Speech Recognition

- With Information Retrieval & Command Prompt -

Project Report

ED5-1-E17

Aalborg University
Electronics and IT

Here you can write something about which tools and software you have used for typesetting the document, running simulations and creating figures. If you do not know what to write, either leave this page blank or have a look at the colophon in some of your books.

AALBORG UNIVERSITY
STUDENT REPORT

Electronics and IT
Aalborg University
http://www.aau.dk

Title:
Speech Recognition

Theme:
Scientific Theme

Project Period:
Fall Semester 2017

Project Group:
ED5-1-E17

Participant(s):
Stefan Bîrs
Tiberiu-Ioan Szatmari
Kamran Thomas Alimagham

Supervisor(s):
Daniel Ortiz-Arroyo
D. M. Akbar Hussain

Copies: 1

Page Numbers: 21

Date of Completion:
October 30, 2017

Abstract:

This report goes into detail about the implementation of distributed speech recognition systems, information retrieval and machine learning. The main objective of the speech recognition system is to provide a handsfree human-computer interface experience, where the system is also able to handle noise.

Contents

Preface **ix**

1 Introduction **1**
1.1 Examples . . . 1

2 Problem Description **3**
2.1 Problem Description . . . 3
2.2 Problem Delimitation . . . 3

3 Speech Processing **5**
3.1 Signal processing . . . 6

4 Machine Learning for Speech **9**
4.1 TensorFlow . . . 9
4.1.1 Tensors . . . 9
4.1.2 TensorFlow Core . . . 10
4.1.3 tf.train API . . . 10
4.1.4 tf.estimator API . . . 10
4.2 Deep neural network for speech . . . 10
4.2.1 Deep Feed Forward (DFF) . . . 10
4.2.2 Recurrent Neural Nerwork (RNN) . . . 10

5 Distributed speech recognizer **11**
5.1 The DSR system . . . 11
5.2 Acoutic models . . . 11
5.3 Language model . . . 11

6 Knowledge-based IR system **13**
6.1 The document retrieval system . . . 13
6.2 The question answering system . . . 13

7 Discussion **15**

8 Conclusion **17**

Bibliography **19**

A Appendix A name **21**

Todo list

- Is it possible to add a subsubparagraph? 1
- I think that a summary of this exciting chapter should be added. 1
- The figures are on GitHub, need to figure out how they work in LaTeX . . 7
- reference this part with $https://www.allerin.com/blog/six-types-of-neural-networks$ 10
- link this part with link above 10

Preface

The project entitled Speech Recognition with Focus on information retrieval and command prompt was made by three students from the Electronics and Computer Engineering programme at Aalborg University Esbjerg, for the P5 project during the fifth semester.

Aalborg University, October 30, 2017

Stefan Bîrs
<sbars15@student.aau.dk>

Tiberiu-Ioan Szatmari
<tszatm15@student.aau.dk>

Kamran Thomas Alimagham
<kalima15@student.aau.dk>

Chapter 1

Introduction

Here is the introduction. The next chapter is chapter 1.

a new paragraph

1.1 Examples

You can also have examples in your document such as in example 1.1.

Example 1.1 (An Example of an Example)
Here is an example with some math

$$0 = \exp(i\pi) + 1\ . \tag{1.1}$$

You can adjust the colour and the line width in the `macros.tex` file

A Paragraph You can also use paragraph titles which look like this.

A Subparagraph Moreover, you can also use subparagraph titles which look like this. They have a small indentation as opposed to the paragraph title.

Is it possible to add a subsubparagraph?

I think that a summary of this exciting chapter should be added.

Chapter 2

Problem Description

2.1 Problem Description

List to wright about here:

- Main Objective/Goal
- What will be required
- How are we going solve the Objective?

2.2 Problem Delimitation

To ensure that the project meets the desired scope, specific success criteria are formed which will be evaluated during the final stages of the report. The following list of requirements are the desired success criteria, where the focus will ve on reaching the given goals.

- Detect human speech
- Design a deep neural network
- Compare different regulation technics (drop out, bach normalization...)
- Design a filter
- Create a network infrastructure for Client/Server Model (Distribution)
- Implement a speech recognition system with information retrival

Brief summary on how are we going to obtain all of these bullet points.

Chapter 3

Speech Processing

Speech is an effortless and highly efficient form of communication. Continuous speech recognition software is readily available from stores and it allows the user to interact with its surrounding without the need of written commands [1, p. 396]. Amazon's Alexa [2] and Microsoft's Cortana [3] are perfect examples of home companions that are able to perform basic tasks received via voice commands. As speech recognition increases in accuracy to the high nineties, reaching a performance of 99% means that this technology will make the leap from being annoying to use to becoming the main way we interact with computers. This idea is further backed up by the fact that people tend to speak much faster than they are able to type [4]. In some cases, speech can be three times faster in conveying information than typing and for professional debaters, speaking can be even six times faster.

Words are carried as sound waves, which are analog signals. In order for the speech to be processed, it needs to be run through a signal processor that selects the frequencies and amplitude of the signal. Further on, the signal is mapped to individual sound units called phones. These phones need to be identified and grouped in such a way to ensure that each word has a different phonetic structure, otherwise, words would be impossible to distinguish from one another.

Table 3.1: Phones

[ay]	<u>ir</u>is
[b]	<u>b</u>in
[er]	<u>bir</u>d
[l]	<u>l</u>ip
[p]	<u>p</u>in
[th]	<u>th</u>in

Phones can have different sounds depending on the context. For example, the phone th in the word *three* has a different sound to th in *then*. To overcome these different variations of the same phones, it is better to abstract the phones into a generalized grouping called a **phoneme**. Phonemes are written in-between forward slashes (*/th/*) and they will have a specific pronunciation for each sound, depending on the context. These phonemes are used as a transitional layer when trying to convert speech to text and vice versa, when speech needs to be synthesized.

3.1 Signal processing

Sound waves that carry human speech are variations in air pressure. The key components of a sound wave are its **amplitude**, which measure the intensity of the sound and the **frequency**, which describes the rate at which the amplitude varies over time. When speaking in a microphone, the change in air pressure causes the diaphragm to oscillate. The size of the oscillations is directly proportional to the amplitude of the signal, while the rate at which the diaphragm oscillates gives represents the rate at which the air pressure changes. At specific time intervals, the signal can be sampled and the data can be used in a wide array of digital signal processing tools. For example, the digital signal can be plotted as an x-y plot, where the y-axis defines the amplitude and the x-axis shows the passage of time. The frequency can be easily be determined from the plot as we find the number of cycles that the signal does per second.

Although a visual difference can be seen when plotting vowels and consonants, a visual inspection of the waveform will not reveal any critical information, making it impossible to see all the phonemes.
To feed this information to the neural network, some preprocessing and signal manipulation is required so the input can be understood and passed through the neurons. These steps are expanded upon in the following subsections.

Preprocessing

No matter how powerful and how advanced a computer is, it still works in discrete time. In order to represent an analogue signal on the computer we have to sample it according to the Nyquist–Shannon sampling theorem, which states that the sampling frequency has to be at least double the frequency of the original signal in order to replicate the original signal without aliasing. For a better understanding of the process, a visual representation of the word "Hello" as it is formed for the input of the neural network is shown in figure**??** where it is sampled at 8000 Hz.

Fourier transform

To analyse a discrete signal we use the Fourier transform. This allows us to convert the signal in the frequency domain and deconstruct it in a series of simple sine waves. These sine waves have specific amplitudes and frequencies and while plotting the FFT of the signal, the dominant waves can be identified by the intensity of the amplitude.

In Figure **??**, a visual representation of the FFT is shown and it can be seen that there are a lot of frequencies that make up the word "Hello". The dominant ones are in the range of 500Hz, and considering that this recording has a minimum amount of noise in it we have to consider the full range of the frequencies, from zero and up to 3500Hz.

Spectrogram

The spectrogram of the signal was computed from the FFT and it shows a visual representation of the spectrum of frequencies marked by the intensity of the color. After the spectrogram is sliced into 25 ms chunks, it is feed as the input to the network. A neural network can find patterns in this data more easily than raw sound waves because audio data and patterns can be seen in the graph.

The figures are on GitHub, need to figure out how they work in LaTeX

Chapter 4

Machine Learning for Speech

4.1 TensorFlow

TensorFlow provides multiple Application Programming Interfaces (APIs) for machine learning. The lowest level API "TensorFlow Core" provides complete programming control [5]. For those who require fine levels of control over their models, TensorFlow Core is a well-suited tool for the job. There are higher level APIs that are built on top of TensorFlow Core. These higher level APIs are typically easier to learn and use than Tensorflow Core [5]. In addition, the higher level APIs such as "tf.estimator" helps with managing data sets, estimators, training and inferences (testing your trained network), as well as, making repetitive tasks easier and more consistent [5].

The subsection below will start with TensorFlow Core. In order to gain an understanding of the basic principles that TensorFlow has to offer, a model shall be made. In the subsection after that, the same model will be implemented with tf.estimator. Knowing TensorFlow Core principles will give us a great mental model of how things are working internally when we use the more compact higher level API.

4.1.1 Tensors

The central unit of data in TensorFlow is the tensor. A tensor consists of a set of primitive values shaped into an array of any number of dimensions. A tensor's rank is its number of dimensions. Here are some examples of tensors:

```
3 # a rank 0 tensor; this is a scalar with shape []
[1., 2., 3.] # a rank 1 tensor; this is a vector with shape [3]
[[1., 2., 3.], [4., 5., 6.]] # a rank 2 tensor; a matrix with shape [2, 3]
[[[1., 2., 3.]], [[7., 8., 9.]]] # a rank 3 tensor with shape [2, 1, 3]
```

customize colors

4.1.2 TensorFlow Core

4.1.3 tf.train API

4.1.4 tf.estimator API

4.2 Deep neural network for speech

4.2.1 Deep Feed Forward (DFF)

The simplest of all neural networks, the feedforward neural network, moves information in one direction only. Data moves from the input nodes to the output nodes, passing through hidden nodes (if any). The feedforward neural network has no cycles or loops in its network.

reference this part with *https : //www.allerin.com/blog/six − types − of − neural − networks*

4.2.2 Recurrent Neural Nerwork (RNN)

The recurrent neural network, unlike the feedforward neural network, is a neural network that allows for a bi-directional flow of data. The network between the connected units forms a directed cycle. Such a network allows for dynamic temporal behavior to be exhibited. The recurrent neural network is capable of using its internal memory to process arbitrary sequence of inputs. This neural network is a popular choice for tasks such as handwriting and speech recognition.

link this part with link above

Long/Short Term Memory (LSTM)

Chapter 5

Distributed speech recognizer

5.1 The DSR system

5.2 Acoutic models

5.3 Language model

Chapter 6

Knowledge-based IR system

6.1 The document retrieval system

6.2 The question answering system

Chapter 7

Discussion

Chapter 8

Conclusion

In case you have questions, comments, suggestions or have found a bug, please do not hesitate to contact me. You can find my contact details below.

Jesper Kjær Nielsen
jkn@es.aau.dk
http://kom.aau.dk/~jkn
Fredrik Bajers Vej 7
9220 Aalborg Ø

Bibliography

[1] R. Callan. *Artificial Intelligence*. Macmillan Education UK, 2003. ISBN: 9780333801369. URL: https://books.google.dk/books?id=S1NOQgAACAAJ.

[2] Darrell Etherington. *Alexa*. https://techcrunch.com/2014/11/06/amazon-echo/. Accessed: 2017-10-30.

[3] *Cortana*. https://support.microsoft.com/en-us/help/17214. Accessed: 2017-10-30.

[4] Arie Ogranovich. *Speed*. https://www.quora.com/Can-we-type-in-English-faster-than-we-speak. Accessed: 2017-10-30.

[5] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: https://www.tensorflow.org/.

Appendix A

Appendix A name

Here is the first appendix