­­­

Optical Character Recognition

Using Python Programming

CONTENTS:

1. Abstract
2. Introduction
3. What is OCR?
4. Neural Network
5. Question Answers
6. Problem Definition
7. Solution Statement
8. Scope
9. Methodology
10. Conclusion
11. Bibliography and Reference

Background

The aim of this project is to develop an Optical Character Recognition (OCR) in Python Programming language for the conversion of image data to characters. OCR is a method of extracting text from images, which can be scanned copies or photographs of printed or handwritten documents.

This project is divided into two parts, first an OCR application is developed that recognizes text from image data and visualizes it by converting it into 2D images i.e. images of text will be preprocessed and then features such as open areas, closed shapes, diagonal lines, slanting and intersections will be extracted. Then in the second part, an OCR engine is created using ANN where image dataset is used as an input which will split it into training and test data. Of the entire dataset, 70% of the data will be training data(labeled) and the rest 30% will be the test data (unlabeled).

Thus feature set and image dataset will be fed to an Artificial Neural Network (ANN) for character recognition.

­­­­

INTRODUCTION

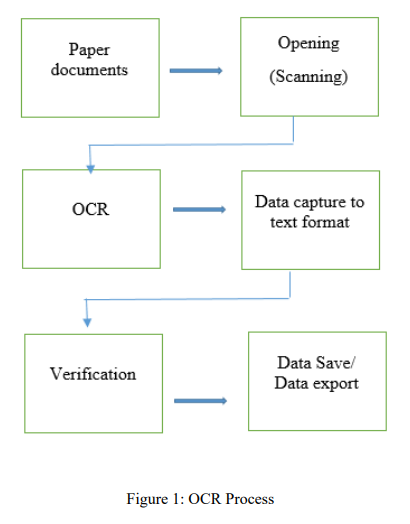
In the running world, there is a growing demand for the software systems to recognize characters in computer system when information is scanned through paper documents as we know that we have number of newspapers and books which are in printed format related to different objects.

These days there is a huge demand in “storing the information available in these paper documents into a computer storage disk and then later reusing this information for searching process.

One simple way to store these paper documents into computer system is to first scan the documents and then store them as IMAGES. But to reuse this information, it is very difficult to read the individual contents and searching the contents from these documents line-by-line and word-by-word. The reason for this difficulty is the font characteristics of the characters in paper documents are different to font of the characters in computer system. As a result, computer is unable to recognize the characters while reading them.

This is where we use OCR- Optical Character Recognition.

It uses our computer’s smarts to recognize letter shapes in an image or scanned document and turn the into digital text, which can be copied and edited as needed.

­­

OPTICAL CHARACTER RECOGNITION (OCR)

OCR stands for optical character recognition, which is a field of computer science that recognizing image-based text from photos and transforms it to real digital character.

OCR is a technology that enables you to convert different types of documents such as scanned paper documents, PDF files or images captured by a digital camera into editable and searchable data. OCR works like human ability in the brain to recognize the letters, numbers and symbols. OCR can read both handwritten and printed text. The performance of OCR is directly related to quality of input documents and pictures.

4444

NEURAL NETWORK

A neural network is defined as a computing system that is made up of several, simple, interconnected processing elements, which process information by their dynamic state response to external inputs.

Neural networks uses concepts borrowed from an understanding of human brains in order to model arbitrary functions.

It is a machine that is designed to model the way in which the brain performs a particular task or function of interest; the network is usually implemented by using electronic components or is simulated in software on a digital computer.

It resembles the brain in two respects:

1. Knowledge is acquired by the network from its environment through a learning process.

2. Interneuron connection strengths, known as synaptic weights, are used to store the acquired knowledge.

An Artificial Neural Network (ANN) models the relationship between a set of input signals and an output signal using a model derived from our understanding of how a biological brain responds to stimuli from sensory inputs.

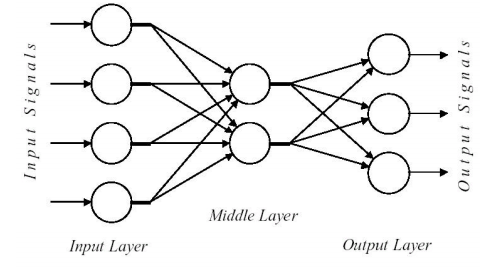
Mostly, ANNs have learning rules that modifies the weighted connections according to the input patterns that it is presented with.

1. What advantages does OCR offer?

* OCR increases the efficiency and effectiveness of while working with scanned documents or image type of files.
* Time Saving- Saves a lot of time when using a digital file rather than paper documents
* Fully Editable & Flexible Documents- Documents becomes editable with OCR. As we can convert files to MS Word and any other editable digital formats, which in further can be copied and pasted to another file.

1. What are the unique features of the Python Programming Language?

* Easy to learn and use- It is developer-friendly and high-level programming language.
* Interpreted Language-Interpreter executes the code line by line at a time thus making debugging easy and thus suitable for beginners.
* Free and open source- Python language is freely available at official web address. The source code is also available.
* Object-Oriented Language- Python supports object-oriented language and concepts of classes and objects come into existence.
* Large Standard Library-Python has large and broad library that provides rich set of modules and functions for rapid application development.
* GUI Programming Support- Graphical User Interfaces can be developed using python.



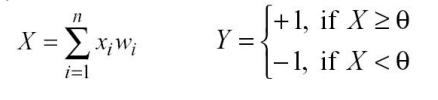
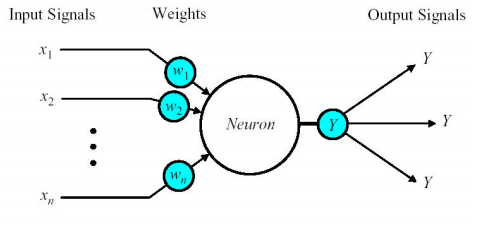
1. What are the parts of neural network and how do they function?

* Neurons:

An Artificial Neural Network (ANN) consists of a number of very simple processors, also called neurons, which are analogous to the biological neuron in the brain.

The neurons are connected by weighted links passing signals from one neuron to another.

Architecture of a typical artificial neural network



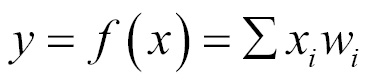
The neuron as a simple computing element.

The neuron computes the weighted sum of the input signals and compares the results with a threshold value, è.

If the net input is less than the threshold, the neuron output is –1. But if the net input is greater than or equal to the threshold, the neuron becomes activated and its output attains a value +1.

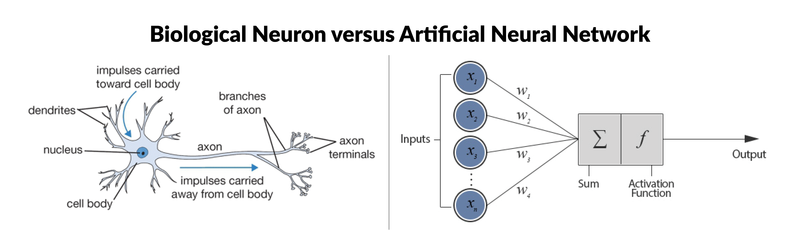
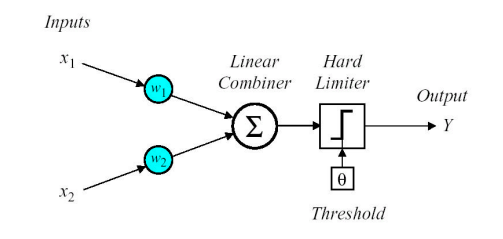
The neuron uses the following transfer or activation function:

This type of activation function is called a sign function.

0

* Input Layer: *:*It basically contains raw data
* Output Layer: The output signal is transmitted through the neuron’s outgoing connection.The outgoing connection splits into a number of branches that transmit the same signal.The outgoing branches terminate at the incoming connections of other neurons in the network.
* Hidden Layer *:*Hidden layers fine-tune the input weightings until the neural network’s margin of error is minimal. It is hypothesized that hidden layers extrapolate salient features in the input data that have predictive power regarding the outputs.
* Weights: Weights in an ANN are the most important factor in converting an input to impact the output. Weight is multiplied to the input to add up to form the output. Weights are numerical parameters which determine how strongly each of the neurons affects the other.

* Bias: Bias is like the intercept added in a linear equation. It is an additional parameter which is used to adjust the output along with the weighted sum of the inputs to the neuron. 
* An activation function: which transforms a neuron's net input signal into a single output signal to be broadcasted further in the network.



1. How does the training of neural networks happen?

Artificial neural networks are relatively crude electronic networks of "neurons" based on the neural structure of the brain.  They process records one at a time, and "learn" by comparing their classification of the record (which, at the outset, is largely arbitrary) with the known actual classification of the record.  The errors from the initial classification of the first record is fed back into the network, and used to modify the networks algorithm the second time around, and so on for many iterations.

Roughly speaking, a neuron in an artificial neural network is

1.) A set of input values (xi) and associated weights (wi)

 2.) A function (g) that sums the weights and maps the results to an output (y).

1. What advantages do ANN offer over conventional techniques?

Ans. Depending on the nature of the application and the strength of the internal data patterns you can generally expect a network to train quite well. This applies to problems where the relationships may be quite dynamic or non-linear. ANNs provide an analytical alternative to conventional techniques which are often limited by strict assumptions of normality, linearity, variable independence etc. Because an ANN can capture many kinds of relationships it allows the user to quickly and relatively easily model phenomena which otherwise may have been very difficult or impossible to explain otherwise.

Problem Definition:

In this project we will explore the process of OCR through ANNs.

**Part 1**

We will develop and test an OCR application that recognizes text from image data and visualizes it by converting it into 2D images.

**Part 2**

we will create an OCR engine using ANN and use the image dataset provided as an input. Then will split into training and test data. Of the entire dataset ,70% of the data will be the training data (labelled) and the rest data will be the test data(unlabelled).

Prerequisites

* A laptop/desktop connected to the scanner /camera/mobile phone
* Windows 7 (or above) with python 3.7(or above) under the Anaconda environment
* Microsoft office/Adobe Reader/Notepad+
* Admin permissions and internet connection for packages installation and updates

Modules and Packages

* OS- The OS module in python provides functions for interacting with the operating system. OS, comes under Python's standard utility modules. This module provides a portable way of using operating system dependent functionality. The \*os\* and \*os.path\* modules include many functions to interact with the file system.
* SYS-The sys module provides information about constants, functions and methods of the Python interpreter. dir(system) gives a summary of the available constants, functions and methods.
* cv2- OpenCV is a library of programming functions mainly aimed at real-time computer vision.

To install opencv:

1. Install Anaconda- anaconda.org
2. Install OpenCV- Python to Anaconda - <https://stackoverflow.com/questions/23119413/how-do-i-install-python-opencv-through-conda>
3. Download OpenCV Package -opencv.org
4. Copy and Paste the cv2.pyd file –
5. The Anaconda Site-packages directory (e.g. C:\Users\AishC\Anaconda\Lib\site-packages in my case) contains the Python packages that you may import.

Copy the cv2.pyd file

from ‘C:\opencv\build\python\3.6\x64’

Paste into

‘C:\Users\AishC\Anaconda\Lib\site-packages

1. Set Environmental Variables-

Right-click on "My Computer" (or "This PC" on Windows 8.1) -> left-click Properties -> left-click "Advanced" tab -> left-click "Environment Variables..." button.

|  |  |  |
| --- | --- | --- |
| 64-bit | OPENCV\_DIR | C:\opencv\build\x64\vc14 |

Append %OPENCV\_DIR%\bin to the User Variable PATH to

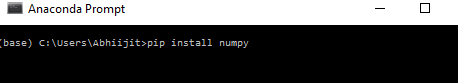
C:\Users\AishC\Anaconda;C:\Users\AishC\Anaconda\Scripts;%OPENCV\_DIR%\bin

1. Test to confirm:

import cv2

print cv2.\_\_version\_\_

* Numpy- NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.



* Neurolab - Neurolab is a simple and powerful Neural Network Library for Python. Contains based neural networks, train algorithms and flexible framework to create and explore other neural network types.

Solution Statement:

We will implement OCR in the python language using ANN. We will develop a program that can read the image data and predict the characters. The program should also be able to determine if a predicted character matches the actual one or not.

We will follow these steps:

1) Loading the image data as input.

2) Pre-processing.

3) Extracting Features.

4) Recognizing characters.

5) Producing the output.

6) Analyzing the results.

Methodology

Part 1:

**We will develop and test an OCR application that recognizes text from image data and visualizes it by converting it into 2D images.**

Prerequisites:

1.Download letter.data from <http://ai.stanford.edu/~btaskar/ocr>

2.Launch Anaconda Spyder

3.Define the path for input\_file =”C:\letter.data”

4.Import the packages:

import os

import sys

import cv2

import numpy as np

Define the input file and visualization parameters

input\_file = 'C:\letter.data'

img\_resize\_factor = 12

start = 6

end = -1

height, width = 16, 8

# Iterate until the user presses the Esc key

with open(input\_file, 'r') as f:

for line in f.readlines():

# Read the data

data = np.array([255 \* float(x) for x in line.split('\t')[start:end]])

# Reshape the data into a 2D image

img = np.reshape(data, (height, width))

# Scale the image and display iy

img\_scaled = cv2.resize(img, None, fx=img\_resize\_factor, fy=img\_resize\_factor)

cv2.imshow('Image', img\_scaled)

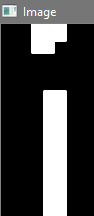
# Check if the user pressed the Esc key

c = cv2.waitKey()

if c == 27:

break

**Output:**

**      **

**Part 2:**

**we will create an OCR engine using ANN and use the image dataset provided as an input. Then will split into training and test data. Of the entire dataset ,70% of the data will be the training data (labelled) and the rest data will be the test data(unlabelled)**

Import the packages in python file:

import numpy as np

import neurolab as nl

# Define the input file & the number of datapoints to be loaded from the input file

input\_file = 'C:\letter.data'

num\_datapoints = 50

# String containing all the distinct characters

orig\_labels = 'omandig'

# Compute the number of distinct characters

num\_orig\_labels = len(orig\_labels)

# Define the training and testing parameters

num\_train = int(0.7 \* num\_datapoints)

num\_test = num\_datapoints - num\_train

# Define the dataset extraction parameters

start = 6

end = -1

# Creating the dataset

data = []

labels = []

with open(input\_file, 'r') as f:

for line in f.readlines():

# Split the current line tabwise

list\_vals = line.split('\t')

# Check if the label is in our ground truth

# labels. If not, we should skip it.

if list\_vals[1] not in orig\_labels:

continue

# Extract the current label and append it

# to the main list

label = np.zeros((num\_orig\_labels, 1))

label[orig\_labels.index(list\_vals[1])] = 1

labels.append(label)

# Extract the character vector and append it to the main list

cur\_char = np.array([float(x) for x in list\_vals[start:end]])

data.append(cur\_char)

# Exit the loop once the required dataset has been created

if len(data) >= num\_datapoints:

break

# Convert the data and labels to numpy arrays

data = np.asfarray(data)

labels = np.array(labels).reshape(num\_datapoints, num\_orig\_labels)

# Extract the number of dimensions

num\_dims = len(data[0])

# Create a feedforward neural network

nn = nl.net.newff([[0, 1] for \_ in range(len(data[0]))],

[128, 16, num\_orig\_labels])

# Set the training algorithm to gradient descent

nn.trainf = nl.train.train\_gd

# Train the network

error\_progress = nn.train(data[:num\_train,:], labels[:num\_train,:],

epochs=10000, show=100, goal=0.01)

# Predict the output for test inputs

print('\nTesting on unknown data:')

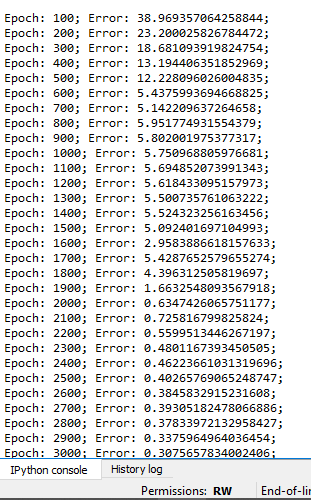
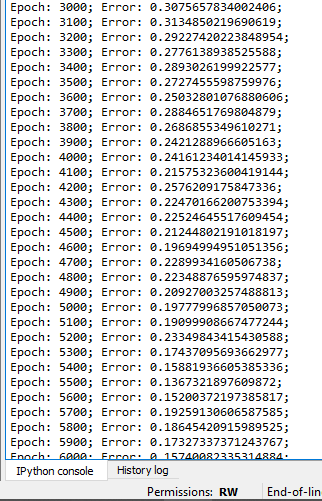
predicted\_test = nn.sim(data[num\_train:, :])

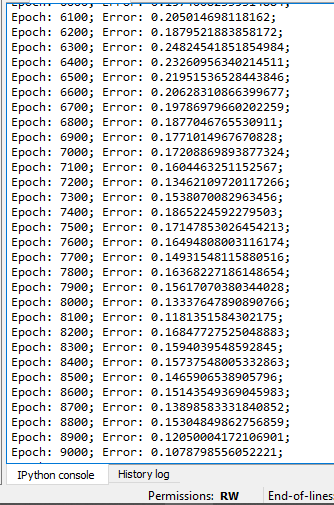
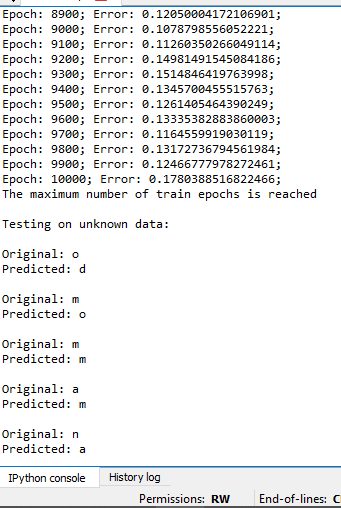
for i in range(num\_test):

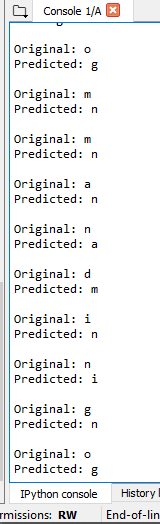
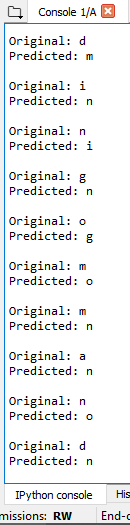
print('\nOriginal:', orig\_labels[np.argmax(labels[i])])

print('Predicted:', orig\_labels[np.argmax(predicted\_test[i])])

Output:

Conclusion

OCR technology is used to convert virtually any kind of images containing written text(hyped, handwritten or printed) into machine-readable text data.

In the first part, an OCR application is developed that recognizes text from image data and visualizes it by converting it into 2D images i.e. images of text will be preprocessed and then features such as open areas, closed shapes, diagonal lines, slanting and intersections will be extracted.

Then in the second part, an OCR engine is created using ANN where image dataset is used as an input which will split it into training and test data. Of the entire dataset, 70% of the data will be training data(labeled) and the rest 30% will be the test data (unlabeled).

Only 2 out of the 15 outcomes were correctly predicted. Due to the sensitive of training size there is inaccuracy in data . The output was predicted inaccurate.

Bibliography:

<http://www.cvisiontech.com/reference/ocr/advantages-of-ocr.html>

<https://www.pearl-scan.co.uk/blog/advantages-of-ocr-optical-character-recognition/>

<http://www.aui.ma/sse-capstone-repository/pdf/The%20Optical%20Capture%20Recognition.pdf>

<https://www.javatpoint.com/python-features>

[file:///D:/Machine\_Learning\_with\_R.pdf](file:///D:\Machine_Learning_with_R.pdf)

<https://towardsdatascience.com/machine-learning-for-beginners-an-introduction-to-neural-networks-d49f22d238f9>

<https://adventuresinmachinelearning.com/wp-content/uploads/2017/07/An-introduction-to-neural-networks-for-beginners.pdf>

<http://dai.fmph.uniba.sk/courses/NN/haykin.neural-networks.3ed.2009.pdf>

<http://mathalope.co.uk/2015/05/07/opencv-python-how-to-install-opencv-python-package-to-anaconda-windows/>

<https://www.abbyy.com/en-eu/finereader/what-is-ocr/>