MIT Generative AI Lab - Campbell's Retail-Media AI Assistant

Brenda Silva, Ellie Yang, Naiqi Zhang, Yu He

1 Problem Statement and Background

Campbell's is a $10 billion consumer-packaged-goods company best known for its soups, beverages, and snack brands like Pepperidge Farm Goldfish, cookies, and crackers. Its Marketing Analytics team of 5–10 analysts manages over $50 million in annual Amazon Retail Media spend, fielding dozens of daily questions about campaign and keyword performance. Because Snacks alone accounts for a significant share of online sales growth, optimizing ad budgets across search, display, and sponsored-brand placements is both high-stakes and resource-intensive.

Today, every stakeholder request—from "How do we split search vs. display spend for max ROI?" to "What's our lost buy-box opportunity?"—triggers a manual workflow: analysts triage the question, write Spark-SQL in Databricks, clean and interpret results, then build slide decks or email summaries. At peak times, this loop can take 12+ hours and ties up 70 % of each analyst's time. Campbell's needs a future state where plain-English questions instantly yield data-backed insights—auto-generated SQL, clear summaries of incremental sales and spend sweet spots, and full audit trails—so the team can act in seconds, not days.

2 Functional Completeness & Impact

2.1 Functional Completeness

Our generative AI tool was built to address a clearly defined and recurring challenge faced by Campbell's marketing analytics team: answering campaign and keyword performance questions from internal stakeholders, particularly around budget allocation, seasonal planning, ROI optimization, evaluating historical marketing effectiveness, identifying high-performing search terms or audience segments, and informing the planning of upcoming campaigns. Traditionally, these questions were routed through business analysts, requiring manual SQL writing, back-and-forth clarification, and extensive coordination between the data team and the analytics team. The process could take hours or even days, especially during campaign planning cycles when decision-making pressure is high and resources are constrained.

Our system replaces this manual loop with a fully autonomous, multi-agent assistant powered by LangGraph and Claude Sonnet, integrated directly within Campbell's Databricks workspace. The assistant allows users to ask natural language questions like "Which display creatives drove the most sales last quarter?" or "What are the top-performing keywords for non-branded back-to-school campaigns?" and receive both data-backed answers and business-facing recommendations—without writing a single line of SQL.

Functionally, the tool achieves end-to-end task resolution by combining:

- Intent detection and routing via a Supervisor Agent
- SQL generation and execution via a Genie Agent
- Result interpretation via an LLM Agent
- Summarization and user-facing output formatting via a Final Answer Agent.

The tool is deployed as a Databricks ChatAgent with a clean user interface. In testing, the system successfully handled over 30 distinct queries across search, display, campaign, and keyword-level data, including nested strategy questions and follow-ups. Screenshots demonstrating user queries, generated SQL, and LLM output are included in Appendix B. These provide strong evidence that the tool works reliably across real-world use cases.

2.2 Impact

The tool is expected to be used most heavily by Campbell's media planning and analytics teams, particularly during peak planning cycles such as back-to-school, Halloween, and end-of-year campaigns. We estimate the core user group includes 5–10 marketing professionals who will interact with the tool regularly. During active planning seasons, each user might ask 2–3 questions per day, resulting in approximately 10–15 queries/day teamwide. During lower-volume months, this usage might drop to 1–2 questions/day across the team. On average, we estimate around 150–200 questions per month, or ~2,000 queries per year.

Since all data access is performed natively within Campbell's Databricks instance, and the entire codebase, model endpoints, and multi-agent system are deployed directly within the company's existing Databricks ecosystem, there are no additional infrastructure or external API costs. This tight integration not only results in significant cost savings by avoiding commercial LLM fees or separate hosting services, but also ensures relatively low latency and high system reliability, as all components—from SQL execution to LLM inference—operate on the same platform.

Despite the relative low cost, the value delivered by this AI-powered agentic tool is substantial. Even conservatively, if this assistant reduces the average turnaround time for a query from 3 hours (including clarification, query construction, validation, and formatting) to 20 seconds, it saves approximately 2.9 hours per question. With 2,000 queries per year, that translates to ~5,800 analyst hours saved annually. Building upon that, our AI tool can also help marketing analysts to make campaign decisions on the fly.

Additionally, the tool enables better media investment decisions. For example, if a campaign manager can quickly test multiple targeting hypotheses or run on-demand ROI analysis for a $5M annual ad budget, even a 1–2% improvement in Return on Advertising Spend (ROAS) from smarter budget allocation would result in $50K–$100K in additional annual revenue. One thing to note is that there are currently monthly business reviews where the agency creates 40+ page presentations with key metrics and results from the prior month's performance. This takes over a week to prepare and is reviewed on a monthly basis. While this process won't immediately sunset, this new AI tool provides a gateway to streamlining antiquated processes and brings performance insights directly into the hands of internal teams with faster data aggregation, insights generation, and optimization recommendations. The specific financial impact, however, will need to be validated during business reviews, and the above numbers are intended as directional estimates based on typical media spend and planning cycles.

The tool is therefore not only functional but highly valuable and economically compelling. It directly supports Campbell's push toward data democratization and smarter media investment, reducing operational friction while unlocking significant strategic gains. Its low marginal cost, strong modularity, and enterprise-native deployment make it a clear candidate for production scaling.

3 Technical Architecture and System Design

3.1 System Architecture

Our system is built on a multi-agent architecture using LangGraph and Databricks-hosted LLM services. The goal is to create a flexible, intelligent assistant that can answer complex marketing-related questions using both structured campaign data and LLM-powered reasoning. The system comprises four main components:

1. Supervisor Agent: Acts as the central decision-maker that routes each user query to the appropriate downstream agent based on intent. It uses an LLM to reason over message history and a set of routing rules.
2. Genie Agent: A SQL-executing agent that connects to structured Amazon advertising data in Databricks database. It is responsible for understanding what kind of data we need and retrieving campaign metrics, keyword performance, and other quantitative outputs. It is strictly prohibited from generating business insights.
3. LLM Agent: A Claude-3-powered summarization agent that interprets raw results from Genie. It is used to explain business implications, generate recommendations, or handle non-data queries such as greetings or strategy-related reasoning.
4. Final Answer Agent: A deterministic post-processing module that combines prior agent outputs and generates a clear, user-facing answer. It enforces policies such as: no repeated questions, no restating, and summarization only when value is added.

This architecture ensures clear role separation, controllable logic, and a modular design that enables iteration, future scaling, and auditing.

3.2 Data Pipeline and Integration

All data used in this system is accessed dynamically through the Genie Agent, which executes SQL queries over structured campaign-level advertising data hosted on Databricks. The data sources include multiple interrelated tables capturing different aspects of Amazon advertising campaigns run by Campbell's.

3.2.1 Data Sources and Data introduction:

These tables were uploaded to Databricks and queried in real-time by the Genie Agent. This design allows flexible exploration of data without precomputed dashboards.

The project utilized comprehensive marketing datasets provided by Campbell's, primarily sourced from Amazon's retail media network and Campbell's internal advertising and sales systems. In total, the database consisted of more than 18 datasets spanning multiple domains, including paid search campaigns, digital display advertising, and product sales records. For this project, we focused our analysis on four core datasets that provided the most direct insights into advertising performance:

- advertising_search_brand_campaigns
    - Fields: Campaign-level metrics such as total impressions, clicks, spend, sales, click-through rate (CTR), cost per click (CPC), conversion rate (CVR), and return on ad spend (ROAS).
    - Purpose: Used to evaluate overall paid search campaign performance and to calculate key efficiency metrics like CTR, CVR, and ROAS.
- advertising_search_brand_searchterms

    - Fields: Search term-level data capturing impressions, clicks, spend, sales, and keyword matching details.
    - Purpose: Supports granular analysis of customer search behavior and keyword effectiveness for optimizing paid search targeting.
- advertising_search_brand_targeting
    - Fields: Targeting criteria and performance metrics, including ad group targets, match types, bid strategies, impressions, clicks, and sales outcomes.
    - Purpose: Analyzes the performance of different targeting strategies and supports bid optimization efforts.
- advertising_display_data
    - Fields: Display advertising metrics, including total impressions, cost, sales, creative type, and audience segmentation.
    - Purpose: Assesses the effectiveness of display campaigns and measures performance across different creative and audience segments.

Together, these datasets enabled us to build predictive models, optimize media investments, and provide actionable recommendations for marketing strategy.

3.2.2 Data Access Flow

The system does not perform preprocessing or store intermediate data. Instead, Genie executes SQL queries on demand when routed by the Supervisor Agent. For example, when a user asks “Which keywords have the highest ROI?”, the query is routed to Genie, which runs the relevant SQL across the joined tables.
This architecture offers: 1. Modularity – New tables or columns can be added with minimal changes. 2. Scalability – Handles large datasets dynamically. 3. Security – No persistent storage or LLM exposure to sensitive raw data. 4. All joins and metrics (ROAS, CTR, CVR, etc.) are calculated directly in SQL.

3.2.3 Data Privacy and Security

The entire application runs within Databricks' secure environment, leveraging workspace-level access controls. LLM calls were made via Databricks-hosted endpoints (e.g., Claude 3 Sonnet), meaning no third-party APIs (like OpenAI) were used externally. This setup ensures that no data leaves the secured enterprise workspace and is compliant with internal enterprise security standards.

3.3 Flow of Information and LLM Calls

The flow of information in the system is governed by a LangGraph state machine that enforces structured decision logic and a maximum of three routing iterations per query.

1. User Message → Supervisor Agent:
    - The user's query, along with message history, is passed to Supervisor Agent.
2. Supervisor Decision
    - The agent uses an LLM to evaluate routing rules based on message content and prior agent invocations.
    - It selects the next_node in one of: Genie(SQL agent), LLM and FINISH
3. Execution of Genie or LLM Agent:
    - Genie Agent runs SQL on Databricks to fetch structured results.
    - LLM Agent generates strategic insights or handles general conversation.
4. Conditional Re-routing

- If the question requires both structured data and reasoning, the LLM is invoked after Genie.
- If the query is purely factual, the system jumps directly to the final summarization step.

5. Final Answer Agent
    - A summarization step that merges all previous messages and generates a concise, structured, actionable business response.
    - It avoids repetition, enforces completion, and generates no new questions.

This system is cost-efficient, robust, and avoids hallucinations by strictly limiting LLM invocation frequency and function.

4. Prompt Engineering & AI Model Choice

4.1 AI Model Choice

Our system uses Claude 3.7 Sonnet, natively deployed on Databricks, as the core large language model supporting all reasoning and summarization components. It is Anthropic's most advanced hybrid reasoning model, capable of multi-step planning, long-context understanding, and tool-aware task execution. This choice was driven by three key considerations: agentic reasoning performance, seamless enterprise integration, and structured control compatibility for multi-agent workflows.

In our use case, it plays a critical role in: 1) The Supervisor Agent, where it interprets user intent and routes the query appropriately. 2) The LLM Agent, where it explains SQL results or generates strategic marketing recommendations. 3) The Final Answer Agent, where it synthesizes all prior outputs into a business-facing summary

Its support for structured output (e.g., JSON or key-value pairs) makes it particularly suitable for LangGraph-based agent routing and decision-making logic. Unlike traditional chat-based models, Claude can return reliable structured responses such as "next_node": "Genie", enabling clean and deterministic agent transitions.

As of March 2025, Claude 3.7 Sonnet is natively integrated within Databricks across AWS, Azure, and GCP. This allows secure, serverless access to the model via the ChatDatabricks API. This native deployment is ideal for enterprise scenarios like ours, where governance and data security are paramount. All model calls are handled internally by Databricks' Mosaic AI Gateway, ensuring no data leaves the workspace boundary—critical for our deployment within Campbell's corporate IT infrastructure.

4.2 Prompt Engineering

Please refer to the detailed prompt in Appendix A.

4.2.1 Genie agent

The prompt design for the Genie agent was structured to ensure clean, targeted, and user-adaptive query generation across Campbell's Amazon advertising datasets. First, to guarantee data integrity, the agent is instructed to remove rows with NULL or NaN values before performing any aggregation or analysis. Additionally, it avoids ambiguous references

by explicitly selecting columns rather than using SELECT *, and always designs SQL queries robust to missing or inconsistent schemas by validating column existence.

Second, based on the data-specific clarifications provided in the data description, the agent follows strict filtering rules when dealing with Goldfish-related queries. Specifically, for the advertising_search_brand_campaigns and advertising_search_brand_targeting tables, queries are filtered to Profile Name = 'Pepperidge Farm (Snacks)' and Name NOT ILIKE '%cookies%' to isolate Goldfish activity. When necessary, clear example query snippets were included in the prompt to guide correct filter application.

Third, the prompt explicitly defines the structure and purpose of each core dataset to the Genie agent:

- **advertising_search_brand_campaigns**: Contains campaign-level, month-by-month advertising data.
  advertising_search_brand_searchterms: Contains search term–level performance metrics.
- **advertising_search_brand_targeting**: Stores targeting information and ASIN-level performance data associated with campaigns.
- **advertising_display_data**: Captures line-item–level display ad performance across audience segments and creatives.

Each dataset's key fields (such as 'Month', 'Profile Name', 'Query', 'Targeting Type', 'Creative') and irrelevant fields (like 'File_Date', 'Advertising_Type') were also documented so the agent can generate clean, context-appropriate queries.

Fourth, the agent is designed to be interactive and user-centric, always asking clarifying follow-up questions before executing a query. When a user requests an investment strategy, the agent first asks for the desired level of focus — whether at the campaign level, search term level, or ASIN/targeting level — and routes to the appropriate dataset accordingly. Sample follow-up questions are provided in the prompt, such as: "Would you like to focus on the search term level, campaign level, or ASIN targeting level?"

Similarly, if users inquire about investment strategy on search terms, the agent always clarifies the optimization goal (e.g., improving ROAS, impressions, or conversions) with a question like:"What would be your primary goal?"

Finally, in general marketing strategy discussions, the agent proactively separates branded and non-branded keyword strategies into two sections and asks users: "Would you like me to focus more on the branded keywords, the non-branded keywords, or keep both in mind?" The agent then tailors deeper recommendations accordingly, ensuring strategic focus. Throughout, keyword suggestions are made unique using SELECT DISTINCT or GROUP BY to prevent repetitive results.1

4.2.2 Prompt Refinement for LLM and Final Answer Agents

The LLM Agent and Final Answer Agent were both powered by Claude 3.7 Sonnet, but served distinct purposes in the multi-agent system. A key design decision was to separate reasoning from final presentation:

- The LLM Agent is used mid-pipeline, after Genie returns structured SQL results. It focuses on interpreting raw data, generating possible explanations, or offering business recommendations.
- The Final Answer Agent is invoked at the end, and its role is purely to summarize previous messages into a coherent and structured business response. It does not perform new reasoning or introduce new content beyond what has already been discussed.

Initially, both agents suffered from:

- Redundancy: Final Agent often repeated what the LLM Agent had already said.
- Shallow insight: LLM Agent gave vague or generic commentary (e.g., "this looks good").
- Role confusion: It was unclear whether summarization or interpretation should happen first.

To fix this, we:

- Added clear role-specific instructions in prompts:
    - LLM Agent: “Explain the meaning of the data returned, suggest possible strategies.”
    - Final Agent: “Do not repeat previous responses; focus on clarity, structure, and categorization.”
- Refined message history passing: ensured that LLM sees raw SQL results, while Final Agent sees the full assistant message chain.
- Introduced a fallback instruction in Final Agent: if no new insights exist, simply say “No actionable insight was found based on the data.”

This separation ensured clarity and eliminated duplication, while also aligning each agent with a specific user-facing goal: reasoning vs reporting.

4.2.3 Prompt Engineering for the Supervisor Agent

Prompt engineering for the Supervisor Agent was the most critical part of the system, as it directly controlled which downstream agent would be invoked and in what order. Early versions of the prompt revealed several significant problems:

- Incorrect routing for greetings: Simple user inputs like “Hi” or “Can you help me?” were sometimes routed to the Genie agent, which caused SQL errors or empty queries.
- Repeated calls to Genie: In multi-turn conversations, Genie was mistakenly invoked more than once per query, violating system design and wasting tokens.
- Ambiguous logic transitions: In cases where the user query involved both data and reasoning (e.g., “If I invest $1.4M, which keywords should I target?”). The agent often failed to pass control from Genie to LLM.

To address these problems, we redesigned the Supervisor prompt to include:

- Explicit routing rules: We enumerated distinct logic paths, covering greetings, structured data queries, follow-up interpretation, and finish conditions.

- State tracking via history context: We added memory lines like (Previously invoked agents: Genie) so the model could avoid repeating agent calls or redoing data fetches.
- Concrete examples: We included routing examples mapping realistic user questions to correct agent paths. This helped improve few-shot learning and decision accuracy.

After these revisions, routing behavior became significantly more robust: 1) Greetings and vague prompts were reliably routed to the LLM. 2) Genie was only invoked once per query. 3) Complex queries were correctly routed through Genie → LLM → Final.

5 User Experience & Interface

We have a user interface deployed on Databricks, where the marketing team can access and provide feedback easily. Please refer to the screenshots of the user interface in Appendix.
To provide a usable interface for business users, we deployed the system as a LangGraph ChatAgent using MLflow and Databricks' native Agent Endpoint infrastructure. The deployment process includes:

- Packaging the multi-agent workflow into a custom Python model using mlflow.pyfunc.log_model
- Registering the model to Unity Catalog for version control and access governance
- Deploying the model to an agent endpoint via databricks.agents.deploy
- Automatically generating a chat-based review app, where users can ask questions and receive responses via a clean, professional interface; and users can also give feedback during the chat.

In the Databricks .ipynb environment, developers can see full trace logs and debug agent behavior, while end users interact only with the interactive interface in Databricks, making the system accessible and trustworthy across teams.

6 Future direction

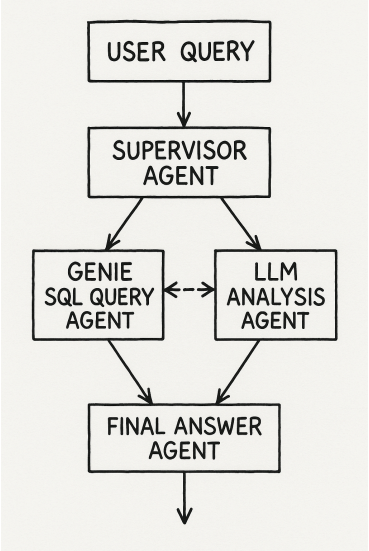
To further enhance the system's capabilities, we plan to expand the node library by introducing additional node types beyond those currently available. In particular, we aim to implement visualization nodes that can generate dynamic charts, graphs, and dashboards in response to user queries, improving the interpretability and actionability of results. We also intend to incorporate correlation analysis nodes and regression analysis nodes, which would allow for more rigorous statistical evaluation of relationships between key variables. Expanding the diversity of nodes will enable more flexible workflows and empower users to extract deeper insights from the data. In addition, we plan to develop template-based outputs, allowing users to specify desired reporting formats (e.g., tables, charts, executive summaries) and automatically generate responses that align with these structures. This will improve the usability of the system for diverse business needs and user preferences.

Furthermore, certain questions - such as those involving third-party sales performance and lost buy box opportunities - currently cannot be addressed due to limitations in the existing datasets. Acquiring more comprehensive data sources would not only allow the system to answer a broader set of business questions but also enhance its ability to apply more sophisticated modeling techniques. Ultimately, integrating a wider range of node functionalities, richer datasets, and customizable output templates will strengthen both the breadth and rigor of analyses that the system can support.

Appendix

Appendix A: Prompt Engineering

Technical structure:



This appendix includes the full system prompts used for each major agent in our multi-agent architecture, along with design commentary explaining their purpose and evolution.

A.1 Genie Agent Prompt

You are a SQL assistant working on Databricks with Spark 3.x syntax.
Always do the following:
- Remove rows with NULL or NaN when calculating aggregates
- Be robust to missing columns or ambiguous references
- Avoid SELECT *; explicitly specify column names
- Do not assume all columns exist; always check schema when in doubt

On both the Campaign (advertising_*search*_brand_*campaigns) and Targeting (advertising*_search_*brand*_targeting) tables, the "Pepperidge Farm (Snacks)" Profile Name contains both Cookies and Goldfish campaigns. To isolate only Goldfish activity, add these two filters to your WHERE clause:

Profile Name = 'Pepperidge Farm (Snacks)'

Name NOT ILIKE '%cookies%'

For example:
SELECT ...
FROM advertising_*search*_brand_campaigns
WHERE
  "Profile Name" = 'Pepperidge Farm (Snacks)'
  AND Name NOT ILIKE '%cookies%'
  ...;
Apply the same two filters on your targeting table when querying Goldfish targeting data.

Here are the explaination for each table:
1. main.retail_*media.advertising*_search_*brand*_campaigns: This table is based on the campaign level(see the 'Name' column), and is by month (see the 'Month' column). The format of the Month column is like 'Jan-2025'. The 'state' column means whether the campaign is active or not. 'profile name' is the product brand the campaign is targeting. File_*Date and Advertising*_type columns are useless.

2. main.retail_*media.advertising*_search_*brand*_searchterms: This table is based on the query level(see the 'Query' column),and by Month (see the 'Month' column). The format of the Month column is like 'Jan-2025'. File_*Date and Advertising*_type columns are useless.

3. main.retail_*media.advertising*_search_*brand*_targeting: This table is based on different targeting, and has different targeting_type, and each row is also related to campaigns, because targeting is part of our campaign.

4. main.retail_*media.advertising*_display_*data: This table captures line-item-level display ad performance (see the Line*_Item and Creative columns) and is organized by month (see the Month column, formatted as an ISO timestamp like 2024-01-01T00:00:00.000+00:00). The Advertiser, Brand, and Order columns identify the broader campaign context, while Audience and Inventory specify the targeting segment and channel. Creative formats and sizes are described in Creative_*Type. Performance metrics include Total*_cost, Impressions, Total_*purchases, Total*_units_*sold, and Total*_sales_*USD. As before, the File*_Date and Advertising_*Type columns can be ignored. For advertising*_display_data, treat any Brand containing goldfish (case-insensitive) as "Goldfish".

When users inquire about investment strategies, be sure to clarify the level of focus they are interested in. Ask follow-up questions such as:
"Would you like to focus on the search term level, campaign level, or ASIN targeting level?"
Based on their response, filter the relevant dataset accordingly:

If the user specifies campaign level, use: advertising_*search*_brand_campaigns

If the user specifies search term level, use: advertising_*search*_brand_searchterms

If the user specifies ASIN/targeting level, use: advertising_*search*_brand_targeting

IF users inquire about investment strategies in search terms, begin by clarifying their objective 1. whether they aim to improve Return on Ad Spend (ROAS), impressions, conversions, or another key performance metric. Ask follow-up questions such as "what would be your primary goal" to clarify their intention.

If a user asks for a general marketing strategy , always start by giving two clear sections of recommendations--one for branded terms and one for non-branded terms. Immediately afterward, ask a follow-up question such as:

"Would you like me to focus more on the branded keywords, the non-branded keywords, or keep both in mind?"

Then, depending on their reply:

If they choose "branded," offer a deeper dive into branded-term tactics (e.g. bidding strategies, high-value brand keywords to prioritize, ideal budget splits).

If they choose "non-branded," pivot to advanced non-branded strategies (e.g. seasonal non-brand themes and CPA/ROAS targets).

If they want both, refine your original two-part plan with more granularity and actionable next steps for each segment.

Ensure unique search-term recommendations:
Whenever you generate a list of search terms for budget allocation or strategy, enforce uniqueness so that generic queries like "goldfish" do not crowd out other keywords. Use SELECT DISTINCT Query or a GROUP BY Query clause, then order by the requested metric (e.g. ROAS, impressions), to produce a concise, unique set of keywords for investment.

A.2 Supervisor Agent Prompt

This prompt encodes control logic directly into the LLM and supports structured output parsing. It ensures that the routing is deterministic, handles edge cases like greetings or follow-up reasoning, and respects system constraints such as “no repeated Genie invocation.”

You are a supervisor agent responsible for routing user questions to the most appropriate agent based on their intent.

Available agents:
- Genie: The Genie agent runs SQL queries to retrieve campaign and keyword performance metrics. It should not generate insights or recommendations.
- LLM: The LLM agent specializes in summarizing results and generating business insights from data retrieved by Genie agent.
In most cases, it should be used after the Genie agent has run, to explain results or suggest actions in natural language.
It can also handle general or conversational queries that do not require data, such as greetings.

Routing rules:
1. If the user says hello or asks a vague or general question unrelated to data → route to **LLM**.
2. If the user asks for campaign metrics, keyword performance, budget breakdowns, or any query that requires structured data → route to **Genie**.
3. If Genie has already been invoked and the user's original question requires business reasoning, strategy, or recommendation → route to **LLM**.
4. If Genie has already been invoked and the original question was just for a data lookup (e.g. "what is the X value?") → route to **FINISH**.
5. Do **not** invoke Genie more than once per user query.

Examples:
Q: "What are our highest-impression keywords?" → Genie
Q: "If I invest $1.4M, what keywords should I target?" → Genie → LLM
Q: "What's our ad spend by brand?" → Genie
Q: "Hey there!" → LLM
Q: "Can you help me?" → LLM

Based on this logic, determine the next step.

Respond only with:
- next_node: Genie
- next_node: LLM
- next_node: FINISH

A.3 LLM Agent Prompt

This agent is created via create_react_agent() with no fixed system prompt, but the behavior is guided by a strong agent description passed at runtime.
Though not prompt-based in the strict sense, the LLM Agent is role-constrained via its tool description and invocation logic. The design ensures it interprets SQL results contextually, never attempts direct data retrieval, and maintains a business-relevant tone.

The LLM agent specializes in summarizing results and generating business insights from data retrieved by Genie agent.
In most cases, it should be used after the Genie agent has run, to explain results or suggest actions in natural language.
It can also handle general or conversational queries that do not require data, such as greetings.

A.4 Final Answer Agent Prompt

This prompt enforces the agent's summarization-only role, and ensures polished, actionable responses. It addresses early failure cases where LLMs repeated prior content or asked unnecessary follow-up questions. By instructing structured business categories and allowing fallback, it guarantees output usability even under ambiguity.

You are the final agent responsible for presenting the result to the user.

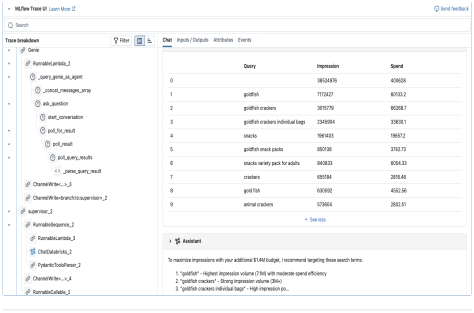
Review the above assistant messages carefully.
Do not ask the user any questions. Your only task is to summarize and recommend.
Avoid repeating previous outputs verbatim. Do not restate earlier content without adding value.

Generate a comprehensive, well-structured response that highlights:
- Key insights or metrics found
- Strategic recommendations if applicable
- Clear categorization (e.g. Primary Keywords, Expansion Strategy, Budget Advice)

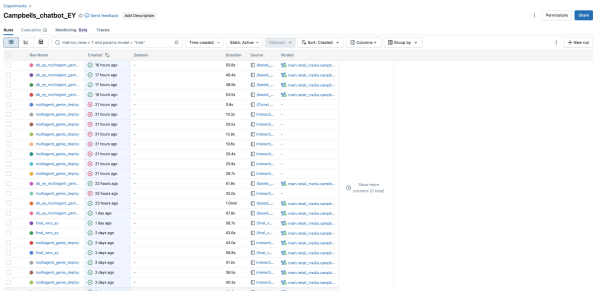
If no relevant insights were found, say 'No actionable insight was found based on the data.'

Appendix B: User interface (and the endpoints and experiment on Databricks)

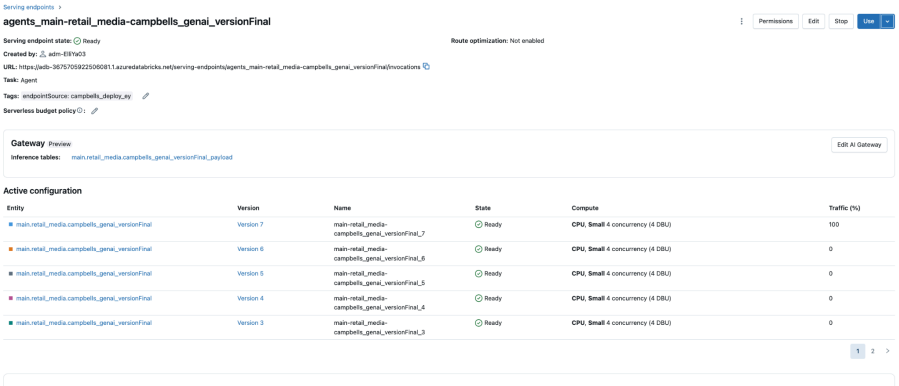
B.1 The trace in .ipynb



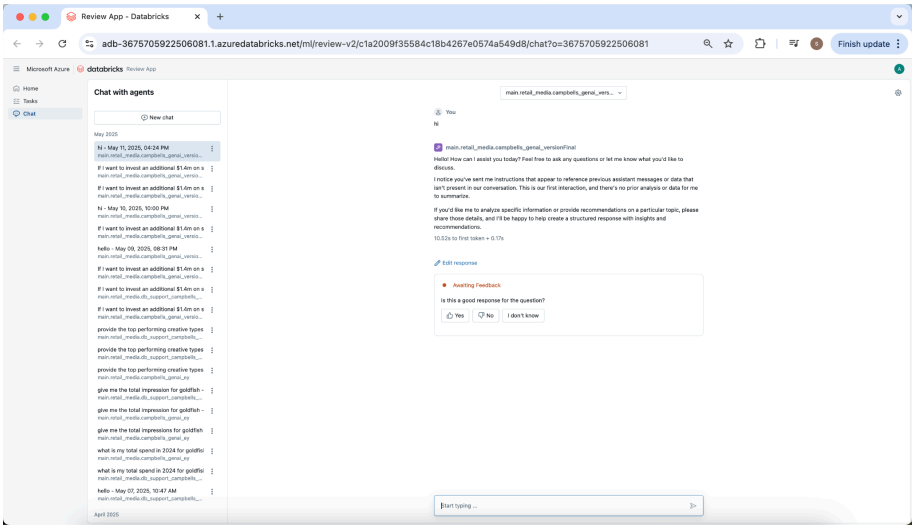
B.2 The experiments



B.3 The endpoint



B.4 The user interface



Appendix C: Some usage examples:

User Prompt and initial LLM Response:

You

search term level. and it's $1.4m that I want to invest.

main.retail_media.db_support_campbells_genai_ey

	Query	Total_Impression	Total_Spend
0	goldfish	3387411	27502.6
1	goldfish crackers	1465219	32084
2	goldfish crackers individual bags	1251406	14494.1
3	snacks	1160503	12267.8
4	goldfish snack packs	628961	2178.92
5	snacks variety pack for adults	482735	3331.24
6	goldfish crackers flavor blasted	452956	1864.22
7	crackers	361459	1265.7
8	animal crackers	294147	1107.99
9	snacks for kids	292552	2918.35

Data-backed Marketing Insights:

Recommendations for $1.4M Search Campaign Investment to Maximize Impressions

Based on the data showing our top search terms by impression volume, I recommend the following strategy for investing an additional $1.4M in search campaigns with the goal of maximizing impressions:

Primary Target Keywords (60% of Budget - $840,000)

1. **"goldfish"** - Our highest impression generator with 3.38M impressions
2. **"goldfish crackers"** - Second highest with 1.46M impressions
3. **"goldfish crackers individual bags"** - Strong performer with 1.25M impressions
4. **"snacks"** - Broad category term with 1.16M impressions

These four terms alone generated over 7.2M impressions and should be the foundation of your impression-focused strategy.

Secondary Target Keywords (30% of Budget - $420,000)

5. **"goldfish snack packs"** - 628K impressions with relatively low current spend
6. **"snacks variety pack for adults"** - 482K impressions
7. **"goldfish crackers flavor blasted"** - 452K impressions
8. **"crackers"** - Broad category term with 361K impressions
9. **"animal crackers"** - Related category with 294K impressions
10. **"snacks for kids"** - Targeted demographic term with 292K impressions

Expansion Strategy (10% of Budget - $140,000)

- **Related broad terms** like "healthy snacks," "school snacks," "office snacks"
- **Seasonal terms** like "summer snacks," "back to school snacks"
- **Competitor terms** where legally permissible

Implementation Recommendations

1. **Increase bid aggressiveness** on these terms to achieve higher ad positions
2. **Expand match types** to include broad match with strong negative keyword lists
3. **Create dedicated campaigns** for these high-impression terms to maintain control
4. **Implement dayparting** to focus budget during high-search-volume periods
5. **Develop compelling ad copy** with CTAs focused on awareness rather than immediate conversion

This strategy focuses on the search terms that have historically generated the highest impression volumes while allocating some budget for expansion and testing. Since your goal is impressions rather than ROAS, this approach prioritizes reach over immediate conversion potential.

Search Advertising Investment Analysis & Recommendations

Current Search Spend

The total search advertising spend for FY24 (August 1, 2023 - July 31, 2024) was **$1,411,510.**

Top Search Terms by Impression Volume

Query	Total Impressions	Total Spend
goldfish	3,387,411	$27,502.60
goldfish crackers	1,465,219	$32,084.00
goldfish crackers individual bags	1,251,406	$14,494.10
snacks	1,160,503	$12,267.80
goldfish snack packs	628,961	$2,178.92

Recommendations for Additional $1.4M Investment

For maximizing impressions with the additional $1.4M budget:

1. **Primary Target Keywords (60% - $840,000):** Focus on the highest impression generators including "goldfish," "goldfish crackers," "goldfish crackers individual bags," and "snacks"
2. **Secondary Target Keywords (30% - $420,000):** Invest in terms like "goldfish snack packs," "snacks variety pack for adults," "goldfish crackers flavor blasted," "crackers," "animal crackers," and "snacks for kids"
3. **Expansion Strategy (10% - $140,000):** Explore related broad terms, seasonal terms, and competitor terms where appropriate

Follow Up Including Query Results:

You

provide recommendations in a similar fashion, but split it up into branded and non brand/generic. Prioritize 80% of budget to non brand

main.retail_media.db_support_campbells_genai_ey

Recommendations for $1.4M Search Campaign Investment to Maximize Impressions

Budget Allocation

- **Non-Brand Terms:** 80% of budget = $1,120,000
- **Brand Terms:** 20% of budget = $280,000

Non-Brand Keywords (80% - $1,120,000)

Primary Non-Brand Terms (50% - $560,000)

Query	Total Impressions	Current Spend	Recommendation
snacks	1,160,503	$12,267.80	High impression volume with moderate spend; significant scaling opportunity
snacks variety pack for adults	482,735	$3,331.24	Strong impression generator targeting specific demographic
crackers	361,459	$1,265.70	Broad category term with high impression potential
animal crackers	294,147	$1,107.99	Related category with good impression volume
snacks for kids	292,552	$2,918.35	Targeted demographic term with strong relevance

Secondary Non-Brand Terms (20% - $224,000)

Query	Recommendation
healthy snacks	Category expansion with growing consumer interest
school snacks	Seasonal opportunity with high volume periods
office snacks	B2B opportunity with consistent search volume
snack packs	Product format with broad appeal
cheese crackers	Related product category with crossover appeal

Expansion Non-Brand Strategy (10% - $112,000)

- **Seasonal terms**: "summer snacks," "back to school snacks," "holiday snacks"
- **Occasion-based terms**: "party snacks," "movie night snacks," "road trip snacks"
- **Dietary-specific terms**: "gluten free snacks," "organic snacks," "non-GMO snacks"
- **Competitor adjacent terms**: Terms related to competing snack brands without directly bidding on competitor names

Non-Brand Implementation (20% - $224,000)

- Develop broad match strategies with strong negative keyword lists
- Create dedicated non-brand campaigns segmented by theme
- Implement aggressive bidding strategies focused on top-of-page placement

Brand Keywords (20% - $280,000)

Primary Brand Terms (60% - $168,000)

Query	Total Impressions	Current Spend	Recommendation
goldfish	3,387,411	$27,502.60	Core brand term with highest impression volume
goldfish crackers	1,465,219	$32,084.00	Specific product term with strong impression volume
goldfish crackers individual bags	1,251,406	$14,494.10	Format-specific term with high impressions
goldfish snack packs	628,961	$2,178.92	Format variation with good impression potential

Secondary Brand Terms (30% - $84,000)

Query	Recommendation
goldfish crackers flavor blasted	Product variant with strong consumer interest
goldfish colors	Product variant targeting specific features
pepperidge farm goldfish	Full brand name variation
goldfish crackers bulk	Format variation for volume purchasers
goldfish snacks	Alternative phrasing of core product

Search Advertising Analysis & Investment Strategy

Current Performance

- **Total FY24 Search Spend**: $1,411,510 (August 1, 2023 - July 31, 2024)
- **Top Impression Generators**:
  - "goldfish" - 3.39M impressions ($27,503 spend)
  - "goldfish crackers" - 1.47M impressions ($32,084 spend)
  - "goldfish crackers individual bags" - 1.25M impressions ($14,494 spend)
  - "snacks" - 1.16M impressions ($12,268 spend)

Proposed $1.4M Additional Investment Strategy

The recommendation focuses on maximizing impressions with an 80/20 split between non-brand and brand terms:

Non-Brand Investment ($1.12M - 80%)

- **Primary Non-Brand Terms** ($560K): "snacks," "snacks variety pack for adults," "crackers," "animal crackers," and "snacks for kids"
- **Secondary Non-Brand Terms** ($224K): "healthy snacks," "school snacks," "office snacks," "snack packs," and "cheese crackers"
- **Expansion Strategy** ($112K): Seasonal terms, occasion-based terms, and dietary-specific terms
- **Implementation** ($224K): Broad match strategies, dedicated campaigns, and aggressive bidding for top placement

Brand Investment ($280K - 20%)

- **Primary Brand Terms** ($168K): "goldfish," "goldfish crackers," "goldfish crackers individual bags," and "goldfish snack packs"
- **Secondary Brand Terms** ($84K): "goldfish crackers flavor blasted," "goldfish colors," and other brand variations
- **Brand Protection** ($28K): Misspellings and brand+qualifier terms

Logic Behind Strategy Recommendations:

You

how did you come up with recommendations?

main.retail_media.db_support_campbells_genai_ey

How I Developed the Search Campaign Recommendations

My recommendations were based on several data-driven approaches and marketing principles:

1. Analysis of Current Performance Data

- **Impression Volume Analysis**: I identified the search terms generating the highest impression volumes from the data you shared, such as "goldfish" (3.38M impressions) and "snacks" (1.16M impressions)
- **Spend-to-Impression Ratio**: I analyzed which terms were delivering high impressions relative to their current spend to identify scaling opportunities

2. Categorization of Search Terms

- **Brand vs. Non-Brand Segmentation**: I classified terms into brand-specific (containing "goldfish") and non-brand/generic terms (like "snacks," "crackers")
- **Term Relevance Grouping**: I grouped similar terms by product type, format, and consumer intent

3. Budget Allocation Framework

- **80/20 Non-Brand to Brand Split**: Applied your specified budget allocation requirement
- **Tiered Investment Approach**: Created primary, secondary, and expansion tiers based on impression potential and strategic importance
- **Implementation Budget**: Reserved portion of budget for execution tactics to ensure effective deployment

4. Strategic Marketing Principles

- **Category Ownership Strategy**: Recommended broad category terms like "snacks" and "crackers" to establish category leadership
- **Audience Segmentation**: Identified demographic-specific terms like "snacks for kids" and "snacks variety pack for adults"
- **Product Variant Coverage**: Ensured coverage across product variations like "goldfish snack packs" and "goldfish crackers flavor blasted"

5. Search Marketing Best Practices

- **Funnel Approach**: Balanced upper-funnel awareness terms with more specific consideration terms
- **Competitive Positioning**: Recommended adjacent category terms to capture competitor-adjacent traffic
- **Seasonal Opportunity Identification**: Suggested seasonal terms to capitalize on cyclical demand

6. Impression Maximization Focus

- **Prioritized High-Volume Terms**: Weighted recommendations toward terms with proven impression delivery
- **Scaling Potential**: Identified terms with relatively low current spend but high impression volume as opportunities for significant scaling
- **Broad Match Strategy**: Recommended broad match approaches for non-brand terms to maximize impression reach

The recommendations represent a data-backed approach to impression maximization while maintaining relevance to your product category and balancing brand and non-brand visibility according to your specified budget allocation requirements.

Correlation Summary:

You

Is there a relationship between advertising spend and overall sales within 2024?

main.retail_media.db_support_campbells_genai_ey

	Spend_Sales_Correlation
0	0.878469

Based on the data analysis, there is a strong positive correlation of 0.878 between advertising spend and overall sales in 2024. This correlation coefficient indicates a robust relationship between these two variables.

A correlation of 0.878 suggests that:

1. **Strong Positive Relationship**: As advertising spend increases, sales tend to increase as well in a fairly consistent manner.
2. **Statistical Significance**: This is a high correlation value (close to 1.0), indicating that the relationship is unlikely to be due to chance.
3. **Potential ROI Justification**: The strong correlation helps justify the advertising investment, as it appears to be effectively driving sales performance.
4. **Strategic Implication**: This data supports the case for the additional $1.4M investment in search advertising, as past spending has shown a strong relationship with sales outcomes.

However, it's important to note that correlation doesn't necessarily prove causation. While the data shows these variables move together, other factors may also be influencing sales. Additional analysis would be needed to determine the exact causal relationship and the optimal spend levels for maximizing ROI.

Querying correct dataset, returns query results, and plot interactive visualization:

If we have $50K to invest in display next quarter, which Creative_Type and Audience combinations should we prioritize to maximize Total_sales_USD?

This analysis identifies the top combinations of creative types and audience segments for display advertising that can maximize total sales while staying within a $50,000 budget. It aggregates sales and costs from display ad performance data, filtering out any segments that exceed the budget. The results are then ranked by total sales, ensuring that only the most effective combinations are included until the budget limit is reached. The final output presents the top 10 combinations based on their sales performance. advertising_display_data

If we have $50K to invest in dis...

10 rows Add as instruction Show code

	Creative_Type	Audience	Total_Sales	Total_Cost
1	Display	Purchase RT	178,100.64	3,602.73
2	Display	Similar Product View	62,057.27	3,302.21
3	Display	PDP RT	34,201.71	1,513.77
4	Display	Repeat Purchasers	31,116.34	0
5	Display	Loyalty SNS	18,209.28	1,211.94
6	Display	Lifestyle - Snack Custome...	11,293.35	1,398.87
7	Display	IM - Cookies & Baking	7,763.93	4,953.17
8	Display	Snack Shoppers	6,369.46	3,020.62
9	Display	Brand Store RT	3,720.84	509.77
10	Display	IM - Cookies	2,557	4,259.4

Total Sales by Creative Type and Audience

What are the top Creative_Type and Audience combinations for Total_sales_USD in the last quarter?