DataCleaningAgent

Overview

The **DataCleaningAgent** is an agentic AI-driven workflow implemented in n8n for automated, intelligent data cleaning, validation, and summarization of tabular datasets sourced from Google Sheets. It leverages multiple specialized agents orchestrated via an agentic pattern to improve data quality at scale.

High-Level Architecture

- **Source:** Google Sheets input as data source.
- **Orchestrator Agent:** Routes rows for imputation, validation, or summarization.
- **Imputer Agent:** Uses LLMs to fill missing/invalid values with best guess and method.
- **Validator Agent:** Uses LLMs to verify data integrity based on cleaning logs.
- **Summarizer Agent:** Generates summaries and audit trails for cleaned rows.
- **Memory and Persistence:** Cleaning logs and edits are persisted back into sheets and optionally vector stores for long term knowledge and retrieval.
- **Output:** Cleaned, annotated rows written back to Google Sheets for analytics/ML readiness.

Agentic Concepts

- **Tools:** Google Sheets nodes, HTTP LLM nodes, function/code nodes.
- **Resources:** Input data, cleaning logs, validation status as shared contextual info.
- **Agents:** Multiple LLM-driven agents specialized by task (impute/validate/summarize).
- **Agentic Workflow:** Orchestrator delegates subtasks with context, receives and merges results.

- **Handoffs:** Structured JSON row handoffs with route tags between agents.
- **Guardrails:** Prompt-level instructions to confine each agent's scope and output format.
- **Memory:** Tracking row-level cleaning history within `_cleaningLog` and optional vector embeddings.
- **Persistence:** All final cleaned data and logs persist on Google Sheets as the single source of truth.
- **Vector Storage:** (Planned) Embedding storage for rapid contextual retrieval in future.
- **Explainability:** All agents produce human-readable logs explaining changes.
- **Extensibility:** Modular workflows support easy addition of new agents or external APIs.

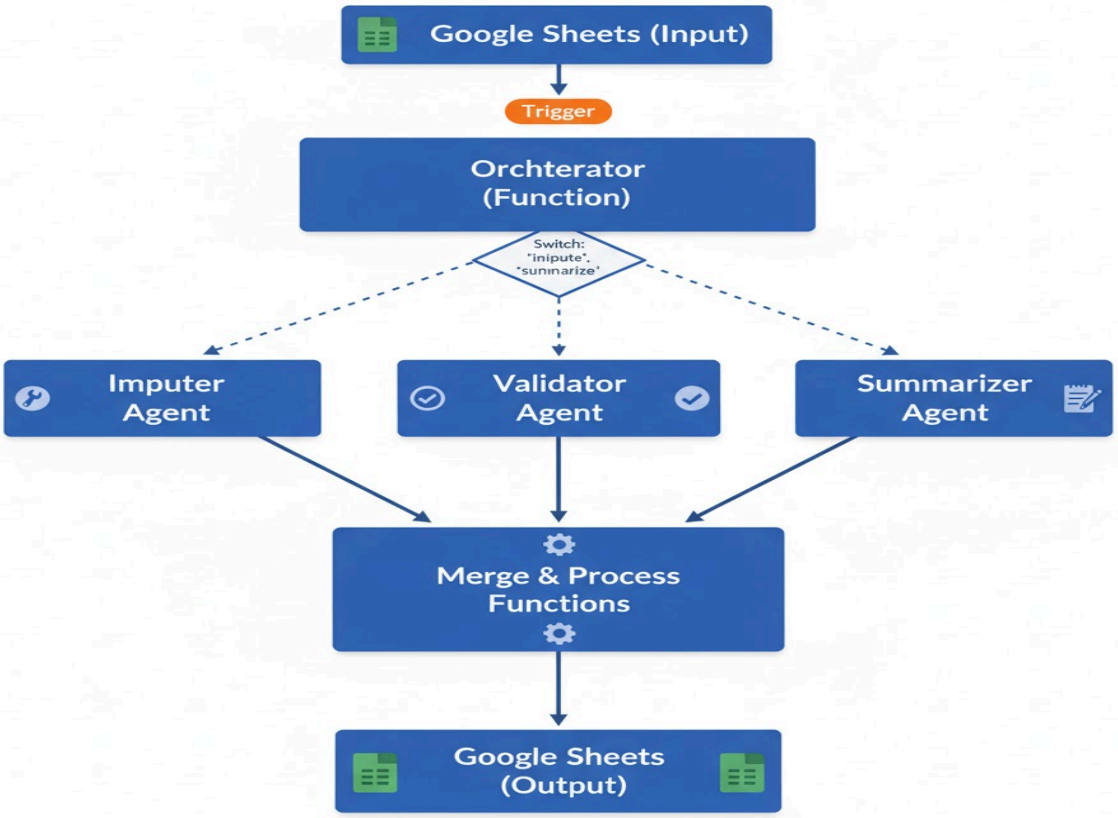
Detailed Workflow and Flow Diagram

1. **Google Sheets Trigger Node:**
    - Polls for new/updated data rows.
2. **Orchestrator Function Node:**
    - Decides action per row: impute missing, validate suspect, or summarize wholesome rows.
3. **Switch Node:**
    - Routes rows to respective agents:
        - Imputer Agent Node: Fills missing values, returns concise suggestions.
        - Validator Agent Node: Checks flagged columns, returns validation flags.
        - Summarizer Agent Node: Produces natural language logs.
4. **Merge Nodes:**
    - Combine outputs from all agents for each input row by position.
5. **Processing Function Nodes:**

- Clean output by filtering unused metadata, maintaining only original row + logs + suggestions.

6. **Google Sheets Output Node:**
    - Updates the target sheet with cleaned and annotated data.

Architecture Diagram (Example)



Low-Level Details

- **Orchestrator Agent:** Uses heuristics or simple rules on row data to label route.
- **Imputer Agent Prompt:**
    - Focus on missing or blank columns only.

    - Provide imputed value and guess method in a concise format.

- **Validator Agent Prompt:**
    - Only return columns listed in cleaning log.
    - Flag columns as valid or invalid with brief reasons.
- **Summarizer Agent Prompt:**
    - Provide a high-level narrative summary of cleaning applied.
- **Merge Logic:**
    - Position-based row merging to combine original data + multi-agent outputs.
- **Logging:**
    - Every step updates the `_cleaningLog` or new suggestion fields.

Guardrails and Memory

- Agents have strict prompt guardrails limiting scope and verbosity.
- Row-level memory maintained in sheet columns and optionally in vector DB for advanced recall.
- Persistent memory enables incremental cleaning over multiple runs.

Extensibility & Maintenance

- Add new agents easily by extending oral orchestrator routing and prompts.
- Integrate vector storage (e.g., Pinecone, Weaviate) for semantic retrieval.
- Monitor and version prompt templates and LLM configurations.