform where we could deploy the application: smartwatches are becoming very interesting gadgets as well as a possible way to exploit the user's wait-time for practising. In fact, a future work could take advantage of the fact that when the user is looking at the time, a word can pop on the screen and through the integrated microphone of the smartwatch, the user can test the pronunciation.
Moreover, the application can be used in a class for helping kids as well. A typical scenario could be that while students are walking around the city during a day outdoor, the application can ask them to say some words leaving the machine the task of helping them with the pronunciation whereas the teacher can teach them the basics for pronouncing in the right way all the syllables.

Application's flow

1) The system retrieves words from the environment based on the GPS location of the smartphone (i.e. Coffee-house, supermarket, cheese, etc.)

2) The application retrieves an image from Google Images corresponding to the word the system got from previous step

3) The application will translate the word from the native language (L1) into the target language (L2) (i.e. Italian, Spanish, German, etc)

4) Both the picture and the word will appear on the screen during the "Unlock phase" (basically when the phone is blocked)

5) The user will use either the internal microphone or the headset to pronounce the word

6) A supervised learning algorithm will calculate the "error" in the pronunciation and based on this measure, the system will assign a score.

7) Also, a pool of native speakers will give a quick score on the pronunciation

* {pg@cs.rochester.edu}
† {davide.berdin.0110@student.uu.se}

8) The score obtained from the speech recognition is used to build a rank to allow the competition among the whole group of testers [1]

Tests and Evaluation

We will test the application on a group of people that have been studied the L2 and want to improve their pronunciation of "common" nouns. The use of images should help the user to improve his/her memory about the word whereas the error valued that the supervised algorithm will yield should keep the user motivated to improve the pronunciation. Also, we will use the crowdsource to retrieve a *second* score of the same word. Once we have both, a comparison between the crowd-score and the speech recognition-score will be created and analyzed.

The key point of the testing phase is to measure the performance of the pronunciation. Basically, we want to analyze the trend of how the pronunciation has improved, how fast the user learns to say words with similar syllables and how many times the user uses the application.

The candidates we are looking for should have the following characteristics:

- basic knowledge of L2
- use the application for 3 weeks
- don't get embarrassed if he/she has to pronounce a word (or more than one) in public.

Last point in particular is very important because this kind of application can be socially awkward since it may happen that the user is queuing for a coffee and wants to test the pronunciation in that moment.
The ideal number of candidates is 10. Considering an average of (roughly) 3 hours of "idle" time during the day, 3 weeks of testing per user, should provide us a consistent set of data.

Time Schedule

Task to be completed:

T1 - Design the user interface for the unlock screen widget [2]

  (a) implementation of the application on a smartphone with dummy functions
  (b) **testing** if the application doesn't interfere with the basic functions of the phone (calls, SMS, etc.)

T2 - Implementation of GPS information

  (a) retrieve location
  (b) extract words from the environment
  (c) save a set of candidate words (based on available audio files)

T3 - Implementation of the word representation through images

  (a) Implementation of the word representation through images
  (b) search the image on Google Images
  (c) download the picture

T4 - Put together

  (a) set the image on the screen
  (b) set the word on the screen

T5 - **Literature study of Speech recognition** [3] [4]

T6 - Implementation of a backend service on a server

  (a) implementation of SL Model - audio fingerprint recognition [5]
    i. We can use (need permission probably) the website `www.wordreference.com` for the audio files as "desired output" for the SL model.

[1] The users will be selected among students and faculties that want to test their pronunciation as well as willing to improve it and possibly learn some new words.

(b) Given the error, calculate the score and send it back to the user

(c) implementation of Crowdsourcing project [6]

(d) retrieve score from the crowd and keep it for comparison with speech recognition

T7 - Build the rank

(a) the system will keep track of all scores and will build the rank

(b) it will keep updated the rank so a user can see his/her position

T8 - Build the log

(a) the system will keep track of the scores

(b) the system will keep track of how many times the user uses the app

(c) the system will keep track of trend in such a way we can understand if the user is improving his/her pronunciation

(d) Other things if necessary

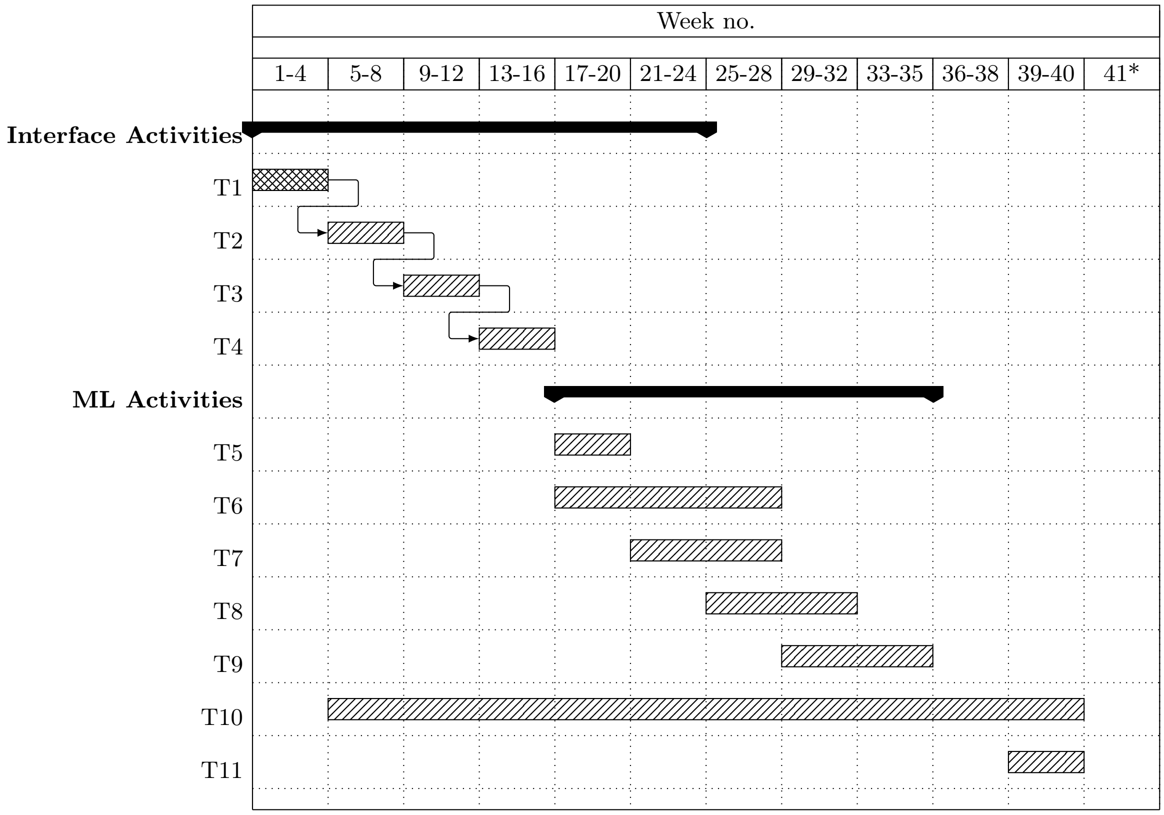
T9 - Analyze the collected data and evaluate those

T10 - Write the report (both the paper and the actual Thesis)

T11 - Final presentation

Time table

The total amount of weeks needed for the project should be 40. The table shows a rough time schedule for all the activities requested to accomplish the project.



(*) Week 41 is an extra week as backup.

References

[1] A. Cooke, M. Kavussanu, D. McIntyre, and C. Ring, “Effects of competition on endurance performance and the underlying psychological and physiological mechanisms,” *Biological psychology*, vol. 86, no. 3, pp. 370–378, 2011.

[2] “Lockscreen widgets api,” 2015. accessed 2015-05-01. Available: `http://developer.android.com/about/versions/android-4.2.html#Lockscreen`.

[3] “Speech recognition android api,” 2015. accessed 2015-05-01. Available: `http://developer.android.com/reference/android/speech/SpeechRecognizer.html`.

[4] “Speech recognition python,” 2015. accessed 2015-05-01. Available: `https://pypi.python.org/pypi/SpeechRecognition/`.

[5] “Audio fingerprint recognition in python,” 2015. accessed 2015-05-01. Available: `https://github.com/worldveil/dejavu`.

[6] “Opensource crowdsourcing framework,” 2015. accessed 2015-05-01. Available: `http://pybossa.com`.