

Enabling On-Device LLMs Personalization with Smartphone Sensing

Shiquan Zhang
The University of Melbourne
Melbourne, Australia
shiquan.zhang@student.unimelb.edu.au

Ying Ma
The University of Melbourne
Melbourne, Australia
ying.ma1@student.unimelb.edu.au

Le Fang
The University of Melbourne
Melbourne, Australia
le.fang1@student.unimelb.edu.au

Hong Jia
The University of Melbourne
Melbourne, Australia
hong.jia@unimelb.edu.au

Simon D'Alfonso
The University of Melbourne
Melbourne, Australia
dalfonso@unimelb.edu.au

Vassilis Kostakos
The University of Melbourne
Melbourne, Australia
vassilis.kostakos@unimelb.edu.au

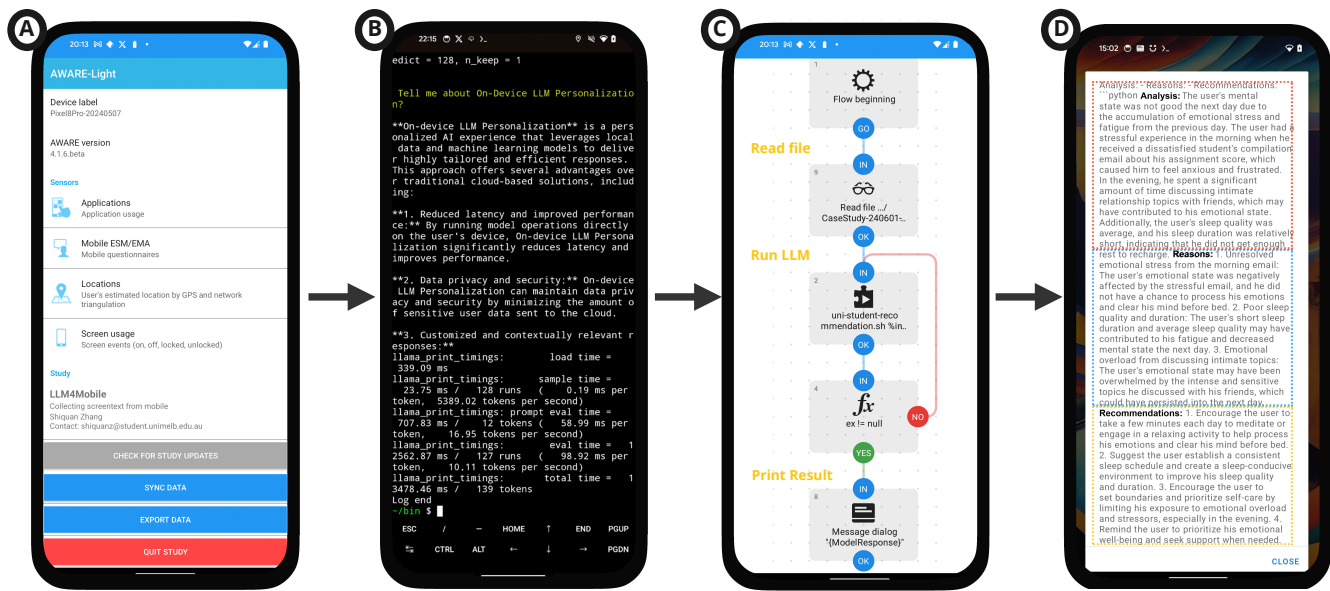


Figure 1: Overview of our work. (A) Sensing data collection from Aware-Light. (B) Implementation of LLM on smartphones. (C) Automatic workflow and trigger in Automate. (D) Personalized analysis and recommendations using LLM.

ABSTRACT

This demo presents a novel end-to-end framework that combines on-device large language models (LLMs) with smartphone sensing technologies to achieve context-aware and personalized services. The framework addresses critical limitations of current personalization solutions via cloud LLMs, such as privacy concerns, latency and cost, and limited personal information. To achieve this, we innovatively proposed deploying LLMs on smartphones with multimodal sensor data through context-aware sensing and customized prompt

engineering, ensuring privacy and enhancing personalization performance. A case study involving a university student demonstrated the capability of the framework to provide tailored recommendations. In addition, we show that the framework achieves the best trade-off in privacy, performance, latency, cost, battery and energy consumption between on-device and cloud LLMs. To the best of our knowledge, this is the first framework to provide on-device LLMs personalization with smartphone sensing. Future work will incorporate more diverse sensor data and involve extensive user studies to enhance personalization. Our proposed framework has the potential to substantially improve user experiences across domains including healthcare, productivity, and entertainment.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

UbiComp Companion '24, October 5–9, 2024, Melbourne, VIC, Australia

© 2024 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-1058-2/24/10

<https://doi.org/10.1145/3675094.3677545>

CCS CONCEPTS

• Human-centered computing → Ubiquitous and mobile computing systems and tools.

KEYWORDS

LLM; On-Device; Smartphone Sensing; Personalization; End-to-End Framework

ACM Reference Format:

Shiquan Zhang, Ying Ma, Le Fang, Hong Jia, Simon D'Alfonso, and Vassilis Kostakos. 2024. Enabling On-Device LLMs Personalization with Smartphone Sensing. In *Companion of the 2024 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp Companion '24)*, October 5–9, 2024, Melbourne, VIC, Australia. ACM, New York, NY, USA, 5 pages. <https://doi.org/10.1145/3675094.3677545>

1 INTRODUCTION

The emergence of large language models (LLMs), exemplified by ChatGPT [21], has revolutionized human-machine interaction and impacted people from all walks of life by leveraging vast amounts of data and sophisticated algorithms to provide users with flexibility and personalized content [6]. Despite their impressive capabilities in understanding, reasoning, and generation, current LLMs face significant limitations, primarily related to privacy and security, since most contemporary LLMs (e.g., ChatGPT, Claude [4], and Gemini [12]), primarily operate in cloud environments which require users to upload their personal data to the cloud, potentially leading to the leakage of sensitive personal information.

Secondly, there are latency and cost issues to consider. Latency can significantly impact user experience, especially when the network is unstable or facing high request volumes and major outage [23]. In critical situations like healthcare monitoring, where real-time data analysis and responses are essential, such latency is unacceptable as it could potentially compromise patient care and safety. When considering cost, cloud LLM services are expensive, preventing their extensive usage. For example, calling APIs from servers can cost around 75 USD per million tokens, as seen with Claude 3 Opus [3].

Lastly, the inability of cloud LLMs to adapt to real-time contextual data from users poses a significant challenge to personalization. Existing LLMs require pre-collected datasets, hindering their usage in personalization scenarios that rely heavily on streaming data. When users request tasks beyond their boundaries, generic LLMs often fail to provide accurate results. Given these issues, it is reasonable to consider smartphones as the ideal platform for sensing human activities and delivering personalized services, as LLMs are not yet extensively deployed on smartphones for this purpose.

In this demo, we aim to address aforementioned challenges by leveraging on-device LLMs combined with smartphone sensing technologies to enable personalized and context-aware services. The objective is to develop an end-to-end framework for on-device personalization with personal multimodal information and customized prompt engineering to meet individual user needs. By overcoming privacy concerns, latency and cost, and limited personal sensor data, our approach can provide more secure, context-aware, and efficient personalized services that directly proceed on users' devices, paving the way for broader applications and improved user experiences in various domains such as healthcare, productivity, and entertainment. To verify the proposed framework, we present a case study exploring a day in the life of a university student and discuss the comparison between on-device and cloud

LLMs. To the best of our knowledge, this is the first framework to provide on-device LLMs personalization with smartphone sensing.

2 RELATED WORK

This section summarizes related studies on current developments and challenges regarding on-device LLMs, smartphone sensing and personalization.

2.1 On-Device LLMs

There is a growing trend towards creating smaller models for deployment on edge devices such as smartphones and wearable devices [1, 17–19]. On-device LLMs refer to LLMs running locally on devices rather than in the cloud, which mitigates the concerns regarding privacy and latency, as the processing occurs locally without connecting to the Internet. Compared to centralized cloud LLMs, which can be plagued by latency and bandwidth issues, on-device LLMs provides faster, more reliable, and more efficient processing, leading to quicker, safer and privacy-preserved decision-making. However, deploying LLMs on edge devices presents challenges due to limited computational resources. Open-source models such as Llama-3-8B [2], Phi-3-mini [1], and Gemma-2B [10] have been introduced and explored for deployment on devices such as PCs and smartphones. Efforts to deploy lightweight LLMs to edge devices are still at an infant stage and applications based on these models have not been comprehensively studied [7, 8, 13, 16, 20, 24, 27]. In comparison, we are the first framework to provide on-device LLMs personalization on smartphones.

Although on-device smaller LLMs have advantages over larger LLMs, they still struggle to meet individual requirements without the inclusion of sufficient contextual knowledge from additional sources.

2.2 Smartphone Sensing and Personalization

With the growing prevalence of smartphones and advancements in their sensing capabilities, smartphones have become a natural platform for understanding the interaction between users, machines, and environments. Smartphone sensing offers unique advantages in terms of cost-effectiveness, user acceptance, and the ability to capture context-aware, fine-grained, and continuous data streams [14]. Meanwhile, personalization has become a research focus which aims to tailor services, content, and user experiences based on individual preferences, behaviours, and contexts [9]. Therefore, smartphones, being inherently personal devices, are ideally suited to facilitate personalization by providing a rich source of user-specific and real-time data.

These works [5, 25, 26, 28], however, either focus on collecting in-depth sensor data without exploring the potential of LLMs or are missing a comprehensive understanding of multi-modal sensors. In comparison, we propose to investigate LLM personalization with extensive multi-modal sensor data.

3 METHODS

This part involves (1) collecting sensor data using AWARE-Light in Section 3.1, (2) deploying LLMs on smartphones in Section 3.2, (3) applying prompt engineering in Section 3.3, and integrating them into (4) an end-to-end pipeline framework in Section 3.4.

3.1 Sensing Data from AWARE-Light

AWARE-Light¹ [26] is an open-source Android software for conducting smartphone sensing studies, which allows users to collect rich sensor data and deploy scheduled questionnaires from both hardware and software on smartphones. It has extensive sensors² such as geolocation, accelerometer, experience sampling method (ESM), keyboard, communications, app usage and screen text. In this demo, we utilized screentext and ESM sensors to collect data, where the screentext sensor [25] can capture all screen text on smartphones, and mobile ESM sensor can periodically collect questionnaire data. Notably, to accommodate on-device scenarios, we specifically extended an export function to export sensor data to local Android files.

3.2 LLMs on Android

We built a working environment to run LLMs on smartphones. Termux³, an Android terminal emulator and Linux environment app, was installed on a Google Pixel 8 Pro with 12GB DRAM. Then, a machine learning compilation engine, a high-performance universal deployment solution that allows native deployment of large language models with compiler acceleration, was set up from llama.cpp⁴, which enables LLMs to run locally with acceleration in Android. Open-source lightweight models, Llama-3-8B [10], were downloaded through Hugging Face⁵ and deployed on the phone. Lastly, LLMs can run locally with a chat interface like ChatGPT in Fig 1.B.

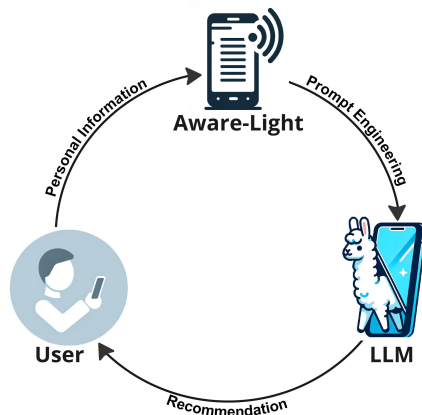


Figure 2: Pipeline of this end-to-end framework. Sensing data collected through Aware-Light from users is processed on the phone with LLMs using prompt engineering to generate personalized responses for users.

¹<https://www.aware-light.org/>

²<https://www.aware-light.org/sensors/>

³<https://termux.dev/en/>

⁴<https://github.com/ggerganov/llama.cpp>

⁵<https://huggingface.co/>

3.3 Prompt Engineering Design

As defined in Equations 1 and 2, a structured prompt template [15] was designed to give LLMs sufficient contextual information. *Instruction* provides the model with explicit guidance on what task needs to be performed, which sets the objective and helps narrow down the model’s response to the desired action. *Context* gives the model the background information necessary to understand the task and produce relevant responses, which is significant to deliver personalization. In this situation, *Context* involves C_{user} , personal information, and C_{domain} , a specific area of knowledge, and $C_{sensing}$, sensing data from AWARE-Light. *Question* specifies what exactly the user wants to know or the information that the user seeks from the model. *Output Format* defines the format in which user wants the information to be presented.

$$Prompt = Instruction + Context + Question + OutputFormat \quad (1)$$

$$Context = C_{user} + C_{domain} + C_{sensing} \quad (2)$$

3.4 An End-to-End Pipeline Framework

Fig 2 illustrates an end-to-end pipeline framework, where sensing data will be collected through AWARE-Light from users, LLMs with prompt engineering will be processed on the phone and finally generate personalized responses to users. Automate⁶, an Android automation tool, is used to trigger the pipeline and visualize the generated responses from LLMs.

4 EXPERIMENTAL SETUP AND EVALUATION

In this part, a case study in Session 4.1 was explored under this framework and the comparison of on-device and cloud LLMs were discussed.

4.1 Case Study

A day of smartphone activity from a university student was captured through screentext and mobile ESM sensors from AWARE-Light. Regarding screentext, fine-grained text data were collected. Considering the verbose input, local Llama-3-8B was utilized to summarize the collected screentext. Regarding ESM questionnaires, morning and nightly questions were delivered at 9 AM and 9 PM, respectively, covering users’ emotional status, sleep duration, sleep quality, fatigue, alcohol consumption, and significant event factors.

A mental state problem was raised to test our framework. More details can be found on prompt design and model responses in Fig 3. Preliminary analysis can be observed from sensor data. Via the ESM questionnaire, the user expressed stress, suffered from a “student complaint” and “drunk alcohol on nightly interview” while had a “bad sleep” even with “average sleep hours last night” on morning interview. As for screentext, an email about “a student complain about assignment score from tutoring”, “unhealthy dinner”, and “chatting intimate relationship in late night” may impact his mental state.

The local LLM provided insightful analysis and explanation based on the captured information, highlighting unresolved emotional stress from a complaint email, poor sleep quality, and intimate nighttime conversations. Personalized analysis and suggestions

⁶<https://llamalab.com/automate/>

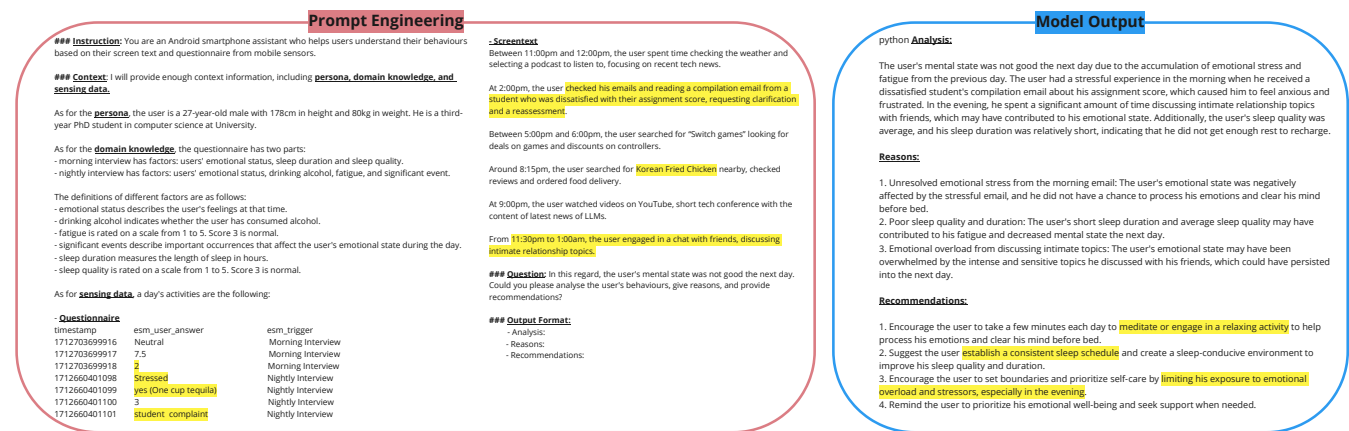


Figure 3: Prompt Engineering Design and Model Output.

were made, “meditate or engage in a relaxing activity before bed”, “establish a consistent sleep schedule”, and “limiting exposure to emotional overload and stressors”, demonstrating the potential for personalized services such as counseling, coaching, and co-piloting.

However, local models still have limitations. Specifically, we occasionally observed that these models exhibit hallucinations, such as generating unrelated content and reversing concepts, as well as ignorance, like failing to consider certain information. Session ?? will discuss potential approaches to mitigate these problems.

4.2 On-Device and Cloud LLMs Comparison

Both cloud and on-device LLMs can deliver personalized services in the case study. Compared to cloud LLMs, our work focusing on on-device LLMs prioritizes **privacy**, as all processing is performed on the smartphone. In terms of **performance**, cloud LLMs can leverage state-of-the-art models, while on-device LLMs may be inferior in accuracy and have a less up-to-date knowledge base. However, we envision future pocket LLMs will be more lightweight, robust and accurate. Regarding **latency**, cloud LLMs can experience higher latency due to potential network instability or high request volumes and major outage [23], affecting real-time performance. In comparison, our framework can provide faster and more reliable responses. In terms of **cost**, on-device LLMs are free, while cloud LLMs require calling APIs from servers such as ChatGPT-4o [22], Claude 3 Opus [3], Gemini 1.5 Pro [11], which can cost from 5 to 75 USD per million tokens. When considering **battery consumption**, on-device LLMs directly impact the device’s battery life. Our tests indicated approximately 16.5% RAM usage and 3% battery drain in 5 minutes on a Google Pixel 8 Pro during our case study on on-device model inference, highlighting one of the future research factors for on-device LLMs. Finally, when examining **energy consumption**, cloud LLMs consume considerable energy due to their extensive computational requirements. An LLM consumes 0.1 J per token for every billion parameters [18]. For example, the cloud LLM ChatGPT, with 175B parameters, consumes approximately 17.5 J per token, resulting in substantial energy usage. On-device LLMs, while consuming less energy overall, still contribute to energy usage, albeit to a lesser extent. For instance, a 7B-parameter model consumes 0.7

J per token. Moreover, the substantial energy consumption of cloud LLMs has significant environmental implications, with ongoing energy usage contributing to increased carbon dioxide emissions and exacerbating environmental problems. As such, enabling on-device LLMs could make significant contributions for ecological AI.

5 DEMONSTRATION

We plan to show a live demo on Google Pixel 8 Pro smartphone, offering an immersive and interactive experience. Users can begin by exploring the AWARE-Light app, observing how it collects various types of sensor data, such as GPS, screentext and ESM. This will provide insights into the foundational data that powers our personalized recommendations. Next, users can engage with on-device LLMs directly on smartphones, where they can create their own prompts or modify pre-built prompts with their sensor data to run LLMs. Lastly, users can view and iterate personalized recommendations based on their interactions and the collected sensor data. This live demo will highlight the seamless integration of data collection, prompting, and personalized recommendation generation, showcasing the robustness and practicality of our end-to-end framework on smartphones.

6 CONCLUSION & FUTURE WORK

We presented a novel end-to-end framework that showcases on-device LLMs to provide more secure, context-aware, and efficient personalized services based on smartphone sensing. Additionally, we conducted a case study to test this framework and deliver personalized recommendations, and made comparison between on-device and cloud LLMs. Our initial experiments demonstrated the great potential of this framework for personalized services. In future work, considering some occasional errors, more contextual information, domain knowledge, and user-specific models, such as fine-tuning models for university scenarios, should be considered. Moreover, we plan to integrate more diverse sensor data and conduct large-scale user studies, allowing us to refine this framework and determine how to deliver more personalized services effectively.

REFERENCES

- [1] Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, et al. 2024. Phi-3 Technical Report: A Highly Capable Language Model Locally on Your Phone. *arXiv preprint arXiv:2404.14219* (2024).
- [2] AI@Meta. 2024. Llama 3 Model Card. https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md
- [3] Anthropic. 2024. Claude AI Pricing. <https://ai-claude.net/pricing/>. Accessed: 2024-06-05.
- [4] Anthropic. 2024. Claude: An AI Language Model. <https://claude.ai/chats>. Accessed: 2024-06-04.
- [5] Florian Bemmman, Timo Koch, Maximilian Bergmann, Clemens Stachl, Daniel Buschek, Ramona Schoedel, and Sven Mayer. 2024. Putting Language into Context Using Smartphone-Based Keyboard Logging. *arXiv preprint arXiv:2403.05180* (2024).
- [6] Yihan Cao, Siyu Li, Yixin Liu, Zhiling Yan, Yutong Dai, Philip S Yu, and Lichao Sun. 2023. A comprehensive survey of ai-generated content (aigc): A history of generative ai from gan to chatgpt. *arXiv preprint arXiv:2303.04226* (2023).
- [7] Samuel Carreira, Tomás Marques, José Ribeiro, and Carlos Grilo. 2023. Revolutionizing Mobile Interaction: Enabling a 3 Billion Parameter GPT LLM on Mobile. *arXiv preprint arXiv:2310.01434* (2023).
- [8] Wei Chen and Zhiyuan Li. 2024. Octopus v2: On-device language model for super agent. *arXiv preprint arXiv:2404.01744* (2024).
- [9] Haiyan Fan and Marshall Scott Poole. 2006. What is personalization? Perspectives on the design and implementation of personalization in information systems. *Journal of Organizational Computing and Electronic Commerce* 16, 3-4 (2006), 179–202.
- [10] Gemma-team. 2024. Gemma. <https://ai.google.dev/gemma>
- [11] Google. 2024. Gemini AI Pricing. <https://ai.google.dev/pricing>. Accessed: 2024-06-05.
- [12] Google. 2024. Gemini: An AI Language Model. <https://gemini.google.com/app>. Accessed: 2024-06-04.
- [13] Vishnu Jaganathan, Deepak Gouda, Kriti Arora, Mohit Aggarwal, and Chao Zhang. 2024. On-Device Video Analysis with LLMs. In *Proceedings of the 25th International Workshop on Mobile Computing Systems and Applications*. 153–153.
- [14] Wazir Zada Khan, Yang Xiang, Mohammed Y Aalsalem, and Quratulain Arshad. 2012. Mobile phone sensing systems: A survey. *IEEE Communications Surveys & Tutorials* 15, 1 (2012), 402–427.
- [15] Yubin Kim, Xuhai Xu, Daniel McDuff, Cynthia Breazeal, and Hae Won Park. 2024. Health-llm: Large language models for health prediction via wearable sensor data. *arXiv preprint arXiv:2401.06866* (2024).
- [16] Yuanchun Li, Hao Wen, Weijun Wang, Xiangyu Li, Yizhen Yuan, Guohong Liu, Jiacheng Liu, Wenxing Xu, Xiang Wang, Yi Sun, et al. 2024. Personal llm agents: Insights and survey about the capability, efficiency and security. *arXiv preprint arXiv:2401.05459* (2024).
- [17] Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Xingyu Dang, and Song Han. 2023. Awq: Activation-aware weight quantization for llm compression and acceleration. *arXiv preprint arXiv:2306.00978* (2023).
- [18] Zechun Liu, Changsheng Zhao, Forrest Iandola, Chen Lai, Yuandong Tian, Igor Fedorov, Yinyang Xiong, Ernie Chang, Yangyang Shi, Raghuraman Krishnamoorthi, et al. 2024. MobileLLM: Optimizing Sub-billion Parameter Language Models for On-Device Use Cases. *arXiv preprint arXiv:2402.14905* (2024).
- [19] llama.cpp team. 2023. llama-cpp. <https://github.com/ggerganov/llama.cpp>
- [20] Joel Ruben Antony Moniz, Soundarya Krishnan, Melis Ozyildirim, Prathamesh Saraf, Halim Cagri Ates, Yuan Zhang, Hong Yu, and Nidhi Rajshree. 2024. ReALM: Reference Resolution As Language Modeling. *arXiv preprint arXiv:2403.20329* (2024).
- [21] OpenAI. 2024. ChatGPT: An AI Language Model. <https://www.openai.com/chatgpt>. Accessed: 2024-06-04.
- [22] OpenAI. 2024. OpenAI API pricing. <https://openai.com/api/pricing/>. Accessed: 2024-06-05.
- [23] OpenAI. 2024. OpenAI's chatbot hit by major outage. <https://status.openai.com/>. Accessed: 2024-06-05.
- [24] Ruiyang Qin, Jun Xia, Zhenge Jia, Meng Jiang, Ahmed Abbasi, Peipei Zhou, Jingdong Hu, and Yiyu Shi. 2023. Enabling On-Device Large Language Model Personalization with Self-Supervised Data Selection and Synthesis. *arXiv preprint arXiv:2311.12275* (2023).
- [25] Songyan Teng, Simon D'Alfonso, and Vassilis Kostakos. 2024. A Tool for Capturing Smartphone Screen Text. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–24.
- [26] Niels van Berkel, Simon D'Alfonso, Rio Kurnia Susanto, Denzil Ferreira, and Vassilis Kostakos. 2023. AWARE-Light: A smartphone tool for experience sampling and digital phenotyping. *Personal and Ubiquitous Computing* 27, 2 (2023), 435–445.
- [27] Hao Wen, Yuanchun Li, Guohong Liu, Shanhuai Zhao, Tao Yu, Toby Jia-Jun Li, Shiqi Jiang, Yunhao Liu, Yaqin Zhang, and Yunxin Liu. 2023. Empowering llm to use smartphone for intelligent task automation. *arXiv preprint arXiv:2308.15272* (2023).
- [28] Huatao Xu, Liying Han, Qirui Yang, Mo Li, and Mani Srivastava. 2024. Penetrative ai: Making llms comprehend the physical world. In *Proceedings of the 25th International Workshop on Mobile Computing Systems and Applications*. 1–7.