# Android Neural Network Optical Character Recognition

# Android Application

# “Ochre”

# Neural Network Optical Character Recognition

**Naea Fairbrother**

**P4217876@live.tees.ac.uk**

**University of Teesside**

**School of Computing and Digital Technologies**

## Final Year Project Report

# CONTENTS

## ABSTRACT

Python neural network implemented via third party module, Chaquopy, and trained with the IAM dataset for handwriting recognition and conversion to digital text, which can then be edited, spellchecked or run through an Azure based text analysis service. All implemented with Firebase cloud storage and realtime database.

## ACKNOWLEDGEMENTS

*Harald Scheidl*(Scheidl, H. (2019) *Githubharald*)[1], an Austrian coder and the original author of the implemented, pre-trained, open-source, python neural network “SimpleHRT”.

Malcolm Smith, the author and owner of *Chaquopy*(Smith, M. (2018) *The easiest way to use Python in your Android app*.)[2], a third-party android studio module that allows android to utilise python code.

# 1. INTRODUCTION

## 1.1 PROBLEM DEFINITION

Handwritten text has always presented the same problem – legibility of penmanship. The art of penmanship has always contained a highly variable state historically. From the *textura*(Nottingham.ac.uk (2019))[A] script of the middle ages to the multiple forms of *cursiva*(Nottingham.ac.uk (2019))[B] scripts utilsed from the later middle ages, right through to present day cursive as it is taught in (British) schools. If reading ones' own handwriting is not bad enough, then reading another's surely would cause issues – everyone has heard the stereotype of doctors handwriting, for example.

## 1.2 JUSTIFICATION

With the current generations' ever-increasing dependency on technology, the quality of handwriting is constantly degenerating, as the written word is often replaced by computers and smartphones. This dependence also leads to a deterioration in spelling and grammar, as auto-correct and spellcheck are now commonplace. The concern over handwriting quality was so severe in Cambridge University in the United Kingdom that it was considering scrapping 800 years of tradition by abandoning written exams entirely, in favour of typed exams(Busby, M. (2017) *Cambridge considers typed exams as handwriting worsens*)[3].

"It is also now estimated that between 5 to 10 percent of the population has dyslexia, though this number could range as high as 17 percent"(Dyslexiahelp.umich.edu (2019))[4]. Dyslexia, and the related dyspraxia (difficulty in activities requiring coordination and movement), are often detrimental to the development of an individuals' handwriting, as the perception of letters and recreation of the associated shapes is hindered.

Studies across the world have had mixed findings regarding students' preferred note taking methods. "Many of the students in our study found [that] making handwritten notes leads to greater retention of data than if it is typed."(Bothwell, E. (2017) *Pen and paper 'beats computers for retaining knowledge'*)[5].

Whilst there are multiple applications available to convert text to digital, process referred to as 'Optical Character Recognition'(SearchContentManagement (2019))[6], nearly all these are designed for printed text and are used simply as a quick and easy form of transfer from hard copy to digital(Play.google.com (2019))[7]. Very few instances are currently available to convert handwritten text to digital, as this process is significantly more complex, as every individuals' handwriting is unique, much like a fingerprint. Graphology, the study of human personality through writing takes this one step further, proposing that it is possible to infer personality traits of the individual from their writing style(The British Institute of Graphologists (2019))[8].

## 1.3 PROPOSAL

A smartphone application to convert handwritten text to digital, which can then use spellcheck and be edited, or have several forms of text analysis performed upon it for the users' benefit.

This could benefit many people but would primarily be targeted towards teenagers and university students.

## 1.4 PARAMETERS

- Mobile
- Quick
- Easy to use
- Broad recognition of handwriting styles
- Recognition of various languages

# 2. RESEARCH

## 2.1 HANDWRITING RECOGNITION

If you have ever filled in a government form, or an exam, and written individual characters in those little, equally spaced boxes, in block capitals – your handwriting was probably read by a computer.

If you have sent a letter via the postal service and written the address on the front of the envelope – your handwriting was probably read by a computer.

If you have written a bank cheque – your handwriting was probably read by a computer.

Computers have been used for 'off-line recognition' - "the automatic conversion of text in an image into letter codes for use within computer and text-processing applications"(En.wikipedia.org (2019) *Handwriting recognition*)[9] – for years.. Several techniques have been utilised to simplify this comparatively difficult task.

### 2.1.1 TRADITIONAL TECHNIQUES

#### PROBLEM DOMAIN REDUCTION

By reducing the number of possible characters that could be found, the speed and accuracy of the recognition system can be boosted.

Primarily this is accomplished either by using specified character ranges, such as only numbers or block capitals.

Unfortunately, this technique is of no use in this application. Trying to implement it would be too limiting on the uses of the application.

#### CHARACTER EXTRACTION

Individual characters in the scanned image are extracted. This means that the computer does not, in fact, see words on a page, but instead views a document as nothing more than a series of letters.

This method does however experience major problems with cursive style lettering, as the joined nature of the letters can often lead to two or more connected characters being returned instead as a single image. This then is incomprehensible to the computer, which must return a single character.

This technique, whilst not useless, has serious limitations in both historic and modern handwriting recognition, as cursive letterforms have been predominant since development in the later middle ages(Nottingham.ac.uk (2019))[10].

**FEATURE EXTRACTION**

Programmers manually determine the properties they feel are important in recognition, giving increased control of what properties are used in identification.

Example properties may include:

- Aspect Ratio
- Percentage of Pixels in a region
- Number of Strokes
- Average Distance from Center
- X or Y Axis Reflection

Whilst this approach increases accuracy, any system that utilises it will be very slow to develop compared to a neural network, as the properties must be manually defined whereas the neural network will be able to learn them automatically.

### 2.1.2 MODERN TECHNIQUES

Where traditional techniques focus on segmenting individual characters for recognition, modern handwriting recognition utilises machine learning and neural networks to teach a computer how to interpret all the characters in a segmented line instead. The neural network then decides on the correct identification of a word through several decision factors which depend on the complexity of the neural network. The final returned word is chosen as the most likely of all possible options through a confidence rating – an internal value assigned by the neural network for how certain it is that the identification is correct. Several new types of techniques have been devised to increase the accuracy of identification.

**CHARACTER RECOGNITION**

Individual character recognition still exists in two basic types, image correlation and feature extraction.

Image correlation compares an image to a saved glyph, pixel by pixel. This technique requires the character to be identified to be isolated, as well as identical scale and similar font style. This technique is only suitable for typewritten text and as such not suitable for this application.

Feature extraction in this case reduces the character to be identified to a series of stylised vector-like features – lines, loops and intersections. These features are then compared to an abstract representation of a character, which may have one or more glyphs saved in memory, based on the neural network training. A “Nearest Neighbour Classifier”, such as the “K-Nearest Neighbours Algorithm”, is then used to determine the nearest match.

The “K-Nearest Neighbours Algorithm” compares against the entire data training set and returns the K most similar instances. Then to determine which of the returned K glyphs is most similar to the character to be identified, a distance measure is used.

Popular distance measures include:

- ***Euclidean Distance*** – calculate square root sum of squared straight-line distance between all points across all vector attributes.
- ***Hamming Distance*** – calculate the distance between binary vectors across all vector attributes.
- ***Manhattan Distance*** – calculate distance between real vectors using sum of absolute difference across all vector attributes.
- ***Minkowski Distance*** – a generalisation of Euclidean and Manhatten.

(Brownlee, J. (2016) *K-Nearest Neighbors for Machine Learning*)[11].

**PRE-PROCESSING**

Multiple techniques have been developed to systematically alter the original image, to increase the chances of successfully identifying.

A few of the available techniques are:

- ***De-skew*** – rotation of the image to perfectly align horizontally or vertically.
- ***De-speckle*** – removal of light and dark spots and smoothing of edges.
- ***Binarisation*** – converting the image to black and white to allow a high contrast between words and background.
- ***Line Removal*** – cleaning up margins, lines or grid boxes which can confuse the computer.
- ***Zoning*** – identification of distinct blocks of text. I.E. columns, paragraphs, etc.
- ***Normalisation*** – conform all identified word segments into a pre-defined aspect ratio, scale and position.

Most written work performed in schools and universities is completed in lined books or notepads, with incidental marks being common, and with the original image being taken on a handheld smartphone camera, the alignment and zoom of each line segment cannot be assumed. As such, each of these simple pre-processing techniques are both important and suitable for the application to facilitate a maximum success rate of identification.

### POST-PROCESSING

Identification accuracy can be increased if output is constrained by a lexicon – a complete set of expected word outcomes – for example, the English dictionary. This means that only the words that appear in the lexicon are valid words for the neural network to give as outputs. This technique however is only suitable in cases where all potential words to be identified in the document are predictable.

This technique may be able to assist with the identification of most words on most documents, however, due to the "all or nothing" nature of this technique, a lexicon may be too limiting. Creating an adequately proportionate lexicon may be possible, however, it would still experience problems with names and slang. As such, this is not suitable for this application.

"Near-neighbour analysis" is a technique that looks at the frequency of concurrent occurrences of words, knowing that some words are often seen together. This means the neural network will consider more likely word combinations when scanning a document, to assist in returning the correct identification.

Giving the neural network a base knowledge of grammar can give the neural network a sense of meaning of the scanned document, allowing greater accuracy through key identifications of nouns and verbs, which used alongside the near-neighbour analysis technique can give a greater accuracy.

The Levenshtein distance is used to measure the difference between sequences. When used to compare two words, the Levenshtein distance is the minimum number of single-character alterations – insertion, deletion or substitution – to change one into the other(Barbar, N. (2018) *The Levenshtein Distance Algorithm - DZone Big Data*)[12]. This algorithm can be implemented in neural networks to assist with most likely words when an identification of a word has either failed or had a low confidence score. It is also useful in identifying words that have spelling mistakes in them, as a low Levenshtein distance can be an indication to the intended word.

## 2.2 MACHINE LEARNING

> "Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine learning focuses on the development of computer programs that can access data and use it learn for themselves."
>
> (Varone, M., Mayer, D. and Melegari, A. (n.d.) *What is Machine Learning?*)[13].

This process begins with the system being fed test data, which the system then uses to discern patterns. These patterns in the example data influence the systems future decisions on any further data. Larger datasets generally provide greater accuracy for future predictions, but this does have an upper limit, which depending on the complexity of the system, the size of the test dataset and the correlation of patterns in the data, can realistically be any value, though anything above 60%, depending on the purpose of the system, can be considered a good result.

Machine learning can be achieved in several ways. Some of the machine learning methods are:

- ***Supervised Machine Learning*** – starts by analysing a known dataset, with labeled examples, to infer a model ruleset. The algorithm can compare its output to the intended, correct output in order to modify its' model. After sufficient training the system can predict the output of any new data. Labeled examples require extra processing, which can take significant time and resources to accomplish.
- ***Unsupervised Machine Learning*** – if the training dataset is neither classified nor labeled, an unsupervised method is used. Instead of figuring out the correct output, it explores the data and infers a model ruleset from any observed patterns. Unlabeled data generally doesn't require any additional resources.
- ***Semi-Supervised Machine Learning*** – falling between supervised and unsupervised, semi-supervised methods use both labeled and unlabeled data in the training set – usually a small quantity of labeled and a large quantity of unlabeled. Systems implementing this form of learning are able to significantly decrease learning times, using the labeled data to imprint a rudimentary ruleset before allowing the algorithm to optimise itself through any observed patterns. This method is usually chosen when the data requires skilled and relevant resources to train from.
- ***Reinforcement Machine Learning*** – interacts with its environment through trial and error, search and reward. This method allows systems to automatically determine the optimal performance within context. A reinforcement signal – a simple reward feedback for the desired behaviour – is required to learn the best action.

### 2.2.1 ARTIFICIAL NEURAL NETWORKS

One of the most important components of machine learning is the artificial neural network. They are systems inspired by brains, intended to emulate how humans learn. A neural network comprises of an input layer and an output layer, as well as another hidden layer, that processes the input layer into something the output layer can use. Neural networks are designed to be able to teach themselves from patterns that are too complex, or numerous, for a human operator to define.

#### PROS AND CONS

+ Flexible – a neural network can process any data that can be expressed numerically
+ Can be used for both regression and classification problems
+ Can be used for both linear and non-linear data
+ Can process data with many features – for example, images – by splitting the problem into many layers of simpler elements
+ Are infinitely scalable – adding more inputs or layers adds to the training of the network
+ Can learn from any amount of data – the more data the network has access to, the more it can refine itself

+ When trained, processing future data is relatively quick
− Neural networks are black box style programs – it is impossible for a human operator to know quantitatively how each independent variable influences each dependent variable
− Computationally expensive to train
− Dependent on training data

(Subscription.packtpub.com. (n.d.). *Pros and cons of neural networks*)[14].

## BACKPROPAGATION

"Backpropagation is a technique used to train certain classes of neural networks – it is essentially a principal that allows the machine learning program to adjust itself according to looking at its past function. Backpropagation is sometimes called the "backpropagation of errors"."

(Techopedia.com. (n.d.). *What is Backpropagation?*)[15].

When training a neural network, backpropagation is what allows the system to learn from its' mistakes and optimise itself. Data is fed into the neural network input layer; the hidden layer applies its processes to the data through various weights which influence the systems decisions; and then data is given from the output layer. During training, the output data is then compared to whatever the correct output should have been – and differences from the systems output and the correct output are then used to alter the weights in the hidden layer, thus altering the decision process for the next input data. This is repeated until the human operator is satisfied with the accuracy of the neural network output. This process is backpropagation and is the main training practice that allows neural networks to be a successful form of data processing. Although used in both supervised and unsupervised machine learning algorithms, it is viewed as a supervised learning method.

### 2.2.2 PROGRAMMING

## STYLES

- ***Object Orientated*** - creating and manipulating the interactions of reusable, logical objects to perform tasks and accomplish objectives(Techopedia.com (n.d.) *What is Object-Oriented?*)[16].
- ***Functional*** – applies mathematical functions rather than procedural code threads with data that is likely to change(Techopedia.com (n.d.) *What is Functional Programming?*)[17].
- ***Imperative*** - linear code that relies on the implementation of subroutines to complete calculations. Procedural programming is a specific type of imperative programming, though they are often used as synonyms(Techopedia.com (n.d.) *What is Procedural Programming?*)[18].
- ***Logic*** - ideally, a series of true assertions used by the system to solve the problem. Realistically, the developer still needs to assist with the procedural interpretation of the logical assertions made(Encyclopedia.com (2004) *logic programming languages*)[19].

- ***General Purpose*** - does not contain language constraints to specific applications – rather, general purpose languages are more flexible, as they are designed for implementing into any application domain.

## LANGUAGES

Due to the complexities and specific requirements in creating machine learning models, some languages are eminently more suitable for the task than others.

### *1. Python*

The most commonly used language in AI development due to being a high-level language, which allows accessibility to a vast range of developers of varying skill levels. Python has a comparatively short development time to other AI development languages and though it is a general-purpose language is able to implement object, functional and procedural orientated styles. There is a sizeable quantity of pre-created libraries to simplify and streamline the creation process. For example, Numpy, a library for scientific computations, or Pybrain, used specifically for machine learning.

### *2. R*

A very efficient language for analysation and manipulation of statistical data, R is especially good when working with mathematical formulae and symbols. It is a general-purpose language; however, it does include multiple packages specifically for machine learning - for example, RODBC, Gmodels, Class and Tm. These packages are ideal for solving business associated problems when implementing machine learning models.

### *3. Lisp*

Originally created by John McCarthy, father of AI, in 1958, Lisp is arguably the most suited language for AI development. It is a general-purpose language, predominantly imperative style, capable of processing symbolic data effectively, and favoured for its' prototyping capabilities, dynamic object creation and automatic garbage collection, Lisp development allows for evaluation and recompilation without need to restart the program. Many of these features originally unique to Lisp have since been migrated into other languages.

### *4. Prolog*

Alongside Lisp in terms of suitability of AI development, Prolog features integrated list handling, making it ideal for tree-style data structures. The language is very efficient for pattern recognition and prototype development and allows simultaneous database creation whilst running. It is a logic style programming language and is often used for medical projects and developing expert AI systems.

### *5. Java*

Easy to use and debug, Java is another high-level language that has much to offer to AI development. It is a general-purpose, class-based, object orientated language, with easy graphical representation of data and designed for improved user

interaction. This language contains package services and the integrated memory manager allows for easier work with large-scale projects. Java also has the Standard Widget Toolkit (SWT) and Swing, or more recently JavaFX, incorporated into it - tools that enable developers to make appealing, sophisticated graphics and user interfaces. Java operates the principle of Once Written Read/Run Anywhere (WORA), utilising Virtual Machine Technology to enable implementation on different platforms, making it very easily portable.

(Nautiyal, D. (n.d.). *Top 5 best Programming Languages for Artificial Intelligence field*)[20].

***6. C++***

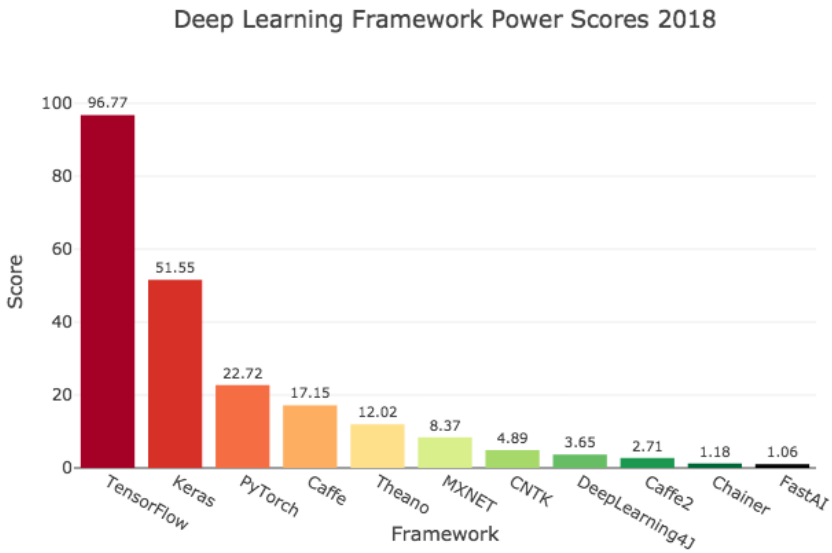
A general-purpose language that can be implemented in an object orientated style. Computationally exceptional, C++ is utilised in search engines and video games due to faster execution and lower response times. It is also able to extensively use algorithms, is efficient in statistical AI techniques and supports reuse of programs through inheritance – however, it is not suitable for multithreading; rather it is best used to implement the core of a system only. Although high-level, this language is unsuitable for less experienced developers, due to the highly complex bottom-up structure.

(Existek.com. (2018) *AI Programming: 5 Most Popular AI Programming Languages*)[21].

### 2.2.3 AI FRAMEWORKS

AI frameworks come in multiple forms – interfaces, libraries or tools – but their purpose is always the same – to provide developers with collections of pre-built, optimised components that simplifies and streamlines the creation of machine learning models(Gupta, B. (n.d.) *What is a Machine Learning Framework & 10 that you need to know)* [22].

The majority of the AI frameworks are written in Python; however, this is not an issue, as many languages have integrated or third-party interpreters to allow them to interface with Python. Indeed, all six of the above-mentioned languages can utilise Python in one way or another.




(Hale, J. (2018) *Deep Learning Framework Power Scores 2018* [online] Kaggle.com

Available at: https://www.kaggle.com/discdiver/deep-learning-framework-power-scores-2018

[Accessed May 2019])

The above scoring system was created by Jeff Hale, using eleven data sources across seven categories, measuring framework usage and popularity, before being combined in a weighted methodology to gauge a final score. All these frameworks are open source, all but one work with python, and some can work in other languages.

### 1. *TensorFlow*

Created by Google to replace Theano, TensorFlow is a Python API. It does have experimental support for Java API however this is not currently stable. This framework is dramatically slower than some other frameworks – it does not support 'inline' matrix operations, instead forcing the system to duplicate a matrix before any operations can be performed on it, a practice that causes TensorFlow to take four times as long as cutting-edge learning tools; however, it does support more than just deep learning – it also contains tools for reinforcement learning too. The framework performs automatic differentiation on a computational graph, enabling it to use the same variation of backpropagation for every neural network – this graph is, however, pure Python, and therefore slow to generate.

### 2. *Keras*

Arguably the best Python API in existence, Keras is a high-level library that is designed to sit atop TensorFlow or Theano to provide simpler, more intuitive control to developers. Keras uses a modular building style, connecting configurable blocks together, it is ideal for fast prototyping as it is easy to extend.

### 3. *PyTorch*

Another Python API, PyTorch offers dynamic computational graphing, which enables a neural network to process variable-length input and output. It is another modular building styled framework, with easy capability to create layers and run via the GPU – it does however suffer from poor documentation, and training data often must be generated manually, slowing down development.

### *4. Caffe*

A Python API running on a C++ codebase designed solely for image classification with convolutional nets, Caffe is unsuitable for other deep-learning applications. This framework is ideal for feedforward networks and image processing, as well as fine tuning existing neural networks, however, new GPU layers can only be created in C++ and are problematic to run on GPU clusters in parallel. Caffe is not a good choice on recurrent networks and can be slow and difficult to use on large-scale networks.

### *5. Theano*

The grandfather of frameworks for deep-learning, originally created in 2007 and until roughly five years ago, the only framework being used. Utilising multidimensional arrays, like Numpy, and used with other libraries, Theano is ideal for data exploration and research. It is quite low-level language, however, there are high-level APIs, such as the afore mentioned Keras, to aid with usage. Theano suffers long compile times with large-scale networks as it will only run on a single GPU. Unfortunately, this framework is developmentally dead as of September 2017.

### *6. MxNet*

Supporting multiple languages, such as C++, Python, R, Julia and Perl, MxNet models take a comparatively small amount of memory to implement and can easily scale to multiple GPUs or even multiple machines. Due to its flexibility of language, portability of platform and scalability, MxNet was adopted by Amazon for its' Web Services Deep Learning framework(Gupta, M. (2017) *MxNet: What is it and How to Get Started*)[23].

### *7. CNTK*

'Computational Network ToolKit' is a framework that can be a library for Python, C++ or C# programs, or a standalone tool through BrainScript, Microsoft's own model description language. CNTK allows development of feed-forward, convolutional and recurrent neural networks. This framework utilises error backpropagation learning, with automatic differentiation and parallelisation across multiple GPUs(Docs.microsoft.com (2017) *The Microsoft Cognitive Toolkit - Cognitive Toolkit – CNTK*)[24].

### *8. DeepLearning4J*

Written in Java for use with the Java Virtual Machine, with a Scala API, DeepLearning4J underlying computations are processed by ND4J, a scientific and linear algebra computational library, that is at least twice as fast as it's Python counterpart, Numpy, when processing large-scale matrices, leading to its adoption by NASA's Jet Propulsion Laboratory. This framework is ideal for solving specific

problems quickly with parallisation across multiple GPUs being automatically integrated into the framework.

### *9. Caffe2*

Like its predecessor, Caffe2 is a Python API running on a C++ engine, however it has been designed to be light-weight and have increased scalability to increase its suitability for production environments.

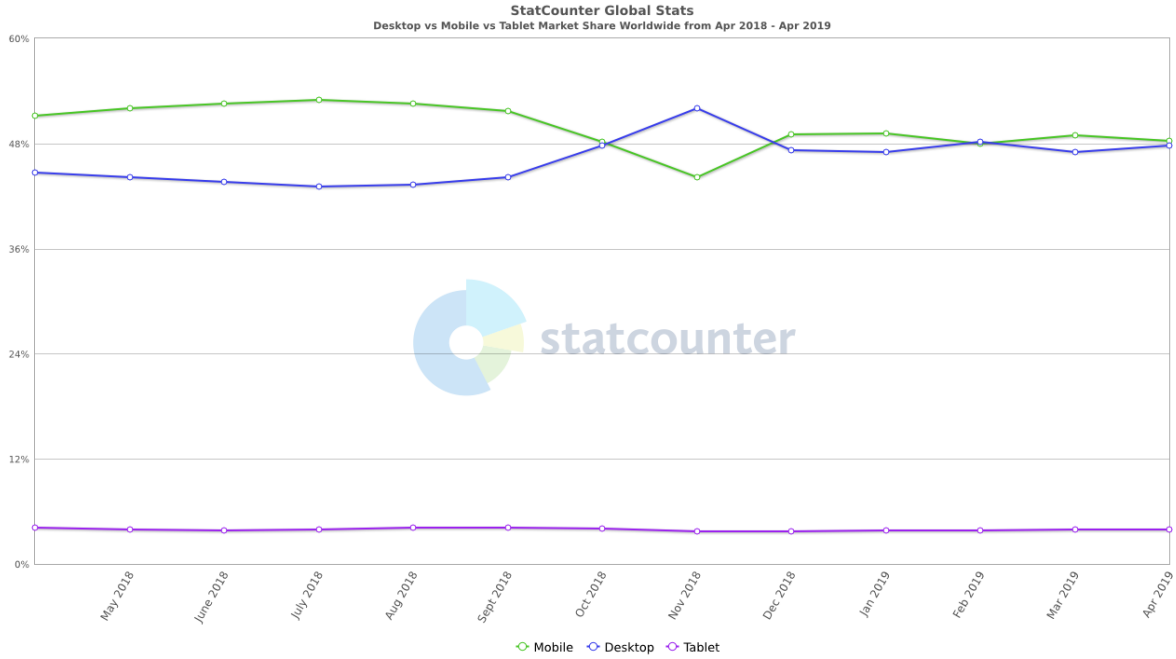
### *10. Chainer*

Before the creation of DyNet and PyTorch, Chainer was the leading framework for dynamic computational graph neural networks which is the preferred methodology for Natural Language Processing (NLP) - the processing of textual data. Benchmarking puts Chainer as significantly faster than other Python-orientated frameworks.

(Skymind (n.d.) *Comparison of AI Frameworks*)[25].

## 2.3 MOBILE TECHNOLOGY

There is a broad selection of mobile technology currently available on the market – smartphones, tablets and notebooks.



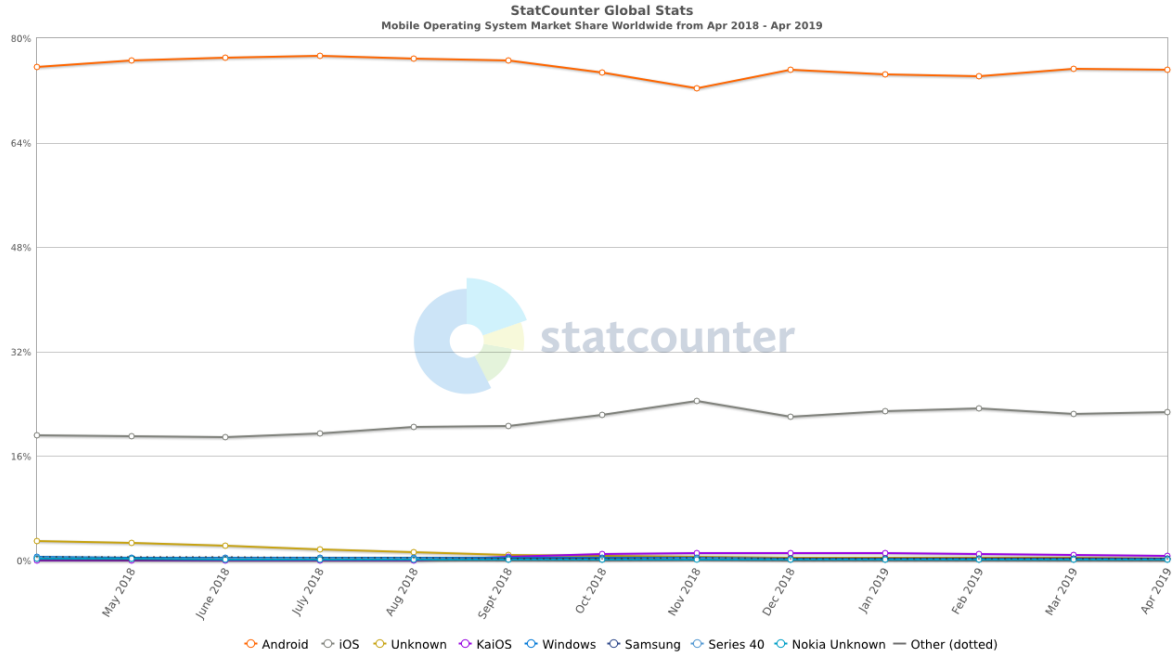

(gs.statcounter.com. 2019. Desktop vs Mobile vs Tablet Market Share Worldwide. [ONLINE]

Available at: http://gs.statcounter.com/platform-market-share/desktop-mobile-tablet

[Accessed May 2019])

This graph clearly shows that the largest market share belongs with the mobile and desktop hardware – though to comply with the portability requirement of this application, it appears most appropriate to focus on the mobile market.

### 2.3.1 OPERATING SYSTEM



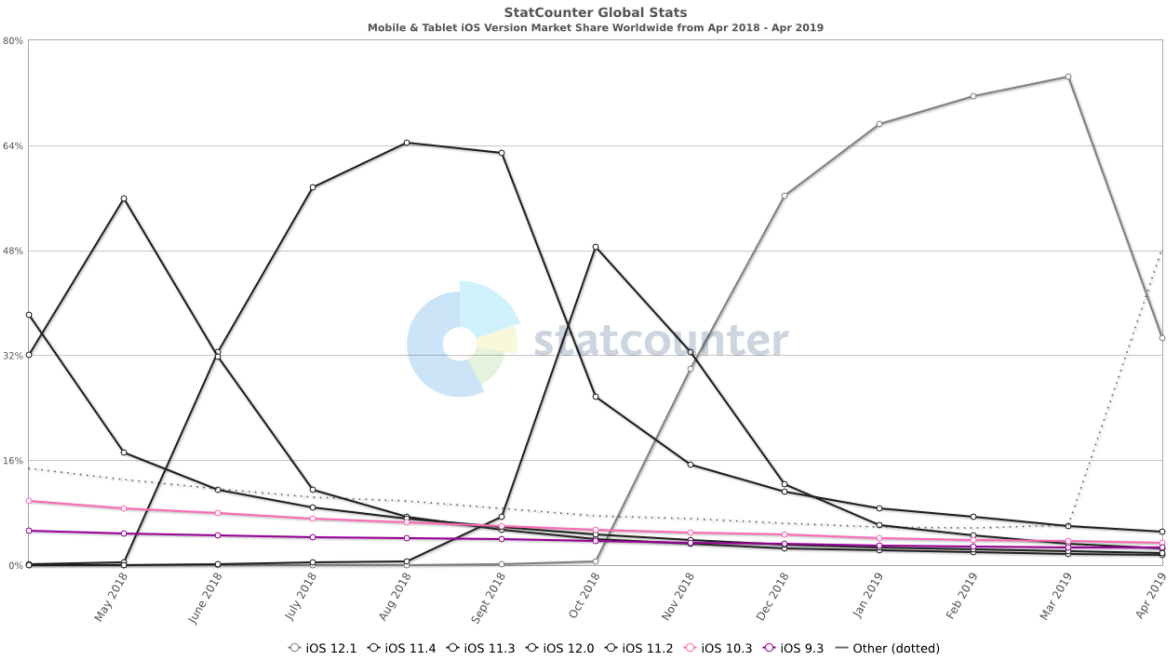

(gs.statcounter.com. 2019. Mobile Operating System Market Share Worldwide. [ONLINE]

Available at: http://gs.statcounter.com/os-market-share/mobile/worldwide/2018

[Accessed May 2019])

The most dominant market shares for mobile operating systems are Apple iOS and Google Android, with Android significantly higher with approximately seventy-five percent of the market, and iOS at approximately twenty-two percent.

## iOS




(gs.statcounter.com. 2019. Mobile & Tablet iOS Version Market Share Worldwide. [ONLINE]

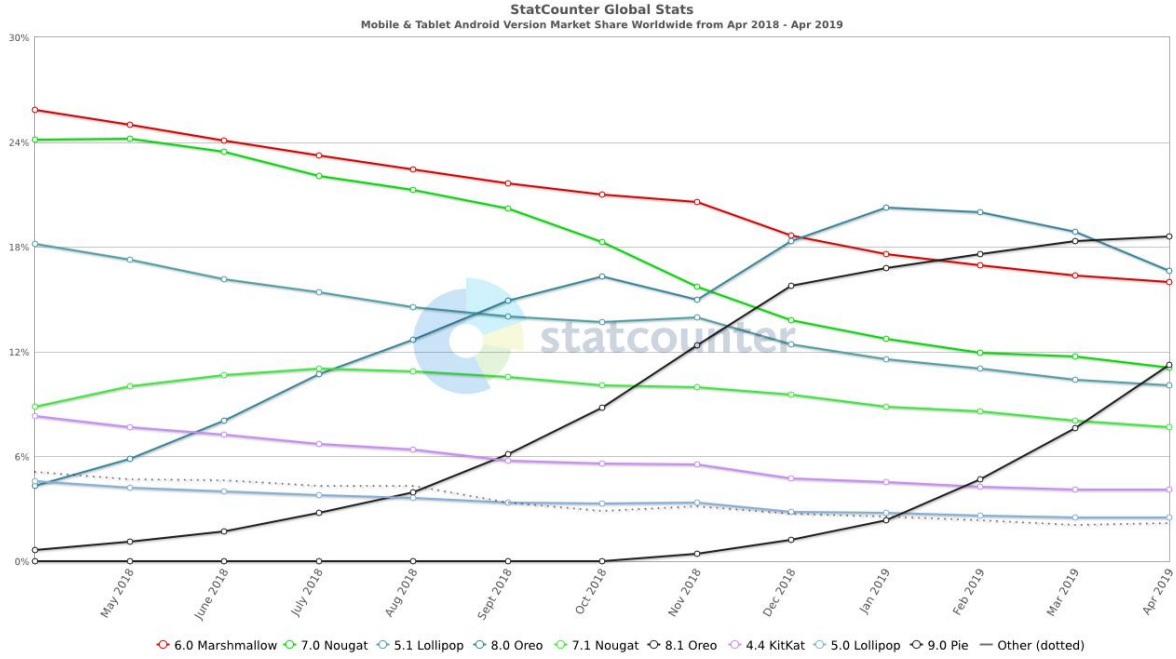
Available at: http://gs.statcounter.com/os-version-market-share/ios/mobile-tablet/worldwide/2018

[Accessed May 2019])

Apple iOS appears to follow a cyclical pattern for the most part, with the most recent version becoming the predominantly used quickly after release and remaining so until the next release. Apple has introduced Basic Neural Network Subroutines (BNNS)(developer.apple.com (n.d) BNNS | Apple Developer Documentation)[26]., under the Accelerate framework, written in the Swift language, starting from iOS 10. The integrated developer environment only works on Mac.

Due to the cyclical nature of Apple iOS being used alongside the entirely new coding language, Swift, and entirely new A.I. framework, Accelerate – but mostly due to the fact that the market share iOS on mobile is less than a third of Android, resulting in a total market share that is able to use neural networks of less than ten percent (9.805%) – Apple iOS is not ideal for this application.

**ANDROID**




(gs.statcounter.com. 2019. Mobile & Tablet Android Version Market Share Worldwide. [ONLINE]

Available at: http://gs.statcounter.com/os-version-market-share/android/mobile-tablet/worldwide/2018

[Accessed May 2019])

Android version usage is much more varied, appearing to have a quick increase upon release followed by a gradual decline. Android Neural Network API (NNAPI) was introduced on all devices running Android Oreo 8.1 and higher(developer.android.com (n.d.) Neural Networks API | Android NDK | Android Developers)[27]. Current trends show that Android Orea 8.1 is currently the most run version and its successor, Android Pie 9.0 currently being fourth most run. Current values result in more than twenty percent of all Android phones being able to run neural networks

(22.461%) with both Oreo 8.1 and Pie 9.0 showing increasing trends, whereas all other versions show decreasing trends.

Android applications are developed in Android Studio, using Java language, both of which are familiar – however, though Python interpreters exist for Java, Android applications cannot by default implement Python code, which previous research has shown to dominate the A.I. frameworks. Android appears to be the more suitable choice for this application.

### 2.3.2 PYTHON INTERPRETATION

- **JYTHON**

Compiles Python into Java bytecode which can then be executed with the Java Virtual Machine (JVM). This enables you to import and use any Java class inside Python. It is designed to allow Python to interface with already existing Java. Jython currently only supports up to Python 2.7. There is no known port of Jython to the Android python(wiki.python.org. 2018. JythonFaq/GeneralInfo)[28]. This interpreter is not suitable for the application.

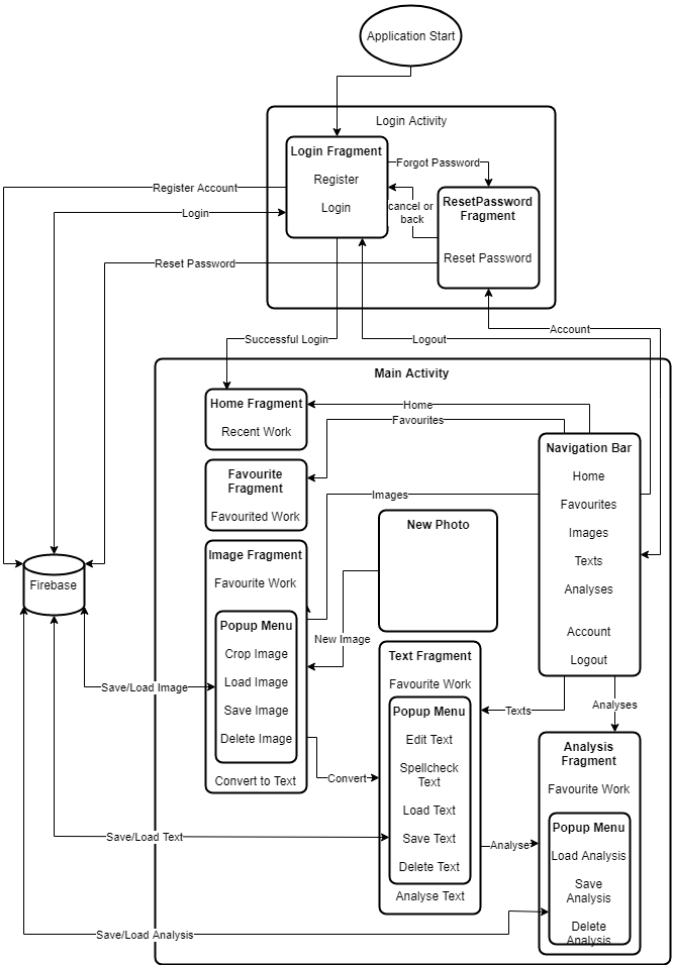
- **PYPY**

Implemented in a subset of the Python language, RPython, PyPy features a compiler designed to support multiple backends, including the JVM. It is designed to improve the performance of Python. PyPy supports Python 2.7. PyPy3 is a fresh out of beta implementation that supports Python 3.5(pypy.org (n.d.) PyPy - What is PyPy?)[29]. This interpreter is not ideal for the application.

- **CHOQUOPY**

Designed to enable easy use of Python in Android applications. Allows free intermixing of Python and Java. From the Java API, allows implementation of Python code in Java – this includes PyPI to take advantage of python packages including SciPy, OpenCV and TensorFlow. From the Python API, it enables all or part of applications to be written in Python, with the entire Android API and user interface toolkit still available. Using Android Studio, Chaquopy is quick and easy to implement into any new or pre-existing application. This is ideal for the application.

# 3. DESIGN AND IMPLEMENTATION




Initial design and implementation utilised activities for all new processes, however, it quickly became apparent that passing data between the activities would be a problem. Also, by implementing fragments it is possible to have the new photo button and the navigation bar persist across all fragments.

Firebase is a simple and easy tool to enable many features, into the application:

- Email & Password Authentication
- Email Link Registration Confirmation
- Email Link Password Reset
- Data Persistence
- Individualised Cloud Storage
- Realtime Database

### 3.1 LOGIN ACTIVITY

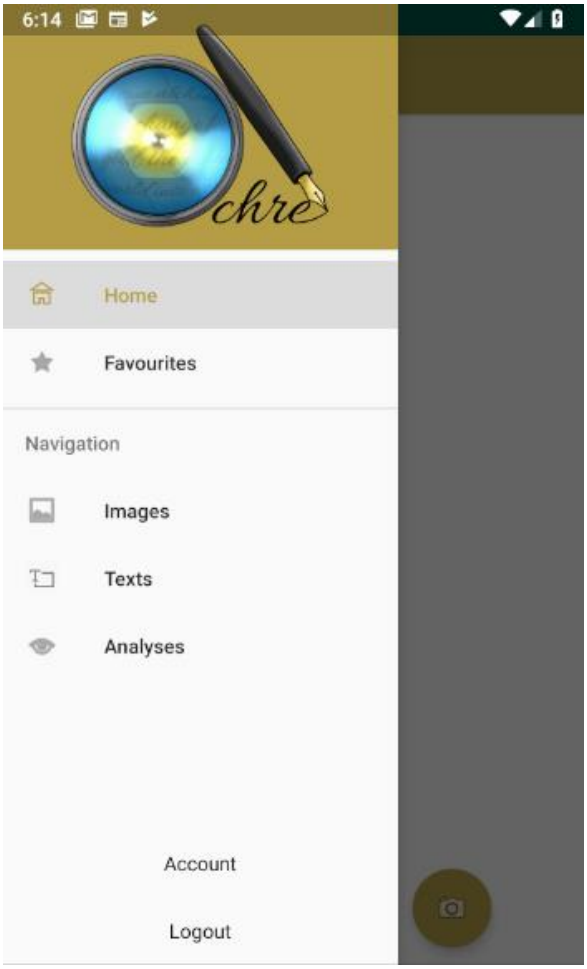
#### LOGIN FRAGMENT


Utilising a standard design for an application login screen. No separate fragment for new registration however – registration is accomplished simply by filling in the fields and pressing the register button. This sends the details to firebase to be stored, and firebase sends an email to the address to confirm registration. Firebase enables the application to persist the login on the device. Coded checks ensure that emails and passwords conform to desired specifications. Android studio introduced an in-built function to enable password boxes to have the eye symbol and toggle password visibility(developer.android.com (n.d.) TextInputLayout | Android Developers)[30].

**RESET PASSWORD FRAGMENT**

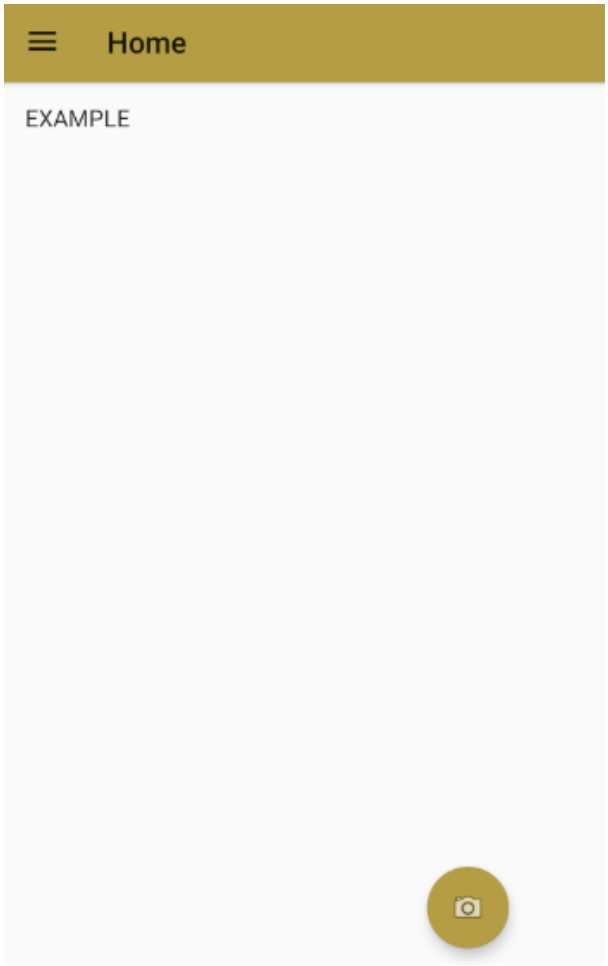

Firebase handles the password reset request by sending an email to the address, which opens an in-browser dialog box to set a new password. The new password however does not have the same constraints that the original password does in the login fragment.

### 3.2 MAIN ACTIVITY



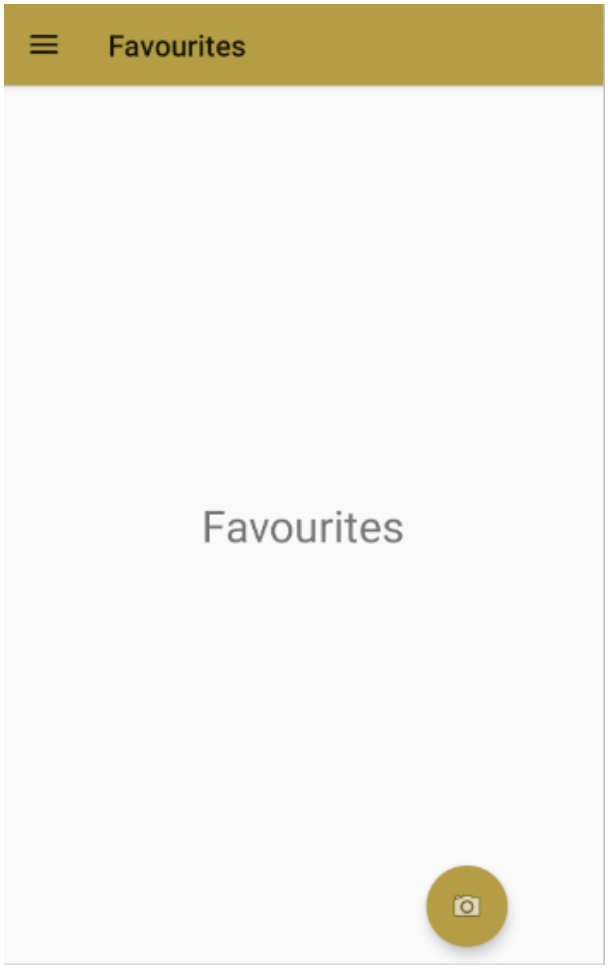

The main activity contains two components that persist throughout all fragments – a button that opens the camera for a new picture to be taken and the navigation bar. The camera button checks first that the current device has a camera, before creating a temporary image file and initiating a new camera intent. Once a photo is taken it is automatically saved to the device gallery. Upon confirming the image, the application returns the image and the user into the image fragment.

**MAIN FRAGMENT**




The home fragment was intended to display a scrollable listview of recent works for easy access. The listview is partially implemented but does not show or provide access to anything.

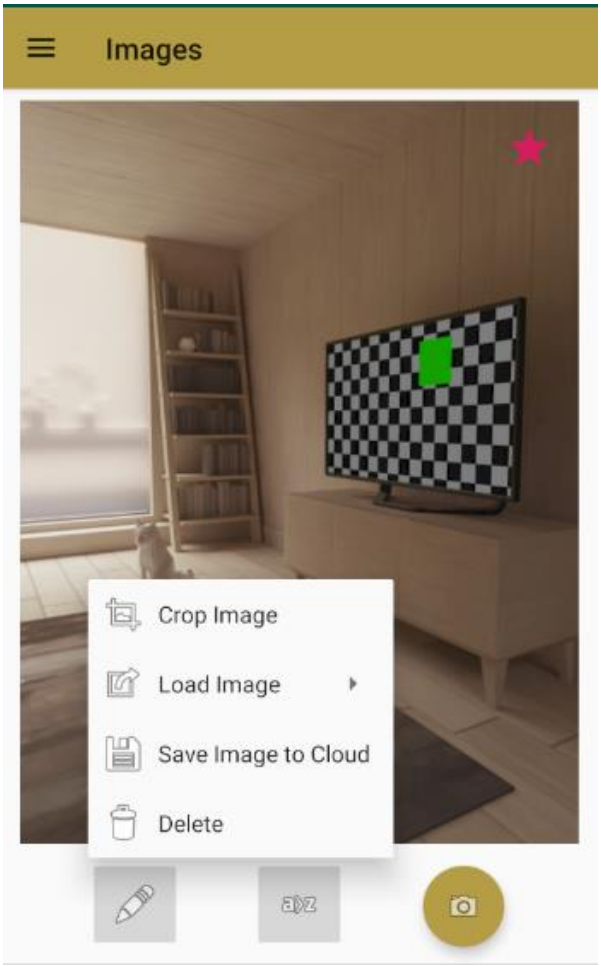
**FAVOURITES FRAGMENT**




This fragment was intended to be easy access to all works that were star favourited. Unfortunately, favouriting is not implemented in the application.

**IMAGE FRAGMENT**


The image fragment if navigated to manually would have a blank field, as no photo would be loaded. This capture shows the resultant image returned by the camera intent.




Although the favourite star is present, no functionality is tied to it in the application currently.

Cropping an image was initially an idea intended to facilitate the neural network with being able to recognise the text by excluding extraneous details - ‘problem domain reduction’ - however, this is not implemented in the current application state.

If the image currently in the image fragment came from the camera intent, it has already been saved locally on the device. The image can from here be saved to firebase cloud storage, which also stores the download URL to the firebase realtime database, which is the only way to retrieve the image without pre-knowledge of the file name.

Images can either be loaded from the device locally, or from the firebase cloud storage.

An image can only be deleted from the local device – the cloud stored image will always be available to download from.

The convert button initialises the neural network.

**CHAQUOPY**

Implementation of Chaquopy into android studio is simple and mostly accomplished through the top level and module level gradle files(chaquopy.com. 2019. Android - Chaquopy

6.2)[31]. Chaquopy requires a licence and will provide free licenses to any open-source projects(chaquopy.com. 2019. Licensing – Chaquopy)[32].

Once implemented, any further python modules must be downloaded via Chaquopy and the gradle with a pip command. Not all modules are available or compatible. 'In our most recent tests, Chaquopy could install about 88% of the top 1000 packages on PyPI. This includes almost all pure-Python packages, plus a constantly-growing selection of packages with native components.' (chaquopy.com. 2019. Android - Chaquopy 6.2).

## NEURAL NETWORK

The neural network implemented in the application is 'SimpleHTR' by Harald Scheidl(github.com. 2019. Handwritten Text Recognition (HTR) system implemented with TensorFlow)[33]. This neural network is pre-trained on the IAM dataset(www.fki.inf.unibe.ch. 2019. IAM Handwriting Database)[34] with a character recognition error rate of approximately ten percent. Not all the stated required module versions(github.com. 2019. SimpleHTR/requirements.txt)[35] are available, but the versions that are available do enable the neural network to work.

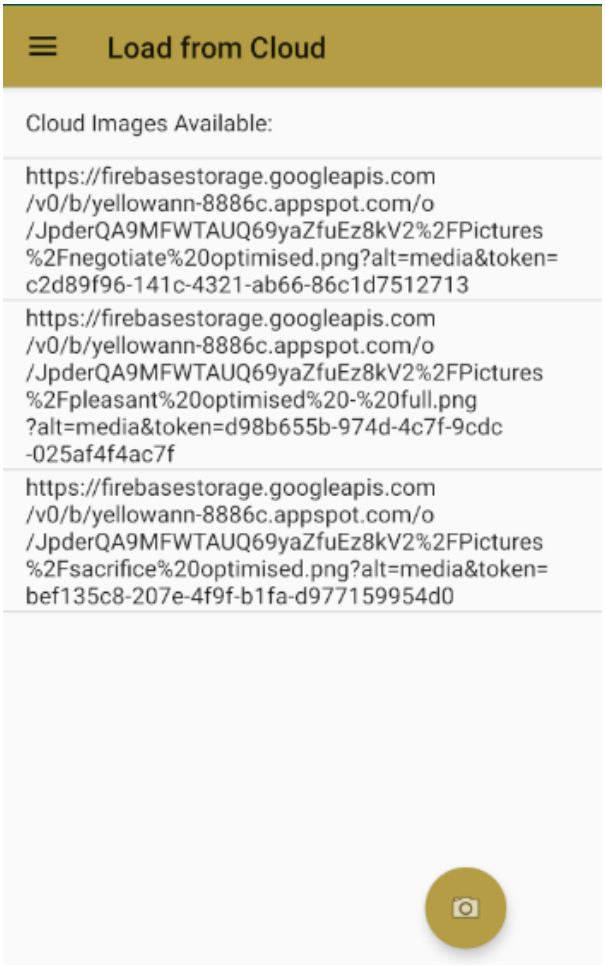
Several small alterations were required to enable the neural network to function with Chaquopy and to be able to read in the image from the image fragment and output to the text fragment.

The words to be read by this neural network have some requirements;

- High contrast between letters and background
- Specific 173x73 pixel sizing
- Word must be left-aligned

The neural network is able to recognise single words, but not more than that. This is adequate as a proof of concept but not suitable for the proposed purpose of the application.

## LOAD FRAGMENT

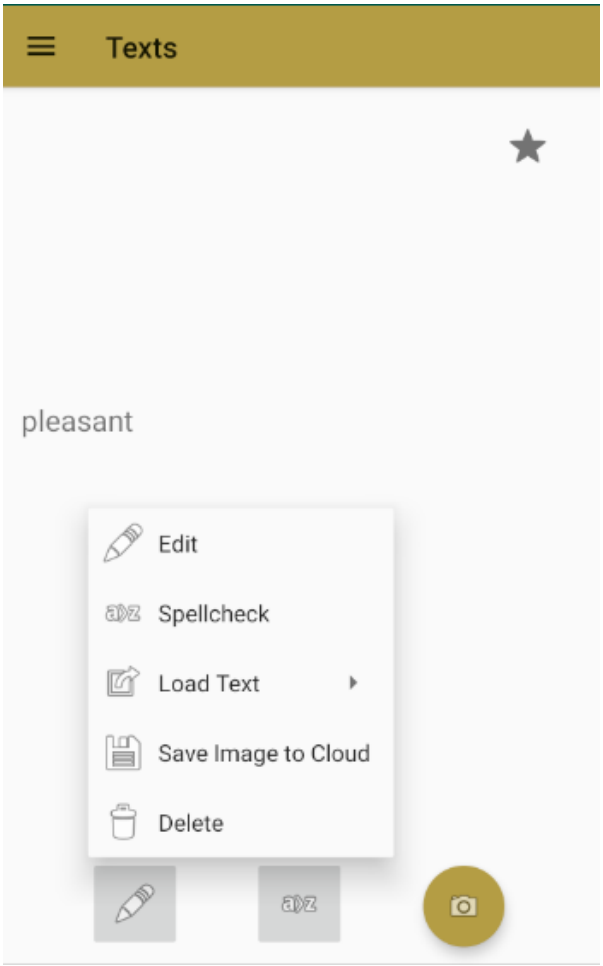



When loading from the cloud, the fragment displays a listview of available downloads, however, this only shows the full download URL which is not user friendly and this long string of characters is not helpful in allowing users to tell which image they are downloading. It is passable as a proof of concept, but not suitable for the proposed application use.

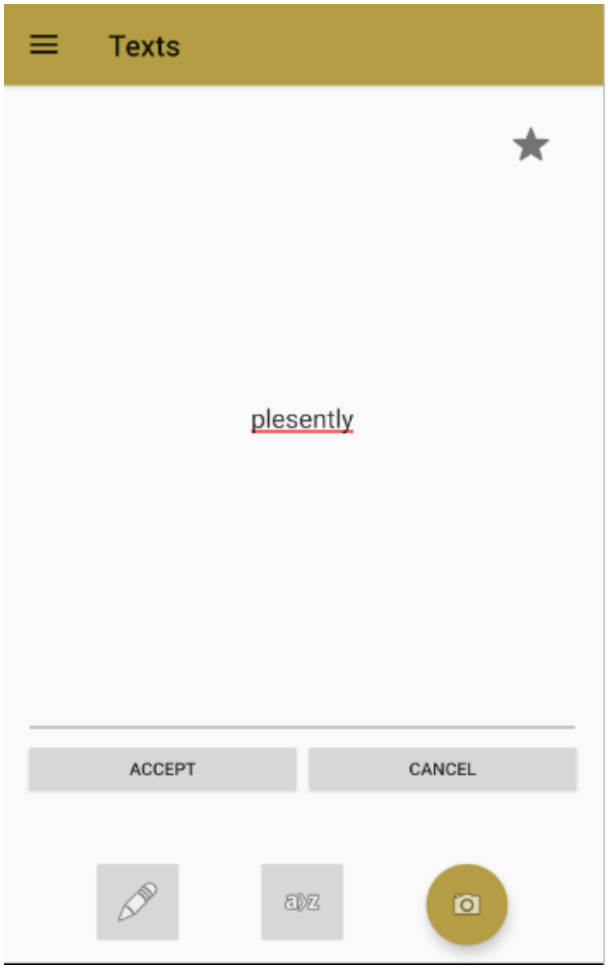
**TEXT FRAGMENT**

The text fragment contains the same basic structure as the image fragment.


The text fragment receives the recognised output word from the neural network and displays it.




The recognised word is not automatically saved locally on the device like the image is but can be saved to firebase realtime database cloud the same as the image. The load function is for both cloud and local. When loaded from the cloud, the text is downloaded onto the device and therefore and further loading is local.

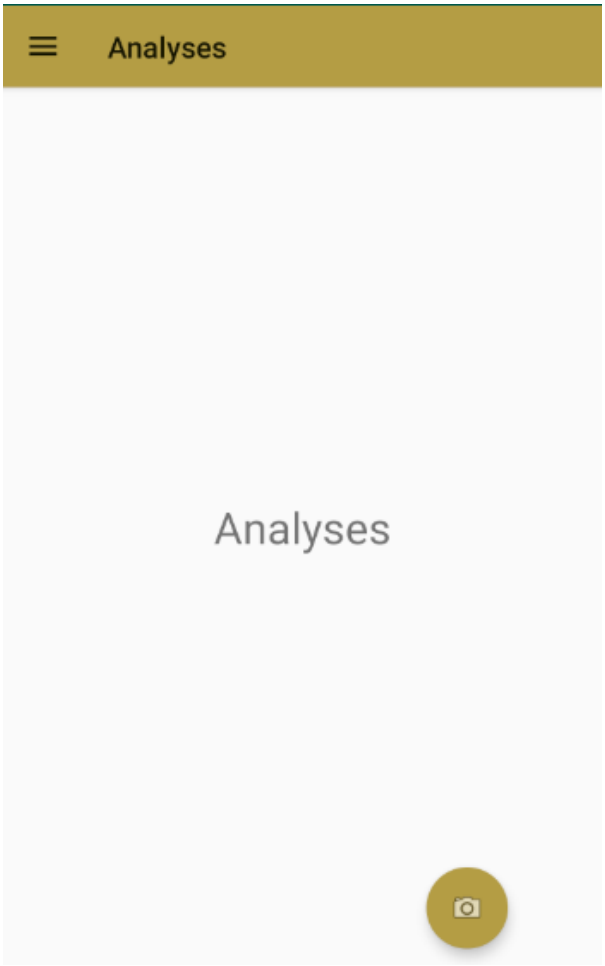



The edit function allows the text field to be manually edited in any way that you would edit any other text field on a mobile device.


The spellcheck function implements a rudimentary spellchecker that is built in to the android API(developer.android.com. 2019. Spell checker framework | Android developers)[36]. Although this spellcheck function is not the most accurate, it is adequate as a proof of concept.

The analyse button is not implemented in the current application.

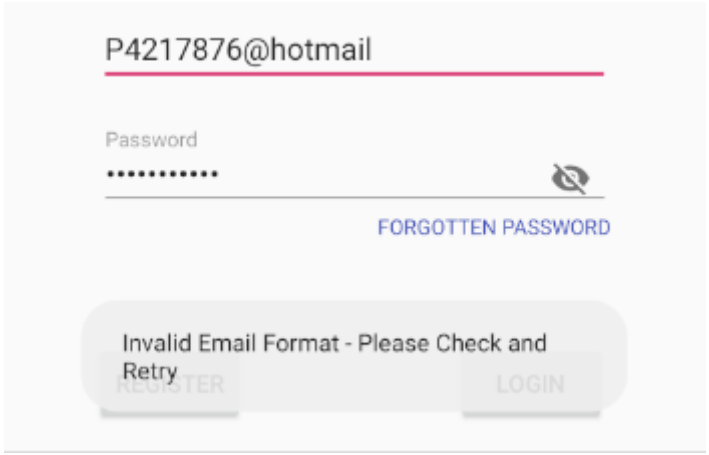
**ANALYSIS FRAGMENT**

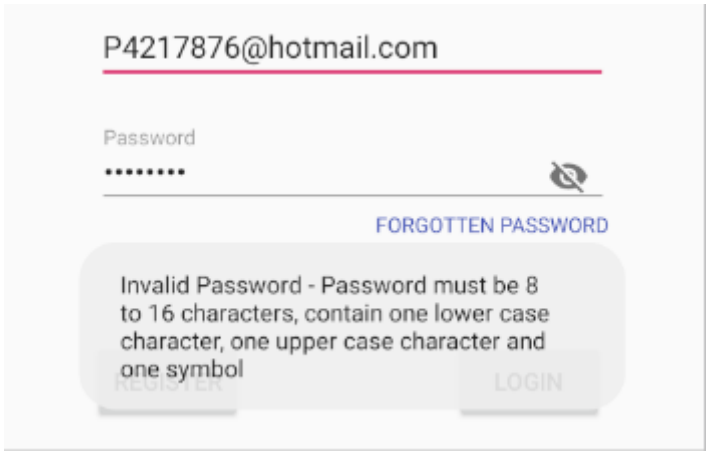



The analysis fragment was meant to receive the output from a modified version of the Azure text analytics cognitive service(docs.microsoft.com. 2019. Quickstart: Using Java to call the Text Analytics API - Azure Cognitive Services | Microsoft Docs)[37].

# 4. TESTING

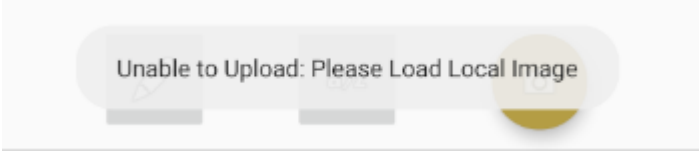
Unfortunately, the application contains no unit tests – however, there was local testing carried out during development and 'try, catch' segments added whenever an input could go awry, toast popups added to keep the user informed and dialog boxes for confirmations. Some examples follow:

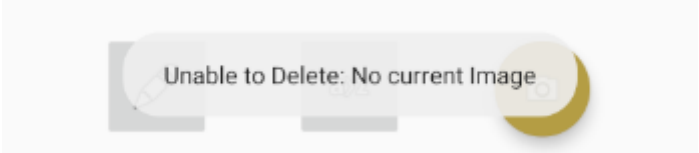

P4217876@hotmail
Password
FORGOTTEN PASSWORD
Invalid Email Format - Please Check and Retry
LOGIN


P4217876@hotmail.com
Password
FORGOTTEN PASSWORD
Invalid Password - Password must be 8 to 16 characters, contain one lower case character, one upper case character and one symbol
LOGIN

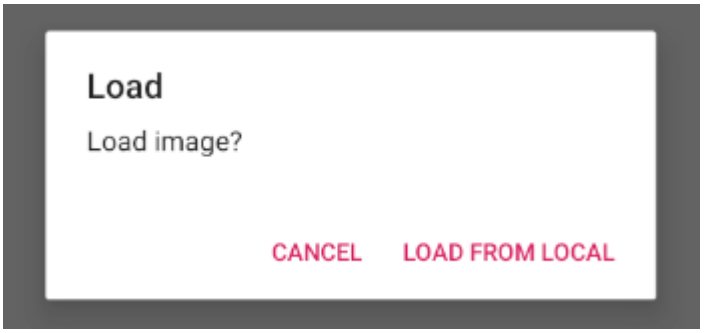

Unable to Crop: Not Implemented


Unable to Convert: No current Image


Unable to Upload: Please Load Local Image


Unable to Delete: No current Image


Unable to Favourite: Not Implemented




```
// if device has camera, create temporary file and take photo
fab.setOnClickListener((v) → {
        if(!MainActivity.this.getPackageManager().hasSystemFeature(PackageManager.FEATURE_CAMERA))
        {
            Toast.makeText( context: MainActivity.this, text: "Unable to Capture new Image - This device has no Camera",
                    Toast.LENGTH_SHORT).show();
        }
        else
        {
            Intent cameraIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
            if (cameraIntent.resolveActivity(getPackageManager()) != null) {
                File photoFile = null;
                try
                {
                    photoFile = createImageFile();
                }
                catch (IOException e)
                {
                    Log.w( tag: "Fab",  msg: "createImageFile:failure");
                    Toast.makeText( context: MainActivity.this,  text: "Error Occurred: " + e.getMessage(),
                            Toast.LENGTH_SHORT).show();
                }
                if (photoFile != null) {
                    Uri photoURI = FileProvider.getUriForFile( context: MainActivity.this,
                            authority: "com.teesside.yellowann.provider", photoFile);
                    cameraIntent.putExtra(MediaStore.EXTRA_OUTPUT, photoURI);
                    startActivityForResult(cameraIntent, REQUEST_IMAGE_CAPTURE);
                }
            }
        }
});
```

```
// initialise LoadFragment with populated list of available cloud images
private void getImagesList()
{
    try
    {
        final LoadFragment load = new LoadFragment();

        Bundle bundle = new Bundle();
        bundle.putSerializable("arrayList", list);
        load.setArguments(bundle);
        new Handler().post(() → {
                getFragmentManager().beginTransaction().replace(R.id.fragment_container,
                        load).addToBackStack(null).commit();
            });
    }
    catch (Exception e)
    {
        Log.w( tag: "getImagesList",  msg: "uploadToCloud.uploadTask:failure", e);
        Toast.makeText(getActivity(),  text: "Error Occurred: "
                + e.getMessage(), Toast.LENGTH_SHORT).show();
    }
}
```

## 5. FUTURE DEVELOPMENT

- Implement home fragment recent works scrollable listview
- Implement favourite work functionality
- Tidy up the download fragment to make it user friendly – a preview of the image and a more meaningful title
- Implement Pre-processing for the neural network that would arrange the text into optimal recognition situation – left-aligning the words, scaling to 173x73

for each word, optimising the contrast and identifying all words in the image and arranging them individually(github.com. 2018. githubharald/WordSegmentation: Segmentation of a text-line into words)[38]

- Implement the Crop Image functionality to the image fragment
- Implement 'CTCWordBeamSearch'(github.com. 2019. Connectionist Temporal Classification (CTC) decoder with dictionary and language model for TensorFlow)[39] which has a significantly increased accuracy
- Implement Azure Text Analytics Cognitive Services

# 6. RETROSPECTIVE

Looking back on this project, I started with a good grounding of research, however, I jumped into the actual application without much actual design planning. Because of this, development was slower than it would otherwise have been with adequate planning.

Implementation of Python code through Chaquopy was not my original intended method for the neural network, however, it proved the most viable method for this application due to the development time constraints.

Ideally, constructing and training my own neural network would've been preferable. Ideally this neural network would have been in python, as research has proven it to be in predominant use for A.I. development – meaning much more documentation, examples and guides to learn from, making a much easier entry into neural network development.

Implementing Chaquopy with Harald Scheidl's 'SimpleHTR' was intended initially to be a proof of concept that Chaquopy could do what android studio alone could not – utilise Python scripts on Android. As a proof of concept, it is a success – but implementing it took over two weeks, which was much too long to enable much more development afterwards. Given the opportunity, I would expand the application to include preprocessing and word beam search – and given even more time, to write and train the neural networks myself.

With more planning before starting development, I feel I could have more functionality implemented, and have investigated the issues that the application sometimes experiences with memory because of so many fragments hanging on the main activity, and tidied some of the interfaces that are not user friendly.

# REFERENCES


1. Harald Scheidl

   Scheidl, H. (2019). *githubharald - Overview*. [online] GitHub.

   Available at: https://github.com/githubharald

   [Accessed May 2019].

2. Chaquopy

   Smith, M. (2018). *The easiest way to use Python in your Android app*. [online] Chaquopy.

   Available at: https://chaquo.com/chaquopy/

   [Accessed May 2019].

3. The Guardian

   Busby, M. (2017). *Cambridge considers typed exams as handwriting worsens*. [online] the Guardian.

   Available at: https://www.theguardian.com/education/2017/sep/09/cambridge-considers-typed-exams-as-handwriting-worsens

   [Accessed May 2019].

4. Dyslexia

   Dyslexiahelp.umich.edu. (2019). *Frequently Asked Questions*. [online]

   Available at: http://dyslexiahelp.umich.edu/answers/faq

   [Accessed May 2019].

5. Times Higher Education

   Bothwell, E. (2017). *Pen and paper 'beats computers for retaining knowledge'*. [online] Times Higher Education (THE).

   Available at: https://www.timeshighereducation.com/news/pen-and-paper-beats-computers-retaining-knowledge

   [Accessed May 2019].

6. Optical Character Recognition

   SearchContentManagement. (2019). *What is OCR (optical character recognition)? - Definition from WhatIs.com*. [online]

   Available at: https://searchcontentmanagement.techtarget.com/definition/OCR-optical-character-recognition

   [Accessed May 2019].

7. Google Play

   Play.google.com. (2019). [online]

   Available at: https://play.google.com/store/searchq=optical%20text%20recognition&c=apps&hl=en_GB

   [Accessed May 2019].

8. Graphology

The British Institute of Graphologists. (2019). *What is Graphology? - The British Institute of Graphologists*. [online]

Available at: https://www.britishgraphology.org/about-british-institute-of-graphologists/what-is-graphology

[Accessed May 2019].

9. Off-line Recognition

En.wikipedia.org. (2019). *Handwriting recognition*. [online]

Available at: https://en.wikipedia.org/wiki/Handwriting_recognition#Results_since_2009

[Accessed May 2019].

10. Handwriting Styles

Nottingham.ac.uk. (2019). *Handwriting Styles - The University of Nottingham*. [online]

Available at: https://www.nottingham.ac.uk/manuscriptsandspecialcollections/researchguidance/medievaldocuments/handwritingstyles.aspx

[Accessed May 2019].

11. K-Nearest Neighbours Algorithm

Brownlee, J. (2016). *K-Nearest Neighbors for Machine Learning*. [online] Machine Learning Mastery.

Available at: https://machinelearningmastery.com/k-nearest-neighbors-for-machine-learning/

[Accessed May 2019].

12. Levenshtein Distance

Barbar, N. (2018). *The Levenshtein Distance Algorithm - DZone Big Data*. [online] dzone.com.

Available at: https://dzone.com/articles/the-levenshtein-algorithm-1

[Accessed May 2019].

13. Machine Learning

Varone, M., Mayer, D. and Melegari, A. (n.d.). *What is Machine Learning?* [online] Expertsystem.com.

Available at: https://www.expertsystem.com/machine-learning-definition/

[Accessed May 2019].

14. Pros and Cons of Neural Networks

Subscription.packtpub.com. (n.d.). *Pros and cons of neural networks*. [online]

Available at: https://subscription.packtpub.com/book/big_data_and_business_intelligence/9781788397872/1/ch01lvl1sec27/pros-and-cons-of-neural-networks

[Accessed May 2019].

15. Backpropagation

Techopedia.com. (n.d.). *What is Backpropagation?* [online]

Available at: https://www.techopedia.com/definition/17833/backpropagation

[Accessed May 2019].

16. Object Orientated Programming

    Techopedia.com. (n.d.). *What is Object-Oriented?* [online]

    Available at: https://www.techopedia.com/definition/3233/object-oriented

    [Accessed May 2019].

17. Functional Programming

    Techopedia.com. (n.d.). *What is Functional Programming?* [online]

    Available at: https://www.techopedia.com/definition/13624/functional-programming

    [Accessed May 2019].

18. Imperative Programming

    Techopedia.com. (n.d.). *What is Procedural Programming?* [online]

    Available at: https://www.techopedia.com/definition/21481/procedural-programming

    [Accessed May 2019].

19. Logic Programming

    Encyclopedia.com. (2004). *logic programming languages*. [online]

    Available at: https://www.encyclopedia.com/computing/dictionaries-thesauruses-pictures-and-press-releases/logic-programming-languages

    [Accessed May 2019].

20. Programming Languages for Artificial Intelligence

    Nautiyal, D. (n.d.). *Top 5 best Programming Languages for Artificial Intelligence field*. [online] GeeksforGeeks.

    Available at: https://www.geeksforgeeks.org/top-5-best-programming-languages-for-artificial-intelligence-field/

    [Accessed May 2019].

21. Most Popular AI Programming Languages

    Existek.com. (2018). *AI Programming: 5 Most Popular AI Programming Languages*. [online]

    Available at: https://existek.com/blog/ai-programming-and-ai-programming-languages/

    [Accessed May 2019].

22. AI Frameworks

    Gupta, B. (n.d.). *What is a Machine Learning Framework & 10 that you need to know* [online] Analytics India Magazine

    Available at: https://www.analyticsindiamag.com/machine-learning-framework-10-need-know/

    [Accessed May 2019].

23. MxNet

    Gupta, M. (2017). *MxNet: What is it and How to Get Started*. [online] becominghuman.ai

Available at: https://becominghuman.ai/mxnet-what-is-it-and-how-to-get-started-52efd9cb52cf

[Accessed May 2019].

24. CNTK

Docs.microsoft.com. (2017). *The Microsoft Cognitive Toolkit - Cognitive Toolkit - CNTK*. [online]

Available at: https://docs.microsoft.com/en-us/cognitive-toolkit/

[Accessed May 2019].

25. Comparison of AI Frameworks

Skymind. (n.d.). *Comparison of AI Frameworks*. [online]

Available at: https://skymind.ai/wiki/comparison-frameworks-dl4j-tensorflow-pytorch

[Accessed May 2019].

26. BNNS

developer.apple.com. (n.d.). BNNS | Apple Developer Documentation. [online]

Available at: https://developer.apple.com/documentation/accelerate/bnns

[Accessed May 2019].

27. NNAPI

developer.android.com. (n.d.). Neural Networks API | Android NDK | Android Developers. [online]

Available at: https://developer.android.com/ndk/guides/neuralnetworks

[Accessed May 2019].

28. Jython

Smith, M. 2018. JythonFaq/GeneralInfo. [online] wiki.python.org

Available at: https://wiki.python.org/jython/JythonFaq/GeneralInfo#What_is_Jython.3F

[Accessed May 2019].

29. PyPy

pypy.org. (n.d.). PyPy - What is PyPy?. [online]

Available at: https://pypy.org/features.html

[Accessed May 2019].

30. Password Visibility

developer.android.com. (n.d.). TextInputLayout | Android Developers. [online]

Available at:
https://developer.android.com/reference/android/support/design/widget/TextInputLayout#setpasswordvisibilitytoggleenabled

[Accessed May 2019].

31. Implementing Chaquopy

chaquopy.com. 2019. Android - Chaquopy 6.2. [online]

Available at: https://chaquo.com/chaquopy/doc/current/android.html

[Accessed May 2019].

32. Chaquopy Licensing

chaquopy.com. 2019. Licensing - Chaquopy. [online]

Available at: https://chaquo.com/chaquopy/license/

[Accessed May 2019].

33. SimpleHTR

github.com. 2019. Handwritten Text Recognition (HTR) system implemented with TensorFlow. [online]

Available at: https://github.com/githubharald/SimpleHTR

[Accessed May 2019].

34. IAM Handwriting Database

www.fki.inf.unibe.ch. 2019. IAM Handwriting Database. [online]

Available at: http://www.fki.inf.unibe.ch/databases/iam-handwriting-database

[Accessed May 2019].

35. SimpleHTR Requirements

github.com. 2019. SimpleHTR/requirements.txt. [online]

Available at: https://github.com/githubharald/SimpleHTR/blob/master/requirements.txt

[Accessed May 2019].

36. Android Spellcheck

developer.android.com. 2019. Spell checker framework | Android developers. [online]

Available at: https://developer.android.com/guide/topics/text/spell-checker-framework

[Accessed May 2019].

37. Azure Text Analytics Cognitive Services

docs.microsoft.com. 2019. Quickstart: Using Java to call the Text Analytics API - Azure Cognitive Services | Microsoft Docs. [online]

Available at: https://docs.microsoft.com/en-gb/azure/cognitive-services/text-analytics/quickstarts/java

[Accessed May 2019].

38. Word Segmentation

github.com. 2018. githubharald/WordSegmentation: Segmentation of a text-line into words. [online]

Available at: https://github.com/githubharald/WordSegmentation

[Accessed May 2019].

39. CTC Word Beam Search

github.com. 2019. Connectionist Temporal Classification (CTC) decoder with dictionary and language model for TensorFlow. [online]

Available at: https://github.com/githubharald/CTCWordBeamSearch

[Accessed May 2019].

# BIBLIOGRAPHY


stackoverflow.com. 2011. android - how to check edittext's text is email address or not? [ONLINE]

Available at: https://stackoverflow.com/questions/6119722/how-to-check-edittexts-text-is-email-address-or-not. [Accessed 2019].

examples.javacodegeeks.com. 2013. Validate Password with Java Regular Expression example | Examples Java Code Geeks - 2019 [ONLINE]

Available at: https://examples.javacodegeeks.com/core-java/util/regex/matcher/validate-password-with-java-regular-expression-example/. [Accessed 2019].

youtube.com. 2019. Android Firebase Send a User Verification Email (Explained) [ONLINE]

Available at: https://www.youtube.com/watch?v=06YKlMdWyMM. [Accessed 2019].

support.google.com. 2019. Customize account management emails and SMS messages - Firebase Help. [ONLINE]

Available at: https://support.google.com/firebase/answer/7000714?authuser=0. [Accessed 2019].

firebase.google.com. (n.d.). Passing State in Email Actions | Firebase. [ONLINE]

Available at: https://firebase.google.com/docs/auth/android/passing-state-in-email-actions?authuser=0. [Accessed 2019].

youtube.com. 2018. Android Firebase Send a Password Reset Email (Explained). [ONLINE]

Available at: https://www.youtube.com/watch?v=t8vUdt1eEzE. [Accessed 2019].

stackoverflow.com. 2016. java - Fill a textview from other activity. [ONLINE]

Available at: https://stackoverflow.com/questions/37236984/fill-a-textview-from-other-activity. [Accessed 2019].

developer.android.com. (n.d.). Create app icons with Image Asset Studio | Android Developers. [ONLINE]

Available at: https://developer.android.com/studio/write/image-asset-studio#create-adaptive. [Accessed 2019].

stackoverflow.com. 2018. java - Androidx Navigation View - 'setNavigationItemSelectedListener' Doesn't Work. [ONLINE]

Available at: https://stackoverflow.com/questions/53131591/androidx-navigation-view-setnavigationitemselectedlistener-doesnt-work. [Accessed 2019].

developer.android.com. (n.d.). Update UI components with NavigationUI | Android Developers. [ONLINE]

Available at: https://developer.android.com/guide/navigation/navigation-ui#add_a_navigation_drawer. [Accessed 2019].

developer.android.com. (n.d.). Menu resource | Android Developers. [ONLINE]

Available at: https://developer.android.com/guide/topics/resources/menu-resource. [Accessed 2019].

stackoverflow.com. 2015. How to add footer to NavigationView - Android support design library? [ONLINE]

Available at: https://stackoverflow.com/questions/30543605/how-to-add-footer-to-navigationview-android-support-design-library/30611068#30611068. [Accessed 2019].

developer.android.com. (n.d.). AlertDialog | Android Developers. [ONLINE]

Available at: https://developer.android.com/reference/android/app/AlertDialog. [Accessed 2019].

developer.android.com. (n.d.). Dialogs | Android Developers. [ONLINE]

Available at: https://developer.android.com/guide/topics/ui/dialogs.html. [Accessed 2019].

github.com. 2014. Fragment Navigation Drawer - codepath/android_guides Wiki. [ONLINE]

Available at: https://github.com/codepath/android_guides/wiki/Fragment-Navigation-Drawer. [Accessed 2019].

youtube.com. 2018. Navigation Drawer with Fragments Part 1 - MENU AND ACTIVITY THEME - Android Studio Tutorial. [ONLINE]

Available at: https://www.youtube.com/watch?v=fGcMLu1GJEc. [Accessed 2019].

youtube.com. 2018. Navigation Drawer with Fragments Part 2 - LAYOUT AND HAMBURGER ICON - Android Studio Tutorial. [ONLINE]

Available at: https://www.youtube.com/watch?v=zYVEMCiDcmY. [Accessed 2019].

youtube.com. 2018. Navigation Drawer with Fragments Part 3 - HANDLING MENU ITEM CLICKS - Android Studio Tutorial. [ONLINE]

Available at: https://www.youtube.com/watch?v=bjYstsO1PqI. [Accessed 2019].

stackoverflow.com. 2014. android - How to get Toolbar from fragment?. [ONLINE]

Available at: https://stackoverflow.com/questions/26998455/how-to-get-toolbar-from-fragment. [Accessed 2019].

stackoverflow.com. 2012. java - Android - How to create clickable listview?. [ONLINE]

Available at: https://stackoverflow.com/questions/13281197/android-how-to-create-clickable-listview. [Accessed 2019].

stackoverflow.com. 2016. android - How to Generate Random names for Images while storing in Internal Memory? [ONLINE]

Available at: https://stackoverflow.com/questions/34850286/how-to-generate-random-names-for-images-while-storing-in-internal-memory. [Accessed 2019].

airpair.com. 2019. Taking Pictures In An Android Fragment Using Intents. [ONLINE]

Available at: https://www.airpair.com/android/taking-pictures-android-fragment-intents. [Accessed 2019].

youtube.com. 2018. Populating ListView from Firebase Real-time Database Simple Steps (Firebase UI). [ONLINE]

Available at: https://www.youtube.com/watch?v=QZPzWNbjAV0. [Accessed 2019].

proandroiddev.com. 2018. Using TensorFlow on Android—step by step code explanation. [ONLINE]

Available at: https://proandroiddev.com/using-tensorflow-in-android-step-by-step-code-explanation-fee36c281f32. [Accessed 2019].

towardsdatascience.com. 2018. Build a Handwritten Text Recognition System using TensorFlow. [ONLINE]

Available at: https://towardsdatascience.com/build-a-handwritten-text-recognition-system-using-tensorflow-2326a3487cd5. [Accessed 2019].

quora.com. 2014. How would I get a neural network to work in an Android app after I have trained it in MATLAB? [ONLINE]

Available at: https://www.quora.com/How-would-I-get-a-neural-network-to-work-in-an-Android-app-after-I-have-trained-it-in-MATLAB. [Accessed 2019].

fki.inf.unibe.ch. 2019. IAM Handwriting Database - Computer Vision and Artificial Intelligence. [ONLINE]

Available at: http://www.fki.inf.unibe.ch/databases/iam-handwriting-database. [Accessed 2019].

developer.android.com. (n.d.). Add a Floating Action Button | Android Developers. [ONLINE]

Available at: https://developer.android.com/guide/topics/ui/floating-action-button. [Accessed 2019].

developer.android.com. (n.d.). Take photos | Android Develoeprs. [ONLINE]

Available at: https://developer.android.com/training/camera/photobasics. [Accessed 2019].

firebase.google.com. 2019. Upload Files on Android | Firebase. [ONLINE]

Available at: https://firebase.google.com/docs/storage/android/upload-files. [Accessed 2019].

stackoverflow.com. 2019. java - Unresolved class 'FileProvider'. [ONLINE]

Available at: https://stackoverflow.com/questions/54914229/unresolved-class-fileprovider. [Accessed 2019].

javatpoint.com. 2019. Android Popup Menu Example. [ONLINE]

Available at: https://www.javatpoint.com/android-popup-menu-example. [Accessed 2019].

code.tutsplus.com. 2012. Capture and Crop an Image with the Device Camera. [ONLINE]

Available at: https://code.tutsplus.com/tutorials/capture-and-crop-an-image-with-the-device-camera--mobile-11458. [Accessed 2019].

londatiga.net. 2011. How to Select and Crop Image on Android. [ONLINE]

Available at: http://www.londatiga.net/featured-articles/how-to-select-and-crop-image-on-android/. [Accessed 2019].

stackoverflow.com. 2013. checkbox - How to change checkbox tick color in android. [ONLINE]

Available at: https://stackoverflow.com/questions/20542693/how-to-change-check-box-tick-color-in-android. [Accessed 2019].

youtube.com. 2018. Pick Photos From Gallery and Crop Image Android Studio Tutorials. [ONLINE]

Available at: https://www.youtube.com/watch?v=oHo_tiRlXDQ. [Accessed 2019].

codinginfinite.com. 2019. Android Crop Image Using UCrop Library | Camera | Gallery. [ONLINE]

Available at: https://codinginfinite.com/android-image-ucrop-camera-gallery/. [Accessed 2019].

youtube.com. 2017. Firebase Storage - Upload and Retrieve Images - Part 1 - PREPARATIONS AND LAYOUT - Android Studio. [ONLINE]

Available at: https://www.youtube.com/watch?v=MfCiiTEwt3g. [Accessed 2019].

firebase.google.com. 2019. Downloading Images with FirebaseUI | Firebase. [ONLINE]

Available at: https://firebase.google.com/docs/storage/android/download-files#downloading_images_with_firebaseui. [Accessed 2019].

stackoverflow.com. 2017. How to get URLs of all images stored in firebase in Android? [ONLINE]

Available at: https://stackoverflow.com/questions/44062720/how-to-get-urls-of-all-images-stored-in-firebase-in-android. [Accessed 2019].

github.com. 2018. Uri downloadUrl = taskSnapshot.getDownloadUrl(); not working - Issue #46. [ONLINE]

Available at: https://github.com/udacity/and-nd-firebase/issues/46. [Accessed 2019].

stackoverflow.com. 2011. java - Populating a ListView using an ArrayList? [ONLINE]

Available at: https://stackoverflow.com/questions/5070830/populating-a-listview-using-an-arraylist. [Accessed 2019].

youtube.com. 2017. Reading Data using a ValueEventListener - Android Firebase. [ONLINE]

Available at: https://www.youtube.com/watch?v=2duc77R4Hqw. [Accessed 2019].

github.com. 2019. githubharald/SimpleHTR: Handwritten Text Recognition (HTR) system implemented with TensorFlow. [ONLINE]

Available at: https://github.com/githubharald/SimpleHTR. [Accessed 2019].

github.com. 2018. klevis/DigitRecognizer: Java Convolutional Neural Network example for Hand Writing Digit Recognition. [ONLINE]

Available at: https://github.com/klevis/DigitRecognizer. [Accessed 2019].

github.com. 2018. klevis/DigitRecognizer: Java Convolutional Neural Network example for Hand Writing Digit Recognition. [ONLINE]

Available at: https://github.com/klevis/DigitRecognizer. [Accessed 2019].

stackoverflow.com. 2017. Running Python Script from Android Activity. [ONLINE]

Available at: https://stackoverflow.com/questions/47968040/running-python-script-from-android-activity. [Accessed 2019].

tensorflow.org. (n.d.). TensorFlow Lite | TensorFlow. [ONLINE]

Available at: https://www.tensorflow.org/lite. [Accessed 2019].

quora.com. 2018. How can I include a Python code in an Android app? Most of the ML algorithms are written in Python, so I was wondering, how are professional AI apps developed? [ONLINE]

Available at: https://www.quora.com/How-can-I-include-a-Python-code-in-an-Android-app-Most-of-the-ML-algorithms-are-written-in-Python-so-I-was-wondering-how-are-professional-AI-apps-developed. [Accessed 2019].

stackoverflow.com. 2010. How to handle ListView click in Android. [ONLINE]

Available at: https://stackoverflow.com/questions/2468100/how-to-handle-listview-click-in-android. [Accessed 2019].

androidinterview.com. 2014. Android Custom Listview with Image and Text using ArrayAdapter. [ONLINE]

Available at: https://www.androidinterview.com/android-custom-listview-with-image-and-text-using-arrayadapter/. [Accessed 2019].

chaquo.com. 2018. Chaquopy - Python SDK for Android. [ONLINE]

Available at: https://chaquo.com/chaquopy/. [Accessed 2019].

stackoverflow.com. 2019. java - Chaquopy in Android Studio module not found. [ONLINE]

Available at: https://stackoverflow.com/questions/55482207/chaquopy-in-android-studio-module-not-found. [Accessed 2019].

github.com. 2019. Android Studio Gradle Sync Issue - Issue #101. [ONLINE]

Available at: https://github.com/chaquo/chaquopy/issues/101. [Accessed 2019].

github.com. 2019. githubharald/CTCWordBeamSearch: Connectionist Temporal Classification (CTC) decoder with dictionary and language model for TensorFlow. [ONLINE]

Available at: https://github.com/githubharald/CTCWordBeamSearch. [Accessed 2019].

github.com. 2018. githubharald/WordSegmentation: Segmentation of a text-line into words. [ONLINE]

Available at: https://github.com/githubharald/WordSegmentation. [Accessed 2019].

developer.android.com. (n.d.). Build local unit tests | Android Developers. [ONLINE]

Available at: https://developer.android.com/training/testing/unit-testing/local-unit-tests. [Accessed 2019].

developer.android.com. (n.d.). Test your app | Android Developers. [ONLINE]

Available at: https://developer.android.com/studio/test. [Accessed 2019].

dyslexiahelp.umich.edu. 2019. Frequently Asked Questions | Dyslexia Help at the University of Michigan. [ONLINE]

Available at: http://dyslexiahelp.umich.edu/answers/faq. [Accessed 2019].

nottingham.ac.uk. 2019. Handwriting Styles - The University of Nottingham. [ONLINE]

Available at: https://www.nottingham.ac.uk/manuscriptsandspecialcollections/researchguidance/medievaldocuments/handwritingstyles.aspx. [Accessed 2019].

scribbr.com. 2019. How to Write Research Methodology in Four Steps | With Examples. [ONLINE]

Available at: https://www.scribbr.com/dissertation/methodology/. [Accessed 2019].

en.wikipedia.org. 2019. Optical character recognition. [ONLINE]

Available at: https://en.wikipedia.org/wiki/Optical_character_recognition. [Accessed 2019].

gs.statcounter.com. 2019. Mobile Operating System Market Share Worldwide. [ONLINE]

Available at: http://gs.statcounter.com/os-market-share/mobile/worldwide/2018. [Accessed 2019].

gs.statcounter.com. 2019. Mobile Vendor Market Share Worldwide. [ONLINE]

Available at: http://gs.statcounter.com/vendor-market-share/mobile/worldwide/2018. [Accessed 2019].

gs.statcounter.com. 2019. Tablet Vendor Market Share Worldwide. [ONLINE]

Available at: http://gs.statcounter.com/vendor-market-share/tablet/worldwide/2018. [Accessed 2019].

expertsystem.com. (n.d.). What is Machine Learning? A definition. [ONLINE]

Available at: https://www.expertsystem.com/machine-learning-definition/. [Accessed 2019].

digitaltrends.com. 2019. What is an artificial neural network? Here's everything you need to know. [ONLINE]

Available at: https://www.digitaltrends.com/cool-tech/what-is-an-artificial-neural-network/. [Accessed 2019].

tensorflow.org. (n.d.). TensorFlow. [ONLINE]

Available at: https://www.tensorflow.org/. [Accessed 2019].

skymind.ai. (n.d.). Comparison of AI Frameworks. [ONLINE]

Available at: https://skymind.ai/wiki/comparison-frameworks-dl4j-tensorflow-pytorch. [Accessed 2019].

techopedia.com. 2019. What is Object-Oriented? - Definition from Techopedia. [ONLINE]

Available at: https://www.techopedia.com/definition/3233/object-oriented. [Accessed 2019].

techopedia.com. 2019. What is Functional Programming? - Definition from Techopedia. [ONLINE]

Available at: https://www.techopedia.com/definition/13624/functional-programming. [Accessed 2019].

techopedia.com. 2019. What is Procedural Programming? - Definition from Techopedia. [ONLINE]

Available at: https://www.techopedia.com/definition/21481/procedural-programming. [Accessed 2019].

geeksforgeeks.org. (n.d.). Top 5 best Programming Languages for Artificial Intelligence field. [ONLINE]

Available at: https://www.geeksforgeeks.org/top-5-best-programming-languages-for-artificial-intelligence-field/. [Accessed 2019].

analyticsindiamag.com. 2019. What is Machine Learning Framework. [ONLINE]

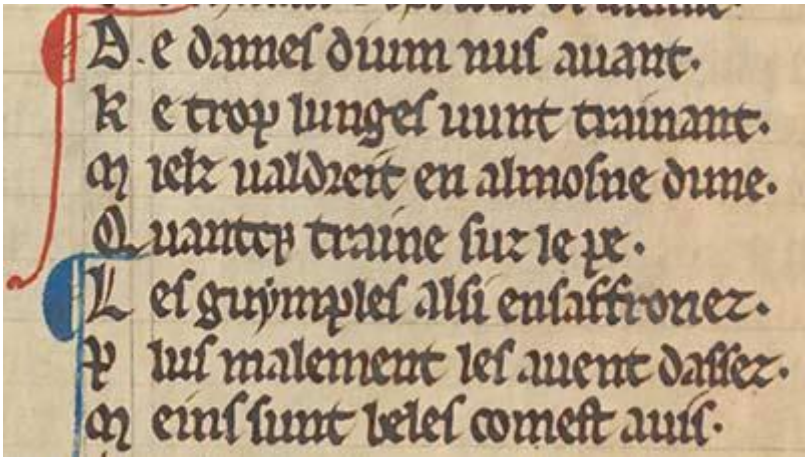
Available at: https://www.analyticsindiamag.com/machine-learning-framework-10-need-know/. [Accessed 2019].

# APPENDICES

A. Textura Script

Nottingham.ac.uk. (2019). *Handwriting Styles - The University of Nottingham*. [online]

Available at: https://www.nottingham.ac.uk/manuscriptsandspecialcollections/researchguidance/medievaldocuments/handwritingstyles.aspx .



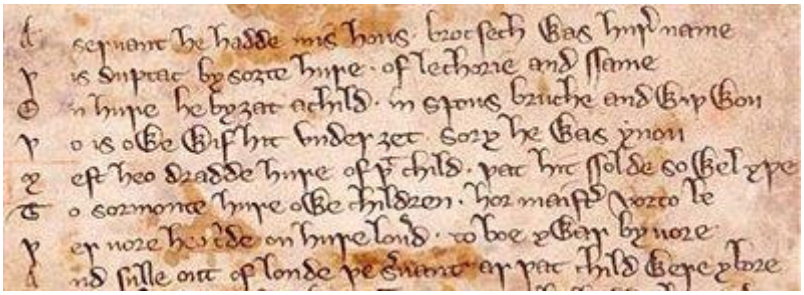
Textura Quadrata

Textura Rotunda

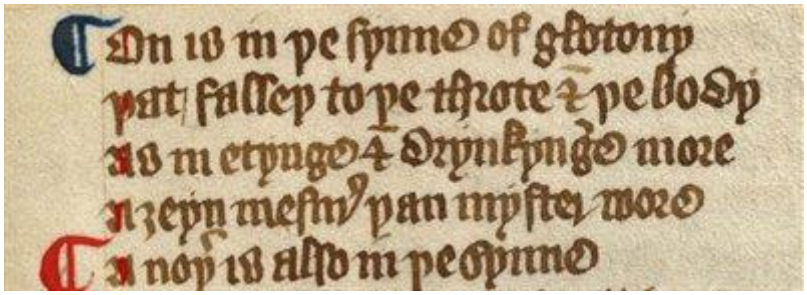
B. Cursiva Script

Nottingham.ac.uk. (2019). *Handwriting Styles - The University of Nottingham.* [online]

Available at: https://www.nottingham.ac.uk/manuscriptsandspecialcollections/researchguidance/medievaldocuments/handwritingstyles.aspx .



Cursiva Angelicana



Cursiva Angelicana Formata