ISBN 978-9937-0-9019-3

PREDICTING PERSONALITY TRAITS FROM HANDWRITING

1st Bharat Raj Joshi
Department of Computer and Electronics Engineering
Kantipur Engineering College
Lalitpur, Nepal
br14joshi@gmail.com

2nd Sumee Adhikari
Department of Computer and Electronics Engineering
Kantipur Engineering College
Lalitpur, Nepal
loona.sumuu@gmail.com

***Abstract*—Whenever we listen to or meet a new person we try to predict personality attributes of the person. Our behavior towards the person is hugely influenced by the predictions we make. Personality is made up of the characteristic patterns of thoughts, feelings and behaviors that make a person unique. Your personality affects your success in the role. Recognizing about yourself and reflecting on your personality can help you to understand how you might shape your future. Various approaches like personality prediction through speech, facial expression, video, and text are proposed in literature to recognize personality. Personality predictions can be made out of one's handwriting as well. The aim of this research project is to examine validity of the Graphological method to assess personality traits. This research project outlines the development of a Supervised Neural Network Model for the personality prediction. Machine learning techniques have been widely used in various fields for complex pattern matching and making decisions. The research based system based on neural networks was proposed that aims to determine the Big Five personality traits with handwriting features from data sets containing both predefined and random texts. The predefined texts add more value if enforced on writers in the training stage. Handwriting Analysis or Graphology is a consistent method for perceiving, surveying and understanding personality through the strokes and structures revealed by handwriting. Handwriting reveals authentic character including excited, fears, validity and various others. Capable handwriting experts called graphologist every now and again recognize the writer with the touch of handwriting. This research project presents a prediction on a big five personality traits from handwriting using FFM and Graphological analysis.**

***Index Terms*—Supervised Neural Network Model, Personality Prediction, Machine Learning, Handwriting Analysis, Graphology, FFM**

I. INTRODUCTION

Handwriting Analysis or Graphology is a scientific method of identifying, evaluating and understanding personality through the strokes and patterns revealed by handwriting which is supposed to be revealing true personality. Handwriting has been used for centuries as a way of communication and expression for humans, but only recently it links to brain activity and the psychological aspects of humans have been studied. The psychological study of handwriting with the purpose of determining the personality traits[1], psychological states, temperament, or the behavior of the writer is called graphology and is still a debatable domain as it lacks a standard, most of the handwriting interpretations being done subjectively by trained graphologists.

Handwriting analysis is not a document examination, which involves the examination of a sample of handwriting to determine the author. Handwriting is often referred to as brain writing. Each personality trait is represented by a neurological brain pattern[2]. Each neurological brain pattern produces a unique neuromuscular movement that is the same for every person who has that particular personality trait. When writing, these tiny movements occur unconsciously. Each written movement or stroke reveals a specific personality trait. Graphology is the science of identifying these strokes as they appear in handwriting and describe the corresponding personality trait[3]. Therefore, handwriting is, from this perspective, an accurate mirror of people's brain. The Big Five Personality Traits, also known as five-factor model FFM and the OCEAN Model, is a taxonomy for personality traits.

Now, let us be enlightened about personality traits in detail. What makes someone who they are? Each person has an idea of their own personality type – if they are bubbly or reserved, sensitive or thick skinned. Psychologists who try to tease out the science of who we are define personality as individual differences in the way people tend to think, feel and behave. There are many ways to measure personality, but psychologists have mostly given upon trying to divide humanity neatly into types. Instead, they focus on personality traits. The Big Five were developed in the 1970s by two research teams i.e. Paul Costa and Robert R. McCrae of the National Institutes of Health and Warren Norman and Lewis Goldberg of the University of Michigan at Ann Arbor and the University of Oregon[1]. The most widely accepted of these traits with the handy OCEAN mnemonic are the Big Five: i. Openness ii. Conscientiousness iii. Extraversion iv. Agreeableness v. Neuroticism The Big Five are the ingredients that make up each individual's personality. A person might have a dash of openness, a lot of conscientiousness, an average amount of extraversion, plenty of agreeableness and almost no neuroticism at all. Or someone could be disagreeable, neurotic, introverted, conscientious and hardly open at all. Here's what each traits entails: i. Openness is shorthand for "openness to experience". People who are high in openness enjoy adventure. They're curious and appreciate art, imagination and new things. The motto of the open individual might be "Variety is the spice of life." People low in openness are just the opposite: They prefer to stick to their habits, avoid new experiences and probably aren't the most adventurous eaters. Changing personality is usually considered a tough process, but openness is a personality trait that's been shown to be subject to change in adulthood. ii. Conscientiousness is the people who are organized and have a strong sense of duty. They're dependable, disciplined and achievement-focused. You won't mind conscientious types jetting off on round-the-world journeys with only a backpack; they're planners. People low in conscientiousness are more spontaneous and freewheeling. They may tend toward carelessness. Conscientiousness is a helpful trait to have, as it has been linked to achievement in school and on the job. iii. Extraversion is possibly the most recognizable personality trait of the Big Five. The more of an extrovert someone is, the more of a social butterfly they are. Extroverts are chatty, sociable and draw energy from crowds. They tend to be assertive and cheerful in their social interactions. Whereas, introverts are on the other hand, need plenty of alone time, perhaps because their brains process social interaction differently. Introversion is often confused with shyness, but the two aren't the same. Shyness implies a fear of social interactions or an inability to function socially. Introverts can be perfectly charming at parties – they just prefer solo or small group activities. iv. Agreeableness measures the extent of a person's warmth and kindness. The more agreeable someone is, the more likely they are to be trusting, helpful and compassionate. Disagreeable person are cold and suspicious of others, and they're less likely to cooperate. Being envious, which can lead to people being perceived as not agreeable, was found to be the most common personality type. Envious people feel threatened when someone else is more successful than they are. v. Neuroticism is the trait in which people high in neuroticism worry frequently and easily slip into anxiety and depression. If all is going well, neurotic people tend to find things to worry about. Unsurprisingly, neuroticism is linked with plenty of bad health outcomes. Neurotic people die younger than the emotionally stable, possibly because they turn to tobacco and alcohol to ease their nerves. In contrast, people who are low in neuroticism tend to be emotionally stable and even-keeled. This project aims to build a system that is able to automatically analyze a set of handwriting features and evaluate the personality of the writer using the Five-Factor Model (FFM). To test this system the first dataset is proposed, that links the FFM personality traits to handwriting features. Proposed System offers an attractive alternative to the standard FFM questionnaire or psychological interviews that are currently used for evaluating personality, because it is easier to use, involves less effort and is faster. It aims the highest accuracy compared to other methods.

II. Literature Review

Handwriting Analysis or Graphology is a scientific method of identifying, evaluating and understanding personality through the strokes and patterns revealed by handwriting. Among the many aspects of handwriting that can serve as scheme to predict personality traits are baseline, size of letter, connecting strokes, spacing between letters, words and lines, starting strokes, end-strokes, word-slant, speed of handwriting, width of margins, and others. Writer individuality rests on the hypothesis that each individual has consistent handwriting, which is distinct from the handwriting of another individual. However, this hypothesis has not been subjected to rigorous scrutiny with the accompanying experimentation, testing, and peer review. As mentioned previously, currently, there is no standard developed in predicting behavior based on handwriting, the majority of Graphological analysis being done by specialized graphologists. However, research was conducted in the area of computer science which aimed to create such systems in order to recognize the behavior from Handwriting in an easier way and also to standardize the Graphological analysis. In the next paragraphs, we present the state-of-the-art in this area as well as several studies which made use of handwriting to determine the psychological traits or mental status of individuals.

Mihai Gavrilescu and Nicolae Vizireanu [1] proposed the first non-invasive three-layer architecture in literature based on neural networks that aimed to determine the Big Five personality traits of an individual by analyzing offline handwriting. They also presented the first database in literature that links the Big Five personality type with the handwriting features collected from 128 subjects containing both predefined and random texts. Testing their novel architecture on those database, they showed that the predefined texts add more value if enforced on writers in the training stage, offering accuracies of 84.4in intra-subject tests and 80.5% in inter-subject tests when the random dataset were used for testing purposes, up to 7% higher than when random datasets were used in the training phase. They obtained the highest prediction accuracy for Openness to Experience, Extraversion, and Neuroticism (over 84%), while for Conscientiousness and Agreeableness, the prediction accuracy was around 77%.

Behnam Fallah and Hassan Khotanlou describe in [2] a research with a similar purpose as the one conducted in this paper, aiming to determine the personality of an individual by studying handwriting. The Minnesota Multiphasic Personality Inventory (MMPI) is used for training their system and a Hidden Markov Model (HMM) is employed for classifying the properties related to the target writer, while a neural network (NN) approach is used for classifying the properties which are not writer-related. The handwriting image is analyzed by these classifiers and compared with the patterns from the database, the output being provided in the form of the personality of the writer on the MMPI scale. Their system offers over 70 percent accuracy at this task.

Similarly, in [3], an instrument for behavioral analysis is described with the task of predicting personality Traits from handwriting. The approach takes into account the following handwriting features: letter "t," lower loop of the letter "y," the pen pressure, and the slant of writing.

Similarly, in [4], it is proposed a way to describe handwriting based on geometric features which are combined using random forest algorithms and kernel discriminant analysis. The system is able to predict gender with 75.05 percent, age with 55.76 percent, and nationality with 53.66 percent when all the writers were asked to write the same text, and 73.59 percent for gender prediction, 60.62 percent for age prediction, and 47.98 percent for nationality prediction when each subject wrote a different text. Since handwriting analysis is a complex task requiring multiple techniques in order to analyze the multitude of handwriting features. One of the most challenging tasks is the one of segmenting the handwritten image into text lines and words.

For this, the Vertical Projection Profile (VPP) [5] method has shown the most promising results and this is the one that we use in this paper for both row and word segmentation. Regarding feature classification, different classifiers are used successfully for each of the handwriting features. For example, for lowercase letters "t" and "f," the most common method used is template matching, for writing pressure gray-level thresholding methods are employed, while for connecting strokes the Stroke Width Transform (SWT) has shown the best classification accuracy compared to other state-of-the-art methods. In the following sections, we present in detail the classifiers used for each of the handwriting features analyzed in the current research project. With all these in mind, the current research proposes a novel non-invasive neural network-based architecture for predicting the Big Five personality traits of a subject by only analyzing handwriting. This system would serve as an attractive alternative to the extensive questionnaire typically used to assess the FFM personality traits and which is usually cumbersome and non-practical, as well as avoid the use of invasive sensors. We focus our attention on handwriting because it is an activity familiar to almost everyone and can be acquired fast and often.

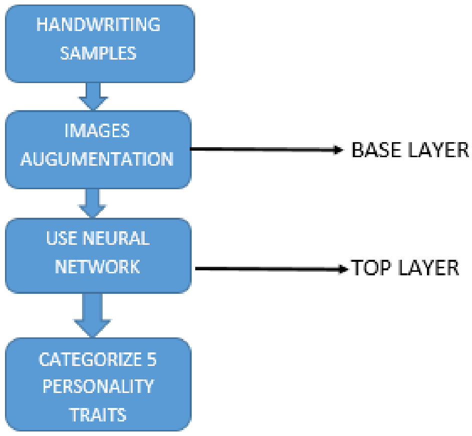

Fig. 1: System Mechanism

III. Problem On hand

It is difficult to predict personalities of people without using an effective system. It takes a whole lot of time and effort to organize FFM questionnaire or psychological interviews to predict personality. It also requires a lot of extra personnel such as psychologists to do the graphological analysis.

IV. Methodology

A. System Mechanism

Our system targets the every fields where the personality of person plays an important role. The model type that we used is Sequential. Sequential is the easiest way to build a model in Keras that allows us to build a model layer by layer. Our system predicts the personality of the people by using their handwriting and classify their personality among the five different traits and displays each traits in certain percentage value, each of them are out of 100%.

We have performed Two Layer Architecture i.e. Base Layer and Top Layer with Image Augmentation and using CNN respectively. Image Augmentation is used to expand our training dataset artificially by creating the modified versions of images existing in the dataset. It creates the variations of the image that provides more images to train on. Then they were passed to neural network(CNN) that is composed of multiple building blocks, such as convolution layers, pooling layers, and fully connected layers, and is designed to automatically and adaptively learn spatial hierarchies of features through a backpropagation algorithm. And then finally the five personality traits were obtained as output from the neural network.

The mechanism of our system is diagrammatically shown in the figure below

1) Handwriting Samples: The different types of handwritten samples were collected that belong to our different traits as per their features. The clean and validated dataset were downloaded from the kaggle website which contained dataset for each traits in

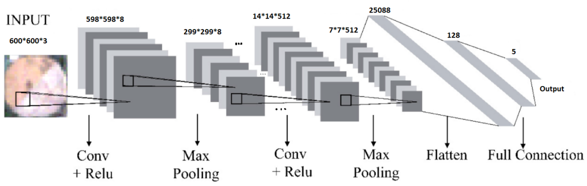

Fig. 2: Convolutional Neural Network[6]

separate folders. They were insufficent for the training and testing purpose. So in collaboration with our supervisor we created additional dataset following the specified criterias for each traits in the refrence paper[1]. Then these dataset were used for training and the testing purpose. It was made sure that approximately 80% of the total dataset were used for the training and rest of the 20% dataset were used for the testing. Accuracy and Loss were the major evaluation criteria for this model.

2) Image Processing: Image processing is a method to perform some operations on an image, in order to get an enhanced image or to extract some useful information from it. Due to the limited number of dataset we went through image augmentation which is the part of the image processing. It can be further explained below:

Image Augmentation: It is the technique that we used to expand our training dataset artificially by creating the modified versions of images existing in the dataset. It creates the variations of the image that provides more images to train on. By creating the image generator by calling ImageDataGenerator() function which provides a quick and easy way to augment images and provides different augmentation techniques like rotation, flip, shift, shear, zoom, etc. by passing different parameters like shear range for shearing, zoom range for zooming, scale pixels for rescaling were carried out in this process. This task comes under the Base Layer.

The dataset was rescaled between 0 and 1 and then different techniques were applied like horizontal flipping was made true to flip the pictures horizontally. Also the shearing range and zoom range was provided to shear and zoom pictures randomly.

B. Use Neural Network

Convolutional neural network is a class of deep learning methods which has become dominant in various computer vision tasks. Convolutional neural network is composed of multiple building blocks, such as convolution layers, pooling layers, and fully connected layers, and is designed to automatically and adaptively learn spatial hierarchies of features through a backpropagation algorithm. The first two, convolution and pooling layers, perform feature extraction, whereas the third, a fully connected layer, maps the extracted features into final output, such as classification. We used Convolutional Neural Network for the further processing. The tasks under the CNN can be categorized under two categories viz. Feature Extraction and the Classification. Going into them in detail:

1) Feature Extraction:

1) Convolutional Layer

   a) The input images are initially provided to the first layer of feature learning i.e convolutional layer. This layer changes the input image into the pixels, we can say a 2D picture is changed to 2D form of image matrix that contains the different pixel values. It preserves the relationship between pixels by learning image features using small squares of input data. It also takes the filter as an input. By multiplying the pixel values contained by image matrix and the filter matrix this layer produces an output matrix that contains convolved features. This multiplication between image matrix and filter matrix is known as feature mapping. This multiplication between image matrix and filter matrix is known as feature mapping. Its can be illustrated as:
   Let height, width and of input image be

   $$H_c, W_c$$

   and of filter matrix be

   $$H_f \ , W_f$$

   respectively; Also let the channel size be 'C'. Then above expressions can be explained mathematically as:
   Size of input image :

   $$(H_c * W_c * d) \tag{1}$$

   Size of filter :

   $$(H_f * W_f) \tag{2}$$

   Let size of feature map is :

   $$((H_c - H_f + 1) * (W_c - W_f + 1) * C) \tag{3}$$

   b) The filter moves to the right with a certain Stride Value till it parses the complete width. Moving on, it hops down to the beginning (left) of the image with the same Stride Value and repeats the process until the entire image is traversed. If omitted, the default is 1: return every character in the requested range. The default value of stride is used that parses the complete width by moving filter to right direction with stride value 1.

   c) The used activation function is ReLU. It is the widely used activation function that introduces the non-linearity. It changes all the negative values of output matrix given by convolution layer to 0. The performance of ReLU is better in compared to other nonlinear functions like tanh or sigmoid. Mathematically, it can be shown as:

$$R(z) = max(0, X) \tag{4}$$

where X is the initial value before passing to activation function as input. The maximum value between O and X is output value from this activation function.

ii. Max pooling was used to reduce the spatial dimensions of the output volume. This layer is responsible for reducing the parameters of the images specially when they are very large. It can be done in many ways like max pooling, average pooling and sum pooling. Max Pooling returns the maximum value from the portion of the image covered by the Kernel. On the other hand, Average Pooling returns the average of all the values from the portion of the image covered by the Kernel. Max pooling was used in our model. It uses the maximum value out of different values under the pool size provided. Generally pool size of 2*2 is used that mostly results in 4 values replaced by one maximum value among them. In this case, the dimensionality of the 2d array from convolutional layer is reduced to half. If we have feature map of size

$$H_f * W_f$$

then after pooling with pool size P they are reduced as

$$(\frac{H_f}{P} * \frac{W_f}{P}) \tag{5}$$

, where

$$H_f, W_f$$

have even value if they have odd value then first 1 is subtracted from them and then they are feed to above equation (4.4).

This 'Feature Learning' process can be repeated as many times as per requirement until satisfied. Number of repetations can be analyzed by looking at the size of the picture to be used and according to it they can be segregated. After completing all the repetitions now the result is passed to the 'classification' category of the convolutional Neural Network thich comes under Top Layer which is described under algorithm.

V. BackPropagation Algorithm

As the task is a pattern recognition task and also considering that our architecture is bottom-up with no feedback loops, we use a feed-forward neural network. Also, with the same premises in mind, the training method will be used is backpropagation, which has proven to be very effective and offers fast learning. The back- propagation algorithm cycles through a forward

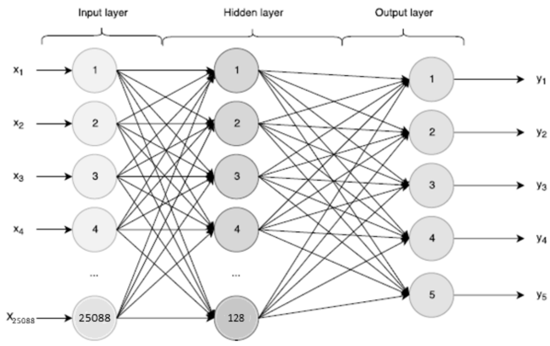

Fig. 3: Neural Network Structure[1]

pass followed by a backward pass through the network. The algorithm alternates between these passes until it "learns" to make good classifications. Forward Pass: In this pass outputs of all neurons in the network are computed. Neuron outputs are computed for each layer from the first hidden layer to the subsequent layers. Outputs of one layer are the inputs of the succeeding layer. This continues layer by layer until we reach the output layer and compute the outputs for this layer. Backward Pass: The error calculated in the forward pass is propagated backwards from the output layer to the input layer. Weight modification is done as the error is propagated.

1) Back Propagation: Detail Algorithm:

i. Initialize the weights and bias to random values.

ii. Consider input values I and target value T .

iii. Do forward pass to calculate hidden layer values

$$I_j = \sum(w_{ij} * O_i) + b_j \tag{6}$$

iv. Take a activation function for input to hidden layer

$$R(z) = max(0, x) \tag{7}$$

v. Calculate input for output layer

$$I_k = \sum(w_{jk} * O_j) + b_k \tag{8}$$

vi. Take a activation function for hidden to output layer i.e Sigmoid activation function.

$$f(x) = \frac{1}{1 + e^{-I_k}} \tag{9}$$

vii. Calculate Errors:

- Error in output layer

$$E_k = O_k(1 - O_k)(T - O_k) \tag{10}$$

- Error in hidden layer

$$E_j = O_j(1 - O_j)\sum(E_k * w_{jk}) \tag{11}$$

viii. Updating Weight and bias

- Hidden layer's weight and bias are updated using following formula:

$$\Delta(w_{ij}) = \alpha * E_j * O_i \tag{12}$$

$$w_{ij(new)} = w_{ij(old)} + \Delta(w_{jk}) \quad (13)$$

$$\Delta(b_j) = \alpha * E_j \quad (14)$$

$$b_{j(new)} = b_j + \Delta(b_j) \quad (15)$$

- Output layer's weight and bias are updated using following formula:

$$\Delta(w_{jk}) = \alpha * E_k * O_j \quad (16)$$

$$w_{jk(new)} = w_{jk(old)} + \Delta(w_{jk}) \quad (17)$$

$$\Delta(b_k) = \alpha * E_k \quad (18)$$

$$b_{k(new)} = b_{k(old)} + \Delta(b_k) \quad (19)$$

ix. Continue from step (ii) to (vi) untill all training sets values are satisfied.

As the algorithms name implies, the errors (and therefore the learning) propagate backwards from the output nodes to the inner nodes. Therefore, backpropagation is used to calculate the gradient of the error of the network with respect to the networks modifiable weights. This gradient is almost always then used in simple stochastic gradient descent algorithm to find weights that minimize the error. Often the term "backpropagation" is used in a more general sense, to refer to the entire procedure encompassing both the calculation of the gradient and its using stochastic gradient descent. Back-propagation usually allows quick convergence on satisfactory local minima for error in the kind of networks to which it is suited.

2) Implementation of Algorithm: In our project we created a model using a backpropagation algorithm forming one input layer and one hidden layer with activation function 'relu', which is a default activation function for many types of neural network as it is easier to train and achieve better performance. This activation function uses piecewise linear function that will output the input directly if it is positive, otherwise, it will output zero. Along with input and hidden layers we have output layer that is activated by sigmoid activation function which is mostly preferred in output layers since it gives multiple true answers in our model i.e classifying into five traits. We used Flatten as a connection between the convolution and dense layers.

We compiled our model by taking three parameters i.e 'adam' as a optimizer that adjusts the learning rate throughout training, 'sparse categorical crossentropy' as our loss function that saves time in memory as well as computation because it simply uses a single integer for a class, rather than a whole vector and 'accuracy' as our metric to see the accuracy score on the validation set when we train the model.

VI. Result

Our main objective was to build a system that can predict the personality by using the handwriting of people under 5 major traits. We collected dataset for each personality traits which was the most difficult task while carrying out project. Since we had not enough dataset we performed image augmentation that helped us to expand our training dataset by creating the modified versions of images that we had collected. The different variations were done in this category like shearing, zooming which were done randomly, flipping that was done horizontally. To ensure the size of data between 0 and 1 rescaling process was carried out. Then after this task of image processing we build a sequential model in which we can created our model layer-by-layer. Under the convolutional neural network the data or image went through those layers repeatedly. The process can be described with an example as shown below:

VII. Feature Extraction Phase

i. Convolutional Layer
In this layer the image is changed to pixel value in 2D array. For eg, though the dimension of the images is high for easy understanding, let us say the image is converted to 6*6 2d array of pixels as shown below:

0	1	0	0	1	0
1	0	1	0	1	0
0	0	1	0	1	0
0	0	1	0	1	1
0	0	0	1	1	0
0	1	0	1	0	1

TABLE I: Image sample changed to 2D matrix of pixel value

Also we had used a filters of size 3*3 which is multiplied by the convolutional matrix to obtain the feature map. For example, let is consider a filter matrix be as shown below :

0	1	0
1	0	1
0	0	1

TABLE II: Filter matrix of size 3*3

Then to extract the feature map the above 2d convolutional matrix was multiplied by the 3*3 filter size matrix . We obtained a matrix as shown below:

4	0	3	1
2	1	3	1
1	2	3	2
0	3	1	3

TABLE III: Extracted Feature Map

eg, if the provided image size is (600*600*3) and the filter size is (3*3) then the feature map will be,
Feature Map size = ((600-3+1)*(600-3+1)*8)
Feature Map size = 598*598*8

ii. Activation function
The activation function called ReLU was used which was resposible for changing all negative values to the zero. It can be described by taking an valid example as shown below. All the pixel value of feature set which were negative values were changed to 0 value as shown below. The array or matrix obtained in this way were then sent to next layer, pooling layer for further processing.

-4	0	3	-1	1	0
6	-4	0	1	-1	3
2	4	-1	0	-2	-4
3	0	-1	2	-1	4
1	3	-2	1	0	2
-1	0	-2	2	1	-3

TABLE IV: Array before applying ReLU function

0	0	3	0	1	0
6	0	0	1	0	3
2	4	0	0	0	0
3	0	0	2	0	4
1	3	0	1	0	2
0	0	0	2	1	0

TABLE V: Array After applying ReLU function

iii. Pooling
Now this matrix undergoes a pooling layer. The pooling can be done as max pooling, average pooling or sum polling but here we preferred max pooling of size 2*2. The dimension of above matrix decreases after pooling without distrubing its features. The matrix produced is tabulated below

6	3	3
4	2	4
3	2	2

TABLE VI: Max pooling using pool size 2*2

If the provided size of feature map to pooling layer is (598*598*8) and the pool size is (2*2) then the 2d array after pooling will be of size,
Pooled size = ((598/2)*(598/2)*8)
Pooled size = (299*299*8)
If the size of feature map to pooling layer is (297*297*16) and the pool size is (2*2) then the 2d array after pooling will be of size
Pooled size = (((297-1)/2)*((297-1)/2)*16), since feature map has dimension of odd value, first 1 is subtracted from feature map and then calculation is done.
Pooled size = (148*148*16) ,

The above process i.e sequence of going through convolutional layer, normalization layer and pooling layer was repeated for five more times. Since our input images has the size 600*600*3 we went to repeat this cycle for 5 more times.
And finally we got comparatively small sized 2D array ie matrix that was changed to 1D array by flattening using flatten() function. Then so found data were feed into the dense layer of the classification category of the convolutional neural network which is an input layer for classification algorithm. for example the flatten array of the above matrix looks like:

6
3
3
4
2
4
3
2
2

TABLE VII: Array from pooling layer after flattening

These values are taken as input for the input layers of the classification algorithm and further calcuation or process takes places which is described below in the classification section.

VIII. Classification Phase

The classification category was done using the back-propagation algorithm. Each steps with example is described as:

For simplicity, Let's assume that we have input layer with five input nodes, hidden layer with two hidden nodes and one output layer with one node .

i. Input Layer: Input layer gets all the values flattened, from 'feature extracting' category. In this layer no mathematical calculations were done. The values were passed to hidden layers. For simplicity let's consider we have five input nodes with values 6, 3, 4, 2, 4

ii. Hidden Layer: Hidden layers get the certain values which were calculated using input values and the random weight along with the random bias value between 0 and 1.

- The value of two nodes of hidden layer were calculated as:

$$H1 = w_{11} * I_1 + w_{21} * I_2 + w_{31} * I_3 + w_{41} * I_4 + w_{51} * I_5 + bias1$$

$$H1 = 0.2{*}6{+}0.3{*}3{+}0.3{*}4{+}0.15{*}2{+}0.25{*}4{+}0.15$$

Th random values of weights and bias were used as above and we obtained value as H1 = 4.75 Similar calculation was done for H2 and we obtained H2 = 4.57

- Now the output from the hidden layers were calculated by using the relu activation function we obtained as
  OH1 = 4.75, OH2 = 4.57
  The negative values if found any were changed to 0 by relu function.

iii. The input to the output layer were also calculated as

$$O = OH1 * w_{21} + OH2 * w_{22} + bias2$$

$$O = 4.75 * 0.1 + 4.56 * 0.15 + 0.2$$

$$O = 1.359$$

iv. The output from the output node was calculated using its sigmoid activation function as:

$$O_o = \frac{1}{1 + e^{-O}}$$

$$O_o = 0.7956$$

v. Error calculation

- Then error at output node was calculated as

$$E_O = O_o(1 - O_o)(T - O_o)$$

$$E_O = 0.7956(1 - 0.7956)(0 - 0.7956)$$

$$E_O = -0.1293$$

- The error at hidden layer was also calculated as

$$E_{H1} = O(1 - O) \sum (E_o * w_{21})$$

$$E_{H1} = 4.75(1 - 4.75) * (-0.1293) * 0.1$$

$$E_{H1} = 0.2303$$

also

$$E_{H2} = 4.57(1 - 4.57) * (-0.1293) * 0.15$$

$$E_{H2} = 0.3164$$

vi. After calculating the error the weight and bias updating process was carried out as

$$\Delta w_{11} = 0.1 * (0.2303) * 6$$

$$\Delta w_{11} = 0.1398$$

Hence,

$$w_{11new} = 0.1398 + 0.2$$

$$w_{11new} = 0.33818$$

Also the bias was updated as

$$\Delta(bias1) = 0.1 * (0.1398)$$

$$\Delta(bias1) = 0.01398$$

$$bias1_{new} = 0.15 + (0.01398)$$

$$bias1_{new} = 0.16398$$

Similarly other weights w12, w13, w14, w14 and bias were updated.
Also the weight and bias of the output layer's node were updated as

$$\Delta w_{21} = 0.1 * (-0.1293) * 4.75$$

$$\Delta w_{21} = -0.06142$$

Hence,

$$w_{21new} = -0.06142 + 0.2$$

$$w_{21new} = 0.1385$$

Also the bias2 was also updated as

$$\Delta(bias2) = 0.1 * (-0.06142)$$

$$\Delta(bias2) = -0.006142$$

$$bias2_{new} = 0.15 + (-0.006142)$$

$$bias2_{new} = 0.143858$$

Similarly w22 was also updated along with its bias.
This process was repeated for other training dataset and later we continue this for more epoch (15 in our model) to obtain much accurate solution and finally the output values were obtained from five output nodes.
And when the picture was provided to the model to test the personality traits, picture goes under all above processes of Convolutional Neural Network and at last the output value of trained data and this data to be tested were compared and with respect to value the output was produced. The sigmoid activation function was used in this case since our model gives the multiple true values as output. Sigmoid was best choice since system displays the each amount of traits that handwriting consists of each per cent, neither the total sum of all traits becomes 100 nor it displays just one trait as maximum and other as minimum in our system.

IX. Model Summary

The CNN model that was developed had six layers repeating in sequence for feature extraction that are convolutional layer and max pooling layer. After completing six cycles then the value obtained as two dimesional array were flattened to get one dimensional array. Then the each values were passed to the input layer of the classification phase where no further mathematical calculations were carried out and then further calculations for other layers were carried out as discussed above. The CNN model can be summarized as shown in given figure that shows the output shape along with its respective layer:

```
Model: "sequential_6"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_35 (Conv2D)           (None, 598, 598, 8)       224
_________________________________________________________________
max_pooling2d_34 (MaxPooling (None, 299, 299, 8)       0
_________________________________________________________________
conv2d_36 (Conv2D)           (None, 297, 297, 16)      1168
_________________________________________________________________
max_pooling2d_35 (MaxPooling (None, 148, 148, 16)      0
_________________________________________________________________
conv2d_37 (Conv2D)           (None, 146, 146, 32)      4640
_________________________________________________________________
max_pooling2d_36 (MaxPooling (None, 73, 73, 32)        0
_________________________________________________________________
conv2d_38 (Conv2D)           (None, 71, 71, 128)       36992
_________________________________________________________________
max_pooling2d_37 (MaxPooling (None, 35, 35, 128)       0
_________________________________________________________________
conv2d_39 (Conv2D)           (None, 33, 33, 256)       295168
_________________________________________________________________
max_pooling2d_38 (MaxPooling (None, 16, 16, 256)       0
_________________________________________________________________
conv2d_40 (Conv2D)           (None, 14, 14, 512)       1180160
_________________________________________________________________
max_pooling2d_39 (MaxPooling (None, 7, 7, 512)         0
_________________________________________________________________
flatten_5 (Flatten)          (None, 25088)             0
_________________________________________________________________
dense_11 (Dense)             (None, 128)               3211392
_________________________________________________________________
dense_12 (Dense)             (None, 5)                 645
=================================================================
Total params: 4,730,389
Trainable params: 4,730,389
Non-trainable params: 0
_________________________________________________________________
```

Fig. 4: Summary of CNN Model

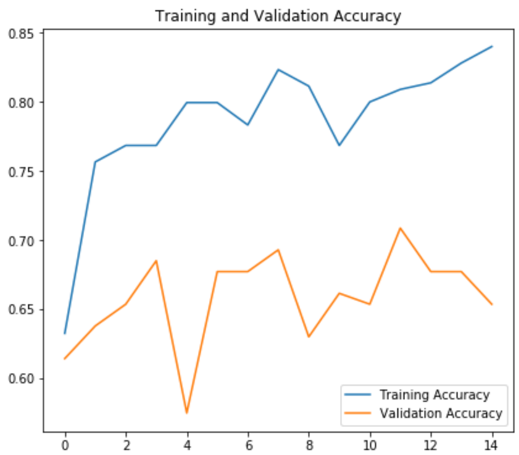

Fig. 5: Training accuracy and validation accuracy plot

X. Accuracy and Loss

The figure below shows the accuracy plot of our system that tabulates the progress on accuracy with respect to number of epoch. The training accuracy of system was approximately 85%.The testing accuracy increased from 64% initially to the 84% at the last epoch.

Also the testing and validation loss are shown below. The validation loss was high due to the more number of outliers in our dataset. We got testing loss less than validation loss which was positive point for us and it is decreasing as the number of epochs increases as shown in graph.

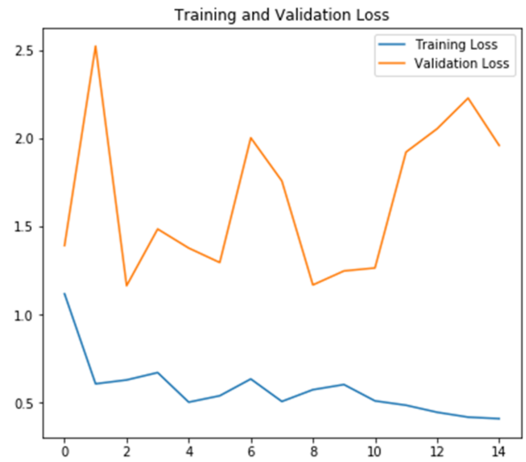

Fig. 6: Training loss and validation loss plot

XI. Limitations

Despite of some changes in the initial plans the objectives were fulfilled and output was observed. The system have few limitations as well, that are:

i. Our system doesn't give accurate result for the languages other than English.
ii. It doesn't displays the accurate result for typed and printed text pictures.

XII. Future Enhancement

Despite of some limitations, we were able to achieve targeted objectives and develop our system which can be enhanced in future.

i. The system can be used for different languages by using dataset of different languages.
ii. The model can give specific trait with maximum value as output.

XIII. Conclusion

The training accuracy of system was approximately 85%.The testing accuracy increased from 64% initially to the 84% at the last epoch. The validation loss was high due to the more number of outliers in our dataset. We got testing loss less than validation loss which was encouraging and it is decreasing as the number of epochs increases. Our system doesn't give accurate result for the languages other than English but this system can be extended to support different languages by using dataset of different required languages.

Acknowledgment

First and foremost, we would like to thank Computer and Electronics Department of Kantipur Engineering College for giving us this opportunity to prepare this project. We would like to express our sincere gratitude towards Er. Rabindra Khati, HOD for his precious encouragement. We would like to thank Er. Anup Kc, Deputy HOD, for his constant guidance and inspiring lectures. We are highly indepted to Er. Dipesh Shrestha and Er. Bishal Thapa for co-operating with us and letting us carry out this project smoothly.Also. we would like to thank our supervisor Er. Ujjwal Prajapati for guiding us in completion of this project. Without his supervision and suggestions, it would have been difficult for us.

Last but not the least, we are thankful to all our teachers and people who have contributed in preparation of this project. We have tried to make this project as accurate as possible. Suggestions and comments from readers are highly appreciable.

REFERENCES

[1] Gavrilescu, Mihai & Vizireanu, Nicolae. (2018). Predicting the Big Five personality traits from handwriting. EURASIP Journal on Image and Video Processing. 2018. 10.1186/s13640-018-0297-3.

[2] B Fallah, H Khotanlou, in Artificial Intelligence and Robotics (IRANOPEN). Identify human personality parameters based on handwriting using neural networks (April 2016)

[3] HN Champa, KR Anandakumar, Automated human behavior prediction through handwriting analysis. 2010 First International Conference on Integrated Intelligent Computing (ICIIC), 160–165 (August 2010)

[4] S Maadeed, A Hassaine, Automatic prediction of age, gender, and nationality in offline handwriting. EURASIP Journal on Image and Video Processing 2014, 10 (December 2014)

[5] V Papavassiliou, T Stafylakis, V Katsouro, G Carayannis, Handwritten document image segmentation into text lines and words. Pattern Recogn. 43, 369–377 (2010)

[6] A Yu, How to teach a computer to see with Convolutional Networks.(November 2018)