Personalized Large Language Models

Stanisław Woźniak, Bartłomiej Koptyra, Arkadiusz Janz,
Przemysław Kazienko and Jan Kocoń
Department of Artificial Intelligence, Wroclaw Tech, Poland
{stanislaw.wozniak, bartlomiej.koptyra, arkadiusz.janz,
przemyslaw.kazienko, jan.kocon}@pwr.edu.pl

Abstract—**Large language models (LLMs) have significantly advanced Natural Language Processing (NLP) tasks in recent years. However, their universal nature poses limitations in scenarios requiring personalized responses, such as recommendation systems and chatbots. This paper investigates methods to personalize LLMs, comparing fine-tuning and zero-shot reasoning approaches on subjective tasks. Results demonstrate that personalized fine-tuning improves model reasoning compared to non-personalized models. Experiments on datasets for emotion recognition and hate speech detection show consistent performance gains with personalized methods across different LLM architectures. These findings underscore the importance of personalization for enhancing LLM capabilities in subjective text perception tasks.**

Index Terms—**NLP, LLM, Personalization**

I. INTRODUCTION

In recent years, large language models (LLMs) have revolutionized Natural Language Processing (NLP) tasks in a variety of areas, demonstrating remarkable capabilities in text generation, sentiment analysis, machine translation, and more. These models are based on a transformer architecture [1] with a large number of parameters. Training such large models requires massive amounts of text data, enabling them to capture complex language patterns and generate consistent and contextually relevant text.

However, while LLMs have impressive generation capabilities, their universal nature is a limitation in scenarios where personalized responses are desired or required. Then, personalization becomes critical in applications such as recommendation systems, chatbots, and personalized content generation, where understanding and tailoring to individual subjective preferences and profiles is critical to user satisfaction and engagement.

Since the language models are zero-shot reasoners [2], one can solve downstream tasks with prompt-based inference. In this way, we can obtain personalized answers by incorporating user-specific information into the instructions, i.e., in-context learning [3]. However, this approach does not update the model weights, so such personalization is impermanent, limited by the context length, and insufficient for specific downstream tasks. Another method we consider is to fine-tune the model in a personalized way. One of our goals is to investigate whether there is a difference between fine-tuning and zero-shot reasoning of LLMs on the subjective tasks.

Our contributions in this paper include: (1) a novel examination of fine-tuning versus zero-shot and few-shot reasoning in LLMs for personalizing subjective text perception, highlighting the superior performance of personalized fine-tuning; (2) a comprehensive evaluation across diverse datasets for emotion recognition and hate speech detection, demonstrating the significant advantages of personalization; (3) the proposal of new methods to enhance the LLM's responsiveness to individual user contexts, advancing subjective text analysis capabilities; (4) empirical validation of our approaches across various LLM architectures, underscoring their efficacy and adaptability; and (5) the release of research repository[1], including code and datasets, to support reproducibility and further advancements in LLM personalization.

Our methodology is based on personalization through the use of basic user-specific context, which consists of user IDs. We utilized multiple fine-tuning approaches and few-shot in-context learning techniques to personalize LLMs for two distinct subjective tasks. Furthermore, we fine-tuned the models in both the classification and the text generation tasks. The results obtained in this work demonstrate that personalization methods based on simplified user-specific information, such as user IDs, have significant potential to enhance LLM performance by up to 165%.

II. RELATED WORK

AI has been increasingly applied to subjective tasks such as sentiment recognition, hate speech detection, and emotion recognition, leveraging techniques like deep learning and natural language processing. Models like transformers [1], including BERT and GPT [4], [5], have shown strong performance by capturing contextual information in text. However, these tasks remain challenging due to the ambiguity and variability in human language, often requiring large, well-labeled datasets to improve accuracy [6], [7]. Bias in training data and model fairness are also critical concerns, as they can affect the system's performance across different demographic groups. Despite these challenges, AI continues to advance in handling these nuanced tasks, showing promise in real-world applications.

Recent research highlights an interest in personalizing language models, emphasizing their significance across conversational interfaces, recommendation systems, and managing sensitive content [8]–[13]. Studies like [14]–[29] underscore the importance of tailoring NLP models to individual beliefs

---

[1]https://github.com/Rikain/llm-finetuning

and preferences to enhance the handling of offensive content and controversial topics. Models that incorporate personal perspectives, as demonstrated in [17] and [30], offered superior predictions by acknowledging individual emotional responses. Kazienko et al. [31] extend this approach by developing deep learning models that account for individual differences, significantly outperforming traditional models in subjective tasks. Moreover, a study in [32] evaluates the performance of ChatGPT and GPT-4 in generating personalized responses, revealing that such customization improves predictive performance.

The bulk of this research has focused on adapting conventional neural network architectures, like LSTM and transformers (BERT, RoBERTa), for personalization in NLP, demonstrating the benefits of aligning models with user-specific characteristics, especially for sensitive or subjective content. However, recent exploration into large language models (LLMs) like ChatGPT and GPT-4, as noted in [32], showcases their potential in few-shot scenarios without task-specific training, highlighting the advanced capabilities of LLMs to cater to individual user requests effectively.

Fine-tuning allows the models to adapt to specific downstream tasks, potentially leading to better performance. On the other hand, LLMs are sophisticated zero-shot reasoners [2]. One can use their abilities to solve downstream tasks with in-context-learning [33] and extensive prompt-based inference [34]. Fine-tuning can be computationally expensive and time-consuming, especially for large language models. Fine-tuning a language model on task-specific data can improve its performance on the task, but it may come at the cost of reduced performance on other tasks. This is due to the risk of catastrophic forgetting [35], where the model may forget some of the knowledge learned during pre-training and alignment processes [36]–[38]. Techniques such as multi-task learning or balancing pre-training and task-specific data might be beneficial for retaining the performance of LLMs in multiple downstream tasks.

To the best of our knowledge, LLM fine-tuning for subjective tasks via user ID inputs, such as personalized emotion recognition or personalized hate speech detection, has not been extensively evaluated, and further research is needed in this area.

III. CONCEPT OF PERSONALIZING LLMs FOR SUBJECTIVE TEXT PERCEPTION

In human communication, interpretation and perception of texts depends not only on the textual content itself. For that reason, especially for subjective tasks like emotion recognition, hate speech, humor, or even sentiment analysis, LLMs should respect individual human preferences and beliefs, making their responses more personalized. Then, the models should be provided with information about the user either at the learning / fine-tuning stage (persistent) or at generation (impersistent). The concept of personalized LLM approaches (and non-personalized baselines) are presented in Fig. 1.

A. Problem Definition

The primary challenge in personalizing LLMs for subjective text perception lies in the model's ability to incorporate individual user preferences, biases, and contexts into its processing mechanism. Given a user u and a text input T, the goal is to generate a response $\hat{Y}_u$ that aligns with u's subjective perception of T. The prediction is defined as:

$$\hat{Y}_u = f(T, C_u)$$

where f is the function modeled by the LLM, and C_u represents the contextual user u information, which includes user preferences, historical interactions, and any other relevant user-specific data.

B. Personalized Text Classification

We propose a personalization approach that modifies the LLM's behavior based on C_u. This can be achieved by adjusting the model's parameters θ or by manipulating the input space to include personalized prompts. The objective function for personalization can be expressed as:

$$\min_{\theta} \mathcal{L}(\theta; \hat{Y}_u, Y_u, T, C_u)$$

where $\mathcal{L}$ is a loss function measuring the discrepancy between the generated response and a set of responses Y_u deemed appropriate by user u. The personalization can be implemented through fine-tuning, where θ is adjusted, or through in-context learning, where C_u is appended to T to guide the model's predictions.

IV. NON-PERSONALIZED BASELINES FOR SUBJECTIVE TEXT CLASSIFICATION

In evaluating the impact of personalization on LLMs, it is essential to establish non-personalized baselines. These baselines represent the model's performance without any adaptation to individual user contexts or preferences. We consider three primary non-personalized approaches: querying an original instruction-tuned model, classification with a new embedding head layer, and generative fine-tuning. Each method offers a different perspective on how LLMs handle subjective text classification in a non-personalized setting.

A. Querying Instruction-tuned Language Models (Q)

This approach involves utilizing a pre-trained LLM that has been instruction-tuned but not further adapted to any specific user data. Given a text input T, the model generates a response $\hat{Y}$ based on the instructions embedded during training:

$$\hat{Y} = f_{\theta}(T)$$

where f_{θ} represents the pre-trained LLM parameterized by θ, and T is the input text. This method evaluates the model's ability to follow instructions and generate appropriate classifications without any additional context or fine-tuning.
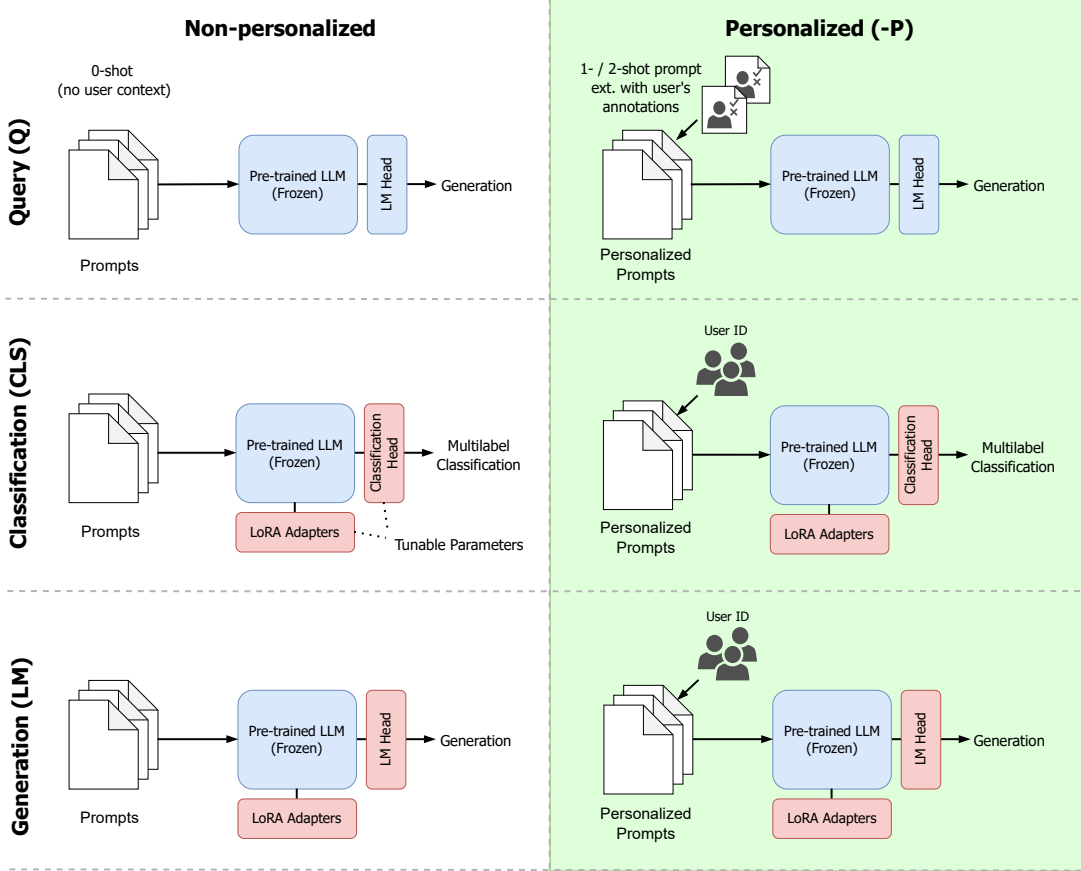
Fig. 1. Non-personalized vs. personalized setups.

B. Classification Head and Model Fine-tuning (CLS)

In the classification approach, a new embedding head layer is introduced to the LLM for the specific task of text classification. The model is then fine-tuned. The objective function for fine-tuning can be defined as:

$$\min_{\theta'} \mathcal{L}_{CLS}(\theta'; \hat{Y}, Y_u, T)$$

where $\mathcal{L}_{CLS}$ is the classification loss function, θ' are the parameters of the fine-tuned model including the new classification head, T is the input text, and Y_u represents user labels. This setup aims to adapt the LLM to perform the task better, still without considering any user-specific personalization.

C. Generative Fine-tuning via Language Modeling (LM)

Generative fine-tuning treats text classification as a text generation problem. The model is fine-tuned to generate a textual label as output, given a descriptive prompt and the input text:

$$\min_{\theta''} \mathcal{L}_{LM}(\theta''; \hat{L}, L_u, T)$$

where $\mathcal{L}_{LM}$ is a loss function suitable for text generation tasks (e.g., cross-entropy loss summed over all positions in the sequence.), θ'' are the parameters of the fine-tuned generative

model, T is the input text, and L_u is the textual label corresponding to T. Unlike the classification approach, which directly predicts labels, this method generates labels as part of a continuous text output.

V. METHODS OF LLM PERSONALIZATION

Personalizing Large Language Models (LLMs) aims to tailor the model's responses to align with individual user preferences, history, and contextual nuances. This section outlines a formal description for various personalization techniques, including few-shot personalization, personalized classification, and personalized language modeling. These methods leverage user-specific data to enhance the model's relevance and accuracy in subjective text perception tasks.

A. Few-shot Personalization (Q-NS)

Few-shot personalization leverages a small number of examples to guide the model towards user-specific interpretations or responses. This technique involves modifying the input prompt to include N examples $(E_1, E_2, \ldots, E_N) \in E_u$ that reflect the user's texts and his perspective or preferences of these texts,

which follows a typical In-Context-Learning setting. The input T with user context C_u are used to generate a response $\hat{Y_u}$:

$$C_u = (E_1, E_2, \ldots, E_N)$$

$$\hat{Y_u} = f_\theta(T, C_u)$$

where f_θ is the pre-trained LLM parameterized by θ, T is the original input text, and examples $E_i = (T_i', Y_i')$, $T_i' \neq T$ are the inputs that illustrate user-specific preferences Y_i' for user-annotated texts T_i'. This method aims to prime the model with a context that mirrors the user's viewpoint, thereby personalizing its output.

B. Personalized classification (CLS-P)

Personalized classification adapts the LLM to specific users by incorporating user identifiers directly into the model's training process. This approach fine-tunes the model's parameters θ to minimize the loss between the predicted labels and true labels, taking into account user-specific data. The objective function for personalized classification:

$$\min_{\theta'} \mathcal{L}_{CLS-P}(\theta'; \hat{Y_u}, Y_u, T, C_u)$$

where $\mathcal{L}_{CLS-P}$ is the personalized classification loss function, θ' are the parameters of the personalized model, T is the input text, Y_u are the user labels, and C_u represents the contextual information (user ID) specific to the user u. This method produces more accurate and personalized label predictions.

C. Personalized Langue Modeling (LM-P)

In personalized language modeling, the goal is to fine-tune the LLM so that its generated text is tailored to the individual user's language use, preferences, or style. In our case, these are user labels in the text version. Like the classification approach, this method fine-tunes the model but focuses on generating personalized text outputs rather than predicting labels. The objective can be defined as:

$$\min_{\theta''} \mathcal{L}_{LM-P}(\theta''; \hat{L_u}, L_u, T, C_u)$$

where $\mathcal{L}_{LM-P}$ is the loss function for personalized language modeling, θ'' are the parameters of the fine-tuned generative model, T is the input text, L_u is the desired textual output for user u, and C_u contains the user-specific contextual information (user ID). This allows the model to generate relevant responses aligned with the user's preferences.

VI. Experiments

We undertook a comprehensive set of experiments to rigorously evaluate our hypotheses, primarily focusing on multi-label classification tasks using several large language models. Our experimental design included all of the approaches described in section V. However, in this section, we provide a detailed description of the experimental scenarios, explaining the models and datasets used to investigate the effectiveness of our methods. Additionally, we have used all the models and datasets described below for scientific purposes, which is in accordance with their licenses.

A. Datasets

In our experiments, we used two English-language datasets: GoEmotions [39] and Unhealthy Conversations [40]. Both corpora encompass annotations contributed by numerous individual annotators, each reflecting their subjective perspectives and opinions in a multi-label classification task. Datasets were partitioned based on textual content, resulting in distinct training, validation, and test sets delineated by individual texts. Empty annotations were omitted from consideration. Each partition was refined to exclude outlier annotators, defined as those with annotation frequencies significantly deviating from the dataset's norm. Specifically, annotators contributing less than 5% of annotations compared to the highest individual annotator were eliminated. Subsequently, the dataset was further refined to incorporate annotations from all annotators across each partition, ensuring comprehensive coverage of annotated data within each subset.

GoEmotions: The GoEmotion dataset under the Apache-2.0 license comprises nearly 58k Reddit comments annotated by 82 distinct annotators, resulting in a total of over 211k individual annotations. Each annotator labeled the data using 28 unique emotional categories, including admiration, amusement, anger, annoyance, approval, caring, confusion, curiosity, desire, disappointment, disapproval, disgust, embarrassment, excitement, fear, gratitude, grief, joy, love, nervousness, optimism, pride, realization, relief, remorse, sadness, surprise, and neutral sentiment. Following the application of our data-cleaning procedure, the dataset was refined to include annotations from 72 annotators. This refinement yielded a training split containing over 146k samples, with validation and test splits each comprising over 18k samples.

Unhealthy Conversations: The second dataset, termed "Unhealthy Conversation", encompasses approximately 230k annotations contributed by 588 annotators across more than 44k distinct online news comments. Each comment was categorized as either healthy or unhealthy, with additional annotations denoting seven attributes: antagonistic, hostile, dismissive, condescending, sarcastic, generalization, or unfair generalization. Following preprocessing procedures, the dataset was refined to comprise a training set consisting of roughly 168k samples, along with validation and test sets, each containing over 20k samples. In the refined iteration of the dataset, the number of annotators was reduced to 427. This dataset is published under the CC BY-NC-SA 4.0 license

B. Models

This study investigates the performance of small and moderate-sized language models developed by different research groups. The following models were selected for the experimental part.

Mistral: Mistral 7B [41] is a language model under the Apache-2.0 license that outperforms previous models such as Llama across diverse benchmarks and approaches the coding performance of Code-Llama 7B [42]. With the use of grouped-query attention (GQA) and sliding window attention (SWA),

Mistral 7B enhances performance and efficiency in comparison to even larger models such as Llama-2-13B [43].

Flan-T5: Flan-T5 [44] is an LLM based on the T5 text-to-text transformer model, which is a common encoder-decoder language modeling architecture. Flan-T5 is specifically designed for natural language understanding and generation tasks, showing the impact of parameter scaling and instruction-based fine-tuning. The model demonstrates that instruction fine-tuning scales effectively with both the number of tasks and the size of the model. In our experimental part, we used Flan-T5-XL model which has 3B parameters under Apache-2.0 license.[2]

Phi-2: The Phi-2 model [45] is a 2.7 billion-parameter small language model (SLM) published under MIT license that challenges the notion that bigger models are always better. It achieves reliable performance in reasoning and language understanding, outperforming models of greater size. This is attributed to different model scaling and the use of high-quality (textbook-quality) training data. Despite its smaller size, Phi-2 demonstrates great performance on specific benchmarks without the need for alignment through Reinforcement Learning from Human Feedback (RLHF) [46].[3]

StableLM: StableLM models include 3B and 7B parameter decoder-only language models, refined through fine-tuning on diverse chat and instruction-following datasets.[4] Utilizing the NeoX transformer architecture [47], these auto-regressive models are designed for chat-based applications.[5] In experiments, we utilised 3 billion-parameter model available at HuggingFace[6], where it was added under the license CC-BY-NC-SA 4.0.

ChatGPT: ChatGPT is a group of models developed by OpenAI under OpenAI API license. Two of the models used in this research are GPT-3.5 and GPT-4. Currently, GPT-4 is one of the best models for multiple tasks, such as zero-shot reasoning. [7]

C. Experimental setting

Our experimental setup takes into account the perspective of both datasets (Sec. VI-A) and models (Sec. VI-B). We designed the settings to provide an understanding of the performance and effectiveness of personalized fine-tuning and in-context learning methods across different methods and subjective tasks.

Most of the LLMs selected for our study have a transformer architecture with a decoder-only configuration. Notably, these models required less computational resources for fine-tuning than models with an encoder-decoder architecture, such as Flan-T5. The experiments on language modeling methods with Flan-T5 model were omitted. In the language modeling setting (LM), we compared solely the decoder-only models. However,

---

[2]https://huggingface.co/google/flan-t5-xl

[3]https://huggingface.co/microsoft/phi-2

[4]https://huggingface.co/stabilityai/stablelm-tuned-alpha-3b

[5]https://huggingface.co/EleutherAI/gpt-neox-20b

[6]https://huggingface.co

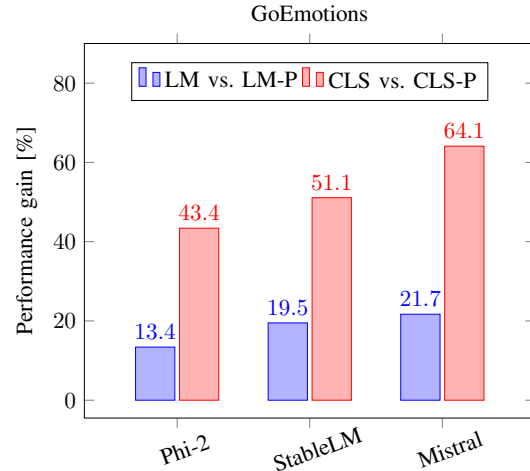[7]https://huggingface.co/spaces/lmsys/chatbot-arena-leaderboard

Fig. 2. Performance gains of personalized vs. non-personalized methods on the GoEmotions dataset.

for the remaining methods, we compared its performance with Mistral 7B model, which has a decoder-only architecture more than twice the size of Flan-T5.

For fine-tuning and evaluation, our computational infrastructure consisted of four NVIDIA GeForce RTX 3090 GPUs, each with 24 GB of vRAM. Due to memory limitations, we employed modern fine-tuning techniques. We load the models using 4-bit NormalFloat (NF4) quantization and use qLoRA [48] on all linear layers with the exception of the very last layer. In the case of CLS scenario the last layer is a newly initialised full layer, and in case of the LM task it is the LM head loaded in full precision. We do training in floating point 16-bit (fp16) for the StableLM and Mistral models and in BFloat16 (bf16) for the Flan-T5 and Phi-2 models. We made these implementations in Python using the PyTorch[8] library and HuggingFace libraries such as transformers[9] and peft[10].

Given ChatGPT's remarkable performance in few-shot prompting, we wanted to examine its effectiveness in query methods (Q-0S vs. Q-1S and Q-2S) compared to the models from Section VI-B. Our investigation included both versions of ChatGPT: GPT-3.5 and GPT-4.

Prompts: Since Large Language Models (LLMs) operate as prompt-based reasoners, the input data for each experiment consisted of an instruction specifying the task to be performed by the model. Each prompt also included the text to be classified and a list of labels from which the model was expected to select the appropriate ones. The prompts we used are shown in the appendix A.

In fine-tuned personalized approaches (LM-P, CLS-P), we directly added the user ID to the prompt by inserting the line: `"### User ID: <user ID>"`. Additionally, in the query (Q-0S, Q-1S, Q-2S) and language modeling (LM, LM-
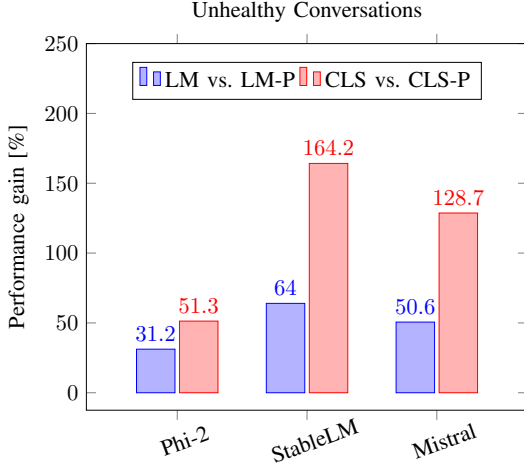
---

[8]https://pytorch.org/

[9]https://huggingface.co/docs/hub/transformers

[10]https://huggingface.co/docs/hub/peft

Fig. 3. Performance gains of personalized vs. non-personalized methods on the Unhealthy Conversations.

Setting Model		LM	LM-P	CLS	CLS-P
GoEmotions					
Phi-2	2.7B	28.99	32.87	30.03	43.07
StableLM	3B	26.55	31.72	27.42	41.44
Mistral	7B	28.36	34.52	26.77	43.94
Unhealthy Conversations					
Phi-2	2.7B	34.97	45.89	31.91	48.26
StableLM	3B	29.61	48.54	16.92	44.68
Mistral	7B	34.29	51.65	23.10	52.83

P) methods, we included a one-sentence request to the model within the prompt, specifying the expected format of the response:

„Please compose your response as a list of chosen labels, separated by commas."

In the Q-1S and Q-2S scenarios, we additionally included example texts and their correct responses within the prompt. Each example was separated from the instruction using a template on a new line: `"### Example <N>: <example> \n### Example <N> Response: <example's response>"`. Where in the Q-1S scenario, the token `<N>` was left empty, while in the Q-2S scenario, `<N>` was replaced with the number 1 or 2, depending on the example.

VII. RESULTS AND DISCUSSION

To compare the results across different approaches, we defined a *gain* metric to quantify the percentage increase in quality of the personalized model relative to the baseline model:

$$gain = \left(\frac{\text{personalized} - \text{baseline}}{\text{baseline}} \right) \times 100\%$$

The results indicate that personalization of subjective task classification has a consistent impact on model performance, leading to significant performance gains (Fig. 2 and Fig. 3). This is empirically evident in personalized fine-tuning, i.e., CLS-P and LM-P settings vs CLS and LM. Moreover, fine-tuning within LM-P and CLS-P settings generally leads to better performance than zero-shot Q-0S and few-shot Q-1S, Q-2S settings. This suggests that while few-shot learning can adapt models to specific tasks without extensive retraining, fine-tuning remains a more effective strategy for maximizing model performance on specialized tasks.

The performance gains from personalization are more pronounced in the Unhealthy Conversations dataset than in the

GoEmotions dataset – see Table II and Table I. This is in line with the research conducted in [32], which shows that the optimal prompt-based personalization strategy (Q-1S and Q-2S settings) needs to be tailored to the specific characteristics and challenges of the task.

The performance of few-shot settings varies across models and datasets, indicating that the effectiveness of few-shot learning might depend on the specific characteristics of models and the task. The GPT-family models from OpenAI, i.e., the GPT-3.5 and GPT-4, are the most consistent in few-shot settings – the performance of Q-0S < Q-1S < Q-2S (see Table II) – meaning they benefit more from an extended user context. On the other hand, the Mistral model does not fully utilize an extended user context in Q-1S and Q-2S few-shot settings despite undergoing an instruction fine-tuning procedure [41] as GPT-family models. Like Mistral, the instruction-based fine-tuning in Flan-T5 does not correspond well with personalized prompt-based approaches for subjective tasks.

The Phi-2 model was at a clear disadvantage with our prompts, as the model was not trained to follow instructions, not did it go through an alignment process known as Reinforcement Learning with Human Feedback (RLHF) or Direct Policy Optimization (DPO) [49].

In our analysis, we observe a clear difference between Language Modeling (LM) and Classification (CLS) tasks, especially when considering their effectiveness in personalized settings across various datasets. Specifically, when dealing with the GoEmotions dataset, the personalized classification (CLS-P) method outperforms the personalized language modeling (LM-P) approach. This difference can be attributed to the GoEmotions dataset containing a wide range of labels, making it more challenging for language modeling techniques to capture subtle emotional nuances effectively. On the other hand, when evaluating the Unhealthy Conversations dataset, personalized language modeling (LM-P) shows notably better performance in one out of three experiments compared to personalized classification (CLS-P).

TABLE II

THE PERFORMANCE OF LARGE LANGUAGE MODELS (LLM) ON *GoEmotion* AND *Unhealthy Conversations* DATASETS, COMPUTED FOR BASELINE QUERY WITH ZERO-SHOT SETTING (Q-0S) VS. PERSONALIZED QUERY WITH 1-SHOT (Q-1S), 2-SHOT (Q-2S) AND FINE-TUNED LANGUAGE MODELING HEAD SETTING (LM-P). F1-MACRO SCORES EXPRESSED IN PERCENTAGE POINTS.

Model	Setting	Q-0S	Q-1S	Q-2S	LM-P
GoEmotions					
Flan-T5	3B	17.97	17.79	17.87	—
Mistral	7B	22.20	17.91	20.52	34.52
GPT 3.5	-	23.33	22.58	22.33	—
GPT 4	-	26.74	26.82	26.64	—
Unhealthy Conversations					
Flan-T5	3B	14.11	12.81	12.68	—
Mistral	7B	18.90	26.12	23.21	51.65
GPT 3.5	-	21.51	23.11	25.49	—
GPT 4	-	24.94	27.29	30.57	—

TABLE III

THE PERFORMANCE OF LARGE LANGUAGE MODELS (LLM) ON *GoEmotion* AND *Unhealthy Conversations* DATASETS, COMPUTED FOR BASELINE CLASSIFICATION (CLS) VS. PERSONALIZED CLASSIFICATION (CLS-P) TO SHOW A COMPARISON BETWEEN OUR BEST MODEL WITH DECODER-ONLY ARCHITECTURE AND OUR MODEL WITH ENCODER-DECODER ARCHITECTURE. METRICS ARE EXPRESSED IN PERCENTAGE POINTS.

Model	Setting	CLS (F1-macro)	CLS-P (F1-macro)	Gain [%]
GoEmotions				
Flan-T5	3B	32.64	45.68	39.95
Mistral	7B	26.77	43.94	64.14
Unhealthy Conversations				
Flan-T5	3B	38.57	59.42	54.06
Mistral	7B	23.10	52.83	128.70

The performance differences between LM-P and CLS-P are less pronounced for Unhealthy Conversations than for GoEmotions. This is likely because Unhealthy Conversations has fewer labels, which suggests that label complexity has a significant impact on the effectiveness of personalized fine-tuning strategies. Tailoring fine-tuning approaches to the specific challenges presented by the task at hand is crucial. Dataset characteristics play a significant role in optimizing model performance.

Table III presents a comparison between Flan-T5, an encoder-decoder 3B parameter model, and Mistral, a decoder-only 7B model which is more than twice its size. Despite the potentially higher gains observed with the decoder-only model, the encoder-decoder architecture achieves better performance after fine-tuning (CLS vs CLS-P settings).

VIII. CONCLUSIONS AND FUTURE WORK

Our research underlines the crucial role of personalization in enhancing large language models (LLMs) for tasks involving subjective text perception. Through comprehensive experiments, we established that personalized fine-tuning significantly outperforms conventional zero-shot and few-shot learning methods, especially in the context of datasets with varying label complexities, such as GoEmotions and Unhealthy Conversations. The findings suggest that the success of personalization strategies is linked to the dataset's characteristics, underscoring the need for task-specific personalization approaches.

The study also reveals that LLMs' architecture and size critically influence the efficacy of personalization. Models like Mistral and the GPT family, which can follow detailed prompts and extended user contexts, show better improvements to models not specifically trained for instruction following or alignment through reinforcement learning.

Future research directions include examining the impact of personalization across a broader array of LLMs and subjective tasks and incorporating more contextual factors into personalization strategies. This could further enhance the precision and user-relevance of LLM outputs in personalized NLP applications.

REFERENCES

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[2] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, "Large language models are zero-shot reasoners," *Advances in neural information processing systems*, vol. 35, pp. 22 199–22 213, 2022.

[3] Q. Dong, L. Li, D. Dai, C. Zheng, Z. Wu, B. Chang, X. Sun, J. Xu, and Z. Sui, "A survey for in-context learning," *arXiv preprint arXiv:2301.00234*, 2022.

[4] J. D. M.-W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of naacL-HLT*, vol. 1. Minneapolis, Minnesota, 2019, p. 2.

[5] E. Cambria, X. Zhang, R. Mao, M. Chen, and K. Kwok, "Senticnet 8: Fusing emotion ai and commonsense ai for interpretable, trustworthy, and explainable affective computing," in *International Conference on Human-Computer Interaction (HCII)*, 2024.

[6] M. Wierzba, M. Riegel, J. Kocoń, P. Miłkowski, A. Janz, K. Klessa, K. Juszczyk, B. Konat, D. Grimling, M. Piasecki *et al.*, "Emotion norms for 6000 polish word meanings with a direct mapping to the polish wordnet," *Behavior Research Methods*, pp. 1–16, 2021.

[7] J. Kocoń, "Deep emotions across languages: A novel approach for sentiment propagation in multilingual wordnets," in *2023 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2023, pp. 744–749.

[8] S. Porsdam Mann, B. D. Earp, N. Møller, S. Vynn, and J. Savulescu, "Autogen: A personalized large language model for academic enhancement—ethics and proof of principle," *The American Journal of Bioethics*, vol. 23, no. 10, pp. 28–41, 2023.

[9] H. Lyu, S. Jiang, H. Zeng, Y. Xia, and J. Luo, "Llm-rec: Personalized recommendation via prompting large language models," *arXiv preprint arXiv:2307.15780*, 2023.

[10] M. Abbasian, I. Azimi, A. M. Rahmani, and R. Jain, "Conversational health agents: A personalized llm-powered agent framework," *arXiv preprint arXiv:2310.02374*, 2023.

[11] Z. Chen, "Palr: Personalization aware llms for recommendation," *arXiv preprint arXiv:2305.07622*, 2023.

[12] J. Wu, R. Antonova, A. Kan, M. Lepert, A. Zeng, S. Song, J. Bohg, S. Rusinkiewicz, and T. Funkhouser, "Tidybot: Personalized robot assistance with large language models," *arXiv preprint arXiv:2305.05658*, 2023.

[13] L. Zhu, R. Mao, E. Cambria, and B. J. Jansen, "Neurosymbolic ai for personalized sentiment analysis," in *Proceedings of HCII*, 2024.

[14] J. Kocoń, A. Figas, M. Gruza, D. Puchalska, T. Kajdanowicz, and P. Kazienko, "Offensive, aggressive, and hate speech analysis: From data-centric to human-centered approach," *Information Processing & Management*, vol. 58, no. 5, p. 102643, 2021.

[15] K. Kanclerz, A. Figas, M. Gruza, T. Kajdanowicz, J. Kocoń, D. Puchalska, and P. Kazienko, "Controversy and conformity: from generalized to personalized aggressiveness detection," 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)*, 2021, pp. 5915–5926.

[16] J. Kocoń, M. Gruza, J. Bielaniewicz, D. Grimling, K. Kanclerz, P. Miłkowski, and P. Kazienko, "Learning personal human biases and representations for subjective tasks in natural language processing," in *2021 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2021, pp. 1168–1173.

[17] P. Miłkowski, M. Gruza, K. Kanclerz, P. Kazienko, D. Grimling, and J. Kocoń, "Personal bias in prediction of emotions elicited by textual opinions," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Student Research Workshop*, 2021, pp. 248–259.

[18] J. Bielaniewicz, K. Kanclerz, P. Miłkowski, M. Gruza, K. Karanowski, P. Kazienko, and J. Kocoń, "Deep-sheep: Sense of humor extraction from embeddings in the personalized context," in *2022 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2022, pp. 967–974.

[19] A. Ngo, A. Candri, T. Ferdinan, J. Kocoń, and W. Korczynski, "Studemo: A non-aggregated review dataset for personalized emotion recognition," in *Proceedings of the 1st Workshop on Perspectivist Approaches to NLP@ LREC2022*, 2022, pp. 46–55.

[20] K. Kanclerz, M. Gruza, K. Karanowski, J. Bielaniewicz, P. Miłkowski, J. Kocoń, and P. Kazienko, "What if ground truth is subjective? personalized deep neural hate speech detection," in *Proceedings of the 1st Workshop on Perspectivist Approaches to NLP@ LREC2022*, 2022, pp. 37–45.

[21] P. Miłkowski, S. Saganowski, M. Gruza, P. Kazienko, M. Piasecki, and J. Kocoń, "Multitask personalized recognition of emotions evoked by textual content," in *2022 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*. IEEE, 2022, pp. 347–352.

[22] P. Miłkowski, K. Karanowski, P. Wielopolski, J. Kocoń, P. Kazienko, and M. Zięba, "Modeling uncertainty in personalized emotion prediction with normalizing flows," in *2023 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2023.

[23] K. Kanclerz, J. Bielaniewicz, M. Gruza, J. Kocoń, S. Woźniak, and P. Kazienko, "Towards model-based data acquisition for subjective multi-task nlp problems," in *2023 IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2023, pp. 726–735.

[24] K. Kanclerz, K. Karanowski, J. Bielaniewicz, M. Gruza, P. Miłkowski, J. Kocoń, and P. Kazienko, "Pals: Personalized active learning for subjective tasks in nlp," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023, pp. 13 326–13 341.

[25] B. Koptyra, A. Ngo, Ł. Radliński, and J. Kocoń, "Clarin-emo: Training emotion recognition models using human annotation and chatgpt," in *International Conference on Computational Science*. Springer, 2023, pp. 365–379.

[26] J. Kocon, J. Baran, and K. Kanclerz, "Multi-modal personalized hate speech analysis using differential dataset cartography." in *DE-FACTIFY@ AAAI*, 2023.

[27] W. Mieleszczenko-Kowszewicz, K. Kanclerz, J. Bielaniewicz, M. Oleksy, M. Gruza, S. Wozniak, E. Dzieciol, P. Kazienko, and J. Kocon, "Capturing human perspectives in nlp: Questionnaires, annotations, and biases." in *NLPerspectives@ ECAI*, 2023.

[28] T. Ferdinan and J. Kocoń, "Personalized models resistant to malicious attacks for human-centered trusted ai," in *The AAAI-23 Workshop on Artificial Intelligence Safety (SafeAI 2023)*, 2023.

[29] ——, "Fortifying nlp models against poisoning attacks: The power of personalized prediction architectures," *Information Fusion*, p. 102692, 2024.

[30] F. Mireshghallah, V. Shrivastava, M. Shokouhi, T. Berg-Kirkpatrick, R. Sim, and D. Dimitriadis, "Useridentifier: Implicit user representations for simple and effective personalized sentiment analysis," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2022, pp. 3449–3456.

[31] P. Kazienko, J. Bielaniewicz, M. Gruza, K. Kanclerz, K. Karanowski, P. Miłkowski, and J. Kocoń, "Human-centered neural reasoning for subjective content processing: Hate speech, emotions, and humor," *Information Fusion*, vol. 94, pp. 43–65, 2023.

[32] J. Kocoń, I. Cichecki, O. Kaszyca, M. Kochanek, D. Szydło, J. Baran *et al.*, "Chatgpt: Jack of all trades, master of none," *Information Fusion*, p. 101861, 2023.

[33] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901.

[34] S. Yao, D. Yu, J. Zhao, I. Shafran, T. L. Griffiths, Y. Cao, and K. R. Narasimhan, "Tree of thoughts: Deliberate problem solving with large language models," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[35] R. M. French, "Catastrophic forgetting in connectionist networks," *Trends in cognitive sciences*, vol. 3, no. 4, pp. 128–135, 1999.

[36] C.-A. Li and H.-Y. Lee, "Examining forgetting in continual pre-training of aligned large language models," *arXiv preprint arXiv:2401.03129*, 2024.

[37] S. Kotha, J. M. Springer, and A. Raghunathan, "Understanding catastrophic forgetting in language models via implicit inference," *arXiv preprint arXiv:2309.10105*, 2023.

[38] Y. Zhai, S. Tong, X. Li, M. Cai, Q. Qu, Y. J. Lee, and Y. Ma, "Investigating the catastrophic forgetting in multimodal large language models," *ArXiv*, vol. abs/2309.10313, 2023.

[39] D. Demszky, D. Movshovitz-Attias, J. Ko, A. Cowen, G. Nemade, and S. Ravi, "GoEmotions: A dataset of fine-grained emotions," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds. Online: Association for Computational Linguistics, Jul. 2020.

[40] I. Price, J. Gifford-Moore, J. Flemming, S. Musker, M. Roichman, G. Sylvain, N. Thain, L. Dixon, and J. Sorensen, "Six attributes of unhealthy conversations," in *Proceedings of the Fourth Workshop on Online Abuse and Harms*, S. Akiwowo, B. Vidgen, V. Prabhakaran, and Z. Waseem, Eds. Online: Association for Computational Linguistics, Nov. 2020.

[41] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. d. l. Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier *et al.*, "Mistral 7b," *arXiv preprint arXiv:2310.06825*, 2023.

[42] B. Roziere, J. Gehring, F. Gloeckle, S. Sootla, I. Gat, X. E. Tan, Y. Adi, J. Liu, T. Remez, J. Rapin *et al.*, "Code llama: Open foundation models for code," *arXiv preprint arXiv:2308.12950*, 2023.

[43] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[44] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma *et al.*, "Scaling instruction-finetuned language models," *arXiv preprint arXiv:2210.11416*, 2022.

[45] Y. Li, S. Bubeck, R. Eldan, A. Del Giorno, S. Gunasekar, and Y. T. Lee, "Textbooks are all you need ii: phi-1.5 technical report," *arXiv preprint arXiv:2309.05463*, 2023.

[46] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang *et al.*, "Training language models to follow instructions with human feedback," *Advances in Neural Information Processing Systems*, vol. 35, pp. 27 730–27 744, 2022.

[47] S. Black, S. Biderman, E. Hallahan, Q. Anthony, L. Gao, L. Golding, H. He, C. Leahy, K. McDonell, J. Phang *et al.*, "Gpt-neox-20b: An open-source autoregressive language model," *arXiv preprint arXiv:2204.06745*, 2022.

[48] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "QLoRA: Efficient finetuning of quantized LLMs," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[49] R. Rafailov, A. Sharma, E. Mitchell, C. D. Manning, S. Ermon, and C. Finn, "Direct preference optimization: Your language model is secretly a reward model," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. [Online]. Available: https://openreview.net/forum?id=HPuSIXJaa9

APPENDIX

GoEmotions Prompts

Prompt for Q-0S and LM scenarios

Categorize the following text by selecting the most appropriate emotion from the provided list. Emotions can be subtle or overt, so analyze the text carefully to make an accurate classification. Please compose your response as a list of chosen emotions, separated by commas.

Text:
<text>

Emotions:
- <list of the all possible labels from hyphens>
Response:

Prompt for CLS scenario

Categorize the following text by selecting the most appropriate emotion from the provided list. Emotions can be subtle or overt, so analyze the text carefully to make an accurate classification.

Text:
<text>

Emotions:
- <list of the all possible labels from hyphens>
Response:

Prompt for Q-1S scenario

Knowing that for the given example was provided the response given below categorize the following text by selecting the most appropriate emotion from the provided list. Emotions can be subtle or overt, so analyze the text carefully to make an accurate classification. Please compose your response as a list of chosen emotions, separated by commas.

Example:
<example text>

Example Response:
<user's annotations for the example>

Text:
<text>

Emotions:
- <list of the all possible labels from hyphens>
Response:

Prompt for Q-2S scenario

Knowing that for the given examples were provided the responses given below categorize the following text by selecting the most appropriate emotion from the provided list. Emotions can be subtle or overt, so analyze the text carefully to make an accurate classification. Please compose your response as a list of chosen emotions, separated by commas.

Example 1:
<first example text>

Example 1 Response:
<user's annotations for the first example>

Example 2:
<second example text>

Example 2 Response:
<user's annotations for the second example>

Text:
<text>

Emotions:
- <list of the all possible labels from hyphens>
Response:

Prompt for CLS-P scenario

Categorize the following text for the specified user by selecting the most appropriate emotion from the provided list. Emotions can be subtle or overt, so analyze the text carefully to make an accurate classification.

User ID:
<user ID>

Text:
<text>

Emotions:
- <list of the all possible labels from hyphens>
Response:

Prompt for LM-P scenario

Categorize the following text for the specified user by selecting the most appropriate emotion from the provided list. Emotions can be subtle or overt, so analyze the text carefully to make an accurate classification. Please compose your response as a list of chosen emotions, separated by commas.

User ID:
<user ID>

Text:
\<text\>

Emotions:
- \<list of the all possible labels from hyphens\>
Response:

Unhealthy Conversation Prompts

Prompt for Q-0S and LM scenarios

Categorize the following text by selecting the most appropriate label from the provided list. Labels represent different types of communication styles or tones, where each category denotes a specific attitude or approach that someone might exhibit when communicating with others. Analyze text carefully to make an accurate categorization. Please compose your response as a list of chosen labels, separated by commas.

Text:
\<text\>

Labels:
- \<list of the all possible labels from hyphens\>
Response:

Prompt for CLS scenario

Categorize the following text by selecting the most appropriate label from the provided list. Labels represent different types of communication styles or tones, where each category denotes a specific attitude or approach that someone might exhibit when communicating with others. Analyze text carefully to make an accurate categorization.

Text:
\<text\>

Labels:
- \<list of the all possible labels from hyphens\>
Response:

Prompt for Q-1S scenario

Knowing that for the given example was provided the response given below categorize the following text by selecting the most appropriate label from the provided list. Labels represent different types of communication styles or tones, where each category denotes a specific attitude or approach that someone might exhibit when communicating with others. Analyze text carefully to make an accurate categorization. Please compose your response as a list of chosen emotions, separated by commas.

Example:
\<example text\>

Example Response:
\<user's annotations for the example\>

Text:
\<text\>

Labels:
- \<list of the all possible labels from hyphens\>
Response:

Prompt for Q-2S scenario

Knowing that for the given examples were provided the responses given below categorize the following text by selecting the most appropriate label from the provided list. Labels represent different types of communication styles or tones, where each category denotes a specific attitude or approach that someone might exhibit when communicating with others. Analyze text carefully to make an accurate categorization. Please compose your response as a list of chosen emotions, separated by commas.

Example 1:
\<first example text\>

Example 1 Response:
\<user's annotations for the first example\>

Example 2:
\<second example text\>

Example 2 Response:
\<user's annotations for the second example\>

Text:
\<text\>

Labels:
- \<list of the all possible labels from hyphens\>
Response:

Prompt for CLS-P scenario

Categorize the following text for the specified user by selecting the most appropriate label from the provided list. Labels represent different types of communication styles or tones, where each category denotes a specific attitude or approach that someone might exhibit when communicating with others. Analyze text carefully to make an accurate categorization.

User ID:
\<user ID\>

Text:
\<text\>

Labels:
- \<list of the all possible labels from hyphens\>
Response:

Prompt for LM-P scenario

Categorize the following text for the specified user by selecting the most appropriate label from the provided list. Labels represent different types of communication styles or tones, where each category denotes a specific attitude or approach that someone might exhibit when communicating with others. Analyze text carefully to make an accurate categorization. Please compose your response as a list of chosen labels, separated by commas.

User ID:
\<user ID\>

Text:
\<text\>

Labels:
- \<list of the all possible labels from hyphens\>
Response: