## MBTI Personality Prediction Using Machine Learning and NLP

[Fateme Taroodi]

**Date**: September 05, 2025

Contents

[MBTI Personality Prediction Using Machine Learning and NLP 1](#_Toc207973440)

[[Fateme Taroodi] 1](#_Toc207973441)

[**1. Introduction** 3](#_Toc207973442)

[**2. Background and Dataset Collection** 4](#_Toc207973443)

[**2.1 Review and Loading of the Dataset** 4](#_Toc207973444)

[**2.2 Management of Class Imbalance** 4](#_Toc207973445)

[**2.3 Text Data Cleaning** 6](#_Toc207973446)

[**2.4 Text Normalization** 6](#_Toc207973447)

[**3. Feature Engineering, Analysis, and Sensitivity** 7](#_Toc207973448)

[**3.1 Sentiment Analysis of Text Data** 7](#_Toc207973449)

[**3.2 Part of Speech Tagging** 8](#_Toc207973450)

[**4. Numerical Feature Extraction from Metadata and Text Standardization** 9](#_Toc207973451)

[**Extraction of Features from Metadata through Counting Method** 9](#_Toc207973452)

[**5. Text Vectorization** 10](#_Toc207973453)

[**5.1 First Method: TF-IDF Vectorizer** 10](#_Toc207973454)

[**5.2 Count Vectorizer Method** 10](#_Toc207973455)

[**6. Exploratory Data Analysis and Visualizations** 11](#_Toc207973456)

[**6.1 Word Cloud Chart for Each Personality Type** 11](#_Toc207973457)

[**6.2 Word Count Distribution Chart Based on Personality Type** 12](#_Toc207973458)

[**6.3 Positive Sentiment Distribution Chart Based on Personality Type** 13](#_Toc207973459)

[**6.4 Personality Types Distribution Chart in the Dataset** 15](#_Toc207973460)

[**7. Model Evaluation and Final Construction** 16](#_Toc207973461)

[**7.1 Final Preparation for Models** 16](#_Toc207973462)

[**7.2 Comparison and Testing of Different Models** 17](#_Toc207973463)

[**7.3 Analysis of Important Features in Final Models** 18](#_Toc207973464)

[**8. Final Model Testing** 18](#_Toc207973465)

[**8.1 Performance Evaluation on Holdout Dataset** 19](#_Toc207973466)

[**8.2 Building a System for Prediction from Input Text Request** 20](#_Toc207973467)

[**9. Final Conclusions** 20](#_Toc207973468)

**1. Introduction**

One of the complexities of human personality is its manifestation in social behaviors and interactions in the fields of social sciences and psychology. Extensive applications have been made to improve communication, career counseling, and individual development through various tools. The Myers-Briggs Type Indicator (MBTI) is one such tool, based on four main dimensions, classifying individuals into 16 personality types. These dimensions consist of: Introversion (I) versus Extraversion (E), Intuition (N) versus Sensing (S), Thinking (T) versus Feeling (F), and Judging (J) versus Perceiving (P).

Traditionally, MBTI personality type determination is conducted through questionnaires and in-depth analyses of responses. This process can be time-consuming, costly, and influenced by biases. With the advent of digital spaces, particularly social networks, there is an opportunity to analyze personality traits using textual content and automated methods. This content can generate individuals' narratives, enabling the extraction of thinking patterns, communication styles, and emotional traits.

The main objective of this project is to utilize machine learning techniques and natural language processing to build an intelligent model for predicting MBTI personality types solely based on individuals' written texts from a collected dataset. We will explain the implementation stages of this project in detail and provide a report on the reading and execution process.

We will describe the project implementation stages in detail in this report. First, the texts are analyzed using post-processing and dimensionality reduction techniques. Then, using standard construction and analysis methods such as feature engineering, information extraction from the structure and semantics of sentences, attachment of labels, and sensitivity analysis, numerical data are converted into textual data for analysis. Finally, we aim to extract better models for improved performance and select the optimal one, which will demonstrate how it can assist in more accurate analysis of online behaviors and provide better services to customers in electronic commerce and personalized content analysis.

These rules play a key role in improving various important variables, providing a connection between changes and electronic commerce analysis. In this project, we use the Instacart dataset, which contains online grocery shopping information for customers. Due to the high volume and diversity of this dataset, it is suitable for analyzing shopping behaviors. One of the serious standards of this project is lift, named SWLT, relative to these standards. They have stronger implementation rules and help more accurately with the extraction of these rules using the FP-Growth algorithm, due to its high efficiency and speed for large data processing.

The project can accurately analyze online behaviors with more precise suggestions, better customer behaviors, and provide improvements in shopping experiences.

**2. Background and Dataset Collection**

The process of building a machine learning model begins with one of the stages where data preparation is performed. This stage includes initial review, handling missing values and imbalances, identification, first review, and standardization of textual data for analysis. In this section, the stages will be explained in detail.

**2.1 Review and Loading of the Dataset**

The project begins with loading the dataset in CSV format, which includes 8675 samples. Each sample belongs to one individual and consists of two main columns:

* **type**: The 16 personality types based on the individual's MBTI classification.
* **posts**: A collection of 50 recent posts from the individual, separated by "|||".

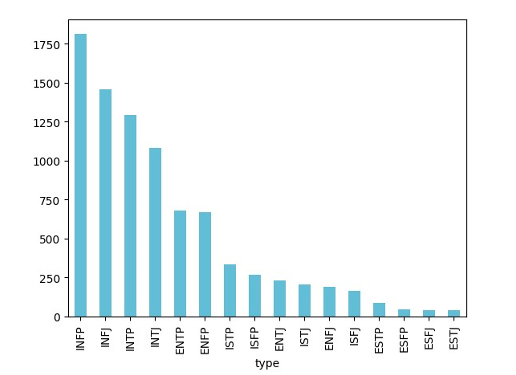
For assurance, after loading, a new review is conducted, and approximately 1% of the data is randomly deleted in a stratified manner (stratify) titled dataset holdout. None of this dataset is final for testing only and will be used for model adjustment and training stages. In the main dataset, which includes 8588 samples, the first review is performed. As shown in the missing values column, there are no missing values in this matter.

**2.2 Management of Class Imbalance**

After distributing the 16 personality types in the dataset, it is evident that it is highly imbalanced. Types such as INFP, INFJ, INTP, and INTJ constitute the majority of the data, while types like ESTJ and ESFJ have very low presence in the distribution chart.

The personality types are shown in the figure below:

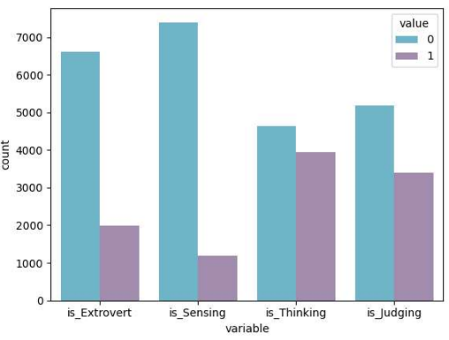
**Figure 1**: Chart of distribution related to personality types



To address this severe imbalance, the model can weaken the performance for repeated classes. To solve this problem, an intelligent approach is adopted: creating a roadmap, which divides the 16 classes into four binary classification problems based on the four dimensions of MBTI. A new column is added to the data, which is independent of each of the four dimensions:

* **is\_Extrovert**: Value 1 for Extraversion (E) and 0 for Introversion (I).
* **is\_Sensing**: Value 1 for Sensing (S) and 0 for Intuition (N).
* **is\_Thinking**: Value 1 for Thinking (T) and 0 for Feeling (F).
* **is\_Judging**: Value 1 for Judging (J) and 0 for Perceiving (P).

The distribution of these four sets is shown below:

**Figure 2**: Chart of distribution related to the four separate personality dimensions

This approach divides the problem into four manageable parts and addresses the imbalance, showing a new dimension in these four that are independent of each other. Correlation between these four new features also indicates that they are independent of each other to a large extent, which is the result of this. Separate development of a model for each would be possible.

**2.3 Text Data Cleaning**

Texts usually contain additional information from the internet called "noise" that can negatively affect the model performance. To prepare the posts column for analysis, the following operations were performed:

1. **Conversion to lowercase**: All texts were converted to lowercase to distinguish between words (e.g., Hello and hello).
2. **Removal of separators**: The "|||" separators that separate the posts were replaced with spaces to convert to a single paragraph.
3. **Removal of links and emails**: Using Regular Expressions, URLs and email addresses were removed from the text, as they lack meaningful value for personality analysis.
4. **Removal of non-alphabetic characters and punctuation**: All characters except English alphabetic letters and spaces were removed, leaving only pure words.
5. **Removal of phrases with MBTI**: In this stage, a very important step is the direct removal of the 16 MBTI personality types from within the text to prevent data leakage. This is necessary; finding a personality type name in the text is a non-realistic prediction, which acts as a label attachment in advance.

**2.4 Text Normalization**

After the initial standardization of the text, two more standardizations were applied:

1. **Lemmatization**: In this stage, words are returned to their base or root form (e.g., "running" to "run"). This aggregates words with similar meanings and reduces vocabulary dimensions. Along with this, stop words such as "the", "a", and "is" are removed from the text, as they are frequent but carry little meaning.
2. **Removal of very short words**: In this final step, words of one or two letters were removed, as they usually lack analytical impact or are remaining noise.

After completing this stage, a new column named clean\_posts was added, which is a completely cleaned and normalized version of the dataset, including the final textual data ready for machine learning processing.

In the next stages of the project, for attachment, the main text and the binary processing files are used in a CSV that has been newly stored.

**3. Feature Engineering, Analysis, and Sensitivity**

After preparing and standardizing the textual data in the previous stage, in this section, we proceed to feature engineering. The goal in this stage is to extract numerical features from the text that raw words alone cannot provide to machine learning models for processing. This is a technique for personality complexity that has been taken into account in this project. To this end, data from natural language richness was used for sentiment analysis and part-of-speech tagging for structural grammar selection and indication.

**3.1 Sentiment Analysis of Text Data**

We perform sentiment analysis to measure the polarity of an individual's writings quantitatively. This can distinguish between personality types; for example, some types tend to use more negative or positive language.

In this project, VADER (Valence Aware Dictionary and sEntiment Reasoner) tool was used for this purpose. This analyzer is specifically for short and informal texts such as posts on social networks. VADER was applied to the clean\_posts column collection for each user. Four composite scores were calculated:

* **pos\_sentiment**: Proportion of positive words and phrases in the text.
* **neg\_sentiment**: Proportion of negative words and phrases in the text.
* **neu\_sentiment**: Proportion of neutral words and phrases in the text.
* **compound\_sentiment**: A normalized overall sentiment score between -1 (completely negative) and +1 (completely positive) that provides a summary of the text's overall sentiments.

Some algorithms such as Naive Bayes work well with negative input values, so all these four sentiment scores were transferred to the [0, 1] numerical range using Min-Max scaling. With this action, four new numerical feature columns were added to the dataset, each providing perspective and insight into the emotional tone.

**3.2 Part of Speech Tagging**

Individuals, independently of the content of the text, can provide abundant information in the context of their personality through their posting network. POS Tagging is a process that specifies the role of each word in a sentence (noun, verb, adjective, pronoun, etc.).

Initially, for this analysis, the posts were processed again (main ones). They were re-processed (completely standardized) from the 50-post collection, then converted from composite sentences to a list using NLTK. Each word in each post was assigned a dedicated tag.

After attachment, for converting this information into numerical form, two approaches were taken:

1. **Statistics for each attachment per user**: Unique to the instruction attachment per user for NN (for noun), or JJ (for adjective): mean and standard deviation statistics. The user's 50 posts use only the amount of diversity or positivity from the structure that provides.
2. **Grouping attachments based on standard categories**: From the composite posts, the number of attachments that are more strongly based on standard categories Stanford such as Noun, Verb, Adj, and Pronoun, and others were grouped. Then, for each of these categories per user, the median number of related words was added and calculated for the user.

At the end of this stage, with the addition of features from sentiment analysis and grammatical structures, these engineering features were enriched. As the main categories, they will be extracted directly from words in the next stages to form numerical vectors, which will provide notable attention in the model.

**4. Numerical Feature Extraction from Metadata and Text Standardization**

In this section, the engineering features are combined with two different approaches. The first approach extracts features (raw) main written from the posts and stores specific words in the network from the text standardization that is vectorization on the condition that it is converted to an understandable and numerical template for the algorithm.

**Extraction of Features from Metadata through Counting Method**

The features of this column are extracted (raw) from the main posts and are preserved. For each user, they are divided by the number and shared with a new column named and calculated:

* **Punctuation Features**:
  + **qm (Question Marks)**: Average number of question marks per post, which can indicate curiosity or skepticism.
  + **em (Exclamation Marks)**: Average number of exclamation marks, indicating energy, excitement outward or.
  + **colons**: Average number of colons (:).
  + **emojis**: Number of simple emoticon patterns like *:* that have been identified.
  + **ellipses**: Average number of ellipses (...) that can indicate pausing thought or hesitation.
* **Word and Post Statistics Features**:
  + **word\_count**: Average word count per post.
  + **unique\_words**: Number of unique words per post for standard, which is Lexical Diversity.
  + **post\_length\_var**: Variance of post lengths per user, how much the individual is variable or consistent.
  + **upper**: Average number of words written in uppercase, which can indicate shouting or emphasis.
* **Content Features**:
  + **link\_count**: Average number of links shared.
  + **img\_count**: Number of image files responded based on (shared to images).

From this addition, 11 features to the main dataset, new with 126 columns shown in the final review. It has not been extended.

**5. Text Vectorization**

The model converts words into numerical vectors, which is necessary and cannot be processed as raw text by machine learning algorithms. In this project, two methods were used from the clean\_posts column, which was prepared in the previous stage. The implementation was performed after preparation to achieve the results of vectorization and to store them compositely for the model stage in order to be used.

For each of the two methods, very frequent words and very rare words were filtered to increase efficiency and reduce noise:

* **min\_df=25**: Words that appeared in at least 25 different users' writings were considered.
* **max\_df=0.8**: Words that appeared in more than 80% of the documents (users) were removed, as they generally have low discriminative power.

With the application of these filters, a vocabulary consisting of 10,046 words was included for unique vectorization per individual.

**5.1 First Method: TF-IDF Vectorizer**

TF-IDF stands for Term Frequency-Inverse Document Frequency. This method not only considers the frequency of a word in a user's writing (TF) but also assigns greater weight to words that are rarer in the entire dataset (IDF). This action causes key and distinctive words in each user's row to receive higher scores, exiting this process as a large matrix where each row is the average of a word from the vocabulary and the user has a TF-IDF score related to it.

**5.2 Count Vectorizer Method**

This method is a simpler model from the Bag-of-Words family, which merely counts the frequency of each word from the vocabulary in the user's writing without weighting. This method provides a direct representation of word frequencies for each user from the frequent vocabulary.

Finally, the two vectorization methods were stored in separate CSV files for the next stages of the model, which were trained using each of them and compared with the other.

**6. Exploratory Data Analysis and Visualizations**

In this section, using engineering features and extracted textual data, we perform exploratory data analysis (EDA) using a technique to visually process data for pattern recognition, anomalies, and deeper understanding of personality types and their relationships. This analysis helps us gain important insights for the next stages, which can serve as a guide for solving and analyzing several charts from constructed visualizations.

**6.1 Word Cloud Chart for Each Personality Type**

**Figure 3**: Word Cloud Chart for Each Personality Type



**Analysis**:

Prominent words among types: There is a significant presence of common and prominent words in the types, such as feel, love, really, time, and say. These have an almost universal presence in all clouds, indicating that topics related to emotions, personal experiences, and social dialogues are central in this dataset.

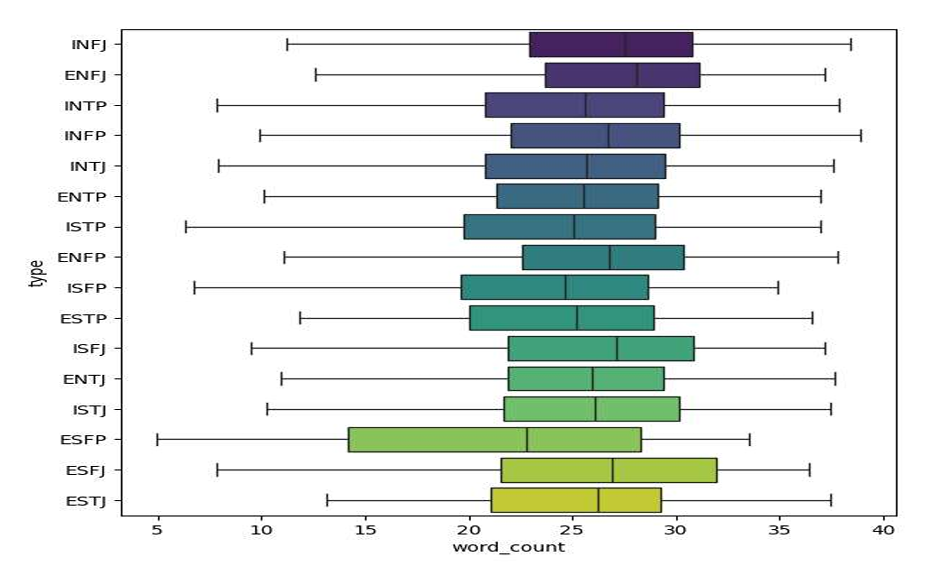
Importance of the Word "Type": The word "type" (referring to personality type) is particularly prominent in clouds related to intuitive types (N), as observed in the data, such as online forums or communities where MBTI types are collected and discussed as a common topic.

**Group-based Creativity**:

* **NF Groups (Intuitive Feelers)**: Types INFJ, INFP, ENFJ, ENFP, with words related to concepts and emotions, such as life, love, feel, and person. These are central and large, with the feeling (F) group fully aligned.
* **NT Groups (Intuitive Thinkers)**: Types INTJ, INTP, ENTJ, ENTP, which, in addition to emotional words, refer more to logical and analytical terms like see, even, much, and actual. They focus more on analytical language.
* **SJ and SP**: These groups also use common emotional words but refer more to practical and tangible experiences such as friend, well, good, and work in this group, showing a greater emphasis.

**6.2 Word Count Distribution Chart Based on Personality Type**

**Figure 4**: Word Count Distribution Chart Based on Personality Type

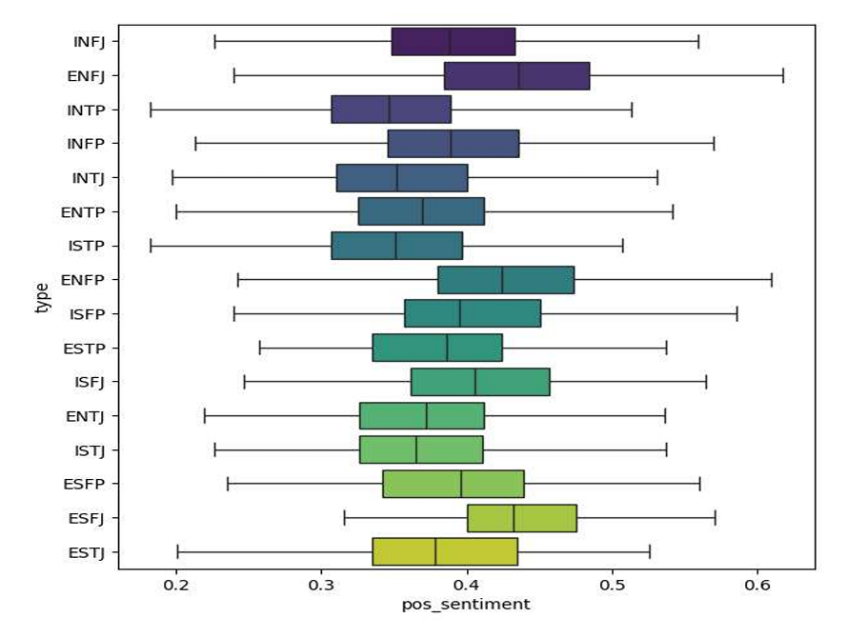


This box chart shows the average word count distribution for each of the 16 personality types, drawn from 50% median data for that type, showing the median line. This visualization allows us to compare personality data horizontally and provides insight into their dispersion.

**Analysis**:

* **Overall Trend in Length**: A notable and clear trend is observed in intuitive types (N) who write longer posts on average. They tend to allocate more word count to their dedicated data, such as INFJ, ENFJ, INTP, INFP, and INTJ, showing higher medians.
* **Comparison of Intuitive (N) and Sensing (S)**: In contrast, sensing types (S) write shorter posts overall, with a significant difference in types like ESTJ, ESFJ, and ESFP placed at the lower end of the chart. This finding indicates that "word count" can ultimately be a strong discriminator between intuitive (N) and sensing (S) personality dimensions.
* **Extroversion/Introversion Effect**: It appears that introverted types (I) tend to be in a quieter environment with fewer indications, for example, median related to INFJ to ENFJ and INTP to ENTP. Although the difference is higher in N/S, it is not as pronounced. This may be because introverted individuals possibly express themselves more in textual templates.
* **Diversity and Dispersion**: The box lengths show horizontal lines and post length dispersion in types, for example. Types like ESFP and ESTJ have shorter boxes (relatively long from very short), which is highly variable. In contrast, types like INFJ and INTP show boxes that are more focused, with longer writings.

**6.3 Positive Sentiment Distribution Chart Based on Personality Type**

**Figure 5**: Positive Sentiment Score Distribution Chart Based on Personality Type

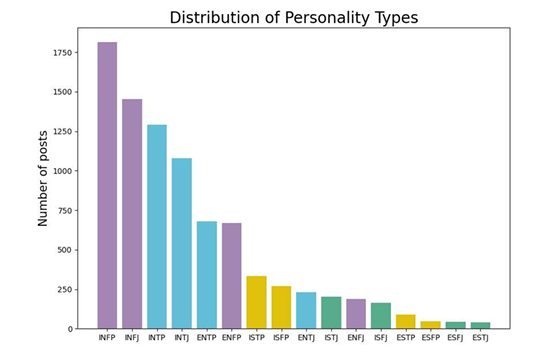
This box chart shows the distribution of positive sentiment scores (pos\_sentiment), measuring the proportion of positive emotional words and phrases in a user's writing. The purpose of this analysis is to examine whether some types tend to use more positive language or vice versa.

**Analysis**:

* Contrary to the word count chart, there are subtle differences here, which are closer to each other. However, observable patterns exist in this state.
* **Systematic Difference Between Thinking (T) and Feeling (F)**: A clear trend in this growth is overall higher positive sentiment scores for feeling types (F). For example, the median score is higher for INFP than INTP, ENFJ than ENTJ, and ISFP than ISTP, fully aligned with the theoretical basis of MBTI, as feeling types tend to express their decisions and emotions based on value and make more emotional language.
* **Slightly Higher for Extroverted Types**: Another trend showing types like extroverted (E) on average slightly higher than their introverted counterparts, titled for example ESFJ and ESFP, which have higher scores placed in the chart.
* In a spectrum of types like ESFJ and ESFP, they tend to use positive words, while on the other side, thinking introverted types like INTP, INTJ, and also INFJ show lower positive sentiment scores, with a more neutral or analytical tone.

**6.4 Personality Types Distribution Chart in the Dataset**

**Figure 6**: Personality Types Distribution Chart in the Dataset



This histogram shows the frequency distribution of each of the 16 personality types in the entire dataset. It shows the color column and is matched with the four main groups defined earlier, known as the introduction.

**Analysis**:

1. **Severe Imbalance in Data**: One of the first and most important points from this chart is a very severe imbalance, which confirms the data as if none were distributed.
2. **Dominance of Introverted Intuitive Types**: The four types INFP, INFJ, INTP, INTJ have the highest frequency in the dataset, which are introverted and intuitive. This group constitutes the majority of the sample, showing a strong pattern in the data, possibly due to a higher likelihood of collecting from an online community (such as a specific psychology forum) where individuals with these personality types and self-interests are present (they have more coloring).
3. **Low Representation of Extroverted Sensing Types**: In contrast, types like ESTJ, ESFJ, ESFP, and ESTP have very low numbers for this group, which is somewhat less educated model for personality distribution with reliable allocation for their prediction possibly non-existent approximately.

**7. Model Evaluation and Final Construction**

In this section, after engineering features and data preparation, multiple models are trained and constructed, meaning the final ones are tested with several different algorithms to select the best model for textual vectorization for each binary personality dimension prediction.

**7.1 Final Preparation for Models**

**Definition of Input Variables (X) and Target (Y)**:

Initially, input variables and targets were specified for model training:

* **Input Variables**: A combination of the most important features extracted in the previous stages was included:
  + **clean\_posts**: The main input titled users' cleaned textual content for analysis.
  + **21 Numerical Features**: Including 11 from sentiment combinations averages, POS tags sets, and 9 counting features (word count, links, punctuation, etc.).
* **Non-Input Variables**: The four binary columns is\_Thinking, is\_Sensing, is\_Extrovert, and is\_Judging were considered as composite targets for each of the four problems per layer.

**Creation of Preprocessing Pipeline**

For optimal data management (numerical and text) execution and stage preparation, a single pipeline was designed intelligently: It performs the following automatically in the pipeline.

1. **Text Processing**: From the clean\_posts column using two methods TF-IDF and Count Vectorizer, additionally a list of stop words, also (hey and mbti) were given to the vectorizers for more construction.
2. **Selection of Better Numerical Features**: All 21 numerical features were used initially. In the pipeline included, using numerical features from MinMaxScaler to the range [0,1] and then using ANOVA F-test, 10 numerical features were selected automatically, which have the highest correlation with the target variable to help increase model efficiency and reduce noise.
3. **Class Imbalance Management**: To counter classes, a technique of random under-sampling was added to the pipeline from this technique to remove random samples from the majority class in the data to prevent model bias towards imbalanced training.

**7.2 Comparison and Testing of Different Models**

In this stage, with two types of textual vectorization (TF-IDF and Count Vectorizer), several layered algorithms were placed for testing consisting of:

* **Logistic Regression**: With three types of regularization: simple, Lasso (L1), and Ridge (L2).
* **Multinomial Naive Bayes**: A simple and fast layer for textual data that is very suitable.
* **Random Forest**: A tree-based learning that decides on a model with complexities.

Then, for each model, relative to 80% training and 20% testing from each of the four dimensions, using appropriate evaluation metrics and new training data, its performance was measured.

**Analysis of Evaluation Results**

The results obtained from the model evaluations show different:

* **Better Performance in T/F Dimension**: All models showed their best performance in predicting the thinking vs feeling dimension with ROC-AUC scores around 0.87 and Geometric Mean around 0.79, which indicates that in this dimension (between languages) there is a very marked and expressible personality preference.
* **The Most Challenging Dimension J/P**: In contrast, the dimension of judging vs perceiving was lower and more difficult, with the highest scores dedicated to their data being somewhat specialized.
* **Final Model Selection**: After comparing the overall results, the model with logistic regression setting using Ridge (L2) and TF-IDF vectorization was selected, which provides the most stable and better results continuously. Therefore, this model was titled the final models for each of the four personality dimensions and the project was selected.

**7.3 Analysis of Important Features in Final Models**

After selecting the final models, another run on the entire training data for each personality dimension model, coefficients were extracted to be analyzed with multipliers. Important words for each personality dimension allocation were identified: the most indicative words for each.

* **Introverted (I) and Extroverted (E)**: Words like crazy, awesome, bored, and fun were related to extroverted, while words like dream, quiet, family, and mind referred to introverted.
* **Sensing (S) / Intuitive (N)**: Words human, world, and idea with being intuitive (N) and words like stereotype, type, and sensor with being sensing (S) were related.
* **Thinking / Feeling**: This dimension clarified words. It had the highest scores feel, love, beautiful, heart with feeling (F) and words boring, plan, efficient, knowledge with thinking (T) in correlation.
* **Judging (J) / Perceiving (P)**: Words indicating structure and planning like plan, others, help with judging (J) and informal and accepting words like shit, yeah, lazy, whatever with perceiving (P) were related.

Finally, the four final testing models (one for each personality dimension) were stored in files (.joblib) for use in future predictions and ready for new ones.

**8. Final Model Testing**

In this stage, to evaluate the real performance of the selected and trained final models from the previous section, no testing is performed in any case so that we pay for unseen evaluation or training stage.

This work is done in two ways: first, on the holdout dataset (Holdout) that was set aside at the beginning of the project, and then on a real text for personality type prediction request.

**8.1 Performance Evaluation on Holdout Dataset**

Including the holdout dataset 1% of the initial data that was completely isolated for unbiased evaluation and finals from performance without them.

**Testing Process**:

All preprocessing and engineering features that were applied to the training were also applied to this new dataset precisely.

This process included:

1. Creation of four binary labels for each personality dimension.
2. Complete text cleaning.
3. Sentiment score calculation.
4. Part-of-Speech Tagging (POS Tagging) averages and grammatical sets.
5. All counting features (word count, links, metadata like emojis, etc.).

After preparation, the four final models (.joblib) that were stored were loaded and for each, combined with another to form the final personality types and perform the prediction.

**Analysis and Results**:

Accuracy on the holdout data for each personality dimension is shown below:

* **Introverted/Extroverted**: Accuracy 69.0%
* **Intuitive/Sensing**: Accuracy 66.7%
* **Feeling/Thinking**: Accuracy 77.0%
* **Judging/Perceiving**: Accuracy 54.0%

The performance on the model is well compatible with the previous testing stage reports. This topic shows that the model may suffer from overfitting (generalization capability and acceptable without being on the holdout) somewhat new.

In the model, similar to the previous stage, the T/F dimension and better performance in allocation dimension J/P with lower (weaker) indication of its performance from the text.

**8.2 Building a System for Prediction from Input Text Request**

To apply that was designed, a system that receives a piece of text can predict its personality type, which is applicable.

**Sample Test**:

For displaying the system's performance, a sample text with real ISTP was given to it:

"I plugged the data into tableau to see how the different features or how various mathematical formulas relate to the Weight. Once I had a few that didn’t have a wide distribution, I just started trying different models, even ones we hadn’t gone over yet. There are a LOT of regression models. I do not like this try everything method, it’s inefficient and illogical."

The system's final prediction was INTP.

**Analysis Result**: Three dimensions out of four dimensions (I, T, and P) were correctly identified, but it erred in the S/N dimension, mistaking sensing (S) for intuitive (N). The presence of words like "data", "formulas", "illogical", "models" which are likely to have balance and conceptual loads relative to them shows a good example of confusion in intuitive patterns. Such points of weakness and strength indicate the model's performance.

**9. Final Conclusions**

In this project, a complete system for predicting MBTI personality types from text was designed and implemented successfully. The system combines techniques from engineering features, natural language processing, and machine learning algorithms, capable of achieving acceptable results. The analysis of some dimensions (thinking/feeling) can be predicted with higher accuracy from text, while others are more challenging.

For the future, this project can take into account improvements:

* **Use of Advanced Models**: Testing complex models based on gradients like XGBoost or deep learning models like LSTM formulas and can help extract more complex language patterns.
* **Increase and Balance Data**: The largest limitation of this project is the small and imbalanced dataset, which can be collected more specifically for rarer types to improve model performance.
* **Model Deployment**: Simple application storage models can be implemented in a web template to receive users' text and predict their personality type.

This text calculation analysis can provide a powerful tool for psychological complexities of humans in the field and provide more exciting research opportunities.