Myers–Briggs Personality Classification from Text Using RoBERTa with LoRA

Seojun Ha
University of Southern Denmark
seha25@student.sdu.dk

January 31, 2026

ABSTRACT

Personality prediction from text has attracted growing interest in natural language processing, yet it remains challenging due to the abstract nature of personality traits and the noise inherent in self-reported labels. This study investigates whether transformer-based language models can infer Myers–Briggs Type Indicator (MBTI) personality types from user-generated text by formulating the task as a 16-class single-label classification problem. Using large-scale Reddit data, we conduct experiments with RoBERTa-base models fine-tuned via Low-Rank Adaptation (LoRA) under varying data sizes and training configurations, and compare the results against an unfine-tuned baseline. The best-performing model achieves an accuracy of 30.77%, compared to 3.01% for the unfine-tuned baseline, with higher performance observed in experiments.

1 Introduction

The Myers–Briggs Type Indicator (MBTI) is widely used in certain cultural contexts, particularly in South Korea, where it frequently appears in everyday social interactions and online communication. In contrast, MBTI is far less prominent in many Western countries, where it is often regarded as a casual or non-scientific personality test. One possible explanation for this discrepancy lies in differences in language use and self-expression. Textual data, especially on online platforms, reflects personal preferences, emotional tendencies, and communication styles, all of which may correlate with personality traits. With the recent success of large-scale language models, an important research question emerges: to what extent can personality-related information be inferred from natural language alone?

The MBTI is a self-report questionnaire that assigns individuals to one of 16 personality types using a four-letter code derived from four dichotomous dimensions: Extraversion/Introversion (E/I), Sensing/Intuition (S/N), Thinking/Feeling (T/F), and Judging/Perceiving (J/P) [1].

Text is one of the most natural ways humans express themselves, and personality often appears directly in word choice and style, emotional patterns, etc. If a model can detect personality traits from the text alone, it can open up possibilities to improve personalized recommendation systems, mental-health tools, and social analysis.

However, predicting MBTI types from text remains a highly challenging task. Linguistic differences between many MBTI categories are often subtle, resulting in ambiguous boundaries that are difficult for models to distinguish. Certain dimensions, such as Judging versus Perceiving, correspond to behavioral preferences that may not consistently manifest in written language. Furthermore, personality is a multifaceted construct influenced by context and situational factors, which cannot be fully represented by limited text samples alone.

Motivated by these challenges, this study investigates whether modern transformer-based language models can extract meaningful personality-related signals from text, despite the inherent noise and ambiguity of MBTI labels. We formulate MBTI prediction as a 16-class single-label classification task and evaluate the effectiveness of parameter-efficient fine-tuning using Low-Rank Adaptation (LoRA)[2].

2 Related Work

Personality prediction from text has been an active research area in computational linguistics and social media analysis for more than a decade. Early studies demonstrated that linguistic features such as word choice, syntactic patterns, and emotional expressions are correlated with psychological traits. For example, Pennebaker et al. [3] showed that function words and affective language can reveal stable personality characteristics, laying the foundation for text-based personality inference. Subsequent work extended these ideas to online platforms, including blogs and social media, where large-scale textual data became available.

More recent studies[4] have focused specifically on predicting personality typologies such as MBTI from user-generated text, often framing MBTI prediction as a multi-class classification problem with sixteen personality types. Early approaches relied on traditional machine learning models such as support vector machines and Naive Bayes classifiers, using bag-of-words or TF–IDF features. With the rise of deep learning, convolutional and recurrent neural networks were later applied, achieving moderate improvements. However, many studies report persistent difficulties in distinguishing closely related MBTI types and handling severe class imbalance, particularly for underrepresented personality categories.

Transformer-based language models, including BERT[5] and RoBERTa[6], have recently become the dominant paradigm for text classification tasks due to their strong contextual representations. Nevertheless, their application to personality prediction remains challenging. MBTI labels are often self-reported and noisy, and fine-grained personality distinctions may not be consistently reflected in short text samples. In addition, full fine-tuning of large transformer models can be computationally expensive and prone to overfitting, particularly when training data is highly imbalanced.

Among these methods, Low-Rank Adaptation (LoRA) offers an attractive trade-off between computational efficiency and task adaptation. However, despite its growing adoption in other NLP tasks, very limited research has investigated the use of LoRA for personality classification. Motivated by this gap, this study explores MBTI classification using a RoBERTa-base model fine-tuned with LoRA, with a particular focus on the impact of data quality and class imbalance.

3 Dataset and Preprocessing

3.1 Dataset

We used two datasets throughout the study. For Experiments 1–5, we used the "Reddit MBTI Dataset" from Kaggle [7], which contains over 13 million posts from 11,773 unique authors. For Experiment 6, we additionally used the *kaggle-mbti-cleaned* dataset on Hugging Face [8] (409k posts) to assess whether performance was driven by data quality rather than model configuration. All experiments used subsets after preprocessing, length filtering, and split-specific sampling.

Table 1: Comparison of MBTI class distributions

(a) Reddit MBTI dataset

MBTI	Number of Posts	Percentage (%)
ENFJ	271,523	2.08
ENFP	411,270	3.16
ENTJ	628,984	4.83
ENTP	1,254,839	9.63
ESFJ	31,530	0.24
ESFP	23,204	0.18
ESTJ	42,405	0.33
ESTP	105,075	0.81
INFJ	2,096,064	16.09
INFP	2,988,437	22.94
INTJ	1,662,731	12.76
INTP	2,702,219	20.74
ISFJ	142,497	1.09
ISFP	79,389	0.61
ISTJ	125,930	0.97
ISTP	462,538	3.55
Total	**13,028,635**	**100.00**

(b) Kaggle MBTI text dataset

MBTI	Number of Posts	Percentage (%)
ENFJ	9,102	2.22
ENFP	32,015	7.81
ENTJ	10,986	2.68
ENTP	32,988	8.05
ESFJ	1,995	0.49
ESFP	2,137	0.52
ESTJ	1,871	0.46
ESTP	4,214	1.03
INFJ	69,842	17.04
INFP	86,678	21.15
INTJ	50,943	12.43
INTP	61,261	14.95
ISFJ	7,846	1.91
ISFP	12,428	3.03
ISTJ	9,601	2.34
ISTP	15,878	3.87
Total	**409,785**	**100.00**

3.2 Data Cleaning and Preprocessing

For data cleaning, although the author of the Reddit MBTI dataset performed basic preprocessing steps such as lowercasing, removing URLs, non-English characters, and filtering posts shorter than 20 characters or longer than 3,000 characters, we re-applied these procedures to ensure consistency and data quality. This additional cleaning step was intended to eliminate any remaining noise that could negatively affect model training.

Another important preprocessing step involved removing texts in which MBTI types were explicitly mentioned. Since such references may leak label information into the input text, retaining them could artificially inflate model performance and reduce the difficulty of the classification task. In addition, because all experiments were conducted with a maximum input sequence length of 256 tokens, texts exceeding this length were removed from the dataset. Rather than truncating longer posts, we chose to discard them entirely in order to reduce training cost and maintain a consistent input distribution across samples. This decision also avoids introducing systematic bias that may arise from truncating longer posts, which often contain more diverse linguistic content.

4 Method

4.1 Model Choice and Baseline Model

We formulate MBTI prediction as a 16-class single-label classification task and use RoBERTa-base as the encoder backbone. RoBERTa is a transformer encoder model that builds on BERT with an optimized pretraining strategy, yielding strong contextual representations for downstream text classification [5, 6].

As a baseline, we evaluated a RoBERTa-base model without any task-specific fine-tuning. The pre-trained backbone was kept unchanged, and a randomly initialized classification head was used to perform direct inference on the test set. No gradient updates were applied to either the encoder or the classifier. This baseline serves as a meaningful lower bound for the MBTI prediction task. While RoBERTa is pre-trained on large-scale text corpora, it does not possess inherent knowledge of personality classification. Without fine-tuning, the model cannot align its internal representations with MBTI-related linguistic cues.

Although the theoretical random-guess accuracy for a 16-class problem is 6.25%, the unfine-tuned baseline performs worse (3.01%) due to highly skewed predictions caused by an untrained classification head. This behavior is expected, as the unfine-tuned classifier lacks task-specific alignment and tends to produce highly skewed predictions toward a small subset of classes.

4.2 LoRA-based Fine-tuning

LoRA was chosen to fine-tune the RoBERTa-base model in a parameter-efficient manner[2]. Compared to full fine-tuning, LoRA trains only a small number of additional parameters while keeping the pre-trained backbone fixed, which significantly reduces memory consumption and computational cost. Full fine-tuning typically requires an order of magnitude more GPU resources, making it impractical under limited hardware constraints.

Instead of updating the full weight matrix $W \in \mathbb{R}^{d\times k}$, LoRA decomposes the update into two low-rank matrices:

$$W' = W + \Delta W = W + BA,$$

where $B \in \mathbb{R}^{d\times r}$ and $A \in \mathbb{R}^{r\times k}$ with rank $r \ll \min(d, k)$. Only A and B are trained, while W remains frozen.

This paper involved multiple experimental settings with different dataset sizes, sequence lengths, and hyperparameters. LoRA enabled efficient experimentation by allowing rapid training iterations without re-training the entire model. By maintaining a shared backbone and modifying only task-specific adapters, the approach ensured fair and consistent comparisons across experiments.

Moreover, the MBTI classification task is inherently noisy due to self-reported labels and overlapping linguistic patterns among personality types. Fully fine-tuning all model parameters may increase the risk of overfitting to such noise. In contrast, LoRA restricts adaptation to low-rank updates, encouraging the model to learn task-relevant features while preserving general language representations.

For training, we used the categorical cross-entropy loss, which is standard for multi-class single-label classification tasks. Given a training sample with ground-truth label y and predicted class probabilities $\hat{p}_i$, the loss is defined as

$$\mathcal{L} = -\sum_{i=1}^{C} y_i \log(\hat{p}_i),$$

where C denotes the number of classes (in our case, $C = 16$), and y_i is a one-hot encoded target vector.

4.3 Handling Class Imbalance: Cluster Based Downsampling

The MBTI datasets used in this study exhibit severe class imbalance, where a small number of types (e.g., INFP, INTP, INFJ) dominate the corpus. This imbalance can inflate overall accuracy while degrading minority-class performance, and it often leads the model to rely on class priors rather than learning discriminative linguistic cues. To mitigate this issue, we applied a cluster-based downsampling strategy to the training split only, while keeping the validation and test splits unchanged to preserve an unbiased evaluation setting.

Our goal was to reduce the number of samples in overrepresented classes while retaining representative examples that preserve semantic diversity. Instead of random undersampling (which may discard informative samples) or naive class truncation (which can collapse intra-class variability) [9], we adopted an embedding-space clustering approach. We first embed each training text using a sentence-level transformer encoder (all-MiniLM-L6-v2, 384-dimensional) [10, 11], followed by L_2 normalization. Because all embeddings are normalized, Euclidean distance provides a practical approximation to cosine distance, allowing efficient centroid-based selection [10].

Target size per class. Let n_c denote the original number of training samples in class c, and let $n_{\min} = \min_c n_c$. We define a target size k that is derived from the minimum class size, but constrained by lower and upper bounds to avoid overly aggressive downsampling:

$$k = \text{clip}(n_{\min},\ \texttt{MIN\_PER\_CLASS},\ \texttt{MAX\_PER\_CLASS}),$$

and the final target for each class is $k_c = \min(k, n_c)$. In our implementation, we set `MIN_PER_CLASS`=10,000 and `MAX_PER_CLASS`=120,000. Minority classes with $n_c \leq k_c$ were preserved without modification.

Two-stage centroid selection. For each overrepresented class, we perform coarse clustering with MiniBatch K-Means to partition the embedding space into k_1 clusters:

$$k_1 = \min(\texttt{KMEANS\_STAGE1},\ n_c),$$

where `KMEANS_STAGE1`=3000. We then allocate the target budget k_c proportionally across clusters based on cluster sizes. For each cluster, we select the samples whose embeddings are closest to the cluster centroid (using Euclidean distance), i.e., we pick the most central examples per cluster. This ensures that the retained subset covers the major semantic regions of the class distribution, rather than being dominated by a narrow writing style or a small subtopic.

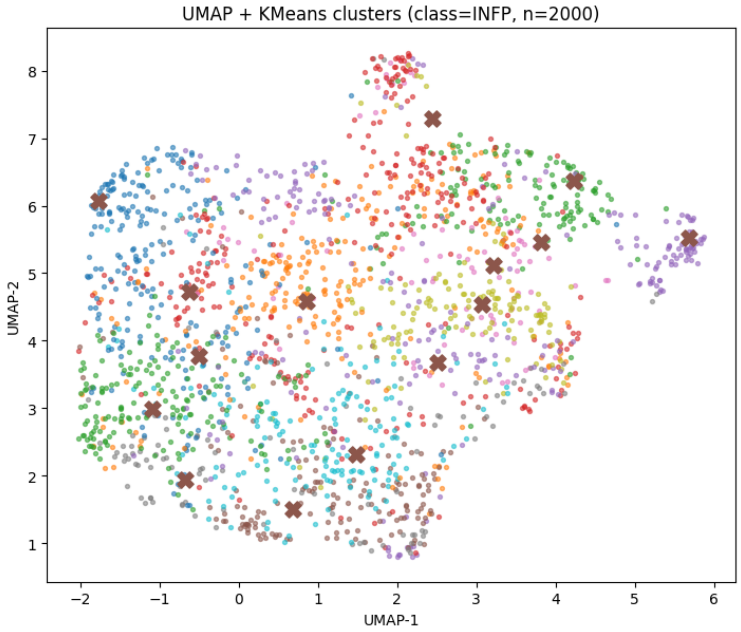

Figure 1: UMAP visualization of sentence embeddings for the INFP class before downsampling. Each point represents a text instance, and colors indicate clusters obtained by MiniBatch K-Means in the embedding space. The figure illustrates the semantic diversity within a single MBTI class, motivating the use of cluster-based downsampling instead of random undersampling.

Dataset size before and after downsampling. At the corpus level, the original dataset contained a total of 13,028,635 text instances. After applying the cluster-based downsampling procedure to the training split, the dataset was reduced to 129,106 instances for the training set. Despite this substantial reduction, the downsampling strategy was designed to retain representative samples from each MBTI class by selecting semantically central instances in the embedding space. The resulting dataset enabled efficient experimentation under limited computational resources while preserving diversity in linguistic patterns.

5 Results

A clear performance gap is observed between experiments trained on large but noisy datasets and those trained on smaller, cleaner data. Despite using fewer samples, Experiment 6 outperforms experiments trained on millions of posts, suggesting that data quality plays a more critical role than data quantity in this task. Earlier experiments (e.g., Exp 3 and Exp 5) exhibit accuracy close to random guessing, highlighting the negative impact of severe class imbalance and insufficient signal-to-noise ratio.

Table 2: Training setup for all experiments

Exp	Dataset Size	Max Seq	LoRA	Epochs	Batch Size	LR	GPU
1	31,658	128	8/16	5	8 (16 eval)	2e-5	T4
2	2,546,635	256	16/32	4	32 (64 eval)	5e-5	A100
3	129,106	256	16/32	3	32 (64 eval)	5e-5	A100
4	2,546,635	256	16/32	3	32 (64 eval)	5e-5	A100
5	129,106	256	16/32	3	32 (64 eval)	5e-5	A100
6	**293,381**	**256**	**32/64**	**5**	**8 (16 eval)**	**2e-4**	**T4**
Baseline	**2,546,635**	**256**	**X**	**X**	**32 eval**	**X**	**T4**

About Experiments 1 and 2. Weighted-F1 scores and axis-level accuracies are omitted for Experiments 1 and 2, as these experiments were conducted as preliminary exploratory studies and were not designed for detailed metric reporting.

Table 3: Overall classification performance across experiments

Exp	Accuracy (%)	Macro-F1	Weighted-F1	E/I Acc	N/S Acc	T/F Acc	P/J Acc
1	19.31	0.0261	–	–	–	–	–
2	30.61	0.1028	–	–	–	–	–
3	6.77	0.0439	0.0747	0.5692	0.3338	0.5790	0.5112
4	28.47	0.0833	0.2408	0.7874	0.9259	0.5989	0.5987
5	6.75	0.0434	0.0745	0.5675	0.3349	0.5789	0.5100
6	**30.77**	**0.1952**	**0.2731**	**0.7795**	**0.8672**	**0.6568**	**0.6296**
Baseline	**3.01**	**0.0037**	**0.0018**	**0.7693**	**0.1367**	**0.5400**	**0.6035**

Axis-level accuracies further reveal that the model captures certain personality dimensions more effectively than others. In particular, the E/I and N/S axes consistently achieve higher accuracy, while T/F and P/J remain more challenging. This aligns with the intuition that introversion–extraversion and sensing–intuition distinctions are more strongly reflected in linguistic style, whereas decision-making and lifestyle preferences are harder to infer from text alone.

Despite achieving moderate accuracy, the overall Macro-F1 and Weighted-F1 scores remain relatively low across most experiments. This discrepancy can be largely attributed to the severe class imbalance present in the dataset, where a small number of dominant MBTI types account for a substantial proportion of samples. In such settings, accuracy may be inflated by correct predictions on majority classes, while F1-scores more strongly penalize poor performance on underrepresented types. As a result, the low F1 values indicate that the model struggles to generalize across all sixteen personality categories evenly, despite improvements in overall accuracy.

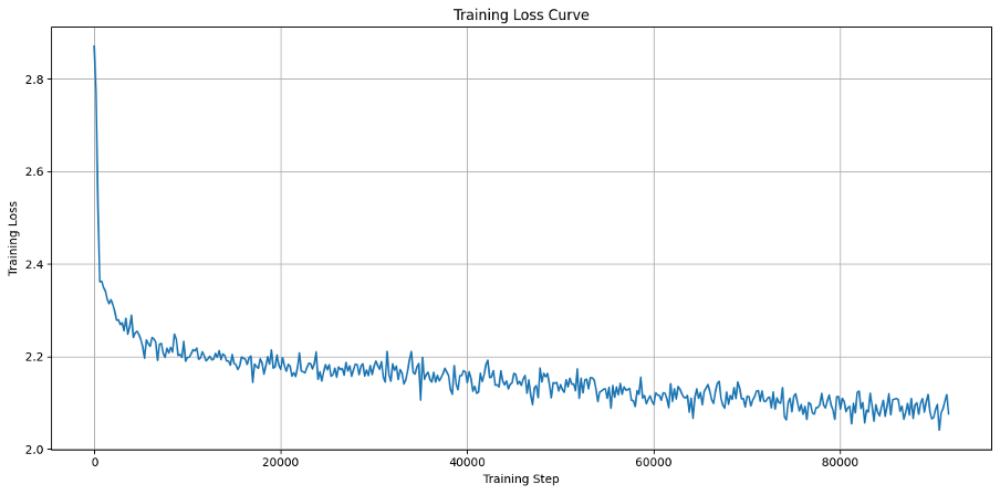

Figure 2: Loss Curve

The training loss curve shows a steady and meaningful decrease as the number of training steps increases. During the early phase of training, the loss drops rapidly, indicating that the model is quickly learning general language patterns and adapting the pre-trained RoBERTa representations to the MBTI classification task. After approximately 10,000–20,000 steps, the curve begins to flatten, and the loss decreases more gradually with small oscillations. This behavior suggests that the model has entered a fine-tuning stage, where improvements become incremental as it tries to capture more subtle personality-related linguistic cues.

In later stages, the loss stabilizes without strong signs of divergence or overfitting, suggesting that the learning rate and LoRA configuration were appropriate for this training setup. The overall downward trend confirms that the optimization process was effective and that the model continued to improve throughout training, even though the noisy and imbalanced nature of the dataset likely limited how much the loss could decrease.

6 Discussion

6.1 Key Findings

The key findings of this study can be summarized as follows:

- Transformer-based models fine-tuned with LoRA are able to extract meaningful personality-related signals from text, achieving up to 30.77% accuracy in a challenging 16-class MBTI classification setting, significantly outperforming an unfine-tuned baseline.
- Data quality was found to be more critical than data quantity. Models trained on smaller but cleaner datasets consistently outperformed those trained on much larger, noisier corpora.
- Severe class imbalance substantially limited model performance, as reflected by low Macro-F1 scores, indicating that gains in accuracy were largely driven by majority MBTI types.
- Axis-level evaluation revealed that certain personality dimensions (E/I and N/S) are more readily inferred from linguistic patterns than others (T/F and P/J), suggesting uneven linguistic observability across MBTI axes.

6.2 Interpretation

Overall, this study achieved an accuracy of 30.77% on a challenging 16-class MBTI classification task. Despite this improvement, the model struggled to achieve balanced performance across all personality types, as reflected by consistently low F1-scores. The results suggest that extreme class imbalance and aggressive downsampling may have reduced the discriminative signals associated with certain MBTI categories. As a consequence, some personality types may have lost fine-grained linguistic characteristics during preprocessing, limiting the model's ability to distinguish them reliably.

6.3 Limitations

There are several limitations in the current approach that likely affected the model's performance. First, the MBTI labels in the Reddit dataset are entirely self-reported rather than professionally validated. Because of this, the data inevitably contains noise, inconsistencies, and possible mislabeling that the model cannot correct. Second, the dataset suffers from a strong class imbalance. Types such as INFP, INTP, and INFJ appear far more frequently than others, and this imbalance makes it difficult for the model to learn rare classes, even after cleaning or downsampling efforts.

Another limitation lies in the task formulation itself. Predicting all sixteen MBTI types from text compresses complex human behaviors into a rigid set of categories, which reduces both interpretability and learnability. A multi-axis approach might have been more appropriate, but the current project did not have the computational resources to explore this direction fully.

The nature of Reddit text adds further challenges. Posts often contain sarcasm, slang, fragmented writing, and highly informal expressions. These characteristics make it difficult for even advanced transformer models to infer stable personality traits from textual behavior. For all these reasons, the model's predictions tend to capture broad tendencies rather than precise MBTI types, suggesting that personality classification from text remains a fundamentally noisy and constrained task.

6.4 Future Work

While this research provides an initial exploration of MBTI prediction using transformer-based models, several promising directions remain for future research. One natural extension would be to reformulate the task into four independent binary classification problems corresponding to the MBTI axes: E/I, N/S, T/F, and J/P. Such a formulation could allow models to focus more precisely on linguistic cues associated with each personality dimension and may lead to more interpretable and robust predictions when combined in a multi-task or ensemble framework.

Further improvements could also be achieved through more extensive experimentation with model configurations and training strategies. In particular, systematic exploration of hyperparameters such as learning rate schedules, LoRA rank configurations, batch sizes, and input sequence lengths may yield additional performance gains. Given the strong impact of data quality observed in this study, future work could also investigate more advanced techniques for handling class imbalance, including controlled oversampling strategies or the use of synthetic data generated by large language models. Such approaches may help alleviate the severe imbalance present in Reddit-based MBTI datasets while enabling more controlled experimental conditions.

Finally, access to larger and more stable computational resources would make it possible to scale experiments to larger models, conduct more thorough ablation studies, and evaluate the generalizability of findings across multiple datasets. Pursuing these directions could lead to a more accurate, fair, and interpretable framework for predicting personality from text.

6.5 Reflection

Reflecting on this research, several design choices stand out as areas where a different approach might have led to clearer insights. One key lesson was the impact of task formulation on both model behavior and evaluation stability. Framing MBTI prediction as a 16-class classification problem exposed the model to severe class imbalance and overlapping linguistic patterns, which substantially influenced performance metrics and error distributions. Although this formulation allowed for a direct mapping to MBTI types, the experimental results revealed how sensitive transformer-based models are to data quality, label noise, and evaluation design. In particular, relying on a single train–validation split limited the reliability of performance estimates, especially for underrepresented classes. This highlighted the importance of robust evaluation strategies, such as stratified cross-validation, when working with highly imbalanced datasets.

More broadly, this project emphasized that effective NLP research is not solely driven by model choice, but by alignment between research questions, data characteristics, and experimental design. Through this experience, we gained a clearer understanding of how early modeling decisions shape downstream results and how careful problem formulation and evaluation planning are critical for drawing meaningful conclusions.

7 Conclusion

Our study examined whether modern NLP techniques can infer MBTI personality types from free-form text. Although MBTI is not scientifically rigorous, its cultural relevance (especially in communities like Reddit and in countries such as Korea) makes it an intriguing target for computational analysis. Through several experiments using RoBERTa with

LoRA fine-tuning, we explored how data size, data cleanliness, and training configurations influence the model's ability to capture personality-related linguistic signals. Across all experiments, models trained on extensive but noisy data often struggled, while the cleaned dataset, despite being much smaller, produced the strongest results. The best-performing model achieved an accuracy of 30.77% in a 16-class classification task, significantly surpassing the random baseline of 6.25%. Axis-level evaluation provided more nuanced insight that the model captured E/I and N/S distinctions relatively well, while T/F and J/P remained much harder to detect. This suggests that certain personality traits may be reflected more consistently in linguistic style, whereas others are too subtle to appear reliably in short textual samples.

These findings highlight both the potential and the limitations of personality prediction from text. The task remains fundamentally difficult due to overlapping linguistic patterns among MBTI types, and the inherent ambiguity of personality expression in natural language. Nevertheless, the results show that transformer-based models can extract meaningful signals under the right conditions. With more balanced datasets, author-level aggregation, or a multi-task framework that incorporates auxiliary linguistic features, future work could improve both performance and interpretability.

Overall, this research provided a deeper understanding of the challenges in modeling human traits through NLP and demonstrated how careful data preparation and thoughtful experimentation can significantly influence model behavior. The insights gained here lay a foundation for future exploration into computational personality analysis.

References

[1] R. A. Woods and P. B. Hill, "Myers-briggs type indicator," in *StatPearls*, Treasure Island, FL: StatPearls Publishing, 2022. Updated September 18, 2022.

[2] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," *arXiv preprint arXiv:2106.09685*, 2021.

[3] J. W. Pennebaker, M. R. Mehl, and K. G. Niederhoffer, "Psychological aspects of natural language use: Our words, our selves," *Journal of Personality and Social Psychology*, vol. 85, no. 2, pp. 291–301, 2003.

[4] S. S. Keh and I.-T. Cheng, "Myers-briggs personality classification and personality-specific language generation using pre-trained language models," *arXiv preprint arXiv:1907.06333*, 2019.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2019.

[6] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized BERT pretraining approach," *CoRR*, vol. abs/1907.11692, 2019.

[7] M. Zhang, "Reddit mbti dataset." `https://www.kaggle.com/datasets/minhaozhang1/reddit-mbti-dataset`, 2020. Accessed: 2026-01.

[8] Shunian, "kaggle-mbti-cleaned." `https://huggingface.co/datasets/Shunian/kaggle-mbti-cleaned`, 2023. Hugging Face Dataset. Accessed: 2026-01.

[9] N. Chawla *et al.*, "Smote: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, 2002.

[10] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *EMNLP*, 2019.

[11] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. Zhou, "Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers," *CoRR*, vol. abs/2002.10957, 2020.