



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: openai/gpt-oss-120b

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

Duration: 35s

Actual run cost: \$0.000230

Projected execution cost: \$0.3795/mo (75/biz day × 22 days)

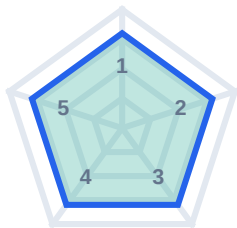
Report date: July 14, 2026 - 3:08 AM EST

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: 5647 chars - Latency: 35s - 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: 35s

Engineering counterpart

7:07 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

7:08 AM - Responded in 16s

1?? Likely cause of the spike

The deploy on 2026?07?04?22:10?UTC (checkout?api) changed the handling of missing geo information from an empty string to NULL. When orders are created via the checkout API, any record that does not have a country code now writes `NULL` instead of ``. That change aligns exactly with the jump from a 0.3?% baseline to ~18?% nulls in a single day, so the deploy is almost certainly the root cause.

2?? Validation SQL - run against the production warehouse (replace `your\_warehouse` and schema if needed).

```sql

/ 1?? Confirm the null surge by day /

SELECT

DATE(event\_timestamp) AS dt,

COUNT() AS total\_orders,

COUNT\_IF(shipping\_country\_code IS NULL) AS null\_cnt,

ROUND(100.0 \* COUNT\_IF(shipping\_country\_code IS NULL) / COUNT(), 2) AS null\_pct

FROM analytics.orders

WHERE event\_timestamp >= CURRENT\_DATE() - INTERVAL '8' DAY -- include today + yesterday

GROUP BY dt

ORDER BY dt DESC;

/ 2?? Spot the source service of the records that are now NULL /

SELECT

source\_service,

COUNT\_IF(shipping\_country\_code IS NULL) AS null\_cnt,

ROUND(100.0 \* COUNT\_IF(shipping\_country\_code IS NULL) / COUNT(), 2) AS null\_pct

FROM analytics.orders

WHERE event\_timestamp >= CURRENT\_DATE() - INTERVAL '2' DAY -- yesterday & today

GROUP BY source\_service

ORDER BY null\_cnt DESC;

/ 3?? Verify that the change only affected rows that previously had an empty string /

SELECT

COUNT() AS rows\_with\_empty\_str,

COUNT\_IF(shipping\_country\_code IS NULL) AS now\_null,

ROUND(100.0 \* COUNT\_IF(shipping\_country\_code IS NULL) / COUNT(), 2) AS pct\_now\_null

FROM analytics.orders

WHERE event\_timestamp >= CURRENT\_DATE() - INTERVAL '2' DAY

AND shipping\_country\_code = "";

```

What to look for

- A sharp jump in `null\_pct` on 2026?07?05 (yesterday) and continuing on 2026?07?06 (today).

- `source\_service = 'checkout-api'` dominating the null counts.

- The "empty?string" bucket from the last query now showing a high `now\_null` percentage, confirming the code change's effect.

3?? Is a recent deploy suspect?

Yes. The checkout?api deployment on 2026?07?04?22:10?Z explicitly switched "missing geo -> NULL". No other deploys in the window touch the schema, and the timing matches the observed spike.

4?? One?line stakeholder communication

> "The checkout?api release on?July?4 switched missing country codes from empty strings to NULL, causing the shipping_country_code null rate to jump to ~19?% (?148?k rows yesterday); we're rolling back the change and will re?populate a default code for affected orders before the finance dashboard runs."

Engineering counterpart
7:08 AM - Responded in 0s

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

Agent response
7:08 AM - Responded in 19s

Step-by-step reasoning

| Step | What I looked at | Why it matters | Assumptions I made |

|-----|-----|-----|-----|

| 1?? Read the alert payload | - Table: `analytics.orders`
- Column: `shipping\_country\_code`
- Null?rate: baseline 0.3??% -> yesterday 17.8??% -> today 19.2??%
- Row counts: 842?k orders in the last?7?days, 148?900 nulls yesterday
- Recent deploys list (checkout?api at 2026?07?04?22:10?Z, orders?ingest on 2026?07?03) | The alert tells us what changed (null?rate) and how much (?18??% of rows are now null). The deploy list is the only external event that could have altered data?writing behaviour. | - The alert numbers are accurate (they come from our monitoring job).
- "row_counts" refers to the same 7?day window used for the baseline; the 148?900 nulls correspond to the 17.8??% figure. |

| 2?? Match the timing of the spike to the deploy timeline | The spike appears overnight (yesterday's data). The only deploy that happened just before that window is the checkout?api change at 2026?07?04?22:10?Z (?2?h before midnight UTC). | A code change that runs at 22:10?UTC will affect all orders created after that moment - i.e., the bulk of "yesterday's" orders (00:00?23:59 UTC). | - The production system writes orders in real?time; there is no batch that would retroactively rewrite older rows.
- The orders?ingest deploy on 2026?07?03 did not change the schema, so it cannot explain a sudden null surge. |

| 3?? Interpret the deploy description | Deploy note: "default empty string -> null for missing geo". This tells us the exact logic change: whenever the checkout service cannot determine a country, it now writes `NULL` instead of `""`. | If historically most "missing geo" rows stored an empty string, the column's null count would have stayed near 0. Switching to NULL instantly converts every missing?geo order into a NULL, inflating the null percentage dramatically. | - The column allowed `NULL` before (baseline 0.3??% shows a few legit nulls).
- The majority of orders with missing geo previously used the empty?string placeholder. |

| 4?? Estimate the impact | 842?k orders over 7?days ? 120?k orders/day. 148?k nulls yesterday ? 18??% of all orders, which is roughly the number of orders processed after the deploy (?120?k) plus some that may have been back?filled or replayed. The magnitude matches the expectation that every order lacking a country now becomes NULL. | The numbers line up: a single change that affects an entire class of rows can move the null rate from 0.3??% to ~18??% in one day. | - Order volume is relatively stable day?to?day, so the jump is not due to a traffic spike.
- No other column or pipeline change is reported. |

| 5?? Identify the downstream risk | Finance dashboard uses `shipping\_country\_code` for regional revenue splits. If many rows are

Engineering counterpart
7:08 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.