# Improving Machine Learning Methods to Enhance Prediction Accuracy of MBTI Dataset

Kaitao Yan
*Hainan International College,Minzu University of China, 27 Zhongguancun South Street, Haidian District, Beijing, China*

Keywords: Myers-Briggs Type Indicator (MBTI), Model Development, Hyperparameter Optimization, Natural Language Processing (NLP).

Abstract: This study presents an optimized machine learning approach to enhance the accuracy and generalization of predicting the Myers-Briggs Type Indicator (MBTI) dataset from Kaggle. Improvements across several modules—namely data preprocessing, feature engineering, model selection, and training methods—resulted in an increase in the accuracy of the original K-Nearest Neighbors (KNN) model from 30% to 45%. Key enhancements in this study include the use of a Term Frequency-Inverse Document Frequency Vectorizer (TfidfVectorizer) instead of a Count Vectorizer for more precise feature extraction, the refinement of text processing through a customized stop word list and a pattern-based token signifier, and the optimization of data processing. Additionally, a comparative analysis of various classification models, such as Support Vector Machines (SVMs) and Random Forest models, is conducted to validate the performance of the improved KNN model across several metrics. The advancements of the enhanced KNN model underscore the effectiveness of the optimization strategy in improving the original KNN model's ability to accurately predict MBTI personality types.

## 1 INTRODUCTION

As most of you know, the Myers-Briggs Type Indicator (MBTI) is heavily used as a personality assessment tool in the field of personal growth, industry inquiries or team development (Tareaf, 2023). It categorizes people into different personality types, 16 in total. That is, how people express certain traits of themselves such as introversion and extroversion, feeling and intuition, thinking and feeling, judging and perceiving, judging and perceiving (Myers & McCaulley, 1989). From the Social Media Trends Report published by Communications of the Association for Computing Machinery, there are 3.8 billion active social media users globally until January 2020, which is expected to grow by 9.2% per year (Violino, 2020). This means that as social media becomes more popular, the rate of information growth will increase. People use social media to share a wide variety of aspects of their daily lives, work experiences, and study progress, and in some scenarios, this information can be used to describe an individual's behavior and their personality (Christian, Suhartono, Chowanda, & Zamli, 2021).

Over time, the interest of researchers in the fields of natural language processing and social sciences has also come to the aspect of automatic personality prediction on the use of social media (Nguyen, Doogruöz, Rosé, & De Jong, 2016). Using machine learning methods, researchers have successfully analyzed and forecasted MBTI personality types by processing text data. Machine learning techniques have played a crucial role in MBTI prediction, starting with the early work by Golbeck et al., who leveraged user-displayed information on Twitter to classify MBTI types (Golbeck, Robles, Edmondson, & Turner, 2011). Later, Hernandez and Knight explored advanced neural network models, including Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM), to build a more sophisticated MBTI predictor using data from social platforms (Hernandez & Knight, 2017). This progression highlights the growing influence of machine learning in MBTI-related research.

In fact, the KNN model is widely used as a basic model for MBTI prediction, but there is a certain gap in accuracy compared to the Bidirectional Encoder Representations from Transformers (BERT) model . This research aims to enhance the model’s



Yan and K.
Improving Machine Learning Methods to Enhance Prediction Accuracy of MBTI Dataset.
DOI: 10.5220/0013526200004619
In *Proceedings of the 2nd International Conference on Data Analysis and Machine Learning (DAML 2024)*, pages 466-472
ISBN: 978-989-758-754-2

prediction accuracy on the widely-used Kaggle MBTI dataset by refining crucial stages, including data preprocessing, feature extraction, model selection, and hyperparameter optimization, bringing its performance closer to that of fine-tuned KNN and weighted KNN algorithms (Shafi, 2021).

# 2 LITERATURE REVIEW

In the process of making performance improvements to the model, the first thing focus on is the data preprocessing part. According to Dr. Christine P. Chai’s research, data preprocessing is essential for preparing databases for modeling and plays a pivotal role in influencing the outcomes of Natural Language Processing (NLP) tasks (Chai, 2022). Indeed, it has also been shown that removing noisy information from text (e.g., Uniform Resource Locator (URLs),HyperText Markup Language (HTML) tags, and special characters) can significantly improve the accuracy of models (Adnan & Akbar, 2019a). In particular, removing such irrelevant textual information enables machine learning models to focus more on meaningful features (Adnan & Akbar, 2019b). Moreover, eliminating stop words is regarded as a vital process for minimizing the dimensionality of text data (Alshanik, Apon, Herzog, Safro, & Sybrandt, 2020). Commonly used tools such as Natural Language Toolkit (NLTK) provide predefined deactivation word lists, but recent studies have shown the growing importance of customizing deactivation word lists based on specific datasets, which is further confirmed by a study published by Dr. Marcellus Amadeus in 2023 (Amadeus & Cruz Castañeda, 2023).

In addition to the data preprocessing part, this research also noticed the lack of feature extraction and improved it. In text classification, CountVectorizer and TfidfVectorizer are two commonly used feature extraction methods. However, CountVectorizer simply calculates the frequency of occurrence of words in the text, while TfidfVectorizer weights words according to their importance in the document (Suryaningrum, 2023). Studies have shown that the Term Frequency-Inverse Document Frequency (TF-IDF) method usually performs better than simple word frequency counting when dealing with low-frequency but meaningful words (Dai et al., 2024). And Dr. Ron Keinan in his 2024 study further illustrated the role of n-grams in capturing contextual information, which is important for improving classification accuracy (Keinan, 2024).

It is well-known that K Nearest Neighbors (KNN), Random Forests, and Support Vector Machines (SVMs) are widely utilized models for text classification, each exhibiting distinct advantages and limitations.KNN is recognized for its straightforward nature, yet it becomes computationally intensive when applied to large datasets. Moreover, as the sample size approaches infinity, its error rate converges to the Bayesian optimal level (Zhang, 2024).Shichao Dr. Zhang's research shows that KNN performs poorly when dealing with high-dimensional data. By comparison, Random Forest and SVM demonstrate superior performance in handling high-dimensional feature spaces, particularly for tasks related to text classification (Shah, Patel, Sanghvi, & Shah, 2020). Hence, a comparative approach was employed to enhance the performance of KNN methods through this strategy.

# 3 OVERVIEW OF THE METHODOLOGY

## 3.1 Data preprocessing

In order to process our MBTI dataset obtained on Kaggle so that it conforms to the model and is free of anomalous data,as shown in Figure 1,this research first focused on the processing of irrelevant information. This research adopted the strategy of removing URLs, HTML tags, punctuation marks, and numbers with the expectation that these steps would improve the neatness and validity of the text. Next, this research improved the word segmentation tool. Unlike the traditional segmentation methods, this research adopt more flexible tools, such as regular expression-based segmentation methods, to improve the effectiveness of segmentation. Moreover, a tailored set of stop words is incorporated to enhance text processing accuracy. To ensure better code readability and maintainability, the steps of data preprocessing are organized into modular functions.

## 3.2 Feature Extraction

In the feature extraction method, this research have adjusted it by switching the word frequency statistics method originally used to a method based on word frequency-inverse document frequency. This method

not only reduces the weight of high-frequency words and highlights the importance of low-frequency words, but also effectively reduces the dimensionality of the feature space, thus avoiding dimensionality catastrophe while ensuring classification accuracy. This improvement enhances the effect of text feature extraction. Simultaneously, this research fine-tuned the parameters, expanded the inclusion of word n-grams (such as bigrams and trigrams), and strengthened the model's capability to identify crucial low-frequency features, thereby boosting prediction accuracy.

### 3.3 Model Training and Selection

As shown in Figure 1, after completing the optimisation of the feature extraction method, this research will further adjust the model and found that the original model was deficient in hyperparameter settings and optimized it. To this end, this research used a grid search approach to tune the key parameters in the K nearest neighbor algorithm and found the best combination of parameters that could improve the classification accuracy. At the same time, this research used a multi-model comparison strategy. Besides enhancing the K-nearest-neighbor algorithm, this research incorporated random forest and support vector machine models for comparative analysis. The KNN classifies data points based on their distance, random forest mitigates high-dimensional noise through multiple decision trees, and SVM optimizes the classification by determining the best hyperplane to separate data in high-dimensional space.Through performance comparison, this research can thoroughly assess how each model performs in MBTI-type prediction tasks, allowing for more effective model optimization.

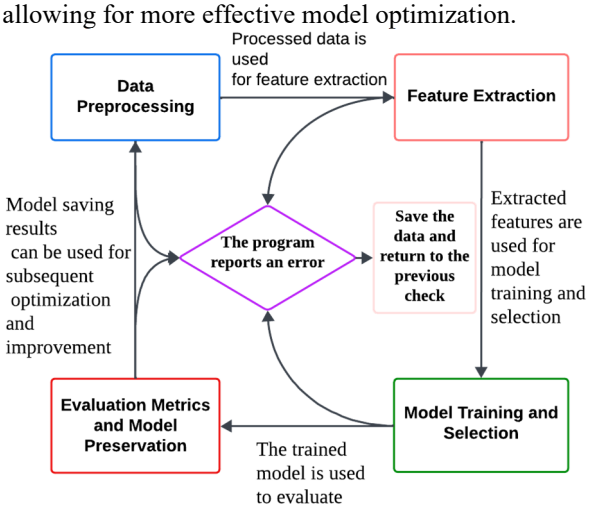


Figure 1: methodology flow chart.

### 3.4 Evaluation metrics and model preservation

As shown in Figure 1, during the last step of the evaluation process of the operating procedure, this research generated reports highlighting essential metrics such as precision, recall, and F1 scores to give a more comprehensive view of model performance, instead of depending only on accuracy. This research stored the outputs of multiple models, such as K Nearest Neighbors, Random Forests, and Support Vector Machines, and performed data visualization to present the detailed reports more clearly and concisely, facilitating future model comparisons and refinements.And, as shown in Figure 1, through discussion and research, the final evaluation results generated by the model will return to influence the individual optimisation steps so that they can continue to be improved.

## 4 EXPERIMENT

### 4.1 Implementation Details

In this experiment, this research optimized the original model based on the original model and used the improved model to process the MBTI dataset. The original model mainly used basic data preprocessing and feature extraction methods, such as simply removing URLs and employing word frequency statistics to extract text features. In contrast, in the improved model, this research introduced a variety of optimization measures.

In terms of data processing, this research first used a custom regular expression to remove HTML tags and combined it with a more flexible segmentation method to segment the text. This research used a customized list of deactivated words while loading common English deactivated words from an external deactivation thesaurus and transforming them into a collection, thus effectively removing important common words that are irrelevant to classification. Next, following the concept of TF-IDF, a feature extraction technique is applied to transform the preprocessed text into numerical representations, improving the model's capability to capture relevant features.

In the model optimization process, this research not only use the improved K-nearest neighbor classifier, but also introduce several classifiers for comparison experiments, including random forest and support vector machine models. Throughout the

training phase, this research utilized a grid search strategy to optimize the model's hyperparameters and ultimately identified the optimal configuration by exploring various hyperparameter combinations and applying cross-validation to assess each setup's performance.

## 4.2 Dataset

This research used the popular MBTI dataset from Kaggle for this study. The dataset consists of 8,675 entries, with each record including an individual's MBTI personality type (represented by a 4-letter MBTI code) along with their 50 latest posts (separated by the symbol '|||' with three spaces between them). A significant portion of the data originates from the PersonalityCafe forum, where numerous users share their MBTI personality types along with their posts. The content of these posts varies significantly in length, from brief sentences to long paragraphs. These data provide a rich source of information for MBTI type prediction tasks. However, there is a large amount of textual noise and unstructured information in the dataset, so it is crucial to preprocess the data. Our main data preprocessing steps include removing URLs, HTML tags, punctuation, and numbers, followed by word splitting and applying a customized list of deactivated words. The cleaned data is used for feature extraction and model training.

## 4.3 Metrics

Model performance is primarily measured using accuracy. Furthermore, a classification report that includes precision, recall, and F1 score is utilized to provide a comprehensive evaluation of the model's classification effectiveness. A detailed explanation of these metrics is provided below.

Accuracy is defined as the proportion of correctly classified samples over the total number of samples in the dataset, i.e:

$$\text{Accuracy}=\frac{\text{True Positives}+\text{True Negatives}}{\text{Total Samples}} \quad (1)$$

In simple terms, the accuracy rate indicates how many predictions the model made correctly overall.

Precision measures the percentage of instances predicted as positive that are actually positive. In other words, it is the ratio of true positives to the total predicted positives, i.e:

$$\text{Precision}=\frac{\text{True Positives}}{\text{True Positives}+\text{False Positives}} \quad (2)$$

Precision focuses on the proportion of correctly predicted positive samples, which indicates the accuracy of the model's predictions for the positive class.

Recall represents the ratio of true positive samples accurately identified as positive by the model out of all actual positive samples. In other words:

$$\text{Recall}=\frac{\text{True Positives}}{\text{True Positives}+\text{False Negatives}} \quad (3)$$

Recall measures the proportion of actual positive samples that the model correctly detects, indicating the model's sensitivity.

The F1 score represents the harmonic mean of precision and recall, calculated using the following formula:

$$\text{F1} = 2 \times \frac{\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \quad (4)$$

The F1 score serves as a harmonic mean of precision and recall, reflecting both precision and sensitivity. A significant gap between precision and recall results in a lower F1 score, while a closer balance between the two leads to a higher value.

In order to facilitate the comparison, this study firstly added the code based on the original model to output the classification reports, on the basis of obtaining these detailed report data, this study used python programming method to do the data visualisation and analysis, similar to the comparison between the old and the new KNN models in Figure 2, according to the comparison between these evaluation metrics and the models as well as the curve graphs, the metrics of the models can be compared more clearly , observing the differences between the old and new models.

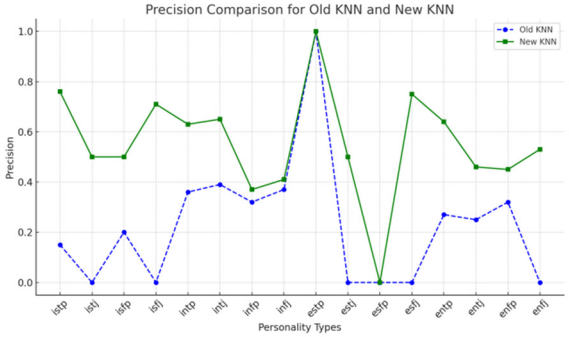


Figure 2: Comparison of Model Detailed Indicator Data.

# 5 RESULTS AND DISCUSSION

During data preprocessing, this research implemented extra text cleaning steps, including eliminating URLs and HTML tags, which are crucial for enhancing text clarity and information quality. Furthermore, replacing the initial tokenization method with a pattern-based tokenizer increased the flexibility of word segmentation, allowing it to better process complex text patterns. The customized deactivated word set of the improved model further improves the accuracy of text cleaning, which is especially important when dealing with large amounts of data with high data noise.

Second, this research upgraded the feature extraction method from simple CountVectorizer to TfidfVectorizer, and captured more text features with ngram_range=(1, 3) and max_features=10000. The advantage of Tfidf feature extraction is that it can better identify important words in the text, not because of the frequency of certain words. The advantage of Tfidf feature extraction is that it can better recognize the important words in the text, and will not lose the ability to capture other important information because of some common words with high word frequency. After the improvement, the model shows better generalization and robustness in feature extraction.

In this study, the original code was modified to produce more detailed outputs, such as accuracy and classification reports that incorporate precision, recall, and F1 scores, enabling a more in-depth evaluation of each model's performance. Additionally, this research evaluated and compared the performance of KNN, Random Forest Classifier (RFC), and SVM models to assess the extent of performance improvement.

KNN model: As can be seen from Figure 6,the accuracy of the optimized KNN model reached 44.61%, showing a notable improvement compared to the original model's 33.49%. And compared to the old model, the new model also improves on various indicators.However,as can be seen in Figure 2 and also Figure 3, from the classification report of precision and recall, it can be seen that the precision of some categories is still 0, which indicates that the model still has obvious under-prediction when dealing with some of the categories, and needs to be further improved in subsequent research.

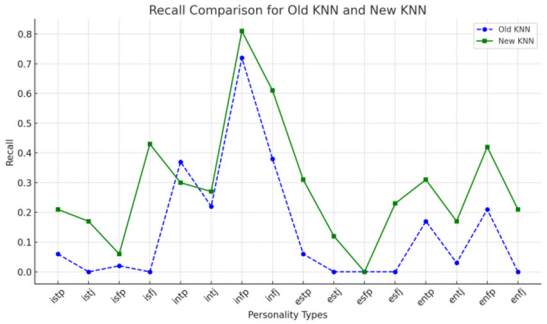


Figure 3: Comparison of detailed metrics data on recall for the three models.

Random forest model: As shown in Figure 5, the Random Forest model has an accuracy of 55.22% and performs effectively in dealing with data complexity and maintaining balance. It is worth noting that its performance in terms of recall is better than its performance on the F1-Score metric, and its overall performance is stronger than the KNN model and lower than the SVM model.

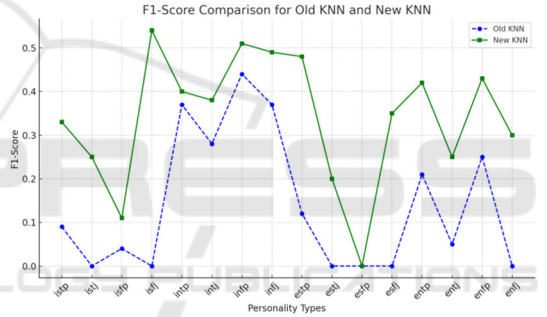


Figure 4: Comparison of F1-Score Detailed Metrics Data for Three Models.

SVM model: As can be seen in Figure 5, the Support Vector Machine model outperforms the other models with an accuracy of 64.03%. The performance on different categories is balanced, with high F1 scores in most cases, and the values of all indicators are relatively close to each other, resulting in an overall excellent performance.

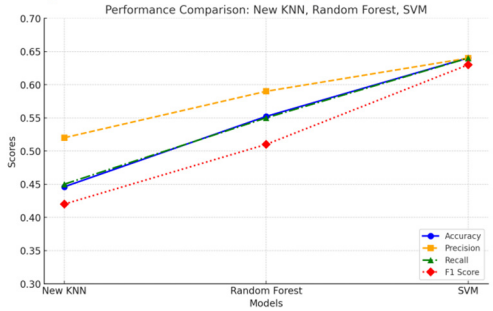


Figure 5: Evaluation results for the three models—KNN, SVM, and Random Forest—using consistent metrics.

Considering the macro average and weighted average values, the enhanced model demonstrates notable gains in precision, recall, and F1-score. The macro average precision increased from 23% to 55% compared to the original model, while the weighted average precision rose from 31% to 64%. This suggests that the enhanced model adapts more effectively to complex textual data and achieves superior classification results even with imbalanced category distribution.

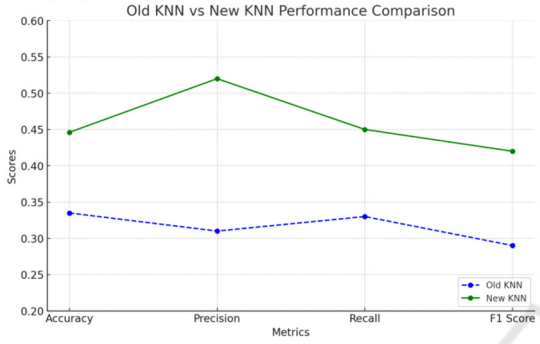


Figure 6: Detailed Output Data Report for Random forest Models.

## 6 CONCLUSIONS

The enhanced KNN model shows notable advancements in text preprocessing, feature extraction, and model selection, leading to a substantial increase in MBTI prediction accuracy. While some categories still exhibit lower precision and recall, comparing it with other models clearly reveals a significant boost in overall performance, including higher classification accuracy and improved generalization. This suggests that the optimization strategy used in this study effectively enhances the KNN model's ability to classify complex text data, making it more suitable for predicting user MBTI in intricate text scenarios.

## REFERENCES

Adnan, K., Akbar, R., 2019. An analytical study of information extraction from unstructured and multidimensional big data. *Journal of Big Data*, 6(1), 1-38.

Adnan, K., Akbar, R., 2019. Limitations of information extraction methods and techniques for heterogeneous unstructured big data. *nternational Journal of Engineering Business Management*, 11, 1847979019890771.

Alshanik, F., Apon, A., Herzog, A., Safro, I., Sybrandt, J., 2020, December. Accelerating text mining using domain-specific stop word lists. In *2020 IEEE International Conference on Big Data (Big Data)*, 2639-2648. IEEE.

Amadeus, M., Castañeda, W. A. C., 2023. Clustering Methods and Tools to Handle High-Dimensional Social Media Text Data. In *Advanced Applications of NLP and Deep Learning in Social Media Data*, 36-74. IGI Global.

Chai, C. P., 2023. Comparison of text preprocessing methods. *Natural Language Engineering*, 29(3), 509-553.

Dai, S., Li, K., Luo, Z., Zhao, P., Hong, B., Zhu, A., Liu, J., 2024. AI-based NLP section discusses the application and effect of bag-of-words models and TF-IDF in NLP tasks. *Journal of Artificial Intelligence General Science (JAIGS)*, 5(1), 13-21.

Golbeck, J., Robles, C., Edmondson, M., Turner, K., 2011, October. Predicting personality from twitter. In *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing*, 149-156. IEEE.

Christian, H., Suhartono, D., Chowanda, A., Zamli, K. Z., 2021. Text based personality prediction from multiple social media data sources using pre-trained language model and model averaging. *Journal of Big Data*, 8(1), 68.

Hernandez, R. K., Scott, I., 2017, December. Predicting Myers-Briggs type indicator with text. In *31st Conference on Neural Information Processing Systems (NIPS 2017)*.

Shafi, H., 2021. A machine learning approach for personality type identification using MBTI framework. *Journal of Independent Studies and Research Computing*, 19(2).

Keinan, R., 2024. Sexism identification in social networks using TF-IDF embeddings, preprocessing, feature selection, word/Char N-grams and various machine learning models in Spanish and English. *Working Notes of CLEF*.

Amirhosseini, M. H., Kazemian, H., 2020. Machine learning approach to personality type prediction based on the Myers–Briggs type indicator®. *Multimodal Technologies and Interaction*, 4(1), 9.

Nguyen, D., Doğruöz, A. S., Rosé, C. P., De Jong, F., 2016. Computational sociolinguistics: A survey. *Computational Linguistics*, 42(3), 537-593.

Tareaf, R. B., 2022, December. MBTI BERT: A Transformer-Based Machine Learning Approach Using MBTI Model for Textual Inputs. In *2022 IEEE 24th Int Conf on High Performance Computing & Communications; 8th Int Conf on Data Science & Systems; 20th Int Conf on Smart City; 8th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys)*, 2285-2292. IEEE.

Shah, K., Patel, H., Sanghvi, D., Shah, M., 2020. A comparative analysis of logistic regression, random

forest and KNN models for the text classification. *Augmented Human Research*, 5(1), 12.

Suryaningrum, K. M., 2023. Comparison of the TF-IDF method with the count vectorizer to classify hate speech. *Engineering, Mathematics and Computer Science Journal (EMACS)*, 5(2), 79-83.

Santini, R. M., Salles, D., Tucci, G., Ferreira, F., Grael, F., 2020. Making up audience: Media bots and the falsification of the public sphere. *Communication Studies*, 71(3), 466-487.

Zhang, S., 2021. Challenges in KNN classification. *IEEE Transactions on Knowledge and Data Engineering*, 34(10), 4663-4675.