

BEHAVIOR PREDICTION BASED ON HANDWRITING USING MACHINE LEARNING

Poonam Dhoot

Dissertation submitted in partial fulfilment of the
requirements for the degree of

Master of Science (MSc) in Data Analytics

at



Supervisor: Dr. Shahram Azizi Sazi

August 2019

Table of Contents

Chapter 1 – Introduction	5
History of Graphology	6
Chapter 2 – Literature Review	7
Introduction	7
Conclusion:.....	15
Chapter 3 - Methodology.....	16
Stage One: Determine Business Understanding.....	17
Stage Two: Data Understanding	18
Stage Three: Data Preparation.....	20
Chapter 4 - Implementation & Modelling.....	25
Decision Tree:	27
Naïve Bayes:.....	27
Support Vector Machine (SVM).....	28
K – Nearest Neighbours (KNN).....	29
Random Forest.....	29
Step 1: Load the dataset	30
Step 2: Clean the dataset.....	31
Step 3: Split the dataset.....	31
Step 4: Train the model	32
R code for.....	32
Chapter 5 - Evaluation & Results	35
Decision Tree	36
Naïve Bayes.....	37
Support Vector Machine - Poly.....	37
Support Vector Machine - RBF.....	37
K – Nearest Neighbours	38
Random Forest.....	38
Chapter 6 - Conclusion & Future work.....	40
References:	41

Declaration

I declare that this dissertation that I have submitted to Dublin Business School for the award of Master of Science in Data Analytics is the result of my own investigations, except where otherwise stated, where it is clearly acknowledged by references. Furthermore, this work has not been submitted for any other degree.

Signed: Poonam Dhoot

Student Number: 10399137

Date: 26th August 2019

ACKNOWLEDGEMENTS

I would like to acknowledge and thank the following people for their support and motivation during this project.

I would like to express my sincere gratitude and appreciation towards my supervisor Dr. Shahram Azizi Sazi, without whose guidance and mentoring I would not have been able to complete this project. His words of constant encouragement and meticulous reviews have helped me improve the work and its quality in many aspects. Short and succinct meetings with him has played a key role in this project and has helped me stay on top of the schedule.

I would also like to thank everyone in Dublin Business School, for their kind support. All my lecturers and academic staffs have been a tremendous help, in making me reach this far in the Master's thesis process. This journey would have not been possible if it wasn't for your support.

I would also like to thank my friends Ashin Basheer and Damanpreet Kahlon for their constant support and motivation without which the project would not have been completed successfully.

Finally, I would like to thank my family for their full-fledged support, without which pursuing this degree wouldn't have been possible.

Abstract

The aim of this thesis is to develop a predictive model to reveal the personality traits of a person based on their handwriting. The handwriting features taken into consideration are baseline, word spacing and pen pressure. These features indicate the social and emotional behavioural traits of a person. Using Computer Vision Library dataset, the handwriting samples are first manually analysed and labelled with traits. Then using image processing techniques feature vectors of handwriting is calculated. The feature vector is then mapped to the manually analysed and labelled traits. The feature vectors and trait are then fed to multiple machine learning models for training and test purpose. Various classification algorithms like Decision tree, Support Vector Machine, K-Nearest Neighbours, Naïve Bayes and Random Forest were compared to find the model with highest accuracy. On comparing all the models, Decision Tree was found the best suited for this prediction problem.

Chapter 1 – Introduction

As kids, we were taught to write in specific way at school, but apparently no one continues to write in the same manner as they were taught, every person picks a different style of writing and hence every handwriting looks different. In fact, as soon as someone can write, he or she gradually alters the shapes and sizes of letters in accordance with individual likes and dislikes. The reason is that our personalities affect the way our handwriting develops after we were taught to write. This is because handwriting is the pattern of our psychology expressed in symbols on the page and these symbols are as unique as our own DNA. When you get to know a person's handwriting well enough, you recognize whose script it is, just as if it were a well-known painting or photograph. Graphology is based on the principle that every individual's handwriting has a character of its own and this is entirely due to the uniqueness of the writer's personality.

So, it is the writer's deviations from the copybook learnt that allows expert graphologists to assess, with the greatest accuracy, the character and capabilities of the writer.

In fact, graphologists are exceptionally fortunate in that they see before them, in black and white, the pattern in symbolic form of a writer's *whole* psychological profile. By contrast, psychoanalysts and psychotherapists all over the world must formulate their own opinions solely based on what is *told* to them over a period by the client in question. Its purpose is to give a realistic view on problems that confront people from all walks of life, every day of their lives.

Handwriting is a universal skill that does not discriminate against sex, race, colour or creed.

Graphology gives an unbiased outline of the unique personality and behaviour of an individual, without them even being present.

"Just from analysing your handwriting, experts can find over 5,000 personality traits," graphologist Kathi McKnight says.

Graphology is a blend of art and science. It is a *science* because it measures the structure and movement of the written forms - *slants*, *angles* and *spacing* are accurately calculated and the pressure is observed in magnification and with precision. And it is an *art* because the graphologist has constantly to keep in mind the total context in which the writing is taking place: the 'gestalt' of the writing. Writing consists of three things - *movement*, *spacing* and *form*. A graphologist studies these variations as they occur in each of these aspects of writing and attaches psychological interpretations to them. Expert graphologists can achieve a very high degree of accuracy.

With the advancement of technology and use of technology in every walk of life to make human tasks easier and reduce the human intervention. Automation is used to perform repetitive, tiring and intensive jobs. Computer does not get tired and gives the same accuracy even when made to work for long hours. Using Machine Learning much more complex tasks can be accomplished by computer like humans. To leverage the machine learning capabilities this model is created to work instead of an expert.

History of Graphology

Handwriting Analysis or graphology is a very old concept. It was founded by Italians in 17th century. A place in Italy, Bologna, oldest university and where they teach graphology even today. France played a major role in laying the foundation of formal study of graphology in 19th century. Later it reached to other parts of the world. Today graphology is taught as a subject across the world in Europe, Israel, North and South America, India and China. ^[1]

Year	Notable Event
1622	Italian doctor Camillo Baldi writes the first known printed publication on the study of handwriting “ <i>How to recognise from a letter the nature and quality of a writer</i> ”.
Late 18th Century	Gainsborough reputedly keeps his model’s handwriting on the easel whilst painting portraits.
1875	French abbot Jean Hyppolyte Michon coins the term "graphology", from the Greek: "graph" meaning 'to write' or 'I write', and 'logos' meaning 'doctrine' or 'theory'.
1895	Wilhelm Preyer, child psychologist, says writing originates in the brain, not in the fingers and that handwriting is actually <i>brainwriting</i> .
1920	Henry Grunfeld, “There is not one case in 60 years where the graphologist has said something that turns out to be wrong”.
1930	Dr Ludwig Klages, widely regarded as the father of modern graphology, publishes the influential ‘Handwriting and Character’.
1949, 1950, 1953, 1954	Dr Eric Singer, an Austrian, living in England, publishes 'Graphology and Everyman'; 'The Graphologist's Alphabet'; 'Handwriting and Marriage'; 'Personality in Handwriting'.
1965	Francis T. Hilliger, (a student of Dr Eric Singer's) sets up his company ‘Handwriting Analysis Ltd’. His business encompasses personnel selection, tuition, graphotherapy (which proposed that negative emotions can result in disease) and work at London’s Old Bailey as an ‘expert witness’. He devises a method for assessing the degree of any trait or characteristic in a sample of handwriting, setting the (first) standards for student examination in the UK.

Table 1: Graphology Timeline Source: britishgraphology.org

Chapter 2 – Literature Review

Introduction

A literature review identifies, evaluates and synthesises the relevant literature within a particular field of research. It illuminates how knowledge has evolved within the field, highlighting what has already been done, what is generally accepted, what is emerging and what is the current state of thinking on the topic. In addition, within research-based texts such as a Doctoral thesis, a literature review identifies a research gap (i.e. unexplored or under-researched areas) and articulates how a particular research project addresses this gap.^[2]

To review the literature means to be able to identify:

- what has been established, discredited and accepted in the field of research
- areas of controversy or conflict among different schools of thought
- problems or issues that remain unsolved
- emerging trends and new approaches
- how your research extends, builds upon, and departs from previous research

The definition of literature review gives a better understanding about the task to perform.

“Handwriting Analysis based on Segmentation Method for Prediction of Human Personality using Support Vector Machine- 2010” by Shitala Prasad, Vivek Kumar Singh and Akshay Sapre.^[3] This paper conducts a classification process for identifying the personality of the writer based on individual handwriting features. For this purpose, the handwriting features taken into consideration are

- | | | | |
|-------|----------------------------|------|-------------------------|
| (i) | size of letters | (iv) | pen pressure |
| (ii) | slant of letters and words | (v) | spacing between letters |
| (iii) | baseline | (vi) | spacing between words |

For this purpose, 100 writers were asked to write a document of 70-80 words in running handwriting. The samples were taken on a blank paper without margins. The documents were then scanned to create a digital collection. The proposed methodology consists of three main steps namely, pre-processing, feature extraction, and classification.

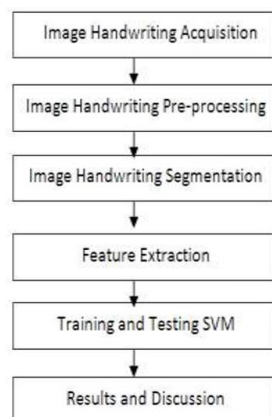


Figure 1: Steps for Handwriting Analysis

MATLAB is the tool used. Pre-processing comprises of opening the digital image and smoothing it. Then image segmentation which is dividing the entire image in lines, words and letters. Each of this segmentation extracts a feature to judge. The third step as mentioned is classification. For this purpose, the machine learning algorithm used is Support Vector Machine (SVM). SVM is much more time efficient and gives more accurate results compared to Neural Network as tested by the authors. There are six features and 16 different criteria based on handwriting styles. The document was fed as input to the model and their psychological behaviour based on individual feature were received as output. Trigonometry and thresholding techniques were used for calculations.

The performance of the proposed model is declared good as it takes very less time for training and testing. Two SVM compliant kernels were tested namely Polynomial and RBF. The accuracy ranges between 87-89% for Polynomial whereas RBF gives an accuracy range of 90-93%. Since, the RBF gives better accuracy that is taken into consideration.

	A	B
Polynomial kernel	87.9%	89.1%
RBF Kernel	90.3%	93.86%

Figure 2: Comparison between SVM models

“Artificial Neural Network for Human Behavior Prediction through Handwriting Analysis- 2010” by Champa H N and Dr. K R AnandaKumar.^[4] This paper proposed a method to predict the personality of a person based on their handwriting. For this study various features found in an individual’s handwriting are considered like the baseline, pen pressure and letter ‘t’. Using Artificial Neural Network (ANN) the personality traits of the writer are revealed. The tool used for this system design is MATLAB.

Three parameters the baseline, pen-pressure and the height ‘t’-bar on the stem of letter ‘t’ are fed as input to the ANN model and in return the model reveals the personality traits of the writer. Various techniques are used for evaluation of the features. Baseline is evaluated using the method of polygonalization. Pen-pressure is evaluated using the thresholding technique of grey-level value. Height of ‘t’ bar on the stem of letter ‘t’ is evaluated by template matching technique. The authors here have considered three types baselines – ascending, descending and level; two types of pen-pressure – heavy/deep, light; five templates were created to be matched for the height of ‘t’ bar on the stem of lowercase letter ‘t’, few mentioned – very high ‘t’ bar but on the stem, very low ‘t’-bar, just above the middle zone of ‘t’ bar, crossed above the stem. Hamming distance helps to make the final decision for the matched pattern.

30 different types of output indicate 30 different types of personality traits of the writer based on the combination of features discussed earlier. Since huge amount of data is involved in the process ANN was chosen by the authors. The authors experimented with epochs and number of nodes in hidden layers to identify the performance variation. Nodes in the hidden layer ranging from 2-12 with step of 2 and epochs from 500 -6000 with step of 500. The performance goal was achieved at number of hidden layer node – 8 and epochs 4500 in shortest duration of time 0.099.

Epochs	Hidden layer nodes					
	02	04	06	08	10	12
Performance						
500	0.53	0.42	0.34	0.35	0.36	0.27
1000	0.36	0.35	0.29	0.31	0.48	0.22
1500	0.32	0.34	0.28	0.29	0.41	0.51
2000	0.22	0.22	0.25	0.25	0.53	0.31
2500	0.21	0.2	0.24	0.16	0.23	0.31
3000	0.16	0.19	0.22	0.12	0.2	0.25
3500	0.12	0.15	0.13	0.15	0.18	0.19
4000	0.16	0.14	0.15	0.13	0.17	0.14
4500	0.15	0.14	0.17	0.099	0.11	0.16
5000	0.13	0.11	0.099	0.099	0.099	0.11
5500	0.11	0.099	0.099	0.099	0.099	0.099
6000	0.099	0.099	0.099	0.099	0.099	0.099

Figure 3: Overall Performance of ANN

The paper very well works to find an individuals' trait based on one feature at a time. It shows which combination of number of nodes in hidden layer and epochs works the best to give the result in least time.

“Handwriting Analysis for Detection of Personality Traits using Machine Learning Approach – 2015” by Prachi Joshi, Aayush Agarwal, Ajinkya Dhavale and 2 others.^[5] This paper proposed a novel approach of machine learning technique to implement the automated handwriting analysis tool. The handwriting features analysed for identifying the personality traits are the baseline, width of margin, slant of letters and height of 't' bar. The various techniques used are –

- (i) Polygonalization – It is a method to divide the plane into multiple polygons, it is to create a closed polygon around a single line of scanned handwriting sample. Using the coordinates of the polygon the slope of the polygon is calculated. This slope corresponds to the slope of the baseline.

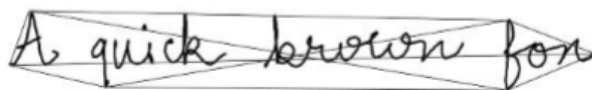


Figure 4: Polygonalization

- (ii) Thresholding Algorithm – It is a simple and effective way of image segmentation. It transforms grayscale images to binary images. This will help to find the width of margin.
- (iii) Template Matching – It is a technique of image processing used to find small parts of an image which match the pre-defined templates. Individual lines, words and characters are isolated and then matched with the template images by correlation. This technique is used for finding letter 't' and slant of the letters.

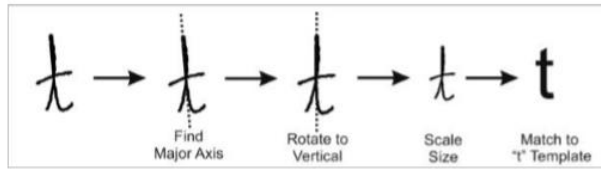


Figure 5: Template Matching

This paper has considered handwriting samples of individuals between the age of 20 and 35 years. Training set was generated using 100 samples which were given to a Graphologist to examine. A feature vector matrix (FVM) of the sample was created. Based on the personality traits given by the expert classes were defined. The FVM was created for every new sample of handwriting. Using KNN classifier an appropriate class was assigned to the new sample based on similarity matrix. Then the sample will be included to the class it is mapped to, to implement the incremental machine learning and increase the accuracy of future results.

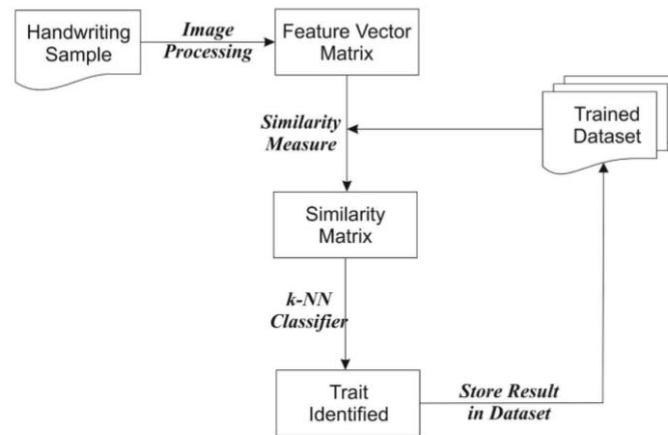


Figure 6: Steps for KNN classification

This paper discusses various image processing techniques like polygonalization, thresholding, template matching to find the feature vectors.

“Development Of An Automated Handwriting Analysis System – 2011” – by Vikram Kamath, Nikhil Ramaswamy, P. Navin Karanth, Vijay Desai and S. M. Kulkarni.^[6] The authors have developed a system for behavioral prediction of a person through an automated system. Automated Handwriting Analysis System (AHWAS) is built using MATLAB tool. Eight features of handwriting are accessed to identify the personality traits of the writer. The features taken into consideration are namely size of letters, slant, pen-pressure, baseline, word spacing, margin, speed of writing and number of breaks. The handwriting is analysed through Image Processing in MATLAB. The methodology consists of three steps primarily: image pre-processing, feature extraction and prediction.

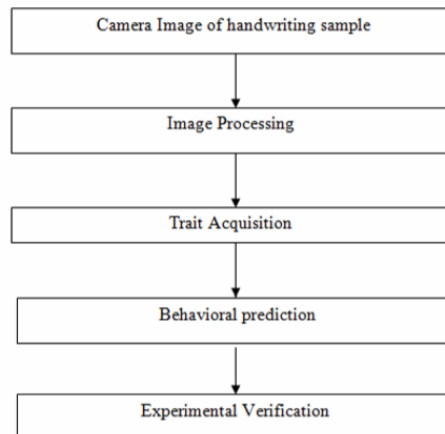


Figure 7: Process for Handwriting Analysis

The study was conducted using 30 samples handwritten text from people of both genders and different ages between 20 and 24. The group had all right-handed writers, physically and mentally sound. Each of them was asked to write a text of approximately 100 words which covered almost all English alphabets. The handwriting samples were first manually analysed by handwriting analysts. The inference provided by analyst were mapped to the samples which then used for calibration and experimental verification. AHWAS was more than 80% successful to identify the features compared with manual analysis.

Handwriting trait	Features determined by AHWAS	Reference	Trait category
Size	2.356 mm	1/16 " - 3/16"	medium
Baseline	4.67°	> 0°	upward
Pressure	0.0225 mm	< 0.025 mm	light
Slant	116°	112° -125°	BC
Breaks	34	>5	disconnected
Word spacing	0.223 mm	> twice width of word	wide
Speed	0.3089	> 20%	fast

Figure 8: Results of AHWAS and manual analysis

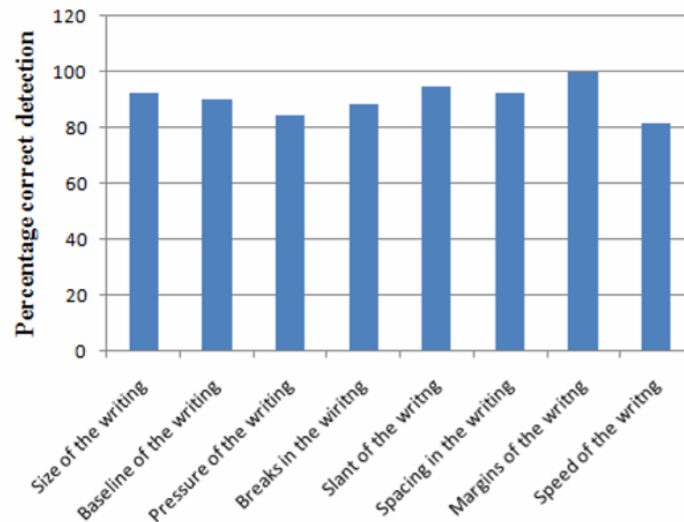


Figure 9: Percentage correct detection of the characteristic traits

“An Improved Method for Handwritten Document Analysis using Segmentation, Baseline Recognition and Writing Pressure Detection – 2016” ^[7] by Abhishek Bal and Rajib Saha. This research proposed an off-line handwritten document analysis through segmentation, skew recognition and writing pressure detection for cursive handwritten document. The segmentation method proposed is based on modified horizontal and vertical projection which helps to segment handwritten document text lines and words even if it is overlapped or skewed. 550 text images from IAM dataset was used to collect colour and grey scale handwritten document. MATLAB is used for implementing the work. The steps included for the process were as follows:

- (i) Image pre-processing – Otsu thresholding binarization technique and median filter noise removal technique.
- (ii) Line Segmentation – It is calculated after creating a horizontal projection histogram of binary document. A threshold is calculated based on the average height of the rising section. Then every rising section is compared with the threshold to get if the line segment is actual binary document or can be neglected. False rising shows the presence of overlapping between two lines or bar in an upper letter.
- (iii) Word Segmentation – It is calculated after creating a vertical projection histogram to measure the distance between two words in a line. A threshold value is used to identify if the distance is between two words or between two characters of a word.

Using the proposed method, the author could successfully identify 95.65% lines and 92.56% word are correctly segmented from the IAM dataset; also normalizes 96% lines and words perfectly with very small error rate.

The paper demonstrates a good way to identify the baseline, lines and words from the scanned image of handwritten documents.

“HABIT: Handwritten Analysis Based Individualistic Traits Prediction – 2013” – by Abdul Rahiman M, Diana Varghese & Manoj Kumar G.^[8] This paper implements an off-line, writer independent handwriting analysis system “HABIT” – Handwriting Analysis Based Individualistic Traits Prediction. This tool predicts the personality of the writer automatically based on the features extracted from the scanned image of writer’s handwriting sample. The handwriting features taken into consideration are pen pressure, baseline, slant of letters and size of writing. To implement the model programming language chosen is Java and Integrated Development environment used is Eclipse – Indigo. The overview of the HABIT system is depicted in figure 1.

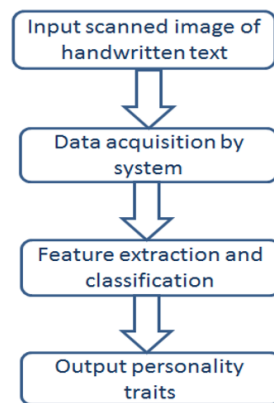


Figure 10: HABIT System- An overview

Figure 2 shows the steps taken for image acquisition and feature extraction.

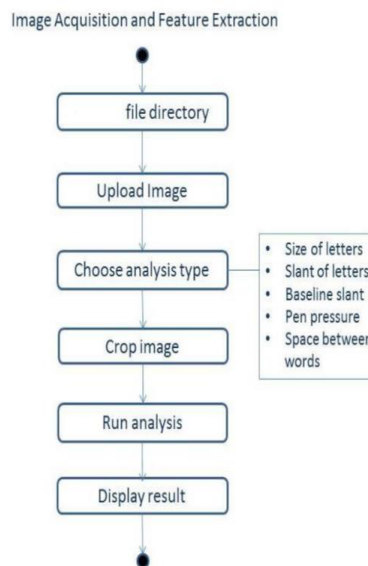


Figure 11: Image Acquisition & Processing

Feature	Formula / Methodology
Baseline / Slant	Starting point (x ₁ ,y ₁) and End point (x ₂ ,y ₂) $\theta = \tan^{-1} \frac{(y_2 - y_1)}{(x_2 - x_1)}$

Pen pressure	Compare the mean of grey scale pixel value with threshold
Spacing between words	The Pythagoras equation to compute the length of the hypotenuse is: $\sqrt{(\Delta x^2 + \Delta y^2)} = c^2$
Size of letters	Pythagoras's Theorem to find the distance between the top point and baseline: $\sqrt{(\Delta x^2 + \Delta y^2)} = c^2$ where Δx is the distance between x co-ordinates and Δy is the distance between y co-ordinates

Table 1: Features and Formula

The method used to implement this is simple linear regression which is an approach to model the relationship between a scalar dependent variable y and an explanatory variable denoted x.

“Personality Analysis Through Handwriting – 2012” by Rashi Kacker and Hima Bindu Maringanti.^[9] In this paper, the author has mapped the handwriting based behavioural traits with existing personality theories and a final output is personality type, temperament along with a detailed report. Based on Myer Briggs type indicator for personalities based on four dichotomies are as follows:

- (i) Extraversion (E) - (I) Introversion
- (ii) Sensing (S) - (N) Intuition
- (iii) Thinking (T) - (F) Feeling
- (iv) Judgment (J) - (P) Perception

These four acts as a basis of Keirsey's Temperament Sorter, based on all the permutation and combination sixteen personality types are generated. All the 16 are shown in Figure 12.

ISTJ (Inspector)	ISFJ (Protector)	INFJ (Counselor)	INTJ (Mastermind)
ISTP (Crafter)	ISFP (Composer)	INFP (Healer)	INTP (Architect)
ESTP (Promoter)	ESFP (Performer)	ENFP (Champion)	ENTP (Inventor)
ESTJ (Supervisor)	ESFJ (Provider)	ENFJ (Teacher)	ENTJ (Field Marshal)

Figure 12: Keirsey's Temperament sorter

The dataset used is a collection of 50 handwriting samples written on A4 size plain white sheet with black ink. The features extracted in this work are margins, size, slant, degree of connection between the letters, spacing between lines and ratio of the three zones in handwriting namely the upper, middle and lower zone.

These features are then mapped to the dichotomies, arrived at by us, in consultation with the domain expert. The mapping is as shown in Figure 13.

Dichotomies	Features
(E) - (I)	Size, Slant
(S) - (N)	Margins, Zonal ratio
(T) - (F)	Space, Slant
(J) - (P)	Degree of connection, Space

Figure 13: Mapping of traits

“Computer Aided Graphology – 1995” by G. Sheikholeslami, S. N. Srihari, V. Govindaraju.^[10] This paper is a study of a person’s character and personality based on their handwriting. The steps taken in order to reveal the personality of the writer is shown in the figure.

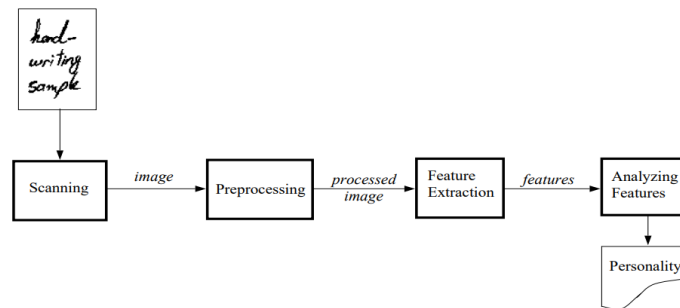


Figure 14: Structure of Computer Aided Graphology system

The features extracted and analysed are page left margin, page right margin, page bottom margin, line spacing, line direction, slant, and upper, middle, and lower zone ratios. This system was mainly designed to prove the validity of the graphology rules that were applied in the implementation of the system. This paper restricts its scope to macro analysis of the handwriting sample. There are no micro features like alphabet, loops etc. taken into consideration.

Conclusion:

Literature Review is conducted to gain knowledge about the study and depth in which the topic is explored and implemented. The research shows that most authors have identified handwriting features using machine learning algorithms like Support Vector Machine, K – Nearest Neighbours, Artificial Neural Network, Linear Regression. Each feature is identified individually, and accuracy is calculated. The most common features used for human behavioural trait identification are baseline, slant, pen-pressure, height of lowercase ‘t’ bar, word spacing. Image processing technique like polygonalization, thresholding and pattern matching are largely used to achieve the results. It is also established that the Neural Network takes very long to process hence can be skipped.

As seen, most of the paper classifies the feature based on the handwriting feature. Many classifications models are created and have decent accuracy. The current research aim is to add more value to the existing world of Handwriting analysis using computer. Using computer in a way that it works as good as graphologists or even better and more accurate. Hence, the aim is to create a predictive model for handwriting analysis.

Chapter 3 - Methodology

Methodology is a structured set of methods, practices, processes and procedures used to attain. The methodology is the general research strategy that outlines the way in which research is to be undertaken and, among other things, identifies the methods to be used in it. These methods, described in the methodology, define the means or modes of data collection or, sometimes, how a specific result is to be calculated. Methodology does not define specific methods, even though much attention is given to the nature and kinds of processes to be followed in a procedure or to attain an objective.^[11]

There are various methodologies available for data mining processes like KDD, CRISP-DM, SEMMA.

KDD - Knowledge Discovery in Database; SEMMA – Sample, Explore, Modify, Model, and Assess ^[12];
CRISP-DM - Cross-Industry Process for Data Mining

CRISP-DM is a great starting framework to understand the general data science process. It likewise may serve individual and small teams well, and if augmented with other project management approaches, might suite larger teams. Specifically, emerging approaches that combine agile project management and CRISP-DM are likely more effective.

Compared to CRISP-DM, SEMMA is even more narrowly focused on the technical steps of data mining. It skips over the initial Business Understanding phase from CRISP-DM and instead starts with data sampling processes. SEMMA likewise does not cover the final Deployment aspects. Otherwise, its phases somewhat mirror the middle four phases of CRISP-DM. Although potentially useful as a process to follow data mining steps, SEMMA should not be viewed as a comprehensive project management approach.

KDDs can be a useful expansion of CRISP-DM for big data teams. However, KDDs only addresses some of the shortcomings of CRISP-DM, and its combination of phases and processes is less straight-forward.

The methodology used for the current study is CRISP-DM. It is a robust and well-proven methodology. It is widely used because of its powerful practicality, its flexibility and its usefulness when using analytics to solve thorny business issues. The model is an idealised sequence of events. The life cycle model consists of six phases with flow indicating the most important and frequent dependencies between phases. The sequence of the phases is not strict. In fact, most projects move back and forth between phases as necessary.



Figure 15: CRISP-DM Process

Stage One: Determine Business Understanding

The first stage of the CRISP-DM process is to identify what do you want to accomplish from business perspective. The goal of this stage is to understand the business requirements and make sure that the outcome of the project is in-line with the important factors to be uncovered. Without a clear understanding of the requirement the task will be totally directionless and end up with right answers to wrong questions.

The objective of this research paper is to predict personality traits of a person based on various handwriting features using machine learning algorithm.

To achieve the objective the following machine learning algorithms are compared to find the one which works the best, that is, gives highest accuracy:

- (i) Decision Tree
- (ii) Naïve Bayes
- (iii) Support Vector Machine
- (iv) K – Nearest Neighbours
- (v) Random Forest

Based on the literature review, it is found that so far, all the research carried out classifies the features, but no study predicts the behaviour based on the handwriting. The overall personality of the writer is never revealed. In this paper, the focus is to predict the overall personality of the writer based on the handwriting sample.

The various industries/fields where this study can be useful are as follows ^[13]:

- (i) **Recruitment** where it is an invaluable aid because an experienced graphologist can pick out the best candidates and advise over suitability.
- (ii) **Management selection** in commerce and industry where it is employed in conjunction with psychometric testing.
- (iii) **Corporate training** where it can highlight staff strengths and flag up weaknesses, potential and motivation.
- (iv) **Security checking** and the evaluation of honesty and integrity.
- (v) **Career guidance** for those seeking employment or a change of direction.
- (vi) **Compatibility assessments** for business and personal relationships.
- (vii) **Personality profiling** for individuals seeking self-awareness for self-development.
- (viii) **Child and family guidance** to help resolve sensitive issues.
- (ix) **Historical profiling** for genealogists and biographers who want to learn more about people who have died.
- (x) **Document examination and forensic analysis** for assessing forgeries and poison pen issues.

Stage Two: Data Understanding

The second stage of CRISP-DM process is to identify data required for the project. Once the data is identified, it needs to be explored and understood better.

Graphology is a technique to interpret the handwriting of a person and reveal his personality traits. The analysis is highly dependent on the skills of the handwriting analysis expert called Graphologist or Analyst. Before diving deep into the topic, it was necessary to read and gather information about Graphology. Few questions to know the topic better are as follows:

- (i) What are handwriting features?
- (ii) What each feature reveals about the personality?
- (iii) Which feature detection can be replicated by the computer?
- (iv) Where to find data to perform this analysis?
- (v) What all analysis can be performed on the dataset acquired?

There are numerous features of handwriting which can help to reveal the personality of the writer. Each feature individually reveals a certain type of behavioural trait of a person. The various handwriting features are pen pressure, baseline, spacing between words, slant, margin, size of letters, height of 't' bar, loops of letters 'l', 'e', 'y' and 'o', hump of letters 'm' and 'n'. For this study, only three important features are used for analysis namely, pen-pressure, baseline and word spacing.

Each of these three features are explained below along with the trait analysis:

Pen-pressure: It is the weight of the handwriting. This feature signifies the amount of energy available in the person.^[14] This energy is not just physical energy, but also mental energy. Based on the pressure the writer can fall into either of the two categories: Heavy or Light writer. Table 2 has an example for each type of pen-pressure along with the trait analysis. ^[15]

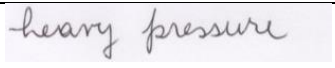

Pen Pressure		Trait Analysis
	Heavy	It indicates that the writer has got a great deal of energy. The energy mentioned here is not only physical energy, but it may be mental energy too. This type of person may keep fighting to the end until he gets what he wants, he may also complete his tasks despite the difficulties that he faces, and he may like physical activities.
	Light	It indicates that the writer has got less energy than a writer writing with heavy pressure. This person may prefer activities that require less physical work and he may also be a little sensitive. This person may also be more flexible, and this may make him more adaptable to changing situations.

Table 2: Pen-Pressure Trait Analysis

Baseline: The imaginary line on which the writer writes on the blank paper is called ‘baseline’. It determines the emotional control and reliability of the writer. The three most fundamental baselines are straight, uphill and downhill. Table 3 has an example for each type of baseline along with the trait analysis.

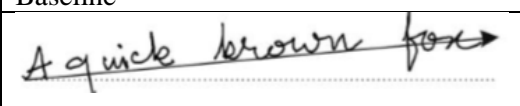
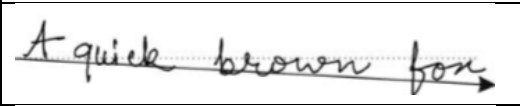
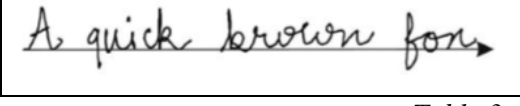
Baseline		Analysis
	Ascending	Optimistic, upbeat, positive attitude. Ambitious, hopeful, emphasizes the positive view, can-do attitude.
	Descending	Tired or overwhelmed, pessimistic. Not feeling hopeful, sees more chances for failure than success.
	Level	Determined, stays on track. Self-motivated, reliable, mind controls emotion, steady, unflinching.

Table 3: Baseline Trait Analysis

Word Spacing: Spacing refers to how far one word is placed from the other. It shows how much distance one wants to keep from people around them. The spacing between the words shows how much room the author needs to feel comfortable, his “social bubble”. This handwriting feature in a sample is obtained by the number of white pixels between the end of the one word and the start of the next word. Table 4 has an example for each type of word spacing along with trait analysis.^{[16][17]}

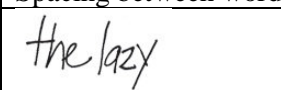
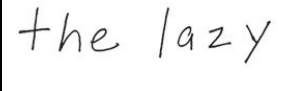
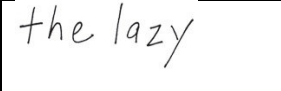
Spacing between words		Analysis
	Narrow word Spacing	This Person craves constant contact with somebody. Craves emotional closeness. Needs someone to speak. His/ her dependency on others is more. The writer can be selfish. Not willing to offer own time and energy to others but expecting same from them! If spacing gets the exceptionally narrow, then a writer is unable to allow space to his dear ones. Might be because of an excessive amount of love and affection from parents. Like being touched.
	Wide word Spacing	This Person prefers space. Even if the person is social, outgoing and jells up with people easily; he/she needs his own time alone. Privacy is a vital aspect for him. Reason can be difficulty in communicating with others or need for isolation for a while or can be afraid of getting emotionally hurt. Independent personality. Dislike being touched. The writer is extremely cautious. The person always thinks before speaking.
	Even word Spacing	Shows Person feels secure around others. One can experience writer’s balance of thoughts. It additionally reveals stability, and reliability about relationships.

Table 4: Word Spacing Trait Analysis

Stage Three: Data Preparation

This is the dataset selection stage for the analysis purpose. While selecting the dataset many factors need to be taken into consideration like the relevance of data, quality of data and technical constraints.

The dataset used for this study is CVL database – Computer Vision Lab database, which is a public database for writer retrieval, writer identification and word spotting. The database consists of 7 different handwritten texts (1 German and 6 English Texts). In total 310 writers participated in the dataset. 27 of which wrote 7 texts and 283 writers had to write 5 texts. For each text an RGB colour image (300 dpi) comprising the handwritten text and the printed text sample is available as well as a cropped version (only handwritten).^[18] The below image is an example of handwritten text in the database. The printed text is between two horizontal separators and below that is the handwritten content. Each image has a unique writer id and text number (on top-right).

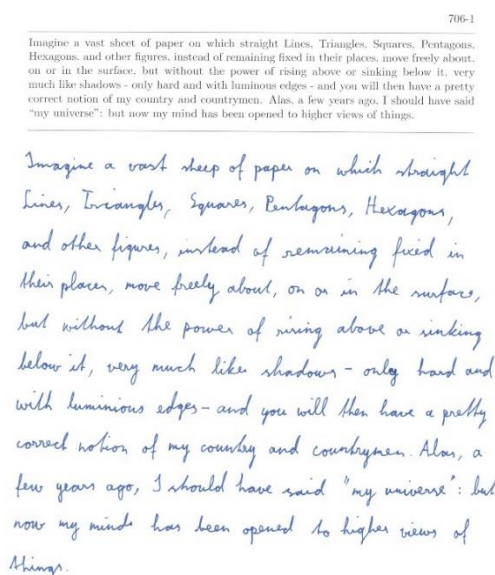


Figure 16: Handwriting Sample from CVL database – page



Figure 17: Handwriting Sample from CVL database – line

The dataset already consists of document text segmented into lines. As mentioned, there are 310 writers, and in all 1604 handwriting samples including both English and German texts. These 1604 samples are further segmented into lines of text which brings it close to 12,000 samples.

Data cleaning process included the following steps:

- (i) Deletion of images with no text
- (ii) Rotating the documents which were scanned tilted
- (iii) Converting the images into grey scale
- (iv) Removing the noise
- (v) Cropping the image based on horizontal separator

After the dataset was cleaned, the images were manually analysed line-by-line to identify the personality traits based on the handwriting features like baseline, pen-pressure and word-spacing. During the manual labelling it was found that most writers had baseline upward or downward hence the baseline considered in the process is only upward and downward. Similarly, it was found that the word spacing was mostly wide or narrow hence only those two are considered in the process. In case of pen-pressure, two categories are considered namely, heavy and light pressure.

Table 6 shows the mapping between Features and label.

Feature	Feature Detail	Label
Baseline	Upward	U
	Downward	D
Word spacing	Wide	W
	Narrow	N
Pen Pressure	Heavy	H
	Light	L

Table 5: Feature – Label Mapping

Based on the handwriting sample the combination of personality revealed are as follows:

Baseline – Word Spacing – Pen Pressure	Label	Personality
Downward – Narrow – Heavy	DNH	Pessimist – Insecure – Intense
Downward – Narrow – Light	DNL	Pessimist – Insecure – Gentle
Downward – Wide – Heavy	DWH	Pessimist – Independent – Intense
Downward – Wide – Light	DWL	Pessimist – Independent – Gentle
Upward – Narrow – Heavy	UNH	Optimist – Insecure – Intense
Upward – Narrow – Light	UNL	Optimist – Insecure – Gentle
Upward – Wide – Heavy	UWH	Optimist – Independent – Intense
Upward – Wide – Light	UWL	Optimist – Independent – Gentle

Table 6: Label – Personality Mapping

The dataset after this exercise is as Figure 18 shows

id	final_trait
0184-6-6.tif	DWL
0184-4-4.tif	UWL
0184-2-2.tif	DWL
0184-3-12.tif	UWL
0184-3-13.tif	DWL
0184-2-3.tif	SWL
0184-4-5.tif	UWL

Figure 18: Dataset snapshot

Once the personality traits were identified based on handwriting, now it was time to make the computer learn from the sample of handwriting. But this information is just not enough. So, with the help of Python programming language using OpenCV library the images were processed.

The steps taken to process the images and find the features like baseline and word spacing based on the handwriting are firstly image segmentation, this process is converting the image into grey scale to separate the foreground and the background. The black/grey pixels are foreground and white pixels are treated as background. After the image segmentation perform the Canny edge detector to detect edges in the image, then perform a dilation and erosion to close gaps in between object edges. Once the closed edged object is found then an OpenCV function – findContours() is called to detect the outlines of the object in the edge map.

Baseline, as defined earlier, is the imaginary line on a blank paper which touches the base of each word. After finding the contours, the focus was on the lowermost part of each contour, as that is of prime importance to identify the inclination of baseline. The average value of all the contours together are taken into consideration. Using those values a line is drew to fit amongst all the points. The line which fits most of the points represents the baseline for that statement. Now that the baseline is found, the next thing to do is find the inclination of it. For this, the starting points of the line (left-most point) is taken as (x_1, y_1) and the end point of the line (right-most point) is taken as (x_2, y_2) and the proportional slope of the line calculated using the formula – $(y_2 - y_1)$.

Since, the proportional value of the slope is enough to identify the inclination of baseline $(y_2 - y_1)$ is works well.

The values calculated consists of both negative and positive values, which makes sense; the negative slope values signify descending slope. Descending slope means the starting point is higher than the end point. The positive slope values signify ascending slope. Ascending slope means the starting point is lower than the end point. Based on these characteristics the behavioural trait of the person is said to be either optimist or pessimist.

Word Spacing, as defined earlier, is the distance between the end of a word and the starting of next word. The number of white pixels is the distance between words called word spacing. The space could either be wide or narrow. The initial step remains the same as baseline, creation of edge map followed by contours. After contour generation, bounding boxes are created around each contour to show each contour as a word. Bounding boxes are the quadrilateral drawn around each contour using the top-left, top-right, bottom-right, bottom-left points of the contour. Once the boxes are created, the average distance between the boxes are calculated using sum of individual distances divided by number of words(contours) found in the image.

$$\text{Average word spacing} = | \text{Sum of distances between words} \div \text{Number of words} |$$

The value calculated is positive, because distance cannot be negative. Lower the value of word spacing, closer the words are to each other. Similarly, higher the values word spacing, farther the words are from each other. Closer words or narrow word spacing indicate that the person is insecure whereas farther words or wide word spacing indicates independent personality.

Pen-Pressure, as defined earlier, is the amount of energy a person puts in to write. For identifying the pen-pressure the Otsu binarization technique, also called as Otsu Thresholding, is used.

Otsu's method, named after its inventor Nobuyuki Otsu, is one of many binarization algorithms. Otsu's thresholding method involves iterating through all the possible threshold values and calculating a measure of spread for the pixel levels each side of the threshold, i.e. the pixels that either fall in foreground or background. The aim is to find the threshold value where the sum of foreground and background spreads is at its minimum.^[19]

The pen-pressure value obtained from Otsu's thresholding method is a value between 0 and 255. The value is in that range because the grey scale values ranges from 0 to 255, 0 for black pixels and 255 for white. This implies that higher the value of pen-pressure, closer the value to white pixel, which means lighter the pen-pressure and lesser the value of pen-pressure, farther the value from white pixels, which means heavier the pen-pressure. Based on the pressure applied while writing, the energy levels of the person is determined. The energy could be either physical or mental energy. Heavy pressure indicates intense personality or high energy whereas light pressure indicates gentle personality or low energy.

The code written in Python using OpenCV calculates the numerical value for each feature. Running through all the handwriting samples the output obtained is as shown in Figure 10.

id	baseline_val	wordspacing_val	penpressure_val
0184-6-6.tif	-9	165.222	222
0184-4-4.tif	16	52.889	221
0184-2-2.tif	-11	65.067	220
0184-3-12.tif	15	108.286	222
0184-3-13.tif	-24	127.125	220
0184-2-3.tif	0	78.417	220
0184-4-5.tif	45	119	221

Figure 19: Feature Vector Dataset

From the values obtained it was hard to identify a pattern as the value range is so large and uneven. It was then realized that the values are so vague because, a single line of text is considered for each handwriting sample, the height and width of each image varies as the number of words and size varies based on text and writers. To keep a standard value which can later be compared to all other values and find the personality trait, normalization technique is used.

For Baseline, to normalize the value the proportional slope value is divided by the height of the image.

$$\text{Normalized Baseline} = (y_2 - y_1) \div \text{Height of the image}$$

For word spacing, to normalize the value, the average value calculated is divided by the width of the image.

$$\text{Normalized Word Spacing} = \text{Average distance between words} \div \text{Width of the image}$$

For pen pressure, to normalize the value, the Otsu's thresholding method output is divided by 255 to get the value between 0 and 1.

$$\text{Normalized Pen Pressure} = \text{Pen pressure} \div 255$$

After the normalized values are calculated, the combined output is mapped to the traits identified when analysed manually. At this stage, the dataset looks like figure 20

id	bl_norm	ws_norm	pp_norm	op
0184-6-6.tif	-0.067	0.8	0.870588	DWL
0184-4-4.tif	0.117	0.258	0.866667	UWL
0184-2-2.tif	-0.096	0.555	0.862745	DWL
0184-3-12.tif	0.11	0.387	0.870588	UWL
0184-3-13.tif	-0.253	0.633	0.862745	DWL
0184-2-3.tif	0	0.515	0.862745	DWL
0184-4-5.tif	0.304	0.4	0.866667	UWL

Figure 20: Dataset after Normalization

This dataset is now ready. The dataset can be fed to the machine learning models.

Chapter 4 - Implementation & Modelling

The overall goal of this project is to ease the process of handwriting analysis, to automate the process of identifying the personality traits from based on the handwriting features. The features chosen here are baseline, word spacing and pen pressure. A Graphologist is an expert of handwriting analysis. The tool intended to be developed is a complementary tool to improve the accuracy and make the process fast. To make the process less tedious and efficient, computer needs to work like the graphologist.

The computer needs to work like human, learn from experiences and be able to make decisions based on knowledge gathered. Machine Learning does the task of not just performing coded steps, but it also learns along the way which makes the results more human like. Humans learn from experiences; the machine learns from data. The more amount of data is fed to the model the better results are retrieved.

Machine learning can be broadly divided into two categories – Supervised and Unsupervised.

Supervised Learning:

Supervised learning is when the model learns based on a labelled dataset. Labelled dataset is the dataset which has the desired outcomes defined. The outcome is a finite set of values. For each input data (x) there is a corresponding output variable (Y).

$$Y = f(x)$$

The goal is to develop a mapping function which approximates the output value for a new input data. For this process, the model is trained using a dataset which consists of labelled output. This dataset acts as a teacher for the model. This dataset helps the model to learn, it is counted as experience for the model. The model runs iteratively through the dataset to learn more about the data and give better results. The algorithm stops learning when an acceptable level of accuracy is achieved.

Supervised Learning can be of further grouped into Regression and Classification.

Regression: This type of supervised learning model can be implemented when the outcome is a real value. The output variable is not a finite set of values but is a real number. Examples of Regression model is when the output is like age, weight, amount in dollars, etc.

Classification: This type of supervised learning model can be implemented when the outcome is a category. The output variable is a finite set of values. Examples of Classification model is when the output is like True or False, male or female, etc. There can be more than two classes as well.

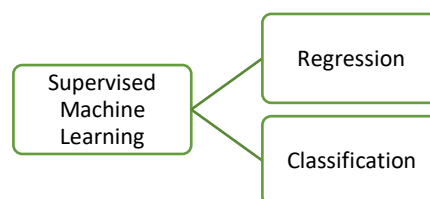


Figure 21: Supervised Machine Learning

Unsupervised Learning:

Unsupervised learning is when the model learns on its own. There is no labelled dataset to check for the correctness of the output. For each input data (X) there is no corresponding output variable. The model structures and distributes the data to find patterns and learn more about the dataset. There is no one correct output and there is no teacher. The algorithms are left on their own to discover and reveal an interesting fact about the data.

Unsupervised Learning can be further grouped into Clustering and Association.

Clustering: This type of unsupervised learning model can be implemented when the aim is discovery of inherent groups in the dataset. Example of the same is grouping the customer based on their purchasing activities.

Association: This type of unsupervised learning model can be implemented when the aim is discovery of rules that describe large portion of the given data. Example of the same is, people who buy product A tends to buy the product B as well, say milk and bread.

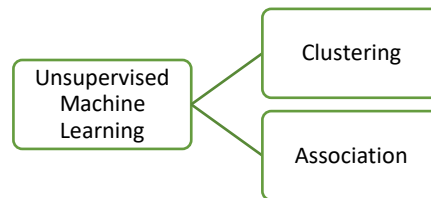


Figure 22: Unsupervised Machine Learning

Now that the types of machine learning are known. Let's figure out which works the best for the current study.

The aim here is to predict the behavioural traits of a person based on their handwriting. The behavioural traits are defined based on their characteristic features. Features are baseline, word spacing and pen pressure. The dataset consists of the feature vector of all the mentioned features. Based on the manual analysis the combined feature vectors are mapped to final trait. These traits are the output labels for the input data, handwriting sample. So, for each handwriting sample there is a corresponding output variable. This explains that current problem at hand can be solved using a Supervised learning model. It is a Predictive modelling problem.

The outcome is a finite set of values and it is categorical in nature. Classification machine learning algorithms will be best suited for this.

To perform this task some of the well-known prediction algorithms are compared.

The machine learning algorithms chosen for comparison are:

- (i) Decision Tree
- (ii) Naïve Bayes
- (iii) Support Vector Machine (SVM)
- (iv) K -Nearest Neighbours (KNN)
- (v) Random Forest

Decision Tree:

Decision Tree is a supervised machine learning algorithm. It can be used for both classification and regression. Exactly like the name it uses a tree-like model to make decisions. Parts of this tree are root, branches and leaves. It is like an upside-down tree with root at the top. Based on the condition or internal node the node is divided into branches or edges. The last layer when the nodes cannot be split any further is called the leaf nodes or the final decision. The factors which are to be considered while modelling the decision tree are the cost of the split, a Gini score gives an idea of how good a split is by how mixed the response classes are in the groups created by the split. One problem with decision tree is that it overfits. To overcome this problem the tree can be stopped after a certain point which will prevent it from creating a complex, large and overfitting model. The splitting can be stopped by using one of the following ways: set a minimum number of training inputs to use on each leaf or set maximum depth for the model. Maximum depth refers to the longest path from root to leaf. To make the decision tree perform better pruning can be used. It involves removing the branches that make use of features having low importance. This way, it reduces the complexity of model, and thus increasing its predictive power by reducing overfitting.^[20]

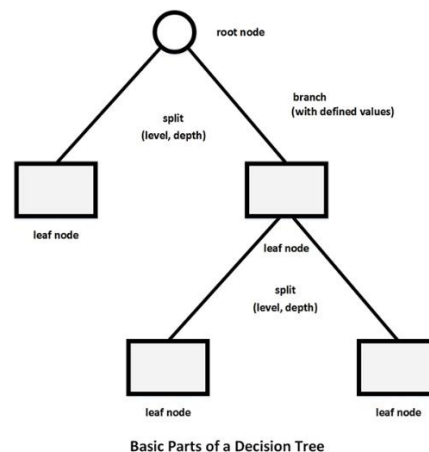


Figure 23: Decision Tree

Naïve Bayes:

The Naïve Bayes machine learning algorithm is based on the Bayes rule, also called as Bayes Theorem. The name is given after the creator of the theorem, Thomas Bayes (1702–61). The Naïve Bayes is a probabilistic machine learning model that is widely used for classification. The algorithm is called “Naïve” because it assumes that all the features that goes into the model are independent of each other. This means that the change in the value of one variable does not influence or change the value of other variables used in the algorithm. This is a probabilistic algorithm, hence has some significant advantages like quickly coded, fast prediction, can be used for real-time analysis because it is very quick. It is highly scalable hence choice of major real-world applications which expect instant response.^[21]

The formula of Naïve Bayes is as follows:

$$P(B|A) = P(A|B) * P(B) / P(A)$$

Support Vector Machine (SVM)

Support Vector Machine (SVM) is a supervised machine learning algorithm. It can be used for both classification as well as regression but largely used for classification. In this algorithm each data point is considered as a point in n-dimensional space with the value of feature being the coordinate of the point. Then the classification is performed by finding the hyperplane that differentiate the classes. Support Vectors are simply the co-ordinates of individual observation. Support Vector Machine is a frontier which best segregates the two classes (hyper-plane/ line).

SVM works on kernel trick. These are functions which takes low dimensional input space and transform it to a higher dimensional space i.e. it converts not separable problem to separable problem, these functions are called kernels. It is mostly useful in non-linear separation problem.^[22]

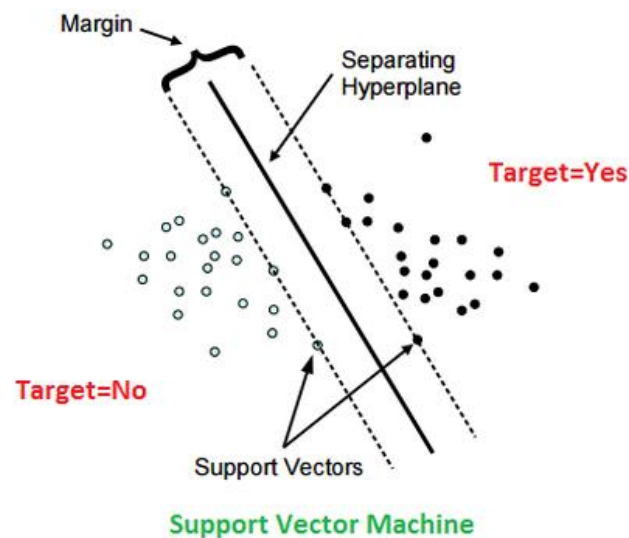


Figure 24: Support Vector Machine

In this case, the model is experimented with multiple kernel values to find the best model. The following kernels are used:

- (i) Polynomial Kernel
- (ii) Radial Basis Function (RBF) Kernel

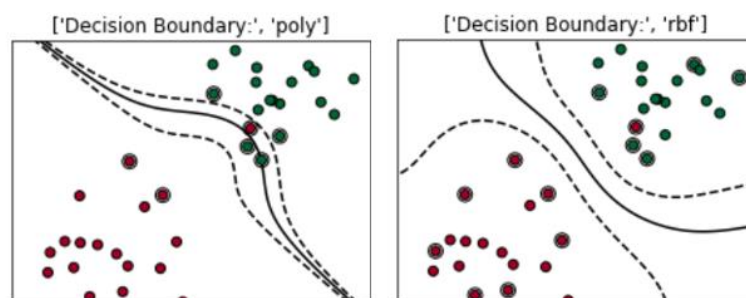


Figure 25: Types of SVM Kernel

K – Nearest Neighbours (KNN)

K-Nearest Neighbours (KNN) is a supervised machine learning algorithm. It can be used for both classification as well as regression. This algorithm works on the basis of the 'k' nearest points. The 'k' in KNN is the nearest neighbours which are considered to classify a given data point. The greater number of points belonging to same class around a new data point the higher the probability for the point to be classified for that class. The value of 'k' can be chosen based on two factors: training error rate and the validation error rate. A value for 'k' needs to be chosen where both training and testing error rate is minimum. After the minima point, it then increases with increasing K. To get the optimal value of K, you can segregate the training and validation from the initial dataset. Now plot the validation error curve to get the optimal value of K. This value of K should be used for all predictions.^[23]

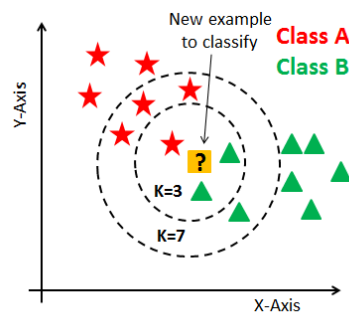


Figure 26: KNN Classifier

Random Forest

Random Forest is a supervised machine learning algorithm. It can be used for both classification as well as regression. It is made of many decision trees. These are ensembles of decision tree. Each tree is created using a subset of the attributes used to classify a given dataset. These decision trees vote on how to classify a given instance set of input, and the random forest bootstraps those votes to choose the best prediction. This is done to prevent overfitting, a common flaw of decision trees. The larger the number of trees the better results can be expected.^[24]

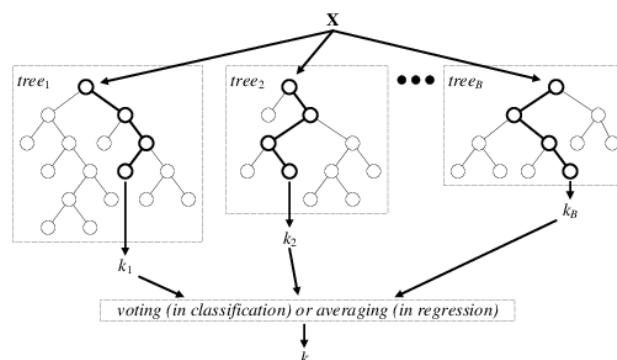


Figure 27: Random Forest Tree

The programming language chosen for creating the machine learning model is R and tool used is R Studio.

The steps taken for developing the model is as follows:

Step 1: Load the dataset

Step 2: Clean the dataset

Step 3: Split the dataset

Step 4: Train the model

Step 1: Load the dataset

The dataset consists of 11829 observations for five variables. The following are the feature and feature description of the dataset.

- (i) Id – This column is the name of the image or the handwriting sample chosen for analysis. The dataset chosen here is CVL dataset which consists of images with extension ‘tif’.
- (ii) bl_norm – This column is the normalized value for the baseline. The value of baseline was calculated using the formula $y_2 - y_1$. The normalized value is the baseline feature vector divided by height of the image as all the images have different heights based on the size of handwriting.
- (iii) ws_norm – This column is the normalized value for the word spacing. The distance between the words was calculated using the absolute value of sum of all the distances divided by number of words. The normalized value is the word spacing feature vector divided by width of the image as all the images have different width based on the text and size of handwriting.
- (iv) op – This column is the output or target variable. This value is mapped with the feature vectors based on manual analysis. The meanings of the output variables are explained in table 11

R Code to load the dataset:

```
df= read.csv(file.choose())

str(df)

'data.frame': 11828 obs. of 5 variables:
 $ i..id : Factor w/ 11828 levels "0052-1-0.tif",...: 2012 2001 1975 1987 1988 1976 2002 2013 1963 2011 ...
 $ bl_norm: num -0.067 0.117 -0.096 0.11 -0.253 0 0.304 -0.093 0.214 0.149 ...
 $ ws_norm: num 0.8 0.258 0.555 0.387 0.633 0.515 0.4 0.525 0.665 0.475 ...
 $ pp_norm: num 0.871 0.867 0.863 0.871 0.863 ...
 $ op : Factor w/ 8 levels "DNH","DNL","DWH",...: 4 8 4 8 4 4 8 4 8 8 ...
```

Figure 28: Structure of dataset

Step 2: Clean the dataset

The first column of the dataset is Id. This column is unique for each row. Since, it adds no value to the prediction process, it can be dropped. The features are numeric in nature and the target variable is factor with 8 levels.

R Code to drop unwanted column:

```
df = df[,-1]
```

```
head(df)
```

```
head(df)
  bl_norm ws_norm pp_norm op
-0.067   0.800 0.870588235 DWL
 0.117   0.258 0.866666667 UWL
-0.096   0.555 0.862745098 DWL
 0.110   0.387 0.870588235 UWL
-0.253   0.633 0.862745098 DWL
 0.000   0.515 0.862745098 DWL
```

Figure 29: Head of dataset

Step 3: Split the dataset

Machine learning process works on the principle of training the model to learn iteratively. Once the model learns it can predict the output for future data. To make sure that the model works well first it needs to be trained. To train the model, since it is a supervised learning, features along with the target variable needs to be fed. The model iteratively works and improve its accuracy. Once it reaches a point when it cannot learn any further from the data it stops.

Once the model is trained then it needs to be validated on data. This validation set needs to have the output variable so that it can be compared to the predicted variable to find the accuracy of the model. Accuracy is the percentage of correctly predicted values. Based on the accuracy it can justified if the model is performing well or if it overfits or underfits.

Overfitting happens when a model learns the detail and noise in the training data to the extent that it negatively impacts the performance of the model on new data. This means that the noise or random fluctuations in the training data is picked up and learned as concepts by the model. The problem is that these concepts do not apply to new data and negatively impact the model's ability to generalize.^[11]

Underfitting refers to a model that can neither model the training data nor generalize to new data. An underfit machine learning model is not a suitable model and will be obvious as it will have poor performance on the training data.^[25]

We need to make sure the model is neither overfitting nor underfitting.

To achieve this the dataset is divided into two parts – Training and Test set. The split ratio can be anything between 50 and 90 as trainset and the rest as test set respectively. It can be compared to find the best fit.

Once the dataset is split, use the major portion of the dataset as trainset to train the model. The remaining portion of data can be used as test set. Using the test set the model performance can be evaluated.

R Code for splitting the dataset df:

```
n=nrow(df)
indexes = sample(n,n*(80/100))
trainset = df[indexes,]
testset = df[-indexes,]
```

trainset	9462 obs. of 4 variables
bl_norm:	num -0.272 0.038 0.37 -0.245 0.065 -0.548 -0.029 -0.032 0.117 -0.021 ...
ws_diff:	num 0.556 0.4 0.417 -0.273 -0.121 ...
pp_norm:	num 0.796 0.871 0.875 0.843 0.894 ...
op :	Factor w/ 8 levels "DNH","DNL","DWH",...: 3 4 8 2 6 2 4 4 8 4 ...

Figure 30: Trainset snapshot

testset	2366 obs. of 4 variables
bl_norm:	num 0 0.214 -0.266 -0.144 0.022 0.121 0.107 -0.107 0.136 0.179 ...
ws_diff:	num 0.75 0.889 0.7 0.25 0.833 ...
pp_norm:	num 0.863 0.859 0.859 0.863 0.855 ...
op :	Factor w/ 8 levels "DNH","DNL","DWH",...: 4 8 4 4 4 8 8 4 8 8 ...

Figure 31: Testset snapshot

Step 4: Train the model

Each algorithm has different parameter requirement. Based on each model requirement the parameters are supplied.

The common parameters passed or functions used to train the model are as follows:

Formula – In this case, $op \sim .$ implies predict variable ‘op’ using all the other variables as input.

Data – The dataset which should be used to perform the function. In this case, for creating model use ‘trainset’ and for testing use the ‘testset’.

predict() – This function is used to check the model accuracy.

confusionMatrix() – It is a function used to create the matrix which helps to evaluate the overall performance of the model.

R code for

Decision Tree –

```
#Decision Tree
library(rpart)
dt <- rpart(op ~., data = trainset, method = "class", cp = 0.02)
pred_dt= predict(dt,testset,type='class')
confusionMatrix(pred_dt,testset$op)
rpart.plot(dt, box.palette="RdBu", shadow.col="gray", nn=TRUE)
```

cp is the complexity parameter. The main role of this parameter is to save computing time by pruning off splits that are obviously not worthwhile.

Naïve Bayes –

```
#Naive Bayes
train_control = trainControl(method="cv", number=30)
nb_model <- train(op~., data=trainset, trControl=train_control, method="nb")
pred_nb_model= predict(nb_model,testset,type='raw')
confusionMatrix(pred_nb_model,testset$op)
```

trControl parameter is cross validation parameter. It improves the model performance

Support Vector Machine –

```
#SVM - Radial (RBF)
SVM_r <- svm(op ~., data = trainset, kernel='radial', cross = 10)
pred_svmr = predict(SVM_r,testset,type='class')
confusionMatrix(pred_svmr,testset$op)

#SVM - Poly
SVM_p <- svm(op ~., data = trainset, kernel='poly', cross = 10)
pred_svmp = predict(SVM_p,testset,type='class')
confusionMatrix(pred_svmp,testset$op)
```

Two types of kernel are used to evaluate how they affect the model performance.

K Nearest Neighbors (KNN) –

```
#KNN
acc = array()
max_acc = 0
best_k = 0
for (i in 1:20){
  knn = kknn(op ~ ., train=trainset, test=testset, k=i)
  pred_knn = predict(knn, testset, type = 'raw')
  tab=table(pred_knn,testset$op)
  accuracy=sum(tab[row(tab)==col(tab)])/sum(tab)
  acc[i] = accuracy
  if (accuracy > max_acc){
    max_acc = accuracy
    best_k = i
  }
}
acc
max_acc
best_k
i = seq(1:20)
plot(i,acc*100, xlab = "k Value", ylab = "Accuracy", main = "Values for k against accuracy",
las = 1)
```

The best value of k needs to be found by trail and error method. Here, a range of values from 1 to 20 are checked to find the best suited k value.

Random Forest –

```
#Random Forest
require(randomForest)
rf = randomForest(op ~ . , type='class', data = trainset, ntree=10)
pred_rf= predict(rf,testset,type='class')
confusionMatrix(pred_rf, testset$op)
```

ntree parameter - Number of trees to grow. This should not be set to too small a number, to ensure that every input row gets predicted at least a few times.

Chapter 5 - Evaluation & Results

Model Evaluation is an integral part of the model development process. It helps to find the best model that represents our data and how well the chosen model will work in the future. Evaluating model performance with the data used for training is not acceptable in data science because it can easily generate overoptimistic and overfitted models. There are two methods of evaluating models in data science, Hold-Out and Cross-Validation. To avoid overfitting, both methods use a test set (not seen by the model) to evaluate model performance.^[26]

There are various techniques for validations each work well for a particular type of machine learning model. Based on that characteristic model evaluation can be divided into two parts:

- (i) Classification Model Evaluation
- (ii) Regression Model Evaluation

Classification Model can be evaluated for performance using any of the following techniques:

- (i) Confusion matrix
- (ii) Gain and Lift Charts
- (iii) Lift Charts
- (iv) K - S Chart
- (v) ROC Chart
- (vi) Area under the Curve (AUC)

A confusion matrix shows the number of correct and incorrect predictions made by the classification model compared to the actual outcomes (target value) in the data. The matrix is $N \times N$, where N is the number of target values (classes). Performance of such models is commonly evaluated using the data in the matrix. The following table displays a 2x2 confusion matrix for two classes (Positive and Negative).^[26]

Confusion Matrix		Target			
		Positive	Negative		
Model	Positive	a	b	Positive Predictive Value	$a/(a+b)$
	Negative	c	d	Negative Predictive Value	$d/(c+d)$
		Sensitivity	Specificity	Accuracy = $(a+d)/(a+b+c+d)$	
		$a/(a+c)$	$d/(b+d)$		

Figure 32: Confusion Matrix

Accuracy: the proportion of the total number of predictions that were correct.

Positive Predictive Value or Precision: the proportion of positive cases that were correctly identified.

Negative Predictive Value: the proportion of negative cases that were correctly identified.

Sensitivity or Recall: the proportion of actual positive cases which are correctly identified.

Specificity: the proportion of actual negative cases which are correctly identified.

Decision Tree

The following tree visualizes the decision tree. The tree displays all the conditions and the branching of nodes. Following that is the confusion matrix which gives the accuracy.

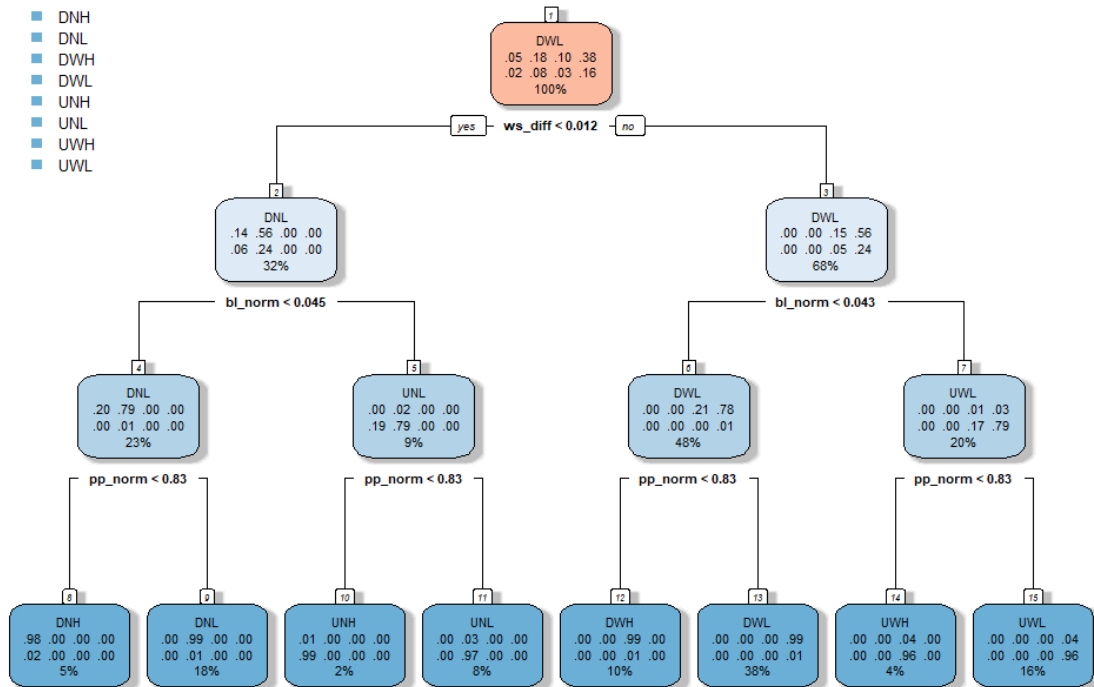


Figure 33: Decision Tree

Confusion Matrix and Statistics

	Reference							
Prediction	DNH	DNL	DWH	DWL	UNH	UNL	UWH	UWL
DNH	115	0	0	0	1	0	0	0
DNL	0	432	0	0	0	5	0	0
DWH	0	0	218	0	0	0	0	0
DWL	0	0	0	890	0	0	0	3
UNH	1	0	0	0	37	0	0	0
UNL	0	1	0	0	0	161	0	0
UWH	0	0	4	0	0	0	83	0
UWL	0	0	0	23	0	0	0	392

Overall Statistics

Accuracy : 0.9839391
 95% CI : (0.9780213, 0.98861)
 No Information Rate : 0.3858833
 P-value [Acc > NIR] : < 0.00000000000000022204
 Kappa : 0.9792461
 McNemar's Test P-Value : NA

The accuracy of Decision Tree model is 98.39%.

Naïve Bayes

Confusion Matrix and Statistics

Prediction	Reference							
	DNH	DNL	DWH	DWL	UNH	UNL	UWH	UWL
DNH	129	0	0	0	2	0	0	0
DNL	0	434	0	0	0	5	0	0
DWH	1	0	226	0	0	0	6	0
DWL	0	0	0	877	0	0	0	19
UNH	0	0	0	0	39	0	1	0
UNL	0	3	0	0	0	167	0	0
UWH	0	0	3	0	0	0	99	0
UWL	0	0	0	4	0	0	2	349

Overall Statistics

Accuracy : 0.9805579

The accuracy of Naïve Bayes model is 98.05%

Support Vector Machine - Poly

Confusion Matrix and Statistics

Prediction	Reference							
	DNH	DNL	DWH	DWL	UNH	UNL	UWH	UWL
DNH	118	1	1	0	1	0	0	0
DNL	10	364	0	2	2	8	0	0
DWH	2	0	211	0	0	0	4	0
DWL	0	68	11	877	0	14	0	67
UNH	0	0	0	0	38	0	0	0
UNL	0	4	0	0	0	147	0	1
UWH	0	0	5	0	0	0	91	0
UWL	0	0	1	2	0	3	13	300

Overall Statistics

Accuracy : 0.9070161

Support Vector Machine - RBF

Confusion Matrix and Statistics

Prediction	Reference							
	DNH	DNL	DWH	DWL	UNH	UNL	UWH	UWL
DNH	130	0	1	0	5	0	0	0
DNL	0	434	0	3	0	6	0	1
DWH	0	0	222	0	0	0	5	0
DWL	0	0	4	873	0	0	0	15
UNH	0	0	0	0	36	0	1	0
UNL	0	3	0	0	0	166	0	3
UWH	0	0	2	0	0	0	100	0
UWL	0	0	0	5	0	0	2	349

Overall Statistics

Accuracy : 0.9763314

SVM was experimented with two kernels and the accuracy showed a huge difference.

SVM Poly – 90.7% ; SVM RBF – 97.63%

K – Nearest Neighbours

KNN was modelled iteratively to find the best value of k. The graph displays all the accuracy values for each value of k between 1 and 20.

The graph shows that k = 15 is the best value with the highest accuracy of 96%.

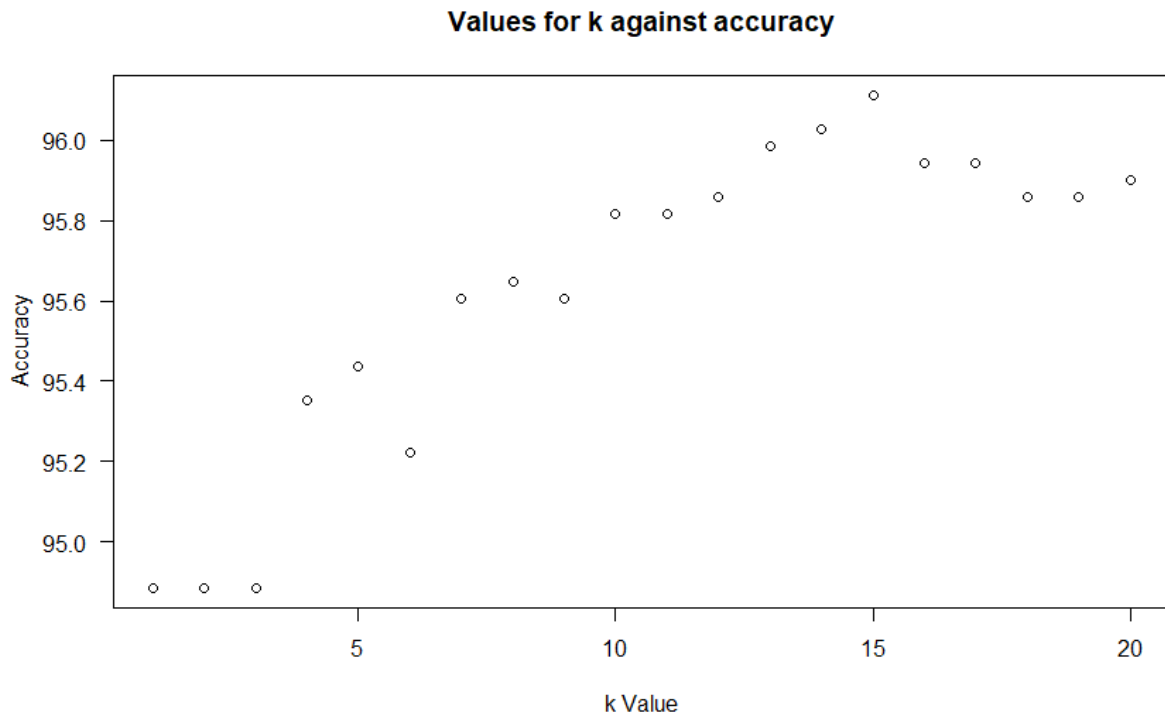


Figure 34: Values of k against accuracy

Random Forest

Plot the error rates or MSE of a Random Forest object.

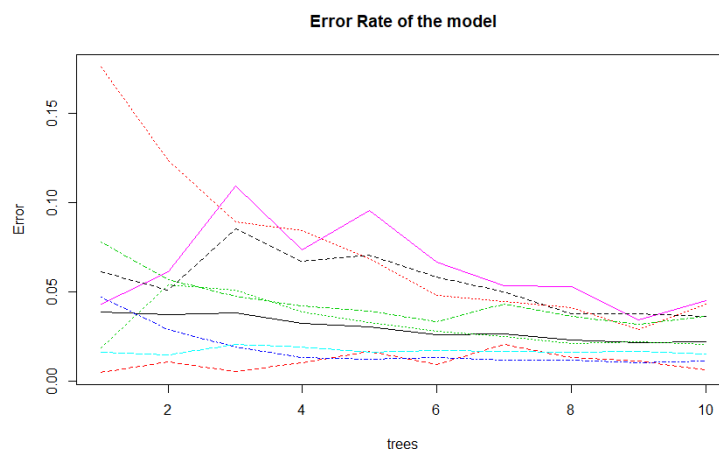


Figure 35: Error rate of Random Forest

Confusion Matrix and Statistics

		Reference							
Prediction		DNH	DNL	DWH	DWL	UNH	UNL	UWH	UWL
DNH		129	0	0	0	1	0	0	0
DNL		0	431	0	0	0	5	0	0
DWH		0	0	226	0	0	0	0	0
DWL		0	0	0	873	0	0	0	14
UNH		1	0	0	0	40	0	0	0
UNL		0	6	0	0	0	167	0	0
UWH		0	0	3	0	0	0	108	0
UWL		0	0	0	8	0	0	0	354

Overall statistics

Accuracy : 0.9801352

The confusion matrix shows that the accuracy of Random Forest model is 98.01%

The overall comparison of the machine learning algorithm looks like this:

Accuracy		
[1,]	"Decision Tree"	"0.9839391"
[2,]	"Naive Bayes"	"0.9805579"
[3,]	"SVM RBF"	"0.9763314"
[4,]	"SVM Polynomial"	"0.9070161"
[5,]	"KNN"	"0.9611158"
[6,]	"Random Forest"	"0.9801352"

Since, Decision tree has the highest accuracy of 98.39% it is the best model.

Chapter 6 - Conclusion & Future work

This paper has proposed a methodology to automate the task of predicting human traits from various handwriting features using machine learning approach. It explores the features of handwriting like pen pressure, baseline and word spacing to reveal personality traits. The CVL open source handwriting dataset is used for conducting the research. The dataset was analysed manually for almost 12,000 samples. The samples were manually analysed for selected features and after thorough study, the images were mapped to personality traits. Using OpenCV in Python for image processing feature vectors were extracted. The feature vector was mapped to the manually analysed and created personality traits. The prepared dataset was then fed to multiple machine learning algorithms Decision Tree, Naïve Bayes, K – Nearest Neighbour, Support Vector Machine (SVM) and Random Forest. SVM model was experimented with different types of kernels like Polynomial and Radial. On comparing the machine learning algorithms, it was found Decision Tree worked the best with an accuracy of ~98.39%.

The proposed system can be used as a complementary tool by the graphologist to improve the accuracy of handwriting analysis and make the process fast. It will also assist the HR/company employer in decision making regarding the suitability of an employee for the specific job and improving the retention of an employee. It will aid the security department to evaluate people's honesty and integrity. It will benefit the Psychiatrist/Psychologist to deal with patients, as handwriting will reveal much more about the patient who are shy or unclear about themselves. It will ease the process of self-awareness and self-development. It will encourage historical profiling by making it much easier for the genealogists and biographers to identify personality traits much more accurately, of people who passed away.

The future work can be to include more features from the micro approach of handwriting analysis like the slant, size of letter, height of 't'-bar, margin, loops of alphabet 'f' and 'l', gradient, concavity of letters, margins, 'i' dot and so on in order to predict more accurate results.

The system can be expanded to link the handwriting characteristics to personality indicators. The popular personality indicators like Myer Briggs, Big Five Personality Traits, etc

An R Shiny app can be built on top of the machine learning models which will make the application user friendly and easy to use.

References:

1. <https://www.britishgraphology.org/about-british-institute-of-graphologists/the-history-of-graphology/>
2. <https://www.monash.edu/rlo/graduate-research-writing/write-the-thesis/introduction-literature-reviews>
3. Shitala Prasad, Vivek Kumar Singh, Akshay Sapre "Handwriting Analysis based on Segmentation Method for Prediction of Human Personality using Support Vector Machine", International Journal of Computer Applications, Volume 8– No.12, October 2010.
4. Champa H. N., 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 2010.
5. Prachi Joshi, Aayush Agarwal, Ajinkya Dhavale, "Handwriting Analysis for Detection of Personality Traits using Machine Learning Approach", International Journal of Computer Applications, vol. 130, no. 15, pp. 0975-8887, November 2015.
6. Vikram Kamath, Nikhil Ramaswamy, P. Navin Karanth, Vijay Desai and S. M. Kulkarni, (2011) "Development of an Automated Handwriting Analysis System", ARPN Journal of Engineering and Applied Sciences Volume 6- No.9, September 2011 ISSN 1819-660.
7. Abhishek Bal, Rajib Saha: An Improved Method for Handwritten Document Analysis using Segmentation, Baseline Recognition and Writing Pressure Detection. In: 6th IEEE International Conference on Advances in Computing and Communications (ICACC-2016), Sept 6–8, 2016, Volume 93, Pages 403–415
8. M. Abdul Rahiman, Diana Varghese, Manoj Kumar G: HABIT: Handwritten Analysis Based Individualistic Traits Prediction In: International Journal of Image Processing; 7 (2) (2013)
9. Rashi Kacker, Hima Bindu Maringanti, "Personality analysis through handwriting", GSTF Journal on Computing (JoC) Vol.2 No.1, April 2012
10. G. Sheikholeslami, S. N. Srihari, V. Govindaraju, "COMPUTER AIDED GRAPHOLOGY", Center of Excellence for Document Analysis and Recognition. Ding, W. and Marchionini, G. 1997 A Study on Video Browsing Strategies. Technical Report. University of Maryland at College Park
11. <https://masterofproject.com/blog/3457/what-is-methodology>
12. <https://en.wikipedia.org/wiki/SEMMA>
13. <https://www.britishgraphology.org/about-british-institute-of-graphologists/what-is-graphology/>
14. http://handwritinglense.com/7Significance_of_Pressure_in_Handwriting_Analysis.html
15. http://www.2knowmyself.com/communication_skills/Graphology_handwriting_analysis_pressure
16. <http://handwritinginsights.com/wp/baseline-handwriting-means/>
17. <https://bhagyashreewarke.com/graphology-spacing/>
18. Florian Kleber, Stefan Fiel, Markus Diem and Robert Sablatnig, CVL-Database: An Off-line Database for Writer Retrieval, Writer Identification and Word Spotting, In Proc. of the 12th Int. Conference on Document Analysis and Recognition (ICDAR) 2013, pp. 560-564, 2013.
19. <http://www.labbookpages.co.uk/software/imgProc/otsuThreshold.html>
20. <https://towardsdatascience.com/decision-trees-in-machine-learning-641b9c4e8052>
21. <https://www.machinelearningplus.com/predictive-modeling/how-naive-bayes-algorithm-works-with-example-and-full-code/>
22. <https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/>
23. <https://www.analyticsvidhya.com/blog/2018/03/introduction-k-neighbours-algorithm-clustering/>
24. <https://skymind.ai/wiki/random-forest>

25. <https://machinelearningmastery.com/overfitting-and-underfitting-with-machine-learning-algorithms/>
26. https://www.saedsayad.com/model_evaluation.htm