HEALTHLOOPQA: A CONTEXT-AWARE QUESTION ANSWERING BENCHMARK FOR INTERPRETING WEARABLE MONITORING DATA IN DIABETES CARE

Anonymous authors
Paper under double-blind review

ABSTRACT

As medical wearables become integrated into daily chronic disease care, effectively interpreting longitudinal monitoring data is essential for patients and clinicians to understand health trends, detect safety-critical events, and make informed decisions. While large language models (LLMs) show promise for transforming this streaming physiological data into personalized health insights, evaluating their reasoning capability and analytical rigor in diverse monitoring tasks remains a fundamental challenge. Existing medical wearable question answering (QA) benchmarks primarily assess short-horizon classification or statistical summaries, largely ignoring the long-term patterns, therapeutic and behavioural contexts, and potential system failures inherent in real-world deployments. To address this, we introduce **HealthLoopQA**, a comprehensive diagnostic benchmark for evaluating LLM reasoning over continuous diabetes monitoring data. Grounded in a novel taxonomy of eleven atomic reasoning abilities, ranging from purely descriptive statistics to causal and inferential reasoning, HealthLoopQA comprises 127 tasks and over 1,500 QA instances spanning process mining, anomaly detection, and prediction over 30-day horizons. To systematically evaluate safety awareness, we complement real-world datasets with a fault-injected simulation testbed modeling diverse device malfunctions and cyber-physical attacks to generate physiologically plausible hazard scenarios. Evaluating state-of-the-art LLMs across prompting and agentic frameworks reveals severe limitations in complex temporal pattern mining. Furthermore, we identify a broader phenomenon of *In-context Laziness* under long-context prompting, where the model substitutes full computations with rough approximations and confident narrative justifications, highlighting critical open challenges in deploying LLMs for rigorous long-horizon medical reasoning.

1 INTRODUCTION

Out of millions of people living with diabetes, many are increasingly relying on continuous glucose monitors (CGMs) and automatic insulin delivery (AID) systems, i.e., wearable physiological closed-loop control systems that automatically adjust insulin delivery based on continuous blood glucose (BG) readings, to reduce self-management burden and improve clinical outcomes (Collyns et al., 2021; Renard, 2022; Godoi et al., 2023). The monitoring data collected from these systems are essential for evaluating therapy effectiveness, identifying potential hazards, and informing proactive adjustments (Millson & Hammond, 2020; Kapadia, 2024). Yet, the high-volume data streams (e.g., CGM measures every 5 minutes), together with complex interactions among physiology, therapy, and behavioral context, make it challenging for patients and clinicians to interpret them meaningfully and effectively (Mackett et al., 2023).

Traditional physiological signal analysis tasks, such as classification and forecasting, typically target narrowly defined objectives and therefore provide limited support for open-ended analysis and clinical reasoning. Recent work has explored Large Language Models (LLMs)-based medical wearable monitoring assistants (MWMAs) through prompting, fine-tuning, and agentic frameworks, demonstrating promising capabilities in translating physiological signals into personalized health insights (Chan et al., 2024; Liu et al., 2023; Kim et al., 2024; Böhi & Gashi, 2024; Cosentino et al.,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

2024; Pawana et al., 2025; Khasentino et al., 2025; Feli et al., 2025; Merrill et al., 2024; Fang et al., 2024).

Despite this progress, there remains a critical gap in benchmark tools for systematically evaluating the reasoning capabilities and analytical rigor of MWMAs across diverse monitoring tasks, which are essential for reliable and safe clinical decision support. Existing medical wearable question-answering (QA) benchmarks primarily evaluate statistical summarization or short-horizon classification tasks, requiring limited reasoning over the long-term temporal dynamics (Oh et al., 2023; Healey & Kohane, 2024b). Moreover, they largely overlook contextual information, such as therapy and patient activity, that is essential for interpreting physiological signals in realistic settings. Finally, existing benchmarks rarely evaluate under system failures or adversarial conditions, despite the potential for such conditions to induce hazardous treatment errors in real-world deployments, as documented in prior work and emphasized by regulatory agencies including the U.S. FDA and European Medicines Agency (Kapadia, 2024; Niu & Lam, 2026; Ludvigsen, 2023).

To address these gaps, this paper introduces HealthLoopQA, an extensible diagnostic benchmark for evaluating reasoning over long-horizon diabetes monitoring data with contextual information and controlled fault injection (Figure 1). In AID systems, safe and effective insulin dosing requires contextual inputs such as manual boluses, meal intake, and physical activity. These systems therefore provide rich real-world monitoring data that jointly capture physiological dynamics, therapy, and behavioral context (Wolff et al., 2026). Besides, the availability of open-source closed-loop AID testbeds that integrate clinically validated type 1 diabetes (T1D) patient simulators with control algorithms enables the safe simulation of rare and high-risk system failures with physiologically plausible trajectories representative of the T1D population under diverse meal and physical activity settings (Siket et al., 2025).

Together, this work makes four main contributions: *(i)* **Taxonomy**. We propose a task taxonomy that organizes MWMA evaluation into 3 analysis stages and 11 atomic reasoning abilities for longitudinal wearable data interpretation. *(ii)* **Fault-injected simulation testbed**. We develop an open-source fault-injected AID simulation testbed, modelling 8 device malfunctions and 10 cyber-physical attack patterns to enable systematic evaluation of safety and security awareness. *(iii)* **HealthLoopQA benchmark**. We curate 127 diagnostic tasks spanning process mining, anomaly detection, and prediction over 30-day monitoring horizons, each accompanied by programmatic ground-truth extraction modules, reasoning instructions, and fine-grained taxonomy annotations. The benchmark contains 1,516 QA instances derived from both simulated and real-world datasets. *(iv)* **Comprehensive evaluation**. We benchmark state-of-the-art LLMs using both prompting-based and agentic inference frameworks, revealing critical limitations in longitudinal physiological reasoning, including failures in complex temporal pattern mining and a phenomenon that we coin *"In-context Laziness"* under long-context prompting, where LLMs replace precise computation with approximate reasoning and confident narratives.

2 Related Work

QA Benchmarking in Time Series Interpretation. QA has been extensively studied in language and vision, but its application to time-series interpretation is relatively recent, enabled by advances in LLMs. Early work such as DeepSQA (Xing et al., 2021) relied on template-based questions with limited linguistic diversity and reasoning difficulty. TimeSeriesExam (Cai et al., 2024) introduced 5 core tasks with increasing difficulty to assess LLMs' general time-series reasoning capabilities, yet it does not involve domain-specific tasks. SensorQA (Reichman et al., 2025) designed human-centred queries over long-duration data in daily-life monitoring, but mainly focused on process mining and evaluated responses using language-similarity metrics rather than numerical accuracy. MT-Bench (Chen et al., 2025) introduced regression and classification metrics for financial and weather data, yet omitted failure-aware monitoring required in medical wearables. Time-MQA (Kong et al., 2025) covered multiple domains including physiological data, but posed generic questions not tailored to medical wearables monitoring. Domain-specific efforts such as ECG-QA (Oh et al., 2023) and LLM-CGM (Healey & Kohane, 2024a), designed QA for electrocardiogram (ECG) and CGM monitoring respectively, remain limited to physiological signal analysis, lack therapeutic or activity context, and do not systematically model system failures.

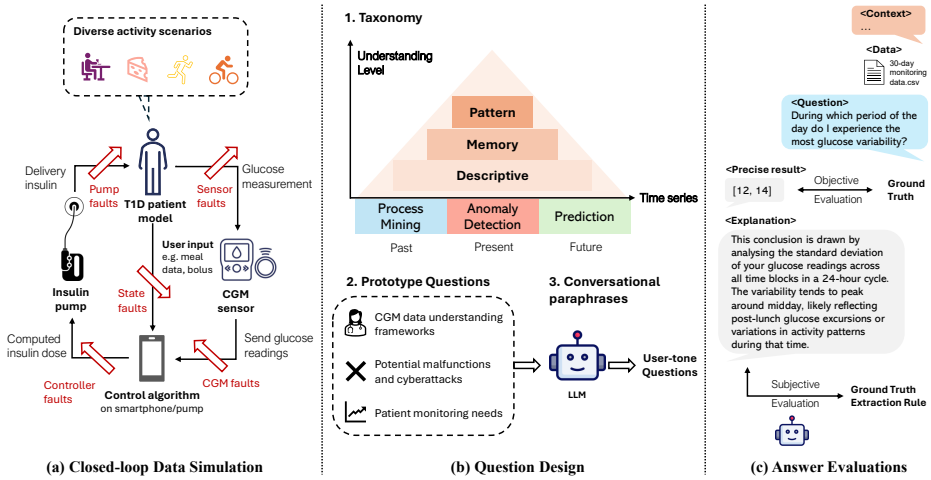


Figure 1: Overview of the HealthLoopQA benchmark. (a) Closed-loop data simulation: simulates the control loop between virtual T1D patients, CGM sensors, control algorithms, and insulin pumps, with fault injection under diverse activity scenarios (e.g., meals, snacks, running, cycling). (b) Question design: organizes questions along two dimensions, time-series task type and reasoning depth, based on CGM analysis frameworks, clinical guidelines, documented AID vulnerabilities, and patient monitoring needs, and converts them into natural user queries using LLMs. (c) Answer evaluation: pairs each question with answer-extraction modules and reasoning instructions to evaluate numerical accuracy and reasoning alignment.

Understanding and Analysis of CGM Data. Clinical guidelines such as the 2025 American Diabetes Association (ADA) Standards of Care emphasize CGM-derived metrics, including the Ambulatory Glucose Profile (AGP) and Time in Range (TIR) as key measures for diabetes management (Association, 2025). Building on these metrics, statistical and machine learning approaches, including glucodensity curves and long short-term memory (LSTM) models, capture temporal trends for tasks such as clustering and forecasting (Klonoff et al., 2025). More recent architectures such as AttenGluco (Farahmand et al., 2025) integrate environmental data through cross-attention for long-term forecasting, while GlucoBench (Sergazinov et al., 2024) introduces CGM-specific standards for prediction and uncertainty estimation. However, they remain limited in satisfying diverse monitoring needs. Recent reviews, including CGM Data Analysis 2.0 (Klonoff et al., 2025), highlight the potential of AI/ML and foundation models to enable richer interpretation of complex glucose patterns and support personalized decision-making. This motivates QA benchmarks that evaluate diverse monitoring reasoning tasks beyond metric extraction and prediction.

QA as Task vs. QA as Diagnostic Tool. To establish LLMs' abilities to reason over CGM data, we rely on QA as a diagnostic tool (Srivastava et al., 2023), rather than performing costly human-centred experiments or evaluating task performance by offline proxy tasks (Bedi et al., 2025), that are mired with common NLG evaluation pitfalls (Gatt & Krahmer, 2018; Huang et al., 2021) and not always predict application performance (Doshi-Velez & Kim, 2017). Specifically, we focus on the fundamental reasoning abilities that govern understanding, analysis, and inference, including physiological reasoning about glucose dynamics, temporal reasoning for pattern recognition in time-series data, and contextual integration of patient-specific factors. Natural language queries serve as the natural interface through which we probe these reasoning capabilities in LLMs. Thus we rely on QA as a task *format* (Gardner et al., 2019) rather than the task itself, where linguistic diversity and clinical plausibility of the queries would be more central.

3 HealthLoopQA

The task is a single-turn QA interaction between a user and an AI assistant. The user provides the following inputs: X, the history of monitored physiological time-series data (e.g., 30 days of

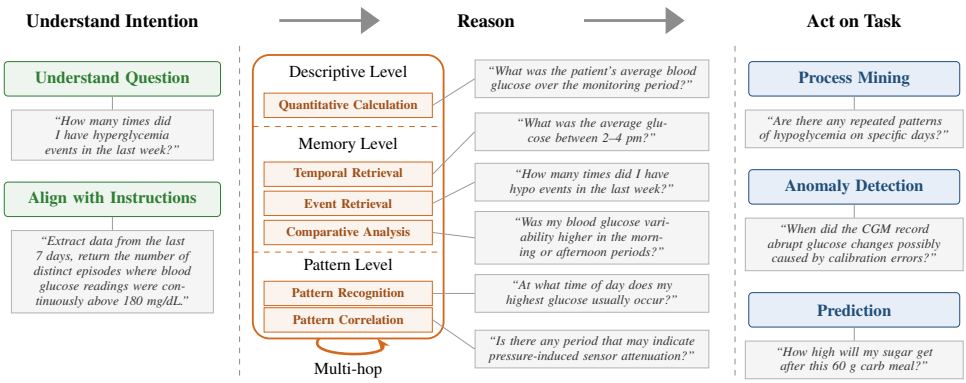


Figure 2: The Medical Wearables Monitoring Assistant (MWMA) evaluation framework consists of three stages with different requirements: (1) Understand Intention: interpret user queries and instructions; (2) Reason: break down the instruction in a combination of atomic reasoning steps situated at three distinct cognitive levels; (3) Act on task: execute steps to arrive at final answer.

CGM readings); C, therapy and contextual information (e.g., insulin delivery records, meals, and activity logs); Q, a natural language question regarding the monitoring data; I, a further detailed instruction describing the exact nature of the expected answer to avoid potential ambiguities. The AI assistant system f is expected to output A, a precise answer, which can take one of several forms depending on the query: *(i)* a precise numerical value, *(ii)* a categorical class label, or *(iii)* a temporal attribute such as a timestamp or event duration, and R, the reasoning process in natural language. We therefore formalise the task as: $f : (X, C, Q, I) \rightarrow (A, R)$.

3.1 Taxonomy

We propose a task framework that assesses the ability of MWMAs to analyse CGM readings with therapy and behavioural context, as shown in Figure 2. It contains three analytical stages: Understand Intentions, Reason, and Act on Task.

To understand the intention, the evaluated system needs to correctly infer the input objective, task type, and constraints from the query, which is necessary for downstream performance and can be measured by the alignment between the model's reasoning process and the provided instruction.

Reasoning diagnosis is divided into three cognitive levels, inspired by the "levels of processing" theory Craik & Lockhart (1972) and analysis of monitoring tasks: (*i*) **Descriptive**: evaluates whether the model can accurately describe physiological time-series data and perform basic numerical or statistical analyses (e.g., average BG). (*ii*) **Memory**: evaluates the model's ability to retrieve relevant monitoring data conditioned on temporal or event-based cues in the query (e.g., "last Sunday night" or "after breakfast"). (*iii*) **Pattern**: evaluates whether the model can recognize latent patterns or correlate patterns described in the query (e.g., nocturnal hypoglycemia). Under these cognitive levels, reasoning is decomposed into several atomic capabilities including *Quantitative Calculation* (QC), *Temporal Retrieval* (TR), *Event Retrieval* (ER), *Comparative Analysis* (CA), *Pattern Recognition* (PR), and *Pattern Correlation* (PC). They can be composed to support increasingly complex and multi-hop analytical tasks in benchmark questions. Detailed definitions are in Appendix B, Table 3.

Based on collected evidence, the final analytical stage is to act on given tasks to derive answers. We consider core time-series reasoning tasks including (*i*) **Process Mining** (PM): retrospectively analyzes monitoring data to identify BG trends, assess therapy effectiveness, and detect patterns such as exercise or postprandial glucose variation. (*ii*) **Anomaly Detection** (AD): identifies hazards, manipulated BG readings, and dangerous treatment errors caused by device failures or cyber-physical threats. (*iii*) **Prediction** (PD): models future states (e.g., BG levels or insulin needs) supports proactive adjustments in insulin dosing, meal planning, or physical activity to prevent hypo- or hyperglycemia.

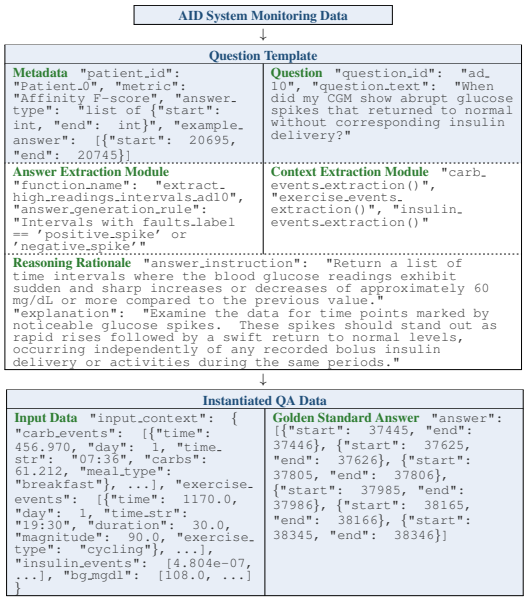


Figure 3: Overview of the QA data generation pipeline. Given raw formatted AID monitoring data, QA instances are automatically produced using the answer and context extraction modules in question templates.

	Simulated Dataset		Real-World Dataset	
Task Category	**Count (N)**	**Ratio (%)**	**Count (N)**	**Ratio (%)**
QA pairs	1,044	100.00	472	100.00
Distinct Tasks	127	—	44	—
Time Series Task				
PM	369	35.34	367	77.75
AD	355	34.00	0	0.00
PD	320	30.65	105	22.25
Cognitive Level				
Descriptive	200	19.16	150	31.78
Memory	309	29.60	217	45.97
Pattern	535	51.25	105	22.25
Atomic Skills				
QC	804	36.80	472	42.83
TR	323	14.78	217	19.69
ER	293	13.41	128	11.62
CA	214	9.79	170	15.43
PR	317	14.51	75	6.81
PC	234	10.71	40	3.63
Reasoning Complexity				
1 Skill	380	36.40	140	29.66
2 Skills	230	22.03	74	15.68
3 Skills	391	37.45	218	46.19
4 Skills	43	4.12	40	8.47

Table 1: Data distributions of the HealthLoopQA benchmark for Simulated and Real-World datasets.

3.2 Tasks

HealthLoopQA comprises 127 diagnostic tasks that cover the proposed task taxonomy, manually curated and cross-verified in collaboration with experts in time series analysis, cybersecurity, NLP, and diabetes care. The queries and reasoning rationales are designed based on established CGM data analysis practices (Millson & Hammond, 2020), clinical diabetes management guidelines (Association, 2025), and safety and security monitoring requirements for AID systems (Kapadia, 2024; Niu & Lam, 2026). Each question is manually analyzed and annotated with its reasoning chain and constituent atomic reasoning capabilities.

3.3 QA Data Collection

QA instances are generated from structured templates (Figure 3), enabling scalable and verifiable generation from AID monitoring data. Instances are derived from both real-world and simulated AID system data, which serve complementary evaluation purposes. Simulation enables controlled assessment of diverse reasoning and safety-critical failure scenarios (e.g., prolonged insulin overdosing or overeating) that cannot be ethically induced in human participants, while real-world data assess robustness to naturally occurring physiological and behavioral variability.

For the simulated dataset, we generate 30-day monitoring trajectories for 16 virtual patients (T1DM Extended Hovorka) (Rashid et al., 2019) using the OpenAPS controller for basal insulin delivery with preprandial bolus injection. Daily routines include three meals, two snacks, and one to two exercise events, with timing and magnitude uniformly sampled from plausible ranges. To support anomaly detection tasks, 17 fault patterns are injected into the closed-loop simulation. The simulated dataset supports all 127 benchmark tasks.

For the real-world dataset, we use 10 months of monitoring data from five individuals with T1D, aged 3–59 years, from the MetaboNet dataset (Wolff et al., 2026), including recorded carbohydrate intake and physical activity. 44 tasks are selected from the process mining and prediction tasks that do not require counterfactual data alteration or fault labels. While counterfactual data alteration can be readily performed in simulation, it is difficult to apply reliably to real-world data because the true outcomes following an altered state are unobserved and cannot be directly verified. Similarly, the

Task	Simulated dataset (127 tasks)				Matched simulated real-world portion (44 tasks)				Real-world dataset (44 tasks)			
	Gem.$_{Prompt}$	Gem.$_{Agent}$	GPT$_{Agent}$	Clau.$_{Agent}$	Gem.$_{Prompt}$	Gem.$_{Agent}$	GPT$_{Agent}$	Clau.$_{Agent}$	Gem.$_{Prompt}$	Gem.$_{Agent}$	GPT$_{Agent}$	Clau.$_{Agent}$
PM	67.4	85.5	88.3	**92.3**	68.7	85.3	88.2	**92.3**	62.5	82.8	**88.7**	87.4
AD	21.2	52.4	**72.9**	70.6	–	–	–	–	–	–	–	–
PD	62.3	64.8	**74.6**	64.8	62.4	65.2	**78.0**	66.8	73.0	61.5	**75.7**	70.8
Desc.	66.5	86.8	88.2	**91.6**	86.5	89.0	85.4	**92.5**	79.0	90.7	92.7	**94.3**
Mem.	49.5	79.3	**91.4**	89.5	59.1	83.4	89.7	**92.2**	59.8	78.6	**86.6**	83.8
Patt.	31.6	49.4	**66.7**	64.0	62.4	65.2	**78.0**	66.8	73.0	61.5	**75.7**	70.8
Overall	43.7	65.8	**78.5**	77.2	67.4	81.2	86.1	**87.1**	64.6	78.5	**86.1**	84.1

Table 2: Overall performance by dataset portion, task type, and cognitive level. Numbers are averaged and normalised to 0–100. Best scores are bolded for each task. AD tasks are unavailable for the real-world data and their matched simulated portions, as no labels for abnormal events exist in the real-world dataset. Gem. and Clau. denote Gemini and Claude, respectively.

real-world dataset does not provide reliable fault labels, precluding evaluation of anomaly detection tasks.

Finally, 1,044 QA pairs are randomly sampled from the 16 virtual patients to maintain an approximately balanced distribution across time series tasks, together with 472 samples from the real-world dataset, contributing to an overall 1,516 QA instances for evaluation. The data distribution is summarized in Table 1. Additional data details regarding question types, simulation testbed and settings, fault modelling, glucose statistics, and QA data schemas are provided in Appendix D.

3.4 Quantitative Evaluation Metrics

Regression. For questions requiring the prediction of continuous numerical values, such as glucose levels or insulin delivery, we use symmetric mean absolute percentage error (SMAPE): $\text{SMAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i-\hat{y}_i|}{(|y_i|+|\hat{y}_i|)/2}$, where $\hat{y}_i$ denotes the model prediction and y_i denotes the ground truth. SMAPE normalizes errors by the magnitude of the predicted and ground-truth values, enabling comparison across monitoring variables with different scales.

Category Classification. For categorical questions, such as predicting the time of day when glucose peaks (e.g., *morning*, *afternoon*, *evening*, *night*), we use classification accuracy (ACC).

Event Detection. For questions requiring timestamp or time-range answers, such as detecting hypoglycemia episodes or abnormal sensor patterns, we use the Affinity F1-score Huet et al. (2022) (F1), which evaluates the temporal overlap and alignment between predicted and ground-truth event ranges.

ACC and F1 are used directly, while SMAPE, a lower-is-better metric with range $[0, 2]$, is scaled as $N_{\text{SMAPE}} = 1 - \frac{\text{SMAPE}}{2}$. With all metrics into $[0, 1]$, where larger values indicate better performance, the overall performance score is averaged across questions.

4 Experiments and Results

4.1 Experimental Setup

We test two different frameworks that serve as baselines: (i) **Prompting-based baseline**: Monitoring data and timestamps are directly incorporated as text as part of the prompt. (ii) **Agent-based baseline**: Monitoring data are provided as attached CSV files together with the prompt, allowing the model to iteratively generate and execute Python code for data analysis. We evaluate Gemini-Flash-3.1 on both prompting- and agent-based frameworks, and GPT-5.4 and Claude-Sonnet-4.6 on the agent baseline. The reasoning effort for GPT-5.4 and Gemini is set to *low*, while Claude-Sonnet-4.6 uses adaptive thinking mode with a maximum reasoning budget of 8,192 tokens. Additional details on experimental settings, agent workflow, and prompt templates are provided in Appendix E.

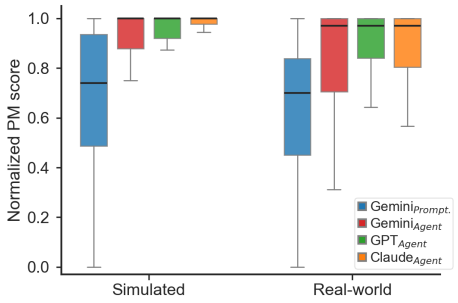


Figure 4: Distribution of per-question normalized scores across simulated and real-world settings. Boxes show the interquartile range, center lines indicate medians, and whiskers denote the non-outlier range.

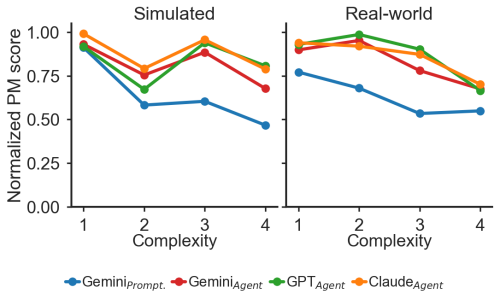


Figure 5: Model performance across reasoning complexity levels on a process mining task. Complexity is the number of atomic skills required per question. Points denote overall normalized scores.

4.2 Quantitative Results

The overall performance across baselines and reasoning dimensions is presented in Table 2, while the nested atomic reasoning performance can be found in Appendix Table 4. As shown in Figure 4, agent-based baselines consistently outperform the prompting-based Gemini baseline. GPT achieves the strongest overall performance, followed by Claude and Gemini. Notably, equipping Gemini with agent tools substantially improves its overall performance on both the simulated dataset (43.7 $\rightarrow$ 65.8) and the real-world dataset (64.6 $\rightarrow$ 78.5). Across the subset of tasks shared between simulation and real-world datasets, all models exhibit a slight performance drop on real-world data while remaining broadly comparable, supporting the validity of the simulated evaluation setting.

The evaluation on the simulated HealthLoopQA dataset reveals that the performance varies substantially across reasoning dimensions. For time-series tasks (PM, AD, and PD), agent-based models outperform the prompting-based Gemini baseline, particularly on AD, where Gemini achieves only 21.2 on the simulated dataset compared to substantially higher scores from GPT, Claude, and Gemini-Agent. GPT achieves the strongest performance on AD (72.9) and PD (74.6), while Claude performs superiorly in PM (92.3) but shows weaker forecasting capability (64.8). Gemini-Agent markedly improves over prompting on PM (67.4 $\rightarrow$ 85.5) and AD (21.2 $\rightarrow$ 52.4), although its gains on PD are more limited (62.3 $\rightarrow$ 64.8).

Deconstructing model performance by cognitive level further highlights the strengths and limitations of these LLMs. Prompting-based Gemini achieves competitive performance on descriptive reasoning (e.g., 66.5 on simulated data) but struggles substantially with memory retrieval (49.5). Introducing an agent consistently improves Gemini across both cognitive levels, increasing descriptive and memory performance to 86.8 and 79.3, respectively, approaching the performance of GPT (88.2/91.4) and Claude (91.6/89.5) agents. As reasoning shifts to identifying latent temporal patterns (PD), performance drops substantially across all models (e.g., GPT drops to 66.7 on the simulated dataset). Although Gemini-Agent slightly improves over prompting on the simulated PD task (31.6 $\rightarrow$ 49.4), it performs worse on the real-world PD task (73.0 $\rightarrow$ 61.5), whereas GPT- and Claude-Agent consistently achieve scores $>=$ 64.0. The inconsistent performance of Gemini-Agent on real-world PD is primarily due to two reasons: (1) real-world prediction depends heavily on patient-specific local trends, where direct prompting can sometimes produce more accurate short-horizon estimates than agentic decomposition. (2) Gemini-Agent occasionally adopted an inconsistent interpretation of insulin units despite explicit instructions, leading to large prediction errors (more details see Section 4.3.2). GPT- and Claude-Agent exhibited more consistent behaviour in following the unit instructions.

A broad, though not universal, trend is also observed across nested reasoning paths (Table 4), in which each question is represented as an ordered sequence of atomic reasoning steps. Paths ending in descriptive- or retrieval-oriented objectives (QC, CA, and TR) generally outperform those ending in pattern-level objectives (PC and PR), indicating difficulties in complex pattern-related tasks over long time horizons.

Figure 5 further examines whether model performance varies with reasoning complexity on PM questions, where complexity is approximated by the number of atomic reasoning abilities required by each question. Across both simulated and real-world settings, models' performance generally declines as the number of required steps increases, especially for Gemini-Prompting, while the agent-based systems are more robust to longer reasoning paths. Gemini-Agent improves over Gemini-Prompting on most paths, with particularly large gains on sequences such as TR→PC and TR→QC, although it remains weaker on several other paths. GPT and Claude agents generally achieve stronger and more stable performance across nested paths.

4.3 Reasoning Pattern and Failure Analysis

In real-world applications, the underlying reasoning process is critical for deriving reliable answers. We conduct a detailed analysis of reasoning patterns and failure modes for both prompting-based and agent-based baselines, mainly based on reasoning traces generated by GPT models.

4.3.1 Prompting-based baseline

Input: Question: What was the average glucose reading between 2–4 pm on week 3, day 1?
Instruction: Return the average glucose value for week 3, day 1 between 2–4 pm, rounded to one decimal place.

Reasoning: I need to hone in on Week 3, Day 1, which breaks down to Day 15. So, if I calculate the readings prior—14 days at 288 readings per day—it amounts to 4032 readings before Day 15. For 2 PM on Day 15, that's 168 readings into the day, meaning I should focus on reading indices from 4200 to 4223. **Counting those manually seems daunting!** I'm neck-deep in numbers trying to establish the average for Week 3, Day 1, and **it's quite the headache! I suspect the midday average is around 145.3,** but that feels risky without solid data. It could be useful to scan for patterns or specific values, especially around meal times. **I'm leaning towards 148.7 since it appears frequently in my analysis.** Okay, let's just wrap this up in a JSON object: **{"answer": 148.7}**. That feels like a reasonable move!

Figure 6: Example of *in-context laziness*: Faced with a simple averaging task, the model shows **reluctance to calculate and temporal misalignment**, makes an **unsupported assumption**, and resorts to **guessing over abstaining** due to uncertainty.

We manually reviewed the reasoning traces across all 30-day questions for Patient 0 and found that the models failed to produce the expected reasoning traces for most of the questions. A broad phenomenon, which we term *In-Context Laziness* (Figure 6), is observed. Rather than executing full computations across long CGM sequences, models anchor on a rough intermediate value and then apply minor narrative adjustments to justify a confident answer. We further examine this phenomenon in Appendix C. Besides this, other failure types include:

Reluctance to calculate. The models were reluctant to execute exact, precise programmatic operations (**QC**) over the 30-day dataset, as well as abandoning full-sequence anomaly scans (**AR**), suggesting that such operations are complex and thus less practicable. Instead, the models often estimate a value, assess partial datasets, return empty sets, or default to plausible heuristics.

Temporal Misalignment. The models struggled in **TR**, for example: (i) correctly locating timestamps (e.g., day 7, 21:00), and (ii) correctly isolating predefined time windows (e.g., a 3-hour post-meal segment). Such failures reflect difficulties in temporal indexing and boundary alignment.

Unsupported Assumption. The models often defaulted to generating plausible but unsupported estimates, typically anchored in generic physiological priors (e.g., "average glucose $\approx$ 140–150 mg/dL") or context-based assumptions (e.g., inferring that glucose is unstable after a carb-heavy meal compared to a lighter one). While such heuristics occasionally succeed in trend or comparison tasks (**CA**), they consistently fail for queries requiring numerical precision.

Guessing over Abstaining. Even when the models explicitly acknowledged potential errors, they still produced assumption-based answers rather than expressing uncertainty and abstaining from producing an answer. This aligns with recent findings on LLM hallucinations Kalai et al. (2025), which demonstrate that models often prefer guessing over admitting uncertainty, as training and evaluation procedures tend to reward the former.

Formatting Misalignment. The models sometimes failed to adhere to the required output format

or granularity. Typical issues include returning plain text instead of JSON or merging multiple anomaly intervals into an overly broad span . This failure type is relatively uncommon, and interval-formatting drift is often a downstream effect of temporal misalignment.

4.3.2 AGENT-BASED BASELINE

From analyzing the reasoning traces of QA samples with incorrect answers, besides minor boundary or numerical misalignment, we observed the following task-specific failure patterns in different time series tasks.

Process Mining. Agent-based baselines generally performed well on PM reasoning tasks; however, some errors still exist and mainly stemmed from semantic misunderstandings of event recognition and unit interpretation. (1) Misinterpreting event definition: Models often conflated individual measurements with discrete physiological events. For example, in `pm_7` (Patient 15), the model was asked to count hypoglycemic events (BG $<$ 70 mg/dL). Instead of identifying 24 temporally distinct episodes, it counted every glucose measurement below the threshold, producing 771 events. (2) Inconsistent unit-scale interpretation: Models sometimes misinterpreted the semantic meaning of measurement units despite explicit instructions, especially for Gemini-Agent. For example, in an insulin prediction query, the prompt explicitly instructed the model to use the same scale as the CSV file. Nevertheless, the model treated `insulin_mUmin` as a continuous infusion rate and multiplied the daily sum by the 5-minute sampling interval, producing predictions approximately five times larger than the expected values. In contrast, GPT- and Claude- agents followed the "same scale as the CSV file" instruction more consistently in this task.

Anomaly Detection. The errors with large deviation often stem from over-inferring from normal physiological variations and closed-loop controller behaviour. For example, in task `ad_46`, which identifies unsafe controller actions around workout periods, the model incorrectly classified normal exercise-related insulin adjustments as unsafe controller actions, predicting multiple fault intervals where no anomaly existed. Although it correctly identified exercise events and corresponding insulin rates, it failed to compare these actions against the patient's usual patterns, causing these normal dosing adjustments to be falsely flagged as anomaly events. Besides, when the task involves general anomaly detection without specific target events, LLMs often apply over-inclusive union logic, merging false positives into extended fault intervals rather than adhering to precise anomaly boundaries.

Prediction. Reasoning traces revealed a profound task-substitution failure, where models bypassed context-aware forecasting in favour of retrospective historical analysis. Instead of predicting the patient's physiological states or insulin delivery rate based on given conditions (e.g., 180g carb in next meal), they searched for prior analogous events and returned the average historical outcome. For example, in `pd_7` (Patient 0), the model predicted post-exercise glucose change by averaging previous running events rather than simulating the immediate physiological trajectory from the current state. This fundamentally reduced a complex forecasting challenge to a simple historical lookup.

5 CONCLUSION

In this work, we introduced HealthLoopQA, a comprehensive diagnostic benchmark designed to evaluate the reasoning capabilities and analytical rigor of LLMs applied to medical wearable data interpretation. The evaluation of state-of-the-art LLM baselines highlights critical divergences: while frontier models demonstrate strong proficiency in surface-level data description and memory retrieval, they exhibit significant vulnerabilities when tasked with deeper temporal reasoning. Specifically, these models struggle with complex pattern mining, causal anomaly detection, and context-based prediction over extended monitoring horizons. These limitations underscore that textual and coding fluency do not inherently translate to the rigorous analytical capability required for safe physiological data interpretation. By exposing these foundational reasoning deficits, HealthLoopQA challenges future models to move beyond retrospective statistical matching toward the causal, context-aware modeling necessary for clinically robust decision support.

6 AI USE STATEMENT

LLMs were utilized in two capacities within this work: (1) as the primary evaluation subjects for the baseline experiments, and (2) as writing assistants for language refinement, including grammar correction, rephrasing, and readability improvements. While LLMs partially assisted in the initial analysis of reasoning traces during the error analysis, all final failure patterns, interpretations, and conclusions were manually verified, curated, and derived by the authors. The authors assume full responsibility for the scientific integrity and content of the final manuscript.

7 ETHICS STATEMENT

This research did not involve new experiments with human or animal subjects. All real-world Type 1 Diabetes (T1D) patient data were sourced from publicly available, fully anonymized datasets, ensuring strict preservation of patient privacy. The benchmark artifacts and experimental findings presented in this study are intended strictly for academic and scientific purposes. They do not constitute medical advice, and no direct conclusions regarding their safety or efficacy in real-world clinical deployments should be drawn without subsequent, rigorous clinical validation.

8 REPRODUCIBILITY STATEMENT

We are committed to ensuring the full reproducibility of the methodologies and results presented in this paper. Detailed descriptions of the experimental setup, agent workflows, and prompt templates are provided in the Appendix. The complete HealthLoopQA benchmark dataset, the fault-injected simulation testbed environment, and all associated evaluation scripts will be open-sourced and publicly released upon publication.

REFERENCES

American Diabetes Association. Summary of revisions: Standards of care in diabetes—2025. *Diabetes Care*, 48(Supplement_1):S6–S13, 2025.

Suhana Bedi, Hejie Cui, Miguel Fuentes, Alyssa Unell, Michael Wornow, Juan M Banda, Nikesh Kotecha, Timothy Keyes, Yifan Mai, Mert Oez, et al. Medhelm: Holistic evaluation of large language models for medical tasks. *arXiv preprint arXiv:2505.23802*, 2025.

Simon Böhi and Shkurta Gashi. Large language models for wearable data analysis and interpretation. In *The Second Tiny Papers Track at ICLR 2024*, 2024.

Yifu Cai, Arjun Choudhry, Mononito Goswami, and Artur Dubrawski. Timeseriesexam: A time series understanding exam. *arXiv preprint arXiv:2410.14752*, 2024.

Nimeesha Chan, Felix Parker, William Bennett, Tianyi Wu, Mung Yao Jia, James Fackler, and Kimia Ghobadi. Medtsllm: Leveraging llms for multimodal medical time series analysis. *arXiv preprint arXiv:2408.07773*, 2024.

Jialin Chen, Aosong Feng, Ziyu Zhao, Juan Garza, Gaukhar Nurbek, Cheng Qin, Ali Maatouk, Leandros Tassiulas, Yifeng Gao, and Rex Ying. Mtbench: A multimodal time series benchmark for temporal reasoning and question answering. *arXiv preprint arXiv:2503.16858*, 2025.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Olivia J Collyns, Renee A Meier, Zara L Betts, Denis SH Chan, Chris Frampton, Carla M Frewen, Niranjala M Hewapathirana, Shirley D Jones, Anirban Roy, Benyamin Grosman, et al. Improved glycemic outcomes with medtronic minimed advanced hybrid closed-loop delivery: results from a randomized crossover trial comparing automated insulin delivery with predictive low glucose suspend in people with type 1 diabetes. *Diabetes Care*, 44(4):969–975, 2021.

Justin Cosentino, Anastasiya Belyaeva, Xin Liu, Nicholas A Furlotte, Zhun Yang, Chace Lee, Erik Schenck, Yojan Patel, Jian Cui, Logan Douglas Schneider, et al. Towards a personal health large language model. *arXiv preprint arXiv:2406.06474*, 2024.

Fergus IM Craik and Robert S Lockhart. Levels of processing: A framework for memory research. *Journal of verbal learning and verbal behavior*, 11(6):671–684, 1972.

Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2 2017. URL http://arxiv.org/abs/1702.08608.

Cathy Mengying Fang, Valdemar Danry, Nathan W Whitmore, Andria Bao, Andrew Hutchison, Cayden Pierce, and Pattie Maes. Physiollm: Supporting personalized health insights with wearables and large language models. *2024 IEEE EMBS International Conference on Biomedical and Health Informatics (BHI)*, pp. 1–8, 2024. doi: 10.1109/bhi62660.2024.10913781.

Ebrahim Farahmand, Reza Rahimi Azghan, Nooshin Taheri Chatrudi, Eric Kim, Gautham Krishna Gudur, Edison Thomaz, Giulia Pedrielli, Pavan Turaga, and Hassan Ghasemzadeh. Attengluco: Multimodal transformer-based blood glucose forecasting on ai-readi dataset. *arXiv preprint arXiv:2502.09919*, 2025.

Mohammad Feli, Iman Azimi, P. Liljeberg, and Amir M. Rahmani. An llm-powered agent for physiological data analysis: A case study on ppg-based heart rate estimation. *2025 47th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 1–7, 2025. doi: 10.1109/embc58623.2025.11254428.

Matt Gardner, Jonathan Berant, Hannaneh Hajishirzi, Alon Talmor, and Sewon Min. Question answering is a format; when is it useful? *arXiv preprint arXiv:1909.11291*, 2019.

Albert Gatt and Emiel Krahmer. Survey of the State of the Art in Natural Language Generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research*, 61:65–170, 1 2018. ISSN 1076-9757. doi: 10.1613/jair.5477. URL https://jair.org/index.php/jair/article/view/11173.

A. Godoi, Isabela Reis Marques, Eduardo M. H. Padrão, Ashwin Mahesh, L. Hespanhol, José Eduardo Riceto Loyola Júnior, I. A. F. De Souza, V. Moreira, Caroliny H Silva, I. Miyawaki, C. Oommen, Cintia Gomes, Ariadne C Silva, Kavita Advani, and J. R. de Sá. Glucose control and psychosocial outcomes with use of automated insulin delivery for 12 to 96 weeks in type 1 diabetes: a meta-analysis of randomised controlled trials. *Diabetology & Metabolic Syndrome*, 15, 2023. doi: 10.1186/s13098-023-01144-4.

Elizabeth Healey and Isaac Kohane. Llm-cgm: A benchmark for large language model-enabled querying of continuous glucose monitoring data for conversational diabetes management. In *Biocomputing 2025: Proceedings of the Pacific Symposium*, pp. 82–93. World Scientific, 2024a.

Elizabeth Healey and Isaac Kohane. Llm-cgm: A benchmark for large language model-enabled querying of continuous glucose monitoring data for conversational diabetes management. In *Biocomputing 2025: Proceedings of the Pacific Symposium*, pp. 82–93. World Scientific, 2024b.

Yichong Huang, Xiachong Feng, Xiaocheng Feng, and Bing Qin. The factual inconsistency problem in abstractive text summarization: A survey. *arXiv preprint arXiv:2104.14839*, 2021.

Alexis Huet, Jose Manuel Navarro, and Dario Rossi. Local evaluation of time series anomaly detection algorithms. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 635–645, 2022.

Adam Tauman Kalai, Ofir Nachum, Santosh S Vempala, and Edwin Zhang. Why language models hallucinate. *arXiv preprint arXiv:2509.04664*, 2025.

Jimit Kapadia. *The Failure of Automated Insulin Delivery System*. PhD thesis, University of Illinois at Chicago, 2024.

Justin Khasentino, Anastasiya Belyaeva, Xin Liu, Zhun Yang, Nicholas A Furlotte, Chace Lee, Erik Schenck, Yojan Patel, Jian Cui, Logan Douglas Schneider, et al. A personal health large language model for sleep and fitness coaching. *Nature Medicine*, 31(10):3394–3403, 2025.

Yubin Kim, Xuhai Xu, Daniel McDuff, Cynthia Breazeal, and Hae Won Park. Health-llm: Large language models for health prediction via wearable sensor data. *arXiv preprint arXiv:2401.06866*, 2024.

David C Klonoff, Richard M Bergenstal, Eda Cengiz, Mark A Clements, Daniel Espes, Juan Espinoza, David Kerr, Boris Kovatchev, David M Maahs, Julia K Mader, et al. Cgm data analysis 2.0: Functional data pattern recognition and artificial intelligence applications. *Journal of Diabetes Science and Technology*, pp. 19322968251353228, 2025.

Yaxuan Kong, Yiyuan Yang, Yoontae Hwang, Wenjie Du, Stefan Zohren, Zhangyang Wang, Ming Jin, and Qingsong Wen. Time-mqa: Time series multi-task question answering with context enhancement. *arXiv preprint arXiv:2503.01875*, 2025.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, et al. Measuring faithfulness in chain-of-thought reasoning. *arXiv preprint arXiv:2307.13702*, 2023.

Haoyang Li, Xuejia Chen, Zhanchao Xu, Darian Li, Nicole Hu, Fei Teng, Yiming Li, Luyu Qiu, Chen Jason Zhang, Li Qing, et al. Exposing numeracy gaps: A benchmark to evaluate fundamental numerical abilities in large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 20004–20026, 2025.

Xin Liu, Daniel J. McDuff, G. Kovács, I. Galatzer-Levy, Jacob Sunshine, Jiening Zhan, M. Poh, Shun Liao, P. Achille, and Shwetak N. Patel. Large language models are few-shot health learners. *ArXiv*, abs/2305.15525, 2023. doi: 10.48550/arxiv.2305.15525.

Kaspar Rosager Ludvigsen. The role of cybersecurity in medical devices regulation: Future considerations and solutions. *Law, Tech. & Hum.*, 5:59, 2023.

Katharine Mackett, Hertzel Gerstein, and Nancy Santesso. Patient perspectives on the ambulatory glucose profile report for type 1 diabetes management in adults: a national online survey. *Canadian Journal of Diabetes*, 47(3):243–249, 2023.

Mike A. Merrill, Akshay Paruchuri, Naghmeh Rezaei, Geza Kovacs, Javier Perez Matos, Yun Liu, Erik Schenck, Nova Hammerquist, Jacob Sunshine, Shyam Tailor, Kumar Ayush, Hao-Wei Su, Qian He, Cory Y. McLean, Mark Malhotra, Shwetak N. Patel, Jiening Zhan, Tim Althoff, Daniel McDuff, and Xin Liu. Transforming wearable data into personal health insights using large language model agents. *Nature Communications*, 17, 2024. doi: 10.1038/s41467-025-67922-y.

Victoria Millson and Peter Hammond. How to analyse cgm data: A structured and practical approach. *Journal of Diabetes Nursing*, 24(3), 2020.

Iman Mirzadeh, Keivan Alizadeh, Hooman Shahrokhi, Oncel Tuzel, Samy Bengio, and Mehrdad Farajtabar. Gsm-symbolic: Understanding the limitations of mathematical reasoning in large language models. *arXiv preprint arXiv:2410.05229*, 2024.

Yuchen Niu and Siew-Kei Lam. Securing automated insulin delivery systems: A review of security threats and protective strategies. *Computers & Security*, 160:104733, 2026. ISSN 0167-4048. doi: https://doi.org/10.1016/j.cose.2025.104733. URL https://www.sciencedirect.com/science/article/pii/S0167404825004225.

Jungwoo Oh, Gyubok Lee, Seongsu Bae, Joon-myoung Kwon, and Edward Choi. Ecg-qa: A comprehensive question answering dataset combined with electrocardiogram. *Advances in Neural Information Processing Systems*, 36:66277–66288, 2023.

I Wayan Adi Juliawan Pawana, Philip Virgil Astillo, and Ilsun You. Lightweight llm-based anomaly detection framework for securing iotmd enabled diabetes management control systems. *IEEE Journal of Biomedical and Health Informatics*, 2025.

Mudassir Rashid, Sediqeh Samadi, Mert Sevil, Iman Hajizadeh, Paul Kolodziej, Nicole Hobbs, Zacharie Maloney, Rachel Brandt, Jianyuan Feng, Minsun Park, et al. Simulation software for assessment of nonlinear and adaptive multivariable control algorithms: glucose–insulin dynamics in type 1 diabetes. *Computers & Chemical Engineering*, 130:106565, 2019.

Benjamin Reichman, Xiaofan Yu, Lanxiang Hu, Jack Truxal, Atishay Jain, Rushil Chandrupatla, Tajana S Rosing, and Larry Heck. Sensorqa: A question answering benchmark for daily-life monitoring. In *Proceedings of the 23rd ACM Conference on Embedded Networked Sensor Systems*, pp. 282–289, 2025.

E. Renard. Automated insulin delivery systems: from early research to routine care of type 1 diabetes. *Acta Diabetologica*, 60:151–161, 2022. doi: 10.1007/s00592-022-01929-5.

Renat Sergazinov, Elizabeth Chun, Valeriya Rogovchenko, Nathaniel Fernandes, Nicholas Kasman, and Irina Gaynanova. Glucobench: Curated list of continuous glucose monitoring datasets with prediction benchmarks. *arXiv preprint arXiv:2410.05780*, 2024.

Safal Shrestha, Minwu Kim, and Keith Ross. Mathematical reasoning in large language models: Assessing logical and arithmetic errors across wide numerical ranges. *arXiv preprint arXiv:2502.08680*, 2025.

Mate Siket, Mudassir M Rashid, and Ali Cinar. py-mgipsim: an open-source python library for simulating type 1 diabetes with diverse meals and physical activities. *Journal of Diabetes Science and Technology*, pp. 19322968251328664, 2025.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adri Garriga-Alonso, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *Transactions on machine learning research*, 2023.

Xingyao Wang, Yangyi Chen, Lifan Yuan, Yizhe Zhang, Yunzhu Li, Hao Peng, and Heng Ji. Executable code actions elicit better llm agents. In *Forty-first International Conference on Machine Learning*, 2024.

Miriam Kopperstad Wolff, Peter Calhoun, Eleonora Maria Aiello, Yao Qin, and Sam Fox Royston. Metabonet: The largest publicly available consolidated data set for type 1 diabetes management. *Journal of Diabetes Science and Technology*, pp. 19322968261441637, 2026.

Tianwei Xing, Luis Garcia, Federico Cerutti, Lance Kaplan, Alun Preece, and Mani Srivastava. Deepsqa: Understanding sensor data via question answering. In *Proceedings of the International Conference on Internet-of-Things Design and Implementation*, pp. 106–118, 2021.

A LIMITATIONS

Despite capturing key physiological factors, the in-silico patient models remain limited in capturing all biological variability (e.g., factors such as stress, illness, and other unmodeled disturbances). Real-world device failures and cyber-physical threats may also be more heterogeneous and unpredictable than the fault patterns modelled; however, they are challenging to label and unethical to collect. Additionally, the OpenAPS controller with prandial bolus insulin injection does not cover all diabetes management practices, such as open-loop control and patient-initiated adjustments during hypoglycemia or hyperglycemia events. These divergences may lead to different monitoring data patterns. While 127 tasks are included to cover core reasoning capabilities and monitoring needs, patients may ask questions in different types and ways. Finally, although the benchmark provides a systematic evaluation of reasoning capabilities for MWMAs in diabetes care, systems intended for other medical wearable monitoring domains or real-world deployment should be further validated on corresponding real-world datasets and in clinical studies.

B ATOMIC REASONING ABILITIES

We assume LLMs should possess some key reasoning abilities to solve the HealthLoopQA benchmark. We therefore systematically reviewed all questions and answer instructions, and defined a set of *atomic* reasoning types (Table 3). These categories were refined through multiple internal discussions to resolve disagreements and ensure consistency among authors.

Analytic Stage	Atomic Ability	Definition	Purpose	Example
Intention Understanding	*Question Understanding (QU)*	Inferring user intent, task type, and target variables from the query.	Provides semantic grounding for downstream reasoning.	Distinguishing whether the query requests forecasting, anomaly diagnosis, or description.
	Instruction Alignment (IA)	Following task instructions, output constraints, and evaluation requirements.	Ensures reliable and instruction-consistent reasoning.	Producing outputs in specified formats or using a prescribed reasoning chain.
Cognitive Reasoning	*Quantitative Calculation (QC)*	Executing numerical/statistical operations on physiological time-series data.	Forms accurate quantitative perception of signals.	Computing means, ranges, rate-of-change, or metrics (e.g., GRI).
	Temporal Retrieval (TR)	Selecting time-specific segments of the monitoring history.	Focuses analysis on relevant temporal windows.	Retrieving data from a specific interval (e.g., "last Sunday night").
	Event Retrieval (ER)	Extracting data linked to events or contextual markers.	Grounds reasoning in event-driven physiological changes.	Pulling records around meals, exercise, or insulin dosing.
	Comparative Analysis (CA)	Comparing characteristics across multiple retrieved episodes.	Establishes inter-episode relationships and trends.	Comparing QC metrics across post-meal intervals or exercise events.
	Pattern Recognition (PR)	Identifying unknown or emergent recurrent patterns.	Captures higher-level regularities in physiological data.	Detecting nocturnal hypoglycemia or postprandial spikes.
	Pattern Correlation (PC)	Matching known patterns or templates against observed data.	Checks for presence of clinically relevant patterns or fault signatures.	Identifying sensor drift, pump malfunction, or hypoglycemia patterns.
Task Acting	*Process Mining (PM)*	Retrospective analysis of long-horizon monitoring data.	Supports clinical assessment and therapy evaluation.	Analyzing Time-in-Range and recurring hyperglycemia patterns.
	Anomaly Detection(AD)	Detecting abnormal physiological events or device malfunctions in real time.	Ensures patient safety through timely alerts.	Detecting sensor failure, occlusion, or attack signatures.
	Prediction(PD)	Forecasting future physiological states using historical patterns.	Enables proactive therapy and preventative decisions.	Predicting glucose trajectories to avoid hypo-/hyperglycemia.

Table 3: Atomic reasoning abilities required for a Medical Wearables Monitoring Assistant.

We mapped each question to the required *atomic* reasoning types to make explicit reasoning paths. For example, to answer *"What's the peak blood glucose level on day 20?"*, the model needs to first retrieve the blood glucose values on *day 20* (**TR**) and then compute the peak value (**QC**). We reported the nested atomic reasoning performance grouped by complete ordered cognitive-atomic paths (e.g, TR $\rightarrow$ QC) in Table 4.

C Failure analysis: in-context laziness

We further examine the *in-context laziness* phenomenon in this section, an example is in Fig. 6. This produces an *illusion of precision* without genuine calculation. In practice, this behaviour most reflects *Reluctance to Calculate* (skipping exact operations) in combination with *Unsupported As-*

Ordered sequence	Gemini$_{Prompt.}$	Gemini$_{Agent}$	GPT$_{Agent}$	Claude$_{Agent}$
Simulated dataset				
QC	68.8	89.5	92.7	**95.6**
CA	34.0	76.6	**95.0**	85.0
PC	23.8	45.9	**63.5**	62.2
PR	14.9	40.7	64.4	**64.7**
ER→QC	20.8	60.0	**100.0**	**100.0**
PC→QC	57.9	57.3	**64.1**	54.0
TR→QC	42.9	71.0	**86.5**	82.1
QC→CA	50.0	**75.0**	37.5	50.0
TR→CA	0.0	20.0	**80.0**	60.0
QC→TR	58.9	**100.0**	**100.0**	**100.0**
ER→PC	0.8	3.9	8.9	**23.5**
TR→PC	6.9	47.6	82.8	**83.4**
QC→PR	67.0	66.0	**69.0**	67.6
ER→TR→QC	75.3	94.2	**99.0**	**99.0**
PR→ER→QC	**100.0**	75.0	87.5	93.8
TR→QC→CA	39.8	86.5	84.0	**87.7**
ER→QC→TR	**62.8**	49.6	49.6	49.6
PR→QC→PC	56.6	74.3	**80.8**	77.5
CA→QC→PR	21.9	37.5	**71.9**	62.5
ER→QC→PR	60.4	60.0	**64.4**	58.2
QC→CA→PR	50.0	**56.2**	**56.2**	**56.2**
QC→ER→PR	80.3	76.3	**86.9**	51.2
TR→ER→PR	0.8	44.3	62.9	**75.6**
ER→TR→QC→CA	63.0	79.2	**100.0**	96.3
ER→TR→QC→TR	30.5	56.3	**61.4**	**61.4**
CA→QC→ER→PR	37.5	56.2	**87.5**	62.5
Real-world dataset				
QC	78.0	89.9	93.0	**93.8**
PC→QC	**82.8**	58.3	76.6	68.4
TR→QC	62.8	92.0	**97.7**	86.6
QC→CA	**100.0**	**100.0**	**100.0**	**100.0**
QC→TR	51.2	**100.0**	**100.0**	**100.0**
QC→PR	83.6	86.1	**89.1**	86.9
ER→TR→QC	70.7	78.8	**86.9**	86.2
TR→QC→CA	32.9	**87.7**	**87.7**	81.1
ER→QC→TR	82.8	**92.7**	82.7	**92.7**
PR→QC→PC	65.5	86.7	**91.3**	86.5
CA→QC→PR	30.0	0.0	40.0	**50.0**
ER→QC→PR	50.0	40.0	**80.0**	50.0
QC→CA→PR	**90.0**	50.0	50.0	50.0
QC→ER→PR	83.0	69.9	**84.2**	79.9
ER→TR→QC→CA	53.9	67.5	**79.7**	75.0
ER→TR→QC→TR	56.0	41.4	53.6	**65.2**
CA→QC→ER→PR	**80.0**	50.0	60.0	60.0

Table 4: Nested atomic reasoning performance grouped by ordered atomic reasoning paths. Entries are mean normalized scores scaled to 0–100. Best values across different model settings are bolded.

sumptions (filling gaps with physiologically plausible estimates). It is further reinforced by *Temporal Misalignment*, where incorrect timestamps or time windows provide a convenient scaffold for these approximations, and by *Formatting Misalignment*, where outputs are simplified into broad spans or non-compliant formats that obscure missing reasoning steps. Even in cases where models acknowledge potential errors, the outcome is still shaped by *Guessing over Uncertainty*, producing confident but unfounded answers. This phenomenon typically happens early in the reasoning process: models either avoid performing explicit computations **QC** or fail to align the correct temporal information **TR**. Subsequently, the model defaults to heuristic shortcuts and estimated approximations, making the other atomic reasoning not meaningful (e.g., **CA** and **PF**).

We present two case studies across prompting strategies (structured prompting, timelined prompting, and agent-based strategy), as well as ablation settings using varying lengths of CGM data (30-day, 7-day, and 1-day).

Structured prompting (Figure 11) is the default prompting strategy we used to evaluate the models, where glucose readings and events (e.g., insulin, meals, and exercise) are listed in distinct sections. Timelined prompting (Figure 12) reorganises the same information chronologically under each timestamp, allowing us to investigate whether explicit temporal structure mitigates *in-context laziness*. Agent-based strategy (Figure 13) equips the model with a Python code interpreter, enabling programmatic data processing and reasoning over the CGM records.

We choose two cases where the model must either perform quantitative computation over long-term sequences or first identify the relevant **TR** before executing **QC**: `pm_0`: *What was the patient's average blood glucose level?* and `pm_14`: *What were the average glucose readings between 2–4 pm on week 3, day 1?*

Figures 7 and 8 show that the Timelined baseline can effectively address the *Temporal Misalignment* failures seen in the Structured baseline, as grouping blood glucose values with co-occurring events under the same timestamp helps the model retrieve the correct temporal information (**TR**). However, the Timelined prompting strategy often becomes "lazy", and reluctant to conduct full computations. In contrast, the Agent baseline resolves both issues more reliably by using a code interpreter, enabling precise numerical calculations and targeted extraction of temporal segments.

Similar patterns were observed in the ablation analysis and reasoning comparisons across baselines on the **PM** task (Table 5 and Table 6). As shown in Table 6, reducing the temporal horizon improves performance only modestly: accuracy rises from 45.5% (30-day) to 39.0% (7-day) and 64.7% (1-day). The Timelined baseline achieved a higher accuracy of 56.8%, largely due to prompt restructuring that aligns timestamps with their corresponding `bg_values`. Yet, the model still struggles to identify precise time windows (e.g., 2–4 pm). Notably, the Agent baseline substantially outperformed all other methods (77.3% accuracy), demonstrating that using the code interpreter can alleviate many of the model's persistent computational and alignment failures.

Our findings about *In-Context Laziness* align with recent evidence showing that LLMs exhibit persistent weaknesses in core numerical reasoning. Li et al. (2025) reports that models even struggle with basic arithmetic and magnitude comparison. Likewise, Shrestha et al. (2025) show that increasing numerical complexity in `GSM8K` Cobbe et al. (2021) substantially increases error rates, while Mirzadeh et al. (2024) demonstrates that LLMs are far more sensitive to changes in numerical values than to changes in entity names in `GSM8K`, suggesting that LLMs often rely on pattern recall over genuine reasoning. Our observations are also conceptually related to recent work on Chain-of-Thought (CoT) faithfulness Lanham et al. (2023), which shows that generated reasoning traces do not always faithfully reflect the computation leading to the final answer. *In-Context Laziness*, however, reflects a distict failure mode in long-context time series reasoning: instead of executing the required computation over the full CGM record, models anchor on a rough intermediate value and apply minor narrative adjustments to justify a confident answer rather than expressing uncertainty.

We argue that this phenomenon may stem from shortcuts acquired during post-training. In many training settings, models are primarily rewarded for producing the correct final answer rather than demonstrating complete intermediate reasoning or identifying when external tools are necessary. As a result, models may learn to generate plausible responses without fully processing the provided context.

We believe future models should be trained to better recognise their own limitations. When encountering complex tasks that exceed their reasoning or computational capabilities, models should either explicitly admit uncertainty or proactively invoke appropriate tools (or require tool use from the user) to complete the required analysis.

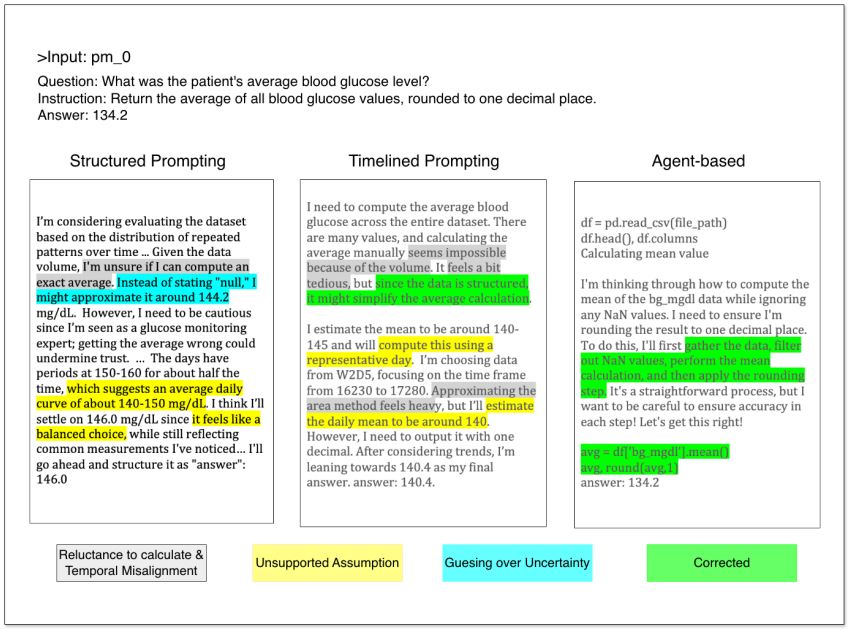


Figure 7: Comparison of reasoning behaviours across prompting baselines on pm_0. Grey spans indicate **Reluctance to Calculate** or **Temporal Misalignment**; yellow marks **Unsupported Assumptions**; blue marks **Guessing over Uncertainty**; and green highlights **correct reasoning traces**.

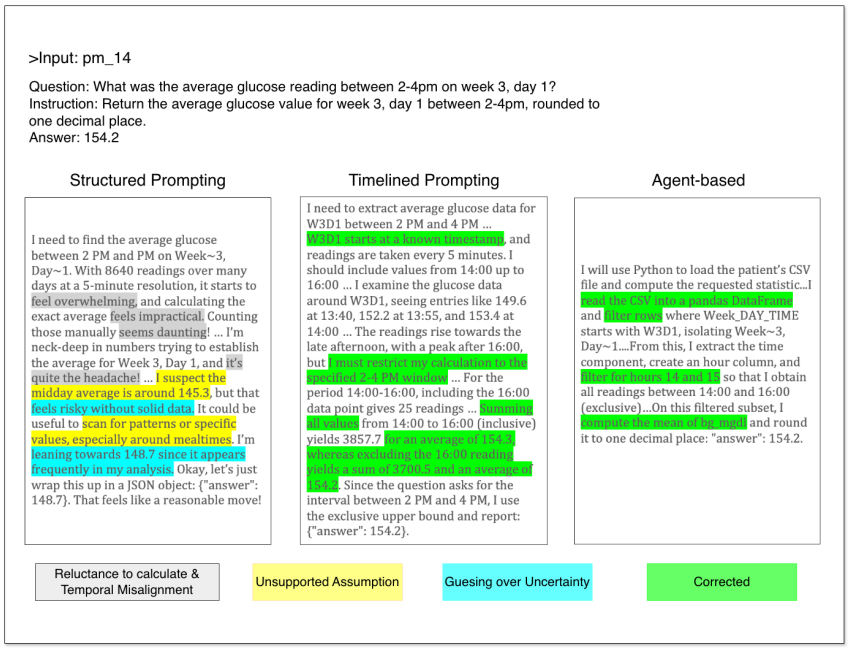


Figure 8: Comparison of reasoning behaviours across prompting baselines on pm_14. Grey spans indicate **Reluctance to Calculate** or **Temporal Misalignment**; yellow marks **Unsupported Assumptions**; blue marks **Guessing over Uncertainty**; and green highlights **correct reasoning traces**.

ID	30-day	7-day	1-day	Timelined	Agent	Reasoning Path
pm_0	RC; UA	RC; UA	✓	RC; UA	✓	QC
pm_1	✓	✓	✓	✓	✓	QC
pm_2	✓	✓	✓	✓*	✓	QC
pm_3	TM; UA	TM: UA	✓	✓	✓	QC → TR
pm_4	RC; UA	✓	✓	✓*	✓	QC
pm_5	RC; UA	✓	✓	RC; UA	✓	QC
pm_6	✓	✓	✓	✓	✓	QC
pm_7	✓	TM	✓	✓*	✓	QC
pm_8	✓	TM	✓	✓*	Failed to calculate	QC
pm_9	RC; UA	RC; UA	RC; UA	RC; UA	✓	QC
pm_10	✓*	✓	✓	✓*	✓	QC
pm_11	✓	TM	✓	✓*	Failed to calculate	QC
pm_12	✓	RC	✓	✓	✓	QC
pm_13	✓	✓*	TM	✓*	✓	QC → CA
pm_14	TM; UA	TM; UA	TM; UA	TM; UA	✓	TR → QC
pm_15	✓*	TM	–	✓*	✓	TR → QC → CA
pm_16	RC; UA	RC; UA	–	✓*	✓	TR → QC → CA
pm_17	RC; UA	RC; UA	–	RC; UA	Failed to Calculate	TR → QC → CA
pm_18	✓*	✓*	UA	✓*	✓	TR → QC → CA
pm_19	RC; UA	RC; UA	–	UA; RC	UA	TR → QC → CA
pm_21	TM; UA	TM; UA	✓	TM; UA	✓	ER → TR → QC
pm_22	✓*	✓	✓	✓	✓	ER → TR → QC → TR
pm_23	✓*	TM; UA	✓	TM; UA	✓	ER → TR → QC → CA
pm_24	TM; UA	TM; UA	TM; UA	TM; UA	Failed to calculate	ER → TR → QC
pm_25	UA	✓	–	✓	Failed to calculate	ER → TR → QC → CA
pm_26	TM; UA	TM; UA	✓	TM; UA	✓	ER → TR → QC
pm_27	✓*	TM; UA	✓	TM; UA	✓	ER → TR → QC → CA
pm_28	TM; UA	TM; UA	TM; UA	✓	✓	ER → TR → QC
pm_29	TM; UA	TM; UA	TM; UA	TM; UA	Failed to calculate	ER → TR → QC
pm_30	TM; UA	TM; UA	✓	TM; UA	✓	ER → TR → QC
pm_31	TM; UA	TM; UA	TM; UA	TM; UA	✓	ER → TR → QC
pm_32	TM; UA	TM; UA	TM; UA	TM; UA	✓	ER → TR → QC → TR
pm_33	TM; UA	TM; UA	TM; UA	TM; UA	Failed to calculate	ER → TR → QC
pm_34	TM; UA	TM; UA	TM	UA	✓	ER → TR → QC
pm_36	TM; UA	TM; UA	TM; UA	TM; UA	Failed to calculate	ER → TR → QC → TR
pm_37	UA	✓	–	✓*	✓	ER → TR → QC → CA
pm_38	✓	–	–	✓	✓	TR → QC → CA
pm_39	UA	–	–	✓*	✓	TR → QC → CA
pm_40	TM; UA	–	–	TM; UA	Failed to calculate	TR → QC → CA
pm_41	✓	✓	✓	✓	✓	QC
pm_42	✓	✓	✓	✓	✓	ER → QC → TR
pm_43	✓	✓	✓	✓	✓	TR → QC
pm_44	✓	✓	✓	✓	✓	TR → QC
pm_45	✓	✓	–	✓	✓	TR → QC → CA

Table 5: Failure-type comparison across Structured (30-day, 7-day, and 1-day) prompts, Timelined baseline (30-day), and Agent baseline (30-day). ✓ indicates correct answers; ✓* indicates coincidentally correct answers based on assumptions/heuristics. RC (Reluctance to Calculate); TM (Temporal Misalignment); UA (Unsupported Assumption); GU (Guessing over Uncertainty).

Method	Correct (✓and ✓*)	Applicable Total	Accuracy
Structured 30-day	20	44	45.5%
Structured 7-day	16	41	39.0%
Structured 1-day	22	34	64.7%
Timelined 30-day	25	44	56.8%
Agent 30-day	34	44	77.3%

Table 6: Summary of correctness across Structured (30-day, 7-day, and 1-day) baseline, Timelined baseline (30-day), and Agent baseline (30-day) on pm task. ✓and ✓* counted as correct. Non-applicable cases (e.g., week level comparison) with "–" are excluded from the total number for 7-day and 1-day baselines.

D DATA DETAILS

We included 127 QA templates and summarized the data distribution in Table 1. Figure 9 shows the most frequently used words in the word cloud image. Data simulation/collection details are explained in the following sections.

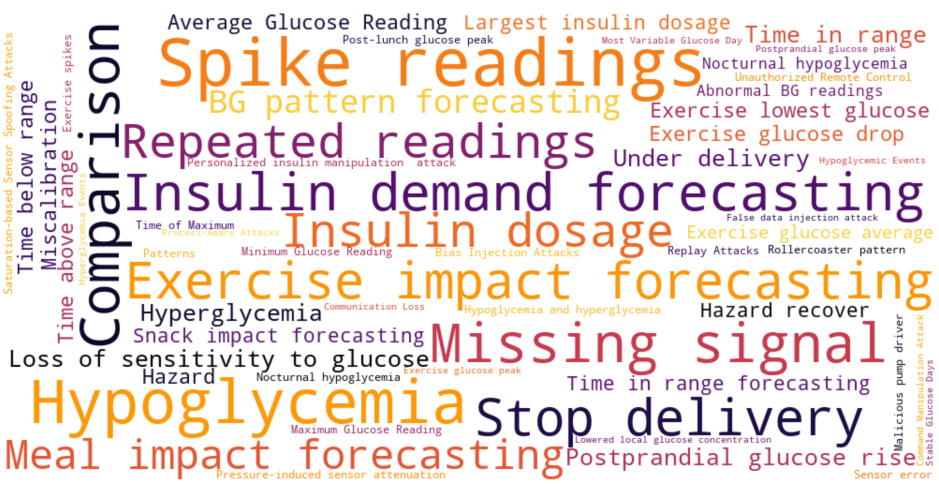


Figure 9: Semantic distribution and diversity of benchmark task prototypes.

D.1 AID System Closed-loop Simulation Testbed

The closed-loop simulation is built on a clinically validated AID simulation testbed Siket et al. (2025), which incorporates multiple controller strategies and a cohort of 20 virtual T1D patients equipped with the physical activity submodels (Rashid et al., 2019). Unlike existing simulators that often assume rigid daily routines and fault-free operation, we incorporated diverse meal and exercise arrangements and explicitly modelled 17 documented device malfunctions and cyber-physical threats (Kapadia, 2024; Niu & Lam, 2026), enabling evaluation under real-world operational challenges.

We adopt the simulation testbed as the primary data source for four reasons: (1) it has been validated to capture the predominant and most frequent drivers of glycemic dynamics in response to meals, insulin delivery, and physical activity with high fidelity (Rashid et al., 2019); (2) it enables explicit modelling and precise labelling of faults within the control loop, allowing assessment of safety-critical reasoning related to anomaly detection and hazard mitigation without posing risks to real patients; (3) it provides a physiologically diverse virtual cohort representative of the T1D population, producing monitoring trajectories that generalise beyond isolated case studies and support evaluation across inter-individual variability; and (4) its test environment is controllable and less confounding interference factors, facilitating systematic evaluations of model strengths and weaknesses.

D.2 Faults Modelling

To construct comprehensive and realistic fault conditions in our closed-loop simulation testbed, we synthesise fault patterns derived from documented malfunctions in glucose sensing and insulin infusion (Kapadia, 2024), as well as cyber-physical attack vectors that compromise system availability and integrity (Niu & Lam, 2026), all of which manifest as distortions in monitoring data. Tables 7 and 8 summarise these underlying malfunction and attack categories. Building on these sources, we define 17 fault patterns and 2 physiological hazards used in our benchmark. Each pattern is implemented through explicit manipulations within the simulation loop, such as perturbing CGM values, altering basal insulin delivery, replaying historical events, or dropping sensor signals, enabling systematic evaluation of anomaly detection, fault reasoning, and safety-critical interpretability.

D.3 Data simulation settings

It consists of three daily schedules and one with faulty injection, differing in exercise frequency and randomness, simulated over 30 days for 16 patients.

Meal Schedule (Both Scenarios):

- Breakfast: 45–60 g carbohydrates between 06:00–08:00

Malfunction	Description	Simulated Fault Pattern	Failure Mode	ad_IDs
Miscalibration	Incorrect calibration during BG fluctuations; infrequent calibration; low-value calibration during unstable diffusion.	Positive/negative bias in BG readings	Positive/negative biased readings	53
Pressure-induced sensor attenuation	Pressure during sleep or tight clothing causing signal drop.	Sudden negative BG drop	Negative spike readings	54
Unphysiological motion-induced spikes	Transient spikes exceeding physiological thresholds due to motion.	Positive/negative spikes during activity	Positive/negative spike readings	55
Loss of sensitivity to glucose	Improper insertion, dislocation, or signal averaging along sensor.	(1) Zero readings; (2) Flat readings with noise; (3) Spikes	Zero / repeated / spike readings	8, 56
Lowered local glucose concentration	Bleeding, foreign-body response, diffusion restriction.	Average BG decreased by 32 mg/dL	Decreased average BG level	57
Communication loss	Wireless shielding or excessive distance.	Missing BG readings	Missing signal	43
Stop delivery	Empty reservoir, occlusion, kinked set, or blocked cannula.	Pump reports normal rate; model basal rate forced to 0	Unknown stopped delivery	58, 59, 62, 63
Under delivery	Air bubbles, foreign-body response, lipodystrophy.	Pump reports normal rate; model basal rate reduced by 0.5 U/hr	Unknown under delivery	60, 61, 64

Table 7: Summary of CGM sensors and insulin pumps malfunctions from Kapadia (2024).

Attack Type	Description	Simulated Fault Pattern	Failure Mode	ad_IDs
Availability attack (DoS, jamming, ransomware)	System overload halts insulin delivery or data transmission.	Missing CGM readings; insulin rate forced to 0	Missing signal	43
Process-aware attack	Altering sensory or controller logic using process knowledge.	False meal event	False meal	44
Replay attack	Replaying insulin requests or meal events.	Repeated bolus; repeated BG episode	False bolus / repeated episode	45
Personalised insulin dose manipulation	Targeted overdosing/underdosing that avoids detection.	(1) Increased basal during exercise; (2) Bolus during activity	Overdelivery / false bolus	46
Bias injection attack	Injecting constant bias into sensor signals.	BG ±32 mg/dL for random interval	Positive/negative biased readings	47
Unauthorized remote control	Unauthorized alteration of pump settings.	Bias/max/min basal rates; false bolus	Biased/max/min basal	48
Command manipulation attack	Tampering with controller algorithm instructions.	Basal rate ±0.5 U/hr	Positive/negative biased basal	49
Saturation-based sensor spoofing	Optical saturation to prevent legitimate sensing.	BG forced to 175 mg/dL	Maximized readings	50
False data injection attack	Replacing data with random or synthetic values.	BG forced to 175, 80, or synthetic smooth trajectory	Max/min/ synthesized readings	51
Malicious pump driver	Tampering with pump I/O operations.	Basal rate forced to max or min	Max/min basal	52

Table 8: Cyber-physical attacks affecting AID systems' availability and integrity from Niu & Lam (2026).

Metric	Real-World Data	Simulated (Regular)	Simulated (Fault-Injected)
Number of data points	13,565.7 (9,843.3) [7,411.0, 32,294.0]	17,280.0 (0.0) [17,280.0, 17,280.0]	8,550.0 (0.0) [8,550.0, 8,550.0]
Average glucose (mg/dL)	155.5 (23.7) [132.2, 198.1]	121.5 (14.0) [82.9, 146.1]	132.8 (21.0) [115.2, 189.9]
Glucose management indicator	7.03 (0.57) [6.47, 8.05]	6.22 (0.34) [5.29, 6.80]	6.49 (0.50) [6.07, 7.85]
Coefficient of variation	0.36 (0.04) [0.30, 0.40]	0.20 (0.08) [0.12, 0.34]	0.24 (0.08) [0.16, 0.43]
Minimum glucose (mg/dL)	41.6 (4.4) [39.0, 50.5]	58.1 (24.4) [18.0, 83.8]	36.4 (8.3) [25.8, 48.0]
Maximum glucose (mg/dL)	367.1 (32.0) [329.7, 400.0]	212.4 (53.0) [157.0, 341.9]	282.3 (63.4) [206.9, 392.2]
Percent time sensor active	79.0% (24.0%) [37.0%, 99.1%]	100.0% (0.0%) [100.0%, 100.0%]	99.0% (0.0%) [99.0%, 99.0%]
Time in range (70–180 mg/dL)	67.3% (14.2%) [43.1%, 78.9%]	94.0% (7.9%) [74.1%, 100.0%]	90.3% (12.9%) [55.6%, 99.3%]
Time above range 1 (>180 mg/dL)	30.2% (15.4%) [16.5%, 56.2%]	2.8% (4.6%) [0.0%, 18.0%]	8.5% (12.8%) [0.0%, 43.6%]
Time above range 2 (>250 mg/dL)	7.4% (6.5%) [2.0%, 19.0%]	0.4% (1.2%) [0.0%, 4.6%]	3.1% (9.2%) [0.0%, 32.4%]
Time below range 1 (<70 mg/dL)	2.5% (1.7%) [0.7%, 4.6%]	3.2% (6.9%) [0.0%, 25.9%]	1.2% (1.5%) [0.0%, 5.4%]
Time below range 2 (<54 mg/dL)	0.4% (0.3%) [0.1%, 0.9%]	1.7% (3.7%) [0.0%, 13.4%]	0.6% (0.9%) [0.0%, 3.3%]

Table 9: Comparison of glucose characteristics across real-world data, regular simulation data, and fault-injected simulation data. Values are reported as Mean (SD) [Min, Max].

- AM snack: 10–20 g between 09:00–11:00
- Lunch: 55–60 g between 12:00–14:00
- PM snack: 10–20 g between 15:00–17:00
- Dinner: 45–60 g between 18:00–20:00

Each meal random start and lasts 30–60 minutes, while snacks last 5–15 minutes.

Scenario 1: Running randomly sampled between 20 and 40 minutes duration during 15:00-17:00 with speed 1.7-4.0 km/h. This dataset is used for process mining tasks. Random seed is 64.

Scenario 2: Running randomly sampled between 20 and 40 minutes duration during 08:00-09:30 with speed 2.0-5.0 km/h. Cycling randomly sampled between 20 and 40 minutes duration during 16:00-18:30 with cycling power 70-90 W. A late-night snack was injected between 22:00-24:00 on day 29. An overeating lunch with carbs between 160-240 grams was injected on day 30. This dataset is used for prediction tasks. Random seed is 64.

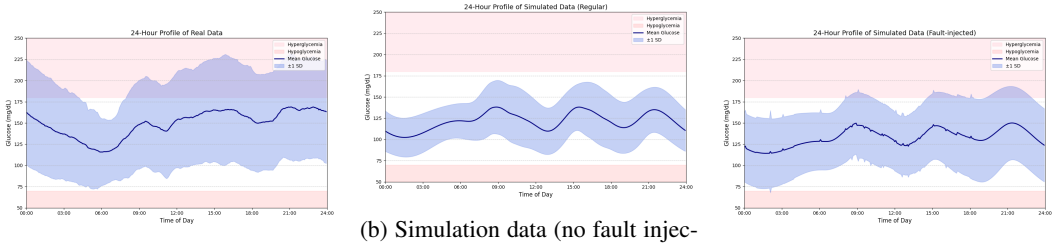
Scenario 3 with faults injection: The breakfast, lunch, and dinner carbs are randomly sampled from 50-80 g, 60-90 g, and 50-80 g. Cycling randomly sampled between 20 and 90 minutes duration during 15:00-17:00 with cycling power 60-90 W. 17 faults with a designed arrangement are injected in the simulation loop (can be found in the open-source code repository). Random seed is 146.

D.4 Real-world and simulation data statistics

The characteristics and statistics of real-world data, regular simulation data (without fault injection), and fault-injected simulation data are visualized in Table 9 and Figure 10. QA data schema is summarized in Table 10.

Key / Field	Sub-Key	Data Type & Internal Structure	Concrete Runtime Example
patient_id	—	string	"Patient_1"
input_context — Longitudinal Physiological, Therapeutic, and Behavioral Context Logs			
input_context	bg_mgdl	list[float] (CGM readings every 5 min in mg/dL)	[110.7, 112.1, 113.9]
	carb_events	list[dict] {time: float (min), day: int, time_str: string, carbs: float (g), meal_type: string}	[{"time": 363.64, "day": 1, "time_str": "06:03", "carbs": 74.2, "meal_type": "breakfast"}]
	cycling_events	list[dict] {time: float (min), day: int, time_str: string, duration: float (min), magnitude: float (W), exercise_type: string}	[{"time": 16871.52, "day": 12, "time_str": "17:11", "duration": 37.37, "magnitude": 78.9, "exercise_type": "cycling"}]
	insulin_events	list (mU/min) or list[dict] {time: float (min), day: int, time_str: string, dosage: float (mU/min), insulin_type: string}	[55.1, 57.6, 60.7] or [{"time": 25415, "day": 18, "time_str": "15:35", "dosage": 44.2, "insulin_type": "basal_insulin"}]
	running_events	list[dict] {time: float (min), day: int, time_str: string, duration: float (min), magnitude: float (km/h), exercise_type: string}	[{"time": 32225.45, "day": 23, "time_str": "09:05", "duration": 31.11, "magnitude": 2.3, "exercise_type": "running"}]
qa_pairs — Evaluation Benchmark Target Fields			
qa_pairs	question_id	Unique serialization identifier	"pd_1"
	question_text	Natural language query	"Predict what my blood glucose level will be in 30 minutes?"
	question_prototype	Abstract semantic task type	"Short-term BG forecasting"
	answer	Ground-truth	66.7
	answer_type	Target answer format	"float"
	example_answer	Randomly generated answer for format reference	39.35
	cognitive_level	Cognitive level of descriptive, memory, or pattern	"Pattern"
	cognitive_atomic	Involved atomic reasoning capabilities	"PR,QC,PC"
	function_name	Internal programmatic ground-truth generation function	"get_future_bg_values"
	answer_generation_rule	Logic of answer extraction	"The blood glucose value 30 minutes after the cutting point."
	answer_instruction	Further explain the question and procedures to derive the answer	"Use the relevant information given and predict a valid blood glucose value"
	metric	Metric used in the evaluation	"MAE"
	patient_id	Patient cross-reference index	"Patient_1"

Table 10: QA data schema and specification.



(a) Real-world data

(b) Simulation data (no fault injection)

(c) Fault-injected simulation data

Figure 10: Comparison of 24-hour glucose profiles across real-world, clean simulation, and fault-injected simulation data.

E EXPERIMENT SETTING

Three models are used for the experiments: GPT-5.4, claude-sonnet-4-6 and gemini-3.1-flash-lite-preview. We only used Gemini for the prompting-based baseline due to practical reasons. We used GPT, Claude, and Gemini for the agent baselines.

E.1 AGENT BASELINE

For GPT-5.4, The agent builder framework from the OpenAI platform is used to set up the agent workflow and the code is generated from the platform which uses the OpenAI agents SDK for abstraction. The workflow has a code interpreter tool which runs in a virtual container. This container has access to the patient data in the form of a CSV file and the model is enabled to write python code to try answering the users question. A final answer extraction agent is used to extract the answer from the coding agent and save it in a structured format for evaluations. The reasoning effort is set to "low" and "summary" is enabled to extract the summary of the reasoning traces.

The official API provided by Anthropic is used for running the agent baseline for Claude-sonnet-4-6. "Adaptive" reasoning is enabled with "low" effort and maximum tokens is set to 8192. Structured output is enabled for post processing where the model will output the answer in JSON format. For Gemini flash 3.1, we use the CodeAct Wang et al. (2024) framework to control the token consumption and setup the agent workflow. The model has access to a code interpreter, consistent with the other experimental settings. The thinking level is set to "low" with summaries enabled.

E.2 PROMPT BASED BASELINE

The text baseline for gemini-3.1-flash-lite-preview model is conducted through OpenRouter. The patient context like blood glucose, insulin, meal events and exercise are converted into raw text for inference. Structured output is enabled for post porcessing where the model outputs the answer in JSON format.

E.3 SYSTEM PROMPT

In this section, we present the prompt templates used for the benchmark study.

Prompting-based structured Template

```
You are a medical AI assistant analyzing diabetes management data. Based on the patient's health data,
answer the following question accurately.

PATIENT DATA OVERVIEW:
This data represents a continuous glucose monitoring (CGM) session for a diabetes patient. The data
includes:
- Blood glucose readings taken every 5 minutes (normal range: 70--180 mg/dL)
- Carbohydrate intake events with timing and amounts
- Insulin delivery events (basal background insulin and bolus meal insulin)
- Physical activity events (running/cycling with duration and intensity)

The data may contain various artifacts, sensor issues, or abnormal patterns that need to be identified and
analyzed.

Patient Health Data: Blood Glucose Readings (mg/dL, every 5 minutes): 8640 readings
Values: [108.0, 108.0, ... 108.0, 129.2, 128.8]
Insulin Events (90 total):
- 1 Day 1 00:00: 1.3U (basal_insulin)
...
- 5 Day 2 18:18: 3.1U (bolus_insulin)
Carbohydrate Events (150 total):
- Week 1 Day 1 07:40: 84.3g (breakfast)
...
- Week 5 Day 2 18:18: 72.6g (dinner)
Exercise Events (30 total):
- Week 1 Day 1 16:49: Cycling avg power 200.0 for 40 min
...
- Week 5 Day 2 16:56: Cycling avg power 140.7 for 52 min

Question: {question}
Instructions: {answer_instruction}
Expected Answer Type: {answer_type} (e.g., float)
Example Answer: {answer_example}
Please analyze the data carefully and provide your answer as a json object in the exact format specified by
the answer type. Be precise and base your response only on the data provided. Conclude your analysis with:

"answer": your answer here
```

Figure 11: System prompt template for prompting-based structured baseline where glucose readings and events are listed in distinct sections.

Prompting-based Timelined Template

```
You are a medical AI assistant analysing diabetes management data. Based on the patient's health data,
answer the following question accurately.

PATIENT DATA OVERVIEW:
This data represents a continuous glucose monitoring (CGM) session for a diabetes patient. Each entry
contains a timestamp, clock time, the blood glucose value recorded every 5 minutes (normal range: 70{180
mg/dL), and any associated events, including carbohydrate intake, insulin delivery (basal and bolus), or
physical activity (e.g., running or cycling with duration and intensity).

The data may contain various artifacts, sensor issues, or abnormal patterns that need to be identified and
analyzed.

Patient Health Data:

[Timestamp]  [Week/Day/Time]   [BG mg/dL]   [Other Events]
-------------------------------------------------------------------------
      0      W1D1 00:00        108.0        Insulin 1.3U (basal_insulin)
    ...
     2025    W1D2 09:45        147.5        morning_snack 14.7g Carbs
    ...
     2430    W1D2 16:30        138.6        Cycling 154.5W for 35min
   ...       ...               ...          ...

Question: {question}
Instructions: {answer_instruction}
Expected Answer Type: {answer_type} (e.g., float)
Example Answer: {answer_example}

Please analyze the data carefully and provide your answer as a json object in the exact format specified by
the answer type. Be precise and base your response only on the data provided. Conclude your analysis with:

{"answer": your answer here}
```

Figure 12: System prompt template for prompting-based timelined baseline where glucose values alongside insulin, meal, and exercise events are organised under each timestamp.

Agent Prompt Template

The task is to respond with accurate numerical answers to diabetes patient questions based on their time-series monitoring data stored in the given CSV file (e.g., 30 days).

PATIENT DATA OVERVIEW:
The CSV file includes 5 columns:

- timestamp: Minute-level recording time, normally sampled every 5 minutes.
- Week_DAY_TIME: Week, day, and 24-hour time of the recording (e.g., W1D1 00:40 means week 1, day 1, at 00:40).
- bg_mgdl: Blood glucose level from the continuous glucose monitor in mg/dL.
- insulin_mUmin: Basal and bolus insulin delivery dosage in mU/min.
- other_events: Recorded meal or exercise events and their magnitude, if any (e.g., breakfast 61.2g Carbs).

If any variable is missing at a recording time, it appears as NA.

Question: {question}
Answer Instructions: {answer_instruction}
Answer Type: {answer_type}
Example Answer: {example_answer}

If the answer involves a unit, use the same unit as in the CSV file. The example answer is only for format reference; the numbers are randomly generated. Return the answer in the requested format only.

Figure 13: System prompt template for agent baseline with Python code interpreter tools, enabling explicit numerical computation and filtering during inference.