Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

CSE3013: ARTIFICIAL INTELLIGENCE

HANDWRITING ANALYSIS & GRAPHOLOGY using ML

NAME	REG NUMBER
Saurabh Kumar Jha	18BCE0197
Nitin Ranjan	18BCE0272
Shubham	18BCE0803

Under the guidance of

Dr. Harshita Patel,

SITE

LIST OF CONTENTS

Abstract 3

Problem Statement 3

Introduction 4

Literature Survey 5

Proposed Structure of Project 18

Pseudo code 18

Dataset used 20

PROPOSED WORK 20

A. Multi-class classification 20

B. Neural Network 20

C. Image OCR 21

D. CNN for symbol Recognition 21

E. A simpler neural network 23

F. Software integration 23

Experimentation and Result 23

A. Experimental Constraints 23

B. Results 24

CODE: 30

Conclusion 39

References 40

Abstract

Among all the unique characteristics of a human being, handwriting carries the richest information to gain the insights into the physical, mental and emotional state of the writer. Graphology is the art of studying and analysing handwriting, a scientific method used to determine a person's personality by evaluating various features from the handwriting. The prime features of handwriting such as the page margins, the slant of the alphabets, the baseline etc. can tell a lot about the individual. To make this method more efficient and reliable, introduction of machines to perform the feature extraction and mapping to various personality traits can be done. This compliments the graphologists, and also increases the speed of analysing handwritten samples. Various approaches can be used for this type of computer aided graphology. Through this project, we have tried to apply machine learning technique to implement automated handwriting analysis.

Keyword: Graphology, personality-trait, handwriting, feature-extraction, machine learning, Convolutional Neural Network, neural networks.

Problem Statement

In offline handwriting recognition, text is analyzed after being written. Information, such as pen stroke, pressure and speed of writing are analyzed. Here the text will be analyzed simultaneously and will be interpreted. It can also be digitalized presenting the use of paper and the possibility of losing the record. It will also save time as all the work is done by the software rather than the person. All these benefits are associated with the software. Mainly the benefit is that we are getting an analyzer as accurate as a professional psychologist.

The project is intended for all the persons who are technology savvy. Apart from this it is intended for project managers, examination testers, psychologist, researchers and software engineering professors. The scope for the product includes psychologist, researchers, people who want to understand their personality from handwriting.

Basically, this project is intended for every person who can write and wants to analyse their handwriting. Ultimately, the objective of the project is to take the first steps in this field of human psychology with Machine Learning with the hope that eventually this analyzer becomes as accurate as a professional psychologist.

Introduction

Technology serves many purposes. One of which is to act as an aid for education. However, there are two big problems with this idea given the lockdown that has been in place at several places all over.

1. Not many have access to computers.
2. The ones that have access to it spend way too much time on it.

So, it would be a good idea to create something that can integrate these two problems. So, an interface that converts written text to digital text will be a good idea.

Handwriting also termed as brain-writing is a useful measure in identifying the characteristic personality traits of an individual. Handwriting analysts also known as graphologists can examine an individual‟s handwriting to predict the personality traits of the writer. Automated handwriting analysis can be used to examine personal traits of candidates during interviews accurately as the accuracy of an analyst highly depends on his skill set. Also hiring a graphologist to analyse hundreds of samples for recruitment purpose will be time consuming and not be feasible economically. This work discusses about a method for analysing real world handwritten text samples with the aid of technology. The analysis is done for specific features of the sample for determining various characteristic behavioural traits of the person.

The first release of this project aims to distinguish between different people's personality through handwriting analysis. This software will be inputting an image of the handwriting of a person and then will tell what their personality is like. To achieve this, simple Neural Network and Back Propagation Algorithm will be used.
Further, research suggests that handwriting might serve as a strong means to determine a person's personality and mood. And so, using graphology which is a forensic study of handwriting analysis, we shall integrate this module in our interface too.

Literature Survey

[1] The authors define graphology and further argues the business needs and impacts of graphology, the previous attempts made at automating the same using various segregation techniques and a brief comparison between the same.

[2] The authors present a brief about graphology and use a sentiment-polarity based model. The authors present their framework where they create a four-phase model with four engines that carry out lexical analysis on words and then give sentiment scores on a normalised polarity scale (i.e. between -1 and 1) and store it in a student specific memory location.

[3] This research article was mostly used for understanding graphology itself. The author tries to explain why it is important to understand graphology, the class features of handwriting analysis, the features that should be looked for and the unique features that professionals try to look for. The author further explains how samples ought to be collected, why handwriting analysis is not treated to be a science and why forgery is always a possibility.

[4] The authors argue that graphology might serve as an excellent mean of introspection and self-growth. They collect 25 random samples and use standard graphological functions to segregate the data and then analyse them. They do mention that it is possible using fuzzy logic functions, however they use image processing. Finally, they compare the data with the one provided by a professional and find a high degree of similarity, thus establishing that it is possible to analyse handwriting and personalities digitally.

[5] One of the founding papers for modern study in graphology, James McNeal provides the idea of how handwriting might serve as a tool to evaluate a person's personality, mood and aid marketeers in establishing better inter-personal relationships. It does not provide any mathematical or experimental basis for it, however it does provide a theoretical basis by arguing that graphologists and psychiatrists stand in agreement with a correlation coefficient of 0.6 to 0.7. However, he does argue that it is too early to use graphology as a tool in the market and it might take a few years before its usability or short-comings are established.

[6] Using seven professional graphologists and seven test subjects, each subject was asked to select 20 traits about himself/herself and the graphologists were asked to determine the traits of the control-subjects using a page of text each subject had written. These datasets were compared. Further, all subjects were subjected to a standard personality test. And this data was then compared to the previous data. This was used to evaluate a coefficient of correlation between the graphologist's study and the actual personality of the control-subjects. The results were not very promising. However, this was on of the earliest experiments in graphology and so the author does not conclusively present any statement for or against it.

[7] This research paper has outlined a behavioral system or a tool to achieve high analysis rate on the nature of the person with the help of artificial neural networks and character recognition and detection systems. The proposed system will automatically detect the traits of the person by letter slants.

[8] In this paper, we propose a study on the complexity measure of an object. It is based on the analysis of different details that may be limited by object contours. They may

be holes or convexity evolution along the contour line. We focus in the same way on empty zones and filled zones. This study leads to a novel measure of the topology complexity – vacuity measure – that quantifies the relation between emptiness or space and objects. Based on the vacuity measure, we propose to define a novel shape descriptor and the associated dissimilarity measure. They can be applied in handwritten character analysis and in object recognition in general. The experiments are performed on a handwritten character dataset (ORIFLAMMS) and the object shape dataset. However, human recognition process considers not only the object itself but also considers the elements not belonging to the object, for example, the holes and contour details of the object. Such elements are defined the vacuity part of the object. Human perception will capture both the vacuity and the object into the process. Therefore, we are working on a vacuity measure and its application on pattern recognition. This measure quantifies the relation of object and its empty parts disposition.

[9] Personality is a fundamental basis of human behaviour. Personality affects the interaction and preferences of an individual. People are required to take a personality test to find out their personality. Social media is a place where users express themselves to the world. Posts made by users of social media can be analysed to obtain their personal information. This experiment uses text classification to predict personality based on text written by Twitter users. The languages used are English and Indonesian. Classification methods implemented are Naive Bayes, K-Nearest Neighbours and Support Vector Machine. Testing results showed Naive Bayes slightly outperformed the other methods. The system will retrieve a collection of traits from

users. Text from user then pre-processed into vector data. Classification process will classify user's text into a labelled dataset. The results are predictions for each Big Five traits, primary personality characteristics and secondary personality characteristics which obtained from the combination between two traits. System developed is a web application.

[10] This paper is to specify the handwritten data of 114 students categorised under three classes, the action, person and event. The authors have tried to perform the analysis on the imagination of the people as the subjects have to write the imagination content video displayed to them and highlight the coloured text of the handwritten data.

[11] This paper presents an experimental framework corresponding graphological features and main aim is to build a model to measure the qualities of the writer and provide resourceful manpower for human resource experts to save both time and efforts and make the selection process easy.

[12] This paper put forward the personal feature traits of those in particular between the age group of 20-35 years when they tackle many interviews. Polygonalization method is used to evaluate the baseline while margin will be calculated using the method of vertical scanning. Supervised machine learning algorithm, Feature Vector Matrix and similarity matrix method are key approaches used over data sets. The planned system can be used as a corresponding tool by the graphologist to recover the accuracy of graphological analysis and also makes the process rapid.

[13] The authors try to describe an online handwriting system that can support 102 different languages. They argue that this new system has 40% lesser errors than a conventional segment-and-decode system. Also, their new algorithm produces 10 times faster results. While most of their data analysis techniques are still OCR based, what is improved is the neural network model used. It is of recurrent nature and bidirectional. These neural networks are fed with Bezier's curves over datasets and the results are finally decoded.

[14] The authors argue that through structural feature extraction followed by established studies on zone, loop, angle and other features, they were able to obtain about 87% accuracy using CNN algorithm. Further, using SVM did not yield any significant improvements. However, using noisy datasets led to a drastic reduction in result accuracy.

[15] The author argues that there already exist several established methodologies for correlating handwriting and the mood of a person. However, they have tried to formulate a new methodology. On a control group of 36 participants and a clinical sample collected from 44 patients, they used Cohen Kappa method to correlate these two datasets with the results produced by 4 professional graphologists. The test was carried out after establishing a global evaluation standardisation for graphical analysis. The coefficient of correlation is high (~0.62) between graphologists and between 0.47 to 0.60 between the psychiatric assessment and graphological assessment. However, Cohen's Kappa yielded better results than Lorusso et al.

[16] Discussing RCNN and how it optimises CNN problems and gives rise to LSTM.

[17] Handwriting Analysts study the handwriting and predict the behavior of a person based on their skills. To make this more accurate, a relatively simpler method has been proposed to anticipate the personality of a person by exploring various handwriting features. The system considers five discriminating features such as breaks, size, space between words, baseline, loop of ‘e’ and few other features like pressure, margin, slant and dot distance in ‘i’. The proposed system can be used as a twin tool by graphologist to improve the accuracy and anticipate the behaviour of a person faster. The estimated weighted accuracy of 93.77 % is achieved.

[18] In this paper, a method has been proposed to predict the personality of a person from the features extracted from his handwriting using Artificial Neural Networks. The personality traits revealed by the baseline, the pen pressure and the letter„t‟ as found in an individual‘s handwriting are explored in this paper. Three parameters, the baseline, the pen pressure and the height of the t-bar on the stem of the letter‗t‘ are the inputs to the ANN which outputs the personality trait of the writer. The evaluation of the baseline is using the polygonalization method and the evaluation of the pen pressure utilizes the grey-level threshold value. The height of the t-bar on the stem of the letter ‗t‘ is calculated using template matching. The performance is measured by examining multiple samples.

[19] This research proposed a tool that can identify small letter ‘t’ character from a set of handwriting and the ambition levels of a person based on the small letter ‘t’ character using back propagation neural network. It will analyze three different angles of t-bars: up-turned, horizontal and down-turned. From there, we will determine the level of ambition of a person. This study has proven that the artificial neural network

is able to classify the level of ambition of person based on handwriting. It learns the input patterns and able to classify them. The implementation of the personality analysis using the back propagation neural network model is the alternate tool for the graphologist to do their personality analysis judgment.

[20] In this paper, an innovative system biometric of specific writers' identification based on technical expert calligraphic and graphology on handwritten script is presented. The system allows for the distinguishing with a success rate of 99.34% among the different writers from our database, using only 5 graphological parameters and integrating them in the automatic biometric system based on NN+MV A for short database or SYM when the number of writers for our database is increased. These graphological parameters are "longitude," "Union of letters," "pressure," "thinningarea," and "infovocalA." Therefore, the use of adequate parameters is a main reason for obtaining stability and efficiency on the implemented system. Finally, the independence of the system is also demonstrated regarding the used classifier, because it provides similar success for 29 writers, and a bit better SYM vs. NN, when the database is increased.The success percentage achieved with five of these characteristics from this database of 29 writers is 99.34%. In new experimentation, with these same parameters and enlarging the database to 70 users, a success rate of 92% was reached.

[21] This paper presents a hybrid KNN-SVM method for cursive character recognition. Specialized Support Vector Machines (SVMs) are introduced to significantly improve the performance of KNN in handwrite recognition. This hybrid approach is based on the observation that when using KNN in the task of handwritten

characters recognition, the correct class is almost always one of the two nearest neighbors of the KNN. Specialized local SVMs are introduced to detect the correct class among these two different classification hypotheses. The hybrid KNN-SVM recognizer showed significant improvement in terms of recognition rate compared with MLP, KNN and a hybrid MLP-SVM approach for a task of character recognition. The experiments demonstrated that the combination of KNN with SVMs experts' pairs of classes that constitute the greatest confusion of kNN, have improved performance in terms of recognition rate. The results showed a significant improvement from 1.00% to 3.61% in recognition rate for all cases tested (uppercase, lowercase and uppercase + lowercase).

[22] The authors propose the usage of graphology to assess Cattell's 16 Personality Features. This is achieved using CNN and based on four specific letters ('a', 'g', 's', 't'). The authors claim that they achieved an accuracy of 98.03% under the limited test they conducted.

[23] This paper includes comparison of various Machine learning algorithms used for personality analysis using letter slant , space between words, Height of t bar and writer identification with the accuracy of the algorithm. Personality detection using ANN will be a helpful and efficient system for personality traits classification.

[24] This paper proposes to deploy a combination of both approaches using Oriented Basic Image features and the concept of graphemes codebook. To reduce the resulting high dimensionality of the feature vector a Kernel Principal Component Analysis has been used. To gauge the effectiveness of the proposed method a performance analysis, using IAM dataset for English handwriting and ICFHR 2012

dataset for Arabic handwriting, has been carried out. The results obtained achieved an accuracy of 96% thus demonstrating its superiority when compared against similar techniques.

[25] This paper aims to evaluate the importance of different groups of features to model handwriting deficits that appear due to Parkinson’s disease; and how those features are able to discriminate between Parkinson’s disease patients and healthy subjects.

[26] The author has suggested a technique in this paper for predicting a person's behaviour from the features derived from his handwriting. In this paper, we examine the personality traits shown by baseline , letter slant, pen strain, letter I and letter f as found in the handwriting of individuals. Five parameters, baseline, slant, pen strain, letter I and letter f are entered in the ANN which outputs the writer's personality trait. Baseline and letter slant evaluation uses the polygonization method, pen pressure evaluation uses a gray-level threshold value and letter I and letter f usage template matching evaluations. To this end, MATLAB is the tool used. The performance is measured by multiple sample analysis.

[27] In this paper a review of datasets, acquisition devices and preprocessing for offline and online systems followed by characters of handwriting like margins, slanting lines, upward writing, connected letters, speed and pressure points that deduce the writer’s personality traits are proposed.

[28] In this paper, we present a deep adaptive learning method for writer identification based on single-word images using multi-task learning. An auxiliary task

is added to the training process to enforce the emergence of reusable features. their proposed method transfers the benefits of the learned features of a convolutional neural network from an auxiliary task such as explicit content recognition to the main task of writer identification in a single procedure.

[29] The author had utilized a solitary line of content right now infer a lot of qualities portraying a writer. Accordingly, it is set between the strategies proposed in the first writing, which utilize either full-content pages or single words. These allude to amounts like the tallness of the fundamental content zones, penmanship width, inclination, and others. In blend with a k-closest neighbor and a neural net classifier, twelve highlights were utilized. For the k- closest neighbor classifier, 87.8 percent of the content lines are allotted by utilizing a subset of the highlights to the correct client. Utilizing the entirety of the element esteems, the neural system classifier accomplished 90.7 percent.

[30] The authors' strategies for perceiving and checking writers referenced right now two significant wellsprings of conduct data with respect to uniqueness in penmanship. In the first place, the standard pen grasp and favored composing inclination and ebb and flow are communicated in the directional surface-level qualities which work in the rakish space at the ink-follow width scale. Second, the custom allograph assortment utilized by every individual recorded as a hard copy is caught by the likelihood of a graphene event. These have handy feasibility and are available

[31] The authors have attempted to establish a relationship between note-taking, its quality, speed of note-taking etc. and compared with objective mechanisms

for studying attention such as verbal and non-verbal tasks, symbol copying etc. The authors suggest that handwriting may in fact be a strong means to understand attention of students and further related it with study goals, working memory and other academic entities.

[32] The authors have tried to study and compare existing solutions and methodologies for handwriting feature extraction and style clustering. After carrying out a space analysis for all the methodologies. The authors worked on a new method to improve the existing k-means method. And observed that when clustering algorithms like Dunn Index were run, the results were more efficient at demonstrating compactness and separation of clusters.

[33] The authors suggest that since handwriting has a certain degree of ubiquity in human transactions, they have created a neural network based on TensorFlow to create a business and scientific tool for handwriting recognition. They claim that this project is of great importance to economy and business analysis.

[34] The authors have tried to mathematically correlate handwriting and psychopathic disorder in criminals and humans. They have generated the hypothesis using Mann-Whitney U Test with a group of 30 people diagnosed with some sort of psychopathic disorder. The authors have come to the conclusion that psychopathic behaviour has no relation with the handwriting of the person or any such relation is a very specific case and no general pattern may be established out of it.

[35] The authors suggest that handwriting is closely related to a person's behaviour and personality and hence ought to be unique to every individual. Using graphological elements like the skewness of the words, slant, pen pressure etc., and using the

standard IAM database, the authors have developed a feature extracting methodology to create an agent that ascertains the individuality of handwritten text.

[36] The authors have worked on a psycholinguistic database for Chinese characters and also using about 203 participants. Using several graphological elements and assessment methods, the authors have concluded that semantic factors such as concreteness and imageability of the characters are effective on accuracy alone while phonographic factors do not affect handwriting execution.

[37] The authors have tried to assess various possible methodologies involved in handwriting recognition, both online and offline. It includes the creation of subsets of attributes and a quantitative analysis is established over them.

[38] Automatic online handwriting data analysis—hand- writing analysis based on both the static and dynamic data from the tablet—is commonly used for handwriting text recognition. These data are analysed and classified using a number of common techniques including statistical pattern recognition algorithms such as hidden Markov models and its derivatives, and artificial intelligence systems such as neural networks and fuzzy assessment

[39] This paper presents a novel joining method that is based on the modification of the original dynamic movement primitive formulation. The new method can reproduce the target trajectory with high accuracy regarding both the position and the velocity profile and produces smooth and natural transitions in position space, as well as in velocity space. The properties of the method are demonstrated by its application to simulated handwriting generation, which are also shown on a robot, where an adaptive algorithm is used to learn trajectories from human demonstration.

[40] The writer has proposed a system for foreseeing a person's precise character characteristics from the highlights got from penmanship utilizing a way to deal with AI. This paper investigates the character attributes of an individual's penmanship uncovered by standard, edge, the inclination of the words and stature of the t- bar. Such qualities were separated from the penmanship tests into include vectors that were contrasted with an informational index at first prepared and afterward mapped to the class with a relating character attribute. The benchmark was estimated utilizing the Polygonization strategy while the edge was resolved utilizing the vertical checking technique. The tallness of the t-bar on the letters in order ' t' and word-incline stem was estimated utilizing comparable layouts.

Proposed Structure of Project

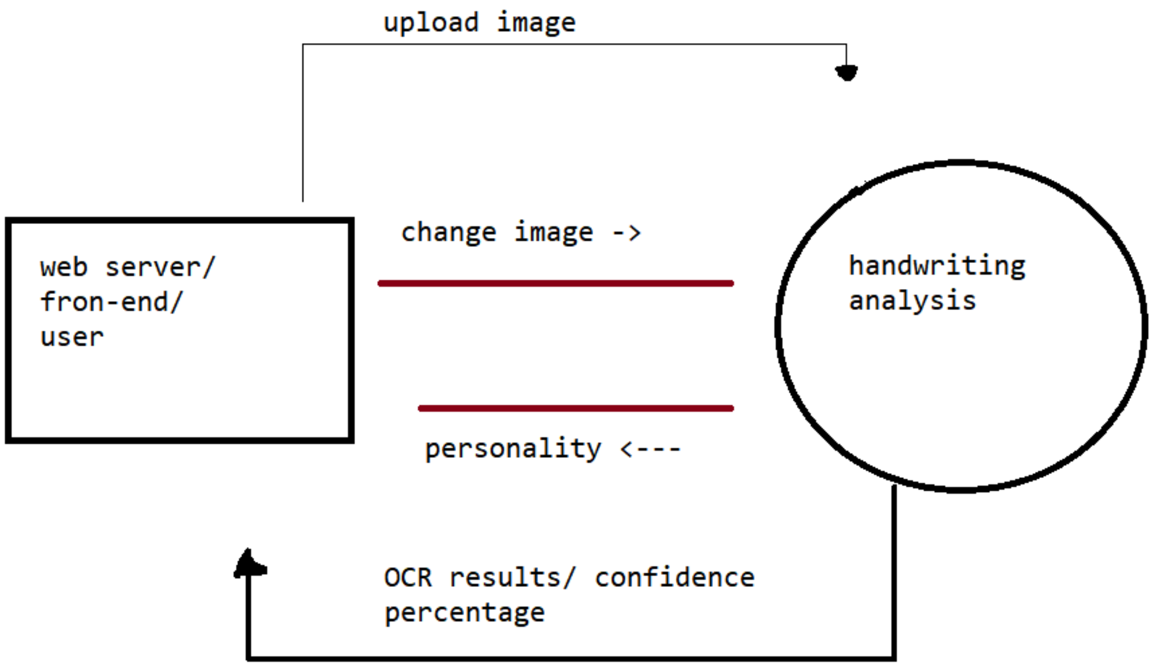


Figure 1: Proposed structure

Pseudo code

→Input

→ Image upload/change

→ pre-processing of image

→ OCR using tesseract JS → feature extraction

→ Neural Network (Fig 1)

→ (if error) Back-propagation

→Result → Output on UI

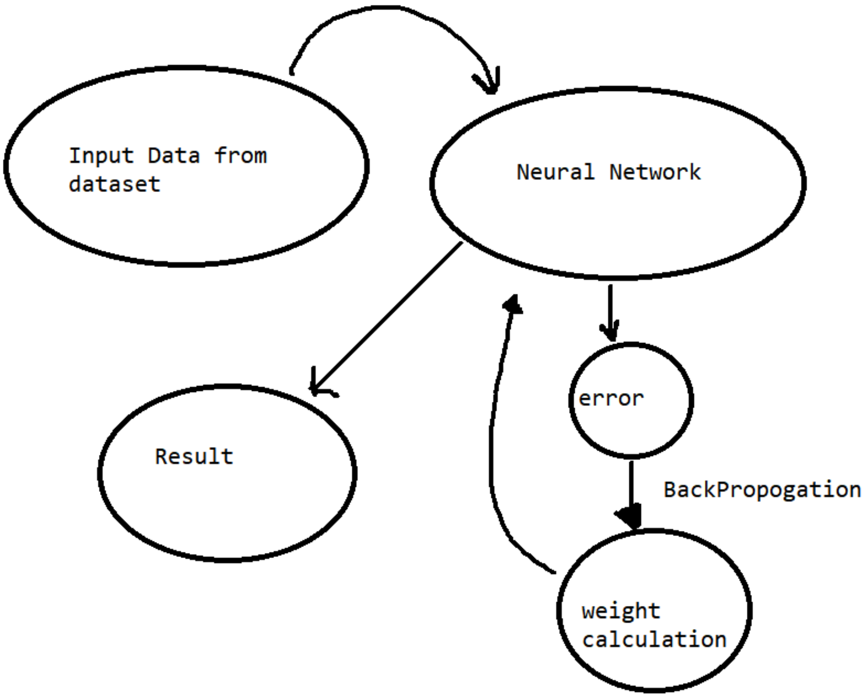


Figure 2: Proposed model of the neural network

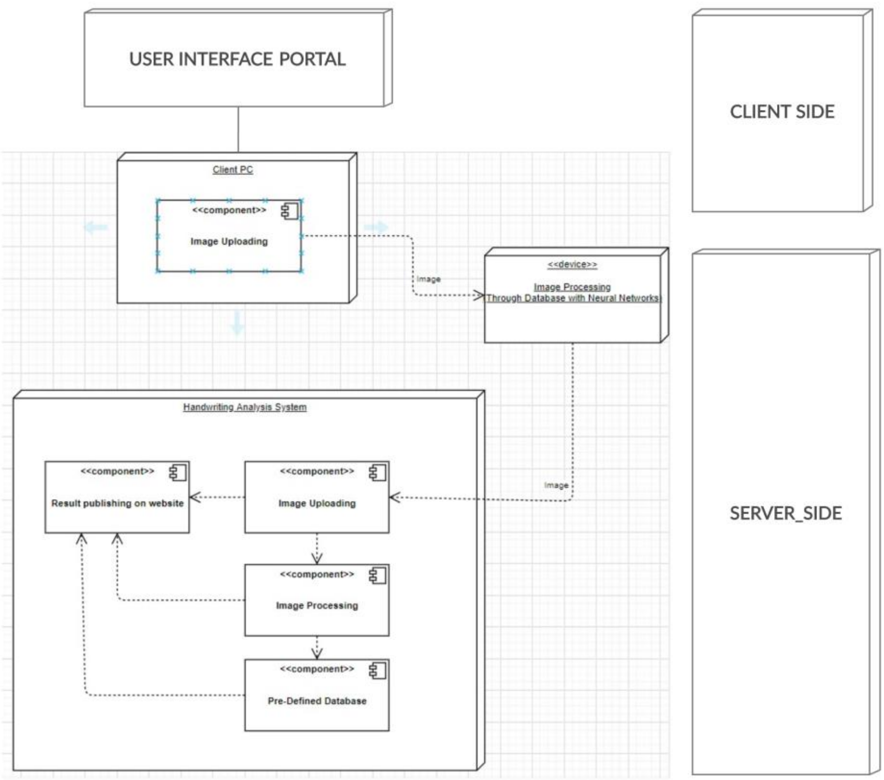


Figure 3: Architecture diagram for the project

Dataset used

The Dataset Used For this project, we have used 4 different data-sets:

1. MNIST database:
    1.1. Devanagari Script Database
    1.2. Mathematics Script Database
    1.3. English Characters Database.
2. IAM database – 1539 images

PROPOSED WORK

The creation of a simple three-layer neural network with two of them being for input and output respectively. This is because of several reasons that shall be discussed in the proposed algorithm.

The project involves the following components:

A. Multi-class classification

In order to classify the data, the authors rejected linear classification owing to the multi-feature nature of the data itself. Regression methodologies like One-vs-All logistic regression yields high accuracy, however, takes a lot of time especially with the data set as large and varied as handwriting. It involves binary logistic regression to every feature and return the highest value as prediction. This is similar to the concept of a confidence variable for each feature and the variable with highest value is returned.

So, the authors decided to rely on neural networks in this project as it would be computationally cheaper, easier to construct and modify and compatible with various open source libraries.

B. Neural Network

An artificial neuron is a device with several inputs and one output and has two modes of operation – training and using modes. So, a program was developed to understand the working of the libraries involved. This is a single layered neural network with the hidden layer having several units. This step recreates the datasets available as coordinates of the alphabet curve into images of PNG format.

GNU Octave 5.2.0 is used to feed forward the neural network and return the cost while the theta values are then determined using back-propagation. An accuracy of 92% was achieved in the neural network.

Figure 4: MNIST database after conversion to image by first neural network.

C. Image OCR

We have used TesseractJS for this purpose as it supports over 60 languages, boxing of symbols and supports confidence variables for each symbol. The TesseractJS engine extracts text from the image uploaded. The confidence variable with the highest confidence is sent to CNN layer for recognition.

However, TesseractJS has only about 71% accuracy as per its official documentation. A better alternative could be Google OCR with 95% accuracy; however, it does not support the confidence variable option.

D. CNN for symbol Recognition

After the symbols have been extracted, they are sent to a Keras-built LeNet-5 architecture. We have undertaken 50 epochs for the CNN training and realised that the data does not converge. The reason for this being the lack of data. Psychology, being a fluid subject and lack of data for handwriting recognition as well, the CNN fails to achieve a very high level of accuracy, especially under accelerated training conditions.

Data augmentation might help to generate additional datasets from the current set itself, however, due to forensic limitations that might arise because of it, the authors suggest against it. Other than this, the authors would also like to note that CRNN is a way better methodology for this purpose since CNN, despite its high accuracy has a very poor recall rate.

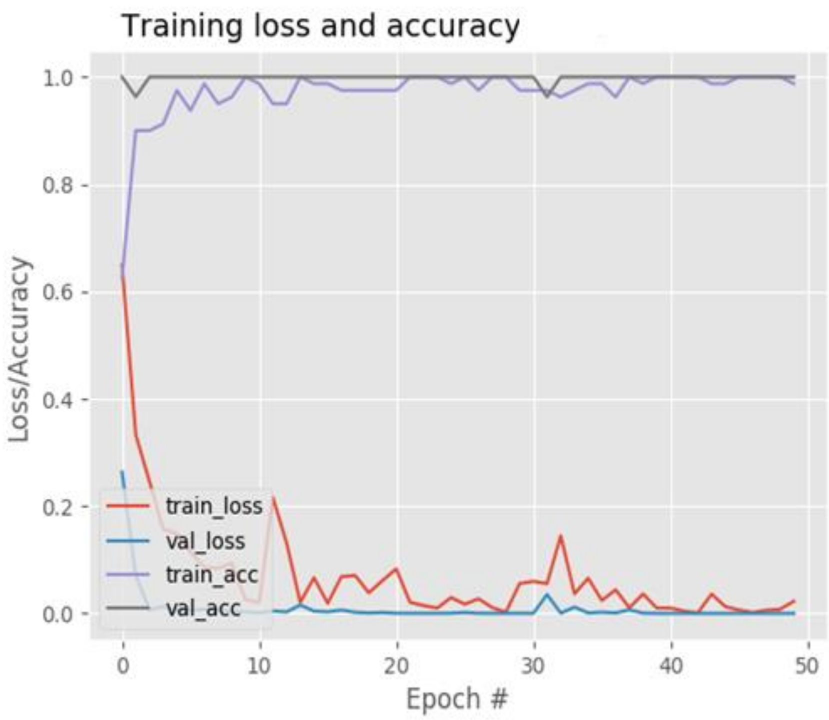


fig 4(a)

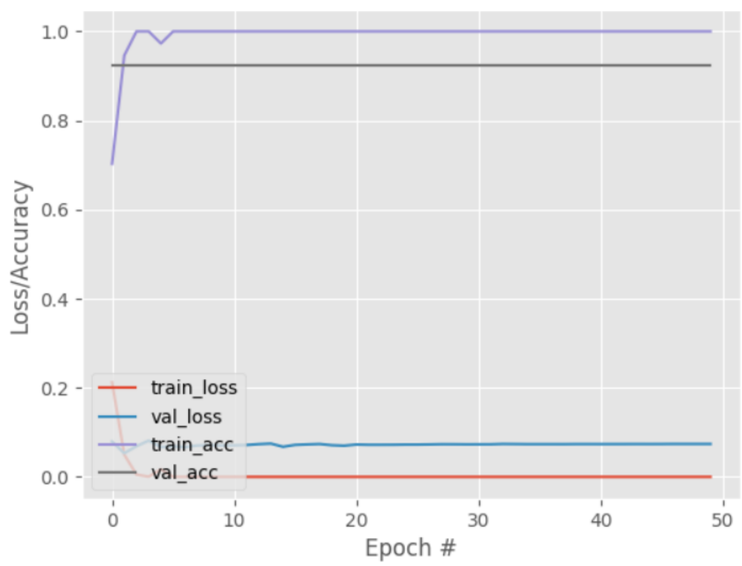


fig 4(b)

Figure 5: Loss vs accuracy in training.

E. A simpler neural network

To solve the problem of lack of data, a simpler neural network might be used. This is based on the hypothesis that simple neural networks yield better performance than CNNs when the dataset is limited. The neural network built has a simple back-propagation algorithm with N sets in the hidden layer of the network.

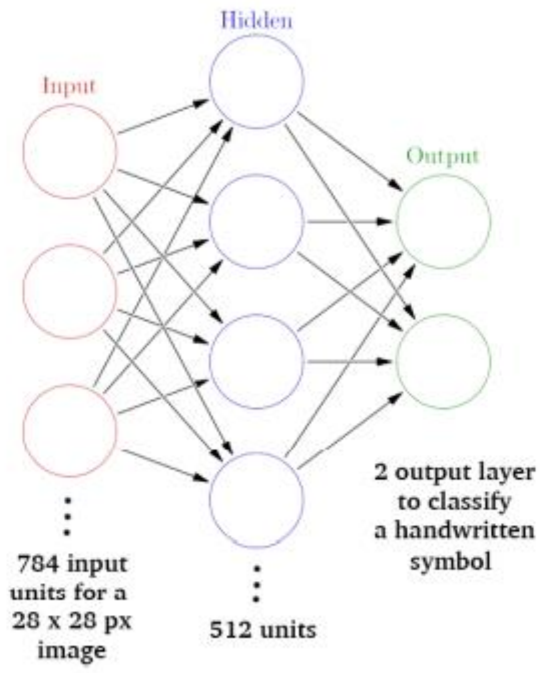


Figure 6: A simple neural network

F. Software integration

To a user, the project is a simple web page where the user can upload images. It has a minimalist UI and obeys all 4 golden rules of UI design. The graphological analysis is presented in a dialog box dedicated to this purpose.

Experimentation and Result

A. Experimental Constraints

Improvements in Graphology-based throughput is highly dependent on experimental evidence. However, due to restriction on movement, we will have to largely rely on available online database. Other than that, we have asked a sample population of 25 students to send a letter written by them and specify their mood too. This shall be then tabulated against the results produced by the software and an experimental accuracy would be determined. However, this experimental accuracy cannot be relied upon strictly as the sample space of the experiment is fairly small.

B. Results

1. USER INTERFACE

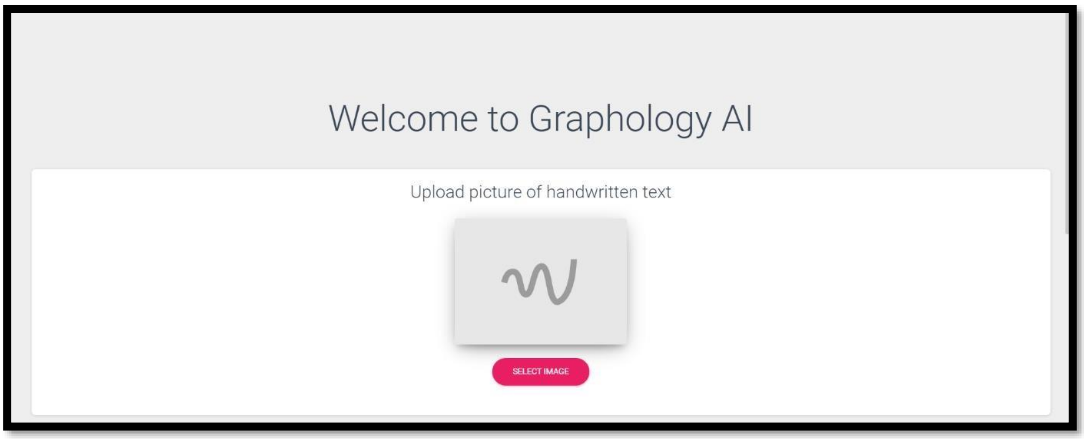
The user interface is minimalistic, simple and obeys the 4 golden rules of UI design which are:

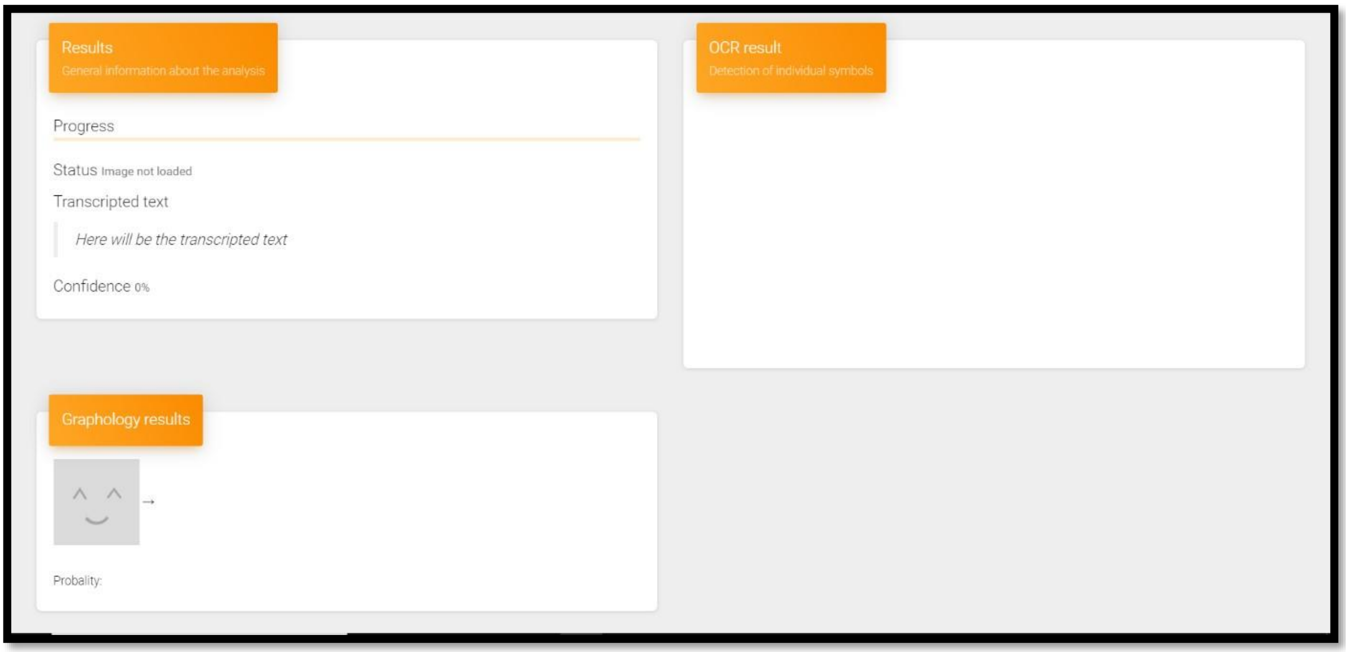
1. The user will find himself/herself/themselves in control of the UI.
2. The user shall not find any difficulty navigating through the project.
3. There is minimal cognitive load on the page.
4. The UI is consistent and pleasing.

2. IMPLEMENTATION

STEP1

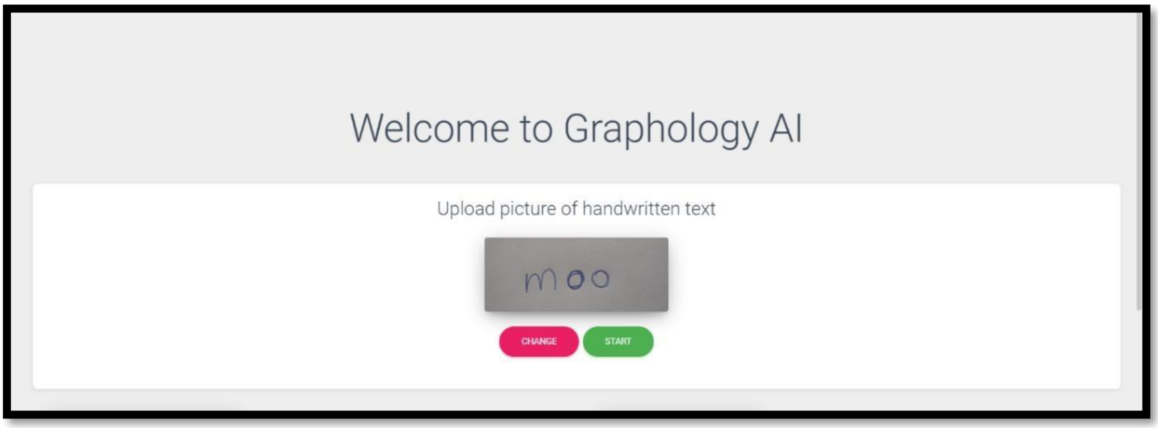
The analyzer opens waiting for the user to input the handwritten image for which graphological analysis has to be done.

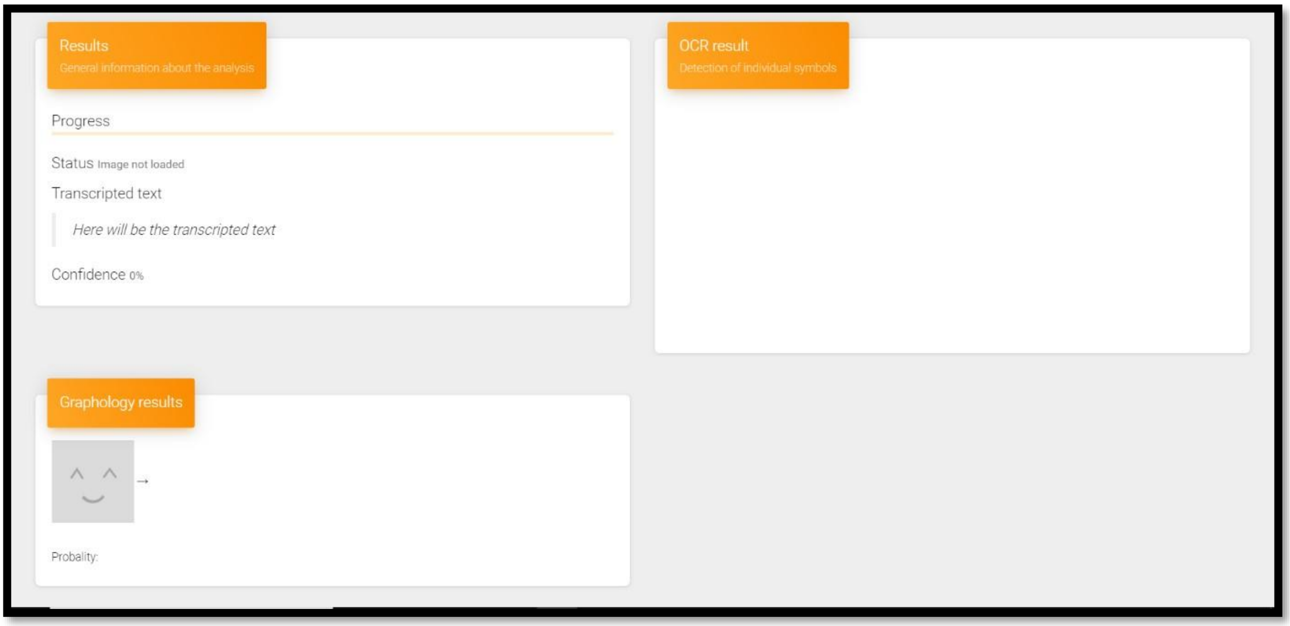




STEP 2

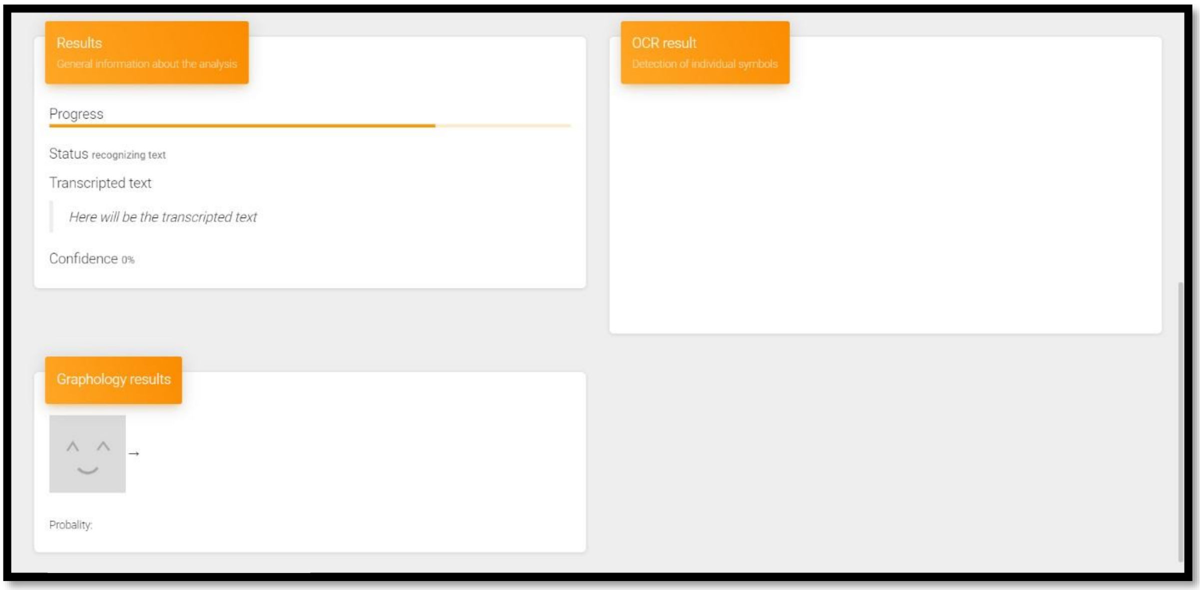
User uploads the handwritten input in the analyzer for which graphological analysis has to be done.





STEP 3

Analyzer starts processing of image by recognizing the character entered by the user. The progress bar shows how much the image has been processed.

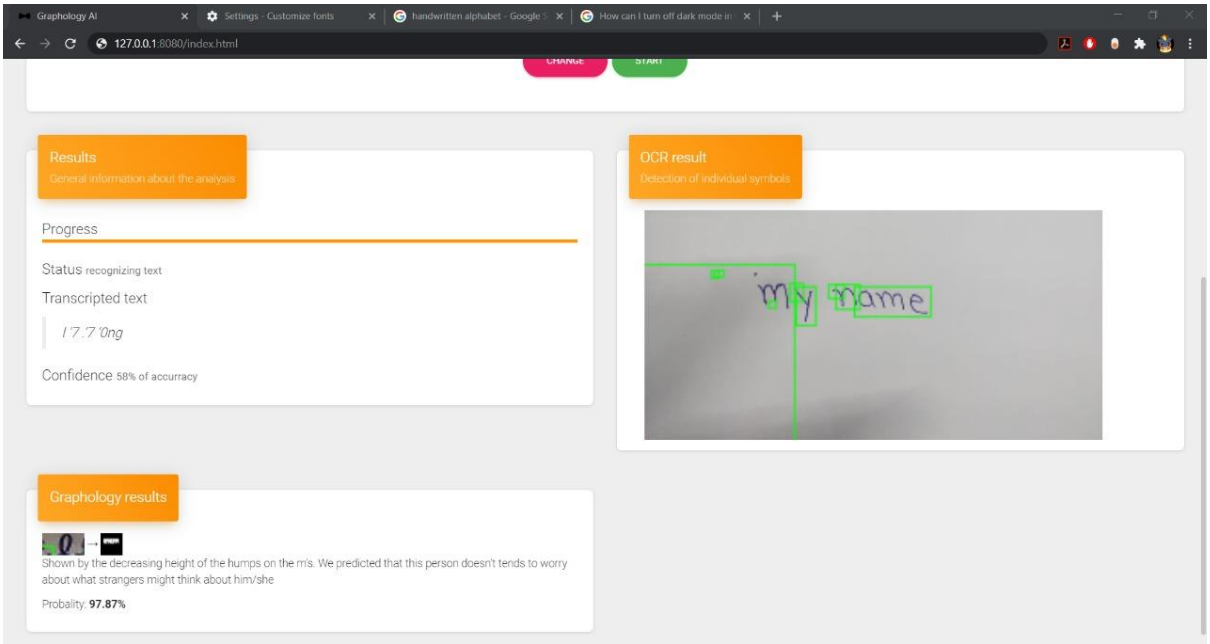


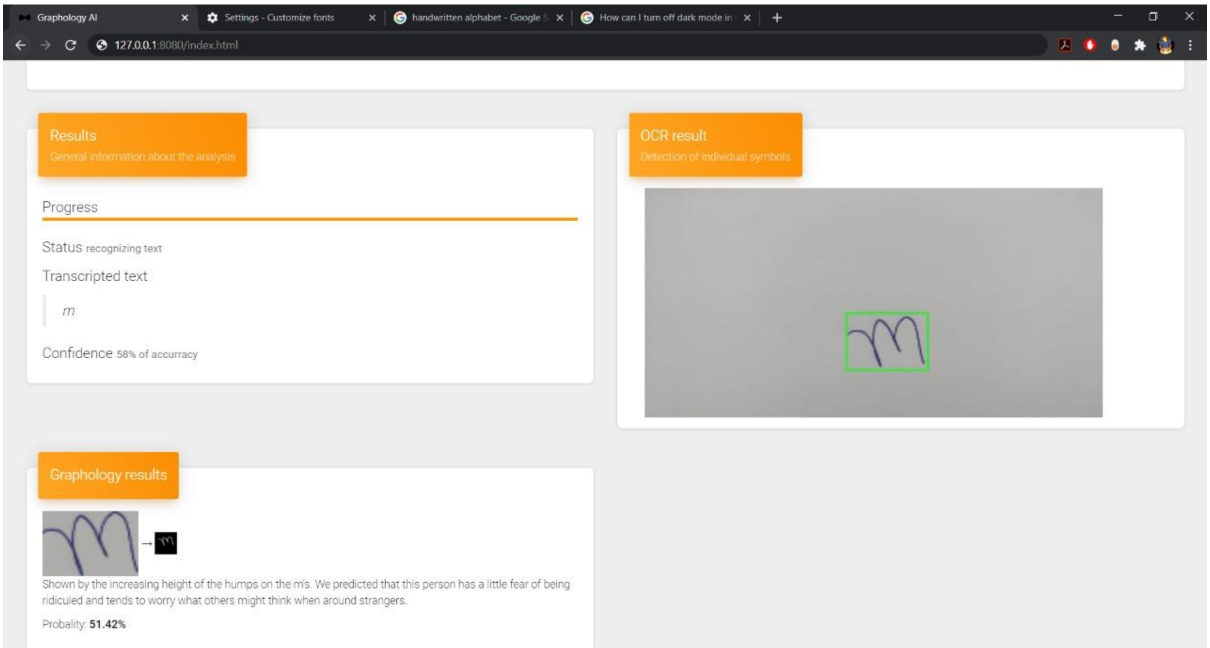
STEP 4

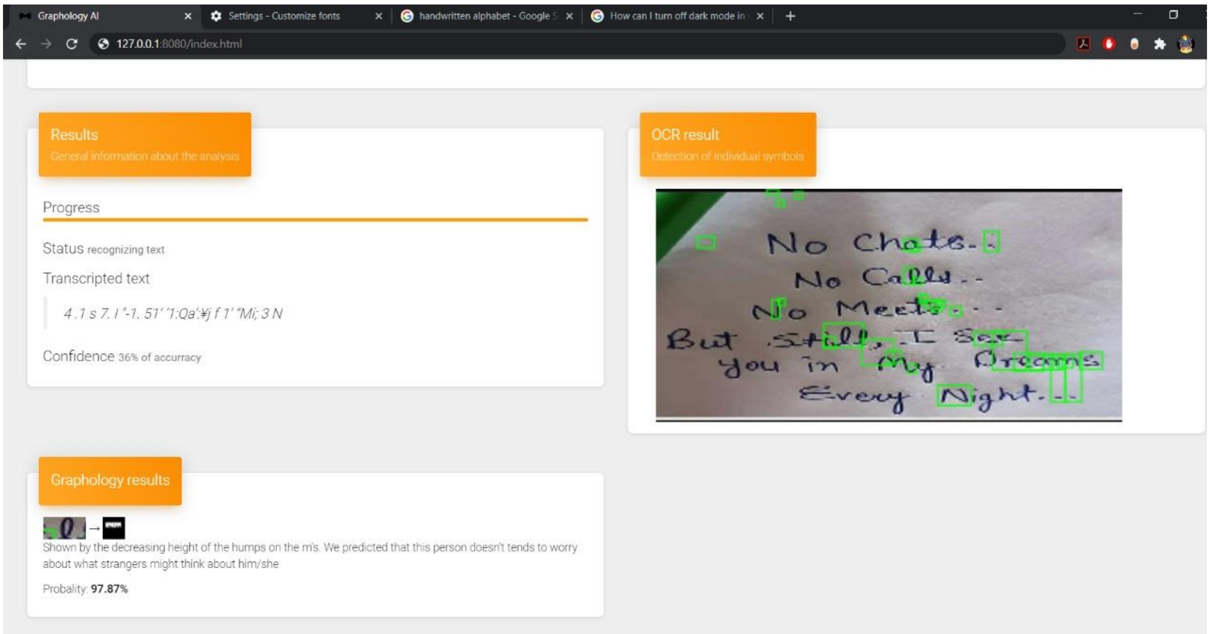
After the image is processed and the character is recognized by OCR, then the analyzer starts personality character check by the way the character has been drawn and comparing its width, height and other attributes from the dataset using neural networks.

After it finds the accurate match to the character, it displays the graphology results corresponding to the attributes found for the uploaded character.

Depending upon how close the attributes match with the uploaded character, it also shows the probability that the personality trait displayed is true along with the accuracy percentage.

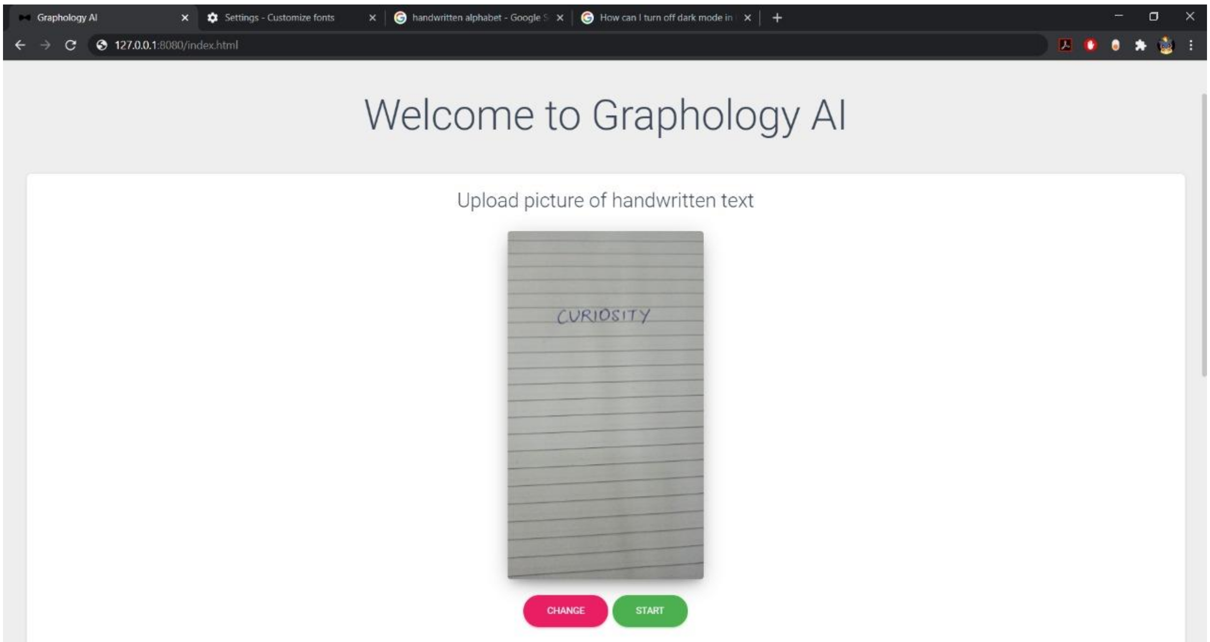


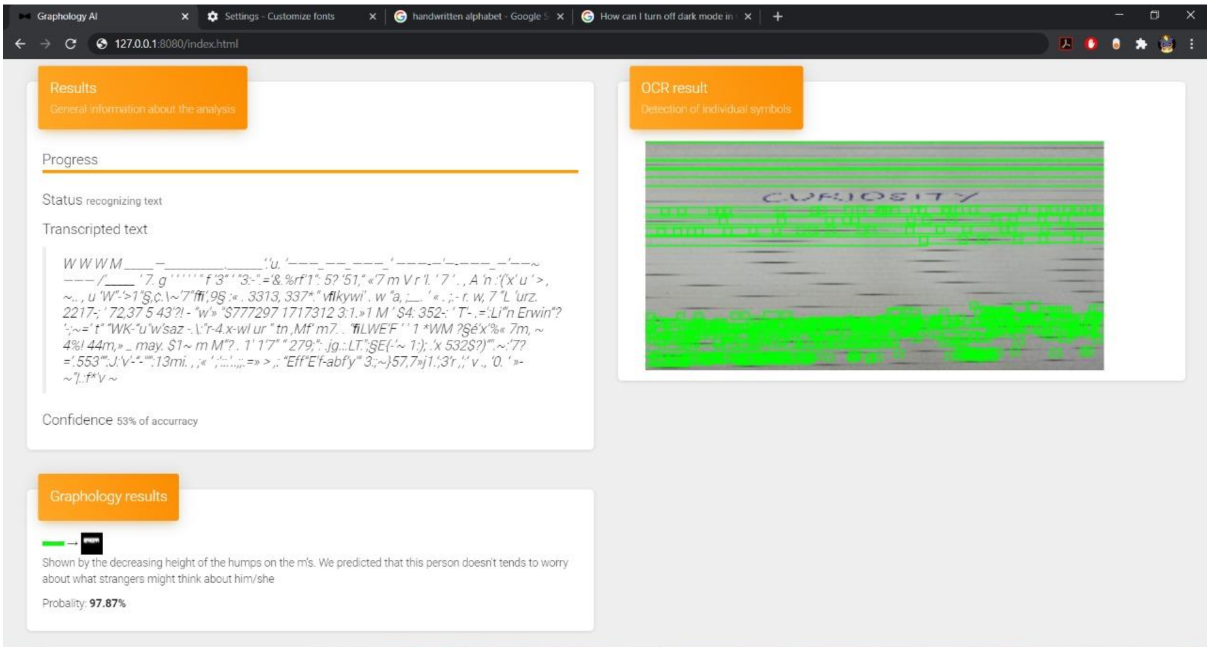




Noisy Outputs-

Noisy outputs are a problem with the project. The project fails to identify the characters partially or completely and hence even though an output maybe derived by the software, the output is clearly not trust-worthy.





(*The project clearly does not recognize that its output is incorrect. The AI agent is not configured to do so yet.*)

CODE:

His is to train the image processing neural network

test_network.py

Users > saurabh > Downloads > Graphology-Machine-Learning-master-2 > Classifier_test > test_network.py

```
# USAGE
# python test_network.py --model social_worried.model --image images/examples/test_positive.png

# import the necessary packages
from keras.preprocessing.image import img_to_array
from keras.preprocessing import image
from keras.models import load_model
import numpy as np
import argparse
import imutils
import cv2

# construct the argument parse and parse the arguments
ap = argparse.ArgumentParser()
ap.add_argument("-m", "--model", required=True,
    help="path to trained model model")
ap.add_argument("-i", "--image", required=True,
    help="path to input image")
args = vars(ap.parse_args())

# load the image
img = image.load_img(args["image"], target_size=(28, 28)).convert('L')
# print("Image size: " + str(np.shape(img)))
orig = image.load_img(args["image"], target_size=(28, 28))
orig = img_to_array(orig)
img = img_to_array(img).ravel()

# pre-process the image for classification
print("Image 1")
print(img)
img = img.astype("float") / 255.0
print("Image 2")
print(img)
img = np.expand_dims(img, axis=0)

# load the trained convolutional neural network
print("[INFO] loading network...")
model = load_model(args["model"])

# classify the input image
```

test_network.py

Users > saurabh > Downloads > Graphology-Machine-Learning-master-2 > Classifier_test > test_network.py

```python
print("[INFO] loading network...")
model = load_model(args["model"])

# classify the input image
print("Shape: " + str(np.shape(img)))
(notSanta, santa) = model.predict(img)[0]
print("Not Social Worried: " + str(notSanta) + " Worried: " + str(santa))

# build the label
label = "Social worried" if santa > notSanta else "Not social worried"
proba = santa if santa > notSanta else notSanta
label = "{}: {:.2f}%".format(label, proba * 100)

# draw the label on the image
output = imutils.resize(orig, width=400)
color = (0, 255, 0) if santa > notSanta else (0, 0, 255)
cv2.putText(output, label, (10, 25),  cv2.FONT_HERSHEY_SIMPLEX,
    0.7, color, 2)

# show the output image
cv2.imwrite("result.png", output)
#cv2.imshow("Output", output)
#cv2.waitKey(0)
```

This python file is to create the final neural network instead of CNN engine.

test_network.py | train_network_bk.py

Users > saurabh > Downloads > Graphology-Machine-Learning-master-2 > Classifier_test > train_network_bk.py

```python
# USAGE
#  social_worried.model

# set the matplotlib backend so figures can be saved in the background
import matplotlib
matplotlib.use("Agg")

# import the necessary packages
from keras.preprocessing.image import ImageDataGenerator
from keras.optimizers import Adam
from sklearn.model_selection import train_test_split
from keras.preprocessing.image import img_to_array
from keras.utils import to_categorical
from pyimagesearch.lenet import LeNet
from imutils import paths
import matplotlib.pyplot as plt
import numpy as np
import argparse
import random
import cv2
import os

from keras.models import Sequential
from keras.layers import Dense
from keras.layers import Dropout
from keras.layers import Flatten
from keras.layers.convolutional import Conv2D
from keras.layers.convolutional import MaxPooling2D
from keras.layers import Input
from keras.layers import concatenate
from keras.optimizers import SGD

# construct the argument parse and parse the arguments
ap = argparse.ArgumentParser()
ap.add_argument("-d", "--dataset", required=True,
    help="path to input dataset")
ap.add_argument("-m", "--model", required=True,
    help="path to output model")
ap.add_argument("-p", "--plot", type=str, default="plot.png",
    help="path to output loss/accuracy plot")
```

test_network.py train_network_bk.py

Users > saurabh > Downloads > Graphology-Machine-Learning-master-2 > Classifier_test > train_network_bk.py

```
    help="path to output loss/accuracy plot")
args = vars(ap.parse_args())

# initialize the number of epochs to train for, initia learning rate,
# and batch size
EPOCHS = 100
INIT_LR = 1e-3
BS = 4

# initialize the data and labels
print("[INFO] loading images...")
data = []
labels = []

# grab the image paths and randomly shuffle them
imagePaths = sorted(list(paths.list_images(args["dataset"])))
random.seed(42)
random.shuffle(imagePaths)

# loop over the input images
for imagePath in imagePaths:
    # load the image, pre-process it, and store it in the data list
    image = cv2.imread(imagePath)
    #image = cv2.resize(image, (28, 28))
    image = img_to_array(image)
    data.append(image)

    # extract the class label from the image path and update the
    # labels list
    label = imagePath.split(os.path.sep)[-2]
    label = 1 if label == "m_worried" else 0
    labels.append(label)

# scale the raw pixel intensities to the range [0, 1]
data = np.array(data, dtype="float") / 255.0
labels = np.array(labels)

# partition the data into training and testing splits using 75% of
# the data for training and the remaining 25% for testing
(trainX, testX, trainY, testY) = train_test_split(data,
```

test_network.py | train_network_bk.py

Users > saurabh > Downloads > Graphology-Machine-Learning-master-2 > Classifier_test > train_network_bk.py

```python
# partition the data into training and testing splits using 75% of
# the data for training and the remaining 25% for testing
(trainX, testX, trainY, testY) = train_test_split(data,
    labels, test_size=0.25, random_state=42)

# convert the labels from integers to vectors
trainY = to_categorical(trainY, num_classes=2)
testY = to_categorical(testY, num_classes=2)

# construct the image generator for data augmentation
aug = ImageDataGenerator(rotation_range=30, width_shift_range=0.1,
    height_shift_range=0.1, shear_range=0.2, zoom_range=0.2,
    horizontal_flip=False, fill_mode="nearest")

# initialize the model
print("[INFO] compiling model...")
model = LeNet.build(width=28, height=28, depth=3, classes=2)

def Pimped_LeNet5():
    """CNN model based on LeNet-5."""

    # Create model:
    model = Sequential()
    model.add(Conv2D(30, (5, 5), input_shape=(1, 28, 28), activation='relu'))
    model.add(MaxPooling2D(pool_size=(2, 2)))

    model.add(Conv2D(15, (3, 3), activation='relu'))
    model.add(MaxPooling2D(pool_size=(2, 2)))

    model.add(Dropout(0.1))  # Let's try to avoid overfitting...
    model.add(Flatten())

    model.add(Dense(128, activation='relu'))
    model.add(Dense(50, activation='relu'))
    model.add(Dense(num_classes, activation='softmax'))

    # Compile model:
    model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])

    return model
```

test_network.py   train_network_bk.py ×

Users > saurabh > Downloads > Graphology-Machine-Learning-master-2 > Classifier_test > train_network_bk.py

```python

    return model

#model = Pimped_LeNet5()
#history = model.fit(trainX, trainY, validation_data=(testX, testY), epochs=15, batch_size=2)

opt = Adam(lr=INIT_LR, decay=INIT_LR / EPOCHS)
model.compile(loss="binary_crossentropy", optimizer=opt,
    metrics=["accuracy"])

# train the network
print("[INFO] training network...")

H = model.fit_generator(aug.flow(trainX, trainY, batch_size=BS),
    validation_data=(testX, testY), steps_per_epoch=len(trainX) // BS,
    epochs=EPOCHS, verbose=1)

# save the model to disk
print("[INFO] serializing network...")
model.save(args["model"])

# plot the training loss and accuracy
plt.style.use("ggplot")
plt.figure()
N = EPOCHS
plt.plot(np.arange(0, N), H.history["loss"], label="train_loss")
plt.plot(np.arange(0, N), H.history["val_loss"], label="val_loss")
plt.plot(np.arange(0, N), H.history["acc"], label="train_acc")
plt.plot(np.arange(0, N), H.history["val_acc"], label="val_acc")
plt.title("Training loss and accuracy on handwritten M")
plt.xlabel("Epoch #")
plt.ylabel("Loss/Accuracy")
plt.legend(loc="lower left")
plt.savefig(args["plot"])
```

This python file is to train the final simple neural network train_network_NN.py

Users > saurabh > Downloads > Graphology-Machine-Learning-master-2 > Classifier_test > train_network_NN.py

```python
import matplotlib
import matplotlib.pyplot as plt

import numpy as np # linear algebra
from keras import models
from keras import layers
from keras.utils import to_categorical

from sklearn.model_selection import train_test_split
from keras.preprocessing.image import img_to_array
from keras.preprocessing import image
import tensorflowjs as tfjs

from imutils import paths
import argparse
import random
import cv2
import os

# Initialize the number of epochs to train for, initia learning rate,
# and batch size
EPOCHS = 50
INIT_LR = 1e-3
BS = 4

# Parse arguments
ap = argparse.ArgumentParser()
ap.add_argument("-d", "--dataset", required=True,
    help="path to input dataset")
ap.add_argument("-m", "--model", required=True,
    help="path to output model")
ap.add_argument("-p", "--plot", type=str, default="plot.png",
    help="path to output loss/accuracy plot")
args = vars(ap.parse_args())

# Initialize the data and labels
print("[INFO] loading images...")
data = []
labels = []

```

train_network_NN.py ×

Users > saurabh > Downloads > Graphology-Machine-Learning-master-2 > Classifier_test > train_network_NN.py

```
# Gather images from path
imagePaths = sorted(list(paths.list_images(args["dataset"])))
random.seed(42)
random.shuffle(imagePaths)

# loop over the input images
for imagePath in imagePaths:
    # load the image, pre-process it, and store it in the data list
    img = image.load_img(imagePath, target_size=(28, 28)).convert('L')
    # print("Image size: " + str(np.shape(img)))
    img = img_to_array(img).ravel()
    # print(np.shape(img))
    data.append(img)

    # extract the class label from the image path and update the
    # labels list
    label = imagePath.split(os.path.sep)[-2]
    label = 1 if label == "m_worried" else 0
    labels.append(label)

# scale the raw pixel intensities to the range [0, 1]
data = np.array(data, dtype="float") / 255.0
labels = np.array(labels)
print("Shape of all data: " + str(np.shape(data)))

# Split data into test and train
(trainX, testX, trainY, testY) = train_test_split(data,
    labels, test_size=0.25, random_state=42)
print(np.shape(trainX), np.shape(trainY), np.shape(testX), np.shape(testY), sep = " -- ")

# convert the labels from integers to vectors
trainY = to_categorical(trainY, num_classes=2)
testY = to_categorical(testY, num_classes=2)

print(np.shape(trainY))

# Create model
model = models.Sequential()
model.add(layers.Dense(512, activation='relu', input_shape=(28 * 28,)))
model.add(layers.Dense(2, activation='softmax'))
```

train_network_NN.py

Users > saurabh > Downloads > Graphology-Machine-Learning-master-2 > Classifier_test > train_network_NN.py

```python
model.compile(optimizer='rmsprop',
              loss='mean_squared_error',
              metrics=['accuracy'])

history = model.fit(trainX, trainY, validation_data=(testX, testY), epochs=EPOCHS, batch_size=BS)

print("[INFO] serializing network...")
# Load Keras model
model.save(args["model"])
tfjs.converters.save_keras_model(model, 'model')

# Evaluate loaded model on test data:
score = model.evaluate(testX, testY, verbose=1)
print('Test loss:', score[0])
print('Test accuracy:', score[1])

# plot the training loss and accuracy
plt.style.use("ggplot")
plt.figure()
N = EPOCHS
plt.plot(np.arange(0, N), history.history['loss'], label="train_loss")
plt.plot(np.arange(0, N), history.history['val_loss'], label="val_loss")
plt.plot(np.arange(0, N), history.history['acc'], label="train_acc")
plt.plot(np.arange(0, N), history.history['val_acc'], label="val_acc")
plt.title("Training loss and accuracy on handwritten M")
plt.xlabel("Epoch #")
plt.ylabel("Loss/Accuracy")
plt.legend(loc="lower left")
plt.savefig(args["plot"])
```

Conclusion

Graphology - the study of handwriting and handwriting analysis - is now an accepted and increasingly used technique for assessment of people in organizations. Handwriting analysis is an effective and reliable indicator of personality and behaviour, and so is a useful tool for many organizational processes, for example: recruitment, interviewing and selection, team-building, counselling, and career-planning.

These features and interpretations provide a small but useful guide as to the way people behave, and particularly how they handle their social requirements. Understanding the personality through handwriting is a valuable way of making the best of both personal awareness and interpersonal situations for the benefit of all concerned.

This interpretation should enable people analysed to use the understanding gained, to help them live their lives to the highest level of satisfaction that they choose. In a professional or organizational context, graphology can play an important part in enabling working relationships to be forged that will enhance the quality of the group or team performance.

As a child we are taught to write, but it's not likely that we still write in the way we were taught. This fact itself helps to explain the reason graphology exists and why graphology can be used to interpret personality.

References

[1] Handwriting Analysis through Graphology: A review by Laxmi Durga and Deepu R, IEEE, 2018

[2] Emotion Predictor Using Social Media Text and Graphology by Chhanda Roy & others, IEEE

[3] Forensic Handwriting Analysis by Dial, Jackie, Salem Encyclopaedia of Science, 2020

[4] Mobile Application for Identifying Personality of Person Using Graphology by Kukuh Adi Prasetyo and others, IES-KCIC, 2017, IEEE

[5] Graphology: A New Marketing Research Technique by JAMES U. McNeal, Journal of Market Research, 1967

[6] MARKETING NOTES AND COMMUNICATIONS, Graphology and Marketing Research: A Pilot Experiment in Validity and Inter-judge Reliability by PAUL E. GREEN VITHALA R. RAO and DIANA E. ARMANI, Journal of Marketing, 1971.

[7] Akshita Chanchlani, Pratima Kharade, Rutuja Kapase, Sonal Janvalka, Aakanksha Jaitly, "Predicting Human Behavior through Handwriting" International Journal for Research in Applied Science & Engineering Technology (IJRASET), ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor:6.887, Volume 5 Issue X, October 2017.

[8] Van-C uong Kieu,Dominique Stutzmann,Nicole Vincent "Vacuity Measure for Handwritten Character Analysis" 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR),IEEE(2017)

[9] Bayu Yudha Pratama ; Riyanarto Sarno "Personality classification based on Twitter text using Naive Bayes, KNN and SVM" 2015 International Conference on Data and Software Engineering (ICoDSE) , IEEE,2015

[10] Bayu Yudha Pratama ; Riyanarto Sarno " Personality classification based on Twitter text using Naive Bayes, KNN and SVM" 2015 International Conference on Data and Software Engineering (ICoDSE), IEEE,2015.

[11] Ricard Coll, Alicia Forn ´ es, Josep Llad ´ os, “Graphological Analysis of Handwritten Text Documents for Human Resources Recruitment,” Computer Vision Center - Computer Science Department Universitat Autonoma de Barcelona Spain {rcoll, afornes, josep}

[12] Prachi Joshi, Aayush Agarwal, Ajinkya Dhavale, Rajani Suryavanshi, Shreya Kodolikar, “Handwriting Analysis for Detection of Personality Traits using Machine Learning Approach”, International Journal of Computer Applications, (0975 – 8887) Volume 130 – No.15, November 2015.

[13] Fast multi-language LSTM-based online handwriting Recognition, by Victor Carbune and others, International Journal on Document Analysis and Recognition (IJDAR) (2020).

[14] Graphology based handwritten character analysis for human behaviour identification, ITET, Research Article, January 2020, Springer by Subhankar Ghosh and others.

[15] Is graphology Useful in assessing Major Depression? By Marco Giannini and others, Sage Magazine Psychological Reports, 2019.

[16] Recurrent Convolutional Neural Network for Object Recognition by Ming Liang and Xiaolin Hu, IEEE.

[17] Automated Handwriting Analysis for Human Behaviour Prediction by Manimala S and others, IJCSE,2016

[18] Artificial Neural Network for Human Behavior Prediction through Handwriting Analysis by Champa H N & Dr. K R AnandaKumar, International Journal of Computer Applications, 2010

[19] Personality Analysis Based On Letter ‘t’ Using Back Propagation Neural Network by Sofianita Mutalib & others, Proceedings of the International Conference on Electrical Engineering and Informatics Institut Teknologi Bandung, Indonesia June 17-19, 2007

[20] Writer Identification Based on Graphology Techniques by Omar Santana & others, IEEE A&E SYSTEMS MAGAZINE, JUNE 2010.

[21] A KNN-SVM Hybrid Model for Cursive Handwriting Recognition by Cleber Zanchettin, Byron Leite Dantas Bezerra and Washington W. Azevedo, WCCI 2012 IEEE World Congress on Computational Intelligence, 2012

[22] Personality Features Identification from Handwriting Using CNN by Sri Hasuthi Fatimah and others 2019 4th International Conference on Information Technology, ICITISEE, 2019.

[23] A Survey: Machine Learning Approach for Personality Analysis and Writer Identification through Handwriting by Mr Vishal Patil, Dr. Harsh Mathur,IEEE Xplore

[24] Writer identification approach based on bag of words with OBI features by Amal Durou, Ibrahim Aref ,Somaya Al-Maadeed, Ahmed Bouridane, Elhadj Benkhelifa,March 2019 Science Direct

[25] Analysis and evaluation of handwriting in patients with Parkinson's disease using kinematics, geometrical and non-Linear features by C.D.Rios-Urrego,J.C.Vasquez-Correa,J.F.Vargas-Bonilla,E.Noth,F.Lopera, May 2019 Science Direct

[26] Thresholding: A Pixel-Level Image Processing Methodology Preprocessing Technique for an OCR System for the Brahmi Script by H. Devi,, Ancient Asia Journal Vol. 1, Dec2006.

[27] A review on challenges and Applications of Digital Graphology by Ranjitha , Akshatha Prabhu , IEEE Xplore

[28] Deep adaptive learning for writer identification based on single handwritten word images by Sheng He,Lambert Schomaker , Science direct egateway, April 2019.

[29] Writer identification using text line based features by U.-V. Marti, R. Messerli, and H. Bunke,, Proc. Sixth Intl Conf. Document Analysis and Recognition (ICDAR),2001.

[30] Prof. SeemaKedar, Ms. VaishnaviNair, Ms. ShwetaKulkarni, Personality Identification through Handwriting Analysis: A Review, International Journal of Advanced Research in Computer Science and software engineering, Volume 5, Issue 1, January 2015

[31] "Cognitive Correlates of Lecture Note Taking: Handwriting Speed and Attention" by Alberto Manzi, Steven Martinez, Yana Durmysheva, North American Journal of Psychology, 2017. Handwriting Styles Clustering: Feature Selection and Feature Space Analysis based on Online Input by Karyna Korovai, Oleksandr Marchenko, Third International Conference on Data Stream Mining & Processing

[32] Handwriting Styles Clustering: Feature Selection and Feature space Analysis based on Online Input by KAryna Korovai and Oleksandr Marchenko, IEEE Third Inernational Conference on Data Mining and Processing, 2020.

[33] An off-line Handwriting Recognition Employing Tensorflow by Hao Zeng, International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering, 2020

[34] The computational analysis of handwriting in individuals with psychopathic personality disorder by Barbara Gawda, DOI in 2019

[35] Feature Extraction from Text Images to study Individuality of Handwriting by Salankara Mukherjee and Ishita De Ghosh, 4th International Conference on Computational Intelligence and Network, IEEE, 2019

[36] Chinese character handwriting: A large-scale behavioural study and a database by Ruiming Wang, Shuting Huang, Yacong Zhou, Zhenguang G. Cai, Behavioral Research Methods, The Psychonomic Society, 2019.

[37] “online and offline handwriting recognition: A comprehensive survey” by R Plamondon and S Srihari, IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE, 2000.

[38] Mirvaziri H, Javidi MM, Mansouri N (2009) Handwriting rec- ognition algorithm in different languages: survey. In: Proceed- ings of the 1st international visual informatics conference on visual informatics: bridging research and practice, Kuala Lum- pur. Springer

[39] Modified Dynamic Movement Primitives for Robotics Applications Exemplified on Handwriting by Tomas Kulvicius, KeJun Ning, Minija Tamosiunaite, and Florentin Wo ¨rgo ¨tter , IEEE TRANSACTIONS ON ROBOTICS, VOL. 28, NO. 1, FEBRUARY 2012

[40] Champa H N and Dr. K R AnandaKumar, “Artificial neural network for human behavior prediction through handwriting analysis”, International Journal of Computer Applications (0975 8887) Volume 2 No.2, May 2011.