COGNITIVE ALIGNMENT IN PERSONALITY REASONING: LEVERAGING PROTOTYPE THEORY FOR MBTI INFERENCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Personality recognition from text is typically cast as hard-label classification, which obscures the graded, prototype-like nature of human personality judgments. We present PROTOMBTI, a cognitively aligned framework for MBTI inference that operationalizes prototype theory within an LLM-based pipeline. First, we construct a balanced, quality-controlled corpus via LLM-guided multi-dimensional augmentation (semantic, linguistic, sentiment). Next, we LoRA-fine-tune a lightweight ($\leq$2B) encoder to learn discriminative embeddings and to standardize a bank of "personality prototypes". At inference, we retrieve top-k prototypes for a query post and perform a retrieve–reuse–revise–retain cycle: the model aggregates prototype evidence via prompt-based voting, revises when inconsistencies arise, and, upon correct prediction, retains the sample to continually enrich the prototype library. Across Kaggle and Pandora benchmarks, PROTOMBTI improves over baselines on both the four MBTI dichotomies and the full 16-type task, and exhibits robust cross-dataset generalization. Our results indicate that aligning the inference process with psychological prototype reasoning yields gains in accuracy, interpretability, and transfer for text-based personality modeling.

1 INTRODUCTION

Understanding a user's personality is a core enabler of personalized AI: it lets systems tailor content and interaction style to individuals rather than rely on one-size-fits-all heuristics. In education, personality-aware tutors adjust pacing and feedback framing to sustain engagement and improve outcomes Sajja et al. (2023); in recommendation, personality signals inferred from everyday text mitigate cold-start and disambiguate intent when history is sparse He et al. (2018); Chen et al. (2012); Li et al. (2023); and in organizational decision support, personality-sensitive analysis of internal communications informs team formation while respecting individual styles Wang (2024). Across these settings, a common requirement is to infer *enduring, person-level dispositions* directly from language produced in the wild, without administering standalone psychometric tests.

Within this context, Myers-Briggs Type Indicator (MBTI) serves as a pragmatic operational code for personality-aware NLP: it is widely understood and used in practice and online communities Myers (1962); McCrae & Costa (1989). The goal here is to infer a user's MBTI type (four dichotomies yielding sixteen types) by extracting latent psychological traits from user-generated text Pan & Zeng (2023); Li et al. (2025); Gjurković et al. (2020). Prior approaches fall into three families. (i) *Lexicon* methods hand-engineer features (e.g., LIWC) and train shallow classifiers Taboada et al. (2011); Komisin & Guinn (2012). (ii) *Neural* models (CNN/RNN/Transformer) learn text representations end-to-end and improve accuracy Ryan et al. (2023); Ashraf et al. (2024); Patil et al. (2024); Zhu et al. (2024b); Shobha et al. (2024). (iii) *LLM-based* methods use prompting, few-shot exemplars, or augmentation to perform zero/few-shot prediction Li et al. (2024); Hu et al. (2024); Li et al. (2025), with recent systems exploiting contrastive objectives, multi-view signals, or knowledge-enhanced reasoning Hu et al. (2024); Bi et al. (2025); Ma et al. (2022); Yang et al. (2021a).

Rosch's *prototype theory* is well-acknowledged in cognitive psychology and asserts that people categorize by comparing to central exemplars, or *prototypes*, rather than applying strict rules Rosch & Mervis (1975). This view aligns with MBTI's graded dichotomies where personality are captured by

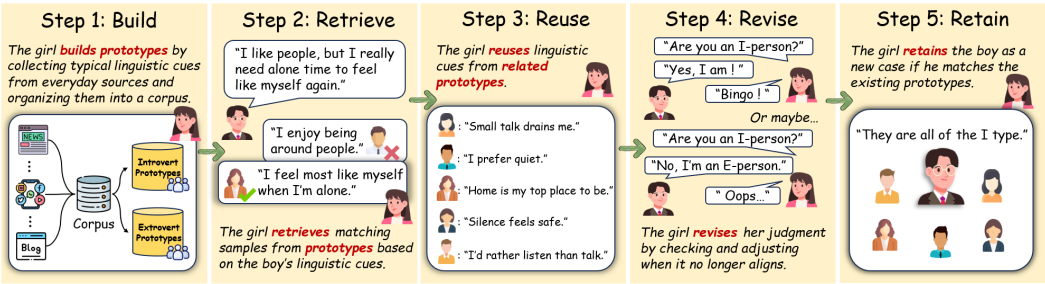

Figure 1: **Illustration of the Retrieve–Reuse–Revise–Retain cycle in ProtoMBTI.** Step 1 (**Retrieve**): match query cues with prototypes. Step 2 (**Reuse**): reuse retrieved patterns as evidence. Step 3 (**Revise**): adjust predictions for consistency. Step 4 (**Retain**): store verified cases to enrich the prototype library. This cycle shows how ProtoMBTI operationalizes case-based reasoning under prototype theory.

preference scales rather than binary categories Myers (1985); McCrae & Costa (1989), and suggests representing personality types via prototypes and reasoning by similarity to them. Large Language Models (LLMs) are well suited to such prototype-guided inference; recent studies indicate that prototype conditioning can improve both performance and faithfulness across tasks Zhu et al. (2024a); Deng et al. (2024); Wei et al. (2025); He et al. (2025); Ren et al. (2024). However, existing studies on text-based MBTI prediction have not aligned with this psychological intuition, treating MBTI labels as fixed categorical targets. We therefore investigate *how prototype-based reasoning may be integrated into LLMs to improve MBTI prediction from text*. Concretely, we ask: (1) how to construct operational personality prototypes consistent with psychological theory, and (2) how to retrieve and integrate them during inference to yield predictions that are accurate, interpretable, and transferable across datasets. We investigate these questions in the setting of MBTI inference from social-media posts.

To answer these questions, we present **ProtoMBTI**, a prototype-based reasoning framework for MBTI inference from social-media text. Our central contribution is to replace flat label prediction with cognitively aligned, exemplar-driven inference. Concretely, we (i) learn a standardized *personality prototype bank* by LoRA-tuning a compact ($\leq$2B) encoder that embeds posts and type descriptors in a shared space; (ii) perform *retrieve–reason–revise–retain* inference, retrieving top-k prototypes and aggregating their evidence via *similarity-weighted voting* with cross-dichotomy consistency checks, then adaptively retaining correct cases to refine the prototype bank over time; and (iii) curate data with class-balanced, multi-dimensional LLM augmentation (semantic, linguistic, sentiment) under automatic quality filtering. Across the well-established Kaggle and Pandora benchmarks, PROTOMBTI surpasses strong neural and LLM baselines on both dichotomy-level and 16-type evaluation. On Kaggle, it achieves an average accuracy of 85.14% across the four dimensions, exceeding prior work by 7.35%. Under distribution shift, *it demonstrates superior cross-dataset transfer*, reaching 96.41% average accuracy on the Pandora test set, which is 30.64% higher than previous results. Moreover, it yields case-based rationales at a fraction of the compute of very large LLMs, and the analysis of results remains aligned with existing psychological insights.

2 RELATED WORK

Predicting MBTI Personality Types from Text. Automatically detecting personality from text has become an increasingly prominent focus in computational psycholinguistics. Although the Big Five framework still dominates in psychological research John & Srivastava (1999), the MBTI Myers (1962); McCrae & Costa (1989) remains widely used in online communities, self-assessment platforms, and workplace settings Quenk (1999). MBTI categorizes individuals into 16 types based on four dichotomies: Introversion vs. Extraversion, Sensing vs. Intuition, Thinking vs. Feeling, and Judging vs. Perceiving. Existing computational approaches to MBTI prediction can be broadly divided into three categories: *LIWC-based methods.* These approaches rely on psycholinguistic lexicons such as LIWC to extract handcrafted linguistic and psychological features Taboada et al. (2011); Komisin & Guinn (2011), which are typically combined with traditional classifiers such as

SVMs Cui & Qi (2017) and XGBoost Tadesse et al. (2018). However, they fundamentally depend on predefined rules, which conflicts with the notion of psychological reality of categories in prototype theory. *Deep learning methods.* Neural architectures such as convolutional neural networks Xue et al. (2018), recurrent neural networks Tandera et al. (2017), and hierarchical encoders are capable of automatically learning features from raw text and generally outperform LIWC-based models Ryan et al. (2023); Ashraf et al. (2024); Shanmukha et al. (2024). Pre-trained transformer models, such as BERT and RoBERTa, have also been widely adopted, either encoding user posts as single documents Jiang et al. (2020); Keh et al. (2019); Patil et al. (2024); Zhang (2023); Tareaf (2022) or within post-level hierarchical structures Shobha et al. (2024); Lynn et al. (2020), and are often combined with external features such as LIWC or graph-based context Zhu et al. (2024b); Yang et al. (2022). Recent efforts such as TrigNet Ma et al. (2022) and Transformer-MD Yang et al. (2021a) leverage hierarchical or knowledge-enriched representations. *Large language model methods.* Recent advances employ large language models (LLMs) for few-shot prediction, data augmentation, or zero-shot classification Li et al. (2024). Representative works include TAE Hu et al. (2024), MBTIBench Li et al. (2025), and ETM Bi et al. (2025), which integrate contrastive objectives or multi-view learning. Although soft-labeling trends have emerged, these methods largely neglect the prototype effect. These three paradigms mainly focus on classifying higher-level MBTI categories and do not address the more fine-grained and challenging task of predicting all 16 personality types. In addition, their designs rely on using the entire training data for prediction, which means that the model makes decisions by referencing all samples. This introduces excessive noise rather than precisely attending to the most relevant prototypes, thereby limiting performance and weakening generalization. Personality traits, by nature, should generalize across diverse contexts. To address these limitations, our proposed ProtoMBTI framework performs personality-relevance retrieval within the prototype bank, selects the prototypes most aligned with the current test input, and then delegates inference to an LLM. This process enables reliable prediction of the full 16-type classification while overcoming the aforementioned limitations and demonstrating stable generalization across datasets and tasks.

Personality Alignment and Prototype-Theoretic Approaches in LLMs. Recent work has explored aligning large language models (LLMs) with human personality traits by manipulating internal activations or identifying personality-sensitive neurons Zhu et al. (2024a); Deng et al. (2024). These approaches control how models *exhibit* personality but do not address how they *reason* about it. In parallel, prototype-based methods have been applied to improve interpretability and generalization in LLM reasoning, such as using sentence-level prototypes for faithful explanations Wei et al. (2025), building logical prototype spaces He et al. (2025), or incorporating exemplars in in-context learning Ren et al. (2024). While these studies highlight the promise of prototypes, they do not explicitly link reasoning to cognitive alignment. Our work differs by introducing the notion of *cognitive alignment*, leveraging prototype theory to align LLM reasoning processes with established psychological mechanisms, thereby enhancing interpretability and generalization in personality inference.

3 Main Inspiration

Prototype theory was first proposed by Rosch (1973); Rosch & Mervis (1975); Rosch (1975), which states that categories are organized around highly typical members (i.e., prototypes), while other members are hierarchically associated with the category according to their degree of similarity to the prototype. In MBTI personality classification, *"Extraversion" or "Introversion"* can be viewed as categories, and prototypical utterances such as "I enjoy going out with friends" for extraversion or "I prefer solitude and reflection" for introversion serve as prototypes of these categories (shown in Figure 1). Building on this view, we model personality type recognition as a *prototype-driven* reasoning process. Specifically, our approach consists of two key stages: (i) constructing a prototype bank that captures the typical characteristics of different personality types, and (ii) leveraging this prototype bank for analogy and matching, thereby inferring the most probable personality type of an input.

In (1), we treat large-scale MBTI datasets as accumulated long-term experience, and use semantic embedding learning to abstract these experiences into balanced and high-quality *cognitive anchors* for each personality type. *In (2)*, individuals are assumed to first hold prototype representations in cognition. When encountering a new case, they *retrieve* the most similar prototype, *reuse* its

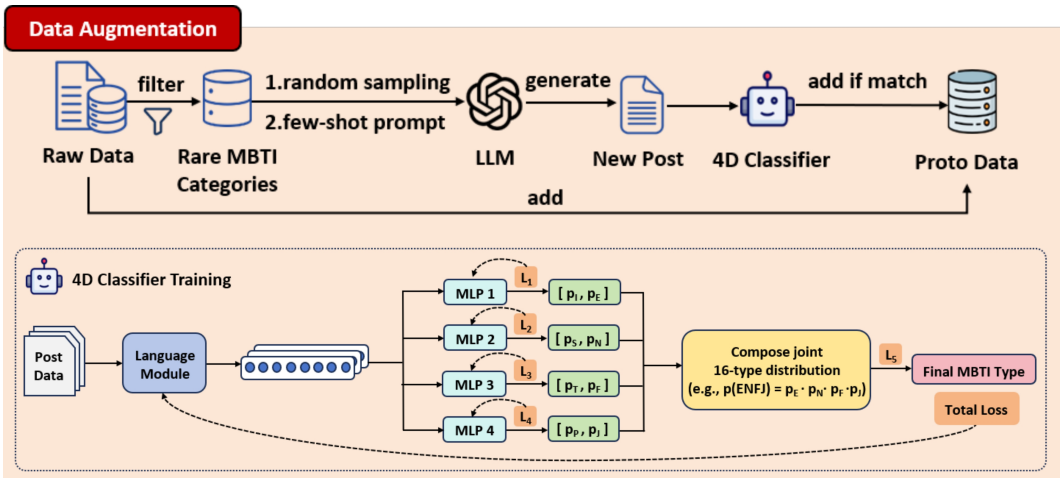

Figure 2: Overview of the LLM-driven augmentation and 4D Classifier training. Rare MBTI categories are expanded via sampling and few-shot prompting, with generated posts filtered by the 4D Classifier. Only samples with consistent four-dimension predictions are retained to form the prototype dataset. The 4D Classifier encodes text with a language module, followed by four dimension-specific heads and a joint type head, optimized with combined losses to ensure reliable filtering.

linguistic or behavioral cues for inference, *revise* the reasoning if inconsistencies arise, and *retain* the new experience once validated. This "retrieve–reuse–revise–retain" cycle closely parallels the reasoning framework of Case-Based Reasoning (CBR) Hatalis et al. (2025); Wiratunga et al. (2024); Aamodt & Plaza (1994); Kolodner (1992), but CBR does not address the cognitive internalization of cases. In this regard, our *"Prototype-Based" Reasoning (PBR) framework* establishes a personality type reasoning method that is more aligned with psychological cognitive processes.

4 Formulation

We model MBTI personality types as a hierarchically structured semantic category system. Specifically, we define four higher-level categories $\mathcal{C}^{(i)}$, where $i \in \{1, 2, 3, 4\}$ corresponds to any of the four MBTI dimensions (E/I, S/N, T/F, J/P). Each higher-level category is a binary classification space that captures cognitive preferences along its respective dimension. Furthermore, the complete MBTI category space is modeled as the Cartesian product of these four higher-level categories, i.e., $\mathcal{C}^{\text{MBTI}} = \prod_{i=1}^{4} \mathcal{C}^{(i)}$. This formulation reflects the cognitive property of *basic-level categories*: personality recognition can be conducted at a coarse granularity over the four dimensions, or at a fine granularity by distinguishing among the 16 complete MBTI types. We define a prototype p as a triplet $\langle a, e, c \rangle$ where (i) a, the *attribute*, refers to observable information such as a post text; (ii) e, the *embedding*, represents the relational features between the attribute a and the category c; and (iii) $c \in \bigcup_{i=1}^{4} \mathcal{C}^{(i)} \cup \mathcal{C}^{\text{MBTI}}$ is the *category*. For instance, if $c \in \mathcal{C}^{(1)}$, then $c \in \{\text{E}, \text{I}\}$. If $c \in \mathcal{C}^{\text{MBTI}}$, then c corresponds to one of the 16 predefined MBTI personality types.

Our Task. Given a user-generated post a, the objective is to infer the author's MBTI personality type within the hierarchical category space. Specifically, the task can be formulated at two levels of granularity: (i) at the upper level, inferring the category $c \in \mathcal{C}^{(i)}$ for each MBTI dimension $i \in \{1, 2, 3, 4\}$, corresponding to four binary classification tasks (E/I, S/N, T/F, J/P); and (ii) at the lower level, inferring the category $c \in \mathcal{C}^{\text{MBTI}} = \mathcal{C}^{(1)} \times \mathcal{C}^{(2)} \times \mathcal{C}^{(3)} \times \mathcal{C}^{(4)}$, which corresponds to the 16 fine-grained MBTI personality types.

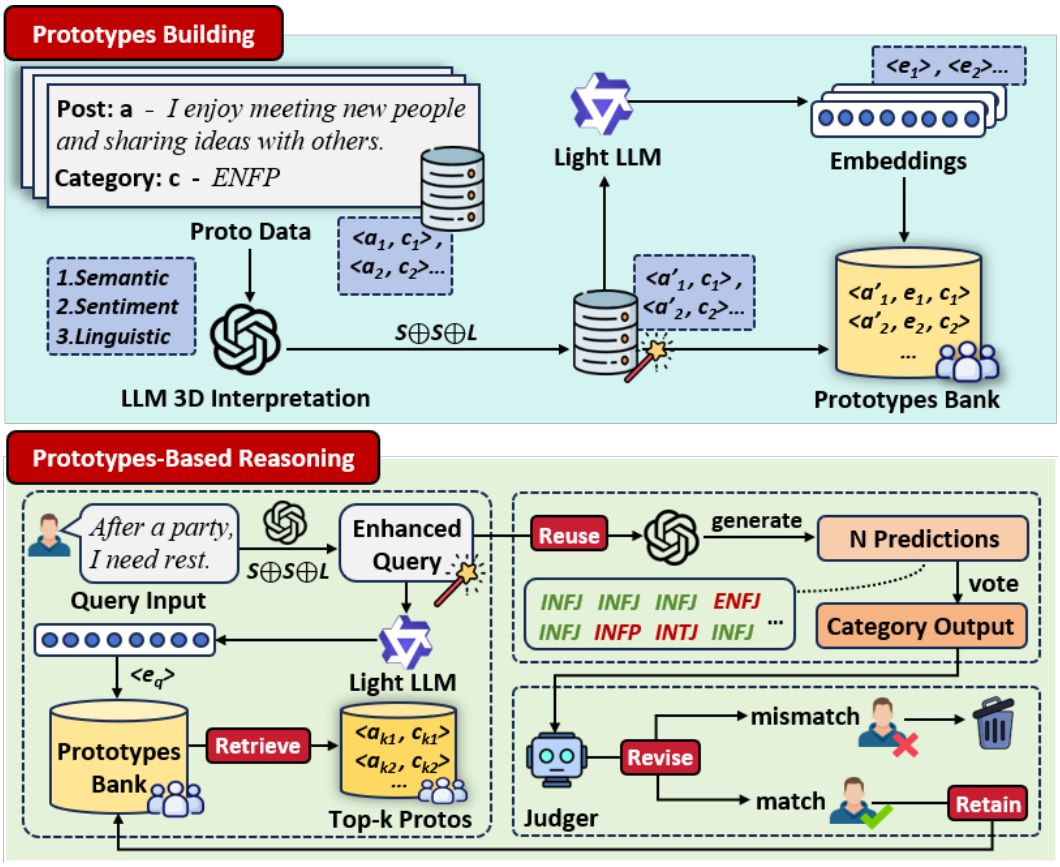

Figure 3: **ProtoMBTI framework with prototype construction and reasoning.** The upper panel shows how posts are semantically, sentimentally, and linguistically interpreted by a light LLM and stored in the Prototype Bank. The lower panel illustrates reasoning via the *retrieve–reuse–revise–retain* cycle, enabling interpretable MBTI detection through prototype-driven inference.

5 METHODOLOGY

5.1 DATA AUGMENTATION

In order to address the imbalance in category distribution and the limitations of raw sample quality, it is necessary to perform data augmentation prior to prototype construction(see more in Appendix F,Algorithm 1). Let the original labeled dataset be denoted as $U = \{\langle a, c\rangle\}$, where a is a user text and $c \in \mathcal{C}^{\text{MBTI}}$ is its personality label. A subset is denoted as $U' \subset U$. We define a data augmentation operator driven by LLMs(shown in Figure 2), $\mathcal{A} : U' \to U_{\text{proto}}$, which consists of two main stages: *(i) class-balanced augmentation with quality filtering* and *(ii) multi-dimensional augmentation*.

Class-balanced augmentation with quality filtering. To address the imbalance among MBTI categories, we first guide the LLM with prompt instructions (see Appendix G, Table 3 for details) to generate new samples $\langle a', c\rangle$ from original samples $\langle a, c\rangle$, ensuring semantic or stylistic variation while preserving category consistency. For each target class c, several category-specific prompts are designed. To ensure label consistency and usability, we employ a 4D Classifier as a quality filter(see detail in Appendix D). It consists of a pre-trained encoder with four dimension-specific heads (I/E, S/N, T/F, P/J) and one overall MBTI type head. Once trained to near state-of-the-art accuracy Lin et al. (2024), the classifier serves as a "gatekeeper" to filter LLM-generated candidates. A sample $\langle a', c\rangle$ is accepted into $\mathcal{A}_1(U')$ only if both the overall type and all four dichotomy predictions match the target label.

5.2 PROTOTYPE BUILDING

Multi-dimensional augmentation. As shown in Figure 3, suppose the original dataset has size n. After class-balanced augmentation and filtering, m new samples are added, resulting in $n + m$ samples in total. To further enrich expression, we perform *semantic*, *linguistic*, and *sentiment* augmentations Hu et al. (2024) on *all* these $n+m$ samples, obtaining attribute-extended representations a^*. This process is also guided by prompt templates (see Appendix G, Table 4). The final augmented set maintains a size of $n + m$, but each sample now has attribute-extended representations modified along semantic, linguistic, and sentiment dimensions.

Formal Representation of the Augmented Dataset. Let the original dataset size be n, and let m denote the number of additional samples obtained after class-balanced augmentation. To provide a unified formulation, we define the overall augmentation operator as $A = A_2 \circ A_1$, where A_1 represents class-balanced augmentation with quality filtering, and A_2 denotes multi-dimensional augmentation across semantic, linguistic, and sentiment dimensions. For an input sample a_j, its augmented representation is defined as $a_j^* = A(a_j)$. Thus, the final augmented dataset has a size of $n+m$, which is formally expressed as $U_{\text{proto}} = \left\{\langle a_j^*, c_j\rangle \mid j = 1, 2, \ldots, n+m\right\}$, where a_j^* denotes the attribute-extended representation of sample j produced by the operator A, and $c_j \in \mathcal{C}^{\text{MBTI}}$ represents its corresponding label. After augmentation, we conduct quality control on generated samples.

Personality Representation Learning. Let the input text be denoted as $a \in U_{\text{proto}}$, with its corresponding personality category $c \in \mathcal{C}^{\text{MBTI}}$. We define a mapping operator $\mathcal{E}$ to learn the relational embedding between text and category, $\mathcal{E} : (a, c) \mapsto r \in \mathbb{R}^d$, where r represents the semantic relation vector between the input text a and the personality category c. To achieve this, we fine-tune a compact ($\leq$2B) encoder on U_{proto} using LoRA, enabling it to capture distinctive MBTI personality features and produce the corresponding personality embedding r. Given an input a, the encoder outputs an overall MBTI type prediction $\hat{c} \in \mathcal{C}^{\text{MBTI}}$ (e.g., INFJ). Its embedding vector is represented as $r = f_\theta(a)$, where f_θ denotes the fine-tuned LLM encoder. During training, we perform supervision only at the overall 16-type classification level. The cross-entropy loss is defined as $\mathcal{L}_{\text{proto}} = \text{CE}(\hat{c}, c)$, where c is the ground-truth label. By updating parameters efficiently through LoRA, the model learns discriminative embeddings for different personality types. Each training sample $\langle a, c\rangle$ is ultimately mapped to a standardized prototype triplet, $p = \langle a, r, c\rangle$, and stored in the prototype bank $\mathcal{P}$ as $\mathcal{P} = \{\langle a, r, c\rangle \mid \langle a, c\rangle \in U_{\text{train}}\}$ (see in Appendix F,Algorithm 2)..

5.3 PROTOTYPE-BASED REASONING

Prototype Retrieval. Let the test set be denoted as $U^* \subset U$, where $U^* \cap U' = \varnothing$, which contains only the posts $post^*$ to be classified. For each $post^*$, we first apply the multi-dimensional augmentation operator A_2 defined during training to obtain the augmented representation $a' = A_2(post^*)$. We define the inference encoding operator $\mathcal{E}^*$, which maps the input a' into an overall embedding vector using the fine-tuned compact encoder, i.e., $r' = \mathcal{E}^*(a')$. We then define the similarity operator S, which measures the similarity between r' and each prototype embedding r in the prototype bank $\mathcal{P}$ using cosine similarity: $S(r', r) = \frac{r' \cdot r}{\|r'\| \, \|r\|}$. Finally, we define the prototype retrieval operator R_k, which selects the top-k prototypes most similar to r' from the prototype bank $\mathcal{P}$: $R_k(r', \mathcal{P}) = \{p_i = \langle a_i, r_i, c_i\rangle\}_{i=1}^k$.

Prototype Inference. We define the prototype inference operator $\mathcal{I}$, which is instantiated by LLMs. It takes as input the target post a' together with the retrieved prototype set $\{a_i, c_i\}_{i=1}^k$ by prompt templates (see Appendix G, Table 5), and outputs a predicted distribution $\hat{y}$, i.e., $\mathcal{I}(a', \mathcal{R}_k(r', \mathcal{P})) \mapsto \hat{y}$. We then compare the predicted result with the ground-truth category c. If the prediction is correct, i.e., $\hat{c} = c$, where $\hat{c} = \arg\max \hat{y}$, we construct a new prototype triplet $p' = \langle a', r', c\rangle$ and update the prototype bank as $\mathcal{P} \leftarrow \mathcal{P} \cup \{\, p' \,\}$(see in Appendix F,Algorithm 3)..

6 EXPERIMENTS

Research Questions. To systematically evaluate the effectiveness and cognitive plausibility of the ProtoMBTI framework, we formulate the following research questions:

- RQ1 (Section 7.1): *Does prototype-based personality detection achieve better performance than existing state-of-the-art (SOTA) models?*
- RQ2 (Section 7.2): *Can the framework's effectiveness and the contribution of prototypes to the reasoning process be empirically validated?*
- RQ3 (Section 7.1): *Do prototypes preserve generalization ability across different test sets?*

Experimental Design. To address *RQ1*, we evaluate ProtoMBTI against baselines on Kaggle and Pandora across both dichotomy and 16-type settings. For *RQ2*, we conduct ablations to examine the contribution of prototypes under varied conditions. For *RQ3*, we test cross-domain transfer between Kaggle and Pandora, assessing robustness under distribution shift. Further experimental details are provided in the Appendix E.

Datasets. We use two standard MBTI datasets: Kaggle (8,675 samples from PersonalityCafe) and Pandora (9,067 samples from Reddit) Gjurković et al. (2020). Both are split into training/validation/test sets (8:1:1). Data augmentation is applied only to training and validation, while test sets remain untouched to ensure fair generalization evaluation. Further statistics are reported in Appendix E.

Model Configurations. We evaluate ProtoMBTI under different configurations by selecting representative models for each component. For post generation and explanation, we use GPT-4o and GPT-4o-mini to compare generative models of different scales. For data augmentation, we employ BERT Devlin et al. (2019), RoBERTa Liu et al. (2019), and DeBERTa He et al. (2020) as the backbone of the 4D Classifier to control the quality of LLM-generated posts. For feature extraction, we consider DeepSeek-1B Bi et al. (2024), Qwen2.5-1.5B Bai et al. (2025), and Llama3-1B Dubey et al. (2024) to assess the effect of different encoder scales and architectures on personality representation learning. Finally, for inference, we compare GPT-4o-mini, Qwen2-72B Bai et al. (2025), and Llama3.1-70B Dubey et al. (2024) to validate the impact of different reasoning engines on final personality prediction.

Metric. We evaluate models using *accuracy* at two levels: (i) four MBTI dichotomies and their average, and (ii) all 16 MBTI types for fine-grained classification. Baseline dichotomy results are taken from prior work Hu et al. (2024); Bi et al. (2025), while 16-type results are estimated from dichotomy accuracies following MBTI classification rules (details in Appendix E).

Implementation and Environment. All experiments are implemented in PyTorch Paszke et al. (2019) with Huggingface Transformers Wolf et al. (2019). Training is conducted on NVIDIA A100 and RTX 4090 GPUs, while large-scale inference uses official APIs for reproducibility. Random seeds are fixed to ensure stability. Additional training details are provided in the Appendix E.

Baselines. We compare ProtoMBTI with a broad set of representative baselines, including traditional machine learning methods, neural network architectures, pretrained language models, and recent LLM-based approaches(see details in Appendix H). Specifically, we include: SVM Cui & Qi (2017), XGBoost Tadesse et al. (2018), BiLSTM Tandera et al. (2017), $\text{BERT}_{\text{mean}}$ Keh et al. (2019), $\text{BERT}_{\text{concat}}$ Jiang et al. (2020), AttRCNN Xue et al. (2018), AttnSeq Lynn et al. (2020), Transformer-MD Yang et al. (2021a), TrigNet Yang et al. (2021b), D-DGCN Yang et al. (2023), GPT4o, TAE Hu et al. (2024), and ETM Bi et al. (2025).

7 Results and Discussions

Overall, we answer RQ1 and RQ3 through performance comparison, and RQ2 through ablation study. In addition, we derive psychological insights from the experimental results and conduct a case study, while discussions on data augmentation, hyper-parameters, and other results are provided in Appendix I.

7.1 Performance Comparison (RQ1,RQ3)

Our proposed ProtoMBTI framework surpasses all existing methods across all metrics, as shown in Table 1, and achieves the best generalization performance on mixed datasets. *Comparison on single datasets.* For the four MBTI dimensions, $\text{ProtoMBTI}_{\text{Qwen}}$ achieves an average accuracy of 85.14% on the Kaggle dataset, significantly higher than the previous best model ETM (77.79%).

Methods	Kaggle						Pandora					
	I/E	S/N	T/F	P/J	Avg.	16-Type	I/E	S/N	T/F	P/J	Avg.	16-Type
SVM	53.34	47.75	76.72	63.03	60.21	12.32	44.74	46.92	64.62	56.32	53.15	7.64
XGBoost	56.67	52.85	75.42	65.94	62.72	14.89	45.99	48.93	63.51	55.55	53.50	7.94
BiLSTM	57.82	57.87	69.97	57.01	60.67	13.35	48.01	52.01	63.48	56.21	54.93	8.91
$\text{BERT}_{\text{concat}}$	58.33	53.88	69.36	60.88	60.61	13.27	54.22	49.15	58.31	53.14	53.71	8.26
$\text{BERT}_{\text{mean}}$	64.65	57.12	77.95	65.25	66.24	18.78	56.60	48.71	64.70	56.07	56.52	10.00
AttRCNN	59.74	64.08	78.77	66.44	67.25	20.03	48.55	56.19	64.39	57.26	56.60	10.06
SN+Attn	65.43	62.15	78.05	63.92	67.39	20.29	56.98	54.78	60.95	54.81	56.88	10.43
Transformer-MD	66.08	69.10	79.19	67.50	70.47	24.41	55.26	58.77	69.26	60.90	61.05	13.70
TrigNet	69.54	67.17	79.06	67.69	70.86	25.00	56.69	55.57	66.38	57.27	58.98	11.98
D-DGCN	68.41	65.66	79.56	67.22	70.21	24.02	61.55	55.46	71.07	59.96	62.01	14.55
D-DGCN+ℓ_0	69.52	67.19	80.53	68.16	71.35	25.64	59.98	55.52	70.53	59.56	61.40	13.99
GPT4o	65.86	51.69	78.60	63.93	66.89	17.11	55.52	49.79	71.25	60.51	59.27	11.92
TAE	70.90	66.21	81.17	70.20	72.07	26.75	62.57	61.01	70.53	59.34	63.05	15.98
ETM	68.97	71.21	86.19	84.78	77.79	35.89	68.57	64.91	66.07	63.53	65.77	18.68
$\text{ProtoMBTI}_{\text{llama}}$	81.92	87.70	86.04	82.47	84.03	71.11	69.05	68.85	68.98	70.82	69.43	50.30
$\text{ProtoMBTI}_{\text{Qwen}}$	83.74	88.10	84.54	84.18	85.14	71.42	71.63	66.98	73.25	70.33	70.55	41.86
$\text{ProtoMBTI}_{\text{GPT4o}}$	82.36	85.55	82.70	80.04	82.66	68.39	70.41	70.65	73.32	71.27	71.41	60.22
$\text{ProtoMBTI}_{\text{mix}}$	**93.54**	**93.54**	**95.69**	**93.23**	**93.50**	**85.54**	**96.25**	**97.08**	**96.75**	**95.54**	**96.41**	**92.13**
$\text{ProtoMBTI}_{\text{mix-ex}}$	91.08	90.77	94.46	90.15	91.62	81.23	90.44	91.25	91.35	90.44	90.87	81.15

Table 1: Performance comparison of ProtoMBTI and baselines on Kaggle and Pandora datasets. Metrics include four dimension accuracies, their average, and the 16-type accuracy (theoretical for baselines computed as the product of the four dimension accuracies, direct prediction for ProtoMBTI). Subscripts denote different LLMs, *mix* for same-source training/testing, and *mix-ex* for cross-source evaluation.

On the Pandora dataset, $\text{ProtoMBTI}_{\text{GPT4o}}$ reaches 71.41%, again exceeding ETM (65.77%). For the 16-type classification task, the best theoretical value reported in prior work is only 35.89% (ETM) on Kaggle, while $\text{ProtoMBTI}_{\text{Qwen}}$ achieves 71.42%, representing a remarkable improvement. These results demonstrate that under single-dataset settings, ProtoMBTI outperforms the current state-of-the-art methods in both four-dimension and 16-type classification. *Comparison on mixed datasets.* When Kaggle and Pandora are combined for training while validation and test sets remain consistent within a single dataset, $\text{ProtoMBTI}_{\text{mix}}$ performs substantially better than single-dataset training. For the 16-type accuracy, performance on Kaggle increases from 71.41% to 85.54%, and on Pandora from 60.22% to 92.13%. The model also achieves the best performance across all four MBTI dimensions under this setting. *Cross-dataset evaluation.* When validation and test sets are swapped across datasets, $\text{ProtoMBTI}_{\text{mix-ex}}$ still maintains strong generalization. For the 16-type classification, the model achieves 81.23% on Kaggle and 81.15% on Pandora. For the four-dimension average, Kaggle reaches 91.62%, only 1.88 percentage points lower than the same-domain $\text{ProtoMBTI}_{\text{mix}}$, and Pandora reaches 90.87%, 5.54 percentage points lower. Although performance decreases compared to the same-domain setting, it remains far superior to single-dataset training.

7.2 Ablation Study (RQ2)

Table 2 presents the ablation study results on the Kaggle dataset. These results highlight the central role of prototypes, data augmentation, and prototype reasoning. We take $\text{ProtoMBTI}_{\text{Qwen}}$ as the full model and compare its ablation performance on the Kaggle dataset. *First, effective prototype selection is crucial:* using random prototypes ($\text{ProtoMBTI}_{\text{RandomProto}}$) or simple semantic retrieval ($\text{ProtoMBTI}_{\text{Semantic}}$) significantly reduces the four-dimension accuracy to 79.66% and 79.50%, and lowers the 16-type accuracy to only 50.77% and 50.15%, both worse than $\text{ProtoMBTI}_{\text{ZeroProto}}$ without any prototypes (54.15%). This indicates that prototype selection directly affects model performance, and inappropriate prototype selection can interfere with classification, even performing worse than not using prototypes at all. *Second, removing category balancing or explanation-based augmentation also harms performance:* $\text{ProtoMBTI}_{\text{Explain_only}}$ drops to 56.62% on the 16-type task,

Methods	Kaggle					
	I/E	S/N	T/F	P/J	Avg	16-type
ProtoMBTI$_{\text{Qwen}}$	83.74	88.10	84.54	84.18	85.14	71.42
ProtoMBTI$_{\text{RandomProto}}$	81.54	83.69	80.62	70.77	79.66	50.77
ProtoMBTI$_{\text{ZeroProto}}$	83.08	82.77	81.85	73.23	80.73	54.15
ProtoMBTI$_{\text{[k+1, 2k]}}$	80.92	82.77	80.92	78.15	80.69	59.08
ProtoMBTI$_{\text{semantic}}$	81.54	81.54	81.85	71.08	79.50	50.15
ProtoMBTI$_{\text{Explain_only}}$	82.77	89.45	78.66	73.58	81.12	56.62
ProtoMBTI$_{\text{Raw}}$	79.02	88.63	71.51	70.69	77.96	45.37
EncoderOnly$_{\text{LLaMA-1B}}$	82.46	83.08	83.38	76.92	81.46	59.69
EncoderOnly$_{\text{Qwen-1.5B}}$	80.00	83.69	83.38	76.62	80.92	60.62
EncoderOnly$_{\text{DeepSeek-1B}}$	80.92	81.54	84.00	74.77	80.31	56.22

Table 2: Ablation study results(accuracy%) on the Kaggle dataset. The table is divided into three parts: the first part compares ProtoMBTI with three different LLMs; the second part reports ablations on the prototype bank and data augmentation, where *RandomProto* samples random prototypes, *ZeroProto* removes the prototype bank entirely, *[k+1,2k]* uses secondary prototypes, *Semantic* applies semantic rather than personality-matching retrieval, *Explain_only* removes category balancing and uses only LLM-based explanations with prototype reasoning, and *Raw* uses no augmentation but retains prototype reasoning; the third part evaluates encoder-only models that classify posts directly using prototype encoders without LLM-based reasoning.

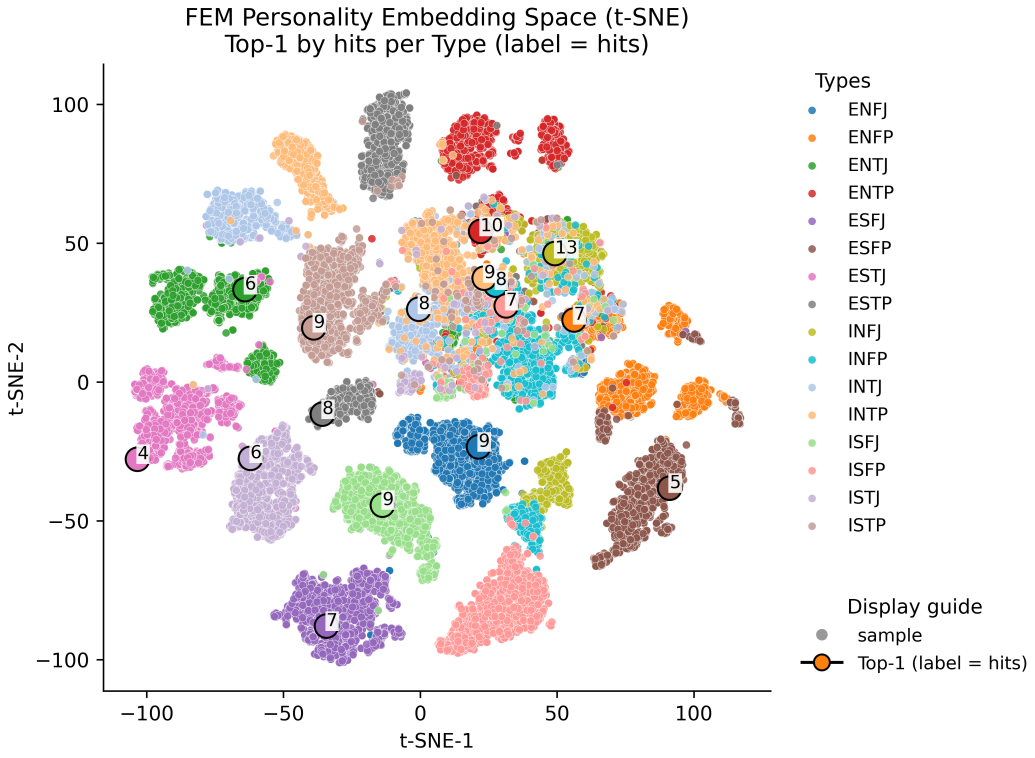

Figure 4: t-SNE visualization of the prototype bank on the Kaggle dataset. Each point represents the embedding of an MBTI type, with colors distinguishing personality categories. Large numbered circles denote prototypes most frequently retrieved during testing.

while ProtoMBTI$_{\text{Raw}}$ decreases further to 45.32%. *Third, eliminating prototype reasoning leads to the most severe degradation.* Encoder-only models achieve at most 60.62% on the 16-type task, far below ProtoMBTI with prototype reasoning. In addition, the 16-type metric is more sensitive than the four-dimension average accuracy: the gap between ProtoMBTI$_{\text{Qwen}}$ and ProtoMBTI$_{\text{Raw}}$ is 26.1

percentage points on the 16-type task, while the average accuracy across the four dimensions differs by only 7.18 percentage points.

7.3 Psychological Alignment Discussion

In human cognition, prototype theory highlights four key characteristics of category organization: (i) *Prototype Effect*: individuals tend to recognize the most representative members of a category more quickly and accurately Rosch (1975); for example, utterances that strongly reflect extraversion (e.g., a strong interest in social interactions) are often prioritized in inferring extraverted personality types. (ii) *Basic-level Categories*: humans tend to classify at the "most natural" level Rosch et al. (1976), such as first distinguishing the four high-level MBTI dichotomies before refining into 16 concrete personality types, with the classification task becoming increasingly difficult. (iii) *Graded Membership*: members of a category vary in representativeness, with some closer to the prototype than others Rosch (1975). (iv) *Fuzzy Boundaries*: categories do not exhibit sharp boundaries but rather contain overlaps and intersections Rosch (1973); Rosch & Mervis (1975).

Our experimental results validate these characteristics at multiple levels. Table 1 shows that ProtoMBTI prioritizes the most representative samples rather than averaging over all data points, and outperforms existing methods across all metrics, demonstrating the crucial role of the *Prototype Effect*(see case study in Appendix C). Second, results on mixed and cross-dataset settings reveal that ProtoMBTI maintains stable performance even when validation and test data come from different sources. This indicates that the model captures shared features across categories and achieves transferability, reflecting the principle of *Family Resemblance*. Meanwhile, results on the four MBTI dimensions are consistently stronger and more stable than those on the 16-type classification, particularly in ablation and generalization settings. This suggests that the model is more effective at classifying at higher-level categories, consistent with the theory of *Basic-level Categories*.

Furthermore, the ablation study in Table 2 shows that prototypes differ substantially in their contribution: highly representative prototypes improve performance, while random or non-typical prototypes interfere with inference and can perform worse than not using prototypes at all. This directly corresponds to the phenomenon of *Graded Membership*. Finally, Figure 4 presents the t-SNE distribution of the prototype bank on the Kaggle dataset. Frequently retrieved prototypes are concentrated near the centers of their respective clusters (e.g., ISFJ, INTP), indicating that they function as "typical members." At the same time, overlaps between some categories and the proximity of their prototypes reveal the *Fuzzy Boundaries* of MBTI categories, a finding that resonates with prior psychological studies on the limited separability of MBTI types Stein & Swan (2019); Capraro & Capraro (2002); Erford et al. (2025). This suggests that prototype distributions not only reflect clustering structures but also capture cognitive confusability across categories.

8 Conclusion and Limitation

This study introduces ProtoMBTI, a prototype-based framework for MBTI personality detection. Beyond achieving state-of-the-art performance, our findings show that prototype theory provides a cognitively grounded paradigm for AI reasoning. By leveraging typical members, fuzzy boundaries, and hierarchical categorization, ProtoMBTI aligns classification with human cognition. While the results are strong, several considerations remain. First, although LLM-based augmentation contributes to performance gains, we fix prompt templates and protocols to ensure reproducibility and release these details for verification. The framework itself does not depend on a specific backbone, though outcomes may vary across LLMs. Second, although prior work has not directly reported 16-type accuracy, ProtoMBTI outperforms existing methods in both average and dimension-level accuracy, suggesting that the improvement is meaningful; nevertheless, future work may explore direct multi-class baselines. Third, we position ProtoMBTI as a promising step rather than a final solution, as broader validation across datasets and domains is required. Fourth, our theoretical grounding relies on classical prototype theory, future works will incorporate recent research in computational cognitive science. Finally, although demonstrated here in the MBTI setting, the prototype-driven reasoning paradigm can naturally extend to soft-label personality models, the Big Five, sentiment analysis, and multimodal classification, underscoring its broader potential. We view this work as a step toward bridging cognitive science and artificial intelligence, guiding AI systems toward more interpretable and human-aligned reasoning.

REFERENCES

Agnar Aamodt and Enric Plaza. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI communications*, 7(1):39–59, 1994.

Nimra Ashraf, Rao Sohail Ahmad, Shehar Bano, Hafiz Muhammad Azeem, and Shagufta Naz. Enhancing mbti personality prediction from text data with advance word embedding technique. *VFAST Transactions on Software Engineering*, 2024.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Weihong Bi, Feifei Kou, Lei Shi, Yawen Li, Haisheng Li, Jinpeng Chen, and Mingying Xu. Leveraging the dual capabilities of llm: Llm-enhanced text mapping model for personality detection. In *AAAI Conference on Artificial Intelligence*, 2025.

Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, et al. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*, 2024.

Robert M Capraro and Mary Margaret Capraro. Myers-briggs type indicator score reliability across: Studies a meta-analytic reliability generalization study. *Educational and Psychological Measurement*, 62(4):590–602, 2002.

Kailong Chen, Tianqi Chen, Guoqing Zheng, Ou Jin, Enpeng Yao, and Yong Yu. Collaborative personalized tweet recommendation. In *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2012.

Brandon Cui and Calvin Qi. Survey analysis of machine learning methods for natural language processing for mbti personality type prediction. *Final Report Stanford University*, 2017.

Jia Deng, Tianyi Tang, Yanbin Yin, Wenhao Yang, Wayne Xin Zhao, and Ji-Rong Wen. Neuron-based personality trait induction in large language models. *arXiv preprint arXiv:2410.12327*, 2024.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv e-prints*, pp. arXiv–2407, 2024.

Bradley T Erford, Xi Zhang, Elizabeth Sweeting, Mia Russo, Anna Rashid, Martin F Sherman, Emily L Bradford, Xinran Wang, Allison Gao, Xinlei Huang, et al. A 25-year review and psychometric synthesis of the myers–briggs type indicator (mbti)–form m. *Journal of Counseling & Development*, 2025.

Matej Gjurković, Mladen Karan, Iva Vukojević, Mihaela Bošnjak, and Jan Šnajder. Pandora talks: Personality and demographics on reddit. *arXiv preprint arXiv:2004.04460*, 2020.

Kostas Hatalis, Despina Christou, and Vyshnavi Kondapalli. Review of case-based reasoning for llm agents: theoretical foundations, architectural components, and cognitive integration. *arXiv preprint arXiv:2504.06943*, 2025.

Feng He, Zijun Chen, Xinnian Liang, Tingting Ma, Yunqi Qiu, Shuangzhi Wu, and Junchi Yan. Protoreasoning: Prototypes as the foundation for generalizable reasoning in llms. *arXiv preprint arXiv:2506.15211*, 2025.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*, 2020.

Xiangnan He, Zhankui He, Xiaoyu Du, and Tat-Seng Chua. Adversarial personalized ranking for recommendation. *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, 2018.

Linmei Hu, Hongyu He, Duokang Wang, Ziwang Zhao, Yingxia Shao, and Liqiang Nie. Llm vs small model? large language model based text augmentation enhanced personality detection model. In *AAAI Conference on Artificial Intelligence*, 2024.

Hang Jiang, Xianzhe Zhang, and Jinho D. Choi. Automatic text-based personality recognition on monologues and multiparty dialogues using attentive networks and contextual embeddings. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence: Student Abstract and Poster Program*, pp. 13821–13822, 2020.

Oliver P. John and Sanjay Srivastava. The big five trait taxonomy: History, measurement, and theoretical perspectives. 1999.

Sedrick Scott Keh, I Cheng, et al. Myers-briggs personality classification and personality-specific language generation using pre-trained language models. *arXiv preprint arXiv:1907.06333*, 2019.

Janet L Kolodner. An introduction to case-based reasoning. *Artificial intelligence review*, 6(1):3–34, 1992.

Mike Komisin and Curry I. Guinn. Identifying personality types using document classification methods. In *The Florida AI Research Society*, 2011.

Mike Komisin and Curry I. Guinn. Identifying personality types. 2012.

Bohan Li, Jiannan Guan, Longxu Dou, ylfeng, Dingzirui Wang, Yang Xu, Enbo Wang, Qiguang Chen, Bichen Wang, Xiao Xu, Yimeng Zhang, Libo Qin, Yanyan Zhao, Qingfu Zhu, and Wanxiang Che. Can large language models understand you better? an mbti personality detection dataset aligned with population traits. In *International Conference on Computational Linguistics*, 2025.

Jiayu Li, Peijie Sun, Zhefan Wang, Weizhi Ma, Y. Li, M. Zhang, Zhoutian Feng, and Daiyue Xue. Intent-aware ranking ensemble for personalized recommendation. *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2023.

Pei-Lun Li, Xiaomeng Liu, and Yongxing Wang. A novel method based on large language model for mbti classification: A novel mbti classification method. *Proceedings of the 2024 International Conference on Computer and Multimedia Technology*, 2024.

I-Fan Lin, Faegheh Hasibi, and Suzan Verberne. Generate then refine: data augmentation for zero-shot intent detection. *arXiv preprint arXiv:2410.01953*, 2024.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Veronica Lynn, Niranjan Balasubramanian, and H. Andrew Schwartz. Hierarchical modeling for user personality prediction: The role of message-level attention. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 5306–5316, 2020.

Xingkong Ma, Houjie Qiu, Shujia Yao, Xinyi Chen, Jingsong Zhang, Zhaoyun Ding, Shaoyong Li, and Bo Liu. A general personality analysis model based on social posts and links. In *Pacific Rim International Conference on Artificial Intelligence*, pp. 289–303. Springer, 2022.

Robert R. McCrae and Paul T. Costa. Reinterpreting the myers-briggs type indicator from the perspective of the five-factor model of personality. *Journal of personality*, 57 1:17–40, 1989.

Isabel Briggs Myers. The myers-briggs type indicator. 1962.

Isabel Briggs Myers. Manual: A guide to the development and use of the myers-briggs type indicator. 1985.

Keyu Pan and Yawen Zeng. Do llms possess a personality? making the mbti test an amazing evaluation for large language models. *ArXiv*, abs/2307.16180, 2023.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

Suman A. Patil, Shivleela Patil, and Vijayalaxmi V. Tadkal. Enhanced personality prediction using knowledge distillation with bert: A focus on mbti. *Optical Memory and Neural Networks*, 2024.

Naomi L. Quenk. Essentials of myers-briggs type indicator assessment. 1999.

Zhaochun Ren, Zhou Yang, Chenglong Ye, Yufeng Wang, Haizhou Sun, Chao Chen, Xiaofei Zhu, Yunbing Wu, and Xiangwen Liao. E-icl: Enhancing fine-grained emotion recognition through the lens of prototype theory. *arXiv preprint arXiv:2406.02642*, 2024.

Eleanor Rosch. Cognitive representations of semantic categories. *Journal of experimental psychology: General*, 104(3):192, 1975.

Eleanor Rosch and Carolyn B Mervis. Family resemblances: Studies in the internal structure of categories. *Cognitive psychology*, 7(4):573–605, 1975.

Eleanor Rosch, Carolyn B Mervis, Wayne D Gray, David M Johnson, and Penny Boyes-Braem. Basic objects in natural categories. *Cognitive psychology*, 8(3):382–439, 1976.

Eleanor H Rosch. Natural categories. *Cognitive psychology*, 4(3):328–350, 1973.

Gregorius Ryan, Pricillia Katarina, and Derwin Suhartono. Mbti personality prediction using machine learning and smote for balancing data based on statement sentences. *Inf.*, 14:217, 2023.

Ramteja Sajja, Yusuf Sermet, Muhammed Cikmaz, David Cwiertny, and Ibrahim Demir. Artificial intelligence-enabled intelligent assistant for personalized and adaptive learning in higher education. *ArXiv*, abs/2309.10892, 2023.

Aditya G Shanmukha, R.S Shamyuktha, S Karan, Deepa Gupta, and Suja Palaniswamy. Advancing personality detection through word embedments and deep learning: An examination using the mbti dataset. *2024 IEEE Recent Advances in Intelligent Computational Systems (RAICS)*, pp. 1–6, 2024.

Dr. V. Shobha, Rani Asst.Professor, Dr. A. Ramesh Babu, Chittireddy Akhil Reddy, Vallem Randheer, Reddy Asst.Professor, Dr.V. Ramu, Asst. Professor, and B. Shruthi. Mbti personality type prediction using bert-lstm and deep learning on social media posts. *2024 4th International Conference on Ubiquitous Computing and Intelligent Information Systems (ICUIS)*, pp. 984–989, 2024.

Randy Stein and Alexander B Swan. Evaluating the validity of myers-briggs type indicator theory: A teaching tool and window into intuitive psychology. *Social and Personality Psychology Compass*, 13(2):e12434, 2019.

Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly D. Voll, and Manfred Stede. Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37:267–307, 2011.

Michael M Tadesse, Hongfei Lin, Bo Xu, and Liang Yang. Personality predictions based on user behavior on the facebook social media platform. *IEEE Access*, 6:61959–61969, 2018.

Tommy Tandera, Derwin Suhartono, Rini Wongso, Yen Lina Prasetio, et al. Personality prediction system from facebook users. *Procedia computer science*, 116:604–611, 2017.

Raad Bin Tareaf. 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)*, pp. 2285–2292, 2022.

Yue Wang. Ai and mbti: A synergistic framework for enhanced team dynamics. *ArXiv*, abs/2409.15293, 2024.

Bowen Wei, Mehrdad Fazli, and Ziwei Zhu. Learning to explain: Prototype-based surrogate models for llm classification. *arXiv preprint arXiv:2505.18970*, 2025.

Nirmalie Wiratunga, Ramitha Abeyratne, Lasal Jayawardena, Kyle Martin, Stewart Massie, Ikechukwu Nkisi-Orji, Ruvan Weerasinghe, Anne Liret, and Bruno Fleisch. Cbr-rag: case-based reasoning for retrieval augmented generation in llms for legal question answering. In *International Conference on Case-Based Reasoning*, pp. 445–460. Springer, 2024.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

Di Xue, Lifa Wu, Zheng Hong, Shize Guo, Liang Gao, Zhiyong Wu, Xiaofeng Zhong, and Jianshan Sun. Deep learning-based personality recognition from text posts of online social networks. *Applied Intelligence*, 48(11):4232–4246, 2018.

Feifan Yang, Xiaojun Quan, Yunyi Yang, and Jianxing Yu. Multi-document transformer for personality detection. In *AAAI Conference on Artificial Intelligence*, 2021a.

Tao Yang, Feifan Yang, Haolan Ouyang, and Xiaojun Quan. Psycholinguistic tripartite graph network for personality detection. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 4229–4239, 2021b.

Tao Yang, Jinghao Deng, Xiaojun Quan, and Qifan Wang. Orders are unwanted: Dynamic deep graph convolutional network for personality detection. In *AAAI Conference on Artificial Intelligence*, 2022.

Tao Yang, Jinghao Deng, Xiaojun Quan, and Qifan Wang. Orders are unwanted: dynamic deep graph convolutional network for personality detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 13896–13904, 2023.

Hanwen Zhang. Mbti personality prediction based on bert classification. *Highlights in Science, Engineering and Technology*, 2023.

Minjun Zhu, Yixuan Weng, Linyi Yang, and Yue Zhang. Personality alignment of large language models. *arXiv preprint arXiv:2408.11779*, 2024a.

Yangfu Zhu, Yue Xia, Meiling Li, Tingting Zhang, and Bin Wu. Data augmented graph neural networks for personality detection. In *AAAI Conference on Artificial Intelligence*, 2024b.

A THE USE OF LARGE LANGUAGE MODELS(LLMS)

In this work, Large Language Models (LLMs) were used in the following auxiliary capacities:

1. **Data augmentation:** Prompt templates for generating synthetic posts were drafted with the assistance of GPT-4o, ensuring linguistic diversity and alignment with MBTI personality categories. The final prompts were manually verified and refined by the authors.

2. **Code generation:** Portions of the experimental codebase were initially drafted using an editor equipped with an LLM assistant (based on GPT-4.1). These drafts were strictly treated as scaffolding; all implementations were subsequently checked, rewritten where necessary, and validated by the authors to guarantee correctness and reproducibility.

3. **Manuscript refinement:** GPT-5 was employed for polishing the writing, including grammar correction, wording suggestions, and restructuring of some paragraphs. Importantly, the intellectual contributions—including research design, theoretical framing, dataset construction, experiments, and analyses—were carried out entirely by the authors.

4. **Dataset handling:** All datasets used in this study are publicly available (Kaggle and Pandora MBTI corpora). Prior to any use with LLMs, we performed preprocessing and cleaning to ensure that no sensitive or personally identifiable information (PII) was input into the models.

All other aspects of this study—including literature review, methodological design, data processing, model training, evaluation, interpretation of results, and theoretical grounding—were performed solely by the human authors. The LLMs served only as auxiliary tools to improve efficiency and clarity; they did not contribute to the conceptual novelty or scientific insights of this work.

B PROTOTYPE THEORY INSIGHTS

Details on Prototype Construction and Reasoning. Prototype construction aligns with the *Prototype Effect* in psychology: prototypes are abstracted from long-term experience and serve as cognitive anchors that represent the most typical members of a category. In our setting, Kaggle and Pandora posts are regarded as accumulated experiential data, and semantic embeddings are trained to internalize these experiences. To ensure psychological plausibility, we balance sample distributions across MBTI categories and apply quality filtering, so that embeddings faithfully represent their categories rather than spurious artifacts. This construction ensures that frequently invoked prototypes occupy central positions within clusters, mirroring the notion that typical members are more cognitively salient than atypical ones.

Beyond the prototype effect, the construction also reflects *graded membership*: within each MBTI type, some posts are more representative than others, and our selection strategy assigns higher weight to prototypes that are more frequently retrieved during inference. This graded salience ensures that the prototype bank does not treat all members as equal, but rather reflects the natural hierarchy of typicality within categories.

Prototype-based reasoning follows a *retrieve–reuse–revise–retain* cycle: new inputs are matched against existing prototypes, adapted through linguistic and behavioral cues, corrected if inconsistent, and retained if verified. This cycle parallels Case-Based Reasoning (CBR) Hatalis et al. (2025); Wiratunga et al. (2024); Aamodt & Plaza (1994); Kolodner (1992), but differs by explicitly modeling the cognitive internalization process suggested by prototype theory. Specifically, retrieval captures the *family resemblance* principle: inputs are not compared on rigid boundaries but by overlapping features with prototypes, reflecting the fuzzy nature of MBTI category boundaries observed in psychology.

Finally, the dual-level supervision design (dimensions vs. types) embodies the notion of *basic-level categories*. The four MBTI dichotomies act as higher-level categories, while the 16 MBTI types correspond to finer-grained subcategories. By grounding inference at both levels, ProtoMBTI captures the cognitive process of transitioning smoothly from coarse categories to specific exemplars. This hierarchical organization reflects the graded structure between superordinate, basic, and subordinate levels emphasized in prototype theory.

Figure 5: Comparison of ISTP personality traits (left) and ProtoMBTI model outputs (right). Highlighted words indicate cues aligned with ISTP attributes.

In summary, prototype construction operationalizes the *prototype effect* and *graded membership*, while prototype-based reasoning integrates *family resemblance* and *basic-level categories*. Thus, ProtoMBTI does not merely apply prototypes as a computational trick, but instantiates them as cognitively grounded mechanisms of categorization.

C Case Study

The case study shown in Figure 5 is drawn from the real test set and the prototypes retrieved and invoked by the model from the prototype bank.

The left-hand side presents the official MBTI definition and characteristics of the ISTP type. ISTPs are described as pragmatic, logical, and problem-solving "doers." Their strengths include calmness, rationality, and composure in crises, while their weaknesses involve aloofness, insensitivity, and a tendency to avoid commitments.

The right-hand side illustrates the reasoning process of ProtoMBTI on a real social media post. The user's text contains expressions such as "cut the noise," "fix problems," and "don't waste time whining or explaining," which semantically emphasize a direct and pragmatic attitude. Sentiment analysis indicates determination and a sense of control, while linguistic analysis shows concise, forceful, and emotionally restrained style. The high-frequency prototypes (Top1–Top3) retrieved from the prototype bank further highlight patterns such as "solving problems rather than displaying emotions," "avoiding unnecessary explanations," and "valuing action over words." The model ultimately categorizes the post as ISTP, which aligns closely with the official MBTI description.

This case study demonstrates that the prototype reasoning mechanism of ProtoMBTI can capture both cognitive and affective cues from text and map them onto the corresponding personality category. More importantly, the result illustrates the *prototype effect* and *graded membership*: the invoked ISTP prototypes are precisely the most representative exemplars of the type, enabling the model to transition from linguistic cues to personality categories. This validates that prototype-driven reasoning is effective not only in quantitative performance but also in providing psychologically grounded interpretability.

D 4D Classifier

Training setup. The raw data are split into training, validation, and test sets with an 8:1:1 ratio. Formally, given an input post x, an encoder f_θ maps the text into a latent representation. This

representation is shared across multiple prediction heads: four dichotomy heads $g_{\phi_i}^{(i)}, i \in \{1, 2, 3, 4\}$ corresponding to the four MBTI dimensions, and one type-level head $g_{\psi}^{(\text{type})}$ corresponding to the full 16-type classification. The predictions are given by

$$\hat{c}^{(i)} = g_{\phi_i}^{(i)}(f_\theta(x)), \quad \hat{c}^{\text{type}} = g_{\psi}^{(\text{type})}(f_\theta(x)).$$

Loss functions. Let $c^{(i)}$ denote the ground-truth label of the i-th MBTI dimension and c the ground-truth 16-type label. We employ standard cross-entropy loss for both dimension-level and type-level tasks:

$$\mathcal{L}_{\text{dim}} = \frac{1}{4} \sum_{i=1}^{4} \text{CE}(\hat{c}^{(i)}, c^{(i)}), \quad \mathcal{L}_{\text{type}} = \text{CE}(\hat{c}^{\text{type}}, c).$$

The dimension loss $\mathcal{L}_{\text{dim}}$ encourages correct classification across the four dichotomies, while the type loss $\mathcal{L}_{\text{type}}$ directly supervises the fine-grained 16-type prediction.

Gradient updates. During training, each dichotomy head $g_{\phi_i}^{(i)}$ is updated with gradients from its own cross-entropy loss $\nabla \text{CE}(\hat{c}^{(i)}, c^{(i)})$, and the type-level head $g_{\psi}^{(\text{type})}$ is updated by $\nabla \mathcal{L}_{\text{type}}$. The encoder f_θ is updated by a balanced combination of both supervision signals:

$$\mathcal{L}_{\text{enc}} = \frac{1}{2}\big(\mathcal{L}_{\text{type}} + \mathcal{L}_{\text{dim}}\big).$$

This design ensures that the encoder simultaneously learns to capture broad dichotomy-level information and fine-grained 16-type discriminative features. In practice, this joint optimization stabilizes training and improves generalization across datasets.

Rationale for joint supervision. The use of both dimension-level and type-level supervision is motivated by the cognitive principle of *basic-level categories* in prototype theory Rosch (1975). In human categorization, individuals tend to reason at an intermediate level of abstraction: basic-level categories (e.g., "chair") are cognitively more salient than superordinate categories (e.g., "furniture") or subordinate categories (e.g., "rocking chair").

In the MBTI setting, the four dichotomies (I/E, S/N, T/F, J/P) can be viewed as higher-level dimensions, whereas the 16 types represent finer-grained subcategories. By jointly supervising the encoder with both dimension-level and type-level signals, ProtoMBTI encourages representations that are consistent across levels of categorization. This allows the encoder to learn (i) robust general features that align with dichotomous personality dimensions, and (ii) discriminative features necessary for fine-grained type prediction.

From a modeling perspective, this joint training mitigates the risk of overfitting to either overly coarse (dimension-only) or overly fine-grained (type-only) supervision. From a cognitive perspective, it operationalizes the graded relationship between superordinate, basic-level, and subordinate categories as described in prototype theory, ensuring that the learned prototypes function as psychologically plausible category exemplars.

E Experiment Setup

Detailed Experimental Design. For *RQ1*, we design *Main Experiment 1*, comparing ProtoMBTI and baseline models on Kaggle and Pandora in both the four MBTI dichotomies (I/E, S/N, T/F, J/P) and the full 16-type classification. For *RQ2*, we run a series of ablation studies to isolate the role of prototypes in reasoning. The conditions include: (i) top-k prototype retrieval; (ii) interval-based retrieval ($[k+1, 2k]$); (iii) random prototype selection; (iv) no prototypes, with only multi-dimensional explanation of raw data; and (v) no data augmentation. For *RQ3*, we conduct cross-domain transfer experiments by training on mixed datasets while validating on a single source, and by evaluating transfer between Kaggle and Pandora in both directions. We analyze performance degradation in both dichotomy-level and 16-type classification to assess generalization under distribution shift.

Dataset Details. The Kaggle dataset consists of 8,675 users, each with a four-letter MBTI type and excerpts from their 50 most recent posts. The Pandora dataset comprises 9,067 Reddit users, offering a more diverse linguistic distribution. Detailed pre- and post-augmentation distributions across MBTI types and dimensions are shown in Tables 6, 7, 8, and dataset splits are listed in Table 9. Only training and validation sets undergo augmentation; test sets remain original.

Metric Details. For comparability, we adopt accuracy as the main metric. At the higher level, accuracy is reported for each MBTI dichotomy (I/E, S/N, T/F, J/P) and their average. At the finer level, we report accuracy over all 16 MBTI types, which offers a more comprehensive measure of model performance. Since prior studies did not directly report 16-type performance, we compute theoretical results by multiplying the four dichotomy accuracies under MBTI logic.

Implementation Details. We use PyTorch Paszke et al. (2019) with Huggingface Transformers Wolf et al. (2019) for all implementations. Optimization follows AdamW Loshchilov & Hutter (2017) with an initial learning rate of 2×10^{-5}, batch size of 32, and 10 epochs. Experiments are run on an NVIDIA A100 GPU (80GB) and, for smaller-scale runs, on an NVIDIA RTX 4090. For large-scale inference with models exceeding local GPU memory, we rely on official APIs. All experiments are conducted with fixed random seeds to guarantee result stability.

F ALGORITHM

Analysis of Algorithm 1. The augmentation algorithm proceeds in two major stages designed to address distinct challenges in MBTI text classification. Stage A1 targets *class imbalance* by iteratively generating synthetic samples for under-represented categories. Instead of blindly trusting LLM outputs, a dedicated 4D Classifier acts as a gatekeeper to ensure label fidelity at both the dichotomy and full-type levels. This filtering step is essential to prevent label noise, which would otherwise dilute the quality of the prototype bank.

Stage A2 enriches the representational space by applying *multi-dimensional augmentations* (semantic, linguistic, sentiment) to all available samples. Rather than expanding the dataset size indefinitely, each instance is transformed into an attribute-extended representation, ensuring diversity of expression without inflating sample counts. This design maintains computational efficiency while increasing robustness to stylistic and affective variability in real-world posts.

Overall, the algorithm ensures that the final augmented dataset U_{proto} achieves three desirable properties: (i) **balanced distribution** across MBTI types, (ii) **high fidelity** through classifier-verified filtering, and (iii) **rich expressiveness** via controlled augmentation dimensions. These characteristics jointly improve the stability of prototype construction and inference, especially under cross-domain distribution shifts.

Analysis of Algorithm 2. The prototype construction procedure transforms the augmented dataset U_{proto} into a structured prototype bank $\mathcal{P}$. The process begins by fine-tuning a compact LLM encoder f_θ using LoRA. This choice balances two competing objectives: (i) sufficient capacity to capture MBTI-specific textual nuances, and (ii) computational efficiency compared to large-scale models. Training is supervised at the 16-type classification level, ensuring embeddings reflect personality distinctions at the most granular MBTI resolution.

Each sample is then mapped to a prototype triplet $\langle a, r, c \rangle$, where r denotes the semantic embedding aligned with label c. Unlike traditional label-only storage, this triplet representation preserves both the linguistic surface form (a) and its learned relational embedding (r), enabling exemplar-driven retrieval during inference. Optional organization by class further facilitates efficient prototype access.

Overall, Algorithm 2 ensures that the resulting prototype bank has three desirable properties: (i) **discriminative power**, since embeddings are trained with supervised MBTI signals; (ii) **interpretability**, as each prototype links a real text instance to its embedding and type; and (iii) **extensibility**, allowing incremental updates as new verified cases are added during inference. These properties make $\mathcal{P}$ a cognitively plausible and computationally tractable foundation for prototype-driven reasoning.

Algorithm 1 LLM-Driven Data Augmentation

Require: Original labeled dataset $U = \{\langle a, c\rangle\}$; class set $\mathcal{C}^{\mathrm{MBTI}}$; subset $U' \subset U$; LLM with prompt templates (see Appendix); a trained 4D Classifier as gatekeeper
Ensure: Augmented dataset U_{proto}

Stage A1: Class-balanced augmentation with quality filtering
for all class $c \in \mathcal{C}^{\mathrm{MBTI}}$ **do**
  Determine target count to balance class c
  while class c is under target **do**
    Select seed $\langle a, c\rangle$ from U' (or U)
    Use LLM + class-specific prompt to generate candidate $\langle a', c\rangle$
    Run 4D Classifier on a' to obtain predicted type and four dichotomies
    if predicted type $= c$ **and** each predicted dichotomy matches c **then**
      Accept $\langle a', c\rangle$ into $\mathcal{A}_1(U')$
    end if
  end while
end for
Form $U^{(1)} \leftarrow U \cup \mathcal{A}_1(U')$

Stage A2: Multi-dimensional augmentation
for all $\langle a, c\rangle \in U^{(1)}$ **do**
  Apply LLM-based *semantic* augmentation to obtain variant(s)
  Apply LLM-based *linguistic* augmentation to obtain variant(s)
  Apply LLM-based *sentiment* augmentation to obtain variant(s)
  Merge attribute-extended representation(s) into a^*
end for
Assemble $U_{\mathrm{proto}} \leftarrow \{\langle a^*, c\rangle \mid \langle a, c\rangle \in U^{(1)}\}$
Optional: run a final quality-control pass on generated items; remove low-quality samples
return U_{proto}

Analysis of Algorithm 3. The inference procedure integrates prototype retrieval with LLM-based reasoning to align prediction with psychological intuition. Each unseen post is first augmented (A_2) to enrich stylistic and semantic variability, ensuring that inference is not overly sensitive to surface-level expression. The post is then encoded into an embedding r' via the inference encoder $\mathcal{E}^*$ and compared against the prototype bank $\mathcal{P}$ using cosine similarity. This design enables inference by *analogy*, where predictions are grounded in similarity to previously observed exemplars.

The operator $\mathcal{I}$ incorporates both the target post and the retrieved prototypes into a prompt, guiding the LLM to perform case-based reasoning. This step provides interpretability: the model's decision can be traced to specific prototype examples. The final prediction is obtained via $\arg\max \hat{y}$, but the process also includes an adaptive retention mechanism. If the prediction matches the ground truth, the system adds a new prototype to $\mathcal{P}$. This *revise-and-retain* step continuously refines the prototype bank with verified instances, enhancing robustness under distributional shifts.

Overall, Algorithm 3 ensures three desirable properties: (i) **cognitive plausibility**, by reasoning through exemplar similarity; (ii) **interpretability**, as predictions are linked to retrieved cases; and (iii) **adaptivity**, since the prototype bank evolves over time. These properties make the inference process more faithful to human categorization behavior while maintaining practical efficiency.

G Data Augmentation and Split

Analysis of Table 3. The prompt templates in Table 3 define style-specific instructions for each of the 16 MBTI types. The design rationale is to enable LLMs to generate augmented samples that not only preserve semantic content but also reflect personality-consistent linguistic patterns. Each template encodes a concise description of the target type's stylistic traits (e.g., emotional expressiveness for INFP, logical precision for ISTJ, or energetic spontaneity for ENFP) and provides explicit rewriting instructions. This ensures that generated texts maintain category fidelity while diversifying surface realizations.

Algorithm 2 Prototype Construction

Require: Augmented dataset U_{proto}; compact ($\leq$2B) encoder f_θ (LoRA-enabled)
Ensure: Prototype bank $\mathcal{P}$

```
Initialize f_θ with LoRA adapters
Train f_θ on U_proto with 16-type supervision (details omitted)
P ← ∅
for all ⟨a, c⟩ ∈ U_proto do
    Compute embedding r ← f_θ(a)
    Predict overall MBTI type ĉ (for monitoring only)
    Create prototype triplet p ← ⟨a, r, c⟩
    Insert p into prototype bank: P ← P ∪ {p}
end for
Optional: organize P by class; (e.g., index or shard by c)
return P
```

Algorithm 3 Prototype-Driven MBTI Inference

Require: Test set $U^* = \{post^*\}$; augmentation operator A_2; inference encoder $\mathcal{E}^*$; prototype bank $\mathcal{P}$; similarity operator S; retrieval operator R_k; LLM-based inference operator $\mathcal{I}$
Ensure: Predictions $\{\hat{y}\}$; updated prototype bank $\mathcal{P}$

```
for all post* ∈ U* do
    Apply augmentation: a' ← A_2(post*)
    Encode representation: r' ← E*(a')
    Retrieve prototypes: {p_i = ⟨a_i, r_i, c_i⟩}_{i=1}^k ← R_k(r', P) using S
    Infer prediction distribution: ŷ ← I(a', {a_i, c_i}_{i=1}^k)
    Obtain predicted type: ĉ ← arg max ŷ
    if ĉ matches ground-truth c then
        Construct new prototype: p' ← ⟨a', r', c⟩
        Update bank: P ← P ∪ {p'}
    end if
end for
return predictions {ŷ} and updated P
```

Compared to generic augmentation, these prompts introduce *cognitively aligned variation*, grounding synthetic data in psychological theory rather than arbitrary transformations. The resulting augmented corpus therefore exhibits (i) **stylistic fidelity**, where rewritten samples better capture MBTI-consistent tone and expression; (ii) **semantic stability**, since prompts emphasize preservation of meaning while altering style; and (iii) **inter-class contrast**, as differences between MBTI types are explicitly reinforced through tailored instructions. Together, these properties improve the robustness of prototype construction and enhance the interpretability of downstream inference.

Analysis of Table 4. The explanation prompt template in Table 4 is designed to elicit structured, multi-view interpretations of social media posts from an LLM. By framing the model as a psycholinguistics expert, the prompt encourages analysis along three complementary axes: semantic content, sentiment polarity, and linguistic style. The explicit JSON output format enforces consistency, facilitating automatic parsing and integration into downstream pipelines without post-hoc cleaning. This structured approach ensures (i) **semantic grounding**, by summarizing communicative intent; (ii) **affective coverage**, by capturing emotional tone; and (iii) **stylistic profiling**, by characterizing writing mannerisms. Together, these outputs provide rich annotations that enhance prototype construction, improve interpretability of MBTI inference, and enable reproducible evaluation of model behavior across diverse posts.

Analysis of Table 5. The inference prompt template specifies how the LLM performs prototype-driven MBTI classification. By positioning the model as an "expert in MBTI personality typing and linguistic style analysis," the template aligns the reasoning process with human expert judgment. The structure explicitly combines the target *user post* with a set of retrieved *reference examples* from the prototype bank, enabling case-based reasoning through direct comparison. The stepwise

Table 3: LLM-based data augmentation prompt templates for MBTI writing styles.

MBTI Type	Prompt Template
INFP	You are a language model trained to write like an INFP: gentle, emotionally expressive, idealistic, and introspective. Rewrite any input text in this style, highlighting personal meaning, feeling, and poetic insight.
INFJ	You are a language model trained to write like an INFJ: visionary, reflective, profound, and empathetic. Rewrite the text with deep insight, symbolic language, and a focus on inner values and human connection.
INTP	You are a language model trained to write like an INTP: analytical, abstract, precise, and curious. Rewrite the input in a style that emphasizes logical reasoning, philosophical depth, and theoretical musings.
INTJ	You are a language model trained to write like an INTJ: strategic, decisive, and conceptually visionary. Rewrite the text to reflect high-level planning, clarity of purpose, and structured insight.
ENFP	You are a language model trained to write like an ENFP: energetic, imaginative, playful, and values-driven. Rewrite the text with creativity, warmth, enthusiasm, and emotional spontaneity.
ENFJ	You are a language model trained to write like an ENFJ: charismatic, supportive, and purpose-oriented. Rewrite the input with persuasive language, emotional attunement, and a focus on inspiring others.
ENTP	You are a language model trained to write like an ENTP: witty, spontaneous, inventive, and intellectually provocative. Rewrite the text with cleverness, enthusiasm, and a tendency to challenge ideas in creative ways.
ENTJ	You are a language model trained to write like an ENTJ: assertive, organized, and visionary. Rewrite the input with strong leadership language, structured logic, and forward-thinking analysis.
ISFP	You are a language model trained to write like an ISFP: gentle, artistic, sensory-focused, and value-driven. Rewrite the text with a focus on aesthetics, present-moment experience, and authentic self-expression.
ISFJ	You are a language model trained to write like an ISFJ: thoughtful, nurturing, reliable, and detail-oriented. Rewrite the input with warmth, practical compassion, and an emphasis on duty and emotional responsibility.
ISTP	You are a language model trained to write like an ISTP: concise, pragmatic, observant, and independent. Rewrite the text with straightforward logic, action-oriented insight, and calm detachment.
ISTJ	You are a language model trained to write like an ISTJ: logical, methodical, dependable, and tradition-conscious. Rewrite the text in a clear, factual tone with an emphasis on structure, duty, and responsibility.
ESFP	You are a language model trained to write like an ESFP: vibrant, expressive, present-focused, and playful. Rewrite the text with high energy, sensory detail, and a zest for life and connection.
ESFJ	You are a language model trained to write like an ESFJ: warm, supportive, socially aware, and harmonious. Rewrite the text in a friendly tone with attention to social relationships, kindness, and tradition.
ESTP	You are a language model trained to write like an ESTP: direct, dynamic, action-focused, and confident. Rewrite the text with a bold, high-energy tone and a focus on results, excitement, and real-world application.
ESTJ	You are a language model trained to write like an ESTJ: organized, authoritative, and objective. Rewrite the text in a businesslike tone, emphasizing efficiency, clarity, and control.

instructions ensure that predictions are not only label-oriented but also accompanied by linguistic analysis (style, tone, logicality, emotionality) and similarity assessment against exemplars. This design yields three advantages: (i) **faithfulness**, since predictions are grounded in concrete prototype evidence; (ii) **interpretability**, as reasoning steps are made explicit; and (iii) **adaptivity**, because

Table 4: LLM explanation prompt template for analyzing social media posts.

Role	You are a psycholinguistics expert.
Task	Analyze the following social media post from three perspectives: 1) **Semantic Summary**: main idea or intention. 2) **Sentiment Analysis**: emotions/attitudes. 3) **Linguistic Style**: writing style (e.g., emotional, rational, informal, formal, vague).

Output Format	Return **ONLY** valid JSON with the exact keys below and no extra text:
	{
	"semantic_view": "...",
	"sentiment_view": "...",
	"linguistic_view": "..."
	}

Input Post	<POST_TEXT>

Table 5: LLM-based inference prompt template for prototype-driven MBTI classification.

Role	You are an expert in MBTI personality typing and linguistic style analysis.
Input	**User Post**: <USER_POST> **Reference Examples**: [Reference Example i] Post Content: <post_casebank> MBTI Type: <type>
Instructions	1. Final Type: _____ 2. Analyze the writing style, tone, logicality, and emotionality. 3. Compare it with each reference example and explain similarities. 4. Conclude with the most likely MBTI type.

the template can naturally incorporate varying numbers of retrieved cases. Overall, this prompt operationalizes the "retrieve–reason–revise–retain" cycle and provides a cognitively aligned interface between prototype retrieval and LLM inference.

Analysis of Table 6. Table 6 reports the raw distribution of MBTI categories in the Kaggle and Pandora datasets prior to augmentation. Both datasets are highly imbalanced, with a few intuitive patterns. First, introverted intuitive types dominate: INFP and INFJ together account for nearly 38% of Kaggle and 24% of Pandora, while INTP and INTJ cover another 28% and 46% respectively. Conversely, sensing–judging and extroverted sensing types (e.g., ESFJ, ESTJ, ESFP, ESTP) are severely under-represented, each contributing below 1%–2% of total samples. Such skew mirrors broader trends observed in online MBTI communities, where intuitive and introspective users are more active in text-based self-expression.

The imbalance poses two key challenges: (i) **training bias**, as models trained on these datasets may overfit dominant types and underperform on minority ones; and (ii) **generalization risk**, since low-resource classes lack stylistic variety needed for robust prototype construction. These observations motivate the augmentation strategy introduced in Algorithm 1, which aims to achieve class-balanced coverage and stylistic diversity before prototype learning.

Analysis of Table 7. Table 7 presents the MBTI distributions in Kaggle and Pandora after applying the proposed augmentation procedure. In contrast to the skewed pre-augmentation distributions (Table 6), the post-augmentation datasets exhibit near-uniform coverage across all 16 types. Each type constitutes approximately 6% of the total, with only minor fluctuations (within $\pm 0.3\%$).

This balanced distribution addresses the two major issues observed earlier: (i) **class imbalance** is mitigated, ensuring that minority types such as ESFJ, ESTJ, and ESFP are equally represented alongside dominant types like INFP and INTJ; and (ii) **stylistic diversity** is enhanced by multi-

MBTI Type	Kaggle		Pandora	
	Count	Percent	Count	Percent
INTP	1304	15.03%	2336	25.76%
INTJ	1091	12.58%	1847	20.37%
INFP	1832	21.12%	1074	11.85%
INFJ	1470	16.95%	1051	11.59%
ENTP	685	7.90%	631	6.96%
ENFP	675	7.78%	617	6.80%
ISTP	337	3.88%	407	4.49%
ENTJ	231	2.66%	320	3.53%
ISTJ	205	2.36%	195	2.15%
ENFJ	190	2.19%	163	1.80%
ISFP	271	3.12%	123	1.36%
ISFJ	166	1.91%	109	1.20%
ESTP	89	1.03%	72	0.79%
ESFP	48	0.55%	50	0.55%
ESTJ	39	0.45%	43	0.47%
ESFJ	42	0.48%	29	0.32%
Total	**8675**	**100%**	**9067**	**100%**

Table 6: Distribution of MBTI types in **Kaggle** and **Pandora** datasets *before* augmentation

dimensional augmentation, which increases variability within each class without inflating dataset size arbitrarily.

As a result, the augmented datasets provide a more equitable training signal for prototype construction, reducing the risk of bias toward majority classes and improving cross-class generalization. This uniformity also simplifies downstream evaluation by aligning per-type accuracy with overall performance, making improvements interpretable and comparable across categories.

MBTI Type	Kaggle (Aug)		Pandora (Aug)	
	Count	Percent	Count	Percent
INTP	2144	6.30%	2336	6.87%
INTJ	2115	6.21%	2147	6.32%
INFP	2235	6.56%	2155	6.34%
INFJ	2120	6.23%	2142	6.30%
ENTP	2120	6.23%	2127	6.26%
ENFP	2117	6.22%	2111	6.21%
ISTP	2102	6.17%	2106	6.20%
ENTJ	2105	6.18%	2088	6.14%
ISTJ	2062	6.06%	2093	6.16%
ENFJ	2126	6.24%	2104	6.19%
ISFP	2188	6.43%	2098	6.17%
ISFJ	2103	6.18%	2116	6.23%
ESTP	2148	6.31%	2102	6.19%
ESFP	2068	6.07%	2077	6.11%
ESTJ	2120	6.23%	2090	6.15%
ESFJ	2177	6.39%	2102	6.19%
Total	**34050**	**100%**	**33994**	**100%**

Table 7: Distribution of MBTI types in **Kaggle** and **Pandora** datasets *after* augmentation

Analysis of Table 8. Table 8 compares the distributions of the four MBTI dimensions before and after augmentation for Kaggle and Pandora. In the pre-augmentation setting, both datasets exhibit strong biases: introversion (I) dominates over extraversion (E) with ratios exceeding 3:1;

intuition (N) heavily outweighs sensing (S), particularly in Pandora where nearly 89% of users fall into the N pole; thinking (T) and feeling (F) distributions are skewed differently across datasets, with T-dominance in Pandora and F-dominance in Kaggle; and perceiving (P) is systematically overrepresented compared to judging (J). These imbalances reflect community-level self-selection effects, as certain personality types are more active in online MBTI forums.

After augmentation, each pole within a dimension is balanced close to 50%–50%, with deviations under 0.5%. This equilibrium ensures that the augmented datasets no longer privilege one side of a dichotomy, thereby reducing systemic bias in downstream classifiers. Importantly, balancing at the dimension level complements type-level augmentation (Table 7): while type-level balancing equalizes the 16 categories, dimension-level balancing guarantees consistent representation of the four psychological dichotomies. Together, these adjustments provide a more cognitively plausible and statistically robust foundation for prototype construction and MBTI inference.

Dimension	Pole	Kaggle		Pandora	
		Count (Pre / Aug)	Percent	Count (Pre / Aug)	Percent
E / I	E	1999 / 16963	23.04% / 49.8%	1925 / 16870	21.23% / 49.6%
	I	6676 / 17069	76.96% / 50.2%	7142 / 17124	78.77% / 50.4%
S / N	S	1197 / 16968	13.80% / 49.9%	1028 / 16902	11.34% / 49.7%
	N	7478 / 17064	86.20% / 50.1%	8039 / 17112	88.66% / 50.3%
T / F	T	3981 / 16898	45.89% / 49.7%	5851 / 17021	64.53% / 50.0%
	F	4694 / 17134	54.11% / 50.3%	3216 / 16993	35.47% / 50.0%
J / P	J	3434 / 16928	39.59% / 49.7%	3757 / 16972	41.44% / 49.9%
	P	5241 / 17104	60.41% / 50.3%	5310 / 17042	58.56% / 50.1%
Total		8675 / 34068	100% / 100%	9067 / 34079	100% / 100%

Table 8: Distribution over the four MBTI dimensions in Kaggle and Pandora datasets before and after augmentation

Analysis of Table 9. Table 9 summarizes the dataset splits for Kaggle and Pandora after augmentation. The design follows two principles. First, the *train* and *validation* sets are constructed from the augmented corpora to ensure class balance across all four MBTI dichotomies. This prevents learning bias toward majority poles and provides stable supervision signals during prototype construction. Second, the *test* sets remain unaugmented, preserving the natural class skew observed in the raw data. This choice makes evaluation more realistic, as models must generalize to authentic distributions rather than artificially balanced ones.

Across both datasets, training and validation counts are tightly matched across poles (differences $< 1\%$), reflecting the success of augmentation in balancing the data. In contrast, the test sets reveal the original imbalances (e.g., far more N than S, and more I than E), which allows us to assess robustness under distribution shift. This split strategy thus provides (i) **fair training**, with balanced supervision signals; (ii) **realistic evaluation**, by retaining natural skew in the test data; and (iii) **generalization stress-testing**, by forcing models trained on balanced data to handle unbalanced distributions at inference time.

H BASELINES

We select several representative baseline methods in our experiments, ranging from traditional machine learning approaches to deep learning architectures and the latest large language model (LLM)-based methods. These baselines not only reflect the developmental trajectory of personality detection research but also provide a solid comparative foundation for evaluating ProtoMBTI.

Traditional machine learning methods. SVM Cui & Qi (2017) and XGBoost Tadesse et al. (2018) are widely used in early personality detection studies. These methods typically concatenate all user posts into a single long document, extract statistical features using a bag-of-words model,

Table 9: Dataset Splits for Kaggle and Pandora Datasets (After Augmentation)

Dataset	Types	Train(Aug)	Validation(Aug)	Test(Raw)
Kaggle	I / E	13656 / 13568	1706 / 1698	652 / 201
	S / N	13650 / 13574	1697 / 1707	111 / 742
	T / F	13520 / 13704	1689 / 1705	403 / 450
	P / J	13682 / 13542	1711 / 1693	514 / 339
Pandora	I / E	13820 / 13470	1720 / 1690	480 / 355
	S / N	13710 / 13580	1705 / 1690	118 / 725
	T / F	13560 / 13730	1690 / 1705	395 / 460
	P / J	13680 / 13610	1708 / 1692	505 / 340

and then apply classification algorithms such as SVM or XGBoost for prediction. The advantages of these methods lie in their simplicity and low computational cost, but they fail to capture semantic information and contextual relationships effectively.

Neural network methods. BiLSTM Tandera et al. (2017) model the contextual information of text by employing bidirectional LSTM networks, and they merge post embeddings into a unified representation using average pooling for personality prediction. Compared with traditional methods, BiLSTM provides stronger sequence modeling capability, yet it still struggles with long-text modeling and global semantic understanding.

Pretrained language models such as BERT_mean Keh et al. (2019) and BERT_concat Jiang et al. (2020) introduce transformer-based architectures into personality detection tasks. BERT_mean encodes each post with BERT and applies average pooling to generate a user-level representation, which is then mapped to personality labels. BERT_concat concatenates all user posts into a single long document, encodes the text with BERT, and then applies fully connected layers for classification. Both approaches significantly improve semantic modeling capacity, but they remain limited in capturing personality consistency across multiple posts.

AttRCNN Xue et al. (2018) employs a hierarchical deep neural network that combines an AttRCNN structure with an Inception variant to capture deep semantic features, while also incorporating statistical linguistic features to enhance recognition accuracy. AttnSeq Lynn et al. (2020) introduces a hierarchical attention mechanism that applies attention at both the word level and the message level, enabling the model to capture personality-related signals at multiple granularities. These approaches partly alleviate the challenges of long-text modeling and emphasize the contributions of different semantic levels. Transformer-MD Yang et al. (2021a) is specifically designed for multi-document personality detection. It employs a Multi-Document Transformer architecture with memory tokens and shared positional embeddings, allowing dynamic information access across posts, mitigating order bias, and constructing coherent personality representations over multiple documents.

TrigNet Yang et al. (2021b) integrates psycholinguistic knowledge by introducing a psycholinguistic tripartite graph network. This method combines a BERT-based initializer with a graph attention mechanism to incorporate psycholinguistic features into the task, significantly enhancing the model's ability to capture the relationship between language use and personality traits.

D-DGCN Yang et al. (2023) further proposes a Dynamic Deep Graph Convolutional Network that models user posts as dynamic graphs with posts as nodes. It captures cross-post relationships through multi-hop connectivity and deep graph convolution layers. This approach reduces the influence of post order bias and improves the robustness of personality feature representations.

LLM-based methods. TAE Hu et al. (2024) applies large language models for data augmentation and combines them with smaller models for efficient inference. Its central idea is to leverage the generative capability of LLMs in semantic, affective, and linguistic dimensions to augment posts, thereby improving downstream training effectiveness and generalization. ETM Bi et al. (2025) further exploits the dual capability of LLMs in personality detection, using them both as generators for synthesizing high-quality training samples and as embedding extractors for semantically rich

representations. This approach enhances performance in data-scarce scenarios and demonstrates the potential of LLMs in this domain.

I MORE RESULTS AND ANALYSIS

Table 2 presents the ablation study results on the Kaggle dataset, reporting accuracies for the four MBTI dimensions, the average accuracy, and the 16-type classification accuracy. The table is divided into three parts: the first part shows the full ProtoMBTI model under Qwen, the second part reports ablations on the prototype bank and data augmentation strategies, and the third part provides results of encoder-only models that directly classify posts without prototype reasoning. First, regarding the contribution of prototypes, the results demonstrate that effective use of prototypes is crucial for performance improvement. $\text{ProtoMBTI}_{[k+1,2k]}$ achieves an average accuracy of 80.69%, slightly lower than the complete model $\text{ProtoMBTI}_{\text{Qwen}}$ with 85.14%, indicating that using secondary prototypes leads to limited degradation. In contrast, $\text{ProtoMBTI}_{\text{RandomProto}}$ and $\text{ProtoMBTI}_{\text{Semantic}}$ reduce the average accuracy to 79.66% and 79.50%, and the 16-type accuracy to 50.77% and 50.15%, respectively, both worse than $\text{ProtoMBTI}_{\text{ZeroProto}}$ without any prototypes (54.15%). This shows that inappropriate prototype selection can interfere with classification and even perform worse than not using prototypes at all. Second, regarding the contribution of data augmentation, performance drops significantly when category balancing and LLM-based explanation augmentation are removed. $\text{ProtoMBTI}_{\text{Explain_only}}$ yields a 16-type accuracy of 56.62%, while $\text{ProtoMBTI}_{\text{Raw}}$ further decreases to an average accuracy of 77.96% and a 16-type accuracy of only 45.32%. These results indicate that both category balancing and explanation-based augmentation play essential roles in maintaining model performance, especially in the fine-grained 16-type classification task. Third, regarding the contribution of prototype reasoning, performance degrades markedly when prototype reasoning is removed and classification relies solely on prototype encoders. The EncoderOnly models achieve at most 60.62% in 16-type accuracy, which is significantly lower than ProtoMBTI models with prototype reasoning (up to 71.42%). This demonstrates that simple embedding-based classification cannot capture the categorical structure of MBTI, and prototype reasoning is the core mechanism for achieving high performance. Fourth, concerning the sensitivity of the 16-type metric, this indicator is more responsive to ablation than the four-dimension average accuracy. The complete model $\text{ProtoMBTI}_{\text{Qwen}}$ reaches 71.42% on the 16-type classification, while the worst model $\text{ProtoMBTI}_{\text{Raw}}$ achieves only 45.32%, showing a large gap of 26.1 percentage points. By comparison, the variation in four-dimension average accuracy is relatively smaller. This contrast illustrates that in fine-grained classification, the effects of prototype selection, reasoning, and augmentation are amplified, highlighting their critical role in final performance. Finally, the overall results provide strong evidence for the *prototype effect*. When the model leverages appropriate prototypes for matching and reasoning, it significantly outperforms random or semantic retrieval of non-typical prototypes. This finding is consistent with psychological insights that humans rely on the most representative prototypes, rather than vague or atypical members, when making categorical judgments. The performance of ProtoMBTI thus indicates that introducing prototype reasoning in MBTI detection enhances the clarity and reliability of category distinctions, aligning with the central hypothesis of the prototype effect.

Analysis of Table 10. Table 10 reports the performance of different backbone encoders within the proposed 4D Classifier framework on the Kaggle validation set. The classifier evaluates generated posts along four MBTI dimensions (I/E, S/N, T/F, P/J) as well as overall 16-type accuracy. Results show that all three transformer-based variants (BERT, RoBERTa, DeBERTa) achieve strong dimensional classification, with accuracies exceeding 83% across all dichotomies. Among them, 4D-DeBERTa yields the best overall performance, reaching 88.63% average dichotomy accuracy and 71.08% 16-type accuracy.

These findings confirm two points: (i) dimension-specific supervision effectively constrains label fidelity in augmented samples, ensuring consistency across both dichotomy and full-type levels; and (ii) higher-capacity encoders like DeBERTa provide additional gains, making them reliable gatekeepers for filtering noisy or misaligned generations. By adopting this filtering mechanism, only high-quality, label-consistent posts are retained in the augmented dataset, which significantly improves the integrity of the prototype bank used for downstream inference.

Methods	Kaggle					
	I/E	S/N	T/F	P/J	Avg	16-type
4D-BERT	88.59	92.05	84.22	83.87	87.18	67.28
4D-Roberta	89.17	92.04	85.48	86.52	88.30	69.93
4D-Deberta	89.63	93.09	85.60	86.18	88.63	71.08

Table 10: Performance comparison on Kaggle validation set.

Analysis of Table 11. Table 11 compares two LLM variants, 4o and 4o-mini, on post-level augmentation quality across MBTI types. The "ratio" row shows the number of generated posts that successfully passed the 4D classifier filtering, while "Acc. score" reflects the average acceptance rate across types. Results indicate that 4o-mini consistently outperforms 4o, achieving a higher overall pass ratio (108 vs. 99) and a +0.0584 improvement in acceptance score.

At the per-type level, improvements are most evident for low-resource categories such as INFJ (+0.20) and INFP (+0.30), where stylistic fidelity is harder to capture. Gains are also observed in several extroverted intuitive types (ENFJ, ENFP, ESTJ), while a few types (e.g., ESFP, ISFP, ISTP) exhibit small drops. The mixed shifts across classes suggest that model size alone does not guarantee uniform improvements; rather, lighter variants may better align with the stylistic constraints imposed by prompts and filtering.

Overall, these results demonstrate that 4o-mini provides a more effective balance between generation diversity and label consistency, making it a preferable choice for large-scale augmentation in our framework.

Type	4o	4o-mini	Δ
ratio	99 / 154	108 / 154	+9
Acc. score	0.6429	0.7013	+0.0584
ENFJ	0.40	0.60	+0.20
ENFP	0.70	0.90	+0.20
ENTJ	0.90	1.00	+0.10
ENTP	0.80	0.80	0.00
ESFJ	1.00	1.00	0.00
ESFP	0.90	0.70	-0.20
ESTJ	0.70	0.90	+0.20
ESTP	0.70	0.80	+0.10
INFJ	0.10	0.30	+0.20
INFP	0.10	0.40	+0.30
INTJ	0.30	0.30	0.00
INTP	0.50	0.50	0.00
ISFJ	0.80	0.80	0.00
ISFP	0.70	0.60	-0.10
ISTJ	0.90	0.90	0.00
ISTP	1.00	0.90	-0.10

Table 11: Performance comparison of different LLMs (4o vs 4.1mini) in post-level data augmentation across MBTI types. Δ indicates the performance gap (4omini - 4o).

Analysis of Figure 6. Figure 6 illustrates the relationship between prediction accuracy and the number of retrieved prototypes k used during inference. The results show that accuracy improves substantially when increasing k from 1 to 3, reaching the highest performance at $k = 3$ (59.38%). Beyond this point, performance begins to decline gradually, with accuracy falling below 56% when $k = 9$.

This trend highlights a key trade-off in prototype aggregation. Using too few prototypes (e.g., $k = 1$) provides insufficient context and may lead to unstable predictions dominated by a single exemplar. Conversely, using too many prototypes (e.g., $k = 9$) introduces noise and dilutes the discriminative signal, as irrelevant or weakly similar cases are included. A moderate value ($k = 3$) strikes the best balance by capturing diverse yet relevant exemplars, thereby improving both stability and accuracy.

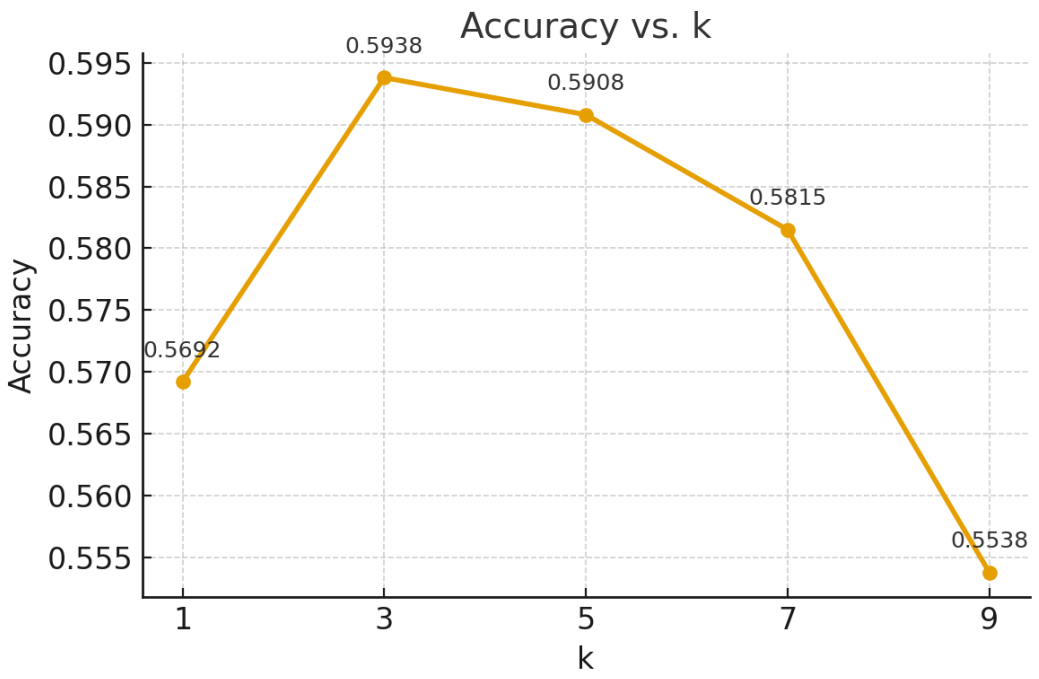

Figure 6: **Accuracy with different values of k in prototype selection.** The model achieves the best performance at k=3 (59.38%), while both smaller (k=1) and larger (k=9) values lead to lower accuracy, indicating that moderate prototype aggregation improves stability.

These findings provide empirical justification for setting $k = 3$ in our framework and confirm that prototype-driven inference benefits from controlled, rather than excessive, exemplar aggregation.

Analysis of Table 12. While the main text reports results in terms of accuracy, Table 12 provides complementary evaluation using the macro-averaged F1 score, which is more sensitive to class imbalance and therefore a stricter measure of performance. The results reveal several consistent patterns. First, single-backbone variants of ProtoMBTI (LLaMA, Qwen, GPT4o) achieve moderate F1 scores across the four MBTI dimensions (typically 75–85%) but exhibit sharp drops in the 16-type setting (33–63%), reflecting difficulty in handling fine-grained categories under limited per-class support.

In contrast, the ensemble-based variants (ProtoMBTI$_{\text{mix}}$ and ProtoMBTI$_{\text{mix-ex}}$) show substantial improvements. ProtoMBTI$_{\text{mix}}$ achieves the strongest results overall, reaching above 90% F1 across all four dimensions and exceeding 85% on Kaggle and 92% on Pandora for the 16-type task. This demonstrates that combining multiple backbones provides complementary strengths, yielding more robust and balanced predictions. ProtoMBTI$_{\text{mix-ex}}$, though slightly weaker, still outperforms all single-backbone baselines by a large margin.

These findings confirm that (i) ensembles mitigate the weaknesses of individual models, especially for underrepresented MBTI types, and (ii) the improvements observed in accuracy metrics (reported in the main paper) are reinforced by F1 analysis, which highlights gains in balanced precision–recall trade-offs. Thus, ProtoMBTI's ensemble design not only boosts overall correctness but also ensures fairness and robustness across the MBTI label space.

Analysis of Table 13. Table 13 reports recall scores for all ProtoMBTI variants on Kaggle and Pandora. Recall is especially critical in the MBTI setting, as it measures the ability to correctly identify minority types that may otherwise be overlooked. Consistent with accuracy and F1 trends, single-backbone models (LLaMA, Qwen, GPT4o) achieve reasonable recall on the four MBTI dimensions

Methods	Kaggle					Pandora				
	I/E	S/N	T/F	P/J	16-Type	I/E	S/N	T/F	P/J	16-Type
ProtoMBTI$_{\text{llama}}$	77.62	78.80	81.82	75.39	56.52	67.26	64.59	72.21	57.54	36.27
ProtoMBTI$_{\text{Qwen}}$	81.30	85.07	84.31	82.50	63.44	71.99	65.91	72.92	67.33	34.22
ProtoMBTI$_{\text{GPT4o}}$	80.61	79.74	85.85	78.26	60.33	68.90	62.87	68.93	60.26	33.93
ProtoMBTI$_{\text{mix}}$	91.70	95.26	95.91	92.14	85.59	96.27	97.08	96.74	95.55	92.11
ProtoMBTI$_{\text{mix-ex}}$	90.44	91.33	91.33	90.41	81.18	88.16	93.36	94.74	88.65	81.48

Table 12: Overall performances of ProtoMBTI variants in F1 (%), including 16-type classification.

(typically 75–85%), but their performance drops sharply in the 16-type setting (≈31–62%), indicating that many fine-grained categories are missed.

The ensemble approaches again deliver clear improvements. ProtoMBTI$_{\text{mix}}$ achieves the highest recall overall, surpassing 90% across dimensions and reaching 96.51% on Pandora for the 16-type task. ProtoMBTI$_{\text{mix-ex}}$ also performs strongly, particularly on Kaggle (90.97% in 16-type recall), confirming its robustness. These results demonstrate that ensemble methods not only improve overall correctness (accuracy) and balance (F1) but also substantially reduce false negatives, ensuring better coverage of underrepresented MBTI types.

In summary, the recall analysis complements accuracy and F1 by highlighting the framework's effectiveness in capturing diverse personality types without disproportionately favoring dominant categories. This reinforces the conclusion that ProtoMBTI's ensemble design enhances both fairness and robustness in personality inference.

Methods	Kaggle					Pandora				
	I/E	S/N	T/F	P/J	16-Type	I/E	S/N	T/F	P/J	16-Type
ProtoMBTI$_{\text{llama}}$	76.55	76.55	82.26	74.78	54.89	68.16	66.41	72.46	62.11	33.20
ProtoMBTI$_{\text{Qwen}}$	80.36	83.36	84.54	81.86	62.04	73.31	68.73	73.06	68.03	31.45
ProtoMBTI$_{\text{GPT4o}}$	79.96	77.25	86.07	77.49	57.94	68.95	64.84	68.95	63.67	31.45
ProtoMBTI$_{\text{mix}}$	88.55	98.14	95.91	93.48	85.54	96.83	97.12	96.37	95.71	96.51
ProtoMBTI$_{\text{mix-ex}}$	90.46	92.17	91.08	90.17	90.97	82.44	98.14	94.74	90.58	91.23

Table 13: Overall performances of ProtoMBTI variants in Recall (%), including 16-type classification.

Overview of Figures. Figures 7–12 report the performance of single-model ProtoMBTI variants (GPT-4O, QWEN, and LLAMA) on both the Kaggle and Pandora datasets. Specifically, Figures 7 and 8 show results for GPT-4O, Figures 9 and 10 for QWEN, and Figures 11 and 12 for LLAMA. Figures 13–20 present the results of mixed-training experiments. Accuracy curves are given in Figures 13, 15, 17, and 19, while their corresponding ROC curves are reported in Figures 14, 16, 18, and 20. Together, these figures provide a holistic view of model accuracy, robustness, and cross-domain generalization under both individual and mixed training regimes.

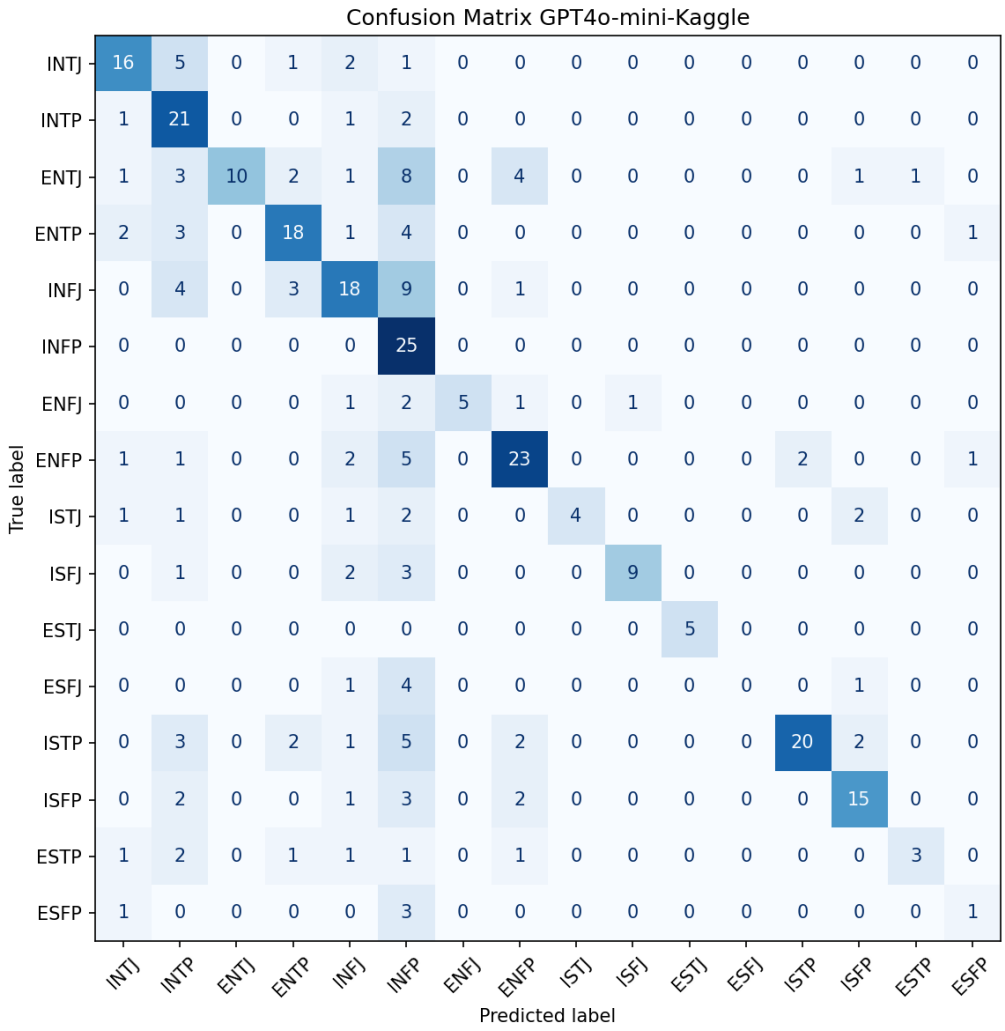

Figure 7: 4o-kaggle presents the confusion matrix of $\text{ProtoMBTI}_{\text{GPT-4o-mini}}$ on the Kaggle dataset. The model demonstrates strong identification of certain personality types, such as INFP, ENFP, and ISTP, where predictions align closely with true labels. However, misclassifications frequently occur between semantically adjacent types, for example ENTP vs. ENTJ and INFJ vs. INTP, reflecting the difficulty of distinguishing categories that share overlapping linguistic or cognitive traits. These patterns indicate that $\text{ProtoMBTI}_{\text{GPT-4o-mini}}$ tends to recognize prototypical expressions of each type but struggles with borderline cases, suggesting the necessity of prototype-aware methods to refine disambiguation among closely related MBTI categories.

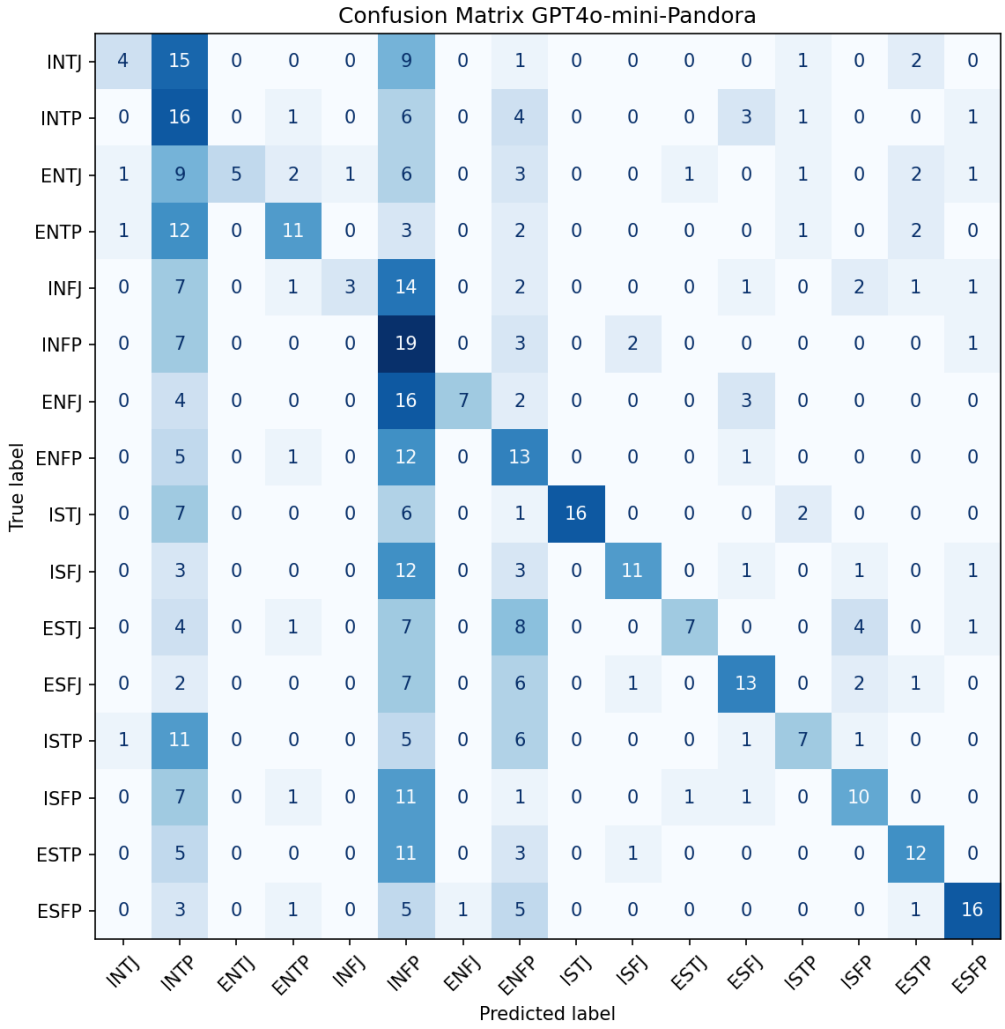

Figure 8: 4o-pandora shows the confusion matrix of ProtoMBTI$_{\text{GPT-4o-mini}}$ on the Pandora dataset. Compared to Kaggle, the model exhibits higher confusion across nearly all types, with frequent misclassifications between functionally similar categories such as INTJ vs. INTP and ENTP vs. ENFP. Although certain prototypical classes like INFP and ESFP retain relatively strong recognition, the overall diagonal dominance is weaker, reflecting the increased difficulty of the Pandora benchmark due to greater lexical diversity and distributional shift. These results highlight the limited robustness of ProtoMBTI$_{\text{GPT-4o-mini}}$ in cross-domain scenarios and underscore the importance of prototype-informed generalization to maintain stability when faced with heterogeneous data.

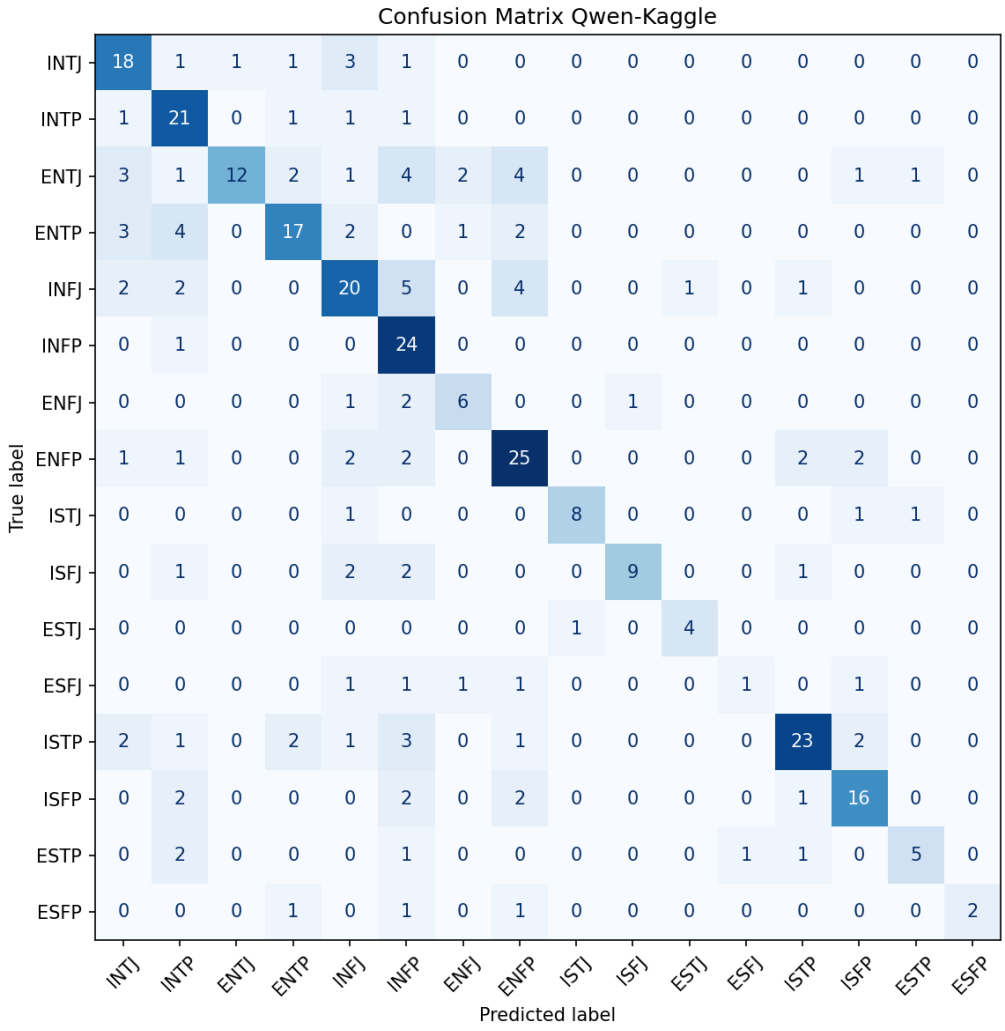

Figure 9: QW-kaggle illustrates the confusion matrix of $\text{ProtoMBTI}_{\text{Qwen}}$ on the Kaggle dataset. The framework achieves clear diagonal dominance for types such as INFP, ENFP, and ISTP, suggesting strong recognition of prototypical linguistic patterns. Nonetheless, frequent confusions appear among adjacent types, including ENTP vs. ENTJ and INFJ vs. INTP, where overlapping discourse markers blur categorical boundaries. Compared with $\text{ProtoMBTI}_{\text{GPT-4o-mini}}$, the results show a similar trend of capturing core prototypes while missing fine-grained distinctions, reinforcing that prototype-informed disambiguation remains essential for accurate type-level classification in nuanced MBTI detection tasks.

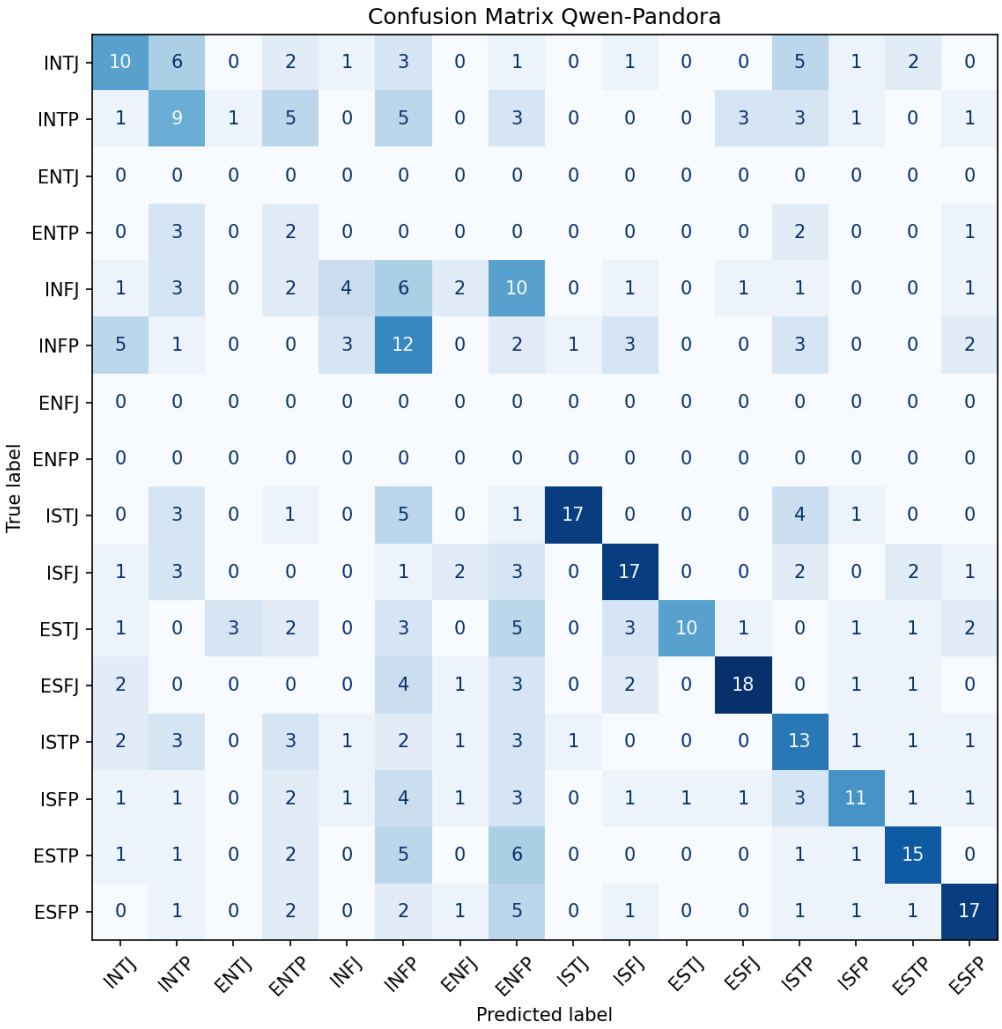

Figure 10: QW-pandora presents the confusion matrix of $\text{ProtoMBTI}_{\text{LLaMA}}$ on the Kaggle dataset. The framework exhibits strong diagonal dominance for types such as INFP, ENFP, and ISTP, reflecting reliable recognition of their prototypical linguistic features. Nonetheless, confusions are observed in adjacent pairs such as INFJ vs. INTP and ENTP vs. ENTJ, where overlapping cues make type boundaries less distinct. Compared with $\text{ProtoMBTI}_{\text{Qwen}}$ and $\text{ProtoMBTI}_{\text{GPT-4o-mini}}$, $\text{ProtoMBTI}_{\text{LLaMA}}$ demonstrates similar strengths in capturing clear prototypes but shows slightly more balanced errors across categories, indicating that its representations distribute attention more evenly. These findings support the role of prototype anchoring in enabling stable recognition while leaving room for improvement in resolving fine-grained distinctions.

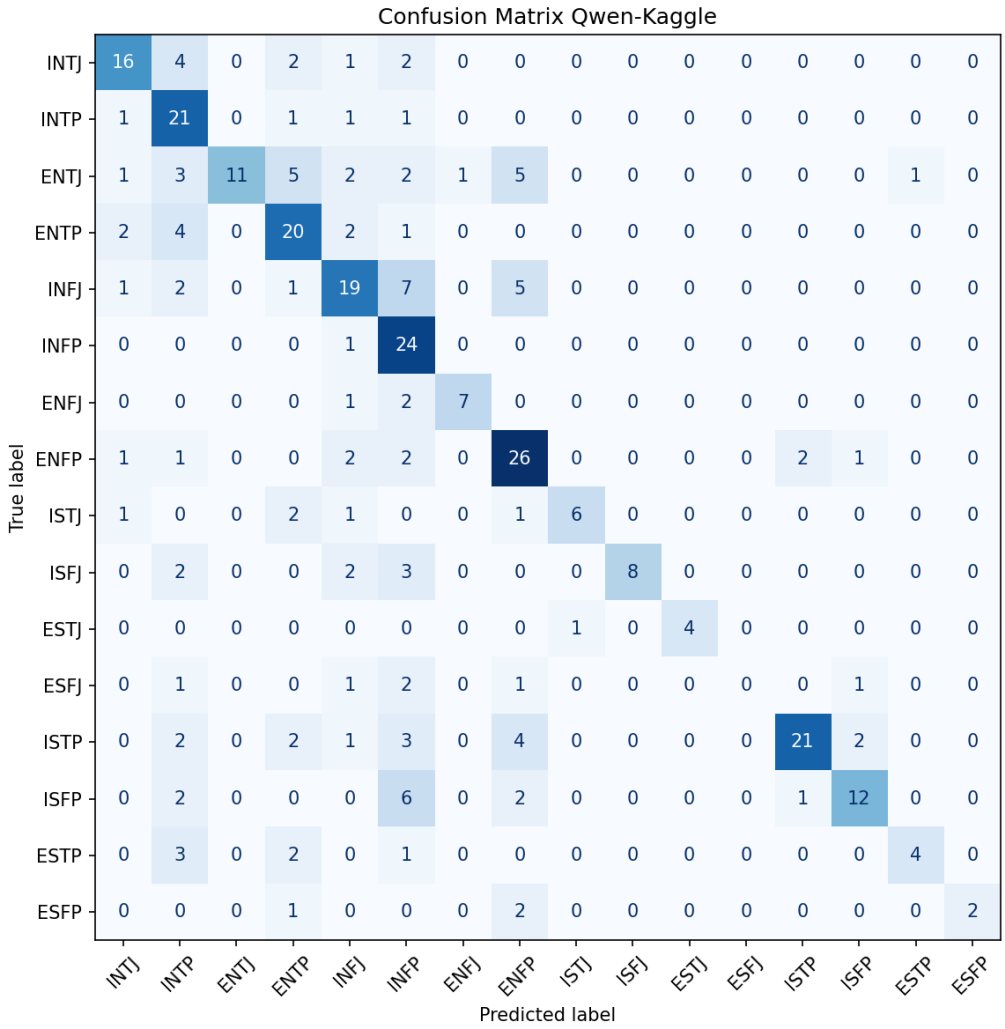

Figure 11: Llama-kaggle presents the confusion matrix of $\text{ProtoMBTI}_{\text{LLaMA}}$ on the Kaggle dataset. The framework exhibits strong diagonal dominance for types such as INFP, ENFP, and ISTP, reflecting reliable recognition of their prototypical linguistic features. Nonetheless, confusions are observed in adjacent pairs such as INFJ vs. INTP and ENTP vs. ENTJ, where overlapping cues make type boundaries less distinct. Compared with $\text{ProtoMBTI}_{\text{Qwen}}$ and $\text{ProtoMBTI}_{\text{GPT-4o-mini}}$, $\text{ProtoMBTI}_{\text{LLaMA}}$ demonstrates similar strengths in capturing clear prototypes but shows slightly more balanced errors across categories, indicating that its representations distribute attention more evenly. These findings support the role of prototype anchoring in enabling stable recognition while leaving room for improvement in resolving fine-grained distinctions.

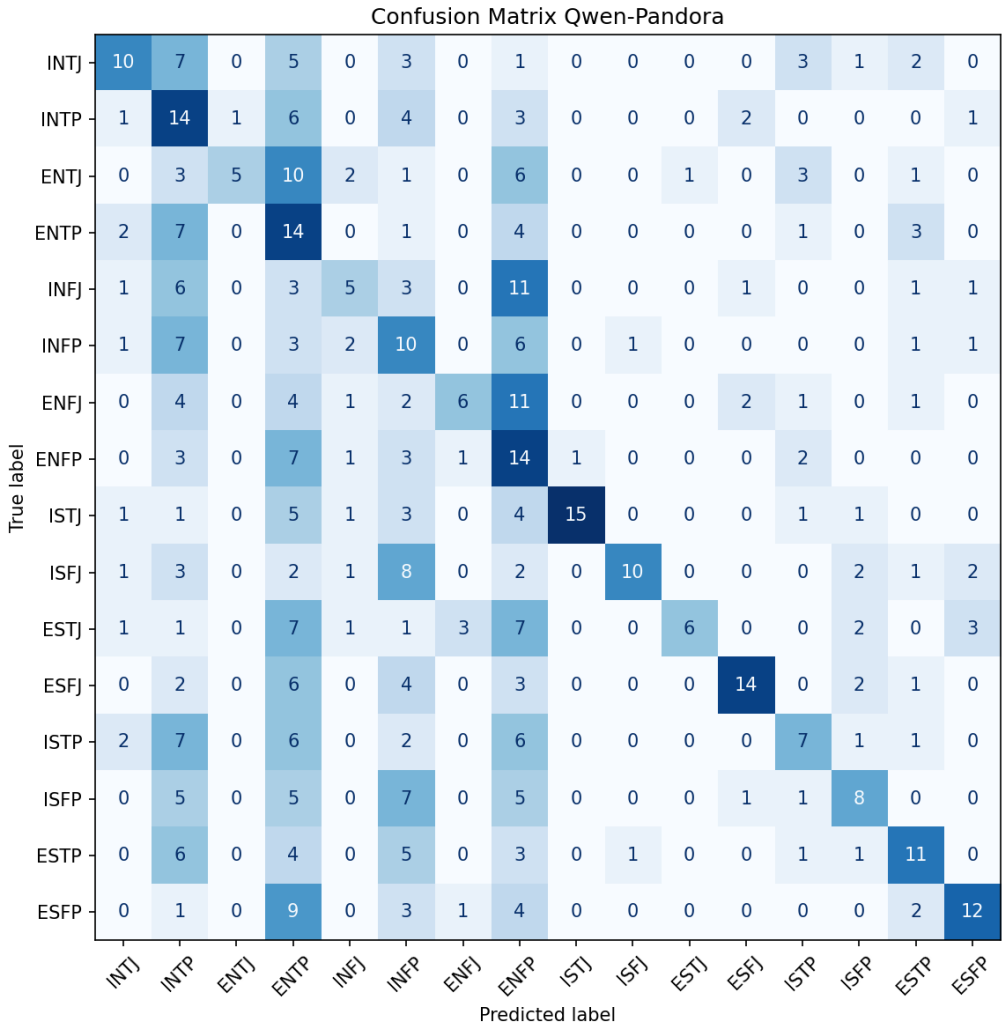

Figure 12: Llama-pandora shows the confusion matrix of ProtoMBTI$_{\text{LLaMA}}$ on the Pandora dataset. In contrast to its relatively strong performance on Kaggle, the model exhibits substantially weaker diagonal dominance, with high misclassification rates across nearly all types. Categories such as INFP, ENFP, and ISFP lose much of their discriminative clarity, while confusions like INTJ vs. INTP and ENTP vs. ENFP are pervasive. Even prototypical categories are less stable, reflecting the challenging lexical and semantic variability of Pandora. Compared to ProtoMBTI$_{\text{Qwen}}$ and ProtoMBTI$_{\text{GPT-4o-mini}}$, the performance of ProtoMBTI$_{\text{LLaMA}}$ under domain shift appears particularly sensitive, underscoring the necessity of integrating stronger prototype-aware transfer mechanisms to enhance robustness in cross-domain MBTI detection.

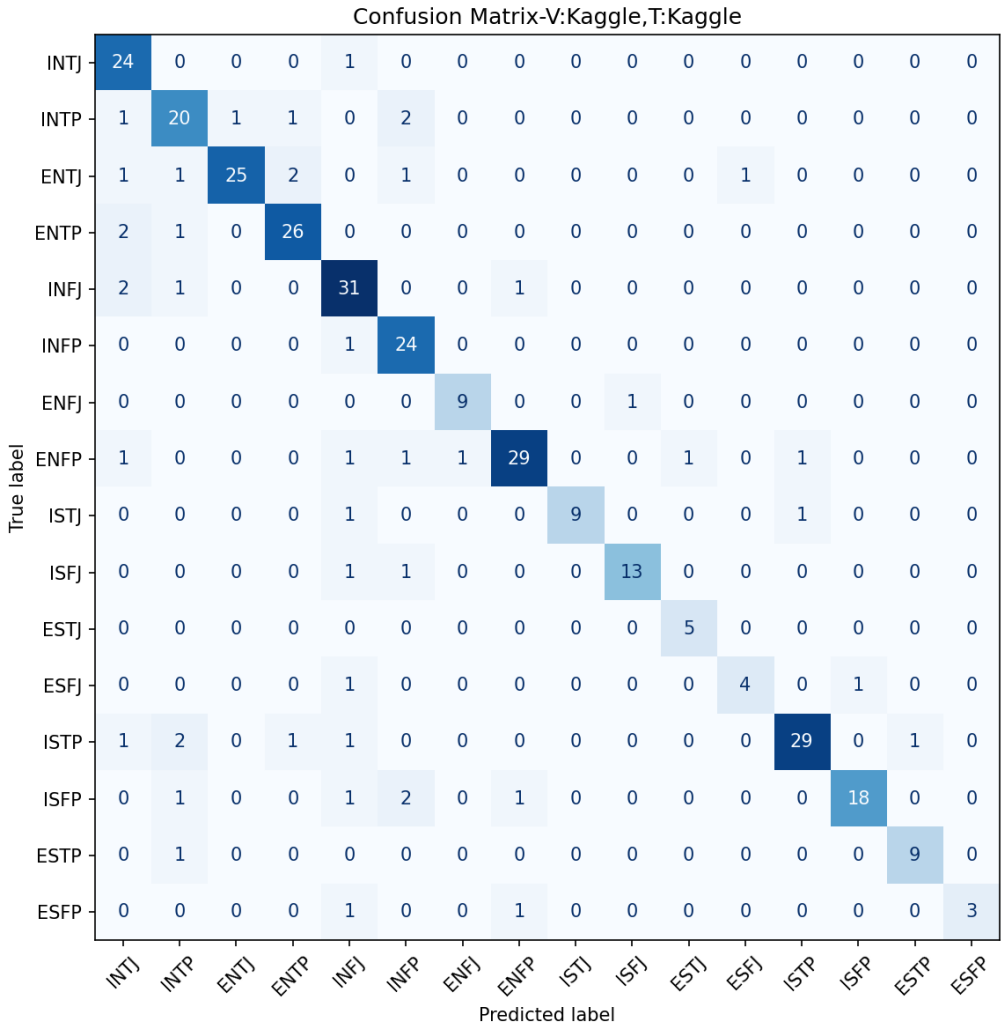

Figure 13: Mix-kaggle-kaggle presents the confusion matrix of ProtoMBTI trained on the mixed Kaggle–Pandora dataset and evaluated with Kaggle as both validation and test sets. The model achieves exceptionally strong diagonal dominance, with types such as INFJ, ENFP, and ISTP reaching near-perfect recognition. Misclassifications are sparse and largely confined to adjacent categories (e.g., INTJ vs. INTP), suggesting that exposure to both Kaggle and Pandora during training enables ProtoMBTI to generalize more robust prototypical boundaries within Kaggle data. Compared with single-dataset training, the mixed setup significantly reduces confusion, highlighting the benefits of prototype-informed learning when leveraging heterogeneous sources to stabilize within-domain MBTI detection.

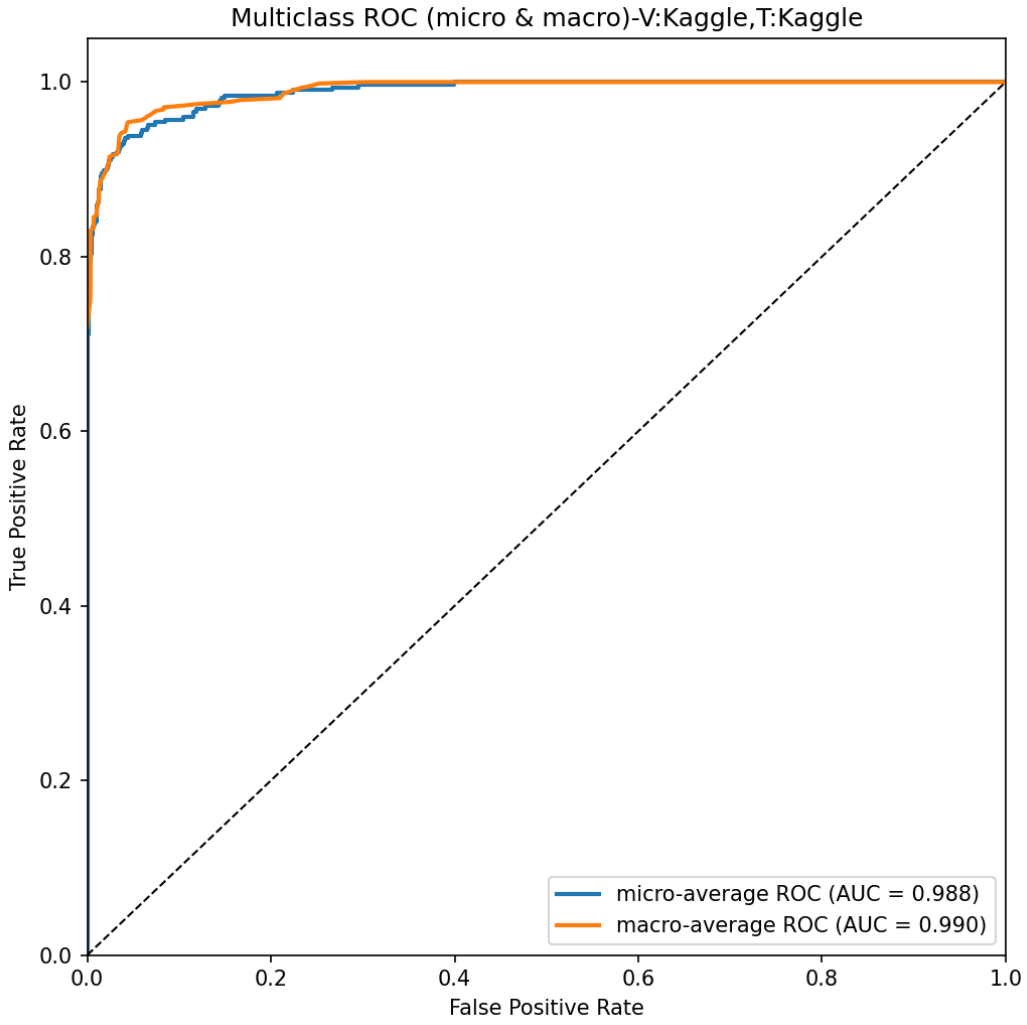

Figure 14: Roc-mix-kaggle-kaggle depicts the micro- and macro-average ROC curves of ProtoMBTI trained on the mixed Kaggle–Pandora dataset and validated/tested on Kaggle. Both curves achieve exceptionally high AUC scores (0.988 for micro-average and 0.990 for macro-average), indicating consistent performance across both frequent and minority MBTI types. The close alignment of micro and macro curves suggests that ProtoMBTI maintains balanced classification ability without overfitting to dominant categories. This outcome demonstrates that incorporating heterogeneous training data enhances the framework's ability to capture robust prototype boundaries, yielding near-optimal discrimination across all MBTI categories in the Kaggle domain.

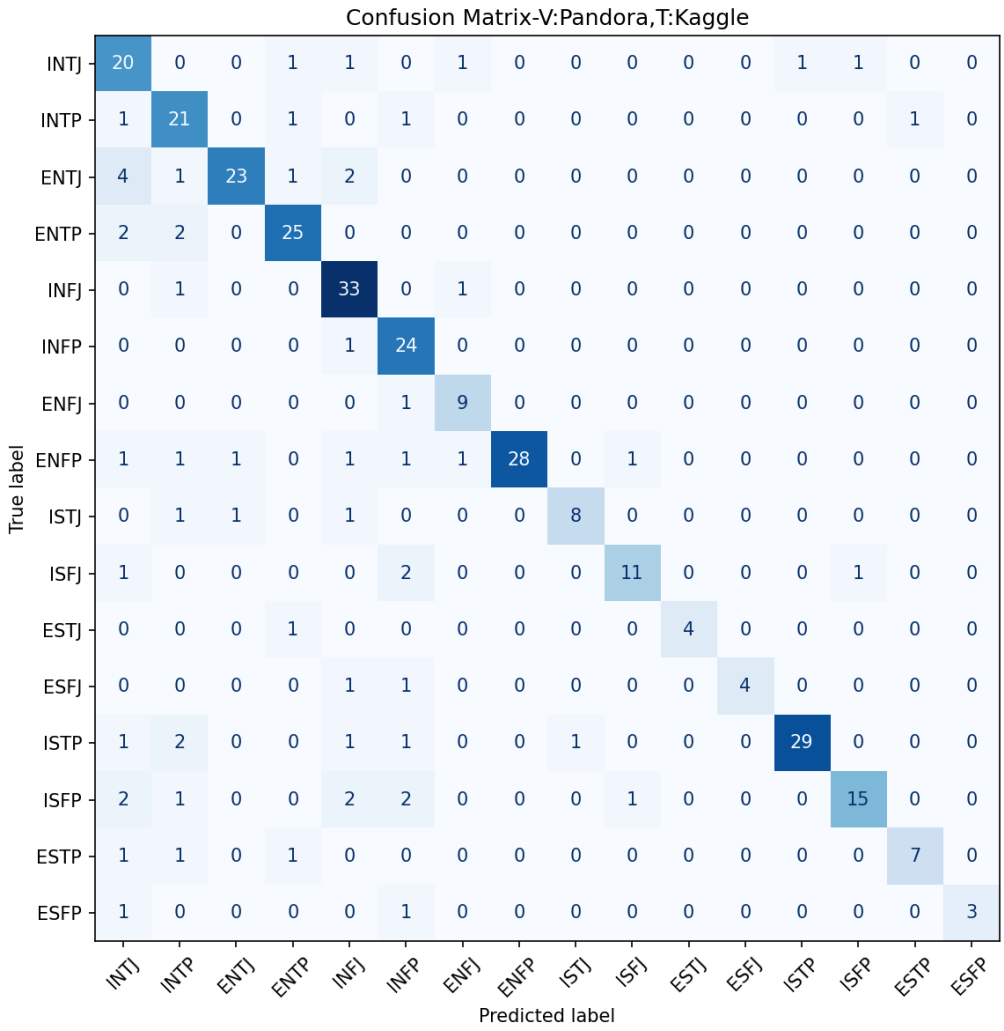

Figure 15: Mix-pandora-kaggle presents the confusion matrix of ProtoMBTI trained on the mixed Kaggle–Pandora dataset with Pandora as validation and Kaggle as test. The model achieves strong diagonal dominance across most categories, with especially high accuracy for INFJ, ENTP, and ISTP. While minor confusions remain (e.g., INTJ vs. INTP and ENTP vs. ENTJ), the overall misclassification rates are very low, indicating effective cross-domain transfer. Compared with training solely on Kaggle, the inclusion of Pandora during validation appears to strengthen the model's ability to capture more generalized prototypes, thereby improving robustness and maintaining high fidelity in Kaggle testing.

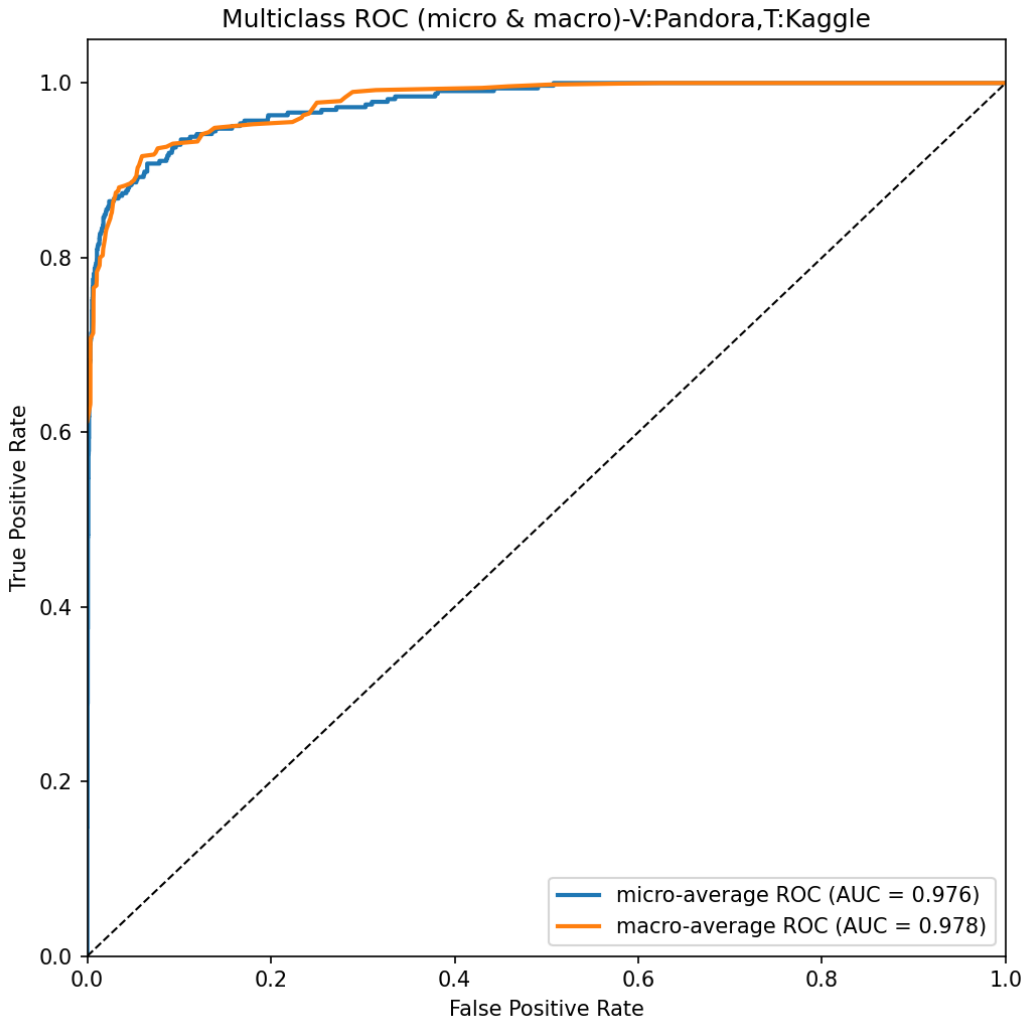

Figure 16: Roc-mix-pandora-kaggle illustrates the micro- and macro-average ROC curves of ProtoMBTI trained on the mixed Kaggle–Pandora dataset with Pandora as validation and Kaggle as test. Both curves achieve high AUC scores (0.976 for micro-average and 0.978 for macro-average), reflecting strong overall discriminative capacity. Compared with the Kaggle-only validation setting, performance remains robust but slightly reduced, indicating that cross-domain validation introduces additional variability. Nevertheless, the close alignment between micro and macro curves suggests that ProtoMBTI continues to balance frequent and minority MBTI categories effectively, highlighting its capacity to generalize prototype boundaries across heterogeneous sources while maintaining high accuracy on the Kaggle test set.

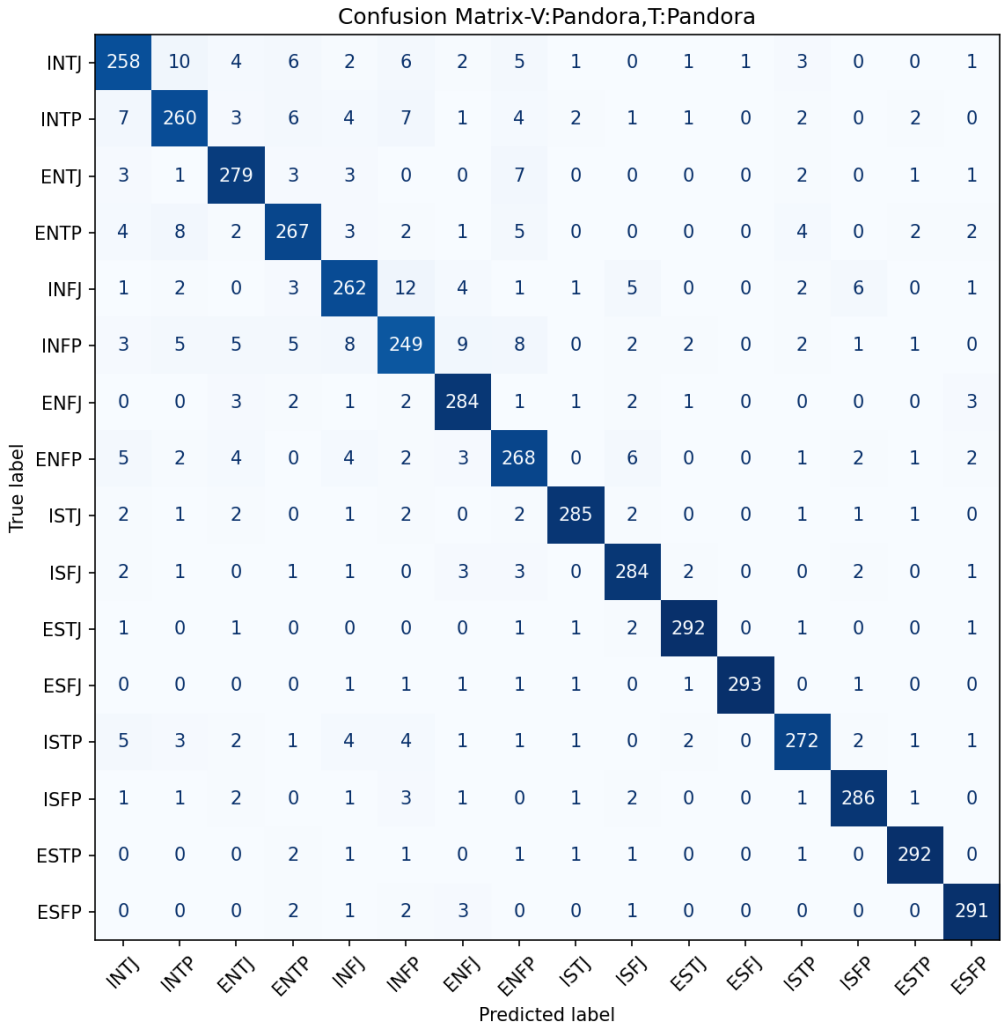

Figure 17: Mix-pandora-pandora presents the confusion matrix of ProtoMBTI trained on the mixed Kaggle–Pandora dataset with both validation and test sets drawn from Pandora. The model demonstrates very strong diagonal dominance across nearly all MBTI categories, with misclassifications kept to a minimal level despite Pandora's inherent lexical and semantic variability. Prototypical types such as ENFJ, ESFJ, and ESTP achieve near-perfect recognition, while even more ambiguous types (e.g., INFP, INFJ) maintain high accuracy. Compared with Kaggle testing, the Pandora–Pandora setting confirms that cross-domain training enables ProtoMBTI to capture generalized prototypes that align closely with the linguistic distribution of Pandora, thus validating the framework's robustness under within-domain evaluation of a heterogeneous dataset.

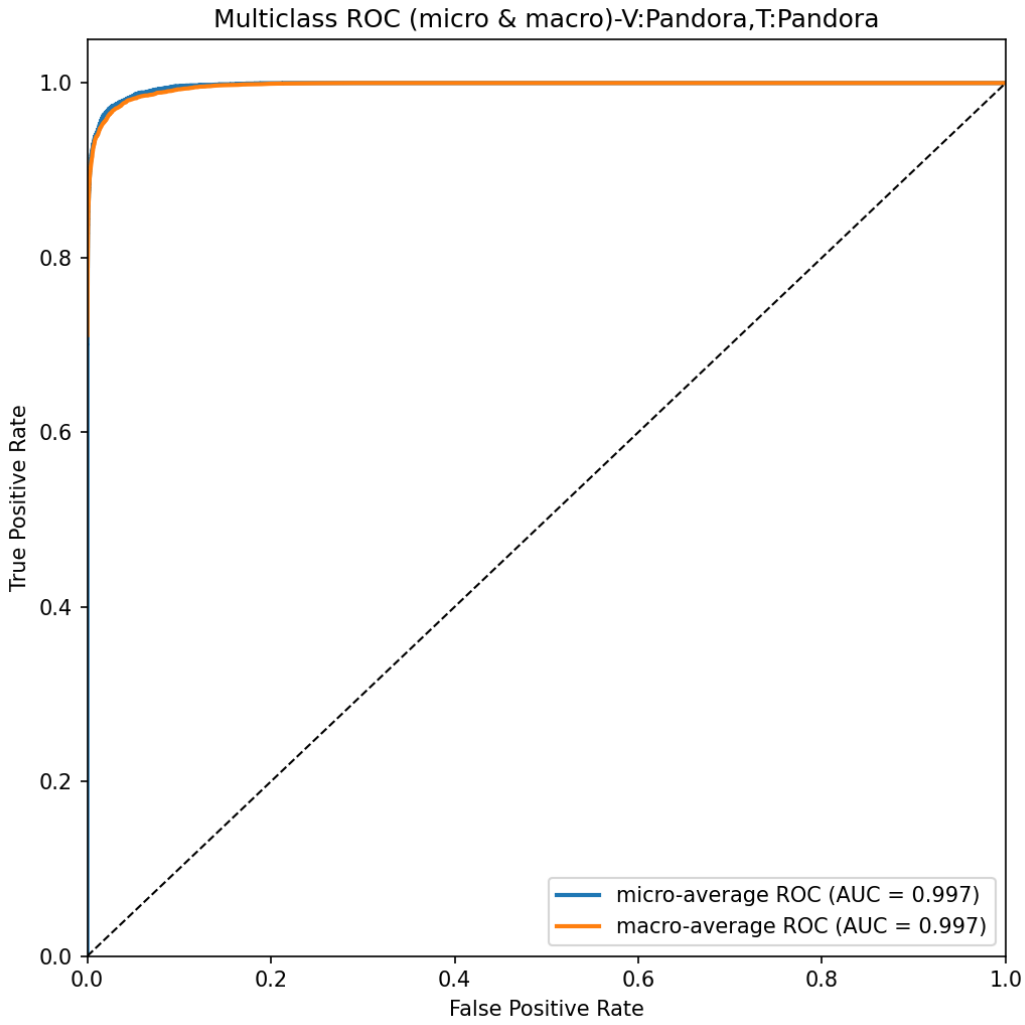

Figure 18: Roc-mix-pandora-pandora illustrates the micro- and macro-average ROC curves of ProtoMBTI trained on the mixed Kaggle–Pandora dataset with both validation and test sets drawn from Pandora. The AUC values for both micro- and macro-averages reach 0.997, indicating near-perfect discrimination across MBTI categories. The close overlap of the two curves demonstrates that ProtoMBTI performs equally well on both frequent and minority types, suggesting a balanced ability to capture prototype features regardless of class distribution. Compared with Kaggle test settings, this within-Pandora evaluation reveals that prototype-informed learning is particularly effective when both validation and testing share the same heterogeneous distribution, confirming the framework's stability and adaptability to complex linguistic variation.

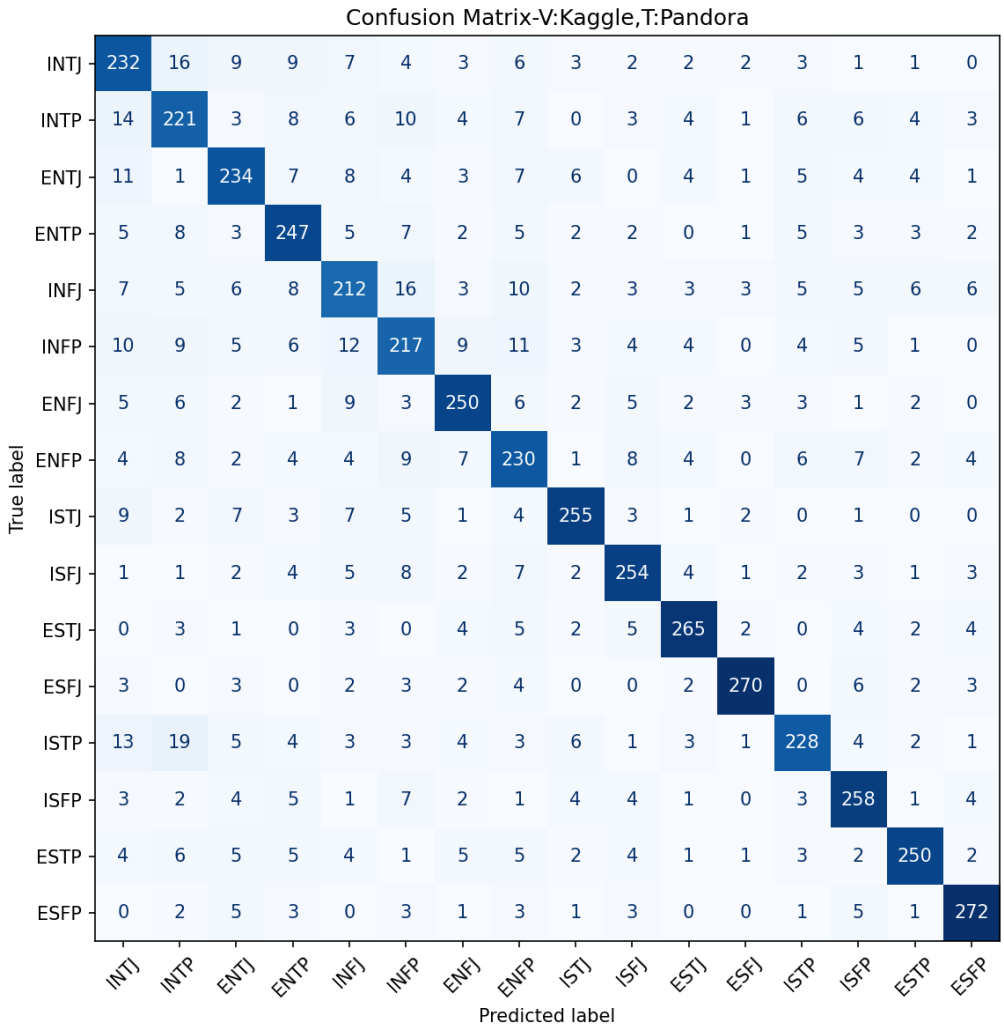

Figure 19: Mix-kaggle-pandora shows the confusion matrix of ProtoMBTI trained on the mixed Kaggle–Pandora dataset with Kaggle as validation and Pandora as test. While the model maintains strong diagonal dominance in several categories such as ENFJ, ESFJ, and ESTJ, error rates increase compared to within-domain evaluations. Notably, adjacent categories like INTJ vs. INTP and INFJ vs. INFP exhibit higher confusion, reflecting the difficulty of transferring prototype boundaries across datasets with different linguistic characteristics. Despite these challenges, overall classification remains robust, suggesting that mixed-domain training improves generalization but still leaves room for refining cross-domain prototype alignment to reduce boundary ambiguity.

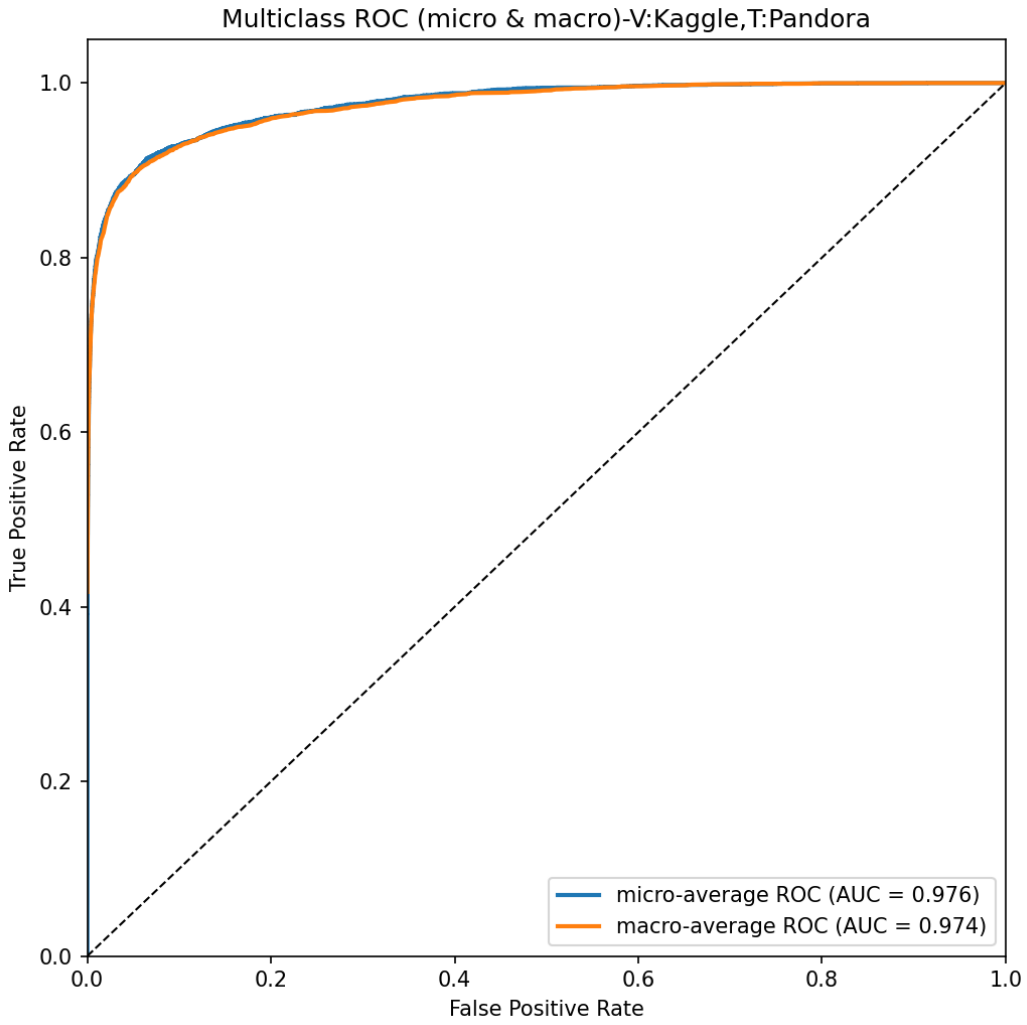

Figure 20: Roc-mix-kaggle-pandora depicts the micro- and macro-average ROC curves of ProtoMBTI trained on the mixed Kaggle–Pandora dataset with Kaggle as validation and Pandora as test. Both curves achieve strong AUC values (0.976 for micro-average and 0.974 for macro-average), confirming robust discriminative capacity across MBTI categories. Compared with Pandora–Pandora evaluation, performance slightly declines, reflecting the challenge of transferring prototype boundaries when validation and test distributions differ. Nevertheless, the close alignment of micro and macro curves suggests that ProtoMBTI preserves balanced treatment of both frequent and minority classes, demonstrating that prototype-informed generalization effectively mitigates, though does not eliminate, the difficulties of cross-domain adaptation.