



Short programming project - Computing Science

Course code: WBCS15002

Credits (EC): 5 points (140 hours, 14 hours/week)

1. General information

Project title: Spoken Language Identification App

Student name and signature: Robbin de Groot

Student number: s3376508

Date: 14/11/2019

First supervisor (name + signature): Kerstin Bunte

Second supervisor (name + signature):

2. Project description:

Spoken language can be described as a waveform, which in turn can be analysed by a computer system. The project is to build an App that can identify the spoken language (through microphone input).

The waveform described is to be analysed using a form of Machine Learning, specifically an artificial neural network. As such a model should be developed and trained. Therefore, training data is required.

The problems that arise in this project primarily have to do with the above mentioned procedures. Finding or developing a model that can fit the data well can be time consuming, given that training times are usually high. Finding the data to be trained is a topic of its own. Given that we work with audio files, finding data and its great numbers may not be trivial.

Many problems that arise in all forms of machine learning arise as well. Classical problems like over- and underfitting will appear and we should have a means of reducing them.

We must consider that file format as well. A high quality audio file may contain noise, especially if the source is unknown. A noisy audio file, however, should still be fit to a certain degree, as recordings from phone microphones are rarely (if not, never) noise free.

If the network is to be ported to an app, we have another constraint: file size. If the network model is too large, it may not be feasible to port it to a mobile device. This constraint on the network makes developing it even more difficult.

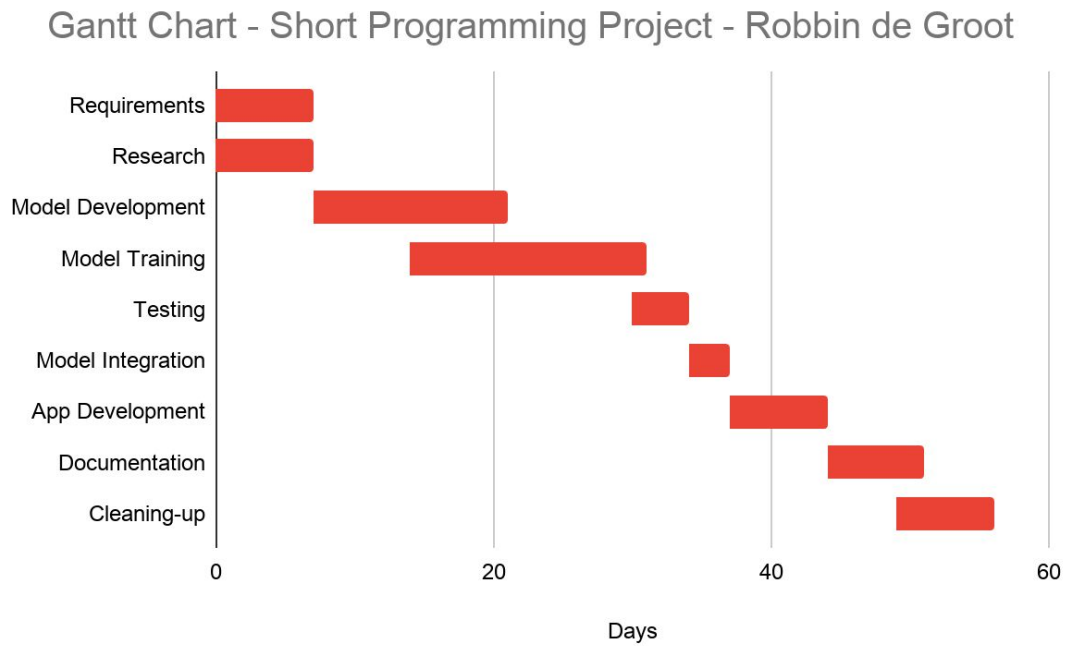
3. Methodology and timeline:

Below is a time table for expected tasks and their deadlines. These will define the deadlines associated with the project. We assume a period of 8 weeks for development. However, 10 weeks may be available. As with any project, we may run into unexpected issues, which may cause a missed deadline. As such, we have a small buffer period at the end.

This order of operations is according to the waterfall model.

Researching various relevant subjects (1 week)	Before we start the project, I need relevant knowledge in some fields of study. I also need to find suitable programming frameworks and datasets in order to be able to define a model and design an app.
Developing and training a model (2-3 weeks)	As described, training and developing of a model is time consuming. As this process can turn into days, maybe even weeks, it is of utmost importance that this starts as soon as possible.
Rigorous testing of the data model (1 week)	As with any machine learning algorithm, it needs to be tested and evaluated for performance. We may have multiple models to choose from, from which the best should be selected. This requires the definition of well-defined criteria for what makes a 'good' model.
Integrating the model into a mobile app (1 week)	Once a model has been selected and trained, it needs to be ported to a mobile host application. This application needs to be able to interact with the network and be able to run on recorded data.
App integration and design (1 week)	The app needs to be able to interact with the microphone and use its recordings. It needs to parse the output data and display it in a somewhat intuitive way. Design is not the main objective, however the app needs do have a degree of usability.
Documenting and cleaning up (1 week)	Once the build is finished, it needs to be sufficiently documented and tidied up to for final delivery.

Tentative time-line (Gantt chart with Work Packages (WP) and Milestones (M) for overlapping intervals shortly explained in a list)



Most of the Gantt chart shown above is according to the *Methodology and Timeline* section. Some topics have been split up in order to distinguish between separate actions and operations per development part.