Report

Abhishek Jilowa (B20CS001)

Abstract - This project's main goal is to use machine learning to create a classifier that can group people into their Myers-Briggs Type Index (MBTI) personality types based on text samples from their social media postings. There are two reasons for developing such a classifier. First, because social media is so widely used, such a classifier would have enough data on which to administer personality tests, allowing more individuals to learn their MBTI personality type, and maybe more reliably and rapidly. This topic is generating a lot of buzz in the psychology community, both academically and in the corporate sector. Thus, our classifier could serve as a verification system for these initial tests as a means of allowing people to have more confidence in their results. Indeed, a text-based classifier would be able to operate on a far larger amount of data than that given in a single personality test.

I. INTRODUCTION

Personality is a crucial factor since it distinguishes one person from another. As a result, understanding personality is an intriguing subject for scholars to explore. Personality prediction has a wide range of applications in the actual world. The use of social media is growing every day. With the advanced development in machine learning (ML), artificial intelligence (AI) and data science has been shown to be effective in assisting in decision making and predictions from the large quantity of data.

Dataset

The dataset contains 8675 rows and 2 columns containing:

- type-label

- posts-feature

	type	posts
0	INFJ	'http://www.youtube.com/watch?v=qsXHcwe3krw|||...
1	ENTP	'I'm finding the lack of me in these posts ver...
2	INTP	'Good one _____ https://www.youtube.com/wat...
3	INTJ	'Dear INTP, I enjoyed our conversation the o...
4	ENTJ	'You're fired.|||That's another silly misconce...

II. METHODOLOGY

Exploring the dataset and pre-processing

- **Selective Word Removal -** As the data set comes from an Internet forum where individuals communicate strictly via written text, some word removal was clearly necessary. For example, there were several instances of data points containing links to websites. We deleted any data points with links to websites since we want our model to generalize to the English language. Next, we deleted so-called "stop words" from the text since we want every word in the data to be as significant as possible (e.g. very common filler words like "a", "the", "or", etc.)

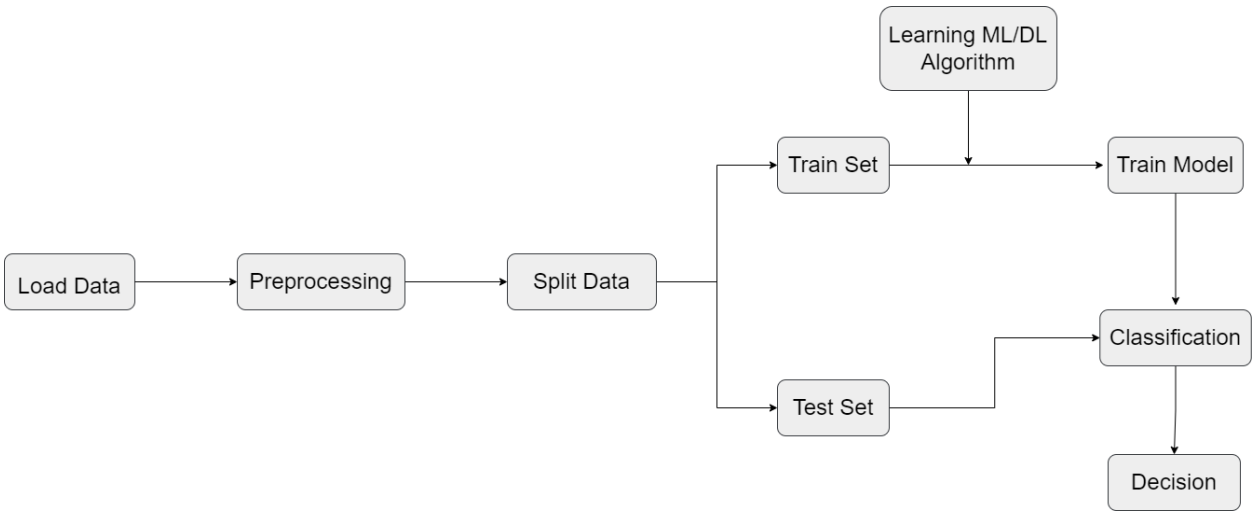
- **Lemmatization** - We lemmatized the text with nltk.stem.WordNetLemmatizer, which means that inflected variants of the same root word were converted to their dictionary form (e.g. "stopping", "stopped", "stops" all become "stop"). We'll be able to take use of the fact that inflected versions of the same word have the same meaning.

- **Tokenization -** The most common terms in the lemmatized text were tokenized. That is, the most popular term was changed to 1, the second most common word to 2, and so on. Any remaining words in the lemmatized text have been eliminated, leaving the text in the form of

integer lists.

Implementation of classification algorithms

End to end Ml-Pipeline was created and applied on the loaded dataset. It is shown below:



- **Random Forest Classifier -** Random Forest Classifiers use boosting ensemble methods to train upon various decision trees and produce aggregated results.It is one of the most used machine learning algorithms. Random forest classifier is used after preprocessing for classification with default parameters and an accuracy of 32.5(approx.) was obtained.

- **LGBMClassifier -** LightGBM is a gradient boosting framework based on decision trees to increase the efficiency of the model and reduce memory usage**.** Lightgbm model was used for training and an accuracy of 57.70(approx.) was obtained.

- **LogisticRegression -** Logistic regression is a process of modeling the probability of a discrete outcome given an input variable. Logistic regression model was used for training and an accuracy of 57.66(approx.) was obtained.

- **Adaboost Classifier** - Adaptive Boosting and is a very popular boosting technique that combines multiple “weak classifiers” into a single “strong classifier”. The Adaboost model was used for training with an accuracy of 27.96(approx.) was obtained.

EVALUATION OF MODELS

All models were trained on 70% of data and accuracy was calculated on testing data. Accuracy of all the models is shown below:

Adaboost Classifier - 27.96

Random Forest Classifier- 32.50

Logistic Classifier - 57.66

LightGBM Classifier - 57.70

Result and Analysis

Different types of model were trained like tree based, SVM, LDA and QDA were trained and tree based accuracy was found to be highest. Tree based LightGBM was giving the highest accuracy.