ISSN: 2321-2152
IJMECE
International Journal of modern electronics and communication engineering

E-Mail
editor.ijmece@gmail.com
editor@ijmece.com

www.ijmece.com

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

TEXT BASED PERSONALITY PREDICTION USING MACHINE LEARNING

Mr.Raghavendra M I
BE M.Tech(Ph.D) Assistant Professor
Department of CSE (Data Science)
Sphoorthy Engineering College (JNTUH)
Hyderabad, India
Email: rmichangi@gmail.com

V.Rishitha
Department of CSE (Data Science)
Sphoorthy Engineering College (JNTUH)
Hyderabad, India.
Email: Vattemrishi@gmail.com

P.Varshitha
Department of CSE (Data Science)
Sphoorthy Engineering College (JNTUH)
Hyderabad, India.
Email: varshithareddy3703@gmail.com

T.Shanmukh Koushik
Department of CSE (Data Science)
Sphoorthy Engineering College(JNTUH)
Hyderabad, India.
Email: tadurikoushik66@gmail.com

G.Uma Shankar
Department of CSE (Data Science)
Sphoorthy Engineering College(JNTUH)
Hyderabad, India.
Email:shivagopagoni10@gmail.com

ABSTRACT
Personality of a person is an important aspect and reveals how the person thinks, how he speaks and how he behaves.Use of social media has tremendously increased worldwide, leading to the generation of voluminous useful data which can be used to extract knowledge about user patterns and behaviour. Thus, the use of social media to predict personality is fruitful as we can get a dataset with good amount of data. The purpose of this project is to take textual data as an input from the user and then run the trained machine learning model on this data to predict four personality traits which are Introversion Vs Extroversion, Sensing Vs Intuition, Thinking Vs Feeling, Judging Vs Perceiving. Processing of large textual data is to be done using Natural Language Processing (NLP) techniques with the help of NLTK libraries to process and categorize the data. For more exploration of the personality from text, preprocessing techniques including tokenization, lemmatization, word stemming, stop words elimination are also exploited. XGBoost is selected for its accuracy and robustness.This project is capable of predicting the personality type of a person based on text the user has written.

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

INTRODUCTION

Personality is an amalgamation of a varied set of features and characteristics of an individual that effects their state of mind, attitude towards others, day to day activities and line of thoughts and reasoning. All the text of an individual that is available on social media sites provide us an opportunity to recognize personality traits of that individual. Personality is what sets people apart from one another, thus it's a vital factor to consider. Personality is an important part of human existence. The study of personality falls under the umbrella of psychological research. Personality is made up of factors such as a person's thoughts, feelings, and conduct, all of which change throughout time. People are divided into different groups of personality types, hence personality prediction is considered as a classification problem in computer science. Different sorts of personality classifications can be determined using a variety of psychological tests. MBTI, Big Five, and DISC are all popular personality assessments. One of the most well-known and extensively used personality tests or descriptions is the Myers-Briggs Type Indicator (MBTI).It defines how people behave and interact with the world around them with four binary categories and 16 total types. They are as follows: Introversion vs Extroversion, Sensing vs Intuition, Thinking vs Feeling, Judging vs Perceiving.

The main objective is to build an application where users can predict one's MBTI personality type from one of their social media posts or any text given by the user. Our algorithm takes excerpts of text as input and predicts a string of four characters where each character determines a personality trait, total of sixteen personality types are possible as an output. Processing of large textual data is to be done using Natural Language Processing (NLP) techniques with the help of NLTK libraries to process and categorize the data. We will use evaluation metrics such as Geometric Mean, f1score, average precision recall score to evaluate the models and select the one that is most effective and performs the best. The output is a string of four characters where each character determines a personality trait so a total of 16 personality types are possible.

EXISTING SYSTEM:

The primary objective of this project is to develop an intelligent application capable of predicting an individual's MBTI (Myers-Briggs Type Indicator) personality type based on a sample of their written text. The input to the system can be a post from any social media platform or any arbitrary piece of text provided by the user. The predicted output is a string consisting of four characters, each representing one dimension of the MBTI personality spectrum—namely, Introversion (I) vs. Extraversion (E), Sensing (S) vs. Intuition (N), Thinking (T) vs. Feeling (F), and Judging (J) vs. Perceiving (P). Since there are four dichotomies with two possible values each, the model is expected to predict one of the 16 possible personality types (e.g., INFP, ESTJ, etc.).

To accomplish this task, the application will utilize advanced Natural Language Processing (NLP) techniques to interpret and analyze the semantics and patterns within the text. The Natural Language Toolkit (NLTK) library in Python will play a pivotal role in text preprocessing, including tasks such as tokenization, lemmatization, stopword removal, and syntactic parsing. These preprocessing steps will prepare the text for feature extraction, allowing the model to focus on the most relevant linguistic indicators of personality.

The processed data will be fed into machine learning models trained to classify the input into one of the 16 MBTI personality types. To ensure the

robustness and accuracy of the system, various evaluation metrics will be employed. These will include the Geometric Mean, which is particularly useful for handling imbalanced data, the F1 score to balance precision and recall, and the Average Precision-Recall score, which provides a more comprehensive view of model performance in cases of class imbalance. Through careful experimentation and validation, the most effective model will be selected based on its performance across these metrics.

Ultimately, this project aims to provide an engaging and insightful tool for users who are curious about their personality type or wish to analyze the personalities of others through written content. The integration of NLP and machine learning for psychological profiling has the potential to open new avenues for understanding human behavior in the digital age.

ADVANTAGES:

1.Business

- Helps individuals and businesses improve communication and team-building.
- Useful for developing customized services or products based on personality traits.

2.Friend Recommendation in Social Media

- Enhances friend suggestion systems by considering personality similarities, not just mutual connections or interests.

3.Movie/Music Recommendation

- Recommends entertainment content by matching users with similar personality profiles.
- Uses previous preferences of one user to guide recommendations for others with similar traits.

5.Marketing and Advertising

- Enables creation of personality-oriented user interfaces and content.
- Helps marketers build better rapport with users by tailoring advertisements and messages to personality types.

6.Human Resources and Recruitment

- Streamlines initial screening by matching personality traits to job roles.
- Identifies candidates whose personality aligns with team dynamics or company culture.

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

DISADVANTAGES:

1.Data Privacy and Ethical Concerns

- Analyzing personal text, especially from social media, raises significant privacy issues. There's a risk of misuse in profiling, discrimination, or manipulation without user consent.

2.Limited Accuracy

- Personality is complex and cannot be fully understood through text alone.
- The predictions may lack precision, especially with short or ambiguous text samples.

3.Bias in Data

- If the training dataset is skewed (e.g., more INFPs than ISTJs), models may become biased.
- Biases in language use due to culture, age, or region can affect prediction accuracy.

4. **Oversimplification of Human Personality**

- MBTI is a simplified model of personality and lacks empirical support in some psychological communities.
- Reducing people to one of 16 types can ignore the nuances of personality.

4.Dependency on Language Style

- The model assumes that language style directly reflects personality, which may not always hold true.
- Some individuals may consciously or unconsciously mask their true personality in writing.

5.Context Ignorance

- Text-based systems may not accurately interpret sarcasm, irony, or context-dependent statements.
- This can lead to incorrect predictions, especially in informal or mixed-language texts.

PROPOSED SYSTEM:

- In this proposed work, we aim to predict four core personality traits associated with the Myers-Briggs Type Indicator (MBTI) model. These traits include Judging vs. Perceiving, Intuition vs. Sensing, Feeling vs. Thinking, and Introversion vs. Extraversion. The system classifies each of these dichotomies independently, ultimately generating a four-letter MBTI personality type as output. This approach allows for a more fine-grained and interpretable personality classification based on user-generated textual data.
- To train and evaluate our models, we utilize a publicly available dataset from Kaggle, which contains social media posts annotated with corresponding MBTI personality types. However, one of the key challenges encountered in this dataset is class imbalance—some personality types are heavily overrepresented while others appear infrequently. This skewness affects model training and often leads to biased predictions that favor the majority classes.
- To address this issue, we apply resampling techniques, particularly oversampling, to artificially balance the dataset. Oversampling works by increasing the number of instances in underrepresented classes, thereby providing the model with a more balanced view during training. This leads to a notable improvement in model accuracy and ensures better generalization across all 16 personality types.
- These steps are essential for reducing noise in the text data and extracting meaningful patterns that reflect personality traits. The cleaned and vectorized data is more suitable for training machine learning models.
- For model development, we implement and compare multiple machine learning algorithms, including Logistic Regression, Support Vector Machines (SVM), Random Forest, and others. After extensive evaluation using metrics like accuracy, F1 score, and precision-recall, we determine that the most effective model is
- XGBoost (Extreme Gradient Boosting). XGBoost is known for its high computational efficiency, robust performance on imbalanced datasets, and ability to handle sparse data like text vectors. Its ensemble-based boosting approach builds a strong predictive model by combining the outputs of many weak learners.
- Overall, the combination of balanced data through oversampling, comprehensive NLP preprocessing, and robust model selection with XGBoost significantly enhances the reliability and effectiveness of personality prediction from text. This approach lays a solid foundation for building scalable, real-world applications that can analyze personality traits from various text sources.

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

MODULES:

Exploratory Data Analysis Module

Exploratory Data Analysis Module The dataset is quite skewed and is not uniformly distributed among the 16 personality types. For example, the most common label, INFP, occurs 89.796 times whereas the least frequent, ISFJ, only occurs 8,121 times. When training on the data in its original form, the model tends to overfit on the predominant type(s) while underperforming on the others since the classes are heavily imbalanced.

Data Preprocessing Module

Different pre-processing techniques have been exploited for more exploration of personality from the text. The techniques include: Cleaning the Data
• Posts will be converted into lower case.
• ||| and punctuations will be replaced by spaces.
• Links and Emails will be dropped.
• Words with one to two character lengths will be dropped. The cleaned data that has been generated after executing the above steps is Lemmitized using NLTK WordNet Lemmatizer. Word stemming is performed. Stop Words dropped at this stage. The result of these steps is stored as "clean posts".
• Input: Raw Kaggle MBTI dataset • Output: Cleaned Dataset

Feature Extraction Module

Counting: Per user average counts are taken for number of question marks, exclamations, colons, emojis, words, unique words, upper case words, links, ellipses and images. These counts are our additional features for the machine learning models.
Count Vectorization : It is a great tool provided by the scikit-learn library in Python. It is used to transform a given text into a vector on the basis of the frequency (count) of each word that occurs in the entire text. This is helpful when we have multiple such texts, and we wish to convert.
CountVectorizer creates a matrix in which each unique word is represented by a column of the matrix, and each text sample from the document is a row in the matrix. The value of each cell is nothing but the count of the word in that particular text sample. Count vectorization makes it easy for text data to be used directly in machine learning and deep learning models such as text classification.
One Hot Encoding : One hot encoding is a technique that we use to represent categorical variables as numerical values in a machine learning model. It allows the use of categorical variables in models that require numerical input. It can improve model performance by providing more information to the model about the categorical variable. It can help to avoid the problem of ordinality, which can occur when a categorical variable has a natural ordering.

Training Module

The output of Module-1, which is the cleaned dataset, is taken as input in this module. The model is trained on this data and saved to the local system. We have built, trained and tested various models on the dataset. Models are: Logistic Regression Random Forest Support Vector Machine XG Boost
• Input: Cleaned Dataset
• Output: Trained Model One of the trained models is considered to predict the results. Here, the XGBoost model gives the best accuracy. The input is given through text and the outputs are predicted.
• Input: Trained XGBoost Model
• Output: Predicted outputs

ALGORITHMS:

Logistic Regression

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.

Logistic regression predicts the output of a

categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, True or False, etc, but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1. In Logistic regression, instead of fitting a regression line, we fit an "S" shaped logistic function, which predicts two maximum values (0 or 1).

Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.

Random Forest Algorithm

Random Forest Algorithm Random forest is a commonly-used machine learning algorithm trademarked by Leo Breiman and Adele Cutler, which combines the output of multiple decision trees to reach a single result. Its ease of use and flexibility have fueled its adoption, as it handles both classification and regression problems. Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning which is a process of combining multiple classifiers to

Support Vector Machine Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

SVM chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine. Effective in high dimensional cases Its memory efficient as it uses a subset of training points in the decision function called support vector.

Different kernel functions can be specified for the decision functions and its possible to specify custom kernels SVM algorithm is not suitable for large data sets. In cases where the number of features for each data point exceeds the number of training data samples, the SVM will underperform.

XG Boost

XGBoost is an optimized distributed gradient boosting library designed for efficient and scalable training of machine learning models. It is an ensemble learning method that combines the predictions of multiple weak models to produce a stronger prediction. It implements machine learning algorithms under the Gradient Boosting framework. It is a scalable, distributed gradient-boosted decision tree (GBDT) machine learning library. It provides parallel tree boosting and is the leading machine learning library for regression, classification, and ranking problems. XGBoost stands for “Extreme Gradient Boosting” and it has become one of the most popular and widely used machine learning algorithms due to its ability to handle large datasets and its ability to achieve state-of-the-art performance in many machine learning tasks such as classification and regression. One of the key features of XGBoost is its efficient handling of missing values, which allows it to handle real-world data with missing values without requiring significant pre-processing.

Encoding

One-Hot Encoding One-Hot Encoding is a popular technique for treating categorical

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

variables. It simply creates additional features based on the number of unique values in the categorical feature. Every unique value in the category will be added as a feature. One-hot encoding is the representation of categorical variables as binary vectors. Vectorization Count Vectorization CountVectorizer is used to convert a collection of text documents to a vector of term/token counts.

It denotes frequency of words occurring in the document.

RESULT :

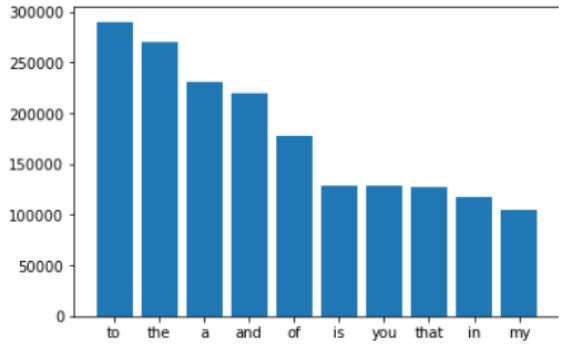
The implementation of the personality prediction system yielded promising outcomes. After extensive preprocessing and feature extraction using Natural Language Processing (NLP) techniques, multiple machine learning models were trained and evaluated on the MBTI dataset obtained from Kaggle. The preprocessing steps, such as tokenization, lemmatization, stemming, stop word removal, and count vectorization, significantly enhanced the quality of the textual data and improved model performance by reducing noise and redundancy.

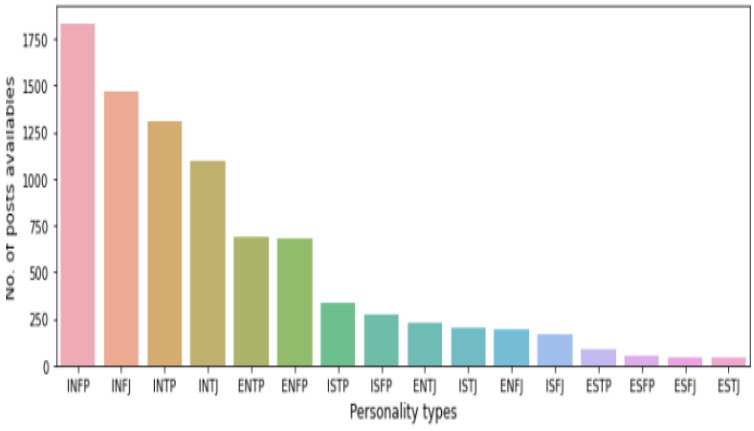
To address the imbalance in class distribution within the dataset, oversampling techniques were applied. This approach helped in balancing the representation of all personality types during training, thereby improving the generalization capability of the models. The dataset was split into different train-test ratios (60:40, 70:30, and 80:20) for thorough evaluation.

Among all the models tested, including Logistic Regression, Support Vector Machine (SVM), and Random Forest, the XGBoost classifier consistently demonstrated superior performance across all four MBTI personality trait dimensions—Introversion vs. Extroversion, Sensing vs. Intuition, Thinking vs. Feeling, and Judging vs. Perceiving. XGBoost achieved the highest accuracy, F1-score, and precision-recall scores, making it the most reliable model for this task.

The model's output was validated using real-world test cases, where user-input text samples were analyzed and the corresponding MBTI personality type was predicted. The predictions closely matched the known labels, showcasing the model's ability to interpret and classify personality traits accurately from textual data.

Overall, the results validate the feasibility of using machine learning and NLP for personality prediction. The system achieved a robust performance, particularly when powered by XGBoost, and it sets a strong foundation for deploying personality-aware applications in various domains such as recruitment, marketing, recommendation systems, and personal development.





ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

```
        return final_type(predictions)

    # === Step 9: Example usage ===
    if __name__ == "__main__":
        test_input = """i am an angry person ."""
        result = predict_personality(test_input)
        print(f"🧠 Predicted MBTI Type: {result}")
```

```
[3]
...
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\sahithi\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\sahithi\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\sahithi\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
Introvert vs Extrovert Accuracy: 86.97%
Sensing vs Intuition Accuracy: 91.24%
Thinking vs Feeling Accuracy: 84.50%
Perceiving vs Judging Accuracy: 80.92%
✅ All models and CountVectorizer saved.
🧠 Predicted MBTI Type: ESFJ
```

CONCLUSION:

Personality is a fundamental psychological parameter that plays a crucial role in distinguishing individuals based on their thoughts, behaviors, emotions, and interactions with others. It influences how people perceive the world, make decisions, and engage with their surroundings. In recent years, the ability to predict personality using written text has emerged as a promising and innovative approach, especially in the age of digital communication and social media.

Unlike traditional personality assessment methods, which rely on time-consuming self-report questionnaires or structured interviews, text-based personality prediction offers a non-intrusive and efficient alternative. This approach leverages the natural linguistic expression of individuals, capturing nuances in language usage, tone, and writing style. As users generate content in the form of tweets, posts, blogs, and messages, this textual data becomes a rich source of behavioral signals that can be mined for personality insights. This not only reduces the burden on the user but also enhances credibility by avoiding potential biases introduced by self-reporting.

The core of this system lies in the application of machine learning algorithms combined with Natural Language Processing (NLP) techniques. By extracting and analyzing linguistic features from text—such as word choice, sentence structure, punctuation usage, and frequency of specific terms—models can be trained to map these features to established personality frameworks like the Myers-Briggs Type Indicator (MBTI).

To ensure the reliability and robustness of the system, multiple machine learning models will be trained and evaluated. These include algorithms such as Logistic Regression, Support Vector Machine (SVM), Random Forest, and XGBoost. Each model will be tested with various combinations of feature vectors obtained from preprocessing techniques like tokenization, lemmatization, stemming, and vectorization. The performance of these models will be assessed using a set of evaluation metrics, including accuracy, F1-score, precision, recall, and geometric mean.

Ultimately, the model that demonstrates the best overall performance across these metrics will be selected as the final classifier for the system. This data-driven and systematic approach ensures that the chosen model not only performs well on training data but also generalizes effectively to new, unseen text inputs. The end goal is to develop a robust, scalable, and accurate personality prediction tool that can be applied in real-world scenarios ranging from recruitment and education to social networking and personalized marketing.

REFERENCES:

[1] Aditi.V.Kunte, Suja Panicker, Using textual data for Personality Prediction: A Machine Learning Approach ,IEEE Access , 2019

[2] Shristhi Chaudary, Ritu Singh, Syed Tausif Hasan and Ms.Inderdeep Kaur, A Comparative Study of Different Classifiers for Myers-Brigg Personality Prediction Model, IRJET 2018.

[3] Brandon Cui, Calvin Qi, Survey Analysis of Machine Learning Methods for Natural Language Processing for MBTI Personality Type Prediction, Stanford,2018

[4] S. Bharadwaj, S. Sridhar, R. Choudhary and R. Srinath, "Persona Traits Identification based on Myers-Briggs Type Indicator(MBTI) - A Text Classification Approach," 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Bangalore, 2018, pp. 1076-1082

[5] Pavan Kumar K. N., Marina L. Gavrilova "Latent Personality Traits Assessment From Social Network Activity Using Contextual Language Embedding"

[6] M. Gjurković and J. Šnajder, "Reddit: A Gold Mine for Personality Prediction." In Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media, pp. 87-97, 2018.

[7] B. Plank, and D. Hovy, "Personality traits on twitter-or-how to get 1,500 personality tests in a week." In Proceedings of the 6th Workshop on Computational Approaches Subctivity, Sentiment and Social Media Analysis, pp. 92-98, 2015.

[8] B. Y. Pratama and R. Sarno, "Personality classification based on Twitter text using Naive Bayes, KNN and SVM," 2015 International Conference on Data and Software Engineering (ICODSE), Yogyakarta, 2015, pp. 170-174.

[9] N. Majumder, S. Poria, A. Gelbukh and E. Cambria, "Deep LearningBased Document Modeling for Personality Detection from Text," in IEEE Intelligent Systems, vol. 32, no. 2, pp. 74-79, Mar.-Apr. 2017.

[10] I. B. Myers, "The Myers-Briggs Type Indicator: Manual", 1962