

Towards Trustworthy Agentic AI for Automated Wearable-Based Health Monitoring

Yuwei Zhang*
yz798@cl.cam.ac.uk
University of Cambridge
Cambridge, UK

Qiang Yang
qiang.yang@cl.cam.ac.uk
University of Cambridge
Cambridge, UK

Alexandra Loh*
awl14@cam.ac.uk
University of Cambridge
Cambridge, UK

Cecilia Mascolo
cecilia.mascolo@cl.cam.ac.uk
University of Cambridge
Cambridge, UK

Abstract

Wearable devices generate increasingly rich multimodal health data, yet transforming raw sensor streams into meaningful health insights remains technically challenging. Heterogeneous sensing modalities and task-specific modeling requirements often necessitate substantial manual effort and domain expertise to develop effective analytics pipelines. While recent large language model-based coding agents offer a promising alternative, general-purpose code generation workflows are not designed to handle heterogeneous wearable data or satisfy the trustworthiness and interpretability requirements critical for health applications. Hence we present an agentic AI system that automatically constructs machine-learning pipelines from heterogeneous wearable sensor data, including ECG, IMU, audio, and multimodal sensing modalities. Rather than transmitting raw sensor data to a large language model, the system generates and iteratively refines analysis code locally executed in a secure sandbox, improving privacy and transparency. The framework also produced interpretable reports containing feature importance analyses, uncertainty estimates, confidence scores, and clinically relevant evaluation metrics. Across all tasks, the framework achieved a 72.5% functional pipeline generation success rate, automatically producing end-to-end workflows encompassing data preprocessing, feature engineering, model training, evaluation, and reporting. Generated models achieved performance comparable to manually developed baselines, improving F1-scores by up to 48%. Our findings demonstrate the potential of agentic AI to enable scalable, trustworthy, and privacy-conscious well-being monitoring from wearable sensor data.

CCS Concepts

• **Computing methodologies** → **Intelligent agents; Machine learning**; • **Applied computing** → **Health informatics**; • **Human-centered computing** → *Ubiquitous computing*.

*Co-primary authors.



This work is licensed under a Creative Commons Attribution 4.0 International License.
UbiComp Companion '26, Shanghai, China
© 2026 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-2533-3/2026/10
<https://doi.org/10.1145/3798063.3841776>

Keywords

Mobile health; LLM; Coding agents

ACM Reference Format:

Yuwei Zhang, Alexandra Loh, Qiang Yang, and Cecilia Mascolo. 2026. Towards Trustworthy Agentic AI for Automated Wearable-Based Health Monitoring. In *Companion of the 2026 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp Companion '26)*, October 11–15, 2026, Shanghai, China. ACM, New York, NY, USA, 7 pages. <https://doi.org/10.1145/3798063.3841776>

1 Introduction

Wearable devices such as smartwatches, rings, and earbuds are increasingly used to support health and well-being monitoring by continuously collecting physiological and behavioral signals including heart activity, movement, sleep, and respiration [2, 26, 31]. This creates new opportunities for personalized and longitudinal health monitoring and leads to growing interest in mobile health systems that support disease detection, activity recognition, health assessment, and behavioral interventions [12, 22].

However, transforming raw sensor streams into meaningful health insights remains challenging. Variation in sensing modality, format, noise, and labeling scheme means different applications demand different analytical approaches, and free-living data collection introduces motion artifacts, missing values, and class imbalance. Consequently, building effective wearable-data pipelines typically requires expertise spanning signal processing, machine learning, and domain-specific health knowledge [8, 14, 32].

Recent advances in large language models (LLMs) and agents suggest a promising alternative: automatically generating data-analysis pipelines from natural-language specifications. However, existing approaches face two key challenges in mobile health domain.

(i) Wearable data complexity. Wearable datasets are highly heterogeneous, spanning multiple sensing modalities, file formats, and sampling characteristics [17]. Developing effective pipelines requires handling diverse data representations as well as addressing noise, missing values, and class imbalance that commonly arise in free-living settings. These characteristics make automated pipeline construction substantially more difficult than in many conventional machine-learning benchmarks.

(ii) Trustworthy health analytics requirements. Health-related applications impose requirements that extend beyond predictive performance. Models must be evaluated using clinically

meaningful metrics and provide interpretable outputs, confidence estimates, and uncertainty quantification to support trustworthy decision making [3, 5]. General-purpose coding agents are typically optimized for generating functional code and are not explicitly designed to satisfy these domain-specific requirements.

In this work, we propose an agentic AI system that can automatically construct mobile-health analytics pipelines directly from raw wearable sensor datasets. Rather than interpreting sensor data itself, the proposed system generates executable analysis code that operates within a secure sandbox environment, keeping raw data outside the LLM context. The system follows a workflow specifically designed for mobile-health tasks, including data exploration, data loading, model exploration, model refinement, evaluation, and report generation. Each phase is guided by mobile-health-specific objectives, such as handling missing data and class imbalance, prioritizing clinically relevant metrics including sensitivity, specificity, and AUC, and generating model explanations, prediction confidence, and uncertainty estimates. To improve robustness, the system incorporates iterative code generation with execution-based feedback and stateful integration of previous execution traces

We evaluate the framework on four mobile-health datasets spanning diverse sensing modalities and analysis tasks, including ECG-based cardiac analysis, IMU-based activity recognition, audio-based COVID-19 detection, and multimodal activity recognition. Across these tasks, the proposed agent achieves higher execution and functional success rates than general-purpose code-generation baselines, demonstrating that domain-specific workflow design is critical for reliable wearable-data analysis. Beyond pipeline generation, the framework produces structured reports supporting more transparent and trustworthy use of AI-generated analytics in well-being applications. Our contributions are threefold:

- We identify key challenges limiting general-purpose LLM agents on wearable-data analysis: data heterogeneity and the need for health-oriented trustworthiness.
- We present an agentic framework that generates and iteratively refines wearable-health ML pipelines, incorporating domain-specific workflow design and health-oriented analysis with interpretability and uncertainty estimates.
- We evaluate the framework across four mobile-health datasets covering ECG, IMU, audio, and multimodal sensing modalities and compare it against general-purpose baselines.

2 Related Work

Mobile Health and Wearable Data Analytics. Wearable devices equipped with sensors such as ECG, PPG, IMU, and microphones enable continuous monitoring of cardiovascular health, physical activity, sleep, stress, and respiratory conditions in free-living settings. Numerous task-specific systems have been developed for applications including arrhythmia detection, stress monitoring, activity recognition, and respiratory disease screening [12, 13, 24, 27]. However, wearable datasets are often heterogeneous, spanning multiple sensing modalities, sampling rates, and file formats, while also exhibiting noise, missing values, and class imbalance [15–17, 23]. Consequently, developing effective wearable-data analysis pipelines typically requires substantial expertise in signal processing, machine learning, and domain-specific health knowledge.

Large Language Models for Health and Wearable Applications. Recent advances in LLMs have motivated their use in health-related applications, including clinical assistance, personal health agents, and wearable-health analytics [4, 5, 9]. Several studies have explored using LLMs to interpret wearable sensor data or health summaries, either through textual representations of sensor measurements or specialized wearable foundation models [3, 14, 28–30]. While these approaches improve accessibility and natural-language interaction, they generally assume the existence of an underlying analytics pipeline and focus on interpreting sensor data or extracted features. In contrast, our work focuses on automatically constructing the machine-learning workflow itself directly from raw wearable datasets.

Agentic AI and Code Generation. Recent agentic AI systems extend LLMs with planning, tool use, and iterative code refinement, and execution feedback has been shown to substantially improve reliability over single-shot generation [1, 6, 7, 10, 11, 19, 25]. However, existing coding agents are designed as general-purpose systems and do not explicitly address the requirements of mobile-health applications, such as handling heterogeneous wearable datasets, prioritizing clinically meaningful evaluation metrics, or generating interpretable and uncertainty-aware reports. Our work specializes the agentic workflow for wearable-based well-being monitoring through domain-specific pipeline generation, evaluation, and reporting procedures.

3 System Design

3.1 Design Goals

The goal of the system is to automatically construct wearable-data analysis pipelines from a natural-language task description and a raw sensor dataset. We design the system around four requirements. First, it should support heterogeneous wearable datasets with different modalities, file formats, sampling rates, and label structures. Second, it should generate executable machine-learning pipelines rather than directly asking the LLM to interpret raw sensor streams. Third, it should improve reliability through execution feedback, since single-shot code generation is often brittle for complex data-analysis tasks. Fourth, it should support trustworthy well-being monitoring by reporting health-relevant metrics, model explanations, prediction confidence, and uncertainty estimates.

3.2 System Overview

Figure 1 shows the overall architecture. The system consists of three main components: an agent controller, an LLM backend, and a sandbox execution environment. The user provides a task description, a data description, and a local dataset. The agent controller decomposes the task into a sequence of analysis phases and prompts the LLM to generate phase-specific Python code. The generated code is then executed in the sandbox on a copy of the dataset. The sandbox returns execution results, error messages, validation outputs, and generated artifacts to the controller, which either advances to the next phase or asks the LLM to revise the code.

This design separates reasoning from data access. The LLM receives task descriptions, summaries, execution traces, and bounded data previews, but not the full raw sensor dataset. Raw data are

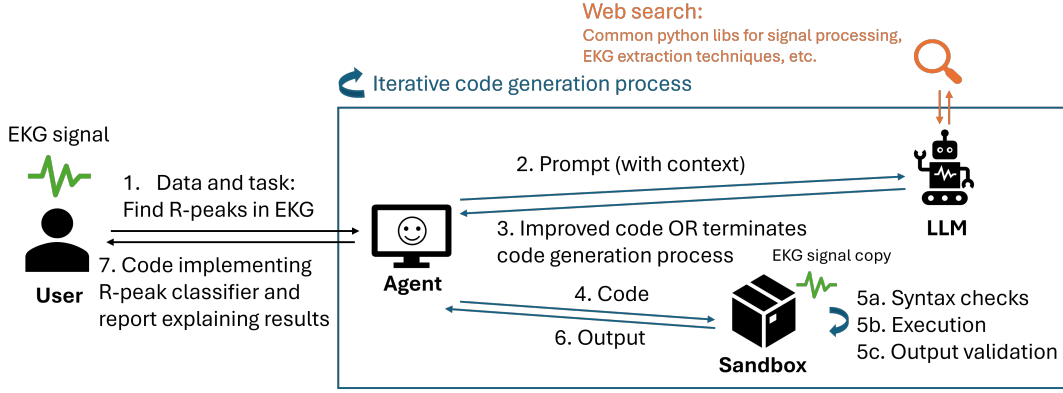


Figure 1: System architecture. The agent coordinates between the user, LLM, and sandbox. The LLM generates code, while the sandbox executes and validates it on a copy of the wearable dataset. The figure shows the system with the evaluation task of identifying R-peaks in EKG data.

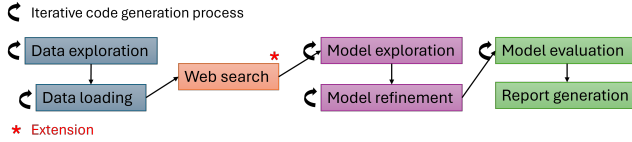


Figure 2: Domain-specific workflow for wearable-data analysis. Each phase generates, executes, and validates code before propagating structured artifacts to downstream stages.

processed by generated code inside the sandbox, which reduces the need to expose sensitive wearable data to the LLM context and prevents generated code from modifying the user’s original files.

3.3 Phase-Based Agent Workflow

Pipeline synthesis is organized as a structured multi-phase workflow tailored to wearable and mobile-health analytics (Figure 2). Each phase consumes a formally defined input state and produces a typed intermediate artifact that serves as the input to subsequent phases. This design enforces modularity, improves interpretability, and enables phase-level validation before execution proceeds.

Data Exploration. The workflow is initialized with an exploratory analysis stage that synthesizes executable code for dataset inspection. The generated program enumerates directory structures, identifies available file formats, extracts metadata describing data organization, and samples representative records. To maintain context efficiency, outputs are truncated and summarized before inclusion in the agent state. The resulting artifact captures structural dataset characteristics that inform subsequent data-ingestion decisions, particularly for heterogeneous wearable-data repositories that lack standardized schemas.

Data Loading. A dataset-specific ingestion module is synthesized from the exploration artifact. This module implements a standardized `load_data()` interface that transforms raw files into a unified representation suitable for downstream modeling. During execution, the generated code characterizes missing-value patterns, class

distributions, subject-level identifiers, temporal structure, and other dataset properties relevant to health-monitoring applications. The resulting artifact consists of a validated dataframe representation together with associated dataset diagnostics.

Web Search (Knowledge Grounding). An optional grounding stage augments the workflow with external task-specific knowledge. The agent retrieves and summarizes information related to established preprocessing procedures, feature-engineering approaches, domain-relevant evaluation criteria, and candidate modeling techniques. This information is summarized and passed to later modeling phases as contextual guidance rather than hard-coded assumptions.

Model Exploration and Refinement. Model development is formulated as a two-stage process comprising candidate exploration and model refinement. Exploration generates and executes experiments across multiple learning algorithms and feature-processing strategies while reporting domain-relevant performance measures. Refinement stage synthesizes a final pipeline encapsulated in a standardized Model abstraction (`train()`, `predict()`, `predict_proba()`) that decouples evaluation from the underlying model family.

Evaluation and Reporting. The evaluation stage executes the finalized pipeline on held-out data and computes metrics appropriate for health-related prediction tasks, including balanced accuracy, F1 score, area under the ROC curve (AUC), sensitivity, and specificity when applicable. Additional analyses include visualization generation, feature-attribution assessment, and confidence-calibrated prediction inspection. A final reporting stage aggregates artifacts produced throughout the workflow into a comprehensive summary documenting dataset characteristics, identified data-quality issues, modeling decisions, predictive performance, uncertainty estimates, and implications for downstream well-being assessment.

3.4 Reliable Code Generation

Each code-generating phase follows an iterative generate–execute–revise procedure. The LLM produces phase-specific code, which is formatted, checked, and executed in the sandbox under a predefined timeout. The agent then validates the execution results and verifies

that the required outputs and artifacts have been produced. When execution fails or the outputs do not satisfy the phase objective, the agent prompts the LLM to revise the code using the previous implementation and the relevant execution feedback.

To maintain a compact and task-focused context, the agent selectively incorporates information from earlier phases and recent execution attempts rather than preserving the complete interaction history. If repeated revisions continue to produce the same failure pattern, the agent discards the unsuccessful local attempt history and requests a new implementation. This recovery step allows the workflow to explore an alternative solution when iterative refinement repeatedly stalls

3.5 Trustworthiness-Oriented Design

The workflow is specialized for well-being applications in three ways. First, prompts explicitly require the agent to consider wearable-data issues such as class imbalance, missing data, and subject-level train-test separation. Second, the model-evaluation phase prioritizes metrics that are meaningful in health contexts, including AUC, sensitivity, and specificity rather than accuracy alone. Third, the final report includes model explanations, confidence scores, and uncertainty-aware interpretations so that generated predictions are presented as decision-support outputs rather than definitive clinical conclusions.

4 Evaluation

We evaluate whether a domain-specific agent workflow improves the reliability of automated wearable-data analysis compared with general-purpose coding assistants. Specifically, we evaluate three research questions:

- **RQ1:** Can the proposed workflow reliably generate executable end-to-end wearable-health analysis pipelines across heterogeneous datasets?
- **RQ2:** Does the domain-specific workflow produce outputs that are more useful for wearable-health decision making?
- **RQ3:** What computational overhead is introduced by the workflow’s iterative exploration and validation stages?

4.1 Experimental Setup

Wearable-health tasks. To evaluate generalizability across health applications, we select four publicly available datasets spanning diverse sensing modalities, prediction tasks, and data complexities.

- **MIT-BIH Arrhythmia Database [21]:** ECG recordings used for cardiac monitoring. The task is *R-peak detection* from raw ECG signals, representing physiological health monitoring and time-series event detection.
- **PAMAP2 [20]:** Wearable inertial and physiological sensors collected during daily activities. The task is *human activity recognition (HAR)*.
- **Coswara [23]:** Respiratory audio recordings collected from healthy and COVID-19-positive participants. The task is *COVID-19 screening* from cough, breathing, and voice recordings, representing acoustic health assessment.

- **Harmony [17]:** A multimodal wearable-sensing dataset combining heterogeneous sensor streams. The task is *human activity recognition* from multiple sensing modalities, requiring sensor-fusion across heterogeneous data sources.

Together these dataset cover cardiovascular monitoring, physical activity analysis, respiratory health assessment, and multimodal behavioral sensing. They reflect common challenges in ubiquitous well-being computing, including heterogeneous file formats, modality-specific preprocessing requirements, missing metadata, and varying machine-learning objectives.

For each task, the agent receives a natural-language objective, a dataset description, and access to the raw dataset downloaded. The agent is asked to generate an executable end-to-end pipeline that loads the data, trains a predictive model, evaluates performance, and produces a final report.

Baselines. We compare our workflow against two representative AI-assisted coding approaches:

- **Gemini 2.5 Pro:** A single-agent baseline where the model receives the same task description and dataset access but without the structured workflow.
- **GitHub Copilot Agent Mode:** A state-of-the-art software-engineering agent capable of autonomously generating and executing code within a development environment.

Both baselines were provided identical dataset access and task descriptions. No dataset-specific prompts, handcrafted preprocessing instructions, or additional domain knowledge were supplied. Each experiment was repeated multiple times to account for variability in LLM generation.

4.2 RQ1: Pipeline Generation Reliability

We evaluate whether the proposed workflow can reliably generate complete wearable-health analytics pipelines across heterogeneous datasets. Following the evaluation protocol used in our full study, we distinguish between two levels of success:

- **Execution success:** the generated code executes without human intervention and produces the expected outputs.
- **Functional success:** the resulting pipeline is both executable and analytically correct, including valid data loading, model training, evaluation, and report generation.

Figure 3 compares success rates across the evaluated datasets. The proposed workflow achieved a mean execution success rate of 0.73, outperforming GitHub Copilot Agent Mode (0.65) and direct Gemini prompting (0.10). The gap becomes more pronounced when considering functional success, which requires not only successful execution but also correct handling of the underlying wearable-health data. Across all evaluated tasks, our framework achieves a 72.5% functional success rate, and the generated models show a substantial improvement in quality, boosting F1-scores by an average of 35% across datasets compared with GitHub Copilot.

Several failures in the baseline systems arose during data understanding and loading. Direct Gemini prompting frequently failed to construct valid loading pipelines for complex datasets, while GitHub Copilot occasionally made incorrect assumptions about dataset schemas, label mappings, or preprocessing requirements. These failures were particularly evident on datasets with non-trivial

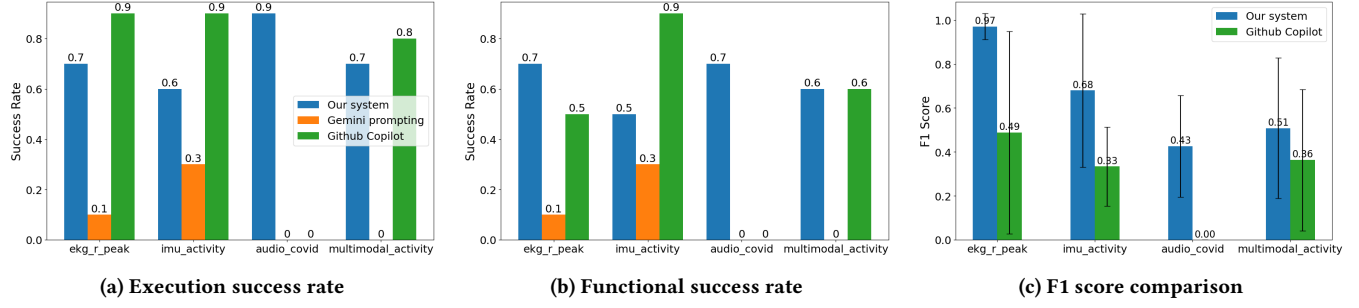


Figure 3: Comparison of end-to-end pipeline generation success rates across wearable-health tasks.

formats such as Coswara, where both baselines consistently failed to load and process the data correctly. Similar issues occurred on Harmony and MIT-BIH, where incorrect assumptions about data structure led to invalid analyses despite successful execution.

A representative failure occurred on the Harmony dataset, where GitHub Copilot generated a fully executable pipeline that reported near-perfect performance. Inspection revealed that the system had inadvertently introduced data leakage during preprocessing, yielding overly optimistic evaluation metrics despite successful execution. Similar failures were observed when baseline agents inferred incorrect dataset schemas or applied inappropriate preprocessing steps without verifying assumptions against the underlying data. More information is included in Appendix¹.

In contrast, the proposed workflow explicitly separates dataset exploration, validation, and model development. Intermediate artifacts generated during exploration and loading are validated before downstream modeling, reducing the likelihood of silent failures that produce plausible but incorrect results. On clean, periodic signals like the MIT-BIH ECG dataset, our workflow achieved a Balanced Accuracy of 98.9% and an F1-score of 97.1%, matching standard domain baselines [18]. Similarly, on the Coswara respiratory audio dataset, the framework achieved an AUC of 68.0% (more results in the Appendix), mirroring the expected performance in literature [23]. These results suggest that structured workflow decomposition improves not only code executability but also the reliability and quality of generated analyses in wearable-health settings.

4.3 RQ2: Does the workflow produce more useful wearable-health analyses?

Beyond pipeline correctness, we evaluate whether the workflow produces outputs that are useful for wearable-health decision making. In this setting, usefulness extends beyond predictive performance and includes transparency, uncertainty communication, and the use of evaluation criteria that are meaningful in health-monitoring applications. The comparison is reported in Table 1.

Across all successful runs, our proposed workflow generated structured reports that summarized dataset characteristics, preprocessing decisions, model-selection rationale, evaluation results, and prediction confidence. Reports consistently included diagnostics

Table 1: Comparison of information included in generated analyses. The proposed workflow consistently produces health-oriented reports containing dataset diagnostics.

Analysis component	Ours	Copilot	Gemini
Dataset diagnostics	100%	43%	50%
Model rationale	100%	0%	0%
AUC and Sensitivity reporting	100%	57%	0%
Prediction confidence	100%	100%	0%
Interpretation of results	100%	19%	0%

such as class-distribution analysis, discussion of potential data-quality issues, and health-relevant metrics including balanced accuracy, sensitivity, specificity, and AUC where applicable. The workflow also reports representative predictions together with confidence estimates and accompanying interpretations, allowing users to assess the reliability of model outputs rather than relying solely on aggregate performance metrics.

In contrast, the baseline systems primarily focused on producing executable code. Even though GitHub Copilot sometimes generated executable training pipelines, the resulting outputs were limited to generic machine-learning metrics and lacked systematic discussion of uncertainty or clinical relevance. Consequently, users received limited insight into why a model was selected, how reliable its predictions were, or which aspects of the data most influenced performance.

Case study. To illustrate these capabilities, we examine a representative report generated for COVID-19 screening using the Coswara dataset². The report demonstrates three characteristics that are particularly relevant for wearable-health applications. First, the workflow performs health-aware model selection rather than optimizing solely for generic machine-learning metrics. During model exploration, the agent compares multiple candidate models and explicitly identifies false negatives as the most critical error type for a screening application. Consequently, it selects the model with the highest sensitivity, prioritizing the detection of positive cases over marginal improvements in overall accuracy. Second, the report emphasizes evaluation metrics that are meaningful in a healthcare

¹Code and appendix is available at <https://github.com/alexlohlw/ai-mobile-health-analysis>

²The report is available at the github page.

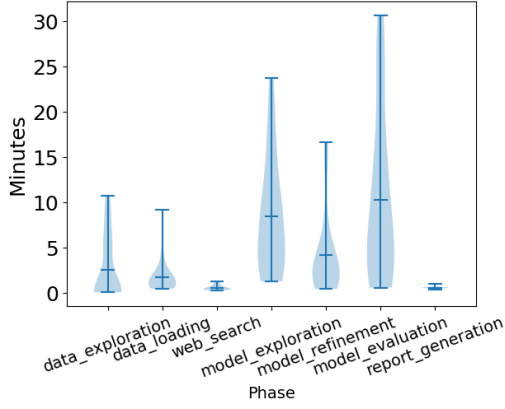


Figure 4: Runtime distribution across workflow phases. Iterative model exploration dominates overall execution time.

context. Rather than reporting accuracy alone, it presents sensitivity, specificity, balanced accuracy, and AUC, and discusses the implications of these metrics for real-world deployment. The report explicitly highlights the trade-off between detecting positive cases and avoiding false alarms, allowing users to understand the model’s strengths and limitations. Third, the workflow communicates uncertainty through prediction confidence estimates. The report includes representative true-positive, true-negative, false-positive, and false-negative examples together with their predicted probabilities and interpretations. This enables users to identify which predictions are likely to be reliable and which should be treated with caution. In particular, the report highlights high-confidence errors as potential failure modes requiring further investigation before deployment. Together, these findings suggest that the proposed workflow contributes not only to reliable pipeline generation but also to the production of more transparent and interpretable wearable-health analyses. By explicitly communicating uncertainty, clinically relevant trade-offs, and model limitations, the generated reports provide information that is more actionable for health-monitoring applications than model predictions alone.

4.4 RQ3: What computational overhead does the workflow introduce?

The proposed workflow introduces additional computation through iterative code generation, execution, validation, and model refinement. To quantify this overhead, we measure runtime across all workflow phases.

Figure 4 shows the distribution of execution time. Successful runs required approximately 28 minutes on average. The majority of runtime was concentrated in the model exploration and refinement stage, where multiple candidate pipelines are generated, executed, and compared before selecting a final solution. Data exploration and loading incurred comparatively small costs, while knowledge grounding and report generation contributed only a minor fraction of total runtime.

The additional computational cost reflects a deliberate design choice. Rather than generating a single solution and assuming correctness, the workflow repeatedly validates intermediate artifacts

and incorporates execution feedback into subsequent decisions. The results from RQ1 indicate that this overhead yields substantially higher reliability and reduces the occurrence of silent analytical failures. For wearable-health applications, where incorrect conclusions may affect downstream well-being assessments, the observed runtime increase represents a practical tradeoff for improved robustness and trustworthiness.

5 Discussion

Our results highlight that executable code is not sufficient for trustworthy wearable-health analytics. Several baseline pipelines completed successfully but produced invalid analyses due to incorrect assumptions about dataset structure or evaluation procedures. This distinction between execution success and functional success is particularly important in well-being applications, where analytical errors may remain unnoticed while generating plausible results.

The findings also suggest that workflow design plays a critical role in agent performance. Rather than relying solely on stronger foundation models, structuring analysis into exploration, validation, modeling, and evaluation stages helps reduce failure modes associated with heterogeneous sensor datasets. This principle may extend to other ubiquitous-computing domains that involve complex data-processing pipelines.

The study has several limitations. Evaluation was conducted on four public datasets and focused on offline supervised-learning tasks. The workflow remains dependent on the capabilities of the underlying LLM and cannot guarantee methodological correctness. Furthermore, comparisons were limited to general-purpose baseline pipelines; future evaluation against more advanced coding agents, such as those with tool-use, multi-agent collaboration, or domain-specific fine-tuning, would provide a more comprehensive picture of the relative benefits of health-specific workflow design. This work also focuses on pipeline generation and evaluation; deployment-side considerations such as inference latency and model compression fall outside the current scope. Future work will also investigate broader sensing applications, stronger validation mechanisms, and human-in-the-loop oversight for high-stakes well-being scenarios.

6 Conclusion

This paper presented a domain-specific agent workflow for automated wearable-health analytics. By decomposing pipeline generation into validated intermediate stages, the workflow improves the reliability of end-to-end analysis on heterogeneous sensing datasets. Evaluation across four well-being monitoring tasks demonstrated higher pipeline success rates than general-purpose coding assistants and revealed the importance of distinguishing executable pipelines from functionally correct analyses. These findings suggest that structured agent workflows are a promising direction for trustworthy AI-assisted analytics in wearable and ubiquitous computing.

Acknowledgments

This work was supported by EPSRC grant EP/Z53447X/1.

References

- [1] Daniil A. Boiko, Robert MacKnight, and Gabe Gomes. 2023. Emergent autonomous scientific research capabilities of large language models. *arXiv:2304.05332 [physics.chem-ph]* <https://arxiv.org/abs/2304.05332>
- [2] Justin Cosentino, Anastasiya Belyaeva, Xin Liu, Nicholas A Furlotte, Zhun Yang, Chace Lee, Erik Schenck, Yojan Patel, Jian Cui, Logan Douglas Schneider, et al. 2024. Towards a personal health large language model. *arXiv preprint arXiv:2406.06474* (2024).
- [3] Zachary Enghardt, Chengqian Ma, Margaret E. Morris, Chun-Cheng Chang, Xuhai “Orson” Xu, Lianhui Qin, Daniel McDuff, Xin Liu, Shwetak Patel, and Vikram Iyer. 2024. From Classification to Clinical Insights: Towards Analyzing and Reasoning About Mobile and Behavioral Health Data With Large Language Models. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 8, 2 (May 2024), 1–25. doi:10.1145/3659604
- [4] Kai He, Rui Mao, Qika Lin, Yucheng Ruan, Xiang Lan, Mengling Feng, and Erik Cambria. 2025. A Survey of Large Language Models for Healthcare: from Data, Technology, and Applications to Accountability and Ethics. *arXiv:2310.05694 [cs.CL]* <https://arxiv.org/abs/2310.05694>
- [5] A Ali Heydari, Ken Gu, Vidya Srinivas, Hong Yu, Zhihan Zhang, Yuwei Zhang, Akshay Paruchuri, Qian He, Hamid Palangi, Nova Hammerquist, et al. 2025. The anatomy of a personal health agent. *arXiv preprint arXiv:2508.20148* (2025).
- [6] Dong Huang, Jie M Zhang, Michael Luck, Qingwen Bu, Yuhao Qing, and Heming Cui. 2023. Agentcoder: Multi-agent-based code generation with iterative testing and optimisation. *arXiv preprint arXiv:2312.13010* (2023).
- [7] Md. Ashraf Islam, Mohammed Eunus Ali, and Md Rizwan Parvez. 2024. MapCoder: Multi-Agent Code Generation for Competitive Problem Solving. *arXiv:2405.11403 [cs.CL]* <https://arxiv.org/abs/2405.11403>
- [8] Arlene John, Stephen J. Redmond, Barry Cardiff, and Deepu John. 2022. A Multimodal Data Fusion Technique for Heartbeat Detection in Wearable IoT Sensors. *IEEE Internet of Things Journal* 9, 3 (2022), 2071–2082. doi:10.1109/JIOT.2021.3093112
- [9] 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. In *Proceedings of the fifth Conference on Health, Inference, and Learning (Proceedings of Machine Learning Research, Vol. 248)*, Tom Pollard, Edward Choi, Pankhuri Singhal, Michael Hughes, Elena Sizikova, Bobak Mortazavi, Irene Chen, Fei Wang, Tasmie Sarker, Matthew McDermott, and Marzyeh Ghassemi (Eds.), PMLR, 522–539.
- [10] Hung Le, Hailin Chen, Amrita Saha, Akash Gokul, Doyen Sahoo, and Shafiq Joty. 2024. CodeChain: Towards Modular Code Generation Through Chain of Self-revisions with Representative Sub-modules. *arXiv:2310.08992 [cs.AI]* <https://arxiv.org/abs/2310.08992>
- [11] Jia Li, Ge Li, Yongmin Li, and Zhi Jin. 2025. Structured Chain-of-Thought Prompting for Code Generation. *ACM Trans. Softw. Eng. Methodol.* 34, 2, Article 37 (Jan. 2025), 23 pages. doi:10.1145/3690635
- [12] Steven Lubitz, Anthony Faranesh, Caitlin Selvaggi, Steven Atlas, David Mcmanus, Daniel Singer, Sherry Pagoto, Michael McConnell, Alexandros Pantelopoulou, and Andrea Foulkes. 2022. Detection of Atrial Fibrillation in a Large Population Using Wearable Devices: The Fitbit Heart Study. *Circulation* 146 (09 2022), 101161CIRCULATIONAHA122060291. doi:10.1161/CIRCULATIONAHA.122.060291
- [13] Ashley Mason, Frederick Hecht, Shakti Davis, Joseph Natale, Wendy Hartogensis, Natalie Damaso, Kajal Claypool, Stephan Dilchert, Subhasis Dasgupta, Shweta Purawat, Varun Viswanath, Amit Klein, Anoushka Chowdhary, Sarah Fisher, Claudine Anglo, Karena Poldon, Danou Veasna, Jenifer Prather, Leena Pandya, and Benjamin Smarr. 2022. Detection of COVID-19 using multimodal data from a wearable device: results from the first TemPredict Study. *Scientific Reports* 12 (03 2022), 3463. doi:10.1038/s41598-022-07314-0
- [14] Mike A. Merrill, Akshay Paruchuri, Naghmeh Rezaei, Geza Kovacs, Javier Perez, Yun Liu, Erik Schenck, Nova Hammerquist, Jake Sunshine, Shyam Tailor, Kumar Ayush, Hao-Wei Su, Qian He, Cory Y. McLean, Mark Malhotra, Shwetak Patel, Jiening Zhan, Tim Althoff, Daniel McDuff, and Xin Liu. 2025. Transforming Wearable Data into Personal Health Insights using Large Language Model Agents. *arXiv:2406.06464 [cs.AI]* <https://arxiv.org/abs/2406.06464>
- [15] G.B. Moody and R.G. Mark. 2001. The impact of the MIT-BIH arrhythmia database. *IEEE engineering in medicine and biology magazine : the quarterly magazine of the Engineering in Medicine & Biology Society* 20 (06 2001), 45–50. doi:10.1109/51.932724
- [16] Karel Mundnich, Brandon M Booth, Michelle l’Hommedieu, Tiantian Feng, Benjamin Girault, Justin L’hommedieu, Mackenzie Wildman, Sophia Skaaden, Amrutha Nadarajan, Jennifer L Villatte, et al. 2020. TILES-2018, a longitudinal physiologic and behavioral data set of hospital workers. *Scientific Data* 7, 1 (2020), 354.
- [17] Xiaomin Ouyang, Zhiyuan Xie, Heming Fu, Sitong Cheng, Li Pan, Niwen Ling, Guoliang Xing, Jiayu Zhou, and Jianwei Huang. 2023. Harmony: Heterogeneous Multi-Modal Federated Learning through Disentangled Model Training. 530–543. doi:10.1145/3581791.3596844
- [18] Jiapu Pan and Willis J Tompkins. 1985. A real-time QRS detection algorithm. *IEEE transactions on biomedical engineering* 3 (1985), 230–236.
- [19] Chen Qian, Wei Liu, Hongzhang Liu, Nuo Chen, Yufan Dang, Jiahao Li, Cheng Yang, Weize Chen, Yusheng Su, Xin Cong, Juyuan Xu, Dahai Li, Zhiyuan Liu, and Maosong Sun. 2024. ChatDev: Communicative Agents for Software Development. *arXiv:2307.07924 [cs.SE]* <https://arxiv.org/abs/2307.07924>
- [20] Attila Reiss. 2012. PAMAP2 Physical Activity Monitoring. UCI Machine Learning Repository. DOI: <https://doi.org/10.24432/C5NW2H>.
- [21] Andrew Roman, Harvard-MIT Division of Health Sciences, Technology, Roger G. Mark, and George B. Moody. 2025. MIT-BIH Arrhythmia Database (Simple CSVs). doi:10.34740/KAGGLE/DSV/12526755
- [22] Kamal Shah, Anran Wang, Yiwen Chen, Jitender Munjal, Sumeet Chhabra, Anthony Stange, Enxun Wei, Tuan Phan, Tracy Giest, Beszel Hawkins, et al. 2025. Automated loss of pulse detection on a consumer smartwatch. *Nature* 642, 8066 (2025), 174–181.
- [23] Neeraj Sharma, Prashant Krishnan, Rohit Kumar, Shreyas Ramoji, Srikanth Raj Chetupalli, Nirmala R., Prasanta Kumar Ghosh, and Sriram Ganapathy. 2020. Coswara — A Database of Breathing, Cough, and Voice Sounds for COVID-19 Diagnosis. In *Interspeech 2020 (interspeech_2020)*. ISCA, 4811–4815. doi:10.21437/interspeech.2020-2768
- [24] Ali Tazarv, Sina Labbaf, Stephanie Reich, Nikil Dutt, Amir M. Rahmani, and Marco Levorato. 2021. Personalized Stress Monitoring using Wearable Sensors in Everyday Settings, Vol. 2021. 7332–7335. doi:10.1109/EMBC46164.2021.9630224
- [25] Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, Rui Zheng, Xiaoran Fan, Xiao Wang, Limao Xiong, Yuhao Zhou, Weiran Wang, Changhao Jiang, Yicheng Zou, Xiangyang Liu, Zhangyue Yin, Shihan Dou, Rongxiang Weng, Wensen Cheng, Qi Zhang, Wenjuan Qin, Yongyan Zheng, Xipeng Qiu, Xuanjing Huang, and Tao Gui. 2023. The Rise and Potential of Large Language Model Based Agents: A Survey. *arXiv:2309.07864 [cs.AI]* <https://arxiv.org/abs/2309.07864>
- [26] Maxwell A Xu, Girish Narayanswamy, Kumar Ayush, Dimitris Spathis, Shun Liao, Shyam A Tailor, Ahmed Metwally, A Ali Heydari, Yuwei Zhang, Jake Garrison, et al. 2025. LSM-2: Learning from Incomplete Wearable Sensor Data. *arXiv preprint arXiv:2506.05321* (2025).
- [27] Yuzhe Yang, Yuan Yuan, Guo Zhang, Hao Wang, Ying-Cong Chen, Yingcheng Liu, Christopher Tarolli, Daniel Crepeau, Jan Bukartky, Mithri Junna, Aleksandar Videnovic, Terry Ellis, Melissa Lipford, Ray Dorsey, and Dina Katabi. 2022. Artificial intelligence-enabled detection and assessment of Parkinson’s disease using nocturnal breathing signals. *Nature Medicine* 28 (08 2022), 1–9. doi:10.1038/s41591-022-01932-x
- [28] Xiaofan Yu, Lanxiang Hu, Benjamin Reichman, Dylan Chu, Rushil Chandrupatla, Xiyuan Zhang, Larry Heck, and Tajana Rosing. 2025. SensorChat: Answering Qualitative and Quantitative Questions during Long-Term Multimodal Sensor Interactions. *arXiv:2502.02883 [cs.AI]* <https://arxiv.org/abs/2502.02883>
- [29] Yuwei Zhang, Kumar Ayush, Siyuan Qiao, A. Ali Heydari, Girish Narayanswamy, Maxwell A. Xu, Ahmed A. Metwally, Shawn Xu, Jake Garrison, Xuhai Xu, Tim Althoff, Yun Liu, Pushmeet Kohli, Jiening Zhan, Mark Malhotra, Shwetak Patel, Cecilia Mascolo, Xin Liu, Daniel McDuff, and Yuzhe Yang. 2025. SensorLM: Learning the Language of Wearable Sensors. In *Conference on Neural Information Processing Systems (NeurIPS)*.
- [30] Yuwei Zhang, Tong Xia, Bianca Emmerich, Yu Yvonne Wu, Dimitris Spathis, Xin Liu, Daniel McDuff, and Cecilia Mascolo. 2026. WEQA: Wearable hEalth Question Answering with Query-Adaptive Agentic Reasoning. *arXiv:2606.18147 [cs.AI]* <https://arxiv.org/abs/2606.18147>
- [31] Yuwei Zhang, Tong Xia, Jing Han, Yu Y Wu, Georgios Rizos, Yang Liu, Mohammed Mosuly, Jagmohan Chauhan, and Cecilia Mascolo. 2024. Towards open respiratory acoustic foundation models: Pretraining and benchmarking. *Advances in Neural Information Processing Systems* 37 (2024), 27024–27055.
- [32] Lili Zhu, Petros Spachos, and Stefano Gregori. 2022. Multimodal Physiological Signals and Machine Learning for Stress Detection by Wearable Devices. In *2022 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*. 1–6. doi:10.1109/MeMeA54994.2022.9856558