**❄️ Snowflake Summit Booth Challenge: AI-Powered Text Classification with Cortex**

**🧩 Overview**

Classify messy invoice line items into standardized service categories using Snowflake Cortex's CLASSIFY\_TEXTfunction.  
This challenge demonstrates Cortex's ability to apply large language models directly inside Snowflake — no external services needed.

**👥 Version 1: Booth Helper Guide (Full Solution + Outputs)**

**✅ Goal**

Predict a category (e.g. "Snowflake Deployment") for each service\_description using AI.

**📦 Setup Tables (Preloaded in Schema)**

A computer screen shot of green text

AI-generated content may be incorrect.-- models/challenge\_001\_setup/invoice\_line\_items.sql

-- models/challenge\_001\_setup/service\_categories.sql



-- models/challenge\_001\_setup/schema.yml

A computer screen shot of white text

AI-generated content may be incorrect.]  
}

**🤖 AI-Powered Classification (Final Model)**

A computer screen shot of a computer code

AI-generated content may be incorrect.

**🧾 Example Output**

**A screenshot of a computer program

AI-generated content may be incorrect.**

**🧑‍💻 Version 2: Attendee Challenge Sheet (Prompt-Only)**

**🏁 Challenge Goal**

You work with raw invoice descriptions like:

* "Deployment of Snowflake project - Phase 1"
* "Data ingestion pipeline optimization"

Your task is to **classify each description** into a standard category using **Snowflake Cortex**'s CLASSIFY\_TEXT()function.

**🎯 Requirements**

* Use invoice\_line\_items and service\_categories as dbt sources and models
* Dynamically pull the full list of categories into an array using ARRAY\_AGG()
* Run CLASSIFY\_TEXT(input, classes) to classify each line item

**📥 Output**

Your result should include:

* line\_item\_id
* service\_description
* predicted\_category

**💡 Hint**

You’ll need to use a CTE to first aggregate all category names into an array:

Then join that into every row using a cross join with {{ ref('invoice\_line\_items') }}.

**🧪 Bonus**

Try adding confidence scores using CLASSIFY\_TEXT\_PROBABILITIES()!