NOV 12, 2024

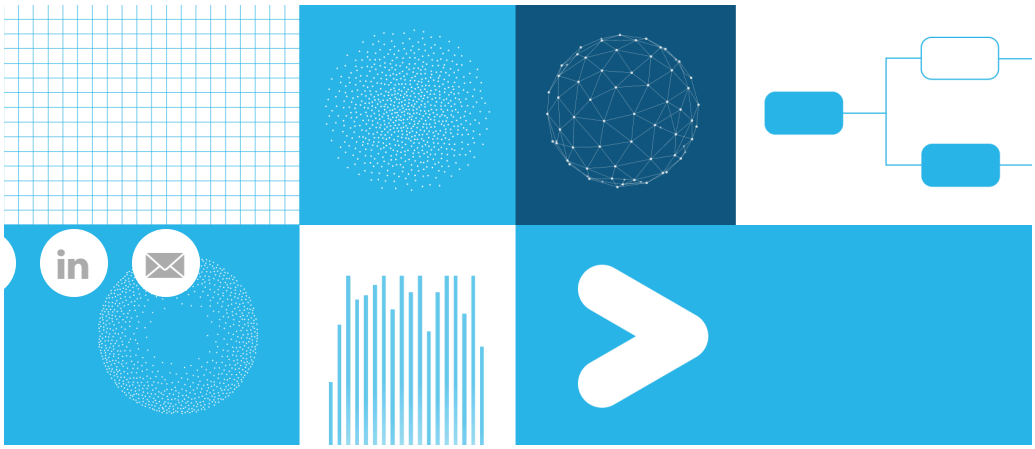
AUTHOR

Aiwen Xu

SHARE

Snowflake Cortex Analyst: Support for Multi-turn Conversation

Gen AI



Snowflake Cortex Analyst has stood out as a trustworthy self-service analytics tool by providing a natural language, conversational interface for business users to receive accurate answers about structured data in Snowflake. **Through the use of a complex workflow where multiple agents interact with each other,** and the recent **support for joins**, Cortex Analyst is now able to achieve upwards of 90% accuracy on real-world text-to-SQL tasks, surpassing other solutions on the market (**based on an internal comprehensive benchmarking suite**).

We were initially laser-focused on achieving high performance on text-to-SQL, so **for our previous release** of Cortex Analyst, we did not offer multi-turn conversational support. But we wanted to provide our users with a chatbot-like experience, giving users the ability to ask follow-up questions to dive deeper into the data. To this end, we recently announced the public preview release of multi-turn conversations in Cortex Analyst.

For example:

> The user asks: “What is the month-over-month revenue growth for 2021 in Asia?”

Then, they can follow up with: “What about North America?” Cortex Analyst will recognize the follow-up, retrieve the context from the conversation history and reframe the second question as: “What is the month-over-month revenue growth for 2021 in North America?” Cortex Analyst then generates the appropriate SQL query to answer this question.

AUTHOR

Aiwen Xu

Adding an LLM summarization agent

A primitive way to add multi-turn conversational support is to simply pass the conversation history to the current LLM calls in Cortex Analyst. However, this approach won’t work well doing it this way. Since our approach is agentic, and each agent has a specific task, adding the conversation history (which can get arbitrarily long) to each LLM call means that each LLM agent needs to understand the question within its conversational context and execute the task simultaneously. This can lead to longer inference times due to more tokens to process. It might

ıls ıs\ ,re nondeterminism, as various LLM agents potentially interpret questions differently. Last but not least , the overall performance could be degraded at each step due to multitasking. Thus, we wanted an approach that preserves the original workflow to the maximum extent possible.

Our approach was to add another LLM agent at the beginning of the Cortex Analyst workflow whose task is to summarize (i.e., to rewrite the current-turn user question into a self-contained question according to the conversation history). This approach does not change the downstream Cortex Analyst workflow, which we know provides high accuracy, as discussed above.

Figure 1. Illustration of the workflow of Cortex Analyst. The blocks in blue represent the original workflow, and the blocks in red are newly added to support multi-turn conversation.

Since summarization is a pure natural language task, we believe in-context learning is sufficient for a trained LLM to perform this task. We provide few-shot examples in the context to cover different scenarios, such as: The current-turn question is related to the previous conversation so the question is rewritten; or the current-turn question is independent from the previous conversation so the conversation history is ignored.

AUTHOR

Aiwen Xu

Evaluating the quality of multi-turn conversation

Next, we evaluated our approach: How well does Cortex Analyst handle multi-turn conversation with an additional agent to rewrite questions? We could do the same evaluation as before — running Cortex Analyst on a set of questions with conversation history and judging the execution output of the generated SQLs. In addition, we also wanted to separately understand the performance of the summarization step itself. To this end, we leverage the LLM-as-a-judge approach to rate the quality of the final rewritten question.

We collected a set of questions with conversation history; generated the rewritten question with the summarization agent; and then asked Mistral Large 2 to decide whether the rewritten question is sufficient: *Based on the conversation history and the current-turn user question, is the rewritten question self-contained, capturing all the necessary information?*

After inspecting all the rewritten questions that were rated poorly by the LLM judge, we observed an approximately 5% error rate in regards to conversation history while using Llama 3.1 8B, which at times returned the second-to-last user question instead of focusing on the current-turn user question. Llama 3.1 70B, on the other hand, did not have this issue, and 96.5% of the rewritten questions were rated as good by the LLM judge. Therefore, we used Llama 3.1 70B as the summarization agent in Cortex Analyst.

Summary

To support multi-turn conversation in Snowflake Cortex Analyst, we have added an additional LLM summarization agent before the original workflow. By using LLM as a judge to evaluate summarization quality, we conclude that this simple approach is actually quite effective. However, one needs to keep in mind the latency-performance tradeoff: Larger models are generally going to have better performance, but we also want the system to be as responsive as possible. Therefore, we should always

select the smallest model that satisfies the need, and in our case, Llama 3.1 70B is sufficient for the summarization task.

Cortex Analyst is becoming more conversational with multi-turn support. A natural next step for Cortex Analyst is to handle more open-ended data-related questions that reveal business insights, so that it can truly become your colleague in business intelligence.

AUTHOR

Aiwen Xu

We invite you to try all the new **Cortex Analyst features**, including multi-turn capabilities, and share your feedback. Here are some additional valuable resources to help you get started:

BUILD Cortex Analyst Announcements: Check out our **roundup blog** for the latest feature updates.

Quickstart Guide: Use our semantic model generator tooling to **create semantic models for Cortex Analyst**.

GitHub Samples Repository: Discover **inspiring examples** on how to put Cortex Analyst to use.

Third-Party Semantic Layers: Learn how to **translate existing third-party semantic layers for use with Cortex Analyst**.

Stay tuned as we continue to enhance support for enterprise-grade, AI-driven self-serve analytics!

RELATED CONTENT

AUG 29, 2024

AUG 08, 2024

AUG 14, 2024

Snowflake Cortex Analyst: Behind the Scenes

Building a conversational self-service analytics product for business users is a complex and challenging endeavor. Trust and accuracy have to be at the core of such a product, as business...

Here's How

Snowflake Cortex Analyst: Evaluating Text-to-SQL Accuracy for Real-World Business Intelligence Scenarios

Being able to build a system that allows business users to ask intricate, complex and...

Delve into the details

Snowflake Cortex Search: High-Quality, Performant Search and Retrieval for Enterprise AI

Search and retrieval systems have always been a critical backbone for knowledge management in enterprises....

Expand your knowledge

Data Cloud
Snowflake for
Marketing
Analytics

Sign up for Snowflake Communications

diana.shaw@snow United States

By submitting this form, I understand Snowflake will process my personal information in accordance with their Privacy Notice. Additionally, I consent to my information being shared with Event Partners in accordance with Snowflake's Event Privacy Notice. I understand I may withdraw my consent or update my preferences here at any time.

SUBSCRIBE NOW

AUTHOR

Aiwen Xu

SHARE