Prompt-based Personality Profiling: Reinforcement Learning for Relevance Filtering

Jan Hofmann[1], **Cornelia Sindermann**[2], **and Roman Klinger**[3]
[1]Institut für Maschinelle Sprachverarbeitung, University of Stuttgart, Germany
[2]Computational Digital Psychology, Interchange Forum for Reflecting on
Intelligent Systems, University of Stuttgart, Germany
[3]Fundamentals of Natural Language Processing, University of Bamberg, Germany
jan.hofmann@ims.uni-stuttgart.de; roman.klinger@uni-bamberg.de
cornelia.sindermann@iris.uni-stuttgart.de

Abstract

Author profiling is the task of inferring characteristics about individuals by analyzing content they share. Supervised machine learning still dominates automatic systems that perform this task, despite the popularity of prompting large language models to address natural language understanding tasks. One reason is that the classification instances consist of large amounts of posts, potentially a whole user profile, which may exceed the input length of Transformers. Even if a model can use a large context window, the entirety of posts makes the application of API-accessed black box systems costly and slow, next to issues which come with such "needle-in-the-haystack" tasks. To mitigate this limitation, we propose a new method for author profiling which aims at distinguishing relevant from irrelevant content first, followed by the actual user profiling only with relevant data. To circumvent the need for relevance-annotated data, we optimize this relevance filter via reinforcement learning with a reward function that utilizes the zero-shot capabilities of large language models. We evaluate our method for Big Five personality trait prediction on two Twitter corpora. On publicly available real-world data with a skewed label distribution, our method shows similar efficacy to using all posts in a user profile, but with a substantially shorter context. An evaluation on a version of these data balanced with artificial posts shows that the filtering to relevant posts leads to a significantly improved accuracy of the predictions.

1 Introduction

Author profiling aims at inferring information about individuals by analyzing content they share. A large and diverse set of characteristics like age and gender (Koppel et al., 2002; Argamon et al., 2003; Schler et al., 2006), native language (Koppel et al., 2005), educational background (Coupland, 2007), personality (Pennebaker et al., 2003; Golbeck et al., 2011; Kreuter et al., 2022), or ideology (Conover et al., 2011; García-Díaz et al., 2022) have been studied so far. Author profiling is often formulated supervised learning in which a full user profile with possibly hundreds or thousands of individual textual instances constitutes the input.

Despite the success of deep learning strategies in various natural language processing tasks, such approaches often underperform when applied to author profiling (Lopez-Santillan et al., 2023). One factor contributing to this may be that models like BERT (Devlin et al., 2019) have constraints on the length of the input they can process, preventing them from processing all content linked to an author at once. Another reason for this may be that not all content shared by an author is equally useful when predicting certain characteristics. Some of the content may even be considered noise, making it difficult for machine learning models to grasp patterns needed when predicting specific characteristics of an author – we are faced with a "needle-in-the-haystack" challenge[1].

With this paper, we approach this challenge and propose to prefilter posts to distinguish between helpful and misleading content before inferring a characteristic. Thereby, accuracy of automated profiling systems could be enhanced, and computational requirements could be reduced. To induce such filter without data manually annotated for relevancy, we study reinforcement learning with a reward function that represents the expected performance gain of a prompt-based system. Therefore, our approach only requires a prompt for a large language model (LLM) and leads to a prefiltering classifier that can, at test time, be applied with a limited number of queries to a large language model. In contrast to retrieval augmented generation setups (RAG, Gao et al., 2024), our setup has the advantage that it does not need to rely on the

[1] https://github.com/gkamradt/LLMTest_NeedleInAHaystack

Proceedings of the 1st Workshop for Research on Agent Language Models (REALM 2025), pages 1–16
July 31, 2025 ©2025 Association for Computational Linguistics

ad-hoc abilities of a retrieval system.

Our contributions are therefore[2]:

- We propose a novel reinforcement learning-based relevance filtering method that we optimize with a reward inferred from the performance of a prompt-based zero-shot predictor.
- We evaluate this method on personality prediction and show that a similar performance can be reached with limited, automatically filtered data, leading to a cheaper and environmentally more friendly social media analysis method.
- We show the potential to improve the predictive performance with a partially artificial, balanced personality prediction corpus that we create via data augmentation. Here, the prediction is significantly more accurate with substantially smaller context.

2 Related Work

2.1 Zero-Shot Predictions with Large Language Models

The terms prompt-based learning or in-context learning point at methods in which we use an LLM's ability to generate text as a proxy for another task. This approach has proven effective for a variety of tasks (Yin et al., 2019; Gao et al., 2021; Cui et al., 2021; Ma et al., 2022; Sainz et al., 2021; Tu et al., 2022, i.a.). For example, in a sentiment polarity classification, a classification instance could be combined with a prompt that requests a language model to output a word that corresponds either to a positive or a negative class ("The food is very tasty." – "This review is positive/negative.").

State-of-the-art text classification methods employ the Transformer architecture (Vaswani et al., 2017), which are both deep and wide neural networks, optimized for parallel processing of input data. However, they have a constrained input length: BERT (Devlin et al., 2019) can use 512 tokens, GPT-3.5 and Llama 2 (Touvron et al., 2023a) allow 4096 tokens, and GPT-4 (Brown et al., 2020) considers 8192 tokens[3]. This situation makes the analysis of long texts challenging and is the motivation for our work: automatically restricting the data to be analyzed in a prompt to the most informative segments.

One approach to solve this issue is to combine language-model based text generation with information retrieval methods. In so-called RAG (retrieval-augmented generation) approaches, the relevant passages for a generation task are first retrieved in text-search manner, which are then fed into the context of the language model (Gao et al., 2024). In contrast to our approach, such methods are optimized for ad-hoc retrieval, to work with any given prompt.

I see Myself as Someone Who ...	Variable	Cor.
... does a thorough job	Consc.	+
... can be somewhat careless	Consc.	−
... is talkative	Extrav.	+
... is reserved	Extrav.	−
... worries a lot	Neurot.	+
... is relaxed or handles stress well	Neurot.	−

Table 1: Example items from the BFI-44 questionnaire (John et al., 1991). Negative scores indicate reversed-scored items.

2.2 Personality in Psychology

Stable patterns of characteristics and behaviors in individuals are known as personality. Personality traits characterize differences between individuals present over time and across situations. Several theories have been proposed attempting to categorize these differences (e.g., Cattell, 1945; Goldberg, 1981; McCrae and John, 1992). Such theories include biologically oriented ones (Cloninger, 1994), as well as lexical approaches including the Five Factor Model (Digman, 1990) and the HEXACO Model (Ashton and Lee, 2007).

The Five Factor Model is one of the most extensively researched and widely accepted models among personality psychologists, and proposes that personality can be described based on five broad domains, the so-called *Big Five* of personality. Oftentimes, the Big Five are named: *openness to experience* (e.g., artistic, curious, imaginative), *conscientiousness* (e.g., efficient, organized, reliable), *extraversion* (e.g., active, outgoing, talkative), *agreeableness* (e.g., forgiving, generous, kind), *neuroticism* (e.g., anxious, unstable, worrying) (Costa and McCrae, 1992). The Five Factor Model originates from the lexical hypothesis stating that personality traits manifest in our language, because we use it to describe human characteristics (Brewer, 2019; Goldberg, 1990; John and Srivastava, 1999).

A commonly used approach to assess the Big Five in individuals is the application of self-report questionnaires, like the *Big Five Inventory* (BFI) de-

[2] The source code used in this study is available at: https://github.com/bluzukk/rl-profiler

[3] https://agi-sphere.com/context-length/, access date 2024-07-22

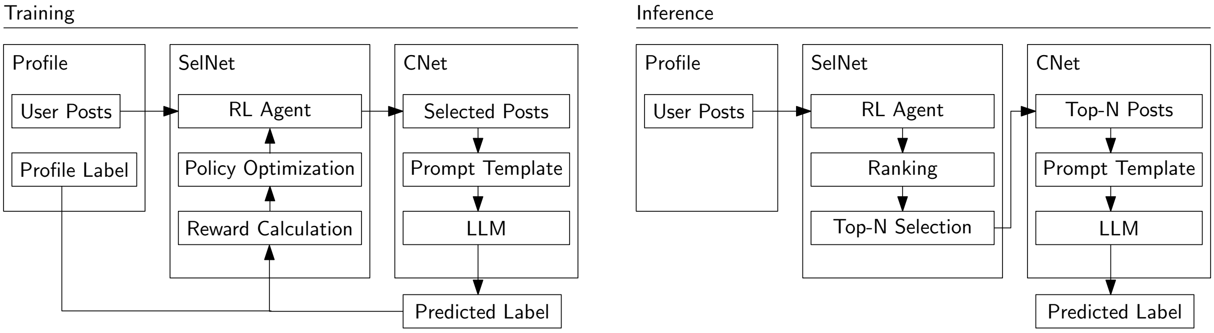

Figure 1: Overview on the workflow of the RL-Profiler (RL: Reinforcement Learning; SelNet: Selection Network; CNet: Classification Network; LLM: Large Language Model).

veloped by (John et al., 1991). This questionnaire consists of 44 short phrases describing a person, and individuals are asked to rate the extent to which they agree that each of these items describes themselves on a five-point Likert scale from 1 (strongly disagree) to 5 (strongly agree). Table 1 shows examples of these items. For example, if a person strongly agrees to "being someone who is talkative" and other related items of the same scale, this can indicate a high level of *extraversion*.

2.3 Automatic Personality Prediction from Text

One of the first attempts to personality prediction in social media was proposed by Argamon et al. (2005), predicting *extraversion* and *neuroticism* from essays on a binary scale, i.e., predicting either a *low* or *high* level of a trait. Further, Schwartz et al. (2013) explored written text on the social media platform Facebook, and found that language use not only differs among people of different age and gender but also among people rated differently along the Big Five traits. In the 2015 PAN shared task (Rangel et al., 2015) the best results predicting personality were obtained by Sulea and Dichiu (2015) using ridge regression in combination with tf–idf weighted character n-grams.

Since then, various deep learning approaches have been applied in attempt to predict personality of users of social media platforms (Khan et al., 2020). These are, however, challenged by the nature of the task: not all posts linked to individuals may be useful, since content and tone of post from the same author may vary depending on factors such as mood, current events, or specific interest at a given time. Personality, however, characterizes differences between persons present over time and across situations. Further, as not all traits are strongly related to each other (Oz, 2015), some posts might provide insights into one trait but not the other. Consequently, there have been very limited efforts to predict personality with the help of large language models (Chinea-Rios et al., 2022). Accordingly, we argue that systems would benefit from learning to differentiate between relevant and misleading text instances by an author.

3 RL-Profiler: Reinforcement Learning by LLM-based Performance Rewards

We assume a profile consisting of a set of textual instances as input, with annotations on the profile, but not instance level, during training. We optimize the instance-relevance filter with information from a profile-level prediction model. This filter decides which textual instances are informative and should be used for the profile-level decision.

Figure 1 illustrates this architecture. Our **RL-Profiler** is devided into (1) the **Selection Network** (**SelNet**) and (2) the **Classification Network** (**CNet**). SelNet corresponds to an agent in the RL sense and selects textual instances from a profile. CNet then uses these instances to predict a profile-level label. During training (left side of Figure 1), we compare this prediction with the given profile-level ground truth to calculate a reward.

3.1 Selection Network (SelNet)

The core component of SelNet is the RL agent adopting a stochastic policy $\pi(a \mid s, \theta)$ with the binary action space $\mathcal{A} = \{Select, Reject\}$, which we implement as a transformer-based classifier with a binary classification head. Here, θ represents the trainable parameters, $a \in \mathcal{A}$ denotes an action, and s is a single text instance from a profile.

During training, an action is sampled from the probabilities given by the agent's current policy. This ensures that the agent is exploring different

Algorithm 1 RL-Profiler: Learning Algorithm

```
1:  Input: Policy $\pi_\theta$ with action space $\mathcal{A} = \{select, reject\}$, a training set $D$ with a set of profiles $\{P_1, ..., P_i\}$, each associated with a set of text instances $\mathcal{S}_P$ and ground-truth $y_P$, and training epochs $E > 0$
2:  Pre-train $\pi(a|s, \theta)$ using NPMI-Annotations
3:  for Epoch $i \leftarrow 1 \ldots E$ do
4:      Shuffle training set $D$
5:      for Profile $P$ in $D$ do
6:          $C \leftarrow \{\}$    ▷ Set of selected instances.
7:          for Instance $s_t$ in $\mathcal{S}_P$ do
8:              Sample action $a_t \in \{select, reject\}$ from $\pi(a_t|s_t, \theta)$
9:              if $a_t = select$ then
10:                 $C \leftarrow C \cup s_t$
11:             end if
12:         end for
13:         $\hat{y}_P \leftarrow$ Prediction of CNet using $C$
14:         $R \leftarrow$ Reward using $y_P$, $\hat{y}_P$ and $C$
15:         $\theta \leftarrow \theta + \alpha \sum_{t=1}^{|\mathcal{S}_P|} (R - b) \ln \nabla_\theta \pi(a_t|s_t, \theta)$
16:     end for
17: end for
```

actions for the same input and the corresponding reward during training. For inference, we adapt the behavior of SelNet: given the set of instances from a profile, the policy of the trained agent is first predicting probabilities for each instance. Then, all instances are ranked by the predicted probability of *selecting* them and only the top-N instances are fed to CNet predicting a characteristic. This ensures that during inference, the agent is no longer exploring different actions but only exploits knowledge learned during training. Further, this forces SelNet to always select a fixed number of instances N from profiles, eliminating the possibility of selecting no instance at all.

3.1.1 Training the RL agent

Algorithm 1 presents the method to train the RL-agent. We use training data consisting of profiles with associated ground-truth labels, and iterate multiple times over this training dataset (Line 3). In each epoch, profiles in the given dataset are randomly arranged (Line 4). Given a single profile from this training set, each instance from the profile is processed individually (Line 7–12): the agent's current policy π predicts a probability for a single instance being relevant or irrelevant. In other words, the agent predicts probabilities whether to select or reject an instance. During training, this action is sampled according to the predicted probabilities (Line 8). The selected text instances are collected in a set C (Line 10), and fed to CNet predicting a profile-level label (Line 13). Using this prediction and the ground-truth label, we then calculate a learning signal R (cf. Equation 1) to update the policy of the agent (Line 14–15).

Reward. After collecting a subset of instances C from a profile, CNet uses this set to predict a label. We use this prediction $\hat{y} \in \{0, 1\}$, the ground-truth label $y \in \{0, 1\}$ associated to the profile, and the number of selected instances $|C|$ to calculate the reward R:

$$R(y, \hat{y}, C) = -2 + \text{sign}(|C|)(3 - 2|y - \hat{y}|) - \lambda|C| \tag{1}$$

with λ being a hyperparameter that aims to decrease the reward based on the number of selected instances. With this formulation of the reward function, we summarize three cases: (1) if the predicted label is equal to the ground-truth annotation we obtain $+1 - \lambda|C|$, (2) if the predicted label is not equal to the ground-truth annotation we obtain $-1 - \lambda|C|$, and (3) if the set of selected posts is empty the reward is set to -2. Maximizing this reward is the goal of the agent. Therefore, the agent needs to learn to *select* instances from profiles such that CNet predicts the ground-truth label correctly, while *rejecting* as many instances as possible without rejecting all of them.

Policy Optimization. To optimize the behavior of the agent based on this reward, we adapt the update rule of the REINFORCE algorithm (Williams, 1992): given a profile P associated with a set of text instances $\mathcal{S}_P$, the parameters in θ are updated based on the reward R and the predicted probabilities of each of the chosen actions following the current policy π:

$$\theta \leftarrow \theta + \alpha \sum_{t=1}^{|\mathcal{S}_P|} (R - b) \ln \nabla_\theta \pi(a_t|s_t, \theta), \tag{2}$$

where b is a baseline. For simplicity, the calculation of b is not shown in Algorithm 1. In our approach we calculate this baseline as the moving average reward given the last 10 update steps, estimating the expected reward given the current policy.

3.1.2 Pre-training using Mutual Information

To improve stability of the training process of the RL agent (Mnih et al., 2015), we add a supervised pre-training step based on information theoretic measures that associate words to labels. We use normalized pointwise mutual information (NPMI, Bouma, 2009; Church and Hanks, 1990) to weigh the association between each word w present in text instances provided by a profile and the corresponding ground-truth label c:

$$\mathrm{NPMI}(w;c) = \left(\ln \frac{p(w,c)}{p(w)p(c)}\right) \Big/ -\ln p(w,c)\,. \quad (3)$$

We estimate these probabilities from the training set, and use the NPMI weights to calculate a relevance-score for individual instances. Here, for each instance $s \in \mathcal{S}_P$ associated to a profile P we first calculate scores for each class c:

$$\mathrm{score}_c(s) = \sum_{w \in s} \mathrm{NPMI}(w;c), \quad (4)$$

and then a relevance-score considering all classes:

$$\text{r-score}(s, c_1, c_2) = \frac{\left|\mathrm{score}_{c_1}(s) - \mathrm{score}_{c_2}(s)\right|}{\left|\{w \mid w \in s\}\right|}, \quad (5)$$

where c_1 and c_2 are the possible labels in a given author profiling problem. Note that, for simplicity, we only consider binary profile-level labels in this study (*high* or *low*), and it is therefore sufficient to define this score for two classes. After calculating a relevance-score (r-score) for each text instance of all authors in the training set, we annotate the top-M instances of each author w.r.t. the highest relevance-scores as *relevant* while others are marked as *irrelevant*. These annotations are then used as a supervised learning signal for pre-training the RL agent (Line 2 in Algorithm 1).

3.2 Classification Network (CNet)

The combination of SelNet and CNet forms a pipeline predicting a label given textual instances from a profile. Given a set of selected text instances, CNet is responsible for predicting this label. In this work, we propose to use a large language model in a prompting setting for this purpose, since such a zero-shot setup does not require any task specific training. Here, the classification task of predicting a label from the selected text instances is verbalized, i.e., reformulated to match the LLM's pre-training objective. CNet therefore creates a prompt using the selected text instances by SelNet and a pre-defined prompt template. We derive the classification result from the tokens the LLM generates in response to such a prompt. The prompt setup is explained in the next section.

4 Experiments

4.1 Experimental Setting and Training Details

We implement RL-Profiler using the PyTorch (Paszke et al., 2019) and HuggingFace's Transformer (Wolf et al., 2020) libraries. For parameterizing the policy of the agent in SelNet, we use *bert-base-uncased*[4], and feed the *[CLS]* token into a binary classification head with a dropout (Srivastava et al., 2014) of 20%. We pretrain the agent using NPMI annotations marking the top-10 (top-M) instances as relevant for 2 epochs, and fix the maximum epochs for reinforcement learning to 200. During reinforcement learning, we fix $\lambda = .05$ for reward calculations, and adapt early stopping by evaluating the current policy on validation data after each epoch using different settings for top-N. Here, we validate the current policy by using the 5, 10, 20, 30, and 50 posts ($N \in \{5, 10, 20, 30, 50\}$) of each profile the current policy predicts the highest probabilities of selecting them. For each of these settings, we save the best model checkpoint based on macro F_1 score. In both training phases we use AdamW (Kingma and Ba, 2015; Loshchilov and Hutter, 2019) with a learning rate of 10^{-6}.

For the classification of the selected text instances (CNet) we use *Llama 2 13B-Chat*[5] (Touvron et al., 2023b) with GPTQ (Frantar et al., 2022), and fix temperature to 0.8 and top-p to 0.9 for all experiments. For all Big Five traits we design individual prompts. Figure 2 show such a prompt for predicting a level of extraversion. Our prompts consist of a system prompt requesting single word answers, context about a trait, the posts selected from a profile, and an instruction. The context stems from items of the BFI-44 (John et al., 1991) questionnaire used to score a particular trait. These items are exemplarily added for a *high* level ("A person with a high level of extraversion may see themselves as ..."), while items that are scored in reversed are added as context for a *low* level (Table 1 shows examples of such items for other traits).

4.2 Corpora

We evaluate our approach on the English subset of the publicly available PAN-AP-2015 data[6] (Rangel et al., 2015). The personality trait annotations in

[4] https://huggingface.co/google-bert/bert-base-uncased

[5] https://huggingface.co/TheBloke/Llama-2-13B-chat-GPTQ

[6] https://zenodo.org/records/3745945

```
<s>[INST] <<SYS>>
one word response
<</SYS>>

Recall the personality trait extraversion.
A person with a high level of extraversion may see themselves as someone who is talkative, or {...}
A person with a low level of extraversion may see themselves as someone who is reserved, or {...}

Consider the following tweets written by the same person:
{tweets}
Does this person show a low or high level of extraversion? Do not give an explanation. [/INST]
```

Figure 2: Prompt template used in CNet for predicting a level of *extraversion*.

	Training		Validation		Testing	
Class	High	Low	High	Low	High	Low
Open.	119	1	30	1	137	1
Consc.	93	3	24	2	113	10
Extrav.	96	12	24	3	114	6
Agree.	90	15	24	4	108	11
Neurot.	83	30	22	8	91	39

Table 2: Corpus statistics of the splits derived from the PAN-AP-2015 (Rangel et al., 2015) corpus (in numbers of profiles).

this corpus are derived from self-assessed BFI-10 online tests (Rammstedt and John, 2007), a short version of the BFI-44. Here, for each author, a score between -0.5 and 0.5 is provided for each Big Five trait. We convert these scores to binary values at a threshold of 0, and use 20% of the training data for validation for each trait, while ensuring a similar class distribution in these sets. Note that this results in different dataset splits for each trait. Table 2 summarizes the statistics of the corpora we derive. On average over all traits and splits, we find that each profile consists of 92.3 individual posts.

4.3 Baselines and Derived Systems

We compare our method to two supervised-learning based approaches, and four systems we directly derive from our method:

Baseline-R: Regression Classifier. For the first baseline, we adapt the best performing system from the 2015 PAN shared task to fit the binary profiling problem. In this system, Sulea and Dichiu (2015) use a ridge regression model with character n-gram tf-idf features. We adapt this approach by converting the ridge regressor into a ridge classifier.

Baseline-B: BERT Classifier. We adapt BERT (*base-uncased*) to the binary classification problem using a classification head. Since the input to BERT is restricted to a maximum of 512 tokens, all posts associated with an author can not be presented to this model at once. Therefore, we propagate the profile-level ground-truth to individual posts, and train BERT on post-level for 2 epochs using a learning rate of $2 \cdot 10^{-5}$ with cross-entropy loss weighted by class distribution. To obtain a profile-level prediction we draw a majority vote from the predictions on individual posts.

ALL+CNet. We explore a variation of RL-Profiler that skips the selection process of SelNet. In *ALL+CNet*, all posts from an author are given to CNet. Note that this is possible in our experimental setting because the data we use only contains a subset of posts from each user's profile.

RND+CNet. In this variation of RL-Profiler, we replace the reinforcement learning agent in SelNet with a random selection of N posts.

PMI+CNet. In this system, the selection process using the trained agent is replaced by selecting N posts based on their *relevance-score* (Equation 5). Here, instances are ranked based on NPMI information and the top-N instances are directly given to CNet. With this system, we aim to provide insights on performance of such a selection system when simply relying on information theoretic measures.

PT+CNet (Pre-train+CNet). Further, the agent trained using reinforcement learning can be replaced by an agent that is only pre-trained on NPMI-Annotations, i.e., we stop training the agent after Line 2 in Algorithm 1.

4.4 Evaluation Procedure and Metrics

We evaluate our experiments using macro-average and weighted-average F_1 scores (average weighted by the number of instances per class).

Performance during evaluation of the individual systems in this study can vary between runs. This is, for example, due to the non-deterministic output generated by the LLM. Therefore, we average scores of 10 individual runs.

		Open.		Consc.		Extrav.		Agree.		Neur.	
System	top-N	m-F$_1$	w-F$_1$	m-F$_1$	w-F$_1$	m-F$_1$	w-F$_1$	m-F$_1$	w-F$_1$	m-F$_1$	w-F$_1$
Baseline-R	all	49.8±0.0	98.9±0.0	47.9±0.0	88.0±0.0	82.5±0.0	96.7±0.0	59.8±0.0	88.2±0.0	66.8±0.0	73.6±0.0
Baseline-B	all	56.5±14.1	99.0±0.2	58.8±8.4	88.8±0.8	76.9±12.4	93.6±5.5	66.2±4.6	88.2±3.2	67.8±1.8	73.6±1.1
ALL+CNet	all	49.8±0.0	98.9±0.0	47.0±1.6	70.5±1.5	48.4±0.1	92.1±0.2	52.5±2.2	75.8±1.3	42.7±2.6	57.0±1.9
RL-Profiler	best	47.7±0.6	94.6±1.2	44.6±2.2	63.9±2.7	57.0±5.7	92.3±0.8	43.1±2.1	70.8±1.3	39.3±2.3	47.0±2.4
RND+CNet	best	49.6±0.1	98.5±0.3	33.4±1.9	45.7±3.0	48.3±0.2	91.8±0.3	41.8±2.0	58.1±1.9	38.8±1.7	46.1±0.8
PMI+CNet	best	49.4±0.2	98.0±0.3	35.4±1.9	48.8±2.9	58.8±3.8	91.4±1.1	42.3±1.6	58.1±1.9	38.0±1.6	42.5±1.7
PT+CNet	best	49.1±0.3	97.5±0.6	34.5±1.9	48.6±1.9	48.2±0.2	91.7±0.4	36.7±2.2	48.1±2.8	38.8±1.7	50.9±1.6

Table 3: Macro F$_1$ (m-F$_1$) and weighted average F$_1$ (w-F$_1$) scores for all models on testing data (average of 10 runs with standard deviation). For models with top-N parameter (lower part in this table), the best setting based on macro F$_1$ score on validation data is chosen for each trait (validation results with all settings for top-N shown in Table 6).

4.5 Results

In this section, we analyze the results of our experiments on the PAN-AP-2015 corpus. To evaluate the effect of the number of selected posts per profile we use validation data: for each trait we select the setting for top-N that produces the best results w.r.t. macro F$_1$ score on validation data, individually for each method/baseline and trait. We provide detailed results for all models and settings for top-N on validation data in the Appendix A.1.

Does the prediction with partial data perform on par or worse in comparison to using all data? Table 3 shows the results. Here, we are in particular interested whether our approach is preferable compared to using all posts of profiles in a zero-shot setting. We therefore compare the third and fourth row in this Table and find that, for all traits except for *extraversion*, our approach (*RL-Profiler*) performs only slightly worse compared to using all posts (*ALL+CNet*). On average over all traits, we find that *RL-Profiler* performs worse by 1.8pp macro F$_1$ (46.3% vs. 48.1%) and 5.2pp weighted F$_1$ (78.9% vs. 73.7%). This is, although our method only uses 10 posts from each profile on average over all traits while the *ALL+CNet* system uses 92.9.

Is RL-Profiler better than randomly selecting instances? One option to limit the amount of data is to choose a number of posts at random. We therefore compare the fourth and fifth row in Table 3, and observe that out method (*RL-Profiler*) is outperforming a random selection (*RND+CNet*) for almost all traits (except *openness*, which has a majorly skewed class distribution). On average over all traits, we find that our method improves macro F$_1$ by 3.9pp (46.3% vs. 42.4%) and weighted F$_1$ by 5.7pp (73.7% vs. 68.0%) compared to random selections. This is although the *RND+CNet* system is using N=50 posts, on four of the five traits, while the proposed system only uses N=5 for these traits (since these settings for N produced the best results for these approaches during validation).

Is the RL necessary or would a purely statistical selection suffice? This finding prompts the question of whether alternative selection methods that bypass costly training could replace our trained RL agent. To explore this, we compare our approach (Row 4) to its variants, *PMI+CNet* (Row 6) and *PT+CNet* (Row 7), and observe that these alternatives generally underperform compared to the trained agent, Further, we compare *RL-Profiler* to the two supervised learning-based systems *Baseline-R* and *Baseline-B*, and find that, on average over all traits, performance decreases by 15.1pp and 18.9pp macro F$_1$, respectively, when using our zero-shot approach.

Computational Analysis. We perform our experiments on a single NVIDIA RTX A6000 (48GB) GPU with AMD EPYC 7313 CPU and present the average prediction time per profile on testing data for different zero-shot systems in Table 4. For the *RL-Profiler* and *RND+CNet* systems, the reported time includes both the time required to select a number of posts from each profile – using the trained agent or random selection, respectively – and the time taken by CNet to generate a prediction based on the selected posts. For the *ALL+CNet* system this time only reflects the duration required to retrieve a prediction from CNet. We find that prediction time is substantially reduced by a reduced number of selected posts. For example, when predicting extraversion, the average prediction time for a profile is reduced by more than 76% moving from 1.65s to 0.38s on the comparison between our method and the system using all available posts in a zero-shot setting.

Variable	*RL-Profiler*	*RND+CNet*	*ALL+CNet*
Open.	0.54 (5)	1.11 (50)	1.72 (94.3*)
Consc.	0.88 (5)	1.29 (50)	2.11 (93.7*)
Extrav.	0.38 (5)	1.10 (50)	1.65 (92.4*)
Agree.	0.61 (5)	1.12 (50)	1.57 (91.9*)
Neur.	1.12 (30)	1.03 (30)	1.78 (92.1*)

Table 4: Average prediction time in seconds per profile on testing data. For the *RL-Profiler* and *RND+CNet* system, the best setting for top-N (in parentheses) based on validation performance is shown for each trait. For *ALL+CNet* the number in parentheses denotes the average number of posts available per profile.

Summary. We find that our approach is preferable to selecting data at random when predicting personality, and only slightly worse compared to using all available posts of profiles. The advantage is that using only a small subset of posts increases efficiency of the zero-shot setting drastically.

4.6 Post-hoc Analysis with Artificial Data

In the results we reported in the previous section we showed that we obtain a similar zero-shot efficacy while improving efficiency. There are presumably two major difficulties that lead to the slight decrease in efficacy. Firstly, predictions on skewed profile labels are notorously challenging. Secondly, it is not ensured that every profile contains information that allows our agent to learn. To evaluate the capabilities of our RL-Profiler approach, we simplify the task by removing profiles of the majority classes and add posts that ensure to express the personality trait of interest. This is a reasonable analysis step, as the corpus we use is likely skewed by the data acquisition procedures and does not represent the real world distribution of personality traits in the population (Kreuter et al., 2022).

We therefore perform a post-hoc analysis on partially artificial data: to ensure class distribution is fairly balanced, we select at most 15 profiles from training, validation and testing data for each class and enrich all profiles with ≈5% artificial posts we generate using Llama 2. These artificially generated posts aim to clearly indicate either a *low* or *high* level of a specific trait, and we add such highly indicative posts to profiles based on their ground-truth annotations. We present examples of artificially generated posts, the process of generating such, and statistics about this partially artificially corpus in the Appendix A.3.

We repeat our experiments on this data and present the results in Table 5 (we present validation results in the Appendix A.2). In contrast to our previous experiments, we find that our method majorly outperforms the setting using all data (68.5% vs. 97.5% macro F$_1$, +29pp on average over all traits). In comparison to a random selection, we observe an even larger improvement (53.5% to 97.5% macro F$_1$, +44pp). Interestingly, on this data, we find that our approach does not only outperform all zero-shot based methods substantially, but also the supervised-learning based models: compared to *Baseline-R* and *Baseline-B*, we observe an improvement of 28.6pp and 25.9pp macro F$_1$, respectively. These results indicate that our method has large potential to improve needle-in-the-haystack personality profiling tasks via prompting.

5 Conclusion and Future Work

We outlined a novel approach for automatic personality prediction from social media data which enables prompt-based predictions to focus on the most relevant parts of an input. Notably, we do not require labels of relevance, but induce the filter only from the prompt-performance on the profile level. While the results on real data shows no performance improvement overall, it does decrease the required context window of the language model. With an experiment on artificial data, we can show a substantial performance improvement. This shows that our method helps the language model to focus on relevant content, instead of leaving this task to the attention mechanisms in the transformer.

The present results provide several directions for future work: One direction is to replace or adapt individual parts of the proposed system. This includes the evaluation of other policy optimization algorithms, exploring the usage of different large language models, or experiment with different policy parameterization techniques. Further, we suggest to study if the requirement for labeled profiles could be relaxed by relying on confidence estimates of the zero-shot classification.

Another interesting question would be if the relevancy assessement of RL-Profiler is similar to what humans find relevant. This requires a future annotation study of relevancy in personality profiling. Finally, it also remains interesting to explore how our approach performs when applied to predicting other concepts like gender or age.

		Open.		Consc.		Extrav.		Agree.		Neur.	
System	top-N	m-F_1	w-F_1	m-F_1	w-F_1	m-F_1	w-F_1	m-F_1	w-F_1	m-F_1	w-F_1
Baseline-R	all	48.4±0.0	90.7±0.0	68.8±0.0	70.8±0.0	70.8±0.0	72.6±0.0	76.4±0.0	76.9±0.0	79.9±0.0	79.9±0.0
Baseline-B	all	77.3±21.2	93.2±8.3	73.4±6.0	74.2±5.8	75.3±16.7	77.1±18.2	68.3±9.5	69.4±9.0	63.5±6.7	63.5±6.7
ALL+CNet	all	48.4±0.0	90.7±0.0	85.2±4.0	85.7±3.9	80.1±6.9	85.0±4.9	78.1±2.0	78.5±2.0	50.6±3.8	50.6±3.8
RL-Profiler	best	100.±0.0	100.±0.0	98.7±2.1	98.8±1.9	93.5±4.8	94.5±4.2	98.8±1.9	98.9±1.8	96.3±1.1	96.3±1.1
RND+CNet	best	48.4±0.0	90.7±0.0	69.0±3.4	68.7±3.6	45.6±7.3	60.9±5.0	71.3±5.9	71.0±5.9	43.1±8.2	43.1±8.2
PMI+CNet	best	45.2±1.5	84.7±2.8	81.7±2.8	82.2±2.8	69.6±3.5	77.6±2.6	67.4±2.8	66.6±2.9	72.9±5.8	72.9±5.8
PT+CNet	best	46.6±1.5	87.4±2.7	67.6±6.2	66.9±6.6	52.1±6.2	65.5±4.1	81.1±3.9	81.0±4.0	62.9±4.2	62.9±4.2

Table 5: Macro F_1 (m-F_1) and weighted average F_1 (w-F_1) scores on artificially enriched testing data (average of 10 runs with standard deviation). For models with top-N parameter (lower part), the best setting based on macro F_1 score on validation data is chosen for each trait (validation results with all settings for top-N shown in Table 7).

Acknowledgments

This paper is supported by the project INPROMPT (Interactive Prompt Optimization with the Human in the Loop for Natural Language Understanding Model Development and Intervention, funded by the German Research Foundation, KL 2869/13-1, project numer 521755488). We acknowledge the support of the Ministerium für Wissenschaft, Forschung und Kunst Württemberg (MWK, Ministry of Science, Research and the Arts Baden-Württemberg under Az. 33-7533-9-19/54/5) in "Künstliche Intelligenz & Gesellschaft: Reflecting Intelligent Systems for Diversity, Demography and Democracy (IRIS3D)" and the support by the "Interchange Forum for Reflecting on Intelligent Systems" (IRIS) at the University of Stuttgart. Further, we acknowledge the support by the Stuttgart Center for Simulation Science (SimTech).

Ethical Considerations

Personality profiling of social media users is an ethically challenging task. We point out that all data we use stems from an established data set, that has been, to the best of our knowledge, collected following high ethical standards. We do not collect any data ourselves. We condemn any applications of social media mining methods applied to data of users who did not actively consent to using their data for automatic processing. This is particularly the case for subjective and imperfect prediction tasks in which the analysis may be biased in a way that discriminates parts of a society, particularly minority groups.

The methods we develop in this paper contribute to a more efficient use of large language models, therefore contributing to a more sustainable and resource-friendly use of computing infrastructure. Nevertheless, automatic analysis methods need to be applied with care, given the resources that they require.

Limitations

While this study provides valuable insights, several limitations should be acknowledged. First, we treated personality traits as binary variables. However, personality is typically understood as a spectrum rather than a binary value. This simplification potentially limits the applicability of our findings to real-world scenarios where personality assessments are more complex. Further, we did not evaluate our approach using very large-scale language models. Performance of our approach with such models therefore remains untested, and future research could explore how our method scales with larger models to better understand its effectiveness.

Finally, due to resource-constraints, we did not perform exhaustive hyperparameter optimization. This includes to allow different numbers of instances for each profile to be considered. However, we did not optimize them for one model more exhaustively than for another. Therefore, we believe that this aspect would not change the main results of our experiments.

References

Dimosthenis Antypas, Asahi Ushio, Jose Camacho-Collados, Vitor Silva, Leonardo Neves, and Francesco Barbieri. 2022. Twitter topic classification. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3386–3400, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Shlomo Argamon, Sushant Dhawle, Moshe Koppel, and James W. Pennebaker. 2005. Lexical predictors of personality type. In *Proceedings of the 2005 Joint Annual Meeting of the Interface and the Classification Society of North America*, pages 1–16.

Shlomo Argamon, Moshe Koppel, Jonathan Fine, and Anat Rachel Shimoni. 2003. Gender, genre, and writing style in formal written texts. *Text & Talk*, 23(3):321–346.

Michael C. Ashton and Kibeom Lee. 2007. Empirical, theoretical, and practical advantages of the HEXACO model of personality structure. *Personality and Social Psychology Review*, 11(2):150–166. Publisher: SAGE Publications Inc.

Gerlof Bouma. 2009. Normalized (pointwise) mutual information in collocation extraction. *German Society for Computational Linguistics and Language Technology (GSCL)*, 30:31–40.

Lauren Brewer. 2019. General psychology: Required reading. *Deiner Education Fund*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.

Raymond B. Cattell. 1945. The description of personality: Principles and findings in a factor analysis. *The American Journal of Psychology*, 58(1):69–90.

Mara Chinea-Rios, Thomas Müller, Gretel Liz De la Peña Sarracén, Francisco Rangel, and Marc Franco-Salvador. 2022. Zero and few-shot learning for author profiling. In *Natural Language Processing and Information Systems: 27th International Conference on Applications of Natural Language to Information Systems, NLDB 2022, Valencia, Spain, June 15–17, 2022, Proceedings*, page 333–344, Berlin, Heidelberg. Springer-Verlag.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29.

Claude R. Cloninger. 1994. Temperament and personality. *Current Opinion in Neurobiology*, 4(2):266–273.

Michael D. Conover, Bruno Goncalves, Jacob Ratkiewicz, Alessandro Flammini, and Filippo Menczer. 2011. Predicting the political alignment of twitter users. In *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing*, pages 192–199.

Paul T. Costa and Robert R. McCrae. 1992. *Neo personality inventory-revised (NEO PI-R)*. Psychological Assessment Resources Odessa, FL.

Nikolas Coupland. 2007. *Style: Language variation and identity*. Cambridge University Press, Cambridge, UK.

Leyang Cui, Yu Wu, Jian Liu, Sen Yang, and Yue Zhang. 2021. Template-based named entity recognition using BART. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1835–1845, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

John M. Digman. 1990. Personality structure: emergence of the five-factor model. *Annual Review of Psychology*, 41(1):417–440.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. 2022. GPTQ: Accurate post-training compression for generative pretrained transformers. *arXiv preprint arXiv:2210.17323*.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3816–3830, Online. Association for Computational Linguistics.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-augmented generation for large language models: A survey. *Preprint*, arXiv:2312.10997.

José Antonio García-Díaz, Salud M. Jiménez Zafra, María Teresa Martín Valdivia, Francisco García-Sánchez, Luis Alfonso Ureña López, and Rafael Valencia García. 2022. Overview of PoliticEs 2022: Spanish author profiling for political ideology. Sociedad Española para el Procesamiento del Lenguaje Natural.

Jennifer Golbeck, Cristina Robles, and Karen Turner. 2011. Predicting personality with social media. In *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '11, page 253–262, New York, NY, USA. Association for Computing Machinery.

Lewis R. Goldberg. 1981. Language and individual differences: The search for universals in personality lexicons. *Review of Personality and Social Psychology*, 2(1):141–165.

Lewis R. Goldberg. 1990. An alternative "description of personality": The big-five factor structure. *Journal of Personality and Social Psychology*, 59:1216–1229.

Oliver P. John, Eileen M. Donahue, and Robert L. Kentle. 1991. The big five inventory—versions 4a and 54. Technical report, Berkeley, CA: University of California, Berkeley, Institute of Personality and Social Research.

Oliver P. John and Sanjay Srivastava. 1999. *The Big Five Trait taxonomy: History, measurement, and theoretical perspectives*, pages 102–138.

Alam Sher Khan, Hussain Ahmad, Muhammad Zubair Asghar, Furqan Khan Saddozai, Areeba Arif, and Hassan Ali Khalid. 2020. Personality classification from online text using machine learning approach. *International Journal of Advanced Computer Science and Applications*, 11(3):460–476.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Moshe Koppel, Shlomo Argamon, and Anat Rachel Shimoni. 2002. Automatically categorizing written texts by author gender. *Literary and Linguistic Computing*, 17(4):401–412.

Moshe Koppel, Jonathan Schler, and Kfir Zigdon. 2005. Determining an author's native language by mining a text for errors. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, KDD '05, page 624–628, New York, NY, USA. Association for Computing Machinery.

Anne Kreuter, Kai Sassenberg, and Roman Klinger. 2022. Items from psychometric tests as training data for personality profiling models of Twitter users. In *Proceedings of the 12th Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis*, pages 315–323, Dublin, Ireland. Association for Computational Linguistics.

Jesus Lopez-Santillan, Luis Carlos Gonzalez Gurrola, Manuel Montes, and Adrián López-Monroy. 2023. When attention is not enough to unveil a text's author profile: Enhancing a transformer with a wide branch. *Neural Computing and Applications*, 35:1–20.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Ruotian Ma, Xin Zhou, Tao Gui, Yiding Tan, Linyang Li, Qi Zhang, and Xuanjing Huang. 2022. Template-free prompt tuning for few-shot NER. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5721–5732, Seattle, United States. Association for Computational Linguistics.

Robert R. McCrae and Oliver P. John. 1992. An introduction to the five-factor model and its applications. *Journal of personality*, 60(2):175–215.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, and 1 others. 2015. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533.

Huseyin Oz. 2015. Personality traits and ideal l2 self as predictors of academic achievement among prospective english teachers. In *ICERI2015 Proceedings*, 8th International Conference of Education, Research and Innovation, pages 5833–5841. IATED.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, and 2 others. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

James Pennebaker, Matthias Mehl, and Kate Niederhoffer. 2003. Psychological aspects of natural language use: Our words, our selves. *Annual review of psychology*, 54:547–77.

Beatrice Rammstedt and Oliver P. John. 2007. Measuring personality in one minute or less: A 10-item short version of the big five inventory in english and german. *Journal of research in Personality*, 41(1):203–212.

Francisco Rangel, Fabio Celli, Paolo Rosso, Martin Potthast, Benno Stein, and Walter Daelemans. 2015. Overview of the 3rd Author Profiling Task at PAN 2015. In *Working Notes Papers of the CLEF 2015 Evaluation Labs*, volume 1391 of *CEUR Workshop Proceedings*.

Oscar Sainz, Oier Lopez de Lacalle, Gorka Labaka, Ander Barrena, and Eneko Agirre. 2021. Label verbalization and entailment for effective zero and few-shot relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1199–1212, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jonathan Schler, Moshe Koppel, Shlomo Engelson Argamon, and James W. Pennebaker. 2006. Effects of age and gender on blogging. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*.

Hansen A. Schwartz, Johannes C. Eichstaedt, Margaret L. Kern, Lukasz Dziurzynski, Stephanie M. Ramones, Megha Agrawal, Achal Shah, Michal Kosinski, David Stillwell, Martin EP Seligman, and 1 others. 2013. Personality, gender, and age in the language of social media: The open-vocabulary approach. *PloS one*, 8(9):e73791.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Octavia-Maria Sulea and Daniel Dichiu. 2015. Automatic profiling of twitter users based on their tweets: Notebook for pan at clef 2015. In *Working Notes of CLEF 2015-Conference and Labs of the Evaluation forum.*, pages 1–8.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, and 49 others. 2023b. Llama 2: Open foundation and fine-tuned chat models. *Preprint*, arXiv:2307.09288.

Lifu Tu, Caiming Xiong, and Yingbo Zhou. 2022. Prompt-tuning can be much better than fine-tuning on cross-lingual understanding with multilingual language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5478–5485, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Ronald J. Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3-4).

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3914–3923, Hong Kong, China. Association for Computational Linguistics.

A Appendix

A.1 Validation Results

		Open.		Consc.		Extrav.		Agree.		Neur.	
System	top-N	m-F$_1$	w-F$_1$	m-F$_1$	w-F$_1$	m-F$_1$	w-F$_1$	m-F$_1$	w-F$_1$	m-F$_1$	w-F$_1$
RL-Profiler	5	**55.4**±10.9	**92.0**±1.7	**49.5**±5.4	**70.6**±3.9	**49.6**±8.0	**82.5**±2.3	**70.1**±3.2	**84.4**±2.5	46.6±6.0	48.5±5.9
RND+CNet	5	45.9±1.6	88.8±3.0	32.4±2.2	43.5±3.5	45.8±0.7	81.4±1.2	44.4±7.3	53.6±9.6	37.6±5.7	36.0±6.9
PMI+CNet	5	44.4±1.0	86.0±2.0	30.5±3.9	40.6±6.2	**57.2**±11.5	**83.7**±4.7	47.3±4.0	57.8±3.5	29.6±5.1	27.8±6.5
PT+CNet	5	45.4±4.3	85.6±2.9	**42.4**±4.6	**58.5**±6.4	45.5±0.6	80.8±1.0	45.5±2.8	52.8±3.7	29.6±5.4	27.6±6.2
RL-Profiler	10	45.7±1.1	88.4±2.2	42.4±6.1	60.4±6.3	44.9±1.2	79.8±2.1	53.6±6.7	70.7±5.3	46.1±4.4	49.6±4.0
RND+CNet	10	45.5±1.1	88.1±2.2	26.5±4.2	34.0±7.0	44.6±1.5	79.3±2.7	39.6±4.7	47.3±6.6	41.3±9.5	40.4±11.3
PMI+CNet	10	44.8±1.1	86.8±2.1	23.5±3.6	29.1±6.1	50.5±8.9	80.2±3.6	45.2±2.3	52.5±3.1	42.8±4.2	43.9±4.7
PT+CNet	10	43.5±3.8	82.3±4.1	32.9±3.7	44.3±5.9	44.4±1.2	79.0±2.2	37.8±4.1	42.4±5.7	33.5±4.6	34.1±4.7
RL-Profiler	20	48.7±0.6	94.2±1.2	43.2±2.2	59.8±3.0	45.8±0.9	81.4±1.5	53.1±4.8	71.8±2.8	42.6±4.8	49.6±4.4
RND+CNet	20	47.5±0.5	92.0±1.0	29.0±4.0	38.1±6.4	45.4±0.8	80.8±1.4	41.8±7.4	50.9±9.8	42.7±5.2	47.9±4.9
PMI+CNet	20	46.5±0.9	90.1±1.7	25.9±1.5	33.1±2.5	43.0±1.2	76.5±2.1	39.7±4.2	46.7±5.2	**45.1**±2.3	**50.2**±2.6
PT+CNet	20	46.3±1.0	89.5±1.9	30.2±4.1	40.0±6.7	44.8±0.8	79.6±1.5	45.0±4.4	54.3±5.4	41.0±3.2	47.1±3.3
RL-Profiler	30	48.9±0.4	94.7±0.8	40.2±4.4	56.0±5.2	47.1±0.0	83.7±0.0	51.5±2.9	68.0±2.8	**48.8**±5.1	**56.6**±4.2
RND+CNet	30	47.8±1.2	92.5±2.3	30.0±2.9	39.7±4.6	46.1±1.1	81.9±1.9	43.5±6.1	53.8±7.9	**44.1**±4.6	**51.7**±3.7
PMI+CNet	30	46.2±0.8	89.4±1.5	30.0±2.5	39.7±4.0	44.8±1.3	79.6±2.4	47.5±5.2	58.0±6.0	38.2±3.0	47.4±3.1
PT+CNet	30	**47.7**±0.7	**92.3**±1.4	33.5±2.8	45.3±4.5	45.1±0.5	80.2±0.8	**47.5**±4.6	**57.2**±4.5	40.9±6.8	48.6±5.5
RL-Profiler	50	48.7±0.6	94.2±1.2	44.5±2.9	61.6±3.8	47.1±0.0	83.7±0.0	50.8±4.3	68.2±2.9	45.6±5.9	58.5±4.2
RND+CNet	50	**48.4**±0.8	**93.7**±1.5	**36.9**±4.1	**50.5**±6.1	**47.1**±0.0	**83.7**±0.0	**47.4**±5.2	**62.3**±7.0	42.0±6.0	56.9±4.6
PMI+CNet	50	**48.3**±0.4	**93.5**±0.8	**34.1**±1.9	**46.3**±3.0	47.1±0.0	83.7±0.0	**51.6**±3.3	**65.9**±3.0	39.0±4.8	54.6±4.0
PT+CNet	50	47.7±0.6	92.4±1.2	37.5±3.8	51.3±5.7	**46.7**±0.5	**83.1**±0.9	44.8±5.3	59.0±5.7	**44.9**±6.7	**58.9**±5.5
ALL+CNet	all	49.2±0.0	95.2±0.0	41.6±2.8	65.6±2.2	47.1±0.0	83.7±0.0	54.8±3.9	76.5±2.7	42.7±5.1	60.1±3.5

Table 6: Macro F$_1$ (m-F$_1$) and weighted average F$_1$ (w-F$_1$) scores for selection-based models with different settings for the top-N hyperparameter on validation data (averages of 10 runs with standard derivation). The best performing setting for top-N (w.r.t. the highest m-F$_1$) for each model and personality trait (highlighted in **bold**) is selected for evaluation on testing data.

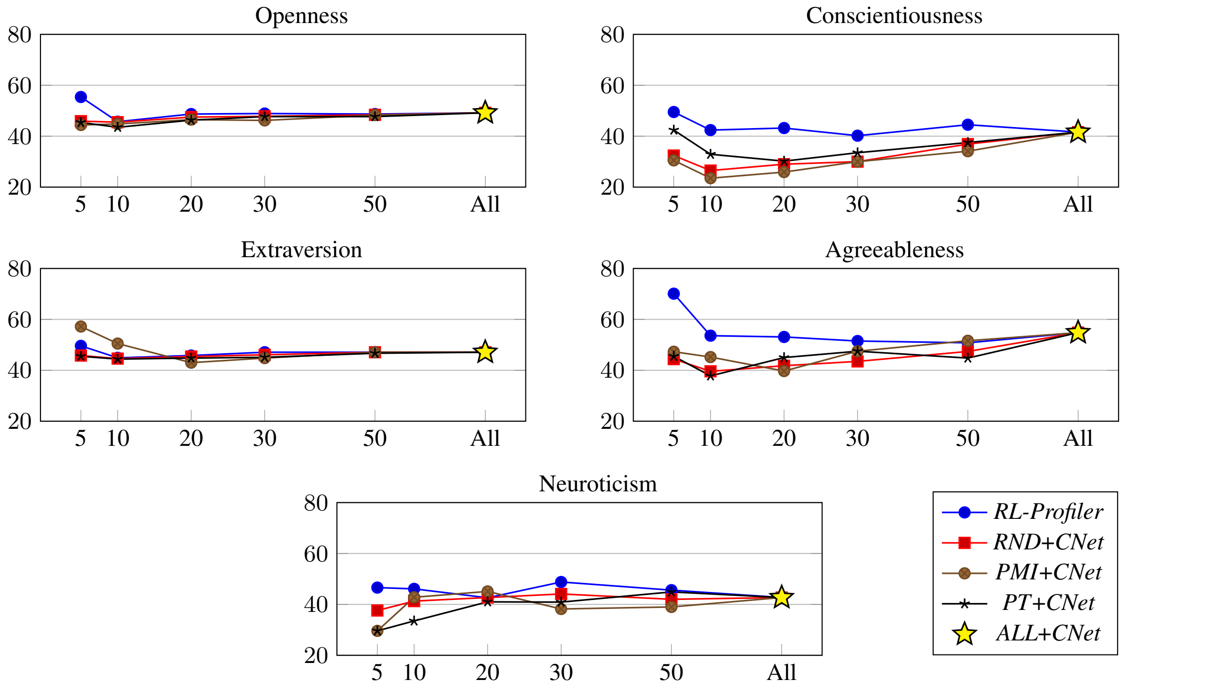

Figure 3: Visual representation of macro F$_1$ scores for selection-based models with different settings for top-N on validation data. The x-axis (not true to scale) shows settings for top-N, i.e., $N \in \{5, 10, 20, 30, 50\}$ (linearly interpolated), while the y-axis shows the corresponding macro F$_1$ scores. If N exceeds the number of available posts in profiles, all models converge to the *ALL+CNet* system since all systems select all available posts.

A.2 Validation Results on Artificially Enriched Data

		Open.		Consc.		Extrav.		Agree.		Neur.	
System	top-N	m-F$_1$	w-F$_1$	m-F$_1$	w-F$_1$	m-F$_1$	w-F$_1$	m-F$_1$	w-F$_1$	m-F$_1$	w-F$_1$
RL-Profiler	5	**100.**±0.0	**100.**±0.0	**100.**±0.0	**100.**±0.0	**98.1**±2.4	**98.3**±2.2	**98.8**±1.9	**98.9**±1.8	**97.7**±2.5	**97.9**±2.3
RND+CNet	5	46.8±1.6	87.8±2.9	37.7±5.6	45.0±8.0	**50.8**±10.6	**76.5**±4.5	59.2±7.1	58.4±7.4	44.3±13.3	42.0±15.2
PMI+CNet	5	78.1±21.0	93.9±5.3	**48.9**±1.4	**60.6**±1.8	56.0±13.6	80.0±5.5	49.3±3.6	48.4±3.6	40.4±5.1	37.7±5.3
PT+CNet	5	70.2±24.6	93.7±5.0	32.1±6.8	40.6±7.5	60.0±12.3	79.3±5.2	67.5±4.1	67.8±4.1	53.7±5.6	53.2±6.1
RL-Profiler	10	100.±0.0	100.±0.0	78.2±6.8	88.4±4.5	82.1±9.1	89.7±5.6	90.7±4.5	90.8±4.5	93.2±4.1	93.8±3.7
RND+CNet	10	51.1±9.1	87.9±2.1	30.1±4.9	33.7±7.4	49.4±11.7	74.1±6.5	58.4±8.3	57.3±8.9	50.0±7.8	49.6±8.9
PMI+CNet	10	**87.8**±13.4	**96.0**±4.5	46.2±4.9	56.9±6.6	43.6±0.6	72.6±1.0	56.8±6.1	55.5±6.7	42.4±3.6	39.2±4.0
PT+CNet	10	77.3±23.0	94.7±5.1	29.1±3.6	32.2±5.6	**81.0**±4.4	**89.4**±2.8	**77.6**±7.4	**77.8**±7.4	60.7±4.8	60.2±5.1
RL-Profiler	20	94.8±16.3	99.1±2.9	63.9±5.3	77.2±5.1	81.5±6.1	89.6±3.7	82.3±2.0	82.2±2.1	78.2±5.3	81.0±4.6
RND+CNet	20	48.0±0.7	90.1±1.4	37.2±4.7	44.4±6.9	50.3±11.0	75.6±5.1	65.0±7.1	64.3±7.4	57.9±6.5	60.7±6.5
PMI+CNet	20	48.2±0.5	90.4±1.0	40.2±4.4	48.6±6.3	66.5±2.8	82.8±1.9	66.0±4.1	65.1±4.3	53.8±4.2	54.3±4.0
PT+CNet	20	**81.4**±16.9	**94.3**±5.1	37.2±5.5	44.3±8.3	75.7±12.4	86.9±6.7	74.8±3.8	74.5±4.0	**69.9**±7.8	**72.8**±6.9
RL-Profiler	30	84.3±25.2	96.9±5.1	54.3±6.1	67.0±7.6	66.7±16.8	84.1±7.0	76.8±3.2	76.6±3.3	77.0±6.0	80.0±5.1
RND+CNet	30	47.9±0.8	89.8±1.6	38.6±7.3	46.1±10.5	50.5±10.9	75.9±4.8	63.8±3.7	63.2±3.8	56.2±7.0	61.0±6.7
PMI+CNet	30	48.4±0.0	90.7±0.0	43.9±3.8	53.9±5.1	**80.1**±8.7	**90.2**±4.0	**68.5**±4.9	**67.8**±5.2	**57.3**±4.3	**59.6**±4.1
PT+CNet	30	50.7±10.9	89.2±2.5	34.8±4.4	40.8±6.6	68.7±5.1	84.0±2.7	74.7±4.9	74.3±5.2	60.4±7.7	65.5±6.6
RL-Profiler	50	63.9±24.9	93.5±4.5	53.0±4.2	65.5±5.2	50.7±11.1	77.9±4.5	74.9±3.3	74.6±3.4	61.5±3.5	67.9±2.9
RND+CNet	50	**53.5**±16.3	**91.6**±2.9	**48.0**±4.0	**59.3**±5.3	48.1±8.3	76.8±3.4	**71.7**±4.9	**71.5**±5.0	**58.0**±7.1	**65.1**±5.8
PMI+CNet	50	48.4±0.0	90.7±0.0	47.1±4.3	58.1±5.7	45.5±0.0	75.8±0.0	67.1±4.9	66.8±5.0	53.4±6.6	60.7±4.8
PT+CNet	50	58.7±21.8	92.6±3.9	**42.5**±4.9	**51.8**±6.9	56.0±13.6	80.0±5.5	71.7±3.3	71.3±3.4	50.8±6.4	59.5±5.2
ALL+CNet	all	89.7±21.8	98.2±3.9	45.8±3.6	63.6±3.6	79.1±14.2	89.9±6.1	77.2±4.0	77.7±3.9	60.7±10.9	67.6±8.4

Table 7: Macro F$_1$ (m-F$_1$) and weighted average F$_1$ (w-F$_1$) scores for models with different settings for the top-N hyperparameter on artificially enriched validation data (averages of 10 runs with standard derivation). The best performing setting for top-N (w.r.t. the highest m-F$_1$) for each model and personality trait (highlighted in **bold**) is selected for evaluation on testing data.

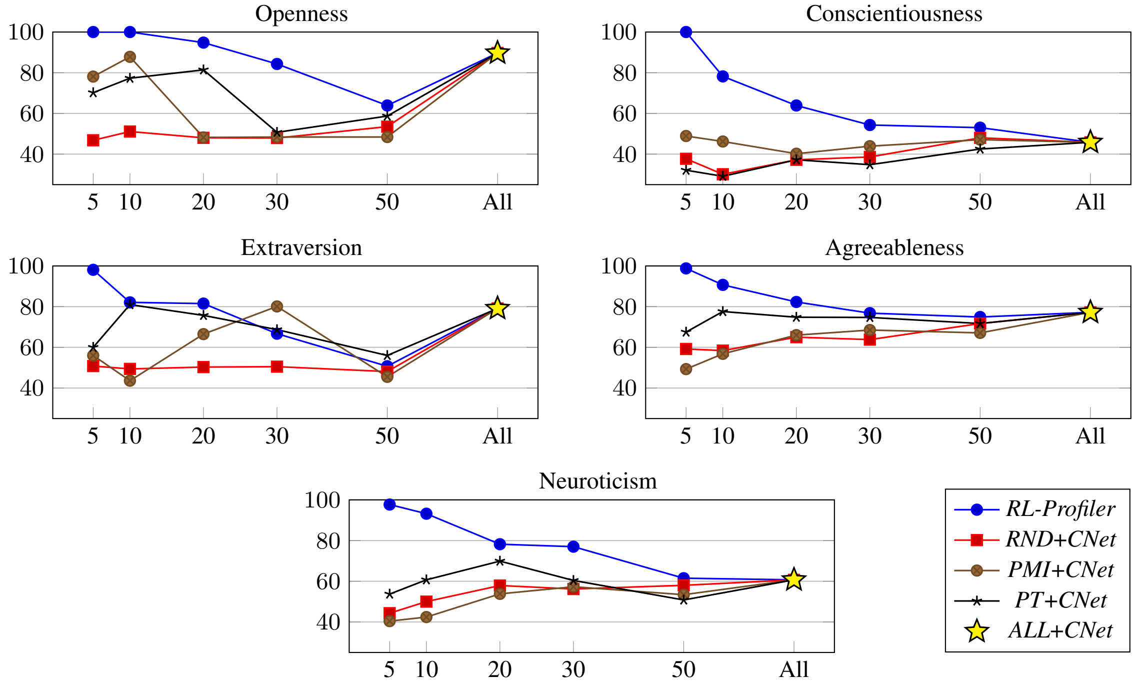

Figure 4: Visual representation of macro F$_1$ scores for selection-based models with different settings for top-N on artificially enriched validation data. The x-axis (not true to scale) shows settings for top-N, i.e., $N \in \{5, 10, 20, 30, 50\}$ (linearly interpolated), while the y-axis shows the corresponding macro F$_1$ scores. If N exceeds the number of available posts in profiles, all models converge to the *ALL+CNet* system since all systems select all available posts.

```
Recall the personality trait extraversion.
A person with a high level of extraversion may see themselves as someone who is talkative, or {...}
Generate ten tweets that are likely written by a person with a high level of extraversion.
+Do not use emojis or hashtags. Try to include the topic {topic}.
```

```
Recall the personality trait extraversion.
A person with a low level of extraversion may see themselves as someone who is reserved, or {...}
Generate ten tweets that are likely written by a person with a low level of extraversion.
+Do not use emojis or hashtags. Try to include the topic {topic}.
```

Figure 5: Prompt templates for generating artificial posts indicating a *high* and *low* level of *extraversion*.

	Training		Validation		Testing	
Class	High	Low	High	Low	High	Low
Openness	15	1	15	1	15	1
Conscientiousness	15	3	15	2	15	10
Extraversion	15	12	15	3	15	6
Agreeableness	15	15	15	4	15	11
Neuroticism	15	15	15	8	15	15

Table 8: Corpora statistics of the splits derived from the PAN-AP-2015 (Rangel et al., 2015) corpus for post-hoc experiments on partially artificially data (in numbers of profiles).

A.3 Artificial Post Generation and Dataset Enrichment

To generate artificial posts indicating either a *low* or *high* level of a certain personality trait we use Llama 2 13B-Chat, and repeatedly prompt the model to generate 10 posts. We present the prompt templates we use for generating artificial posts for the *extraversion* trait in Figure 5. Here, the task of generating posts is verbalized by the phrase "Generate ten tweets that are likely written by a person with a high level of extraversion". Similarly to the prompts used in CNet for prediction levels of a trait, we include BFI-44 items to enrich context. The prompts for the other Big Five traits follow a similar structure.

To further encourage diversity in the generating posts, since different profiles should be enriched with different posts (because it would be trivial for a model to find these posts if they are always the same), we task the LLM to include a topic in the generated posts. For this, we compile a list of 12 topics we derive from the work by Antypas et al. (2022) covering many discussion points in social media:

- News
- Social Concern
- Sports
- Music
- Celebrity & Pop Culture
- Film, TV & Video
- Diaries & Daily Life
- Arts & Culture
- Science & Technology
- Fitness & Health
- Family
- Relationships

We present examples of artificially generated posts for different topics and personality traits in Table 9.

We use the artificially generated posts in our post-hoc analysis. Here, we draw subsets from the datasets we derived from the PAN-AP-2015 corpus. For each trait, split and class we randomly select 15 profiles. Note that in some partitions, there are less than 15 profiles and this process therefore changes class distribution in each sub-corpus. Table 8 shows statistics of the datasets we obtain by this. Afterward, we enrich each profile in these dataset splits with five artificially generated posts we randomly choose from the pool of generated posts (ensuring we use each artificial post only once) based on their ground-truth annotation, e.g., for profiles annotated with a *low* level of *extraversion*, we add generated posts that aim to indicate a *low* level of *extraversion*.

		Topic	Generated Post
Openness	High	News	I'm not just a thinker, I'm a dreamer. And I believe that dreams can change the world.
		Social Concern	Why are we still tolerating systemic injustices? It's time for change.
		Music	I love how music can transport me to another world, even if just for a few minutes
		Science & Techn.	Building a DIY robot using an Arduino board and exploring its capabilities.
		Family	Family is a source of comfort and strength, especially during challenging times.
	Low	Social Concern	I don't care about the homeless crisis in our city, it's not my problem.
		Sports	Just finished a 9-to-5 shift at the office and now I'm ready to watch some football!
		Music	Classical music is the only real music. Everything else is just noise.
		Arts & Culture	I don't read books. I don't see the point of wasting time on made-up stories.
		Science & Techn.	I don't see the point of space exploration. We have enough problems on Earth
Conscientiousness	High	Sports	I strive for efficiency in everything I do, whether it's on the field or in the weight room.
		Music	I've been practicing my guitar for hours every day to perfect my technique.
		Film, TV & Video	I'm so impressed by the cinematography in the latest blockbuster. It's like a work of art.
		Diaries & Daily Life	I find solace in my daily routine, it brings me a sense of stability and predictability.
		Fitness & Health	I track my progress and adjust my plan as needed to ensure I'm reaching my fitness goals.
	Low	News	Can't find my homework... or my textbook... or my notes. Anyone have a photocopy?
		Sports	I think I might have accidentally signed up for a relay instead of a solo race
		Film, TV & Video	I'm so addicted to my favorite TV show that I can't stop thinking about it. I need help!
		Diaries & Daily Life	I just spent $100 on a new outfit instead of paying my rent. Oopsie.
		Relationships	I know I said I would call my partner back yesterday, but uh... I forgot?
Extraversion	High	News	I'm so excited to share the latest scoop with all my followers!
		Music	Just discovered a new artist and I can't stop listening to their music!
		Diaries & Daily Life	I just tried the craziest new food trend and it was so good! I can't wait to try more
		Fitness & Health	Feeling so strong and confident after a killer leg day at the gym.
		Relationships	I'm not scared of rejection. I'll put myself out there and see what happens!
	Low	Sports	I prefer to focus on my own improvement rather than comparing myself to others.
		Music	My favorite way to relax is to listen to calming music and meditate.
		Science & Techn.	My mind is always racing with ideas, but I struggle to express them out loud.
		Fitness & Health	I'm not a fan of loud, crowded gyms, I prefer to work out at home in my own space.
		Family	I love my family, but sometimes I just need a little alone time to recharge.
Agreeableness	High	Social Concern	I'm a team player, and I think collaboration is the key to success.
		Sports	I can't believe we won! It's all thanks to our teamwork and determination.
		Diaries & Daily Life	I think it's important to be open-minded and accepting of others.
		Fitness & Health	I'm so grateful for my fitness community - they inspire me to be my best self every day.
		Family	I love being a part of our family's traditions and making new memories together.
	Low	News	I can't believe the media is still covering that story, it's such a non-issue.
		Social Concern	I don't have time for weak people, they need to toughen up.
		Sports	Why should I have to follow the rules? The other team is always cheating anyway.
		Science & Techn.	Technology is ruining our society. We need to go back to simpler times.
		Family	My family is always trying to tell me what to do. Newsflash: I don't need their advice.
Neuroticism	High	News	I can't believe what I just heard on the news. It's like, what is even happening?!
		Sports	I'm so tense before every game. I can't relax, no matter how hard I try.
		Diaries & Daily Life	I've been doing yoga for months and still can't touch my toes.
		Arts & Culture	Why can't I just enjoy a simple painting without overanalyzing every brushstroke?
		Family	My family is always causing drama. I just want peace and quiet!
	Low	Social Concern	I'm not perfect, but I strive to be a good listener and a supportive friend.
		Celebr. & Pop Cult.	I don't stress about fashion or beauty trends. Comfort and simplicity are key for me!
		Diaries & Daily Life	I'm proud of my ability to remain emotionally stable, even in difficult situations.
		Arts & Culture	The beauty of nature is a never-ending source of inspiration for my art.
		Family	Family vacations are the best kind of stress-free fun.

Table 9: Examples of posts generated using Llama 2 13B-Chat that aim to indicate either a *low* or *high* level of one of the Big Five traits.