HealthSLM-Bench: Benchmarking Small Language Models for Mobile and Wearable Healthcare Monitoring

Xin Wang[*], **Ting Dang**[*], **Xinyu Zhang**[†], **Vassilis Kostakos**[*], **Michael Witbrock**[†], **Hong Jia**[†*]
[*]School of Computing and Information Systems, University of Melbourne, Australia
[†]School of Computer Science, University of Auckland, New Zealand
xw17@student.unimelb.edu.au, {ting.dang, vassilis.kostakos}@unimelb.edu.au,
{xinyu.zhang, m.witbrock, hong.jia}@auckland.ac.nz

Abstract

Mobile and wearable healthcare monitoring play a vital role in facilitating timely interventions, managing chronic health conditions, and ultimately improving individuals' quality of life. Previous studies on large language models (LLMs) have highlighted their impressive generalization abilities and effectiveness in healthcare prediction tasks. However, most LLM-based healthcare solutions are cloud-based, which raises significant privacy concerns and results in increased memory usage and latency. To address these challenges, there is growing interest in compact models, Small Language Models (SLMs), which are lightweight and designed to run locally and efficiently on mobile and wearable devices. Nevertheless, how well these models perform in healthcare prediction remains largely unexplored. We systematically benchmarked SLMs on health prediction tasks using zero-shot, few-shot, and instruction fine-tuning approaches, and deployed the best performing fine-tuned SLMs on mobile devices to evaluate their real-world efficiency and predictive performance in practical healthcare scenarios. Our results show that SLMs can achieve performance comparable to LLMs while offering substantial gains in efficiency and privacy. However, challenges remain, particularly in handling class imbalance and few-shot scenarios. These findings highlight SLMs, though imperfect in their current form, as a promising solution for next-generation, privacy-preserving healthcare monitoring.

1 Introduction

The proliferation of mobile and wearable devices, coupled with recent advances in deep learning, has significantly advanced the landscape of continuous health monitoring [1–4]. These technologies enable a range of real-time applications, from the detection of physiological anomalies [5] to the delivery of personalized interventions [6]. Meanwhile, large language models (LLMs) have demonstrated remarkable generalization in processing heterogeneous data and performing diverse downstream tasks [7, 8]. Early studies indicate that LLM-based analysis can provide a deeper contextual interpretation of sensor data and enable more adaptive health monitoring systems compared to conventional approaches [9].

Despite this promise, major obstacles impede the practical deployment of LLM-driven wearable health solutions. Current approaches usually depend on cloud-based inference, necessitating data transmission to external servers, which raises concerns around user privacy, data security, and communication latency [10–13]. Alternatively, on-device deployment is hindered by severe resource constraints typical of mobile and wearable hardware, as well as the real-time requirements of health applications, rendering full-sized LLMs infeasible for timely inference. These challenges highlight a critical need for efficient, privacy-preserving techniques that achieve competitive performance with LLMs, while being suitable for deployment on resource-limited mobile and wearable devices.

Small Language Models (SLMs) present a promising alternative by reducing memory consumption and facilitating deployment on mobile and wearable devices. On-device inference with SLMs not only lowers communication latency but also enhances the protection of sensitive personal data, while maintaining competitive performance on natural language processing tasks [14–17]. Nevertheless, their ability to interpret sensor data from mobile and wearable devices and accurately infer health conditions in real-world settings remains an open question. Although prior work [18] has demonstrated the feasibility of using SLMs on mobile devices to predict simple health status (e.g., fatigue, sleep quality), there is still a lack of comprehensive benchmarking that thoroughly evaluates SLMs for a wide range of health applications.

To bridge this gap, we present a comprehensive benchmark, HealthSLM-Bench, which aims to evaluate a variety of state-of-the-art (SOTA) SLMs on a suit of health prediction tasks spanning three publicly available datasets. Our benchmark systematically assesses model performance using three evaluation protocols: zero-shot, few-shot, and instruction-based fine-tuning. To assess practical feasibility, we further deploy top-performing fine-tuned models on mobile devices and rigorously evaluate their on-device efficiency in terms of memory usage and inference latency. Experimental results demonstrate that SLMs can achieve comparable performance compared with SOTA healthcare LLMs across eight healthcare monitoring tasks, while substantially reducing memory and latency overheads. Our main contributions are as follows:

- We introduce, HealthSLM-Bench, an extensive benchmark that systematically evaluates nine SOTA SLMs on eight health prediction tasks across three real-world mobile and wearable datasets.
- We investigate various evaluation paradigms, including zero-shot, few-shot, and instruction-based fine-tuning, providing a comprehensive performance analysis under different adaptation scenarios.
- We demonstrate the feasibility of deploying fine-tuned SLMs on resource-constrained mobile devices and quantify their efficiency in terms of real-world memory and latency footprints.

2 Related Work

LLMs for health monitoring. With the rise of mobile and wearable devices, a variety of human-centered sensing signals can be continuously collected, enabling ongoing monitoring of human health in daily life. Recent studies have shown that the physical status data collected by mobile devices is strongly associated with health status [19–21]. Their work demonstrates how passive wearable sensor data can be effectively utilized to predict depression in adolescents using traditional ML models. However, these approaches, typically trained on specific datasets or tailored architectures, often struggle to generalize across heterogeneous tasks, and contexts [22]. LLMs, powered by their generalization capabilities, have shown great success in the healthcare domain. For example, Health-LLM [23] and MultiEEG-GPT [24] demonstrate the effectiveness of leveraging LLMs in healthcare monitoring through textual and physiological data. Instead of just deploying these models directly for healthcare applications, recent work has explored domain adaptation strategies such as few-shot prompting, instruction tuning, and domain-specific fine-tuning to improve performance on medical tasks [25]. Notably, PaLM2 [26] illustrates the benefits of combining diverse adaptation strategies (e.g. few-shot and fine-tuned) across medical datasets. Meanwhile, evaluations of GPT-4 highlight that SOTA LLMs may reduce the reliance on extensive adaptation, as they already demonstrate strong capacity for medical reasoning with limited supervision [27]. More recently, applied systems such as PhysioLLM [28] have integrated LLMs with wearable sensor data to provide personalized health insights, highlighting their adaptability across users and contexts. However, despite these advances, their computational overhead makes them impractical for privacy-sensitive, real-time mobile healthcare monitoring.

Small Language Models. SLMs are defined as models that are smaller in scale relative to the widely recognised LLMs, typically comprising no more than 7 billion parameters [29]. Recent research has highlighted the efficiency and strong task performance of SLMs as lightweight alternatives to LLMs, particularly for deployment in resource-constrained environments [30, 31]. For example,

Table 1: An Example of Prompt Construction for Zero-shot learning. Z_S represents "Zero-shot".

Context	Prompt
Instruction	You are a personalized healthcare agent trained to predict fatigue which ranges from 1 to 5 based on physiological data and user information.
Main Query	The recent 14-days sensor readings show: {14} days sensor readings show: Steps: {"1476.0, 4809.0, ..., NaN"} steps, Burned Calories: {"169.0, 419.0 ..., NaN"} calories, Resting Heart Rate: {"53.24, 52.24, ..., 51.40"} beats/min, Sleep Minutes: {"110.0, 524.0, ..., 481.0"} minutes, [Mood]: 3 out of 5. What would be the predicted fatigue level?
Output Constraints	The predicted fatigue level is:

$$\textit{Prompt } Z_S = \textit{Instruction}_{Z_S} + \underline{\textit{Main query}} + \textit{Output Constraints} \tag{1}$$

Phi-3-mini-4k-Instruct, developed by Microsoft [32], contains 3.8 billion parameters and is trained on a curated blend of synthetic and high-quality public datasets, emphasizing reasoning capabilities. TinyLlama-1.1B [33] builds on Llama 2 through parameter reduction and subsequent fine-tuning using UltraChat, a broad synthetic dialogue dataset. Similarly, Google's Gemma2-2B [34], based on Gemini research, demonstrates robust results in text generation, summarization, and reasoning benchmarks. SmolLM-1.7B from HuggingFace [35] further diversifies training by leveraging synthetic educational materials and a breadth of domain samples, and Qwen2-1.5B [16] achieves SOTA performance in both coding and mathematics despite its small footprint. Meta's Llama-3.2 series [36] continues this trend by releasing 1B and 3B parameter models designed for edge applications. While these developments affirm the viability of SLMs for a range of natural language processing tasks, the current literature leaves the open question of how effectively these compact models generalize to health prediction tasks. This is especially salient for high-stakes applications in healthcare, where accuracy and timeliness are paramount.

Deployment of On-Device SLMs. Deploying SLMs on mobile and wearable devices is an interesting yet challenging task due to constraints such as limited computational power, memory, battery life, and the need for efficient, real-time processing. MobileAIBench [31] evaluated SLMs on an iPhone 14, offering a comprehensive framework for assessing latency, memory usage, and overall efficiency. Their results established the practical viability of running compact language models on mobile hardware. More recent research [18] explored SLMs for health event prediction [37] in a zero-shot context, underscoring their promise as privacy-preserving and practical alternatives to LLMs for healthcare monitoring on mobile and wearable devices. Despite these advances, existing work remains limited in scope. MobileAIBench concentrated on generic NLP tasks rather than domain-specific health applications. As a result, there is a lack of systematic analysis of real-world efficiency of SLMs in health-related tasks after deployment on mobile devices. In comparison, our study addresses this gap by conducting comprehensive evaluations of SLMs on mobile platforms, using detailed efficiency metrics to assess their practical feasibility for mobile health monitoring applications across various datasets, model structures, and tasks.

3 HealthSLM-Bench

We benchmark a variety of SLMs for mobile and wearable health applications using zero-shot and few-shot learning which enables in-context learning with a limited number of task-specific examples. Additionally, we instruction-tune these models on health datasets, aiming to significantly enhance their effectiveness for healthcare monitoring tasks.

3.1 Zero-shot and Few-shot Learning

Zero-shot learning. In the zero-shot learning setting, models were evaluated without prior exposure to any example inputs during inference. Each model was provided only with a task instruction, a

Table 2: An Example of Prompt Construction for Few-shot learning. Z_S and F_S represent “Zero-shot” and “Few-shot”, respectively.

Context	**Prompt**
Instruction	You are a health assistant. Your mission is to read the following examples and return your prediction based on the health query.
Examples	<example 1>, <example 2>, ... <example N>
Question	Finally, please answer to the below question: <*Prompt* Z_S>

$$\mathit{Examples} = (\mathit{Prompt}\ Z_S + \mathit{Answer})_N \tag{2}$$

$$\mathit{Prompt}\ F_S = \mathit{Instruction}_{F_S} + \mathit{Examples} + \mathit{Prompt}\ Z_S \tag{3}$$

main query describing the 14-day summary of sensor readings, and explicit output constraints (e.g., restricting output labels for fatigue to values within the range [1–5]), as shown in Table 1. This setup was designed to evaluate the intrinsic ability of the models to interpret and respond to healthcare-related queries based solely on task instructions. The zero-shot protocol thus serves as a baseline for performance, providing a reference point for subsequent experiments involving few-shot learning and instruction tuning.

Few-shot learning. Few-shot learning [38] was employed to enhance task comprehension by augmenting the model inputs with a small set of labeled examples. Unlike zero-shot learning, which relies solely on the model’s generalized knowledge, this approach leverages in-context learning to better interpret task-specific data. As shown in Table 2, the few-shot prompt (*Prompt* F_S), formalized in Equation 3, consists of an explicit instruction $\mathit{Instruction}_{F_S}$, a set of N example pairs (*Prompt* Z_S + *Answer*$)_N$, and the target query *Prompt* Z_S. Specifically, the $\mathit{Instruction}_{F_S}$ directs the model to review the N examples before responding to the target query. Each example follows the same structure as the zero-shot prompt, i.e., consisting of a task instruction and a main query, but also includes the corresponding answer. This design enables the model to ground its predictions in observed input–output patterns, capturing relationships that may be less apparent in a zero-shot setting. In our experiments, we varied the number of examples $N \in \{1, 3, 5, 10\}$ to examine its impact on performance, aiming to identify the most effective configuration. To maximize on-device efficiency, we did not implement chain-of-thought reasoning (CoT) [39] and self-consistency (SC) [40], as both introduce additional token generation and computational overhead that limit practicality on resource-constrained edge devices.

3.2 Instructional Tuning

Instructional tuning adapts language models to follow task-specific instructions by further training them on curated instruction–response pairs [41]. Unlike zero-shot or few-shot learning, which relies on a sole task description or in-context prompts at inference time, instructional tuning updates the model parameters themselves, enabling more robust and persistent task alignment. Specifically, the instruction–response pairs were formatted using the Alpaca-style template [42], which provides a lightweight and standardized structure widely adopted in instruction-tuning benchmarks [23, 43–45]. To enable efficient fine-tuning, we employed Low-Rank Adaptation (LoRA) [46], which introduces trainable low-rank decomposition matrices into the attention and feed-forward layers while keeping the original weights frozen. LoRA is particularly well-suited for on-device inference, as it allows effective model adaptation with minimal memory and computational overhead.

4 Experimental Setup

4.1 Datasets

We evaluate our methods using three health wearable sensor datasets: PMData [37], GLOBEM [47], and AW-FB [48]. From these datasets, we extract features derived from smartwatches raw sensor data, including steps, calories burned, resting heart rate, and sleep metrics, and use self-reported labels such as fatigue, stress, and readiness. For health event prediction, we format the temporal sequences of these features into 14-day windows and incorporate them into query prompts to generate predictions.

The predictions produced by SLMs are then compared with the self-reported ground-truth labels. Details of each dataset are provided below. The detailed task categorization and label distribution are provided in the Appendix.

PMData is a dataset that integrates life-logging and activity-logging information, comprising personalized health monitoring data collected from 16 participants over a period of five months. Using the Fitbit Versa 2 smartwatch wristband [49], objective signals such as calories burned, resting heart rate, step count, sleep duration, and more were gathered. In addition, participants provided self-reported measurements of their health status via the PMSys sports logging application, such as fatigue, mood, stress, etc. In our setting, these self-reports were categorized into prediction tasks with labels for fatigue, readiness, sleep quality, and stress [18, 23].

The dataset contains data relevant to the following research tasks:

- **Stress (STRS):** Quantification of individual stress levels, utilising both physiological measurements and self-reported data.
- **Readiness (READ):** Evaluation of preparedness for physical exertion or exercise, based on physiological and behavioural indicators.
- **Fatigue (FATG):** Detection and monitoring of fatigue states, as evidenced by physiological signals and self-assessment.
- **Sleep Quality (SQ):** Comprehensive assessment of sleep quality, including metrics such as total sleep duration, sleep efficiency, and the frequency and duration of nocturnal awakenings.

GLOBEM is a passive sensing dataset for health-domain analysis. Data were gathered from 497 participants between 2018 and 2021 using a custom mobile application alongside continuous fitness tracker monitoring (24/7). This dataset captures a wide range of daily human routines, including step counts, sleep efficiency, time spent in bed after waking, time to fall asleep, and wake periods while in bed. These signals reveal associations between everyday behaviors and well-being outcomes. In our experiment, we use these behavioral signals as inputs and predict mental health conditions such as depression and anxiety [23].

The dataset contains data relevant to the following mental health assessments:

- **Depression (DEP):** Detection of depressive symptoms using machine learning models that analyse user behaviour and linguistic patterns.
- **Anxiety (ANX):** Identification of anxiety through behavioural indicators, such as disrupted sleep patterns, and physiological responses, including elevated heart rate.

AW_FB is a wearable dataset designed by Harvard University to study the relationship between physical activity patterns and physiological metrics, gathered from 46 participants that wear GENE-Activ [50], Apple Watch Series 2 [51] and a Fitbit Charge HR2 [52] in a lab-based protocol. The recorded sensor data includes daily step count, heart rate, activity duration, burned calories, and metabolic equivalent of task (MET) Value. This dataset was tested to predict 6 different physical activity intensities, including lying, sitting, walking self-paced, 3 METS, 5 METS, and 7 METS.

The dataset contains data relevant to the following physiological and behavioural assessments:

- **Calorie Burn (CAL):** Estimation of individual energy expenditure during physical activities.
- **Activity (ACT):** Classification of physical activity types based on sensor-derived data.

4.2 Models

We selected nine SOTA SLMs ranging from 1B to 4B parameters, including Google's Gemma-2-2B-it [34], Microsoft's Phi-3-mini-4k-instruct and Phi-3.5-mini [53], HuggingFace's SmolLM-1.7B [35], Alibaba's Qwen2-1.5B and Qwen2.5-1.5B [16], TinyLlama's TinyLlama-1.1B [54], and Meta-Llama's Llama-3.2-1B and Llama-3B [36]. Further details are provided in Appendix C.

4.3 Implementation details

Data processing. Following previous work [18, 23, 55], we standardize all datasets into daily sequences spanning 14-day windows. Task-specific labels are assigned accordingly. Each dataset

Table 3: Performance of LLMs and SLMs under **zero-shot (ZS)** setting across eight healthcare monitoring tasks. **STRS**: Stress, **READ**: Readiness, **FATG**: Fatigue, **SQ**: Sleep Quality, **ANX**: Anxiety, **DEP**: Depression, **ACT**: Activity, **CAL**: Calories. Best result is in **bold**, second-best result is underlined. '-' denotes model failed to produce valid prediction.

		PMData				**GLOBEM**		**AW-FB**	
	Model	**STRS (↓)**	**READ (↓)**	**FATG (↑)**	**SQ (↓)**	**ANX (↓)**	**DEP (↓)**	**ACT (↑)**	**CAL (↓)**
LLMs (ZS)	MedAlpaca	0.76	2.18	46.8	0.68	1.23	0.89	21.7	35.0
	PMC-Llama	1.33	4.83	0.00	2.25	2.33	2.23	–	43.4
	Asclepius	0.43	**1.44**	27.3	0.45	**0.82**	1.10	–	**28.9**
	ClinicalCamel	0.40	2.11	58.1	**0.37**	0.97	0.79	16.3	43.4
	Flan-T5	**0.36**	1.82	56.8	0.56	2.84	2.89	23.4	66.0
	Palmyra-Med	0.83	5.01	43.5	0.44	2.07	1.99	**29.7**	75.3
	Llama 2	0.57	2.86	41.2	0.89	1.19	1.23	–	–
	BioMedGPT	0.37	2.12	61.2	0.41	0.95	**0.85**	12.2	–
	BioMistral	0.55	2.12	56.6	0.45	0.90	–	18.4	41.0
	GPT-3.5	–	2.38	70.8	0.87	–	–	13.8	36.4
	GPT-4	–	2.22	**72.2**	0.73	–	–	22.6	75.2
	Gemini-Pro	0.79	1.69	34.0	0.78	1.03	0.95	17.7	31.4
	Mean	0.64	2.56	41.54	0.60	1.43	1.44	19.53	47.60
SLMs (ZS)	Gemma-2-2b-it	0.72	2.07	52.84	0.47	0.91	**0.53**	-	105.12
	Phi-3-mini-4k	0.46	**1.52**	62.88	0.48	1.08	1.26	17.39	93.80
	SmolLM-1.7B	1.42	2.99	11.04	1.00	2.59	2.87	**21.74**	277.21
	Qwen2-1.5B	**0.40**	2.03	63.21	**0.45**	1.42	1.65	14.05	185.22
	TinyLlama-1.1B	0.43	2.06	51.17	0.47	2.40	2.58	19.73	198.72
	Llama-3.2-1B	0.40	1.87	**63.79**	0.69	1.51	1.85	11.71	280.32
	Llama-3.2-3B	0.67	2.24	40.80	0.47	1.26	0.75	15.72	**19.70**
	Phi-3.5-mini	0.41	2.34	61.20	0.46	**0.88**	0.84	15.38	56.75
	Qwen2.5-1.5B	0.56	2.25	62.88	0.93	1.37	1.63	15.72	72.20
	Mean	0.61	2.15	52.20	0.60	1.49	1.55	16.40	143.23

is extracted, randomly shuffled, and split into training and testing subsets in an 8:2 ratio. The tasks are categorized as either classification (fatigue, readiness, sleep quality, stress, anxiety, depression, activity) or regression (calories). The label distributions for each task are provided in the Appendix.

Model deployment. To assess efficiency and feasibility, we deploy the top-performing health-domain–adapted SLMs, which is adapted for the health domain and instructional tuned using health-related datasets, on an iPhone 15 Pro Max equipped with 8 GB of RAM. These models are converted to the GGUF format (Generalized Graphical Unified Format) [56] to ensure compatibility with lightweight inference engines such as Llama.cpp [57]. Due to the strict memory constraints of mobile devices, we apply 4-bit quantization to enable efficient deployment. As shown in prior studies [31], quantization lowers computational costs while maintaining most of the model's performance. Both the conversion and quantization steps are performed using Llama.cpp [58].

Evaluation metrics. To evaluate model performance under *zero-shot*, *few-shot*, and *instructional-tuning* settings, we use mean absolute error (MAE) for regression tasks and accuracy for classification tasks. For efficiency evaluation of mobile deployment, we assess the models latency using metrics such as Time-to-First-Token (TTFT), Input Tokens Per Second (ITPS), Output Tokens Per Second (OTPS), and Output Evaluation Time (OET) and Total Time. In addition, We also track CPU and RAM usage to evaluate on-device resource consumption. Further details are provided in the Appendix E.

5 Results and Discussion

We compare the performance of SLMs and SOTA LLMs under the same settings as in [23].

5.1 Overall Performance

Zero-shot learning. As shown in Table 3, SLMs achieve comparable or better performance than LLMs across the three health datasets. For stress prediction, SLMs achieve a lower mean MAE of 0.61, compared to 0.64 for LLMs, where lower values indicate better performance. SLMs also outperform LLMs in readiness and fatigue prediction, with a mean MAE of 2.15 for SLMs versus

Table 4: Performance of LLMs and SLMs under **few-shot (FS)** setting across across eight healthcare monitoring tasks. **STRS**: Stress, **READ**: Readiness, **FATG**: Fatigue, **SQ**: Sleep Quality, **ANX**: Anxiety, **DEP**: Depression, **ACT**: Activity, **CAL**: Calories. Best result is in **bold**, second-best result is underlined. '-' denotes model failed to produce valid prediction.

		PMData				**GLOBEM**		**AW-FB**	
	Model	**STRS** (↓)	**READ** (↓)	**FATG** (↑)	**SQ** (↓)	**ANX** (↓)	**DEP** (↓)	**ACT** (↑)	**CAL** (↓)
LLMs (FS-best)	MedAlpaca	0.78	1.94	36.2	0.69	**0.97**	**0.56**	19.3	36.7
	GPT-3.5	0.94	**1.62**	**73.9**	0.77	1.98	0.68	**26.3**	26.5
	GPT-4	**0.76**	1.64	61.3	**0.60**	1.11	0.60	15.4	**24.0**
	Gemini-Pro	1.10	2.20	24.8	0.80	1.30	1.05	15.0	37.2
	Mean	0.90	1.85	49.05	0.72	1.34	0.72	19.0	31.1
SLMs (FS-1)	Gemma-2-2b-it	**0.41**	2.30	**59.87**	**0.45**	2.04	2.40	-	24.22
	Phi-3-mini-4k	0.43	1.56	47.83	0.46	1.99	**1.94**	21.40	21.58
	SmolLM-1.7B	0.41	1.31	51.51	0.46	3.12	3.47	**22.10**	19.94
	Qwen2-1.5B	0.41	1.29	51.51	0.46	2.15	2.47	14.40	19.07
	TinyLlama-1.1B	0.41	1.30	51.51	0.46	3.10	3.39	14.00	18.97
	Llama-3.2-1B	0.55	1.50	51.51	0.65	2.32	3.03	20.40	18.43
	Llama-3.2-3B	0.79	1.87	28.76	0.54	**1.84**	2.01	18.10	37.45
	Phi-3.5-mini	0.41	1.36	51.51	0.46	3.06	3.42	14.40	51.33
	Qwen2.5-1.5B	0.43	**1.28**	54.52	0.47	3.10	3.44	14.70	**18.04**
	Mean	0.47	1.53	49.84	0.49	2.52	2.84	17.40	25.45
SLMs (FS-3)	Gemma-2-2b-it	0.48	1.66	44.82	0.49	-	-	-	-
	Phi-3-mini-4k	0.41	1.67	44.82	**0.45**	0.88	**0.54**	19.40	54.98
	SmolLM-1.7B	-	-	-	-	**0.87**	0.58	15.40	18.97
	Qwen2-1.5B	0.41	1.68	**51.51**	0.46	0.88	**0.54**	23.10	19.85
	TinyLlama-1.1B	-	-	-	-	2.93	3.04	14.40	**17.90**
	Llama-3.2-1B	0.44	1.73	49.83	0.54	0.88	**0.54**	15.40	18.47
	Llama-3.2-3B	0.41	1.78	**51.51**	0.47	1.19	1.12	**24.10**	19.27
	Phi-3.5-mini	0.41	**1.43**	**51.51**	0.46	0.91	0.55	**24.10**	32.94
	Qwen2.5-1.5B	**0.40**	1.44	37.12	0.76	1.37	0.64	18.10	**17.90**
	Mean	0.42	1.63	47.30	0.52	1.24	0.94	19.20	25.04
SLMs (FS-5)	Gemma-2-2b-it	0.48	1.35	**61.54**	0.47	-	-	-	-
	Phi-3-mini-4k	0.41	**1.32**	57.19	0.49	0.88	**0.56**	22.10	37.27
	SmolLM-1.7B	-	-	-	-	**0.87**	0.76	17.10	**18.58**
	Qwen2-1.5B	0.41	1.42	51.51	**0.46**	1.20	1.12	20.40	29.41
	TinyLlama-1.1B	-	-	-	-	3.15	3.51	**24.10**	37.00
	Llama-3.2-1B	0.44	1.42	52.51	**0.46**	1.18	1.38	15.10	27.18
	Llama-3.2-3B	0.41	1.59	52.17	**0.46**	1.18	1.23	18.40	28.54
	Phi-3.5-mini	0.41	1.41	51.51	**0.46**	1.46	1.56	**24.10**	23.69
	Qwen2.5-1.5B	**0.40**	1.44	41.47	0.49	1.28	1.52	17.40	28.50
	Mean	0.42	1.42	52.56	0.47	1.40	1.45	19.84	28.77
SLMs (FS-10)	Gemma-2-2b-it	0.49	**1.41**	**63.55**	0.50	1.23	1.09	-	-
	Phi-3-mini-4k	1.01	1.70	32.78	**0.45**	0.82	0.63	17.70	18.46
	SmolLM-1.7B	-	-	-	-	**0.77**	**0.53**	15.10	19.13
	Qwen2-1.5B	**0.41**	1.55	56.19	0.47	0.87	0.54	17.70	18.00
	TinyLlama-1.1B	-	-	-	-	-	-	21.10	**17.17**
	Llama-3.2-1B	0.89	1.61	8.36	0.46	0.87	0.77	15.70	19.47
	Llama-3.2-3B	0.49	1.83	39.80	0.48	2.04	1.23	19.10	18.06
	Phi-3.5-mini	0.42	**1.41**	34.11	0.48	0.77	1.10	**22.10**	18.87
	Qwen2.5-1.5B	0.66	2.47	33.44	0.50	0.87	0.54	17.40	19.09
	Mean	0.63	1.71	38.32	0.48	1.03	0.801	18.2	18.53

2.56 for LLMs, and a higher mean accuracy of 52.2% for SLMs compared to 41.54% for LLMs. For other tasks, including sleep quality, anxiety, depression, and activity, SLMs perform within a similar range to LLMs. Among the SLMs, Gemma-2-2B-it and Phi-3-mini-4k consistently deliver strong results for fatigue and readiness, while Qwen2.5-1.5B matches or exceeds LLM performance on several tasks. However, SLMs do have some limitations. SmolLM-1.7B often underperforms relative to LLMs, and most SLMs struggle with calorie estimation, where the mean MAE is 143.23 for SLMs compared to 47.6 for LLMs, suggesting that regression tasks may be more challenging for SLMs.

In sum, under zero-shot settings, SLMs generally match or surpass LLMs on most health prediction tasks, notably achieving better results in stress, readiness, and fatigue predictions. Leading SLMs, such as Gemma-2-2B-it and Phi-3-mini-4k, show consistent strength compared with SOTA LLMs.

Few-shot learning. The few-shot (FS) results are shown in Table 4. For LLMs, we compare the best few-shot performance (FS-best) to SLMs using a range of few-shot sample counts (1, 3, 5, 10) in

Table 5: Performance of LLMs and SLMs under **instruction tuning (LoRA)** setting across eight healthcare monitoring tasks. **STRS**: Stress, **READ**: Readiness, **FATG**: Fatigue, **SQ**: Sleep Quality, **ANX**: Anxiety, **DEP**: Depression, **ACT**: Activity, **CAL**: Calories. Best result is in **bold**, second-best result is underlined. '-' denotes model failed to produce valid prediction.

		PMData				**GLOBEM**		**AW-FB**	
	Model	**STRS (↓)**	**READ (↓)**	**FATG (↑)**	**SQ (↓)**	**ANX (↓)**	**DEP (↓)**	**ACT (↑)**	**CAL (↓)**
LLMs (lora)	HealthAlpaca-lora-7b	0.53	**1.40**	50.0	0.58	**0.62**	0.51	27.4	43.6
	HealthAlpaca-lora-13b	**0.34**	1.56	**54.8**	**0.39**	1.04	**0.67**	**29.0**	**39.6**
	Mean	0.44	1.48	52.4	0.49	0.83	0.59	28.2	41.6
	Gemma-2-2b-it	-	-	-	0.51	1.27	1.02	**34.40**	**2.80**
	Phi-3-mini-4k	0.40	2.14	62.20	0.52	**0.81**	0.71	22.40	9.67
	SmolLM-1.7B	0.93	1.68	15.40	0.89	0.84	0.54	16.10	18.87
	Qwen2-1.5B	0.43	1.52	62.20	0.47	0.92	0.97	18.70	5.21
SLMs (lora)	TinyLlama-1.1B	**0.40**	**1.30**	**63.20**	0.47	0.83	0.67	22.10	5.51
	Llama-3.2-1B	0.43	2.25	49.80	0.81	0.86	**0.54**	19.20	5.78
	Llama-3.2-3B	0.60	1.53	40.80	**0.47**	0.88	**0.54**	22.10	3.64
	Phi-3.5-mini	0.49	1.55	62.20	0.92	0.88	0.66	19.40	12.09
	Qwen2.5-1.5B	0.87	1.49	13.00	0.87	1.04	0.79	21.70	4.57
	Mean	0.57	1.68	46.10	0.66	0.93	0.72	21.80	7.57

SLMs. As shown in Table 4, even when provided with in-context examples in the one-shot setting (FS-1), SLMs demonstrate competitive performance compared to their larger counterparts across multiple healthcare monitoring tasks, and also outperforms zero-shot SLMs on average.

Comparing the performance across different few-shot settings reveals interesting patterns in SLM behavior. In the FS-1 setting, SLMs achieve competitive performance levels compared to LLMs across most tasks. For instance, SLMs achieve a mean of 0.47 for stress prediction compared to LLMs' 0.90, and 0.49 for sleep quality compared to LLMs' 0.72. As the number of few-shot examples increases from FS-1 to three-shot (FS-3), five-shot (FS-5), and ten-shot (FS-10), the performance shows task-dependent variations. For stress prediction, the mean performance remains relatively stable across all few-shot settings. Similarly, sleep quality prediction maintains consistent performance throughout the different few-shot configurations.

However, certain tasks exhibit different response patterns to increased few-shot examples. Anxiety and depression prediction tasks show notable improvement as the number of examples increases from FS-1 to FS-3, with further refinement observed in subsequent settings. This suggests that mental health prediction tasks may benefit more from additional contextual examples compared to physiological monitoring tasks when using SLMs without fine-tuning, which has also been observed in recent work comparing SLMs and LLMs in mental health prediction tasks [55]. As shown in Table 4, we also observed that the collapse pattern appears at FS-1, FS-3, and FS-5, but does not occur at FS-10. This phenomenon was observed only in PMData and LifeSnaps tasks, such as stress, fatigue, sleep quality and sleep disorder, while readiness remained unaffected and no collapse was noted in GLOBEM or AW-FB tasks. Upon label distribution inspection (*cf.* Appendix F.1), this trend appears to stem from limited label representation under low-shot settings, where only a few examples are provided. As the number of examples increases to FS-10, the broader label coverage yields a more representative distribution, thereby mitigating the collapse.

Overall, SLMs perform competitively with LLMs in few-shot healthcare tasks, even with just one example. More examples help models achieve more stable and reliable performance.

Instruction tuning. As shown in Table 5, both SLMs and SOTA LLMs [23] are instruction-tuned, yet SLMs outperform LLMs in tasks such as fatigue and calorie estimation. Specifically, SLMs achieve much lower mean values for fatigue (46.1 for SLMs *vs.* 52.4 for LLMs) and calorie estimation error (7.57 for SLMs compared to 41.6 for LLMs), demonstrating their superior accuracy in these important health measures. Although LLMs perform slightly better in stress, readiness, and activity prediction—with lower mean values for stress (0.44 for LLMs *vs.* 0.57 for SLMs), readiness (1.48 *vs.* 1.68), and higher mean values for activity (28.2 *vs.* 21.8)—these differences are relatively modest compared to the clear advantages of SLMs in fatigue and calorie estimation. For other tasks such as sleep quality, anxiety, and depression, both SLMs and LLMs show similar performance, with only minor differences in mean values. Notably, SLMs like TinyLlama-1.1B and Phi-3-mini-4k stand out for their strong and consistent results across multiple tasks. In less-performing cases (e.g., activity,

Table 6: Efficiency & Utilization of LLMs & SLMs on the PMData dataset.

	Model	TTFT(s)	ITPS(t/s)	OET(s)	OTPS(t/s)	Total Time(s)	CPU(%)	RAM(GB)
PMData	Phi-3-mini-4k	6.39	112.39	0.96	13.49	7.61	**70.20**	6.48
	TinyLlama-1.1B	**1.37**	**527.01**	**0.35**	**45.89**	**1.79**	117.98	**5.17**
	Llama-2-7b	29.12	24.74	27.85	3.04	57.43	379.31	7.15

anxiety and sleep quality) of SLMs, we observed that SLMs tend to predict only the majority classes without attempting to predict other, weaker classes (i.e., class-imbalance bias; *cf.* Appendix F.2), causing the model to become stuck at sub-optimal performance on those tasks. To mitigate this issue, we applied data augmentation (e.g., oversampling) to balance the label distribution (*cf.* Appendix B), which improved coverage of minority classes but did not lead to any performance gains.

In sum, these findings demonstrate that SLMs, when properly tuned, are not only competitive but often superior to LLMs for specific healthcare tasks, particularly fatigue and calorie estimation. This highlights the potential of SLMs for efficient, accurate, and practical healthcare applications, making them a compelling choice where resource efficiency and task-specific performance are essential.

5.2 Deployment Efficiency

To investigate efficiency and computational cost in real-world deployment, we ran inference with the two top-performing models, Phi-3-mini-4k and TinyLlama-1.1B, which were instructional-tuned using LoRA, on an iPhone 15 Pro Max with 8GB memory capacity. Since the SOTA LLM HealthAlpaca-lora-7b [23] did not release its checkpoint, we compared the on-device performance of selected SLMs against the baseline Llama-2-7b (the backbone of HealthAlpaca-lora-7b) using PMData to evaluate deployment efficiency. For fair comparison, we random select a total of ten samples from PMData for both Llama-2-7b, Phi-3-mini-4k and TinyLlama-1.1 to evaluate latency and hardware utilization.

As shown in Table 6, the efficiency results of the two instruction-tuned SLMs on PMData demonstrate that SLMs preserve their latency and memory advantages over Llama-2-7b. Both TinyLlama-1.1B and Phi-3-mini-4k outperform Llama-2-7b in latency and throughput. Specifically, Phi-3-mini-4k achieves a $4.6\times$ faster time-to-first-token (TTFT) and $29\times$ faster output evaluation time (OET), with gains of over $+350\%$ in both input tokens per second (ITPS) and output tokens per second (OTPS). TinyLlama-1.1B shows even larger margins, with $21\times$ faster TTFT, $79\times$ faster OET, and more than $+2,000\%$ ITPS compared to Llama-2-7b. The memory footprint of the SLMs is also much smaller. Specifically, Phi-3-mini-4k uses 9% less RAM, and TinyLlama-1.1B uses 28% less than Llama-2-7b. Comparing the two SLMs, Phi-3-mini-4k offers moderate efficiency gains in some metrics but is consistently slower than TinyLlama-1.1B.

Overall, SLMs achieve substantial reductions in both input processing latency and generation latency, making them as ideal and practical solutions for resource-constrained mobile health applications.

6 Conclusion and Future Work

In this paper, we introduce HealthSLM-Bench, a comprehensive benchmark designed to systematically evaluate SOTA SLMs on healthcare monitoring tasks under zero-shot, few-shot, and instruction-tuning scenarios. Furthermore, we assess the efficiency of these models following instruction-tuning through on-device deployment experiments. Our study shows that SLMs can match or even surpass much larger LLMs after adapted with few-shot and instructional tuning while delivering superior efficiency gain, making them practical for real-time on-device deployment. At the same time, we also identified their limitations in few-shot prompting and restricted effectiveness in instruction tuning, particularly under class-imbalanced datasets. Both limitations point to several promising directions for future work. The first is to investigate the underlying causes of the few-shot anomaly and explore robust prompt design to prevent collapse. Another direction is to explore imbalance-aware training approaches, for example by adjusting loss weighting or augmenting minority-class samples, to reduce class bias during SLM fine-tuning. Additionally, leveraging adaptive techniques such as test-time adaptation [59] could further strengthen SLM generalisation in health applications. Taken together, our benchmark establishes SLMs as a promising yet imperfect solution for efficient and privacy-preserving healthcare applications, motivating further exploration to address these challenges.

References

[1] Cecilia Dinh-Le, Rebecca Chuang, Sonia Chokshi, and Devin Mann. Wearable health technology and electronic health record integration: scoping review and future directions. *Journal of Medical Internet Research*, 21(9):e12861, 2019.

[2] Nhat Pham, Hong Jia, Minh Tran, Tuan Dinh, Nam Bui, Young Kwon, Dong Ma, Phuc Nguyen, Cecilia Mascolo, and Tam Vu. Pros: an efficient pattern-driven compressive sensing framework for low-power biopotential-based wearables with on-chip intelligence. In *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*, pages 661–675, 2022.

[3] Hong Jia, Young D Kwon, Dong Mat, Nhat Pham, Lorena Qendro, Tam Vu, and Cecilia Mascolo. Ur2m: Uncertainty and resource-aware event detection on microcontrollers. In *2024 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 1–10. IEEE, 2024.

[4] Yu Wu, Dimitris Spathis, Hong Jia, Ignacio Perez-Pozuelo, Tomas I Gonzales, Soren Brage, Nicholas Wareham, and Cecilia Mascolo. Udama: Unsupervised domain adaptation through multi-discriminator adversarial training with noisy labels improves cardio-fitness prediction. In *Machine Learning for Healthcare Conference*, pages 863–883. PMLR, 2023.

[5] Lucas Gabrielli et al. Ai on the pulse: Integrating wearable sensors, ambient intelligence, and large language models for continuous health monitoring. *arXiv preprint arXiv:2508.03436*, 2025. Accessed: 2025-09-04.

[6] Yassir Ghadi et al. Wearable eeg and ai for real-time personalized health monitoring and intervention. *Journal of Cloud Computing*, 14(1):1–15, 2025.

[7] Emilio Ferrara. A survey on large language models for sensor-based human activity recognition and health monitoring. *Sensors*, 24(15):5045, 2024.

[8] Muhammad Imran et al. Llasa: Multimodal large language models for interpreting human activity from inertial sensor data. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 2024.

[9] Amir Khasentino et al. Personal health llms: Towards context-aware and adaptive health monitoring from wearable sensor data. *Nature Medicine*, 2025. In press.

[10] Anish Das. Security and privacy challenges of large language models. *ACM Computing Surveys*, 58(2):1–38, 2025. doi: 10.1145/3712001. URL `https://dl.acm.org/doi/10.1145/3712001`.

[11] H. Li, Y. Chen, J. Luo, Y. Kang, X. Zhang, Q. Hu, C. Chan, and Y. Song. Privacy in large language models: Attacks, defences and future directions. *arXiv*, 2024. URL `https://arxiv.org/pdf/2310.10383`. Available: https://arxiv.org/pdf/2310.10383.

[12] Tianqi Xu, Wei Zhang, Chen Li, and Yifan Wang. Camel: Energy-aware llm inference on resource-constrained devices. *arXiv preprint arXiv:2508.09173*, 2025.

[13] Hui Wang, Qiang Liu, and Mei Chen. Large language models on edge devices: Challenges and opportunities for intelligent data analysis. *Frontiers in Computer Science*, 7:1538277, 2025.

[14] Microsoft. Phi-3 technical report: A highly capable language model locally on your phone. Technical Report, 2024. URL `https://arxiv.org/pdf/2404.14219v4`.

[15] Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. Tinyllama: An open-source small language model. `https://github.com/jzhang38/TinyLlama`, 2024. TinyLlama achieves approximately 70-80% of LLaMA2's performance on commonsense reasoning tasks such as HellaSwag and ARC-Challenge.

[16] Qwen. Qwen2-1.5b: A new series of large language models, 2024. URL `https://huggingface.co/Qwen/Qwen2-1.5B`. Accessed: 2024-09-01.

[17] Gemma Team and Google DeepMind. Gemma 2: Improving open language models at a practical size. Technical report, Google DeepMind, 2024. For full author list, see Contributions and Acknowledgments section. Correspondence to `gemma-2-report@google.com`.

[18] Xin Wang, Ting Dang, Vassilis Kostakos, and Hong Jia. Efficient and personalized mobile health event prediction via small language models. In *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking*, ACM MobiCom '24, page 2353–2358, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400704895. doi: 10.1145/3636534.3698123. URL `https://doi.org/10.1145/3636534.3698123`.

[19] Brandon Ballinger, Joy Hsieh, Avesh Singh, Nitish Sohoni, Jae Wang, Fangfei Li, Amit Sharma, Akshay Sharma, Gregory M. Marcus, Suchi Saria, and Daniel Halperin. Deepheart: Semi-supervised sequence learning for cardiovascular risk prediction. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI)*, pages 2079–2086. AAAI Press, 2018. URL `https://ojs.aaai.org/index.php/AAAI/article/view/11532`.

[20] Kristján Hallgrímsson, Tom Goodwin, Sujit Ghosh, Peter Bühlmann, Christian Mathys, Vincent Lefort, Ara Darzi, Lionel Tarassenko, and David A. Clifton. Learning individualized cardiovascular responses from large-scale wearable sensors data. In *Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI)*, pages 941–948. AAAI Press, 2019. URL `https://ojs.aaai.org/index.php/AAAI/article/view/3834`.

[21] T. Mullick, A. Radovic, S. Shaaban, and A. Doryab. Predicting depression in adolescents using mobile and wearable sensors: Multimodal machine learning–based exploratory study. *JMIR Formative Research*, 6(6):e35807, 2022. doi: 10.2196/35807. URL `https://formative.jmir.org/2022/6/e35807`.

[22] Sebastian Kasl, Nathanael S. Holtzman, Md. Masudul Islam Shandhi, Tanishq Gupta, Jiang Kuang, Gregory D. Hager, Shawn S. Lam, and Suchi Saria. On the generalizability of wearable-based machine learning for respiratory virus detection. In *Proceedings of the 9th Machine Learning for Healthcare Conference (MLHC)*, volume 248 of *Proceedings of Machine Learning Research*, pages 437–461. PMLR, 2024. URL `https://proceedings.mlr.press/v248/kasl24a.html`.

[23] Yubin Kim, Xuhai Xu, Daniel McDuff, Cynthia Breazeal, and Hae Won Park. Health-llm: Large language models for health prediction via wearable sensor data. In Tom Pollard, Edward Choi, Pankhuri Singhal, Michael Hughes, Elena Sizikova, Bobak Mortazavi, Irene Chen, Fei Wang, Tasmie Sarker, Matthew McDermott, and Marzyeh Ghassemi, editors, *Proceedings of the fifth Conference on Health, Inference, and Learning*, volume 248 of *Proceedings of Machine Learning Research*, pages 522–539. PMLR, 27–28 Jun 2024. URL `https://proceedings.mlr.press/v248/kim24b.html`.

[24] Yongquan Hu, Shuning Zhang, Ting Dang, Hong Jia, Flora D Salim, Wen Hu, and Aaron J Quigley. Exploring large-scale language models to evaluate eeg-based multimodal data for mental health. *arXiv preprint arXiv:2408.07313*, 2024.

[25] Xuhai Xu, Bingsheng Yao, Yuanzhe Dong, Saadia Gabriel, Hong Yu, James Hendler, Marzyeh Ghassemi, Anind K. Dey, and Dakuo Wang. Mental-llm: Leveraging large language models for mental health prediction via online text data. In *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, volume 8, pages Article 32, 32 pages. Association for Computing Machinery, March 2024. doi: 10.1145/3643540.

[26] Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023.

[27] Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. Capabilities of gpt-4 on medical challenge problems. *arXiv preprint arXiv:2303.13375*, 2023.

[28] Cathy Mengying Fang, Valdemar Danry, Nathan Whitmore, Andria Bao, Andrew Hutchison, Cayden Pierce, and Pattie Maes. Physiollm: Supporting personalized health insights with wearables and large language models. *arXiv preprint arXiv:2406.19283*, 2024.

[29] Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, et al. Minicpm: Unveiling the potential of small language models with scalable training strategies. *arXiv preprint arXiv:2404.06395*, 2024.

[30] Zhenyan Lu, Xiang Li, Dongqi Cai, Rongjie Yi, Fangming Liu, Wei Liu, Jian Luan, Xiwen Zhang, Nicholas D. Lane, and Mengwei Xu. Demystifying small language models for edge deployment. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14747–14764, Vienna, Austria, 2025.

[31] Rithesh Murthy, Liangwei Yang, Juntao Tan, Tulika Manoj Awalgaonkar, Yilun Zhou, Shelby Heinecke, Sachin Desai, Jason Wu, Ran Xu, Sarah Tan, et al. Mobileaibench: Benchmarking llms and lmms for on-device use cases. 2023.

[32] Microsoft. Phi-3-mini-4k-instruct: A lightweight, state-of-the-art open model, 2024. URL `https://phi.microsoft.com/phi-3-mini-4k-instruct`. Accessed: 2024-09-01.

[33] TinyLlama. Tinyllama-1.1b-chat-v1.0: A compact llama model with 1.1b parameters, 2024. URL `https://github.com/jzhang38/TinyLlama`. Accessed: 2024-09-01.

[34] Google. Gemma 2: A lightweight, state-of-the-art open model family, 2024. URL `https://huggingface.co/google/gemma-2-2b`. Accessed: 2024-09-01.

[35] HuggingFaceTB. Smollm-1.7b-instruct: A series of small language models, 2024. URL `https://huggingface.co/HuggingFaceTB/SmolLM-1.7B-Instruct`. Accessed: 2024-09-01.

[36] Meta AI. Llama 3.2 model card. Hugging Face, 2024. Release date: September 25, 2024. Includes lightweight text-only (1 B, 3 B) and multimodal (11 B, 90 B) models.

[37] Vajira Thambawita, Steven Alexander Hicks, Hanna Borgli, Håkon Kvale Stensland, Debesh Jha, Martin Kristoffer Svensen, Svein-Arne Pettersen, Dag Johansen, Håvard Dagenborg Johansen, Susann Dahl Pettersen, Simon Nordvang, Sigurd Pedersen, Anders Gjerdrum, Tor-Morten Grønli, Per Morten Fredriksen, Ragnhild Eg, Kjeld Hansen, Siri Fagernes, Christine Claudi, Andreas Biørn-Hansen, Duc Tien Dang Nguyen, Tomas Kupka, Hugo Lewi Hammer, Ramesh Jain, Michael Alexander Riegler, and Pål Halvorsen. Pmdata: A sports logging dataset. In *Proceedings of the 11th ACM Multimedia Systems Conference*, MMSys '20, page 231–236, New York, NY, USA, 2020. Association for Computing Machinery. doi: 10.1145/3339825.3394926.

[38] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared 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 M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020. URL `https://arxiv.org/abs/2005.14165`.

[39] John Wei, Michael Bosma, Dale Schuurmans, and et al. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of NeurIPS 2022*, 2022. URL `https://arxiv.org/abs/2201.11903`.

[40] Xinyang Wang, Hinrich Schütze, and et al. Self-consistency improves chain-of-thought reasoning in language models. In *Proceedings of NeurIPS 2022*, 2022. URL `https://arxiv.org/abs/2203.11171`.

[41] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022. URL `https://iclr.cc/virtual/2022/oral/6255`.

[42] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori Hashimoto. Stanford alpaca: An instruction-following llama model. `https://crfm.stanford.edu/2023/03/13/alpaca.html`, 2023. Accessed: 2025-04-27.

[43] Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Chandu, David Wadden, Kelsey MacMillan, Noah A. Smith, Iz Beltagy, and Hannaneh Hajishirzi. How far can camels go? exploring the state of instruction tuning on open resources. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/ec6413875e4ab08d7bc4d8e225263398-Abstract-Datasets_and_Benchmarks.html.

[44] Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Sam Shah, Ali Ghodsi, Patrick Wendell, and Matei Zaharia. Dolly: The first truly open-source instruction-tuned model. Databricks blog, 2023. Fine-tuned on the Stanford Alpaca dataset.

[45] Vicuna Team. Vicuna: An open-source chatbot impressing gpt-4 with 90LMSYS blog post, 2023. Instruction tuning inspired by Alpaca's methodology.

[46] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *Proceedings of the Tenth International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

[47] Xuhai Xu, Han Zhang, Yasaman Sefidgar, Yiyi Ren, Xin Liu, Woosuk Seo, Jennifer Brown, Kevin Kuehn, Mike Merrill, Paula Nurius, Shwetak Patel, Tim Althoff, Margaret E. Morris, Eve Riskin, Jennifer Mankoff, and Anind K. Dey. Globem dataset: Multi-year datasets for longitudinal human behavior modeling generalization, 2023. URL https://arxiv.org/abs/2211.02733.

[48] Daniel Fuller. Replication data for: Using machine learning methods to predict physical activity types with apple watch and fitbit data using indirect calorimetry as the criterion, 2020. URL https://doi.org/10.7910/DVN/ZS2Z2J. Accessed: 2025-04-29.

[49] Fitbit Inc. Fitbit versa 2: Health & fitness smartwatch. https://www.fitbit.com/global/us/products/smartwatches/versa2, 2019. Accessed: 2025-08-30.

[50] Activinsights Ltd. Geneactiv: Raw data accelerometer for physical activity and sleep research. https://www.activinsights.com/products/geneactiv/, 2015. Accessed: 2025-08-30.

[51] Apple Inc. Apple watch series 2. https://support.apple.com/kb/SP745, 2016. Accessed: 2025-08-30.

[52] Fitbit Inc. Fitbit charge 2: Heart rate + fitness wristband. https://www.fitbit.com/global/us/products/trackers/charge2, 2016. Accessed: 2025-08-30.

[53] Microsoft Corporation. Fine-tune small language model (slm) phi-3 using azure machine learning. https://techcommunity.microsoft.com/t5/ai-machine-learning-blog/fine-tune-small-language-model-slm-phi-3-using-azure-machine/ba-p/4130399, 2024. Accessed: 2025-06-09.

[54] LLaMA Open Source Team. Tinyllama: A distilled version of llama for efficient language tasks. https://github.com/TinyLlama, 2024. Highlights the use of knowledge distillation for TinyLlama-1.1B derived from LLaMA-13B.

[55] Hong Jia, Shiya Fu, Feng Xia, Vassilis Kostakos, and Ting Dang. Beyond scale: Small language models are comparable to gpt-4 in mental health understanding. *arXiv preprint arXiv:2507.08031*, 2025.

[56] Hugging Face. Gguf, 2023. URL https://huggingface.co/docs/hub/en/gguf. Accessed: 2025-09-05.

[57] Ggerganov. Ggerganov/llama.cpp: Llm inference in c/c++. URL https://github.com/ggerganov/llama.cpp.

[58] Georgi Gerganov and community. llama.cpp: Efficient llm inference in c/c++. https://github.com/ggml-org/llama.cpp, 2023. Released March 10, 2023; accessed 2025-09-06.

[59] Hong Jia, Young D. Kwon, Alessio Orsino, Ting Dang, Domenico Talia, and Cecilia Mascolo. Tinytta: Efficient test-time adaptation via early-exit ensembles on edge devices. In *Advances in Neural Information Processing Systems*, volume 37, pages 43274–43299, 2024.

[60] Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL `https://arxiv.org/abs/2412.15115`.

[61] Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[62] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2009.

Appendix

A Implementation Details

We fine-tune our SLMs on a NVIDIA A100 80GB GPUs with a batch size of 128 with 3 number of epochs for the purpose of fine-tuning, with Adam optimizer and a learning rate as 5e-5 (cosine learning rate scheduler and dynamic warmup steps of 5% of dataset size). It took about 7 hours for 9 SLMs in 3 epochs of training with the default training setting. We adopt greedy decoding method with sampling set to False. We utilize the same prompt of zero-shot for LoRA tuned SLMs inference. To ensure re-productiveness, we employ the greedy decoding strategy to make the output prediction deterministic. While most language models default to sampling-based decoding (e.g., top-k, top-p), we explicitly disabled these strategies to maintain reproducibility across runs. To better simulate edge-device conditions, where computational resources are constrained, we capped the maximum number of generated tokens at 30. Generation stops once this limit is reached, even if the answer is incomplete, which balances efficiency and response quality. The codes and fine-tuned models will be made publicly available upon the release of the camera-ready version of this paper.

B Additional Experiments

To solve the class-imbalance issue we observed under instructional-tuning (LoRA), we employed a data augmentation method (oversampling) to balance minority classes in the training data. Specifically, samples from underrepresented classes were randomly duplicated until all classes reached the size of the majority class. In the fine-tuning setup, we specifically apply label-grouping prepossessing strategy, which groups samples by their labels instead of arranging them in a random order. This grouping ensures that each mini-batch contains samples with a consistent label, stabilizing gradient updates and improving class representation during fine-tuning. For better assess the effectiveness of class-imbalance mitigation, model performance was further evaluated using the Macro F1-score, in addition to Accuracy and MAE.

Table 7: LoRA performance of SLMs fine-tuned on augmented datasets (PMData & GLOBEM) compared to those fine-tuned on the original dataset (with label-grouping). The best results are shown in **bold**, and the second-best results are underlined. “OR” denotes fine-tuning with the original dataset, while “OS” denotes fine-tuning on the oversampled datasets.

	Model	**STRS (↓)**	**READ (↓)**	**FATG (↑)**	**SQ (↓)**	**ANX (↓)**	**DEP (↓)**
LLMs	HealthAlpaca-lora-7b	0.53	**1.40**	50.0	0.58	**0.62**	0.51
	HealthAlpaca-lora-13b	**0.34**	1.56	**54.8**	**0.39**	1.04	**0.67**
	Mean	0.44	1.48	52.4	0.49	0.83	0.59
SLMs (OR)	gemma-2-2b-it	-	-	-	-	1.064	0.576
	Phi-3-mini-4k	0.395	1.997	62.5	0.468	**0.806**	0.545
	SmolLM-1.7B	0.526	1.753	12.4	0.900	0.890	0.542
	Qwen2-1.5B	**0.388**	**1.304**	63.2	0.462	0.940	0.933
	TinyLlama-1.1B	0.398	1.311	**63.5**	0.475	0.876	0.555
	Llama-3.2-1B	0.415	2.003	52.8	**0.448**	0.866	**0.535**
	Llama-3.2-3B	0.501	1.525	38.1	0.495	0.882	**0.535**
	Phi-3.5-mini	0.445	1.642	62.2	0.957	0.833	0.766
	Qwen2.5-1.5B	-	1.354	49.1	0.720	-	-
	Mean	0.438	1.611	50.5	0.616	0.895	0.623
SLMs (OS)	gemma-2-2b-it	**0.48**	1.73	41.70	0.48	1.98	1.94
	Phi-3-mini-4k	1.46	3.70	15.4	1.79	1.01	0.92
	SmolLM-1.7B	1.03	1.85	21.4	1.21	1.68	1.95
	Qwen2-1.5B	1.68	3.32	0.7	0.80	2.82	2.84
	TinyLlama-1.1B	0.61	4.80	**58.9**	0.64	3.48	1.15
	Llama-3.2-1B	1.01	7.02	31.8	0.90	**0.84**	**0.61**
	Llama-3.2-3B	0.75	3.31	29.8	**0.48**	1.03	1.25
	Phi-3.5-mini	0.55	2.45	50.8	0.73	0.92	0.84
	Qwen2.5-1.5B	1.08	**1.55**	4.4	1.00	1.46	1.72
	Mean	0.96	3.30	28.3	0.89	1.69	1.47

Table 8: LoRA Performance of 9 SLMs fine-tuned with augmented dataset compared to finetuned with original. All tasks are evaluated by Macro F1-score, which higher value indicates better performance. Best results in **bold**, second best are underlined. “-” denotes invalid or unreasonable responses generated by models, while “N/A” indicates the result is not available in original paper.

	Model	**STRS (↑)**	**READ (↑)**	**FATG (↑)**	**SQ (↑)**	**ANX (↑)**	**DEP (↑)**
LLMs	HealthAlpaca-lora-7b	N/A	N/A	19.0	N/A	N/A	N/A
	HealthAlpaca-lora-13b	N/A	N/A	**45.0**	N/A	N/A	N/A
SLMs (OR)	gemma-2-2b-it	-	-	-	-	14.8	15.1
	Phi-3-mini-4k	12.9	2.4	15.4	15.4	13.2	**17.8**
	SmolLM-1.7B	12.6	5.0	6.0	11.6	**19.6**	16.9
	Qwen2-1.5B	**14.3**	4.0	15.5	18.1	12.0	14.3
	TinyLlama-1.1B	13.7	3.9	17.5	14.2	16.0	17.2
	Llama-3.2-1B	12.7	**7.5**	**18.4**	**30.2**	14.0	15.3
	Llama-3.2-3B	**14.3**	6.9	17.1	21.8	12.3	15.3
	Phi-3.5-mini	12.4	5.1	15.4	8.9	15.2	17.4
	Qwen2.5-1.5B	-	6.2	13.6	23.2	-	-
	Mean	13.3	5.1	14.9	17.9	14.6	16.2
SLMs (OS)	gemma-2-2b-it	**24.8**	**6.3**	17.5	**32.6**	7.4	8.7
	Phi-3-mini-4k	8.7	3.3	6.7	7.0	15.5	12.0
	SmolLM-1.7B	16.7	5.0	16.1	12.0	10.8	7.5
	Qwen2-1.5B	1.4	2.9	0.3	19.6	6.4	4.0
	TinyLlama-1.1B	16.3	1.3	19.4	22.7	5.9	9.9
	Llama-3.2-1B	6.9	3.6	12.3	18.1	**21.8**	**19.5**
	Llama-3.2-3B	18.8	5.3	15.0	19.0	17.4	17.6
	Phi-3.5-mini	14.8	2.1	**19.6**	29.9	15.6	11.6
	Qwen2.5-1.5B	4.3	3.8	2.0	8.0	7.7	2.9
	Mean	12.5	3.7	12.1	18.8	12.1	10.4

Performance comparison (Accuracy, MAE). As shown in Table 7, SLM performance generally declined when fine-tuned on oversampled datasets (OS) compared with the original datasets (OR). For instance, the mean error of STRS, READ, SQ, ANX, and DEP increased from 0.438, 1.611, 0.616, 0.805, and 0.623 to 0.961, 3.303, 28.3, 0.892, and 1.691, respectively, while the mean accuracy of FATG decreased from 48.0 to 28.3. When comparing the best results, SLMs fine-tuned on OS also showed a similar decline, though the degradation was less pronounced than in the mean performance. Specifically, the lowest MAE of STRS and READ increased from 0.388 and 1.304 to 0.482 and 1.549, respectively (a similar trend was observed for SQ, ANX, and DEP), while the highest accuracy of FATG dropped from 63.5 to 58.9. For some models, such as Qwen2-1.5B and Qwen2.5-1.5B, exhibited the most severe degradation, with FATG accuracies of only 0.7% and 4.4%, compared to 63.2% and 49.1% fine-tuned under the original datasets. These observations indicate that strategies like oversampling tends to amplify existing noise in minority classes, leading to reduced generalization under instruction-tuning.

Analysis on distribution inspection. Upon inspection, as shown in Figure 7, SLMs fine-tuned on oversampled datasets did demonstrate improved coverage of minority classes compared to those fine-tuned on the original datasets. Models, such as *gemma-2-2b-it* and *phi-3.5-mini*, began predicting a broader range of labels on STRS, FATG, and SQ, bringing their frequency distributions closer to the true distribution, where labels 2–4 are the most representative. Similarly, on the GLOBEM dataset (Figure 8), the best-performing models (*Llama-3.2-1B* and *Phi-3.5-mini*) expanded their predictions beyond dominant labels (0, 1) to include rare labels such as 3 and 4. This improvement indicates that oversampling mitigates single-class dominance and enables SLMs to capture a more balanced label distribution. However, residual mismatches still persist. In particular, *Qwen2.5-1.5B* tends to over-predict label 4 on PMData tasks and label 2 on GLOBEM tasks, suggesting that oversampling amplifies noise in underrepresented classes and thereby explains the degradation in performance observed earlier.

Performance comparison (Macro F1-Score). As a more reliable measure of performance on these class-imbalanced datasets, the Macro F1-scores reported in Table 8 further support our earlier observations on the predicted distributions. In particular, SLMs fine-tuned on oversampled datasets achieved higher F1-scores than those trained on the original datasets across most tasks, consistent with their predicted distributions being closer to the true label distribution. For example, on PMData tasks such as STRS, FATG, and SQ, the top-performing models obtained F1-scores of 24.8, 19.6, and 32.6 trained on oversampled data, compared to 14.3, 18.4, and 30.2 trained on original data. Similarly, on GLOBEM tasks (ANX, DEP), SLMs trained with oversampled datasets achieved the best F1-scores of 21.8 and 19.5, surpassing 19.6 and 17.8 trained with original datasets. These results confirm that oversampling enhances class diversity and balances predictions, even though overall accuracy and MAE slightly decline. However, as oversampling simply duplicates existing data, it cannot introduce new variability, which may cause models to overfit to minority patterns and limit generalization on unseen data.

In sum, oversampling improves class balance and prediction diversity for SLMs, as reflected by higher Macro F1-scores and broader label coverage. However, these gains come at the cost of reduced performance, likely due to overfitting on duplicated samples and weaker generalization. These observations highlight oversampling as an effective yet imperfect strategy for mitigating class imbalance in SLM predictions, suggesting that future work should explore more advanced augmentation methods that maintain its stability while increasing data diversity.

C Small Language Models

We selected 9 most state-of-the-art SLMs between 1 to 4B from top-tier tech companies. The details of each SLMs are listed below:

- **Phi-3-mini-4k-Instruct** [32]: Microsoft's smallest model in the Phi-3 family. It has 3.8 billion parameters, trained on a combination of synthetic data and selected publicly available website data, with an emphasis on high-quality and reasoning-dense properties.
- **Phi-3.5-mini-Instruct** [32]: A upgrade version of phi-3-mini-4k-instruct. It is built in the same architecture and dataset upon phi-3, but trained with a focus on reasoning dense data for better instruction alignment and multi-step reasoning.
- **TinyLlama-1.1B-Chat-v1.0** [33]: Distilled version of Llama 2. It uses the same architecture and tokenizer as LLaMA but is compact with 1.1 billion parameters. It was fine-tuned on the UltraChat dataset (contains field-cross synthetic dialogues generated by ChatGPT), making it compatible with a wide range of tasks.
- **Gemma2-2B-it** [34]: Google's SOTA open-source model, built on the same research and technology as the Gemini models but scaled down to 2 billion parameters. It is well-suited for text generation tasks such as question answering, summarization, and reasoning.
- **SmolLM-1.7B-Instruct** [35]: HuggingFace's flagship model, it has 1.7 billion parameters and is trained on SmolLM-Corpus which consists of synthetic textbooks, stories, and educational Python and web samples.
- **Qwen2-1.5B-Instruct** [16]: Ailibaba's state-of-the-art SLM in Qwen2 family. It has only 1.5 billion parameters and is trained on diverse instruction-followed tasks. The included coding and mathematics data for training makes it perform well in coding and quantitative reasoning tasks.
- **Qwen2.5-1.5B-Instruct** [60]: An upgraded version of Qwen2. It is built on the same dataset and architecture, but places greater emphasis on coding and mathematics tasks, making it more optimized for reasoning and math.
- **Llama-3.2-1B-Instruct** [36]: Meta-llama's state-of-the-art SLM. It shares the identical architecture and pre-trained datasets upon Llama3, but is compressed to 1B parameters.
- **Llama-3.2-3B-Instruct** [36]: 3B version of Llama-3.2-1B-Instruct.

D Task categorization and Label Distribution

D.1 PMData

- Stress (STRS): Estimation of an individual's stress level based on physiological data and self-reported measures. (0-5, Classification)
- Readiness (READ): Assessment of an individual's readiness for physical activity/exercise. (0-10, Classification)

- Fatigue (FATG): Monitoring of signs of tiredness or exhaustion based on sports and life-log data in the last 14 days. (1-5, Classification)

- Sleep Quality (SQ): Estimation of an individual's sleep quality. (1-5, Classification)

All tasks is assessed with factors including total sleep time, Steps, mood and other sports data like Burned Calories and Resting Heart Rate over a continuous 14-day period. In terms of range, most tasks are evaluated on a scale of 1-5 or 0-5. A score of 3 represents a normal condition, and 1-2 are scores below normal states, and 4-5 are scores above normal states. For the task of readiness, the scale ranges from 0 to 10, where 0 reflects no readiness for physical activity, and 10 indicates high preparation for exercise.

The **label distribution** for each task in this dataset is shown as below:

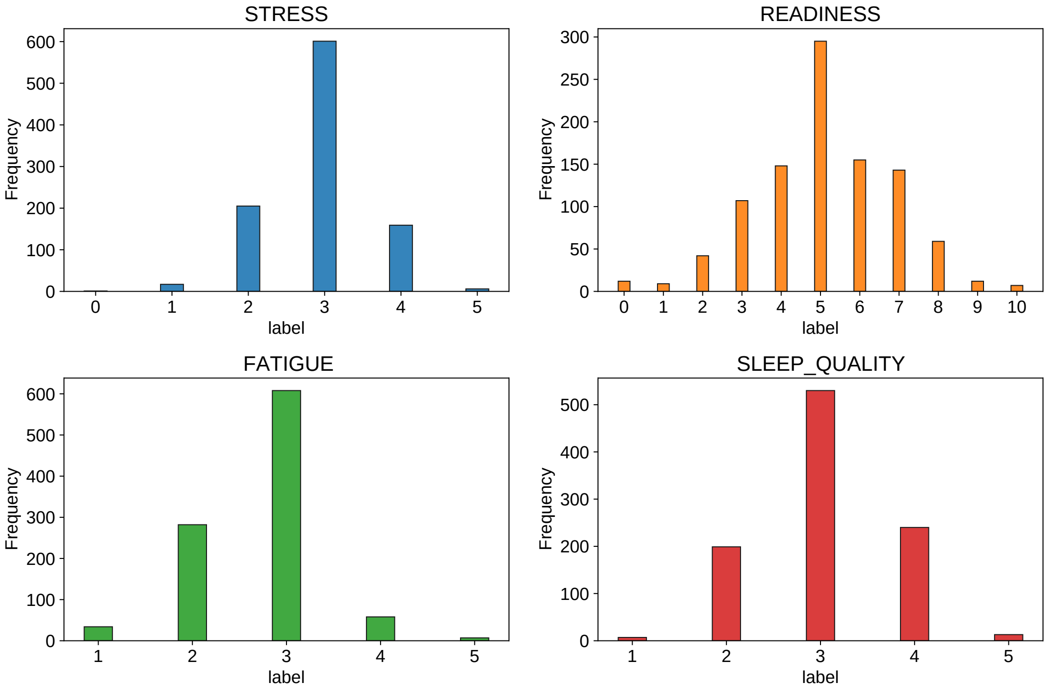

Figure 1: The label distribution of the four tasks in PMData

D.2 GLOBEM

- Depression (DEP): estimation of a depression score that analyzes patterns in user's sleeping behavior and activity levels. (0–4, Classification)

- Anxiety (ANX): estimation of an anxiety score that relies on behavioral markers such as irregular sleep patterns or heightened physiological responses, e.g. increased heart rate, reduced activity levels, and increased sleep disturbances (0–4, Classification)

Both the two tasks are assessed on the average of daily steps, sleep efficiency, duration the user stayed in bed after waking up, duration the user spent to sleep, duration the user stayed awake but still in bed, and duration the user spent to fall asleep in the last 14 days. A value of 0 implies the disorder is not present, while a value of 4 indicates severe disorder. Any values between 0 and 4 denote their severity accordingly, such as a value of 1 indicates mild disorder, 2 refers to moderate, and 3 refers to Moderately Severe.

The **label distribution** for each task in this dataset is shown below:

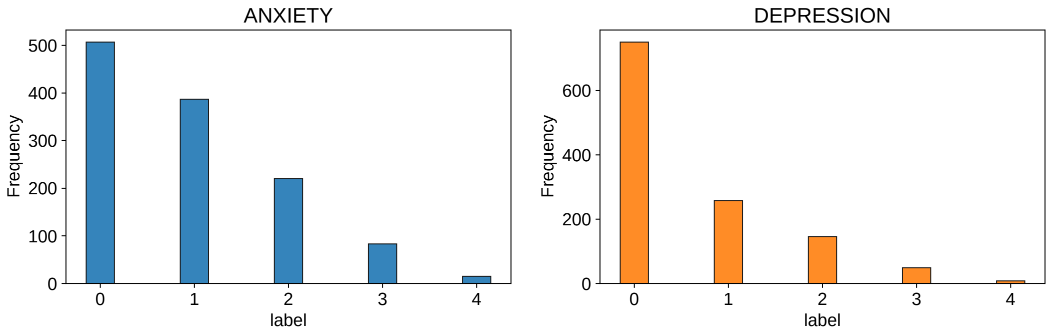

Figure 2: The label distribution of the two tasks in GLOBEM

D.3 AW_FB

- Activity (ACT): estimation of individual's activity intensity type based on sensor data. (0-5, Classification)
- Calories (CAL): estimation of burned calories that are expended by an individual during physical activities. (no constraint, Regression)

Activity is predicted by Steps, Burned Calories, and Heart Rate obtained during an activity period. This label ranges from 0 to 5, corresponding to Self Pace Walk, Sitting, Lying, Running 7 METs, Running 5 METs, and Running 3 METs respectively. Calories are calculated based on Steps, Heart Rate, Duration, Activity Type, and MET Value, where a higher value indicates greater energy expenditure.

The **label distribution** for each task in this dataset is shown below:

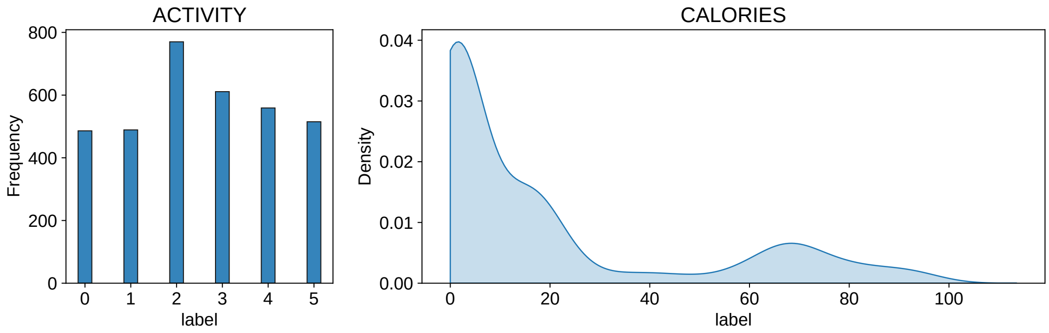

Figure 3: The label distribution of the two tasks in AW_FB

E Evaluation Metrics

E.1 Performance Evaluation

For SLMs performance evaluation, Mean Absolute Error (MAE) and Accuracy are utilized to assess model prediction performance on health event prediction.

Accuracy (%) [61] measures the proportion of correctly predicted instances out of all instances. It provides an overview of whether a model performed well overall, with higher values indicating better performance. However, accuracy does not capture the severity or magnitude of errors in misclassified cases, as all errors are treated equally.

Mean Absolute Error (MAE) [62] quantifies the average magnitude of prediction errors by computing the absolute difference between predicted and actual values. Lower MAE indicates better alignment with ground truth. Unlike Accuracy, which only reflects correctness, MAE distinguishes between small and large errors. For example, predicting "3" when the true label is "4" yields an error of 1, while predicting "3" when the true label is "10" yields an error of 7. Thus, MAE captures not only whether predictions are correct but also how close incorrect predictions are to the true values.

In health event prediction, we used both Accuracy and MAE to provide complementary insights. For instance, models that achieve slightly lower accuracy but maintain consistently low MAE may be preferable, as they deliver more reliable outputs than those with higher accuracy but large error magnitudes.

E.2 Efficiency and Utilization Evaluation

To further evaluate the efficiency and the actual latency in real cases, all state-of-the-art (SOTA) SLMs that show strong promise will be deployed in processing healthcare field data on a real iPhone 15 Pro Max. To better demonstrate the importance of efficiency on mobile devices, the widely used LLM, Llama 2, is selected and serves as a comparison to the fine-tuned SLMs. The following metrics suggested by MobileAIBench [31] are adopted to evaluate both efficiency and utilization:

- **Time-to-First-Token** (TTFT, sec): TTFT is defined by the time of the first token generated to respond to the prompt. It primarily assesses latency, where a lower TTFT indicates a faster response time, allowing users to perceive quicker feedback from the SLM.

- **Input Token Per Second** (ITPS, tokens/sec): ITPS is defined by the number of input tokens being processed per second, which refers to how fast the model can read and understand the prompts.

- **Output Token Per Second** (OTPS, tokens/sec): OTPS is defined by the number of tokens produced per second after starting to produce tokens, which refers to how fast the model can produce the answer, and access the inference speed. A higher value indicates higher efficiency.

- **Output Evaluation Time** (OET, sec): The time model takes to complete a response - assess the overall efficiency of generating an entire response. A lower value indicates higher efficiency.

- **Total Time**: The total time it takes to produce a complete response after receiving a prompt is a comprehensive efficiency metric for how long a model takes to complete a given task from start to finish. A lower value indicates higher efficiency.

- **CPU (%)**: An amount of computational resources used in the inference process.

- **RAM (GB)**: An amount of memory needed to run a model during the inference process.

During batch evaluations, latency metrics such as TTFT, ITPS, OTPS, OET, and Total Time are calculated as the average time spent or average token processed/generated over a sample size of N (we used 10). CPU utilization is measured by the average load per second during inference, while RAM usage is reported as the maximum memory allocated to the device when the model is running.

F SLMs Prediction Distribution

F.1 Few-shot Distribution

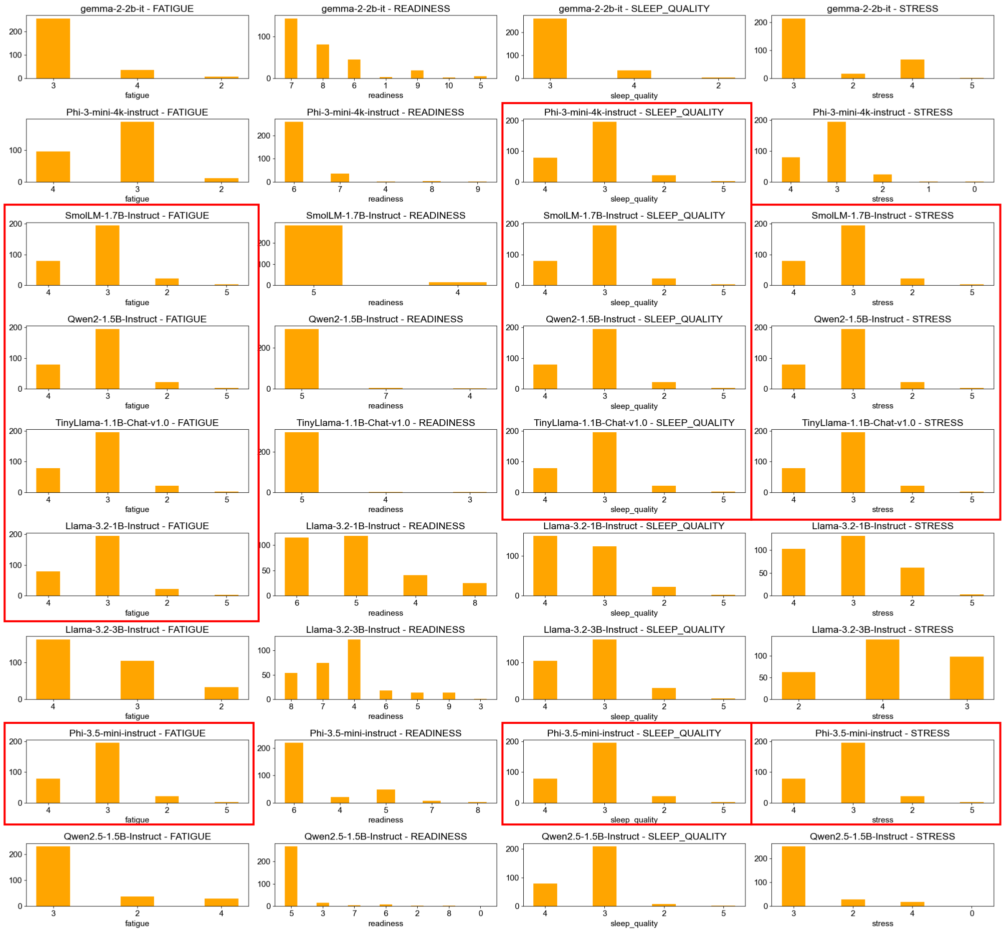

Figure 4: Distribution of predictions for the four tasks in PMData under FS setting

F.2 Instructional tuning (LoRA) Distribution

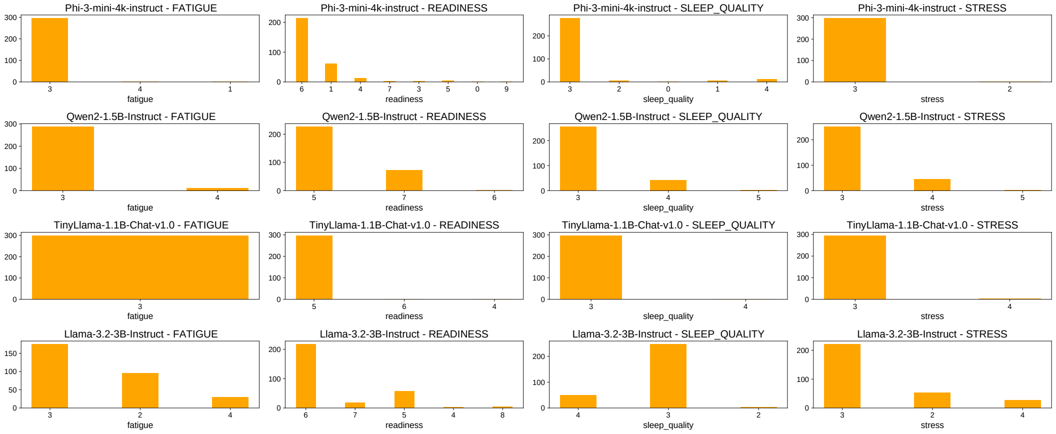

Figure 5: Distribution of predictions for the four tasks in PMData

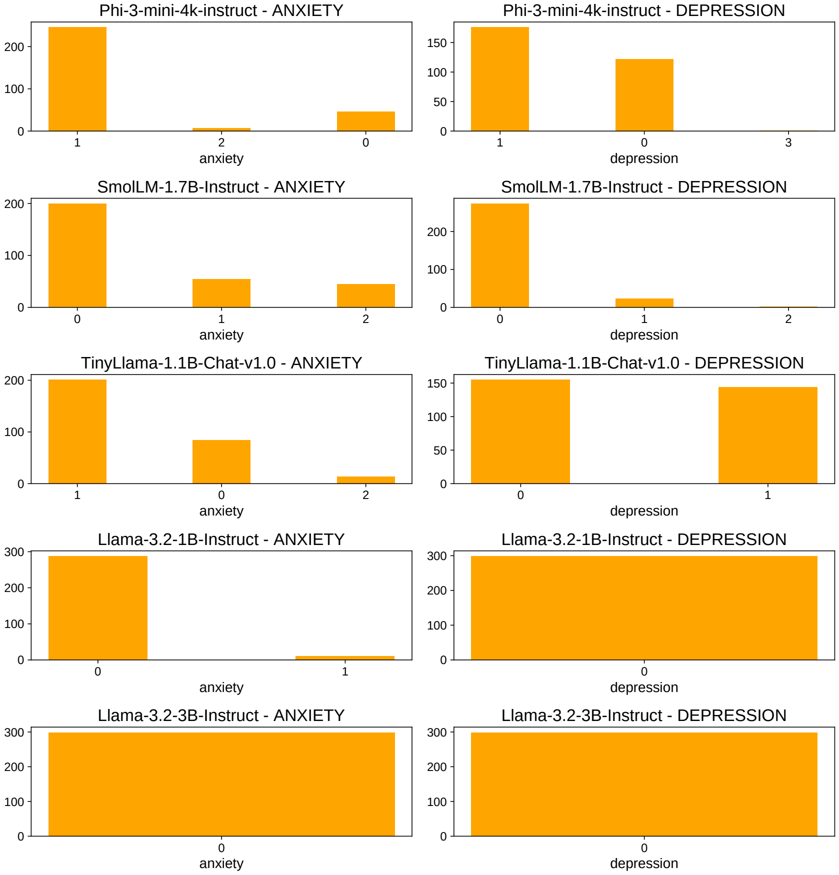

Figure 6: Distribution of predictions for the two tasks in GLOBEM

G Data Augmentation

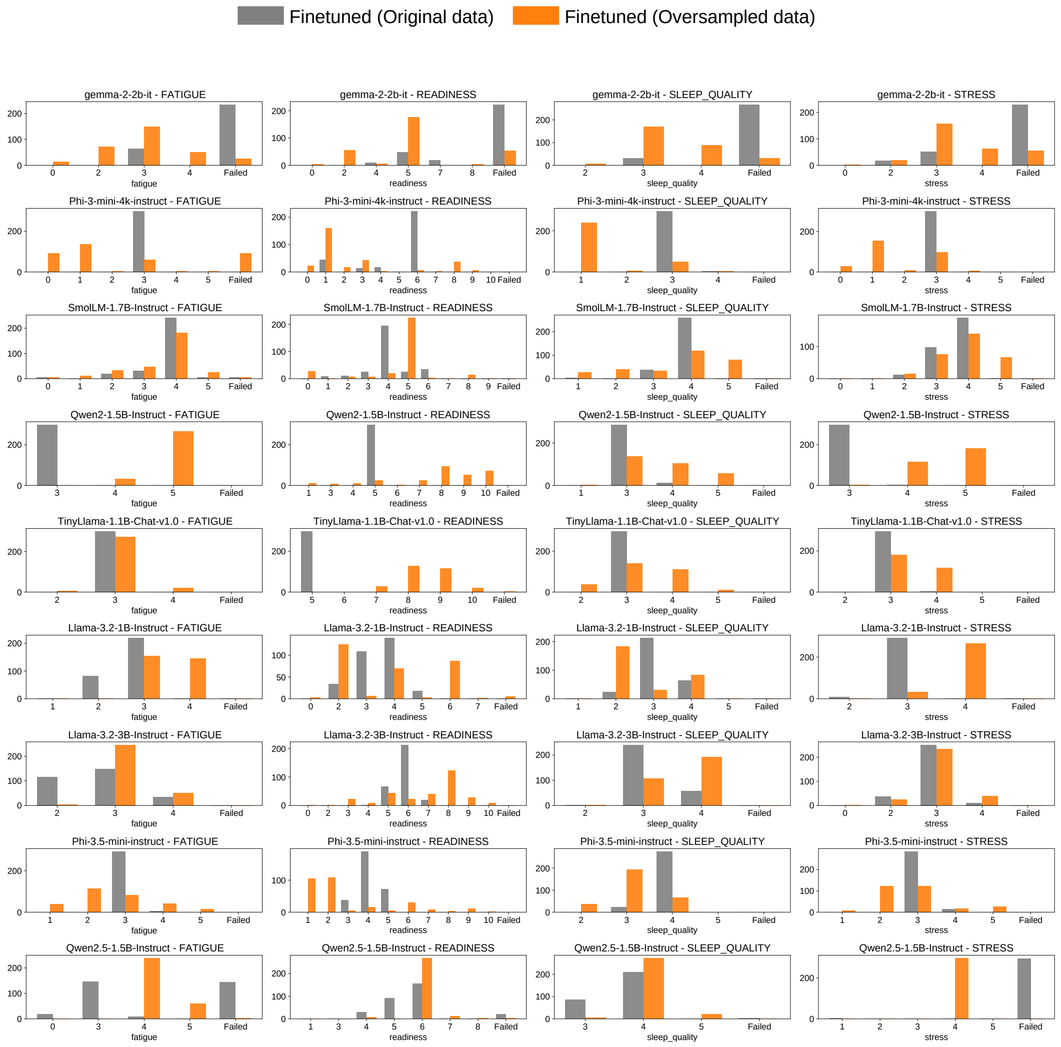

Figure 7: Predicted distributions of SLMs (LoRA) on PMData, comparing models fine-tuned on the original and oversampled datasets.

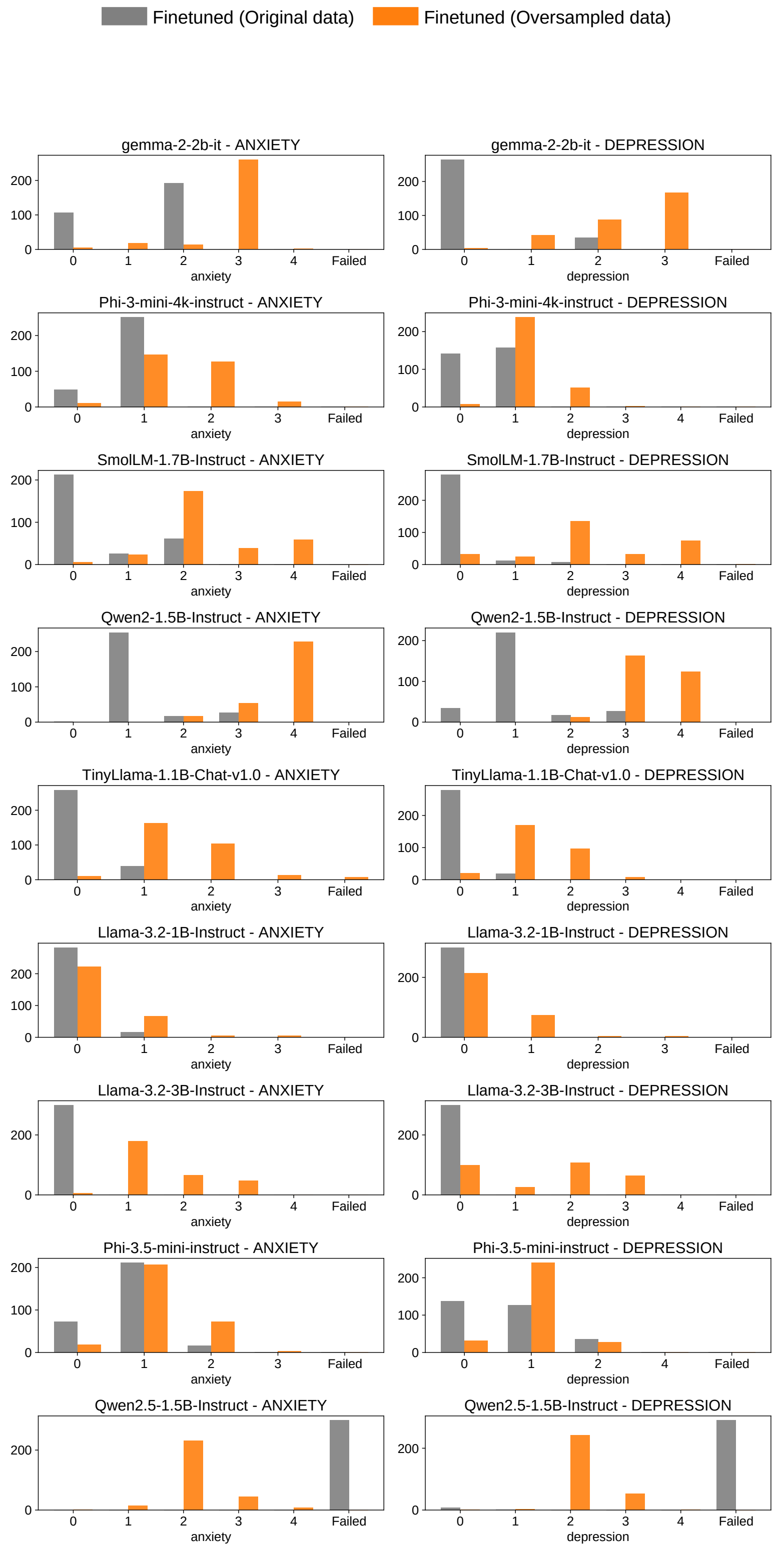

Figure 8: Predicted distributions of SLMs (LoRA) on GLOBEM, comparing models fine-tuned on the original and oversampled datasets.