Myers Briggs Personality Prediction Using Machine Learning Models

Aman Srivastava
M.Tech CSE MT21007
aman21007@iiitd.ac.in

Yukti Goswami
M.Tech CSE MT21109
yukti21109@iiitd.ac.in

Jatin Agarwal
M.Tech CSE MT21032
jatin21032@iiitd.ac.in

Abstract

Personality can be defined as the unique traits that differentiate them from others. It includes their thinking, feeling, behavior, and representing their characteristics. People nowadays share their thoughts, feelings, and information, more specifically their personality traits, on social media platforms more often than anywhere else, making them a source of data generation. Various machine learning models have already been implemented in this field.

This report will present NLP techniques of Feature Extraction like Bag of Words and TF-IDF as baselines models and some other machine and deep learning models to get more accurate predictions. We will also try to predict our personality and verify how precisely our final model is performing, which will interest psychology and private sector organizations.

Subjects: Machine Learning (ML), Deep Learning (DL), Natural Language Processing (NLP).

Keywords: ML (Machine Learning), MBTI (Myers-Briggs Type Indicator), TF-IDF (Term Frequency — Inverse Document Frequency), CV (Count Vectorizer), SVM (Support Vector Machine), KNN (K- Nearest Neighbours), MLP (Multilayer Perceptron)

1 Introduction

"Predicting the Personality of a person on the basis of his/her writing style from the 16 Myers Briggs personality types."

Predicting personality is currently a hot topic and has various applications, including career counseling, candidate screening, and many other business applications. It is a fact that 89 out of 100 fortune companies use this data to earn more profits. Isabel Myers-Briggs and Katherine Briggs designed a Myers-Briggs type indicator instrument called "type table." 16 personality types (as shown in figure 1) are combined and labeled into four dimensions where each dimension represents two personality types as:

- Extroversion-Introversion (E-I)
- Sensation-Intuition (S-N)
- Thinking-Feeling (T-F)
- Judgement-Perception (J-P)

Our dataset will represent these four letter-codes as the form of personality as the target variable.

Figure 1: Myers Briggs 16 personality types

2 Literature Review

The researchers of paper [1] compared various existing machine learning models like SVM, Logistic Regression etc., based on their accuracies. They have collected a dataset via questionnaire among users and pre-processed it. Then they add seven new features in the dataset and find the average number of words per feature. They then analyzed the Pearson correlation between words

per comment and ellipses per comment over the dataset; They found that INFJ, INTP, and ENFP have the highest correlation. Finally, they found that the Random Forest model is more robust and accurate. In paper[2], Researchers found earlier questionnaire approaches as time and money inefficient, so they gathered data from social media and summarised various models and approaches based upon pre-processing techniques and accuracy. The authors of the paper [3] tried to develop a classifier that takes social media posts as input and predicts one of the MBTI types as output. They have pre-processed data by performing tokenization, Lemmatization and removal of URL and unwanted data and implementing RNN models to get the best accurate model. In paper [4], Researchers used NLP techniques to create feature vectors further by combining TF-IDF and word2vec representations. Then, implement three ensemble models and infer that the stack method is more accurate by comparing F1 scores. In paper[5], researchers developed a new model based on Gradient Boosting Framework and showed the accuracy of the XGBoost model outperforms the existing models. In paper [6], researchers have performed character-level TF and word-level TF-IDF on Judging-Perceiving prediction on social media data and found that word-level features have been recommended over character-level features. They have implemented five classifiers and concluded that SVM and LightGBM outperformed other classifiers in the task of MBTI Judging-Perceiving Prediction. They have also analyzed eight datasets and found that the Kaggle dataset is the best among those with the highest F1-Scores.

3 Data Set

	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...

Figure 2: First five rows of dataset

The main dataset[7] is publicly available on kaggle.com containing 8675 rows of data. It has only 2 columns:

- Myers Briggs Type Indicator (MBTI) personality type for a person (e.g ESFP, INTJ etc)
- Last 50 things posted by the person on social media, separated by three pipe lines. This data has been collected from the users through a forum from personalityCafe.com.

3.1 Data Set Analysis

We have performed analysis on this dataset in python and extracted some relevant outcomes like
(1) There exists 16 unique personality types,
(2) INFJ is the most frequently occurring type in the dataset and
(3) all the posts are unique.
We have done graphical analysis by plotting charts like bar graphs, pie charts etc.

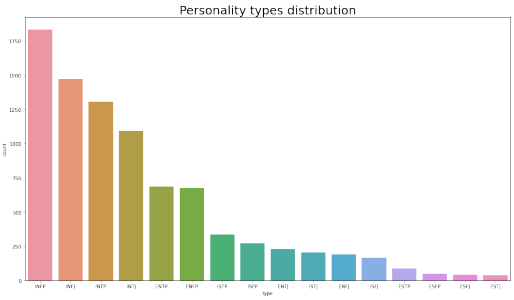

Figure 3: Personality Type Distribution using Bar Graph

As can be seen from the bar graph, INFP is the most frequently occuring personality type and ESTJ is the least frequently occuring type. From this we can infer that people having type INFP are more expressive in their feelings on social media even being introvert and ESTJ are less expressive even being extrovert.

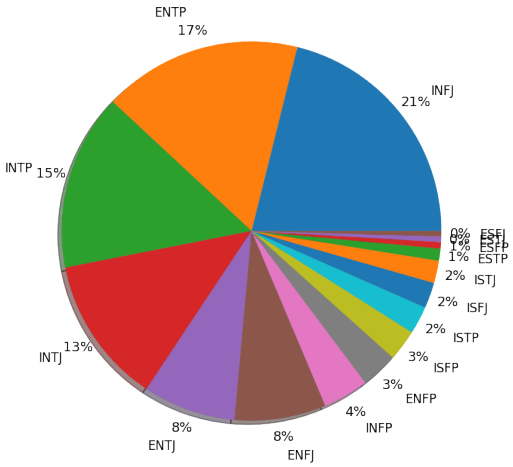

Figure 4: Personality Type Distribution using Pie Chart

We can infer from the above chart that INTP, INFJ, ENTP are the three highest occurring types in the dataset, and we have a class imbalance problem that needs to get handled.

We have also observed that the most common words are mostly the English stop words which are irrelevant for our performing personality predictions.

3.2 Data Preprocessing

This step involves cleaning our dataset by removing unnecessary parts of data that would have no role in the prediction task. We have performed the following stages of data pre-processing:

- **Removal of URLs:** Since URLs are just hyperlinks that do not contain nutritional information, they need to remove them.
- **Removal of End tokens** includes removing tokens like '?', '!',. '' etc.
- **Removing words containing digits:** This means removing alphanumeric words such as U467, abc@9834 etc., that have no role in predictions.
- **Lower casing Words:** It makes all the words in lowercase so that words like 'A' and 'a' can be considered the same.
- **Removing Multiple letters repeating Words:** Words such as 'yayyeeee', 'iloooooveyou', 'happpy' etc., require removal as these have no significance in our analysis.
- **Removing Parentheses and extra spacing:** It will be better to remove them since these are irrelevant.
- **Lemmatization:** Inflected words like 'developed', 'developing' etc., into the single word 'develop' to squeeze word-set.
- **Removing Stop words:** We need to remove words like 'a', 'the', 'to' etc., to cut off from the dataset.

After data preprocessing, we split the dataset into 70-30% proportions to derive the training and test dataset.

4 Feature Extraction Techniques

We have used two feature extraction techniques in Natural Language Processing to reduce the number of features and construct the feature-set that can summarize original features:

4.1 Bag Of Words:

We have used Sklearn's *CountVectorizer()* to implement this NLP model to get the frequency of words in a frequency vector representation present in the dataset.

4.2 TF-IDF:

This technique gives importance to the frequent words that appear in the fewer documents to signify the relevance of each word. We have implemented it through sklearn's *TfidfVectorizer()* by performing fit on the training set and then transforming the result on the test set.

5 Baseline Models

We have chosen SVC and RandomForest as the baseline models where we have implemented CountVectorizer() as the baseline technique. In extension to this, we have performed the TF-IDF technique on similar models and analyzed them based on their F1-Scores. Afterward, we have done it graphical representation between Models and F1-Score as below:

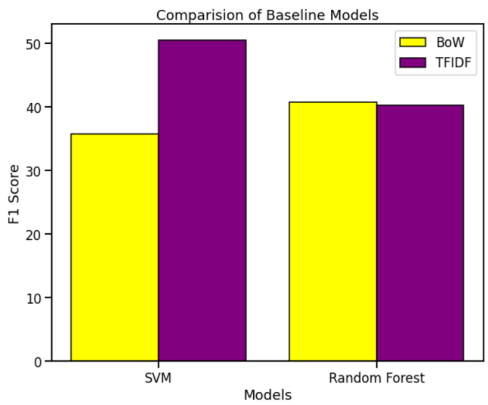

Figure 5: Comparison of Baseline Models

We can infer from the above graph that there is a significant improvement in the SVM classifier after applying the TF-IDF technique compared to the RandomForest classifier.

6 Other ML Models

In the extension of this, We have done a similar implementation on other machine learning models on each feature extraction technique separately as follows:

6.1 XGBoost Classifier:

It implements a Gradient Boosted Decision tree for high performance. We have used sklearn, xgboost *XGBClassifier()*, and calculated F1-Scores.

6.2 CatBoost Classifier:

It provides a Gradient Boosting Technique mainly used for categorical features. We have used catboost *CatBoostClassifier()* with **iterations=50** and **learning rate=0.5.**

6.3 KNN Classifier:

This supervised learning technique has been implemented through sklearn's *KNeighborsClassifier()* with **n_neighbor=200.**

7 Results & Analysis

We have analyzed both models that have implemented pre-mid sem, and post-mid sem on feature extraction techniques BoW and TF-IDF based on their F1-Scores obtained.

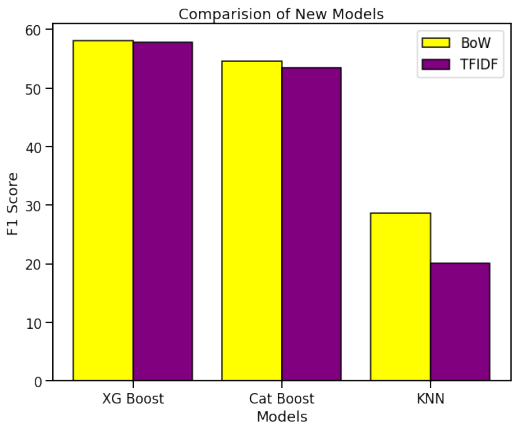

Figure 6: Comparison of New Models

It can be inferred from the above graph that XGBoost outperforms all the other models implemented so far with F1-Scores of **58%** for both BoW and TF-IDF. However, the KNN classifier underperforms, with only **28.71%** for BoW and **20.04%** for TF-IDF.

Support Vector Classifier(SVC) depicts the most excellent optimistic effect of TF-IDF implementation with a positive difference of around **15%** in F1-Scores. On the other hand, KNN classifier still comes under the category of most poor performer with a negative difference in its F1-Score of **8%**.

8 MBTI Across 4-Axis

The performance of all the above ML models in terms of F1 score is around 50% - which is pretty bad. So instead of training the models on all 16 types of personalities, we then tried to teach models across four classifiers as 4 MBTI axes as shown below:

16 Personality types across four axes:
1. Introversion (I) – Extroversion (E)
2. Intuition (N) – Sensing (S)
3. Thinking (T) – Feeling (F)
4. Judging (J) – Perceiving (P)

So, In this way, We have built four target variables I/E, N/S, F/T, P/J, across four dimensions and performed binary classification. The dataset looked as follows:

	type	posts	I/E	N/S	F/T	P/J
0	INFJ	enfp and intj moments sportscenter not top te...	I	N	F	J
1	ENTP	im finding the lack of me in these posts very ...	E	N	T	P
2	INTP	good one of course to which i say i know thats...	I	N	T	P
3	INTJ	dear intp i enjoyed our conversation the other...	I	N	T	J
4	ENTJ	youre fired eostokendot thats another silly mi...	E	N	T	J

Figure 7: New Tweaked Dataset

We then performed the same data pre-processing and then performed TF-IDF on it with *CountVectorizer()* having **max_features=5000** and finally split it along with **stratify** for handling class imbalance and then sent for further processing.

Analysis

We have trained the same five models across 4 MBTI axes, i.e. SVM, Random Forest, XG-Boost, Cat-Boost KNN.

It can be inferred from the below comparison that F1 scores for each model are in the range of 70s-90s, which is a good improvement over our previous methodology on all 16 types of MBTI personalities with highest F1 on N/S and lowest on P/J.

The performances of these models across the 4 MBTI axis are shown below :

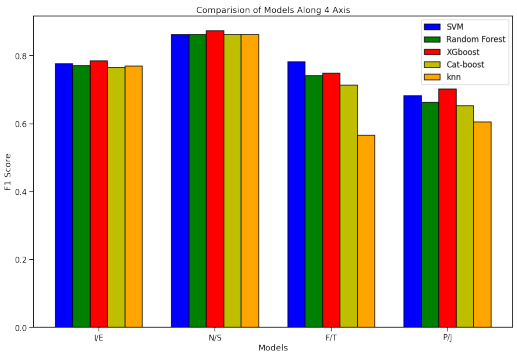

Figure 8: Comparison among 4 MBTI Axis

XGBoost classifier outperforms among all the models as it has the highest average F1-Score **(77.75%)** among all the models while KNN has the least **(70.06%)**.

Here are the average F1-Scores for the above combined models:

- SVM Average F1 score: **77.592%**
- Random Forest Average F1 score: **75.943%**
- XGBoost Average F1 score: **77.75%**
- CatBoost Average F1 score: **74.827%**
- KNN Average F1 score: **70.06%**

9 Deep Learning Model: Multi-Layer Perceptron

Apart from the ML models, we have also tried one Deep learning model called Multilayer Perceptron and analyzed how deep learning performs on this problem.

9.1 Hyper Parameter Tuning

We performed GridSearch on the number of layers the number of neurons on each hidden layer. The plot of hidden layer sizes Vs. F1 scores are shown in figure 9:

In figure 9. 4-[2,1] represents a total of 4 layers, including one input layer, one output layer 2 hidden layers of sizes 2, 1 respectively. Similarly, 5-[3,2,3] represents a total of 5 layers, including one input layer, one output layer, three hidden layers of sizes 3, 2, 3 nodes, respectively.

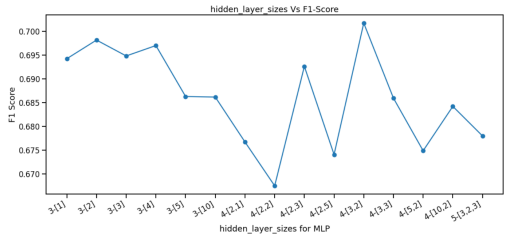

Figure 9: Hyperparameter Tunning in MLP

As we can see from the plot, MLP has performed the best when we have four layers, including one input layer, one output layer, and two hidden layers of sizes 3 and 2, respectively. So we will be using this MLP model for our further analysis of the entire dataset.

9.2 Performance of MLP & Comparison with XGboost:

We have implemented sklearn's MLPClassifier with the best parameters obtained from the hyper-parameter tuning and **max_iter = 200**, activation function as **'relu'** and computed F1-Score. The average F1 score achieved by this model is **71.26%** - worse than XGboost.

The comparison is shown below:

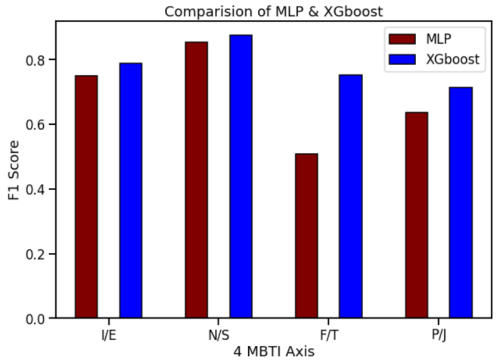

Figure 10: Comparison between MLP XGboost

There is no notable improvement after applying MLP, and still, XGBoost outperforms it with more F1-Scores among all the 4 MBTI axis.

10 Predicting Our Personality Type Using our final Model

We have tried to predict one of our group member's (Jatin Agarwal) personality to verify our results. We passed around ten statements written by him to our model. His actual personality type is INFJ, and XGboost has predicted his personality type as

INFP.

Actual Personality Type: INFJ
Predicted Personality Type: INFP

We can see that our final model, i.e., XGBoost, has performed well along the first 3 MBTI axis but shown deviation in the case of the 4th MBTI axis, i.e., J/F. The reason for this wrong prediction is that the statements that we passed to our model for prediction might contain more words belonging to 'P'-Type than 'J'-Type. Another reason is that the F1 score is the lowest for J/F class, so it has more possibility for wrong prediction than other classes.

11 Conclusion

This work has demonstrated that there is no significant difference in the F1-Scores between the BagOfWords and TF-IDF implementations for most of the models. However, We have seen that SVC has the most optimistic impact on TF-IDF implementation, while KNN has the most pessimistic impact. XGBoost classifier has the highest F1-Score of 58% among all.

After Tweaking the dataset by adding four new columns, training four different classifiers on each new feature, and combining them, it has been observed that we have achieved significant improvement in the F1-Scores in which XGBoost outperforms among all.

We have also implemented one deep learning model, MLP and hyperparameter tuning, to find the best parameter and then compare it with XGBoost, but still, XGBoost exceeds both.

12 Contribution of each member in the group

The project is a collective effort of each member of the team. We worked together and monitored every component equally and simultaneously.

The primary responsibilities of each group member are as follows :

Yukti Goswami *(M.Tech CSE MT21109)* : Creation, editing, and designing of content and presentation, reports, and research papers, performed MLP modeling with Aman, and had a significant hand in accomplishing the documentation-related part.

Jatin Agarwal *(M.Tech CSE MT21032)* : Worked upon implementing TF-IDF and BoW techniques and implemented various models and comparison charts. Apart from that, I performed a literature review, Baseline modeling, Blog, and Report formations.

Aman Srivastava *(M.Tech CSE MT21007)* : Implementation of the baseline while managing and keeping everything in place and up-to-date. Also implemented various ML models across 4 MBTI axes and one Deep Learning model and compared the results.

13 References

[1] Improving Intelligent Personality Prediction using Myers-Briggs Type Indicator and Random Forest Classifier

[2] Personality Prediction from Social Media Text: An Overview

[3] Predicting Myers-Briggs Type Indicator with Text Classification

[4] PERSONALITY IDENTIFICATION BASED ON MBTI DIMENSIONS USING NATURAL LANGUAGE PROCESSING

[5] Machine Learning Approach to Personality Type Prediction Based on the Myers–Briggs Type Indicator

[6] Predicting judging- perceiving of Myers-Briggs Type Indicator (MBTI) in online social forum

[7] Dataset Linkhttps://www.kaggle.com/datasnaek/mbti-type

14 Google Drive Link for the dataset and Code File

https://drive.google.com/drive/folders/1YaYGPTeHqIg8axlIRWKDJWsZkC17FfSN?usp=sharing