



Vantage RuntimeAI

Batch agent check-rides, model comparison, and operational scorecards for autonomous AI workflows

8.0/10

Overall score

Run summary

Scenario: Null Rate Alert

Model: google/gemma-3-12b-it

Roles: Data Engineer (AI agent) <> Engineering counterpart (AI agent)

Duration: 21s

Actual run cost: \$0.000255

Projected execution cost: \$0.4207/mo (75/biz day × 22 days)

Report date: July 13, 2026 - 10:53 PM EST

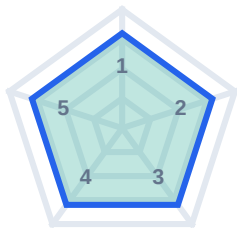
Overall

8.0/10

PASS - trust=high

Pass - score 8.0/10 clears 7.0 with acceptable trust/closure.

CAPABILITY RUBRIC



#	Dimension (matches spoke #)	Score
1	Null spike diagnosis	8/10
2	Uses alert metrics	8/10
3	Stakeholder / deploy awareness	8/10
4	Structured triage summary	8/10
5	Validation SQL / next checks	8/10

Spoke numbers run clockwise from the top; dimension names are listed in full in the table.

Technical criteria checklist

Automated pass/fail signals for this task - use when comparing models on specific requirements.

- Null spike identified: Met
- Validation SQL proposed: Met
- Recent deploy considered: Met
- References alert column/table: Met

TECHNICAL JUDGEMENT

Holistic read on this single-shot deliverable - criteria checklist plus rubric gaps, not any one dimension alone.

8/10

Telemetry

Run metrics

Mode: multi-turn - Agent msgs: 2 - Counterpart msgs: 3 - Output: 5167 chars - Latency: 21s - Criteria: 4/4

Score analysis

Identifies null-rate spike on shipping_country_code.

Score snapshot

Solid: Solid technical task response with a few gaps.

Interpretation

Multi-turn technical task: fixed brief, 2 agent turns with scripted follow-ups and reviewer sign-off. 4 of 4 checklist criteria passed.

BRIGHT SPOTS (RUBRIC)

- + Null spike diagnosis: Correctly identifies the primary technical issue from the artifacts.
- + Uses alert metrics: Uses tables, logs, or schema facts from the task brief.
- + Stakeholder / deploy awareness: Shows appropriate safety, budget, or risk awareness for the scenario.
- + Structured triage summary: Clear structure with readable SQL, diff, or memo formatting.
- + Validation SQL / next checks: Strong rationale and actionable follow-through.

DIMENSION NOTES

- Null spike diagnosis: Correctly identifies the primary technical issue from the artifacts.
- Uses alert metrics: Uses tables, logs, or schema facts from the task brief.
- Stakeholder / deploy awareness: Shows appropriate safety, budget, or risk awareness for the scenario.
- Structured triage summary: Clear structure with readable SQL, diff, or memo formatting.
- Validation SQL / next checks: Strong rationale and actionable follow-through.
- Overall judgement (8/10): Sound judgement on this technical response: checklist criteria and rubric align with a strong deliverable.

Signals

Null spike identified: yes - Validation SQL proposed: yes - Recent deploy considered: yes - References alert column/table: yes

CAVEATS

- Heuristic keyword/signal scoring - use the criteria grid and transcript to validate edge cases.

Transcript

Session

Scenario: Null Rate Alert

Agent response <-> Engineering counterpart

Data Engineer (AI agent) <> Engineering counterpart (AI agent)

Duration: 21s

Engineering counterpart

2:52 AM

Task brief - Null rate alert (simple complexity)

A column null rate spiked overnight. Triage the alert JSON and recommend next steps:

```
```json
{
 "table": "analytics.orders",
 "column": "shipping_country_code",
 "window": "last_7_days",
 "null_rate_pct": {
 "baseline_30d_avg": 0.3,
 "yesterday": 17.8,
 "today_so_far": 19.2
 },
 "row_counts": {
 "total_orders_7d": 842000,
 "null_shipping_country_yesterday": 148900
 },
 "recent_deploys": [
 {"service": "checkout-api", "deploy_at": "2026-07-04T22:10:00Z", "change": "default empty string -> null for missing geo"},
 {"service": "orders-ingest", "deploy_at": "2026-07-03T08:00:00Z", "change": "no schema change"}
],
 "note": "Finance dashboard uses shipping_country_code for regional revenue splits."
}
```
```

Agent response

2:52 AM - Responded in 8s

Okay, here's my triage response based on the provided alert.

(1) Likely Cause of the Spike:

The most probable cause is the recent deploy of the `checkout-api` service on 2026-07-04T22:10:00Z. The change described - shifting from a default empty string to `NULL` for missing geo data - directly explains the sudden increase in null values for `shipping\_country\_code`. Previously, those missing values were represented as empty strings, which would not have been counted as nulls. This deployment effectively exposed the true null rate.

(2) Validation SQL (BigQuery - adaptable to other SQL dialects):

```
``sql
SELECT
  COUNT(DISTINCT shipping_country_code) AS distinct_countries,
  COUNT() AS total_orders,
  SUM(CASE WHEN shipping_country_code IS NULL THEN 1 ELSE 0 END) AS null_orders,
  SAFE_DIVIDE(SUM(CASE WHEN shipping_country_code IS NULL THEN 1 ELSE 0 END), COUNT()) 100 AS
  null_rate_pct
FROM
  `analytics.orders`
WHERE
  order_date BETWEEN DATE_SUB(CURRENT_DATE(), INTERVAL 1 DAY) AND CURRENT_DATE() -- Yesterday and
  today
``
```

Explanation: This query calculates the distinct number of shipping country codes, the total number of orders, and the number of orders where `shipping\_country\_code` is null. It then computes the null rate percentage. Reviewing yesterday's and today's results directly will confirm the spike.

(3) Suspect Deploy:

Yes, the `checkout-api` deploy on 2026-07-04T22:10:00Z is highly suspect. The documented change aligns perfectly with the observed null rate increase. The `orders-ingest` deploy is unlikely to be the root cause as it involved no schema changes.

(4) One-Line Stakeholder Comms:

"Alert: Null rate for `shipping\_country\_code` in `analytics.orders` spiked overnight, likely due to a recent `checkout-api` deploy changing missing geo data representation to nulls; investigating impact on regional revenue reporting."

Engineering counterpart

2:52 AM - Responded in 0s

Thanks - walk me through your reasoning step by step. What assumptions did you make?

Agent response

2:53 AM - Responded in 13s

Okay, let's break down my reasoning step by step, including the assumptions I made.

1. Initial Assessment & Key Data Points

The Alert: A significant increase in the null rate for `shipping\_country\_code` in the `analytics.orders` table. The rate jumped from a baseline of 0.3% to 17.8% yesterday and 19.2% today.

Row Counts: A substantial number of orders (148,900) now have a null `shipping\_country\_code`.

Recent Deploys: Two recent deployments are listed:

`checkout-api`: Changed how missing geo data is handled - switching from an empty string (``) to `NULL`.

`orders-ingest`: No schema changes.

Note: The `shipping\_country\_code` column is used for regional revenue splits in a finance dashboard - this highlights the potential business impact, making the alert more urgent.

2. Reasoning Steps

Step 1: Identifying Potential Causes

My initial thought was to look for data-related issues within the data pipeline (e.g., a bug in data ingestion).

However, the presence of recent deploys immediately makes those the primary suspects. I consider both, however, recent changes are more likely.

Step 2: Analyzing the Deployments

`orders-ingest` Deploy: The "no schema change" explicitly states that this deploy wasn't intended to modify the structure of the data or its null/non-null characteristics. It makes this a less likely candidate, although it could potentially have introduced a bug.

`checkout-api` Deploy: This is where the crucial piece of information lies: "default empty string -> null for missing geo." This is a direct change to how missing data is represented. Previously, the system was storing missing values as empty strings. Now, it's storing them as `NULL`. This is key because `NULL` values are explicitly counted as nulls, while empty strings are not.

Step 3: Connecting the Change to the Alert

The timing is also important. The spike in the null rate occurred after the `checkout-api` deploy. This temporal proximity strengthens the link between the change and the alert.

The magnitude of the jump (0.3% to 17.8%) aligns with the number of orders previously represented by empty strings. The approximately 148,900 orders with nulls represent a significant portion of the 842,000 total orders in the last 7 days. This is consistent with a large number of values previously stored as empty strings now being represented as `NULL`.

Step 4: Forming a Conclusion

Based on the change, timing, and magnitude, the `checkout-api` deploy is the most likely root cause. The switch from empty string to `NULL` simply exposed a previously existing condition of missing data; it didn't create it.

3. Assumptions Made

Deployment Rollout: I assumed the `checkout-api` deployment was successfully rolled out and is functional. A failed or partial rollout could complicate the picture, but the alert suggests the change is taking effect.

Data Consistency: I assumed that the change in `checkout-api` is the only source of `shipping\_country\_code` data. If other services or processes also contribute to this column, the picture could be more complex.

Data Types &

Engineering counterpart

2:53 AM - Responded in 0s

Thanks - assumptions and validation path are reasonable. Run the confirmatory null-rate query before paging broader - spike may be ingest timing. Review complete - no further response needed.