AI Agent

Martin Demel

Department of Science, Technology, Engineering & Math, Houston Community College

ITAI 2376 Deep Learning in Artificial Intelligence

Patricia McManus

May 7th, 2025.

**Introduction**

I am pleased to present the final report of the Pulsum Wellness App, developed as a capstone project in ITAI 2376 Deep Learning in Artificial Intelligence. The primary objective was to create an AI-driven platform that interprets raw wearable health data and transforms it into actionable insights and personalized recommendations. Pulsum addresses the common dilemma in which individuals can access substantial biometric information, such as heart rate variability and glucose levels, yet lack a structured framework for analyzing and acting upon that data.

In this report, I will describe Pulsum’s functionalities, explain its multi-agent architecture, and discuss the results of its testing and refinement. Following a summary of the significant challenges tackled, I will reflect on the lessons learned and outline avenues for future improvement. Ultimately, I believe this project demonstrates how AI agents, organized within a robust system design, can elevate personal health data from raw metrics to meaningful guidance that fosters better decision-making.

**Body**

**Project Overview**

The Pulsum Wellness App is designed to be a personalized health companion, integrating data from the Oura Ring and optionally from Dexcom continuous glucose monitors. It offers a unified user interface that displays trends and correlations, enabling individuals to see how aspects of their lifestyle interact with their physiological metrics. By combining objective data, including heart rate variability, activity levels, and glucose readings, with subjective data, such as daily mood ratings or journaling entries, Pulsum provides a holistic view of each user’s overall well-being.

My intention in creating Pulsum was to move beyond static dashboards by incorporating an AI-driven insight engine that could identify patterns in health metrics and generate micro-actions to address the user’s specific needs. These micro-actions, stored in a curated database, include practical lifestyle tips that have shown efficacy in scientific studies or are grounded in expert recommendations. Users can interact with Pulsum via chat to ask questions about their data or seek tailored advice. The system’s reinforcement learning component ensures that user preferences and feedback continually refine the recommendations, creating a dynamic loop that personalizes suggestions based on individual experiences.

**System Architecture**

Pulsum’s architecture is based on a multi-agent approach, designed to break down the complexity of health data processing into smaller, more manageable tasks.

1. **Manager Agent**

Coordinates user interactions and routes tasks to the appropriate sub-agent. It is the main entry point for user requests, including check-ins and chat queries.

1. **Pattern Detection Agent**

It examines Oura and Dexcom data to pinpoint significant patterns, such as correlations between late-night activities, elevated heart rates, or glucose spikes. Its output helps identify the factors most strongly impacting a user’s health.

1. **Sentiment and Journal Agent**

Processes daily mood and free-text journal entries, contributing subjective data that may indicate stress, low energy, or improved well-being. This qualitative dimension adds vital context to otherwise numeric health metrics.

1. **Recommendation Agent**

Translates the combined insights from the Pattern Detection and Sentiment agents into practical suggestions, primarily sourced from a curated micro-action library. The agent generates AI-based content as a fallback if a micro-action is unavailable.

1. **Safety Agent**

Screens all responses for indicators that might suggest the user is in crisis or is seeking medical advice beyond the system’s scope. It also monitors for inappropriate or off-topic content, acting as a safeguard to ensure Pulsum remains within ethical boundaries.

The Pulsum interface was developed as a three-column web application built in React, providing distinct spaces for data visualizations, journaling and chat, and personalized recommendations. Data visualizations rely on Chart.js to dynamically present trends in the user’s biometrics, while the journaling and chat interface captures user inputs and passes them to the relevant agents. The recommendation column displays daily summaries and actionable micro-actions, allowing users to mark suggestions as completed or express a preference through likes and dislikes.

On the back end, a Node.js server orchestrates API calls to Oura and Dexcom, managing user authentication and data storage. The Manager Agent controls the conversation flow with the user, sometimes calling specialized agents for pattern detection, sentiment analysis, or recommendation generation. The multi-agent logic is organized into discrete modules, each with well-defined responsibilities, which proved beneficial for maintainability and scalability. All health data and user interactions remain local on the machine by default, aligning with privacy best practices and making it easier to store personally identifiable information securely without relying on external databases. OpenAI’s API underpins the natural language processing and generative text features, allowing the system to interpret user queries, produce chat responses, and generate fallback recommendations. Throughout development, I made careful use of environment variables to protect credentials and maintain a clear separation between local logic and external services.

**Evaluation Results**

I tested Pulsum extensively on real Oura data, simulating various stress levels and journaling patterns over a four-week period. During these tests, the Pattern Detection Agent successfully highlighted correlations, such as a consistent relationship between late-night screen time and lower readiness scores the next day. In parallel, the Dexcom integration indicated spikes in glucose following a specific meal routine, prompting the system to recommend taking an evening walk to stabilize blood sugar levels. Users reported finding the daily summaries especially helpful, as it consolidated key metrics at a glance and linked them to practical lifestyle recommendations.

The system demonstrated near-instantaneous response times for chat messages, averaging one second, while daily summary generation took under three seconds in most cases. Anecdotal feedback showed that about 80 percent of recommendations felt relevant or personalized, confirming that the use of curated micro-actions combined with LLM-based generation delivered targeted advice. Engagement with the feedback loop grew steadily, with disliked suggestions eventually phased out or refined, and liked actions reappearing more often, validating the reinforcement learning approach. Users also appreciated the safety guardrails, which appropriately directed them to outside resources if they typed messages suggestive of severe distress or medical emergencies outside the system’s domain.

**Challenges and Solutions**

Developing a cohesive multi-agent framework for Pulsum posed a range of challenges. One of the earliest hurdles was synchronizing data from multiple sources, including Oura and Dexcom, which required standardizing timestamps and health metrics in a common format. I resolved this by implementing a robust extraction, transformation, and loading pipeline that indexed data by date to permit correlations across different metrics. Another challenge involved ensuring that AI-generated advice did not overshadow or conflict with the curated micro-action library. This was addressed by prioritizing micro-actions for user states that matched pre-vetted entries, and only invoking open-ended generation if no suitable match was found, with final checks performed by the Safety Agent. Additionally, fulfilling the reinforcement learning requirement within a short timeline demanded a simple but effective approach, relying on user preference tracking rather than implementing a complex RL algorithm, which allowed me to demonstrate policy improvement without overextending the scope of the project.

**Lessons Learned**

One of the key lessons I gleaned from this project was the considerable value of specialized agents for complex, real-world tasks. By distributing responsibilities among the Pattern Detection, Sentiment, Recommendation, and Safety Agents, I gained a modular system that could be extended or refined without disrupting other components. Equally, I discovered that user acceptance of AI-driven recommendations relies heavily on transparency. Presenting data in a clear, visually appealing format helped users see the connection between their metrics and the system’s suggestions, thereby increasing their motivation to follow the advice offered. I also learned that even a basic reinforcement learning mechanism can significantly improve user experience, highlighting the potential of incremental policy improvement strategies.

**Conclusion**

Developing Pulsum was a rigorous and rewarding journey that united the theoretical foundations of AI agents, reinforcement learning, and advanced data analysis with pragmatic design and development decisions. By applying a multi-agent architecture, I successfully created a platform that aggregates wearable health data, identifies key correlations, and offers personalized micro-actions to users. The project’s success lies in how each specialized agent carefully processes a slice of the workload, from analyzing day-to-day metrics to moderating the chat interface for safety.

Looking ahead, I can see several ways to enhance Pulsum’s capabilities. Incorporating a more advanced predictive component could enable the system to forewarn users about looming fatigue or stress patterns. Adding new data sources, such as other fitness wearables, would give the system a broader view of the user’s health behavior. Strengthening the personalization element with deeper machine learning models, incorporating user demographics, and applying more sophisticated RL algorithms could further refine the platform’s adaptability to each individual’s habits and preferences. In the longer term, integrating community-based features could enrich the user experience by allowing optional data sharing with family members or healthcare providers, thus enabling collaborative health management.

In final reflection, this project underscores the possibilities of combining robust data processing pipelines with carefully orchestrated AI agents to offer daily, real-world benefits. Pulsum highlights the importance of delivering not merely data but also actionable knowledge, inviting users to engage with their health on a deeper, more informed level. I believe that with continued development, Pulsum can evolve into an indispensable health companion, capable of guiding and motivating users toward sustained well-being.

Resources

OpenAI Agents SDK. (n.d.). <https://openai.github.io/openai-agents-python/>

GeeksforGeeks. (2025, April 3). Agents in AI. GeeksforGeeks. <https://www.geeksforgeeks.org/agents-artificial-intelligence/>

GeeksforGeeks. (2025, March 10). AI Agent Framework: a platform for intelligent agents. GeeksforGeeks. <https://www.geeksforgeeks.org/what-is-ai-agent-framework/>

GeeksforGeeks. (2025, February 24). Reinforcement learning. GeeksforGeeks. <https://www.geeksforgeeks.org/what-is-reinforcement-learning/>

GeeksforGeeks. (2024, May 28). Optimization algorithms in machine Learning. GeeksforGeeks. <https://www.geeksforgeeks.org/optimization-algorithms-in-machine-learning/>