Predicting Personality from Tweets using Machine Learning

Nathan Holmes NHOLMES@EMAIL.UNC.EDU
Namita Krishna NAMITAK@EMAIL.UNC.EDU
Andrew Mu ANDYLMU@EMAIL.UNC.EDU
Nabeel Rahman NRAHMAN@UNC.EDU
University of North Carolina at Chapel Hill

Abstract

This study explores machine learning's application to analyze social media, specifically Tweets, for precise Myers-Briggs Type Indicator (MBTI) personality identification. Using four diverse machine learning models, we evaluate and compare their efficacy in understanding the nuanced relationship between social media language use and personality traits, offering practical implications for both machine learning and social sciences.

Keywords: MBTI Personality, Social Media, Machine Learning

1. Introduction

Classification is a fundamental domain in machine learning research, permeating our daily lives through advanced models that process data with refined sophistication. From discerning spam emails to categorizing patient diagnoses, we are exposed to the practical applications of these advanced technologies daily, whether we realize it or not. This project delves into the realm of social media, leveraging a diverse dataset of Tweets that encapsulate both qualitative and quantitative dimensions. Within this paper, we undertake the construction and evaluation of accuracy in identifying MBTI types based on users' tweet content. Our approach involves employing a K-Nearest Neighbor (KNN) classifier model, a Logistic Regression (LR) model, a Multinomial Naive Bayes (MNB) model, and a Random Forest (RF) classifier model to discern nuanced patterns and trends.

2. Data Collection and Model Development

Our dataset, obtained from Kaggle, consists of a collection of Tweets paired with their corresponding user's MBTI types. This provided us with diverse and invaluable data for identifing patterns between tweet content and a user's personality. After downloading the dataset, we employed preprocessing techniques, including tokenization and vectorization using tools like TfidfVectorizer from scikit-learn[1]. Subsequently, we split the dataset into training and testing sets, facilitating the development of various classification models. The K-Nearest Neighbors classifier, Logistic Regression, Multinomial Naive Bayes, and Random Forest classifier were trained and evaluated to capture nuanced patterns in the data, and each underwent fine-tuning of hyperparameters. Accuracy metrics and confusion matrices were then generated, providing a comprehensive assessment of each model's performance in predicting personality types from social media content.

1. In the case of the Multinomial Naive Bayes model, CountVectorizer was used instead of TfidfVectorizer to improve the model's accuracy

3. Evaluating Model Results

To assess the performance of our varying classification models, we focus on two key metrics: accuracy and confusion matrices. Both of these metrics involve measurements of true positive (TP), true negative (TN), false positive (FP), and false negative (FN).

The accuracy of a model is determined by the ratio of correctly predicted instances to the total number of instances:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

Confusion matrices are visual representations of a model's performance, providing insights into the number of true positive, true negative, false positive, and false negative predictions by class. This allows us to see which MBTI types we are identifying correctly, and which ones we are confusing. We also take this a step further, and with each model calculate the distribution of incorrectly predicted MBTI types across the number of traits, or letters, they are away from being correct.

3.1. Multinomial Naive Bayes

The Multinomial Naive Bayes (MNB) model, well-suited for text classification tasks, emerged as a fitting choice for our Twitter dataset. Rooted in Bayes' theorem, the probability estimation equation for MNB forms the core of our implementation:

$$P(\mathbf{x}|y, \boldsymbol{\theta}) = \prod_{i=1}^{n} \frac{N_{yi} + \alpha}{N_y + \alpha \cdot N_{\text{total}}}$$

Here, the probability of a given tweet belonging to a personality class given model parameters, $P(\mathbf{x}|y, \boldsymbol{\theta})$, is equal to the product ($\prod_{i=1}^{n}$) of all possible probability estimates of a word being in tweets by a specific MBTI type ($N_{yi} + \alpha$) out of the total count of all words in tweets of that personality type ($N_y + \alpha \cdot N_{\text{total}}$). Both the numerator and the denominator include the Laplace smoothing parameter, α.

We utilized CountVectorizer to convert text data into feature vectors, focusing on term frequencies in each document. To optimize the model's performance, hyperparameter fine-tuning was implemented, specifically exploring different values for the Laplace smoothing parameter α. Laplace smoothing, crucial for handling the zero probability problem that occurs when the model encounters words it was not trained on, was determined to be most effective with α set at 0.1.

In our evaluation, the MNB model demonstrated an accuracy of 0.48 in predicting Myers-Briggs personality types based on tweet content. Complementing accuracy, a confusion matrix visually represented the model's performance across various personality types, offering a nuanced understanding of the model's strengths and potential areas for improvement. For example, this matrix revealed that 57.93% of all mistakes were incorrect by only a single MBTI trait.

3.2. K-Nearest Neighbors

The choice of the K-Nearest Neighbors (KNN) model was driven by its simplicity, interpretability, and suitability for text classification tasks. The KNN algorithm uses proximity to make classifications or predictions about the grouping of an individual data point. For this study, the Euclidean distance formula was employed:

$$D(x, y) = \sqrt{\sum_{i=1}^{k} (x_i, y_i)^2}$$

This formula calculates the geometric distance between two points, where each data point corresponds to a user's set of features. These features are derived from the users' tweets after applying TF-IDF (Term

Frequency-Inverse Document Frequency), a method of text vectorization that transforms raw textual data into numerical features. This approach considers both the frequency of terms in a document and their importance across the entire corpus, providing a nuanced representation of the text content.

The key parameter in the KNN model is k, representing the number of neighbors considered during prediction. A grid search was conducted to determine the optimal k value for the given dataset. Consequently, a value of 21 was chosen based on the highest accuracy achieved during the tuning process.

Ultimately, the KNN model achieved an 46% accuracy when predicting Myers Briggs personality types using information derived from tweet content. Upon performing further analyses into the data provided by the confusion matrix for this model, we found once again that more than half of the misclassifications (about 53.7% this time) could be attributed to discrepancies in only a single MBTI trait.

3.3. Random Forest Analysis

We also used the Random Forest (RF) model, which operates by constructing multiple decision trees during the training phase and outputting the class that is the mode of the classes from individual trees. Like we did with the KNN model, we used TF-IDF vectorization to convert text data into quantifiable features.

A pivotal aspect of decision tree construction within the RF model is the Gini impurity, a criterion for gauging the probability of a specific feature being wrongly classified when chosen randomly. The Gini impurity I_G at a node is calculated as follows:

$$I_G(p) = 1 - \sum_{i=1}^{J} p_i^2 \tag{1}$$

Here, J represents the total number of classes, and p_i denotes the fraction of elements belonging to class i in the subset. This measure helps in determining the feature splits that form the decision nodes of the trees, playing a crucial role in the model's predictive power.

Our implementation of the RF model achieved an accuracy of 56% and, consistent with our other models, a significant portion of the misclassifications, accounting for 50% of all incorrect predictions, was observed to be just one letter off from the actual MBTI type.

3.4. Logistic Regression

The Logistic Regression (LR) model was chosen for its versatility and effectiveness in text classification tasks. Rooted in statistical modeling, the LR model estimates the probability of a particular tweet belonging to a specific personality class. The probability estimation equation for LR can be expressed as:

$$P(y|\mathbf{x}, \boldsymbol{\theta}) = \frac{1}{1 + e^{-\boldsymbol{\theta} \cdot \mathbf{x}}}$$

Here, $P(y|\mathbf{x}, \boldsymbol{\theta})$ represents the probability of a tweet being classified as a certain personality type given the model parameters ($\boldsymbol{\theta}$).

To convert the text data into meaningful features, we employed the TF-IDF vectorization technique. Next, in the quest to optimize model performance, hyperparameter tuning was executed by exploring an expanded parameter grid. The parameters considered included the regularization strength (C), the maximum number of iterations (max_iter), and the solver algorithm. The expanded parameter grid led to an increase in accuracy, reaching a plateau at approximately 66%.

This higher accuracy in comparison to our other models showcases the efficacy of the LR model in predicting Myers-Briggs personality types based on tweet content. Furthermore, the confusion matrix-provided insights into the model's performance across various personality types revealed that this model also exhibited a balanced prediction profile, with the 55.6% of misclassifications wrong by only a single MBTI trait. This emphasizes the model's capability to capture subtle distinctions within the diverse range of personality types.

4. Results

While our Logistic Regression model came out on top, all models were able to generate accuracies greater than 45%, a noteworthy accomplishment given the complexity of the classification task. We also consistently saw through confusion matrices that mistakes made by models were, more often than not, confusing two MBTI personality types that only differed by a single trait. This pattern underscores the nuanced nature of personality type classification and suggests a high degree of similarity among certain MBTI types, as reflected in their linguistic expression on social media. The confusion matrix for our LR model can be seen in Figure 1, while a table providing further insights into the distribution of incorrectly predicted MBTI types across the number of traits it differs from the correct type by is provided in Table 1.

Taking the findings of the LR model a step further, we generated a feature importance analysis to reveal that certain keywords and phrases were either more or less predictive of specific MBTI types than others, which is presented in Figure 2. These findings open up new avenues for exploring the interplay between language use and personality traits, offering valuable contributions to fields like psycholinguistics and social media analytics.

5. Tables and Figures

Letters Off	Incorrect %	Total %
1 Letter Off	55.63%	19.08%
2 Letters Off	30.76%	10.55%
3 Letters Off	12.61%	4.32%
4 Letters Off	1.01%	0.35%

Table 1: Distribution of incorrect traits for LR model

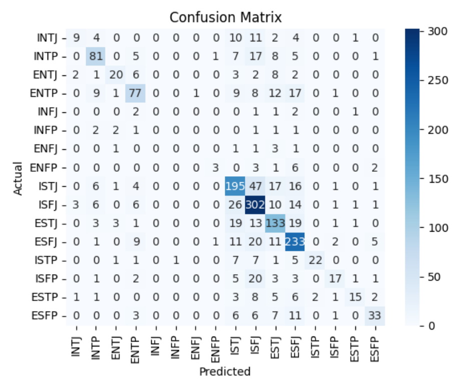

Figure 1: Confusion matrix for LR model

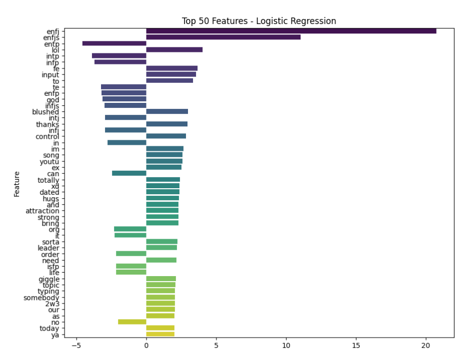

Figure 2: Feature importance analysis for LR model

.

6. References

J, M. (2017). (MBTI) Myers-Briggs Personality Type | Kaggle.
Accessed 8 December 2023

K-Nearest Neighbor(KNN) algorithm. (2023, November 9). GeeksforGeeks.
GeeksforGeeks. https://www.geeksforgeeks.org/k-nearest-neighbours/.
Accessed 8 December 2023

Silva, J. (2023). COMP 562 Lecture 5. COMP 562- University of North
Carolina at Chapel Hill.

Silva, J. (2023). COMP 562 Lecture 7. COMP 562- University of North
Carolina at Chapel Hill.

Silva, J. (2023). COMP 562 D-Trees Lecture 3. COMP 562- University of
North Carolina at Chapel Hill.