Nitte Meenakshi Institute of Technology,

Department of Computer Science and Engineering

18CSE751 Introduction to Machine Learning

Learning Activity Report

Personality type prediction

Submitted by:

Gagan R (1NT18CS041)

Monish K (1NT18CS100)

A S Prithvi Raj (1NT18CS001)

Under the guidance of:

Dr. Vani V

Dept. C S & E, NMIT

Abstract

The Myers–Briggs Type Indicator (MBTI), describes the preferences of an individual in four dimensions and these basic dimensions combine into one of 16 different personality types. These four dimensions or basic meta programs are Extroversion–Introversion (E–I), Sensation–Intuition (S–N), Thinking–Feeling (T–F), and Judgment– Perception (J–P).

Each dimension represents two types of personalities. Figure 1 shows a key of the eight personality types used in the Myers–Briggs Type Indicator.

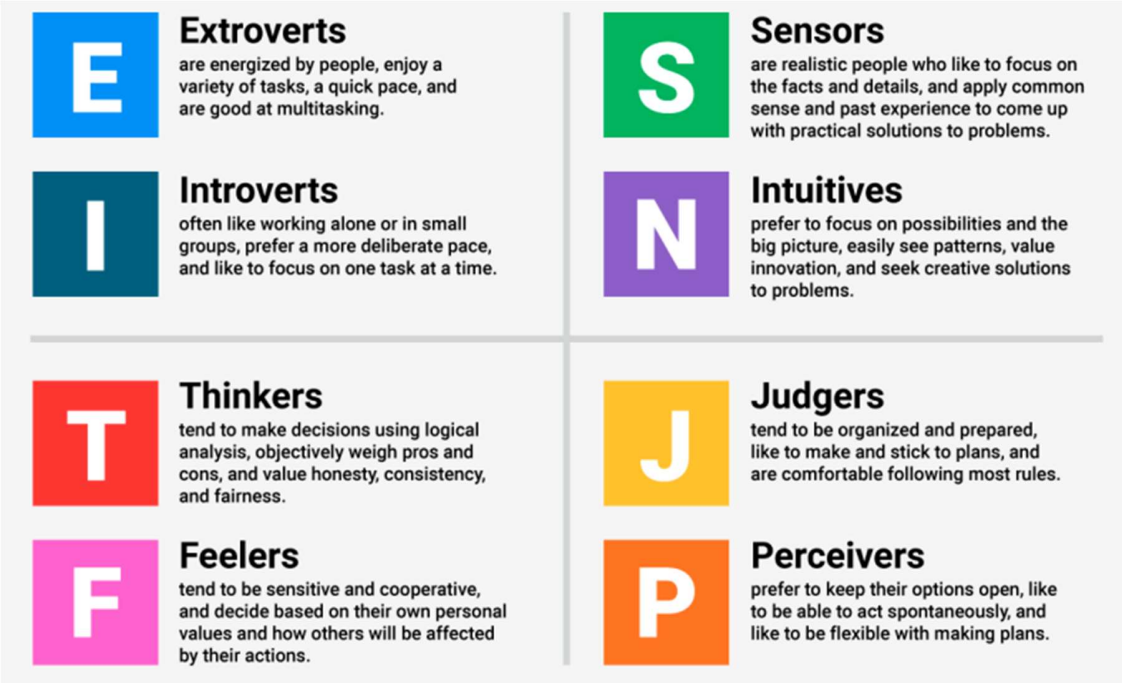

Fig 1. Personality types.

Personality is derived from the Latin word persona, which means describing the behavior or character of an individual. It has been said that the meaning of personality is reflected in the very nature of the attitude of a person that can be distinguished from other people. Personality, according to Hall and Lindzey, is “the dynamic organization within the individual of those psychological systems that determine his characteristic behavior and

thought.” This system determines the unique way in which an individual adapts to an environment. Personality is a description of the individual’s self-image that influences their behavior uniquely and dynamically, and this behavior may change through the process of learning, experience, education, etc.

As discussed above, the preferences of an individual are categorized into four dimensions, and different combinations of the personality type key in these categories represent 16 different personality types based on the Myers–Briggs Type Indicator.

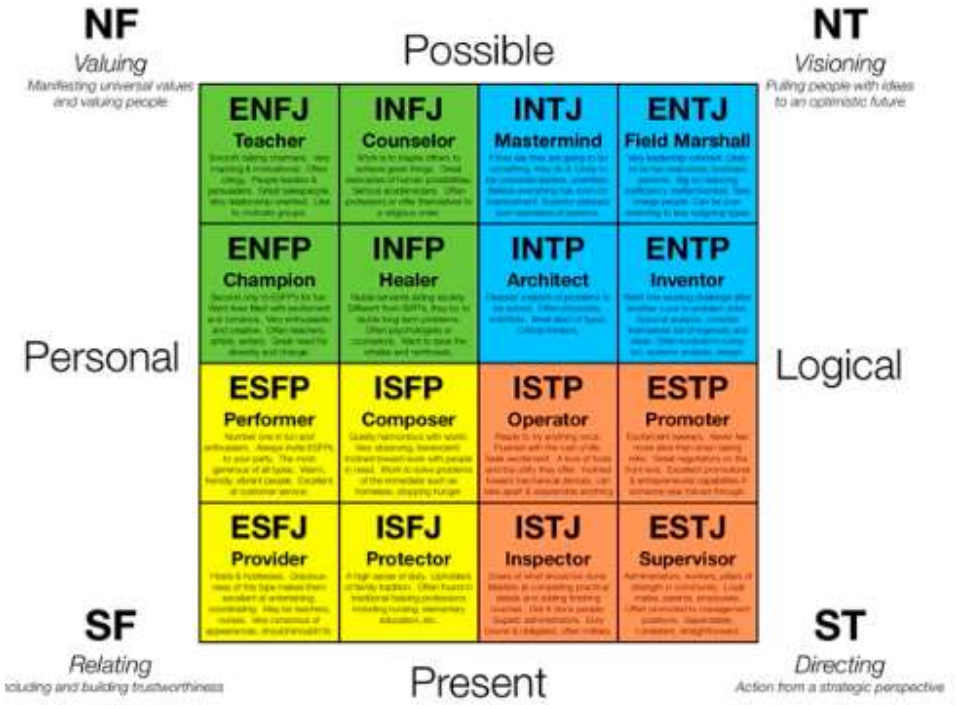

Fig 2. Personality types in MBTI.

As a result, the Myers–Briggs Type Indicator (MBTI) has been used in this project to predict the personality type of individuals.

Table Of Contents

1. Introduction

Machine learning is a subfield of artificial intelligence (AI). The goal of machine learning generally is to understand the structure of data and fit that data into models that can be understood and utilized by people.

Although machine learning is a field within computer science, it differs from traditional computational approaches. In traditional computing, algorithms are sets of explicitly programmed instructions used by computers to calculate or problem solve. Machine learning algorithms instead allow for computers to train on data inputs and use statistical analysis to output values that fall within a specific range. Because of this, machine learning facilitates computers in building models from sample data to automate decision-making processes based on data inputs.

Machine learning is a continuously developing field. Because of this, there are some considerations to keep in mind as you work with machine learning methodologies or analyze the impact of machine learning processes.

1.1 Motivation

Personality tests are often used to gain insight into who people are and what motivates them. From an employer perspective, understanding the personality of a potential hire can shed light on their work style and how they might fit into the company's work culture.

Stemming from the field of psychology, personality tests have been used to better understand character traits in a variety of settings—including, if not especially, the workplace. Otherwise, they can be useful for psychological diagnoses by mental health professionals, personal development or nurturing positive relationships with others. Over the years, an innumerable number of personality tests have come into popularity, many of which still circulate or are easily accessible online.

Thus, we have decided to use Myers-Briggs Type Indicator (MBTI) to test the personality of a person.

1.2 Problem statement

The Myers-Briggs Type Indicator is often used by companies during the hiring process. Its questions determine where an applicant falls within four key groupings: extraversion vs. introversion, judging vs. perceiving, intuition vs. sensing and thinking vs. feeling. The results of these groupings place test-takers into one of 16 personality types. Thus, making a one stop solution for the employers to select a perfect candidate for any scenario.

1.3 Objectives

- To understand classification models
- To achieve low accuracy for multi class
- To use a semi-parametric approach and observe over-fitting
- Construct a tree using Gini index and observe better-fitting

2. Data Pre-processing

The dataset contains over 8600 rows of data where each row contains:

- Type - Personality type (4 letter MBTI code)
- Post - Last 50 things a user has posted (Each entry separated by "|||" (3 pipe characters))

Data cleaning is the process of fixing or removing incorrect, corrupted, incorrectly formatted, duplicate, or incomplete data within a dataset. When combining multiple data sources, there are many opportunities for data to be duplicated or mislabeled. If data is incorrect, outcomes and algorithms are unreliable, even though they may look correct.

Thus, in our code we have removed unwanted links and symbols to avoid bad outcomes.

This function is achieved by the class cleaner.

2.1 Natural Language Processing (NLP)

NLP is a field in machine learning with the ability of a computer to understand, analyze, manipulate, and potentially generate human language.

While reading data, we get data in the structured or unstructured format. A structured format has a well-defined pattern whereas unstructured data has no proper structure. In between the 2 structures, we have a semi-structured format which is comparably better structured than unstructured format.

Cleaning up the text data is necessary to highlight attributes that we're going to want our machine learning system to pick up on. Cleaning (or pre-processing) the data typically consists of a few steps:

1. **Remove stopwords**

   Stopwords are common words that will likely appear in any text. They don't tell us much about our data, so we remove them. E.g.: "English".

2. **Lemmatization**

   Lemmatizing derives the canonical form (‘lemma’) of a word. i.e., the root forms. It is better than stemming as it uses a dictionary-based approach i.e., a morphological analysis to the root word. This is performed on each word in the cleaned data.

   Stemming is typically faster as it simply chops off the end of the word, without understanding the context of the word. Lemmatizing is slower and more accurate as it takes an informed analysis with the context of the word in mind.

3. **TFIDF**

   It computes the “relative frequency” that a word appears in a document compared to its frequency across all documents. It is more useful than “term frequency” for identifying “important” words in each document (high frequency in that document, low frequency in other documents).

   It is used for search engine scoring, text summarization, document clustering.

   TFIDF vectorization is used on each row lemmatized sentences to extract top 1000 tokenized attributes.

3. Machine learning methods

3.1. Multi class CART

CART is a powerful algorithm that is also relatively easy to explain compared to other ML approaches. It does not require much computing power, hence allowing us to build models very fast.

While we need to be careful not to overfit our data, it is a good algorithm for simple problems.

Use Gini impurity for selecting the best split.

Drawbacks:

- Low accuracy is observed

3.2. Multi class SVM

SVM are applied on binary classification, dividing data points either in 1 or 0. For multiclass classification, the same principle is utilized. The multiclass problem is broken down to multiple binary classification cases, which is also called *one-vs-one*. In scikit-learn one-vs-one is not default and needs to be selected explicitly (as can be seen further down in the code). *One-vs-rest* is set as default. It basically divides the data points in class x and rest. Consecutively a certain class is distinguished from all other classes.

The number of classifiers necessary for *one-vs-one multiclass classification* can

$$\frac{n * (n-1)}{2}$$

be retrieved with the following formula (with n being the number of classes):

In the one-vs-one approach, each classifier separates points of two different classes and comprising all one-vs-one classifiers leads to a multiclass classifier.

Drawbacks:

- Poor accuracy and precision.
- Slow modeling.
- Consumes a lot of memory space.

3.3. Binary classification SVC

The objective of a Linear SVC (Support Vector Classifier) is to fit to the data you provide, returning a "best fit" hyperplane that divides, or categorizes, your data. From there, after getting the hyperplane, we can then feed some features to your classifier to see what the "predicted" class is. This makes this specific algorithm rather suitable for our uses. Also apply the class_weight as balanced.

RBF kernel:

RBF is the default kernel used within the sklearn's SVM classification algorithm and can be described with the following formula:

$$K(x, x') = e^{-\gamma||x-x'||^2}$$

where gamma can be set manually and has to be >0. The default value for gamma in sklearn's SVM classification algorithm is:

$$\gamma = \frac{1}{n\,features * \sigma^2}$$

Briefly:

||x - x'||² is the squared Euclidean distance between two feature vectors (2 points).

Gamma is a scalar that defines how much influence a single training example (point) has.

So, given the above setup, we can control individual points' influence on the overall algorithm. The larger gamma is, the closer other points must be to affect the model.

Drawbacks:

- Over-fitting.

3.4. Binary CART

Same as Multi class CART but with max_depth as 100.

Drawbacks:

- Less accurate than binary SVC.

Advantage:

- Not over-fitting.

4. Results

	precision	recall	f1-score	support
INFJ	0.24	0.26	0.25	34
ENTP	0.43	0.41	0.42	147
INTP	0.20	0.22	0.21	49
INTJ	0.38	0.36	0.37	154
ENTJ	0.00	0.00	0.00	11
ENFJ	0.00	0.00	0.00	11
INFP	0.12	0.17	0.14	6
ENFP	0.12	0.12	0.12	16
ISFP	0.53	0.53	0.53	280
ISTP	0.58	0.61	0.59	375
ISFJ	0.44	0.40	0.42	222
ISTJ	0.48	0.48	0.48	248
ESTP	0.10	0.11	0.10	28
ESFP	0.30	0.31	0.31	48
ESTJ	0.27	0.30	0.29	47
ESFJ	0.43	0.44	0.44	59
accuracy			0.45	1735
macro avg	0.29	0.30	0.29	1735
weighted avg	0.45	0.45	0.45	1735

	precision	recall	f1-score	support
INFJ	0.64	0.62	0.63	34
ENTP	0.69	0.60	0.64	147
INTP	0.51	0.37	0.43	49
INTJ	0.74	0.60	0.66	154
ENTJ	0.60	0.27	0.37	11
ENFJ	0.00	0.00	0.00	11
INFP	0.67	0.33	0.44	6
ENFP	0.62	0.31	0.42	16
ISFP	0.67	0.71	0.69	280
ISTP	0.67	0.83	0.74	375
ISFJ	0.66	0.71	0.69	222
ISTJ	0.67	0.74	0.70	248
ESTP	0.53	0.32	0.40	28
ESFP	0.56	0.38	0.45	48
ESTJ	0.83	0.53	0.65	47
ESFJ	0.77	0.63	0.69	59
accuracy			0.67	1735
macro avg	0.61	0.50	0.54	1735
weighted avg	0.67	0.67	0.67	1735

0 : Introvert , 1 : Extrovert

	precision	recall	f1-score	support
0	0.89	0.92	0.91	1307
1	0.72	0.67	0.70	428
accuracy			0.86	1735
macro avg	0.81	0.79	0.80	1735
weighted avg	0.85	0.86	0.85	1735

0 : Introvert , 1 : Extrovert

	precision	recall	f1-score	support
0	0.84	0.86	0.85	1307
1	0.53	0.51	0.52	428
accuracy			0.77	1735
macro avg	0.69	0.68	0.68	1735
weighted avg	0.77	0.77	0.77	1735

0 : Intuition , 1 : Sensing

	precision	recall	f1-score	support
0	0.94	0.96	0.95	1509
1	0.67	0.58	0.62	226
accuracy			0.91	1735
macro avg	0.80	0.77	0.79	1735
weighted avg	0.90	0.91	0.91	1735

0 : Intuition , 1 : Sensing

	precision	recall	f1-score	support
0	0.92	0.91	0.91	1509
1	0.44	0.47	0.45	226
accuracy			0.85	1735
macro avg	0.68	0.69	0.68	1735
weighted avg	0.86	0.85	0.85	1735

0 : Feeling , 1 : Thinking

	precision	recall	f1-score	support
0	0.87	0.88	0.88	934
1	0.86	0.85	0.85	801
accuracy			0.87	1735
macro avg	0.87	0.87	0.87	1735
weighted avg	0.87	0.87	0.87	1735

0 : Feeling , 1 : Thinking

	precision	recall	f1-score	support
0	0.75	0.75	0.75	934
1	0.71	0.70	0.70	801
accuracy			0.73	1735
macro avg	0.73	0.73	0.73	1735
weighted avg	0.73	0.73	0.73	1735

0 : Perseving , 1 : Judging

	precision	recall	f1-score	support
0	0.83	0.85	0.84	1058
1	0.75	0.73	0.74	677
accuracy			0.80	1735
macro avg	0.79	0.79	0.79	1735
weighted avg	0.80	0.80	0.80	1735

0 : Perseving , 1 : Judging

	precision	recall	f1-score	support
0	0.75	0.76	0.76	1058
1	0.62	0.61	0.62	677
accuracy			0.70	1735
macro avg	0.69	0.69	0.69	1735
weighted avg	0.70	0.70	0.70	1735

5. Conclusion

- Successfully implemented different classification methods
- On comparison of various methods, we found that some of the models are precise whereas some are accurate
- Also, we found some drawbacks for each implemented method

6. Takeaway

On completion of this project, we were able to understand how two different models work for the same problem in the field of Natural Language Processing.

7. References

[1] Scikit-learn documentation: https://scikit-learn.org/stable/

[2] Machine learning mastery: https://machinelearningmastery.com/

[3] Towards data science: https://towardsdatascience.com/machine-learning/home

[4] Analytics Vidhya: https://www.analyticsvidhya.com/machine-learning/

Appendix

1. Dataset

Kaggle: https://www.kaggle.com/datasnaek/mbti-type

2. Code

GitHub: https://github.com/gaganr17/personality-prediction-system/blob/main/Project-ML.ipynb

3. Setup

3.1. Install Python 3
3.2. Download scikit-learn libraries