Personality Forecasting with ML and AI: Model Adaptation

Minor project report submitted in partial fulfillment of the requirement for the degree of Bachelor of Technology in Computer Science Engineering

By

Prachi Varshney – 211288

Isha - 211293

UNDER THE SUPERVISON OF:

Mr. Ramesh Narwal

(Assistant Professor)

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology,

Waknaghat-173234

Solan, Himachal Pradesh

TABLE OF CONTENT:

CERTIFICATE

ACKNOWLEGEMENT

We want to express our heartfelt gratitude to all those who have contributed to the successful completion of this project on "Personality Identification using Machine Learning Algorithms and Big Five Model."

First and foremost, we would like to express our sincere appreciation to my supervisor Mr. Ramesh Narwal (Assistant Professor), for their invaluable guidance, support and encouragement throughout this endeavor. Their expertise and insightful feedback have been instrumental in shaping the direction of this project and enhancing its quality. Additionally, we really grateful and wish to express our profound indebtedness to Supervisor Mr. Hari Singh Rawat (Assistant Professor), Department of CSE, Jaypee University of Information Technology, Waknaghat. Their deep knowledge and keen interest in the field of Personality Prediction provided the necessary guidance to carry out this project.

Furthermore, we would like to acknowledge the contributions of [Dataset Source] , whose dataset formed the basis of our analysis . Their patience understanding and motivation have been a constant source of strength.

We would also generously welcome each one of those individuals who have helped our straightforwardly or indirectly in making this project a success. In this unique situation, we might want to thank the various staff members, both educating and non-instructing, who have provided their convenient help and facilitated our undertaking.

Finally, we must acknowledge with due respect the constant support and patience of our parents.

Thank you to everyone who has played a part, no matter how big or small, in bringing this project to fruition. Your support and collaboration have been truly appreciated.

Warm regards,

Prachi Varshney – 211288

Isha - 211293

ABSTRACT:

This project delves into the realm of personality prediction using machine learning (ML) and artificial intelligence (AI) techniques, with a focus on the Myers-Briggs Type Indicator (MBTI) method. The primary objectives encompass preprocessing a Twitter dataset, training and testing five ML models—Naive Bayes, Support Vector Machine (SVM), XGBoost, Random Forest, and Decision Tree—and evaluating their performance metrics such as accuracy, F1 score, etc. Furthermore, the study aims to identify the most effective model among the five and apply interpretability techniques like LIME and SHAP to gain insights into the model's decision-making process.

The motivation behind this project lies in the growing significance of understanding human behavior and personality traits in various domains such as psychology, marketing, recruitment, and personalized recommendation systems. Social media platforms like Twitter offer a wealth of user-generated data that can provide valuable insights into individual personalities, preferences, and behaviors. By leveraging ML and AI techniques, it becomes possible to extract meaningful patterns and predict personality traits from such data, thereby enabling personalized services, targeted interventions, and improved user experiences.

The preprocessing phase involves meticulously cleaning and transforming the raw Twitter data to prepare it for ML model training. Techniques such as tokenization, stop-word removal, and stemming are employed to enhance the quality of the dataset. Additionally, feature engineering methods are applied to extract meaningful features from the text data, aligning with the principles of the MBTI method.

Five popular ML models are selected and trained using the preprocessed Twitter dataset. The models are evaluated based on various performance metrics including accuracy, F1 score, precision, and recall. This comparative analysis provides insights into the strengths and weaknesses of each model in predicting personality traits from Twitter data.

In the real world, personality prediction holds immense potential across various domains. In psychology and counseling, it can aid in understanding individuals' cognitive and emotional processes, facilitating personalized therapy and intervention strategies. In marketing and advertising, it can enable targeted campaigns and product recommendations tailored to consumers' personality profiles. In recruitment, it can assist in identifying candidates whose personality traits align with organizational culture and job requirements. Overall, personality prediction using ML and AI techniques offers numerous practical applications for enhancing decision-making processes and improving user experiences across diverse domains.

Chapter 1: INTRODUCTION

1.1 Introduction

Understanding human personality traits plays a pivotal role across various domains such as psychology, marketing, recruitment, and recommendation systems. With the proliferation of social media platforms like Twitter, there exists a vast repository of user-generated data that can offer insights into individual personalities. This project endeavors to harness the power of machine learning (ML) and artificial intelligence (AI) techniques to predict personality traits from Twitter data, using the Myers-Briggs Type Indicator (MBTI) method as a framework.

1.2 Objective

The objective of this project is to develop a machine learning model capable of predicting personality types based on text data extracted from Twitter posts. Specifically, the project aims to achieve the following goals:

1. Data Preprocessing: Preprocess the Twitter dataset to clean and transform the raw text data into a format suitable for machine learning model training. This includes tasks such as lowercasing, removing special characters, URLs, mentions, and hashtags, as well as tokenization and lemmatization.

2. Model Training and Evaluation: Train and test five different machine learning models—Naive Bayes, Support Vector Machine (SVM), Decision Tree, Random Forest, and XGBoost—using the preprocessed Twitter data. Evaluate the performance of each model using metrics such as accuracy, precision, recall, F1-score, and ROC-AUC score.

3. Identify the Best Model: Compare the performance of the five models and identify the most effective one for personality prediction based on the evaluation metrics. Select the best-performing model for further analysis and interpretation.

4. Interpretability Analysis: Apply interpretability techniques such as LIME (Local Interpretable Model-Agnostic Explanations) and SHAP (SHapley Additive exPlanations) to gain insights into the decision-making process of the selected model. This step aims to enhance the transparency and trustworthiness of the predictive model by providing explanations for individual predictions.

5. Flask Interface Development: Develop a user-friendly web interface using Flask, where users can input a Twitter ID. The interface will retrieve the user's posts, analyze them using the trained machine learning

model, and provide insights into the user's personality type.

1.3 Motivation

The motivation behind this project stems from the growing importance of understanding human behavior and personality traits in various real-world applications. Personality prediction not only aids in psychological research and counseling but also has practical implications in marketing, recruitment, and personalized recommendation systems. By leveraging ML and AI techniques, it becomes feasible to derive valuable insights from social media data, thereby enabling tailored interventions, targeted marketing strategies, and improved user experiences.

1.4 Libraries Used:

The code is written in Python, a widely-used programming language known for its simplicity and readability. Let's break down the libraries and modules you've imported and used in your project:

1. Pandas: Used for data manipulation and analysis. It provides data structures like Data Frame and Series, making it easy to work with structured data.

2. NumPy: Known for numerical computing capabilities, it provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays.

3. Seaborn: Built on top of Matplotlib, Seaborn is used for creating visually appealing statistical graphics. It provides a high-level interface for drawing attractive and informative statistical graphics.

4. Matplotlib: A versatile plotting library for Python, Matplotlib enables you to create a wide variety of static, interactive, and animated visualizations.

5. re (Regular Expressions): Python's built-in module for working with regular expressions, allowing you to search, match, and manipulate text using patterns.

6. NLTK (Natural Language Toolkit): A comprehensive library for natural language processing (NLP) tasks, including tokenization, stemming, lemmatization, and more.

7. WordCloud: Specifically designed for creating word clouds, a visual representation of text data where the size of each word indicates its frequency or importance.

8. Counter: Part of the collections module, Counter is used for counting the occurrences of elements in a list or other iterable.

9. Contractions: A Python library for expanding contractions in text, converting words like "can't" to "cannot".

10. Flask: A lightweight web application framework used for building web applications and APIs in Python.

11. Pickle: A module for serializing and deserializing Python objects, allowing you to save trained models to disk and load them later.

12. Scikit-learn (sklearn): A powerful machine learning library that provides simple and efficient tools for data mining and data analysis. It includes various algorithms for classification, regression, clustering, dimensionality reduction, and more.

13. XGBoost: An optimized distributed gradient boosting library designed for efficient and accurate large-scale machine learning tasks.

14. Imbalanced-learn (imblearn): A library for tackling the problem of imbalanced datasets in machine learning. It provides methods for oversampling, undersampling, and combining sampling strategies to address class imbalance.

These libraries and modules collectively provide a robust toolkit for preprocessing and analyzing text data, training machine learning models, evaluating model performance, and deploying the final model using Flask for web-based applications.

1.5 Technical Requirements :

1. Python Environment: Ensure the project is developed using Python programming language, leveraging its extensive libraries and frameworks for data preprocessing, machine learning, and web development.

2. IDE or Text Editor: Utilize a suitable integrated development environment (IDE) or text editor for

Python development, such as PyCharm, Visual Studio Code, or Jupyter Notebook, to facilitate efficient coding and debugging processes.

3. Data Handling Libraries: Employ Pandas for efficient data manipulation and analysis, allowing for seamless handling of structured data in the form of DataFrames and Series.

4. Numerical Computing: Harness the power of NumPy for numerical computing tasks, enabling efficient handling of large arrays and matrices, along with mathematical operations.

5. Visualization Tools: Utilize Matplotlib and Seaborn for data visualization, enabling the creation of informative and visually appealing plots, charts, and graphs to illustrate insights derived from the data.

6. Text Processing Libraries: Leverage NLTK for natural language processing (NLP) tasks such as tokenization, stemming, lemmatization, and stopword removal, facilitating the preprocessing of textual data extracted from Twitter posts.

7. Machine Learning Frameworks: Employ scikit-learn (sklearn) for implementing machine learning algorithms, including classification models such as Naive Bayes, Support Vector Machine (SVM), Decision Tree, Random Forest, and XGBoost.

8. Imbalanced Data Handling: Address the challenge of imbalanced datasets using imbalanced-learn (imblearn) library, which provides techniques for oversampling, undersampling, and combining sampling strategies to mitigate class imbalance.

9. Model Interpretability Techniques: Apply LIME (Local Interpretable Model-Agnostic Explanations) and SHAP (SHapley Additive exPlanations) for model interpretability, enabling the understanding of individual predictions made by the machine learning model.
10. Serialization and Model Persistence: Utilize the pickle module for serializing trained machine learning models to disk, enabling model persistence for future use without retraining. This ensures that the trained models can be loaded and used efficiently in the Flask application.

By adhering to these technical requirements, the project can be developed effectively, ensuring robustness, scalability, and maintainability of the machine learning model and the web interface.

Chapter 2: Feasibility Study, Requirements Analysis and Design

2.1 Feasibility Study

2.1.1 Problem Definition

The project aims to develop an intelligent personality classification system capable of analyzing written text to predict an individual's Myers-Briggs Type Indicator (MBTI) personality type. Leveraging natural language processing (NLP) techniques and machine learning algorithms, the system categorizes users into one of the 16 MBTI personality types, encompassing Extraversion (E) vs. Introversion (I), Sensing (S) vs. Intuition (N), Thinking (T) vs. Feeling (F), and Judging (J) vs. Perceiving (P). The system's primary objectives include building a robust dataset containing diverse textual data labeled with MBTI types, preprocessing the text to extract relevant features, training and fine-tuning machine learning models for accurate personality prediction, and implementing user-friendly interfaces for easy access and interpretation of results. Furthermore, the project incorporates interpretability techniques such as Local Interpretable Model-agnostic Explanations (LIME) and SHAP (SHapley Additive exPlanations) to provide users with insights into the model's decision-making process, enhancing transparency, trust, and understanding of the classification outcomes. The system's expected outcome includes empowering users to gain valuable insights into their personality traits, fostering self-awareness, personal growth, and effective communication in both personal and professional contexts.

2.1.2 Problem Analysis:

Problem Analysis:

1. Data Acquisition and Preparation:

   1.1. Challenge: Gathering a diverse and representative dataset of textual posts labeled with MBTI types may be challenging due to the subjective nature of personality classification.

   1.2. Approach: Utilize web scraping techniques to collect user-generated content from social media platforms, forums, and other online sources. Implement data cleaning procedures to handle noise, inconsistencies, and missing values in the dataset.

2. Feature Extraction and Engineering:

   2.1. Challenge: Transforming raw text data into meaningful features that capture linguistic patterns and nuances relevant to personality traits.

   2.2. Approach: Employ NLP techniques such as tokenization, lemmatization, and TF-IDF (Term

Frequency-Inverse Document Frequency) vectorization to preprocess and extract features from the textual data. Explore techniques like n-grams and word embeddings to capture contextual information.

3. Model Selection and Training:

   3.1. Challenge: Identifying suitable machine learning algorithms and fine-tuning hyperparameters to develop accurate and reliable personality classification models.

   3.2. Approach: Experiment with various classification algorithms such as Naive Bayes, Support Vector Machines (SVM), decision trees, random forests, and gradient boosting methods like XGBoost. Employ cross-validation and grid search techniques to optimize model performance.

4. Interpretability and Explainability:

   4.1. Challenge: Ensuring transparency and interpretability of the classification models to provide users with meaningful insights into the personality prediction process.

   4.2. Approach: Integrate interpretability techniques like LIME and SHAP to generate local explanations for individual predictions, highlighting the most influential features and their impact on the model's decisions. Visualize model explanations through interactive dashboards or intuitive graphical interfaces.

5. User Interface Design and Accessibility:

   5.1. Challenge: Designing a user-friendly interface that enables users to interact with the system, input their text data, and interpret the personality classification results effectively.

   5.2. Approach: Develop a web-based or mobile application with intuitive input forms for users to submit their textual posts. Present the classification results in a clear and understandable format, accompanied by visualizations, summary statistics, and personalized insights tailored to each user's MBTI type.

6. Evaluation and Validation:

   6.1. Challenge: Assessing the performance and generalization capabilities of the personality classification models across different datasets and user populations.

   6.2. Approach: Conduct rigorous evaluation metrics such as accuracy, precision, recall, F1-score, and ROC-AUC (Receiver Operating Characteristic - Area Under the Curve) score to measure the models' predictive performance. Validate the models on unseen data and conduct user studies to gather feedback on the system's effectiveness and usability.

7. Ethical Considerations and Privacy Protection:

    7.1. Challenge: Addressing ethical concerns related to user privacy, data security, and potential biases in the classification process.

    7.2. Approach: Implement robust privacy protocols to anonymize user data and secure sensitive information. Mitigate biases through fair sampling techniques, model explainability, and transparent communication of limitations and uncertainties associated with personality prediction.

By addressing these key challenges through comprehensive data analysis, model development, and user-centric design, the intelligent personality classification system aims to provide users with valuable insights into their personality traits while upholding ethical standards and ensuring transparency in the decision-making process.

2.2 Requirements:

2.2.1 Functional Requirements:

1. Data Collection and Loading:
    1.1. Load the dataset from a CSV file into a Pandas DataFrame for further processing and analysis.

2. Data Exploration and Preprocessing:
    2.1. Display initial data insights using head(), info(), and describe() methods.
    2.2. Visualize the distribution of personality types using count plots.
    2.3. Split personality types into four dichotomies (E-I, N-S, F-T, J-P) and visualize their distributions.

3. Text Cleaning and Normalization:
    3.1. Expand contractions in the text data.
    3.2. Convert text to lowercase.
    3.3. Remove mentions, hashtags, URLs, non-alphabetic characters, and extra spaces.
    3.4. Filter out short words (less than three characters).

4. Feature Engineering:
    4.1. Count the number of words and characters in each post.
    4.2. Generate and visualize most frequent words, excluding stop words.
    4.3. Create and display word clouds for overall text data and subsets based on personality types.

5. N-gram Analysis:
    5.1. Tokenize text and remove stop words.
    5.2. Generate and analyze bigrams and trigrams.
    5.3. Visualize the most frequent n-grams.

6. Text Vectorization:
    6.1. Use TF-IDF Vectorizer to transform text data into feature vectors.

7. Model Training and Evaluation:
    7.1. Train multiple classifiers (Naive Bayes, SVM, Decision Tree, Random Forest, XGBoost) for each personality dichotomy (E-I, N-S, F-T, J-P).
    7.2. Evaluate models using metrics like accuracy, precision, recall, F1-score, and ROC-AUC score.
    7.3. trained models and vectorizer to disk.

8. Model Selection and Persistence:
    8.1. Select and save the best-performing models for future predictions.

9. Explainability:
    9.1. Use LIME to provide explanations for model predictions.
    9.2. Use SHAP to visualize feature importance and model decision-making processes.

10. Model Prediction and Interpretation:
    10.1. Load and use the saved models for making predictions on new data.
    10.2. Generate interpretable explanations for individual predictions using LIME and SHAP.

11. Documentation and Reporting:
    11.1. Document all steps, methodologies, and results for inclusion in the project report. This includes data preprocessing, model training, evaluation metrics, and explanation techniques.

2.2.2 Non-Functional Requirements

1. Scalability: Ensure that the system can handle a growing amount of data and users without compromising performance. This includes the ability to scale horizontally (adding more machines) or vertically (upgrading hardware) as needed.

2. Performance: Specify the expected response times for various operations within the system, such as data processing, model training, and inference. Aim for efficient algorithms and optimize resource utilization

to meet these performance goals.

3. Reliability: The system should be highly reliable, with minimal downtime and robust error handling. Implement mechanisms for fault tolerance, graceful degradation, and automated recovery to ensure continuous operation.

4. Security: Safeguard sensitive user data and prevent unauthorized access or malicious attacks. Implement encryption, authentication, and access control mechanisms to protect data integrity and confidentiality.

5. Maintainability: Design the system with modularity, code readability, and documentation in mind to facilitate easy maintenance and future enhancements. Use version control systems and follow coding standards to streamline development workflows.

6. Usability: Prioritize user experience by designing intuitive interfaces, providing helpful feedback, and minimizing cognitive load. Conduct usability testing and gather feedback from users to iteratively improve the system's usability.

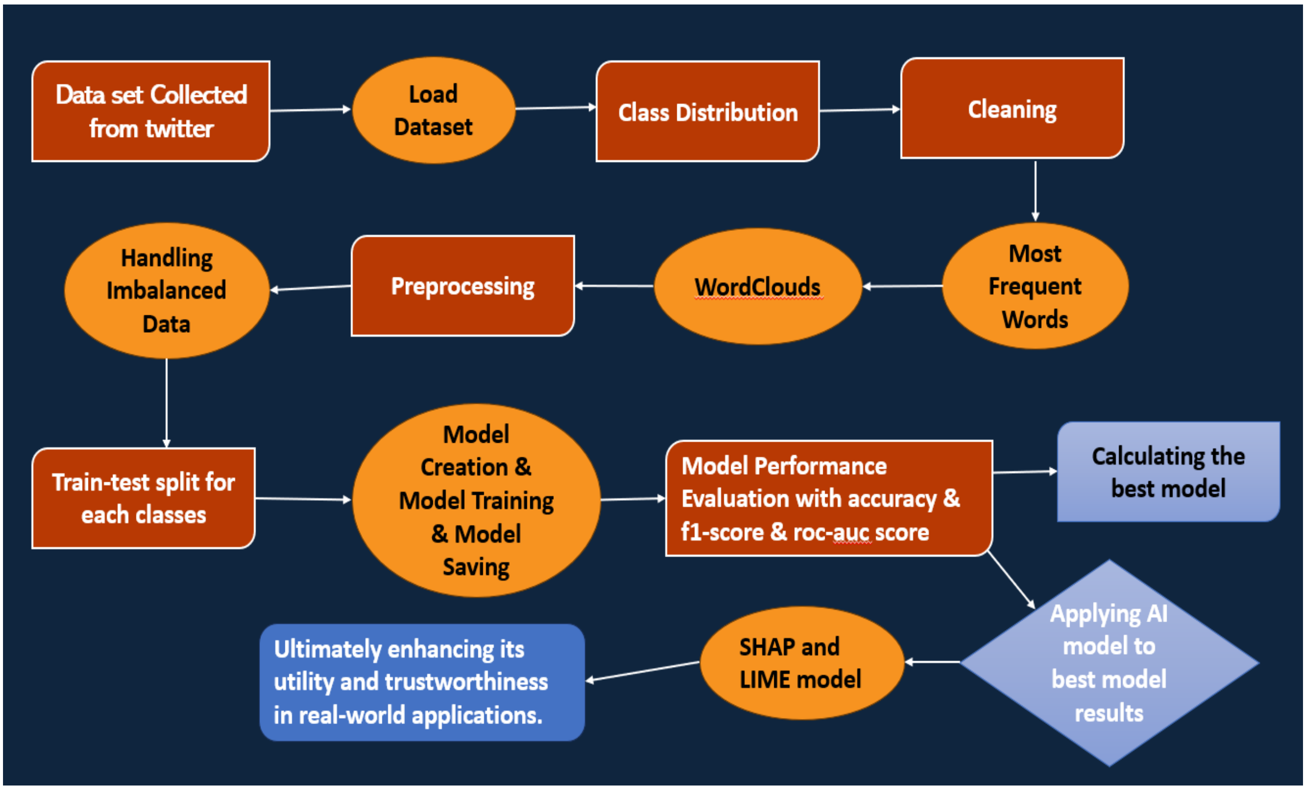
7. Compatibility: Ensure compatibility with different devices, operating systems, and web browsers to maximize accessibility for users. Test the system across various environments and configurations to identify and resolve compatibility issues.

8. Scalability: Ensure that the system can handle a growing amount of data and users without compromising performance. This includes the ability to scale horizontally (adding more machines) or vertically (upgrading hardware) as needed.

9. Interoperability: Design the system to integrate seamlessly with external services, APIs, and data sources. Define clear interfaces and protocols to facilitate interoperability with third-party systems and enable data exchange.

10. Compliance: Adhere to relevant legal, regulatory, and industry standards, such as data protection regulations (e.g., GDPR), privacy policies, and security certifications. Conduct regular audits and compliance checks to ensure adherence to these standards..

2.3 E-R Diagram / Data-Flow Diagram (DFD):



Chapter 3: IMPLEMENTATION

3.1 Date Set Used in the Minor Project

The dataset used in the minor project comprises posts categorized by Myers-Briggs Type Indicator (MBTI) types, namely INFJ, ENTP, INTP, and INTJ. Each type is associated with a series of posts, featuring a mixture of text and links to images or videos. The posts cover a wide range of topics, including personal experiences, reflections, opinions, and interactions with others. Some posts discuss relationships, personality traits, and behaviors characteristic of each MBTI type. This dataset offers a glimpse into the diverse interests, perspectives, and communication styles of individuals belonging to different MBTI types, providing valuable insights for personality analysis and understanding human behavior in online environments.

3.2 Date Set Features:

3.2.1 Types of Data Set:

The dataset provided consists of textual data categorized by Myers-Briggs Type Indicator (MBTI) types. Specifically, it includes posts made by individuals of four different MBTI types: INFJ, ENTP, INTP, and INTJ. These posts contain a variety of content, including personal reflections, opinions, discussions, and links to external content such as images and videos. The dataset primarily comprises unstructured text data, which can be analyzed for patterns, sentiment, and other insights related to personality traits and communication styles associated with each MBTI type.

3.2.2 Number of Attributes, fields, description of the data set:

The dataset consists of one main attribute, which is the "posts" attribute. Each entry in this attribute represents the posts made by individuals of different Myers-Briggs Type Indicator (MBTI) types. Here's a breakdown of the dataset:

Attribute: "posts"

Description: This attribute contains textual data representing the posts made by individuals of the INFJ, ENTP, INTP, and INTJ MBTI types. Each post may include various types of content, such as text, links to videos or images, questions, statements, responses, and discussions.

Since the dataset consists of only one attribute, there are no additional fields or attributes to describe.

3.3 Algorithm / Pseudo code of the Project Problem:

1. Import necessary libraries:
   - pandas as pd
   - numpy as np
   - seaborn as sns

- matplotlib.pyplot as plt
- re
- nltk
- contractions
- WordCloud
- Counter
- BigramAssocMeasures, BigramCollocationFinder
- TrigramAssocMeasures, TrigramCollocationFinder
- ngrams
- WordNetLemmatizer
- TfidfVectorizer
- MultinomialNB, SVC, DecisionTreeClassifier, RandomForestClassifier
- XGBClassifier
- metrics from sklearn
- RandomOverSampler from imblearn

2. Load data from "mb_data.csv" using pandas read_csv() function.

3. Define a function `show_class_distribution(data, x)` to visualize the distribution of personality types.

4. Define a function `divide_types(df)` to split the type column into four separate columns: E-I, N-S, F-T, J-P.

5. Define a function `fix_contractions(df)` to expand contractions in the 'posts' column.

6. Define a function `clean_data(df)` to preprocess text data by converting to lowercase, removing usernames, hashtags, URLs, non-alphabetic characters, and short words.

7. Define a function `plot_counts(df, column, xlabel)` to plot the distribution of word and character counts.

8. Define a function `get_most_frequent(data, stop_words)` to find the most frequent words in the cleaned text data.

9. Define a function `show_most_frequents(most_frequent_words)` to visualize the most frequent words using a bar plot.

10. Define a function `show_wordcloud(data, stopword_list)` to generate and display a word cloud of the cleaned text data.

11. Define a function `show_sub_wordclouds(data, type_column, column, size)` to generate and display sub-word clouds for each personality type.

12. Define a function `get_ngrams(data, n_gram, new_column)` to extract n-grams (bigrams and trigrams) from the cleaned text data.

13. Define functions to find and plot the most frequent n-grams.
14. Define functions to remove stopwords and perform lemmatization.

15. Split the dataset into training and testing sets.

16. Create TF-IDF vectors for the text data using the TfidfVectorizer.

17. Define a function `create_models()` to initialize machine learning models.

18. Define a function `make_dummies(data, columns)` to create dummy variables for categorical columns.

19. Define a function `show_distribution(data, x)` to visualize the distribution of categorical variables.

20. Apply oversampling to handle class imbalance.

21. Train machine learning models, evaluate performance metrics, and save the best models.

22. Optionally, save the TF-IDF vectorizer and trained models to files.

23. Optionally, load a saved model and explain predictions using LIME or SHAP.

3.6 Screen shots of the various stages of the Project:

- Dataset used: Used this “mb_data.csv” for the personality prediction.

<table>
<tr><th></th><th>A</th><th>B</th></tr>
<tr><td>1</td><td>type</td><td>posts</td></tr>
<tr><td>2</td><td>INFJ</td><td>'http://www.youtube.com/watch?v=qsXHcwe3krw|||http://41.media.tumblr.com/tumblr_lfouy03PMA1qa1rooo1_500.jpg|||enfp and intj moments https://www.youtube.com/watch?v=iz7lE1g4XM4 spo</td></tr>
<tr><td>3</td><td>ENTP</td><td>'I'm finding the lack of me in these posts very alarming.|||Sex can be boring if it's in the same position often. For example me and my girlfriend are currently in an environment where we have to creatively us</td></tr>
<tr><td>4</td><td>INTP</td><td>'Good one _____   https://www.youtube.com/watch?v=fHiGbolFFGw|||Of course, to which I say I know; that's my blessing and my curse.|||Does being absolutely positive that you and your best friend cou</td></tr>
<tr><td>5</td><td>INTJ</td><td>'Dear INTP,   I enjoyed our conversation the other day. Esoteric gabbing about the nature of the universe and the idea that every rule and social code being arbitrary constructs created...|||Dear ENTJ sub,   I</td></tr>
<tr><td>6</td><td>ENTJ</td><td>'You're fired.|||That's another silly misconception. That approaching is logically is going to be the key to unlocking whatever it is you think you are entitled to.   Nobody wants to be approached with BS...|||</td></tr>
<tr><td>7</td><td>INTJ</td><td>'18/37 @.@|||Science is not perfect. No scientist claims that it is, or that scientific information will not be revised as we discover new things. Rational thinking has been very useful to our society....|||INF</td></tr>
<tr><td>8</td><td>INFJ</td><td>'No, I can't draw on my own nails (haha). Those were done by professionals on my nails. And yes, those are all gel. You mean those you posted were done by yourself on your own nails? Awesome!|||Probal</td></tr>
<tr><td>9</td><td>INTJ</td><td>'I tend to build up a collection of things on my desktop that i use frequently and then move them into a folder called 'Everything' from there it get sorted into type and sub type|||i ike to collect odd objects,</td></tr>
<tr><td>10</td><td>INFJ</td><td>I'm not sure, that's a good question. The distinction between the two is so dependant on perception. To quote Robb Flynn, "The hate you feel is nothing more, than love you feel to win this war."|||Good qu</td></tr>
<tr><td>11</td><td>INTP</td><td>'https://www.youtube.com/watch?v=w8-egj0y8Qs|||I'm in this position where I have to actually let go of the person, due to a various reasons. Unfortunately I'm having trouble mustering enough strength t</td></tr>
<tr><td>12</td><td>INFJ</td><td>'One time my parents were fighting over my dad's affair and my dad pushed my mom. The fall broke her finger. She's pointed a gun at him and made him get on his knees and beg for his life. She's...|||I'm go</td></tr>
<tr><td>13</td><td>ENFJ</td><td>'https://www.youtube.com/watch?v=PLAaiKvHvZs|||51 :o|||I went through a break up some months ago. We were together for 4 years and I had planned my life around that relationship. I wasn't the one</td></tr>
<tr><td>14</td><td>INFJ</td><td>'Joe santagato - ENTP|||ENFJ or ENTP?   I'm not too sure of his type yet|||You know you're not INFJ if heavy Fi doesn't make you want to violently bang your head against a wall lol You know you're not IN</td></tr>
<tr><td>15</td><td>INTJ</td><td>'Fair enough, if that's how you want to look at it. Like I stated before, they were incredibly naive in their comments... However, they think those are things that would help us because those are the...|||For m</td></tr>
<tr><td>16</td><td>INTP</td><td>'Basically this... https://youtu.be/1pH5c1JkhLU|||Can I has Cheezburgr?|||I am very fond of my top hat too. I certainly did not expect to see a thread about top hats on here haha.|||Streets of Rage 2 for S</td></tr>
<tr><td>17</td><td>INTP</td><td>'Your comment screams INTJ, bro. Especially the useless part.|||Thanks for the information. Doesn't interfere with anything I've ever experienced (with INFJs). Plus, your signature is the lyrics from one of my</td></tr>
<tr><td>18</td><td>INFJ</td><td>'some of these both excite and calm me: BUTTS bodies brains community gardens camping camping with dogs hiking with dogs chillin with animals|||I would hope that no one engages the INTP's baiting</td></tr>
<tr><td>19</td><td>INFP</td><td>'I think we do agree. I personally don't consider myself Alpha, Beta, or Foxtrot (lol at my own joke). People are people. We both agree that having emotions isn't the same as being weak, whiny,...|||Literatur</td></tr>
<tr><td>20</td><td>INFJ</td><td>'I fully believe in the power of being a protector, to give a voice to the voiceless. So in that spirit I present this film, and hope it it recieved in the spirit of compassion. Om Mani Padme Hum ...|||Yes, you are</td></tr>
<tr><td>21</td><td>INFP</td><td>'That's normal, it happens also to me. If I am in high mood, I can act like a 478. Depressed, like a 468. Satisfied and relaxed, 451. But the real type of mine is 458.|||How do they say? (...) in sheep's clothing.</td></tr>
<tr><td>22</td><td>INTP</td><td>'Steve Job's was recognized for his striving for efficiency and practicality. His genius is in his systemization of inventions, less so than in invention. This is where claims of Se and Te come from.|||Pencil. Not i</td></tr>
<tr><td>23</td><td>INFJ</td><td>'It is very annoying to be misinterpreted. Especially with regards to your core, to your intentions and desires. Like when people keep saying that you're in love with somebody for whom you only had...|||x93I</td></tr>
<tr><td>24</td><td>ENTJ</td><td>'Now I'm interested. But too lazy to go research it, because it's time-consuming :(|||Welcome to the club, mate!   https://s-media-cache-ak0.pinimg.com/originals/a3/18/64/a31864d4b4f164aa86dcba9115</td></tr>
<tr><td>25</td><td>INFP</td><td>'45016 urh sorry uh. couldn't resist.|||all of you enfjs, please collectively marry me.|||When an ENFJ is interested in you, you will know it. :D enfjs are my favorite for this reason.|||she seems to have a simi</td></tr>
<tr><td>26</td><td>ENTJ</td><td>'Still going strong at just over the two year mark. I have made noticeable changes and do not plan on slowing. I have attached my 2 year progress picture, but with my face cropped out, you know to...|||Con</td></tr>
<tr><td>27</td><td>INFP</td><td>'Personally, I was thinking this would be more of an SJ type job in a ways.|||I was having some issues a while back finding a job. Couldn't get a job in the arts and crafts store, which was my ideal. I did get a j</td></tr>
<tr><td>28</td><td>ENFP</td><td>'He doesn't want to go on the trip without me, so me staying behind wouldn't be an option for him. I think he really does believe that I'm the one being unreasonable. He still continues to say that...|||I'm stil</td></tr>
</table>

- Uploading dataset: Uploading the dataset and obtaining the most basic information about the dataset.

In [2]:

```python
data = pd.read_csv("mb_data.csv")
data.head(10)
# to see how data looks and to check whether the dataset is uploaded fine
```

Out[2]:

<table>
<tr><th></th><th>type</th><th>posts</th></tr>
<tr><td>0</td><td>INFJ</td><td>'http://www.youtube.com/watch?v=qsXHcwe3krw|||...</td></tr>
<tr><td>1</td><td>ENTP</td><td>'I'm finding the lack of me in these posts ver...</td></tr>
<tr><td>2</td><td>INTP</td><td>'Good one _____ https://www.youtube.com/wat...</td></tr>
<tr><td>3</td><td>INTJ</td><td>'Dear INTP, I enjoyed our conversation the o...</td></tr>
<tr><td>4</td><td>ENTJ</td><td>'You're fired.|||That's another silly misconce...</td></tr>
<tr><td>5</td><td>INTJ</td><td>'18/37 @.@|||Science is not perfect. No scien...</td></tr>
<tr><td>6</td><td>INFJ</td><td>'No, I can't draw on my own nails (haha). Thos...</td></tr>
<tr><td>7</td><td>INTJ</td><td>'I tend to build up a collection of things on ...</td></tr>
<tr><td>8</td><td>INFJ</td><td>I'm not sure, that's a good question. The dist...</td></tr>
<tr><td>9</td><td>INTP</td><td>'https://www.youtube.com/watch?v=w8-egj0y8Qs||...</td></tr>
</table>

In [3]:

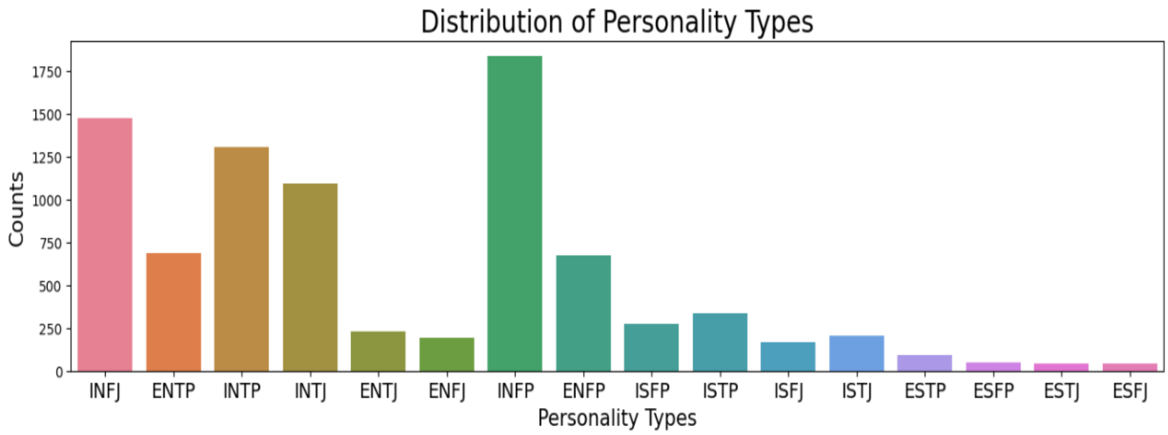
```python
data.info()
# to check how many entries are loaded
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8675 entries, 0 to 8674
Data columns (total 2 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   type    8675 non-null   object
 1   posts   8675 non-null   object
dtypes: object(2)
```

- Personality Type Distribution: Classification of dataset into different post into MBTI Personality classifications.

```
In [7]: show_class_distribution(data, xticks_size=14)
```

```
C:\Users\PRACHI VARSHNEY\AppData\Local\Temp\ipykernel_1608\4147169498.py:6: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and se
t `legend=False` for the same effect.

  sns.countplot(x=x, data=data, palette=palette)
```



- Cleaning the post:

```
In [14]: # fix_contractions:


def fix_contractions(df, column_name = "posts", new_column="cleaned_post"):
    df[new_column] = df[column_name].apply(lambda x: contractions.fix(x))
    return df

data = fix_contractions(data)
def clean_data(df, column_name = "cleaned_post"):
    df[column_name] = df[column_name].apply(lambda x: x.lower())
    df[column_name] = df[column_name].apply(lambda x: re.sub(r'@([a-zA-Z0-9_]{1,50})', '', x))
    df[column_name] = df[column_name].apply(lambda x: re.sub(r'#([a-zA-Z0-9_]{1,50})', '', x))
    df[column_name] = df[column_name].apply(lambda x: re.sub(r'http[s]?://\S+', '', x))
    df[column_name] = df[column_name].apply(lambda x: re.sub(r'[^A-Za-z]+', ' ', x))
    df[column_name] = df[column_name].apply(lambda x: re.sub(r' +', ' ', x))
    df[column_name] = df[column_name].apply(lambda x: " ".join([word for word in x.split() if not len(word) <3]))
    return df

data = clean_data(data)
```

```
In [15]: data.loc[7,"cleaned_post"]
```

```
Out[15]: 'tend build collection things desktop that use frequently and then move them into folder called everything from there get sorte
d into type and sub type ike collect odd objects even work lot people would call junk but like collect old unused software ill
take that off your hands have bunch old adobe think its quite normal tend only see friends real life every couple months said e
arlier some people just not get but the good ones edit mostly mean tolerate where when sleep dreaming another form being awake
how many more layers this are there any thoughts about sleep keep night edit sometimes too scared thanks wish was free follow i
nterests desired feel though wishes are meant for impossible things seeing you mean visual interpreting seeing mentally underst
anding the concept hello feel though incapable creating anything and wish could cannot stand the interviewer christ that laugh
intj hmmm would interesting see intj this show doubt they would that interesting the general public though know yourself and yo
urself you think sounds more like which one you think sounds like you why you require input from others know what you are quest
ion intjs lean more towards alternative rock then other types music and why answer well you went through all the pages and then
sorted all the songs genre style sometimes look people and see them well the outside least doing all these things and saying al
```

- Filtered data:

```
In [16]: data["words_count"] = data["cleaned_post"].apply(lambda x: len(x.split()))
         data.head(5)
```

Out[16]:

	type	posts	E-I	N-S	F-T	J-P	cleaned_post	words_count
0	INFJ	'http://www.youtube.com/watch?v=qsXHcwe3krw|||...	I	N	F	J	and intj moments sportscenter not top ten play...	454
1	ENTP	'I'm finding the lack of me in these posts ver...	E	N	T	P	finding the lack these posts very alarming sex...	874
2	INTP	'Good one _____ https://www.youtube.com/wat...	I	N	T	P	good one course which say know that blessing a...	653
3	INTJ	'Dear INTP, I enjoyed our conversation the o...	I	N	T	J	dear intp enjoyed our conversation the other d...	820
4	ENTJ	'You're fired.|||That's another silly misconce...	E	N	T	J	you are fired that another silly misconception...	782

```
In [17]: def plot_counts(df, column, xlabel):
             fig = plt.figure()
             plt.xlabel(xlabel)
             plt.ylabel("Frequency")
             df[column].plot.hist(bins=25)
```

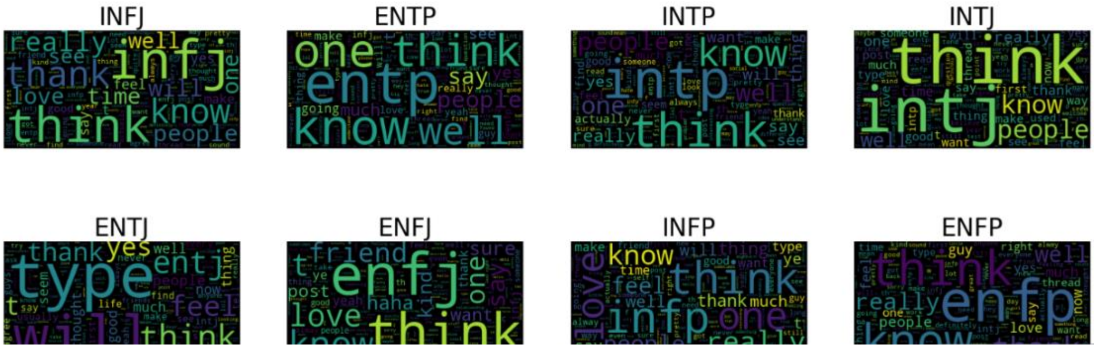
- Most Frequent words:

```
In [24]: most_frequents = get_most_frequent(data, stopword_list)
         most_frequents[:10]

Out[24]: [('like', 69678),
          ('would', 52964),
          ('think', 49837),
          ('people', 48150),
          ('know', 38174),
          ('one', 37173),
          ('really', 35343),
          ('get', 30806),
          ('time', 27610),
          ('feel', 23337)]
```

```
In [25]: def show_most_frequents(most_frequent_words, top=20):
             most_frequent_df = pd.DataFrame(most_frequent_words)
             plt.figure(figsize=(16,4))
             my_cmap = plt.get_cmap("viridis")
             plt.bar(x=most_frequent_df.iloc[:top, 0], height=most_frequent_df.iloc[:top, 1], color="slateblue")
             plt.xlabel("Words", size=17)
             plt.ylabel("Counts", size=17)
             plt.title("Most Frequent Words", size = 20)
             plt.show()
```

- Word Clouds:

```
In [30]: show_sub_wordclouds(data, type_column="type" , column="cleaned_post", size=(4,4))

C:\Users\PRACHI VARSHNEY\AppData\Local\Temp\ipykernel_1608\948420303.py:8: MatplotlibDeprecationWarning: Auto-removal of overla
pping axes is deprecated since 3.6 and will be removed two minor releases later; explicitly call ax.remove() as needed.
  plt.subplot(*size, j+1)
```



- Training and testing Data:

```
In [53]: training_data = data[["cleaned_post", "E-I", "N-S", "F-T", "J-P"]].copy()
         training_data.to_csv("training_data.csv", index=False)
         training_data.head(5)
```

Out[53]:

	cleaned_post	E-I	N-S	F-T	J-P
0	intj moment sportscenter top ten play prank li...	I	N	F	J
1	finding lack post alarming sex boring position...	E	N	T	P
2	good one course say know blessing curse absolu...	I	N	T	P
3	dear intp enjoyed conversation day esoteric ga...	I	N	T	J
4	fired another silly misconception approaching ...	E	N	T	J

```
In [54]: def make_dummies(data, columns=["E-I", "N-S", "F-T", "J-P"]):
             for column in columns:
                 temp_dummy = pd.get_dummies(data[column], prefix="type")
                 data = data.join(temp_dummy)
             return data
```

```
In [55]: training_data = make_dummies(training_data)
         training_data.head()
```

Out[55]:

- Training Model:

```
In [*]: for model_item in models.items():
            for X_train, X_test, y_train, y_test in zip(x_all_train, x_all_test, y_all_train, y_all_test):
                # Model creation and prediction
                model = model_item[1]
                print(f"{model} is training for {y_train.name}...")
                model.fit(X_train, y_train)
                pred = model.predict(X_test)
                # Performance evaluation metrics
                evaluation_df.loc["Accuracy",y_train.name][model_item[0]]      = round(metrics.accuracy_score(y_test, pred), 3)
                evaluation_df.loc["Precision",y_train.name][model_item[0]]     = round(metrics.precision_score(y_test, pred), 3)
                evaluation_df.loc["Recall",y_train.name][model_item[0]]        = round(metrics.recall_score(y_test, pred), 3)
                evaluation_df.loc["F1-Score",y_train.name][model_item[0]]      = round(metrics.f1_score(y_test, pred), 3)
                evaluation_df.loc["Roc-Auc Score",y_train.name][model_item[0]] = round(metrics.roc_auc_score(y_test, pred), 3)
                # Save model
                filename = f'{model}{model_item[0]}_{y_test.name}.sav'
                print(filename)
```

```
MultinomialNB(alpha=0.01) is training for E-I...
MultinomialNB(alpha=0.01)NaiveBayes_E-I.sav
MultinomialNB(alpha=0.01) is training for N-S...
MultinomialNB(alpha=0.01)NaiveBayes_N-S.sav
MultinomialNB(alpha=0.01) is training for F-T...
MultinomialNB(alpha=0.01)NaiveBayes_F-T.sav
```

- Finding best model:

```
In [85]: filename='vectorizer.pkl'
pickle.dump(vectorizer, open(filename, 'wb'))
```

```
In [86]: import os
import pickle
from sklearn.ensemble import RandomForestClassifier
from xgboost import XGBClassifier  # Assuming you're using XGBoost for boosting algorithms

# Assuming you have best_models defined
best_models = [("RandomForest", RandomForestClassifier()), ("Xgboost", XGBClassifier())]  # Replace with your actual best_models

# Define the directory to save the models
models_dir = "saved_models"
if not os.path.exists(models_dir):
    os.makedirs(models_dir)

# Save the best two models into files
for i, (model_name, model) in enumerate(best_models, 1):
    model_filename = os.path.join(models_dir, f"best_model_{i}_{model_name}.sav")
    pickle.dump(model, open(model_filename, 'wb'))
    print(f"Saved {model_name} model to {model_filename}")

print("Best two models have been saved into files:")
for model_name, _ in best_models:
    print(f"{model_name} -> {os.path.join(models_dir, f'best_model_{i}_{model_name}.sav')}")
```

```
Saved RandomForest model to saved_models\best_model_1_RandomForest.sav
Saved Xgboost model to saved_models\best_model_2_Xgboost.sav
Best two models have been saved into files:
RandomForest -> saved_models\best_model_2_RandomForest.sav
Xgboost -> saved_models\best_model_2_Xgboost.sav
```

- Applying Lime on best model:

```
In [104]: import lime
import lime.lime_tabular
loaded_xgb_model.fit(X_train_ei, y_train_ei)

# Now that the model is fitted, you can proceed to explain instances with LIME
explainer1 = lime.lime_tabular.LimeTabularExplainer(X_train_ei, feature_names=vectorizer.get_feature_names_out(), class_names=['E
explainer2 = lime.lime_tabular.LimeTabularExplainer(X_train_ei, feature_names=vectorizer.get_feature_names_out(), class_names=['N
explainer3 = lime.lime_tabular.LimeTabularExplainer(X_train_ei, feature_names=vectorizer.get_feature_names_out(), class_names=['F
explainer4 = lime.lime_tabular.LimeTabularExplainer(X_train_ei, feature_names=vectorizer.get_feature_names_out(), class_names=['J

for instance in data_to_explain:
    explanation1 = explainer1.explain_instance(instance.toarray()[0], loaded_xgb_model.predict_proba, num_features=10)
    explanation2 = explainer2.explain_instance(instance.toarray()[0], loaded_xgb_model.predict_proba, num_features=10)
    explanation3= explainer3.explain_instance(instance.toarray()[0], loaded_xgb_model.predict_proba, num_features=10)
    explanation4 = explainer4.explain_instance(instance.toarray()[0], loaded_xgb_model.predict_proba, num_features=10)
    explanation1.show_in_notebook()
    explanation2.show_in_notebook()
    explanation3.show_in_notebook()
    explanation4.show_in_notebook()


 # Display explanation (you can choose other methods for visualization)
```

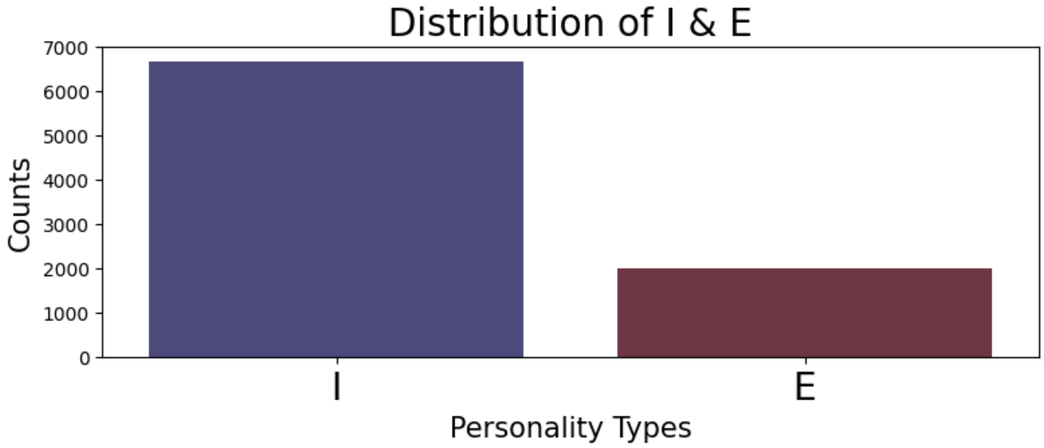
- Applying SHAP AI Model:

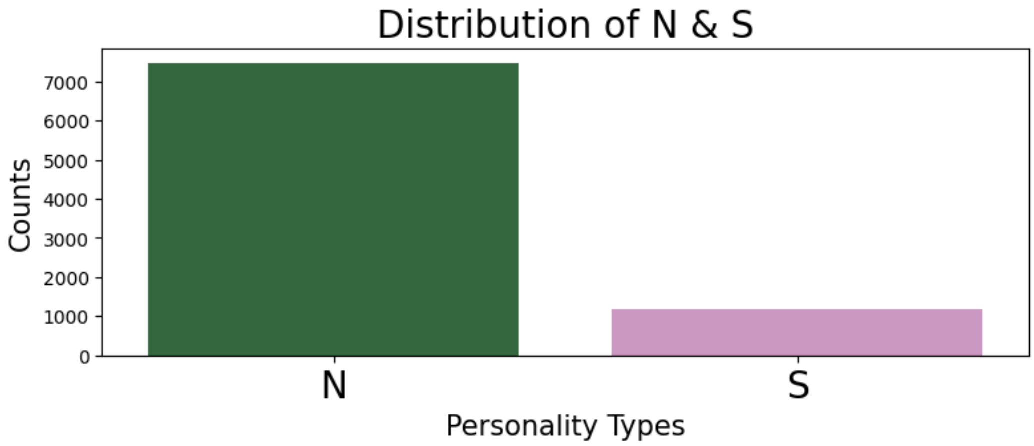
```
In [105]: import shap
import pickle
import xgboost as xgb
import numpy as np
from sklearn.feature_extraction.text import TfidfVectorizer

# Initialize JavaScript for SHAP visualizations
shap.initjs()

# Load the vectorizer and model
with open('vectorizer.pkl', 'rb') as f:
    vectorizer = pickle.load(f)

with open('xgboost_model.pkl', 'rb') as f:
    loaded_xgb_model = pickle.load(f)

# Fit the XGBoost model with training data (assuming you haven't fitted it already)
loaded_xgb_model.fit(X_train_ei, y_train_ei)

# Convert data_to_explain to dense format
data_to_explain_dense = data_to_explain.toarray()

# Now that the model is fitted, create a SHAP explainer
explainer = shap.TreeExplainer(loaded_xgb_model)

# Compute SHAP values for the data you want to explain
shap_values = explainer.shap_values(data_to_explain_dense)

# Visualize SHAP summary plot
shap.summary_plot(shap_values, data_to_explain_dense, feature_names=vectorizer.get_feature_names_out())
```

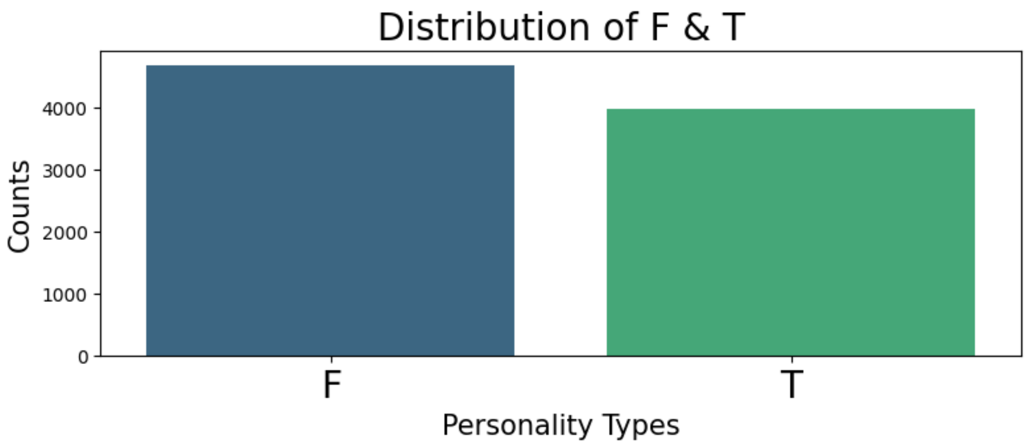
Chapter 04: RESULTS

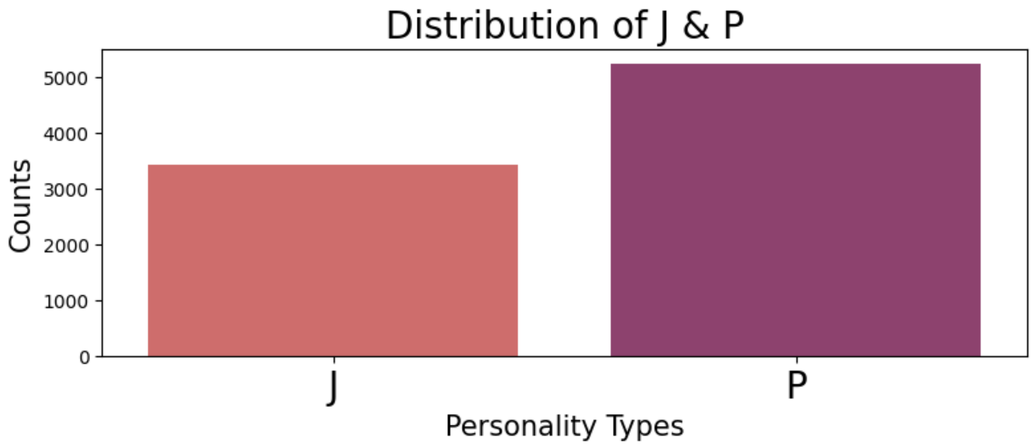
4.1 Discussion on the Results Achieved:

1. Distributing the data and classifying it in the mbti classes: Classifying the dataset into MBTI Classes to check how much the data is imbalanced by looking at the plot so thata we can remove the baiseness while traning the data.









2. Cleaned post data where the extra words and meaning less words and all the stop words , commas, symbols and removed and only meaningful words are stored.

	type	posts	E-I	N-S	F-T	J-P	cleaned_post	words_count
0	INFJ	'http://www.youtube.com/watch?v=qsXHcwe3krw|||...	I	N	F	J	and intj moments sportscenter not top ten play...	454
1	ENTP	'I'm finding the lack of me in these posts ver...	E	N	T	P	finding the lack these posts very alarming sex...	874
2	INTP	'Good one _____ https://www.youtube.com/wat...	I	N	T	P	good one course which say know that blessing a...	653
3	INTJ	'Dear INTP, I enjoyed our conversation the o...	I	N	T	J	dear intp enjoyed our conversation the other d...	820
4	ENTJ	'You're fired.|||That's another silly misconce...	E	N	T	J	you are fired that another silly misconception...	782

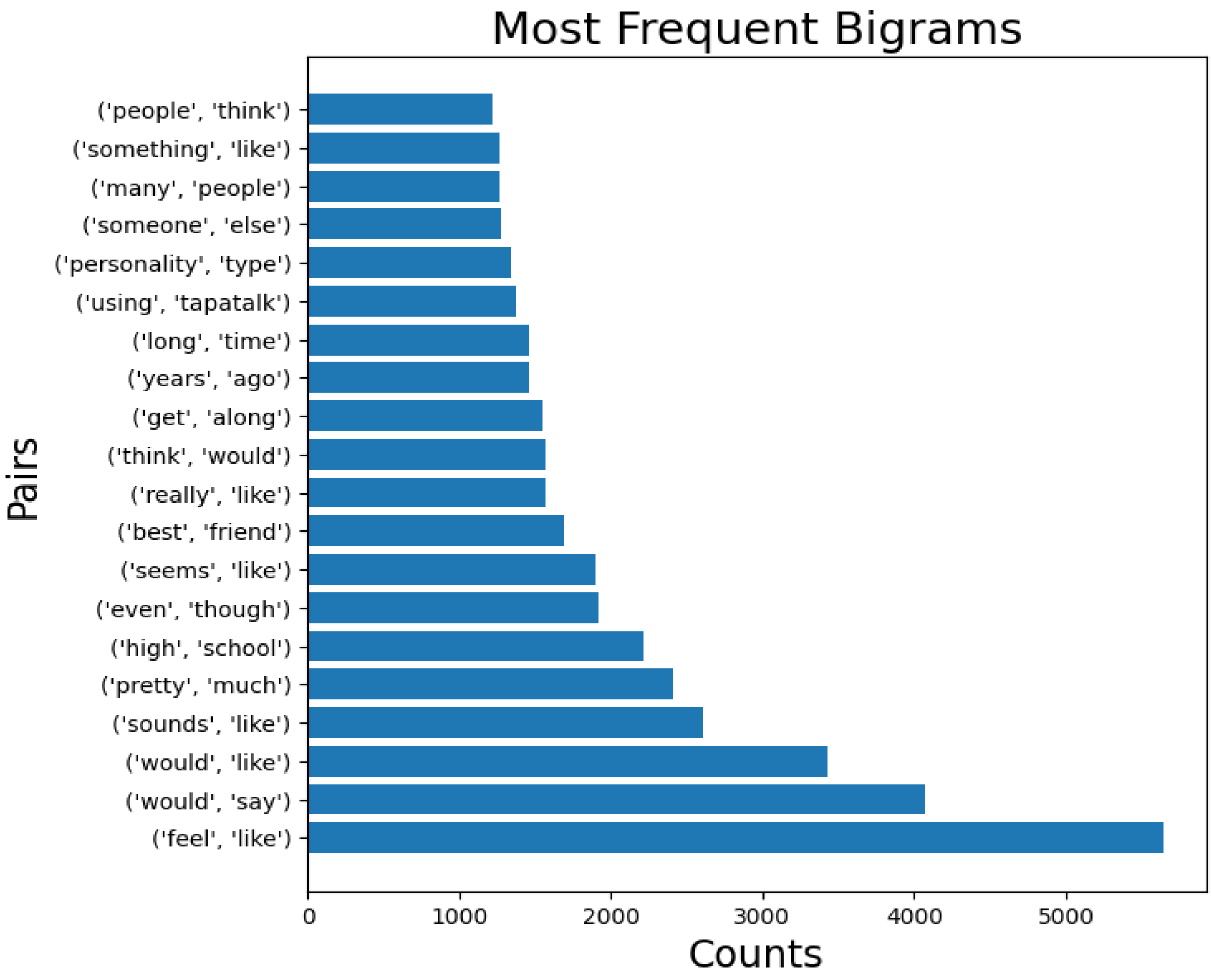
3. Cleaned post data where the extra words and meaning less words and all the stop words , commas, symbols and removed and only meaningful words are stored.with the word counts.

	type	posts	E-I	N-S	F-T	J-P	cleaned_post	words_count	char_count
0	INFJ	'http://www.youtube.com/watch?v=qsXHcwe3krw|||...	I	N	F	J	and intj moments sportscenter not top ten play...	454	2764
1	ENTP	'I'm finding the lack of me in these posts ver...	E	N	T	P	finding the lack these posts very alarming sex...	874	5104
2	INTP	'Good one _____ https://www.youtube.com/wat...	I	N	T	P	good one course which say know that blessing a...	653	4000
3	INTJ	'Dear INTP, I enjoyed our conversation the o...	I	N	T	J	dear intp enjoyed our conversation the other d...	820	4938
4	ENTJ	'You're fired.|||That's another silly misconce...	E	N	T	J	you are fired that another silly misconception...	782	4692

4. Plotting the most frequent words found in the dataset.

5. Word Clouds are drawn for the words fornd under different personality

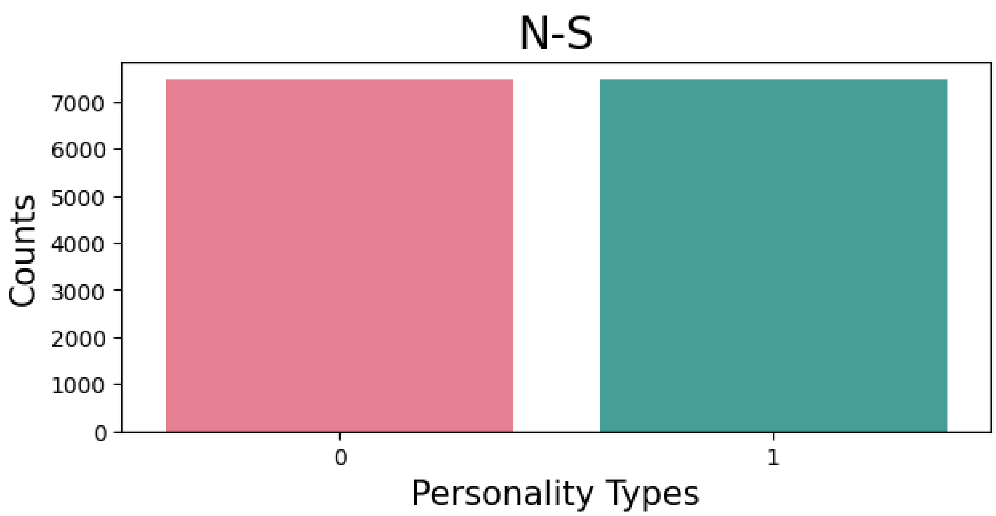
6. Checking the pairs of the word that are occurring together and plotting them.

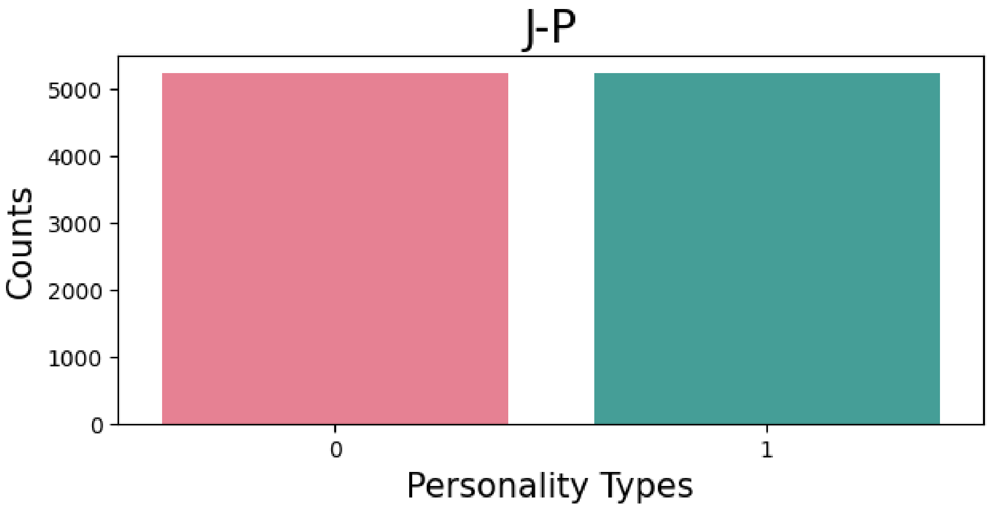


7. Traning data :

	cleaned_post	E-I	N-S	F-T	J-P
0	intj moment sportscenter top ten play prank li...	I	N	F	J
1	finding lack post alarming sex boring position...	E	N	T	P
2	good one course say know blessing curse absolu...	I	N	T	P
3	dear intp enjoyed conversation day esoteric ga...	I	N	T	J
4	fired another silly misconception approaching ...	E	N	T	J

8. Removing the biasness of data:





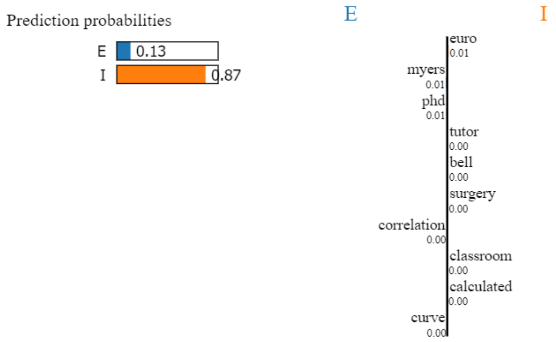
9. Outputs obtained after training the model.

		NaiveBayes	SVM	DecisionTree	RandomForest	Xgboost
Accuracy	E-I	0.813	0.9	0.785	0.953	0.938
	N-S	0.902	0.951	0.805	0.993	0.976
	F-T	0.816	0.855	0.754	0.843	0.846
	J-P	0.721	0.799	0.738	0.823	0.845
Precision	E-I	0.82	0.886	0.811	0.987	0.916
	N-S	0.903	0.978	0.772	0.992	0.996
	F-T	0.812	0.848	0.791	0.83	0.849
	J-P	0.726	0.802	0.785	0.904	0.843
Recall	E-I	0.802	0.918	0.746	0.918	0.965
	N-S	0.904	0.923	0.871	0.995	0.956
	F-T	0.823	0.865	0.69	0.861	0.842
	J-P	0.705	0.791	0.65	0.72	0.846
F1-Score	E-I	0.811	0.902	0.777	0.952	0.94
	N-S	0.903	0.95	0.819	0.993	0.976
	F-T	0.818	0.856	0.737	0.845	0.845
	J-P	0.715	0.797	0.711	0.802	0.845
Roc-Auc Score	E-I	0.813	0.9	0.785	0.953	0.938
	N-S	0.902	0.951	0.805	0.993	0.976
	F-T	0.816	0.855	0.754	0.843	0.846
	J-P	0.721	0.799	0.737	0.822	0.845

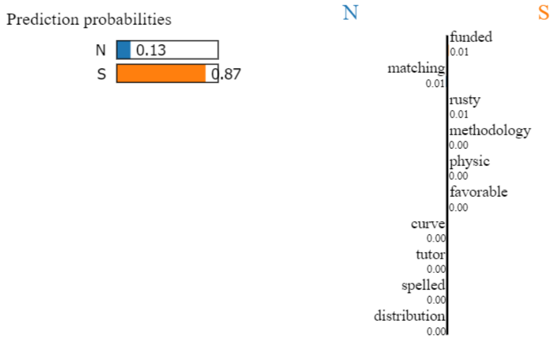
10. Here we have calculated the best model among the 5 models thata we have applied

```
Saved RandomForest model to saved_models\best_model_1_RandomForest.sav
Saved Xgboost model to saved_models\best_model_2_Xgboost.sav
Best two models have been saved into files:
RandomForest -> saved_models\best_model_2_RandomForest.sav
Xgboost -> saved_models\best_model_2_Xgboost.sav
```

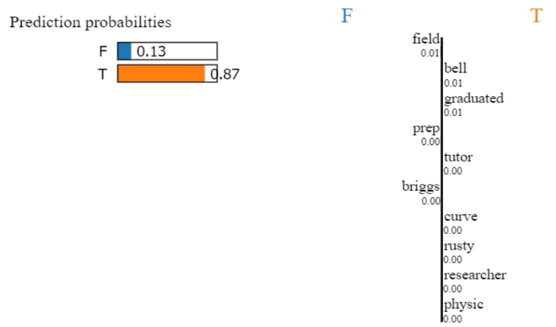
11. Outputs obtained after applying the lime on the xgboost model.



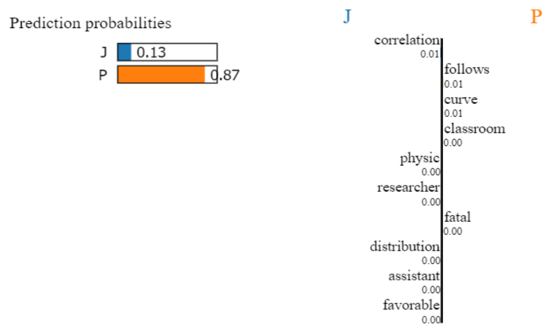
Feature	Value
euro	0.14
myers	0.26
phd	0.11
tutor	0.09
bell	0.11
surgery	0.08
correlation	0.27
classroom	0.11
calculated	0.04
curve	0.20



Feature	Value
funded	0.05
matching	0.04
rusty	0.05
methodology	0.05
physic	0.37
favorable	0.09
curve	0.20
tutor	0.09
spelled	0.04
distribution	0.13

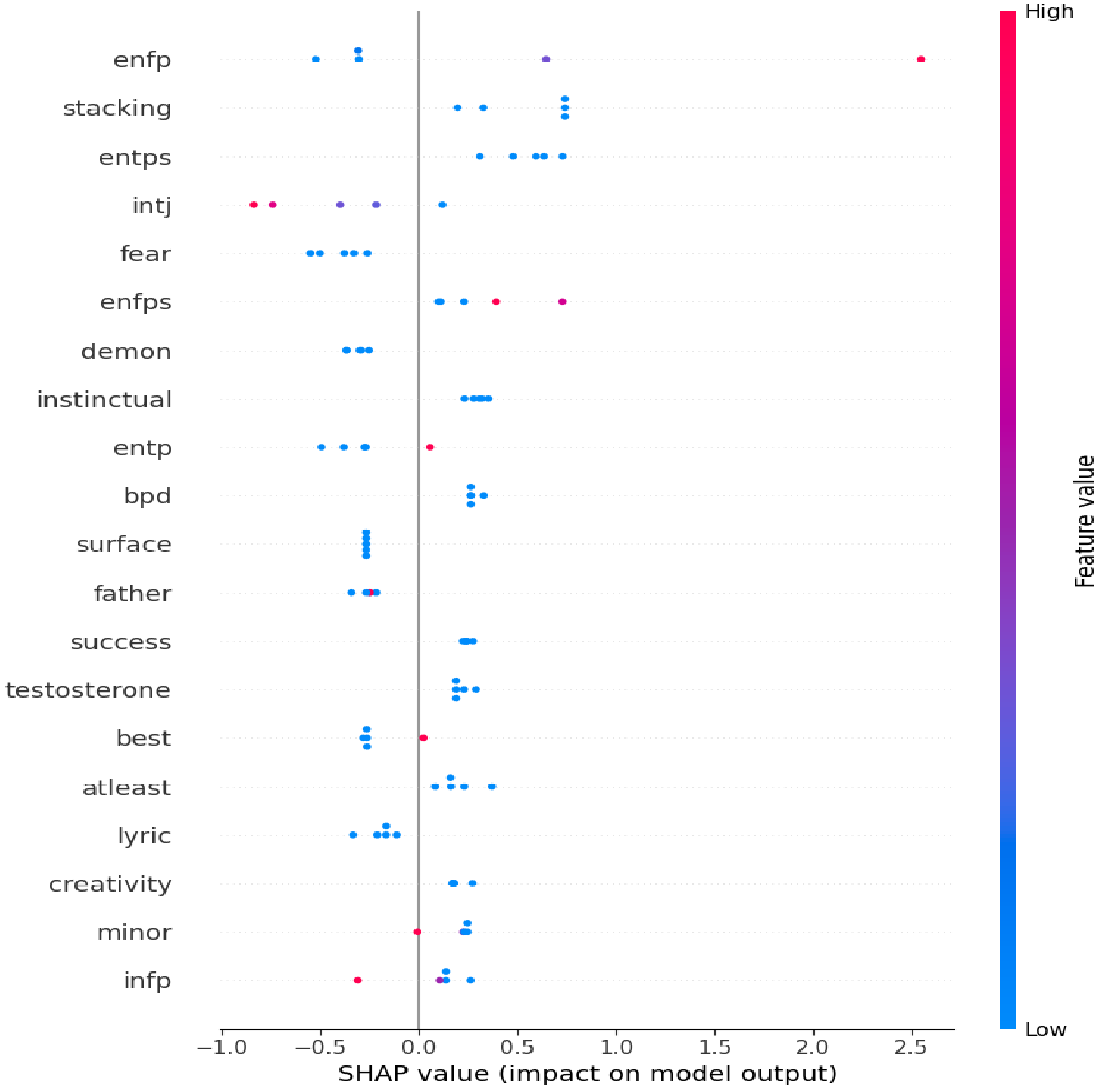


Feature	Value
field	0.15
bell	0.11
graduated	0.10
prep	0.04
tutor	0.09
briggs	0.25
curve	0.20
rusty	0.05
researcher	0.13
physic	0.37

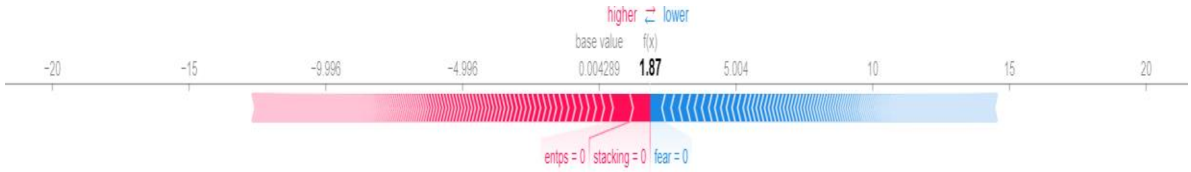


Feature	Value
correlation	0.27
follows	0.07
curve	0.20
classroom	0.11
physic	0.37
researcher	0.13
fatal	0.05
distribution	0.13
assistant	0.12
favorable	0.09

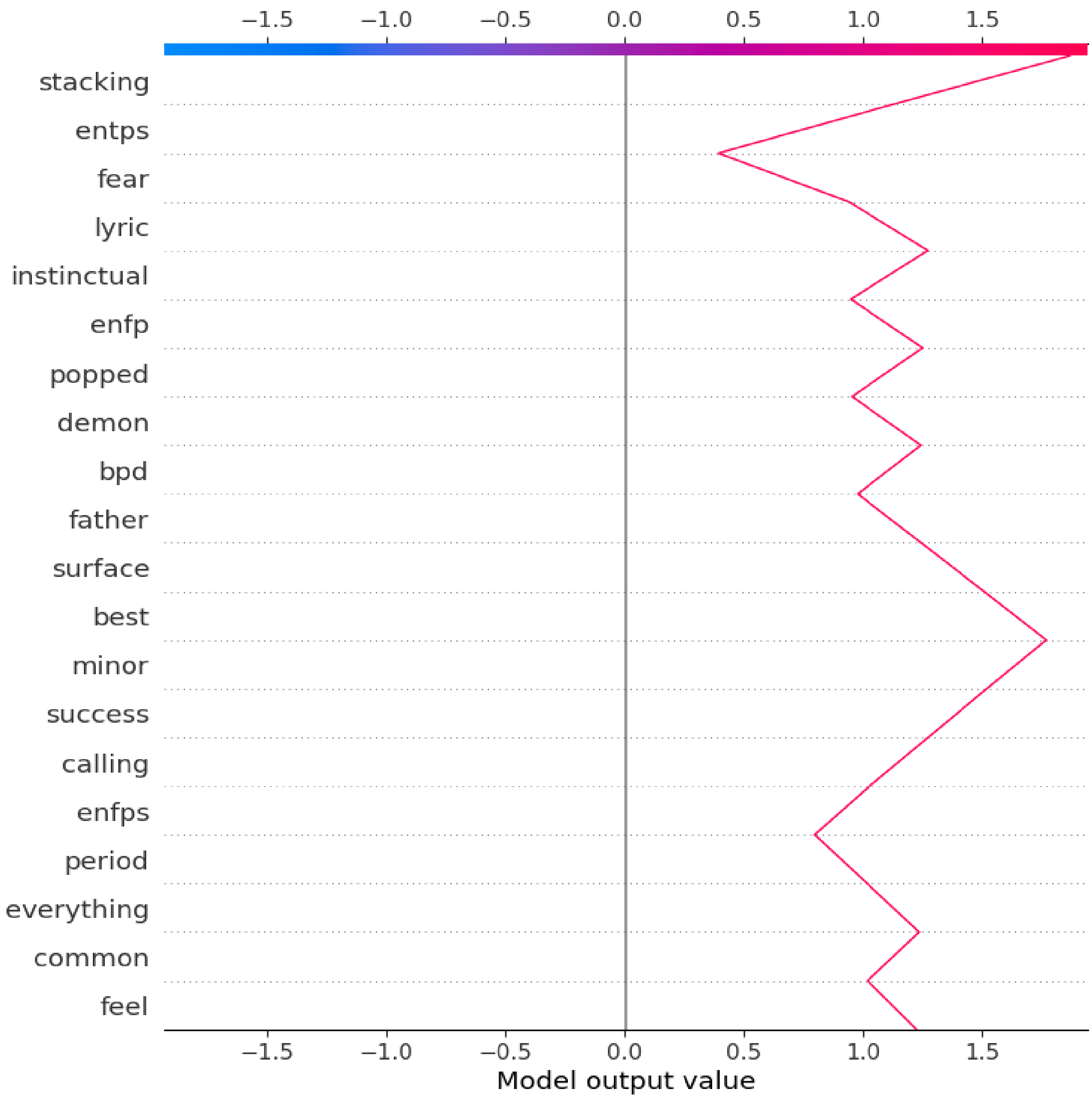
12. Outputs obtained after applying the shap on the xgboost model.



Force Plot:



Dependency Plot:



4.2 Application of the Minor Project

The project of personality prediction using machine learning (ML) models along with LIME (Local Interpretable Model-agnostic Explanations) and SHAP (SHapley Additive exPlanations) applied to a Twitter dataset has a wide range of applications across various domains. Here are some key applications:

1. Targeted Marketing and Advertising:

    1.1. Personalized Campaigns: Understanding the personality traits of users can help marketers design personalized advertising campaigns that resonate more deeply with specific audiences.

    1.2. Product Recommendations: Tailored product recommendations based on personality traits can increase conversion rates and customer satisfaction.

2. Customer Service Enhancement:

2.1. Tailored Communication: Customer service agents can adapt their communication styles based on the personality profiles of customers, leading to better customer experiences.
2.2. Chatbot Personalization: Chatbots can be designed to respond in ways that are more appealing to different personality types.

3. Human Resources and Recruitment:

3.1. Candidate Screening: Employers can use personality insights from social media to screen candidates, assessing traits that align with job requirements and company culture.
3.2. Team Composition: Understanding personality traits can help in forming balanced and cohesive teams.

4. Content Creation and Social Media Management:

4.1. Engaging Content: Content creators can tailor their posts to match the personality preferences of their audience, increasing engagement and follower loyalty.
4.2. Influencer Marketing: Brands can select influencers whose personality profiles align with their brand values and target audience.

5. Mental Health and Well-being:

5.1. Early Detection of Issues: By analyzing social media posts, signs of mental health issues can be identified early, allowing for timely intervention.
5.2. Personalized Interventions: Mental health professionals can tailor their interventions based on the personality traits of individuals.

6. Education and E-Learning:

6.1. Personalized Learning: E-learning platforms can adapt content delivery based on the personality traits of learners, improving learning outcomes.
6.2. Student Engagement: Educators can use personality insights to engage students more effectively.

7. Product Development:

7.1. User Feedback Analysis: By understanding the personalities of users providing feedback on products, companies can better prioritize feature development and product enhancements.

8. Personal Development:

    8.1. Self-awareness: Individuals can gain insights into their own personality traits through their social media activity, aiding in personal growth and self-improvement.
    8.2. Career Guidance: Personality insights can guide individuals towards careers that are well-suited to their traits and preferences.

9. Social Research:

    9.1. Behavioral Studies: Researchers can use personality predictions to study behavioral trends and patterns in different demographic groups.
    9.2. Sociological Insights: Understanding personality distributions can provide insights into social dynamics and cultural trends.

By leveraging LIME and SHAP for model interpretability, this project not only predicts personality traits but also provides clear explanations for the predictions, increasing trust and transparency in the AI models used. This interpretability is crucial for applications in sensitive areas like mental health, human resources, and personalized learning, where understanding the rationale behind predictions can significantly impact decision-making and outcomes.

4.3 Limitation of the Minor Project

While the project of personality prediction using ML models along with LIME and SHAP applied to a Twitter dataset has numerous applications, it also faces several limitations:

1. Data Quality and Bias:

    1.1. Data Representation: The quality and representativeness of the Twitter dataset can significantly impact the accuracy of personality predictions. Twitter users might not represent the broader population, leading to biased results.
    1.2. Noise and Misinterpretation: Social media posts can be noisy and may not always accurately reflect an individual's personality. People may portray different personas online compared to their real-life behavior.

2. Privacy and Ethical Concerns:

2.1. Data Privacy: Collecting and analyzing social media data raises significant privacy issues. Users might not consent to their data being used for personality prediction, leading to ethical concerns.
2.2. Misuse of Information: Predicted personality information could be misused for manipulation, such as in targeted advertising or political campaigns, raising ethical dilemmas.

3. Model Limitations:

3.1. Algorithmic Bias: ML models can inherit and amplify biases present in the training data, leading to unfair or discriminatory outcomes.
3.2. Generalization: The model trained on Twitter data might not generalize well to other social media platforms or offline behavior.

4. Interpretability and Explainability:

4.1. Complexity of Explanations: While LIME and SHAP provide model interpretability, their explanations can sometimes be complex and difficult for non-experts to understand.
4.2. Local vs. Global Explanations: LIME and SHAP provide local explanations (specific to individual predictions), which may not always capture the overall behavior of the model.

5. Dynamic Nature of Social Media:

5.1. Changing Behavior: Social media behavior can change over time due to various factors such as trends, events, or personal growth. A static model might not capture these dynamic changes accurately.
5.2. Context Dependency: The context of tweets (e.g., sarcasm, humor) is often crucial for understanding personality traits, and ML models might struggle with accurately interpreting context.

6. Granularity of Personality Traits:

6.1. Trait Accuracy: Predicting nuanced personality traits from short tweets can be challenging. The limited character count on Twitter might not provide sufficient information to accurately gauge complex personality dimensions.
6.2. Big Five Limitations: Many personality prediction models focus on the Big Five personality traits, which might not capture all relevant aspects of an individual's personality.

7. Cultural and Linguistic Diversity:

7.1. Language Differences: Personality prediction models trained on data in one language might not perform well on data in another language due to cultural and linguistic differences.
7.2. Cultural Context: Personality expression can vary significantly across cultures, and a model trained on a dataset from one cultural context might not be applicable to another.

8. Scalability and Computational Resources:

8.1. Resource Intensive: Training and interpreting complex ML models, especially with techniques like LIME and SHAP, can be computationally intensive and require significant resources.
8.2. Real-time Processing: Applying personality prediction models in real-time applications might be challenging due to the computational overhead of interpretability methods.

Addressing these limitations requires careful consideration of ethical implications, ongoing validation and updating of models, and possibly integrating additional data sources to improve the robustness and fairness of the predictions.

4.4 Future Work

For future work in the project of personality prediction using ML models along with LIME and SHAP applied to a Twitter dataset:

1. Improving Data Quality and Diversity:

1.1. Broader Data Sources: Expand the dataset by incorporating data from multiple social media platforms (e.g., Facebook, Instagram, LinkedIn) to ensure a more comprehensive and representative dataset.
1.2. Enhanced Preprocessing: Develop advanced natural language processing (NLP) techniques to better handle noisy and unstructured text data, thereby improving the accuracy of personality predictions.

2. Addressing Privacy and Ethical Concerns:

2.1. Data Anonymization: Implement robust anonymization techniques to protect user identities and ensure compliance with data privacy regulations like GDPR.
2.2. Ethical Frameworks: Develop and adhere to stringent ethical guidelines for data collection,

analysis, and the application of personality predictions to prevent misuse and ensure responsible AI usage.

3. Enhancing Model Accuracy and Generalizability:

    3.1. Advanced ML Models: Experiment with state-of-the-art ML and NLP models (e.g., transformer-based models like BERT or GPT) to enhance the accuracy and reliability of personality predictions.
    3.2. Cross-Platform Validation: Validate the model across various social media platforms and diverse demographic groups to ensure its robustness and generalizability.

4. Dynamic and Context-Aware Models:

    4.1. Temporal Analysis: Incorporate temporal dynamics into the model to account for changes in user behavior over time, improving the model's adaptability to evolving social media environments.
    4.2. Contextual Understanding: Enhance the model's ability to understand and interpret the context of social media posts, such as sarcasm, humor, and cultural references, for more accurate personality predictions.

Focusing on these areas will address current limitations and significantly improve the effectiveness, fairness, and applicability of the personality prediction models.

References:

[1] K. Sönmezöz, Ö. Uğur, and B. Diri, “MBTI Personality Prediction With Machine Learning,” in *2020 28th Signal Processing and Communications Applications Conference (SIU)*, Oct. 2020, pp. 1–4. doi: 10.1109/SIU49456.2020.9302239.

[2] G. Ryan, P. Katarina, and D. Suhartono, “MBTI Personality Prediction Using Machine Learning and SMOTE for Balancing Data Based on Statement Sentences,” *Information*, vol. 14, no. 4, Art. no. 4, Apr. 2023, doi: 10.3390/info14040217.

[3] M. M. Tadesse, H. Lin, B. Xu, and L. Yang, “Personality Predictions Based on User Behavior on the Facebook Social Media Platform,” *IEEE Access*, vol. 6, pp. 61959–61969, 2018, doi: 10.1109/ACCESS.2018.2876502.

[4] “Information | Free Full-Text | MBTI Personality Prediction Using Machine Learning and SMOTE for Balancing Data Based on Statement Sentences.” Accessed: Feb. 17, 2024. [Online]. Available: https://www.mdpi.com/2078-2489/14/4/217

[5] S. Bharadwaj, S. Sridhar, R. Choudhary, and R. Srinath, “Persona Traits Identification based on Myers-Briggs Type Indicator(MBTI) - A Text Classification Approach,” in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Sep. 2018, pp. 1076–1082. doi: 10.1109/ICACCI.2018.8554828.

[6] “Computational personality recognition in social media | User Modeling and User-Adapted Interaction.” Accessed: Feb. 22, 2024. [Online]. Available: https://link.springer.com/article/10.1007/s11257-016-9171-0

[7] A. V. Sheth and D. R. Pandhare, “Myers-Briggs Personality Prediction Using Machine Learning Techniques,” vol. 6, no. 1, 2024.

[8] S. Singh and W. Singh, “AI-based personality prediction for human well-being from text data: a systematic review,” *Multimed. Tools Appl.*, Oct. 2023, doi: 10.1007/s11042-023-17282-w.

[9] L. C. Lukito, A. Erwin, J. Purnama, and W. Danoekoesoemo, “Social media user personality classification using computational linguistic,” in *2016 8th International Conference on Information Technology and Electrical Engineering (ICITEE)*, Oct. 2016, pp. 1–6. doi: 10.1109/ICITEED.2016.7863313.

[10] “Enriching Social Media Personas with Personality Traits: A Deep Learning Approach Using the Big Five Classes | SpringerLink.” Accessed: Feb. 22, 2024. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-50334-5_7