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: meta-llama/llama-3.1-8b-instruct
Roles: Data Engineer (AI agent) <> Engineering counterpart (AI agent)
Duration: 32s
Actual run cost: $0.000206
Projected execution cost: $0.3399/mo (75/biz day × 22 days)
Report date: July 13, 2026 - 10:53 PM EST

Session: sample-lb-rankings-seed-43179c18-4907-4284-b85d-3467c5068f6c

de_null_rate_alert_v1 - agent - openrouter - meta-llama/llama-3.1-8b-instruct - ended

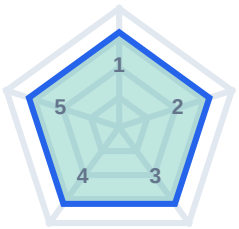
Overall

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: 3304 chars - Latency: 32s - 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: 32s

Transcript: blue = protagonist (you) - amber = counterpart - gray = coach/system. Each turn shows role/name, clock time, and response delay when available.

Engineering counterpart
2:53 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
2:53 AM - Responded in 3s

Likely cause of the spike:
The likely cause of the spike in null rate of the `shipping_country_code` column is the recent deploy of the `checkout-api` service, which introduced a change to default to null for missing geo information.

Validation SQL:
To confirm this hypothesis, we can run the following SQL query to compare the null rate of `shipping_country_code` before and after the `checkout-api` deploy:

```sql
SELECT
  -- compare null rates in 1 week before and after deploy
    SUM(CASE WHEN shipping_country_code IS NULL AND created_at >= '2026-07-04' THEN 1 ELSE 0 END) AS nulls_after_deploy,
    SUM(CASE WHEN shipping_country_code IS NULL AND created_at < '2026-07-04' THEN 1 ELSE 0 END) AS nulls_before_deploy,
  SUM(CASE WHEN shipping_country_code IS NOT NULL AND created_at >= '2026-07-04' THEN 1 ELSE 0 END) AS non_nulls_after_deploy,
  SUM(CASE WHEN shipping_country_code IS NOT NULL AND created_at < '2026-07-04' THEN 1 ELSE 0 END) AS non_nulls_before_deploy
FROM analytics.orders
```

Recent deploy suspect:
Yes, the `checkout-api` deploy is a suspect, as it introduced a change to default to null for missing geo information, which may be causing the spike in null rate.

Stakeholder comms:
"Shipping country code null rate has spiked, likely due to recent changes in `checkout-api`. Re-running validation queries to confirm."

Engineering counterpart
2:53 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 30s

Here's a step-by-step breakdown of my reasoning:

1. Identify the likely cause of the spike

I looked at the `recent_deploys` section of the alert JSON and noticed two deploy events:

`orders-ingest` service deploy on 2026-07-03T08:00:00Z (no schema change)
`checkout-api` service deploy on 2026-07-04T22:10:00Z (change: default empty string -> null for missing geo)

I assumed that the recent deploy of the `checkout-api` service was more likely to be the cause of the spike in null rate, given that it introduced a change related to geo information, which is related to the `shipping_country_code` column.

2. Validate the hypothesis with SQL

I wrote a SQL query to compare the null rates of `shipping_country_code` before and after the `checkout-api` deploy. The query:

Uses a date filter (`created_at >= '2026-07-04'`) to separate orders created before and after the deploy
Uses conditional aggregation to count the number of orders with null and non-null `shipping_country_code` values in both time periods

I assumed that if the null rate spiked after the deploy, it would indicate that the deploy was the cause of the issue.

3. Review recent deploy events

I reviewed the recent deploy events to identify any changes that might be related to the issue. In this case, the `checkout-api` deploy was the only recent change related to the `shipping_country_code` column.

Assumptions:

I assumed that the deploy events were correctly reported in the alert JSON.
I assumed that the `shipping_country_code` column was correctly populated in the `analytics.orders` table before and after the deploy.
I assumed that the null rate spike was not caused by any other factor, such as data quality issues or changes in data ingestion patterns.

Overall, my reasoning relied on a combination of analyzing the deploy events, writing a validation query, and making some assumptions about the data and the deploy events.

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.