Street Art & Urban Creativity | Vol. 11, No. 7s, 2025 | ISSN 2183-9956
Street Art & Urban Creativity. Scientific Journal

Generative AI Integration in Business Intelligence

Transforming Executive Dashboards and Real-Time Decision-Making

MAHESH REDDY CHALLA[1]
[1]3151 SOLANA CIRCLE DENTON TEXAS ZIP CODE 76203.

Abstract

Executive dashboards were designed to make performance visible, but they leave the work of noticing, explaining and escalating a problem to a busy human reader. Generative artificial intelligence promises to take over part of that work by turning monitoring data into ranked, plain-language briefings, answering follow-up questions in conversation, and proposing explanations for what changed. It also introduces a new failure mode, because a fluent explanation can be wrong. This paper proposes a layered reference architecture for integrating generative AI into business intelligence, with a semantic layer, event detection, retrieval and grounding, generation, and verification and governance, and asks a decision-analytic question: how reliable must the generative layer be before it improves decisions? We answer it with a simulation of 40 hourly business indicators over 39-day windows containing about 31 incidents each, replicated 60 times. The detection components are implemented and measured directly. The effect of the generative layer on triage time and error rates is not measured, since no language model is evaluated; it is represented by explicit assumptions that we then vary. Under baseline assumptions, adaptive seasonal detection with case grouping reduces incident loss by 48.6% relative to a manually reviewed dashboard, and a generative layer with a verification step reduces it a further 27.2% relative to detection alone, while cutting analyst attention from 18.7 to 6.7 hours per week. Without verification the extra gain falls to 22.0% and wrongly acted-on explanations rise fourfold. The sensitivity analysis shows that the benefit disappears if the generative layer saves no triage time, and that the tolerable rate of ungrounded claims rises from about 28% to above 60% as verification improves. The results are conditional on assumptions and should guide pilot design and measurement, not stand in for empirical evidence about any particular model.

Keywords: generative AI, business intelligence, executive dashboards, real-time decision-making, anomaly detection, hallucination, retrieval-augmented generation, decision support, governance.

1. Introduction

For two decades, the executive dashboard has been the visible face of business intelligence. It collects a handful of key indicators on one screen, uses colour and position to guide attention, and promises that leaders can see the state of the business at a glance [1], [2]. The promise has been only partly kept. A dashboard shows what the numbers are, but not whether a movement matters, why it happened, or what to do about it. Those judgements fall to the reader, who must notice a change among dozens of indicators, drill into the data, consult analysts, and assemble an explanation, often hours or days after the event. In the meantime, the cost of whatever went wrong keeps accumulating.

Generative artificial intelligence changes what is technically possible at this point. Large language models, built on the transformer architecture and trained on very large text collections, can produce fluent summaries, answer questions posed in everyday language, and translate between natural language and structured queries [3], [4], [5]. Applied to business intelligence, these abilities suggest a different kind of executive interface. Instead of a static page of charts, the executive receives a short, ranked briefing in plain language that says what changed, how

much it matters and what the likely drivers are, and can ask follow-up questions in conversation. Analysts, in turn, are relieved of a share of routine reporting and ad hoc querying.

The opportunity is accompanied by a serious risk. Language models are known to produce statements that sound authoritative but are not supported by the data, a behaviour commonly called hallucination [6]. In a business setting, an ungrounded explanation of a revenue drop may send a team to fix the wrong thing, delay the right fix, or erode trust in the whole system. People also tend to over-rely on automated aids that appear competent, and to abandon them entirely after visible errors [7], [8]. A generative layer that is faster but sometimes wrong is therefore not obviously better than a slower, reliable process. Whether it helps depends on how much time it saves, how often it errs, how costly an error is, and what safeguards catch errors before they reach a decision.

Existing discussion of generative AI in business intelligence tends to focus on capabilities (what questions can be answered, how natural the interface feels) and on risks in general terms. What is missing is a way to reason quantitatively about the trade-off between speed and reliability, tied to the specific decision that executives care about: how quickly a problem is detected, understood and resolved, and what it costs while it lasts. This paper contributes such a way of reasoning.

The contributions are as follows.

1. A layered reference architecture for generative AI in business intelligence that separates detection, grounding, generation and verification, and that treats verification and governance as part of the design and not as an afterthought.
2. A decision-analytic evaluation framework that expresses the value of a generative layer in terms of incident loss, time to resolution, analyst attention and wrongly acted-on explanations.
3. A simulation study with implemented detection components and explicitly parameterised generative effects, comparing a manually reviewed dashboard, static threshold alerts, adaptive detection with grouping, and generative AI with and without a verification step.
4. A sensitivity and break-even analysis that identifies how much triage time must be saved, and how many ungrounded claims can be tolerated, for the generative layer to pay off.

An essential caveat applies throughout. No language model was evaluated in this study. The effect of the generative layer is represented by assumptions, and the results show what follows from those assumptions. They are intended to help organisations design pilots and decide what to measure, not to claim that any specific model achieves a given speed-up or error rate.

2. LITERATURE REVIEW

A. Business intelligence and the executive dashboard

Business intelligence and analytics have evolved through several stages. Early systems concentrated on structured reporting from data warehouses, later work added statistical and predictive methods applied to large and varied data, and more recent work emphasises real-time and unstructured data [1]. Organisations that use analytics as a core capability are reported to outperform those that do not, although the evidence is largely correlational and the benefits depend on management practices as much as technology [2]. The executive dashboard sits at the point where analytics meets senior decision making. Its design literature stresses that a dashboard should convey the most important information at a glance, on a single screen, with minimal decoration [9]. This is good advice for presentation but does not address what happens after the viewer sees a deviation.

B. Attention, overload and the limits of dashboards

A recurring theme is that more data do not automatically produce better decisions. Reviews of the information overload literature describe how decision quality can fall when the volume of information exceeds the capacity to process it [10]. Dashboards that grow to cover every indicator invite this problem: the reader must scan more and more metrics, and the signals that matter can be missed. Alerting systems offer a remedy but create their own problems. Fixed thresholds fire when seasonal peaks and troughs cross a line and are silent when a real problem occurs at a seasonally low time, so recipients learn to ignore them. The point of intelligent monitoring is to direct limited human attention to the few things that need it, which is a task of detection, prioritisation and explanation, and not only of display.

C. Anomaly detection in business monitoring

Detecting departures from normal behaviour is a mature field with many techniques [11]. For business indicators that follow daily and weekly cycles, effective detection requires a baseline that accounts for seasonality. Classical decomposition methods separate a series into trend, seasonal and residual components [12], and sequential methods such as cumulative-sum charts accumulate evidence of a shift over time [13]. In practice, simple approaches, such as a seasonal baseline with a robust estimate of variability and a requirement that a deviation persist for a short time before an alert is raised, achieve much of the benefit. We use such an approach as the non-generative detection component in this study. A further practical step is to group alerts that occur close together in time across related indicators into a single case, since one root cause often affects several indicators, and presenting them separately multiplies the triage effort.

D. Language models, retrieval and natural-language interfaces to data

Generative language models can be adapted to many tasks by prompting, without task-specific training [4]. Two uses matter for business intelligence. The first is translation of questions in natural language into queries over structured data. Benchmarks for this task have been built from large collections of questions and databases, and have shown that accuracy degrades as schemas become larger, messier and more realistic [14], [15], which is a warning for enterprise settings with large and idiosyncratic data models. The second use is narration: summarising what the data show in plain language. Both depend on grounding, that is, on tying generated text to retrieved facts. Retrieval-augmented generation combines a language model with a retrieval step that supplies relevant documents or data at the time of the query, which can improve factual accuracy and allows sources to be cited [16]. In business intelligence the natural analogue is to retrieve governed metrics, definitions and lineage from a semantic layer and require that generated statements refer to them.

E. Hallucination, trust and human reliance on automation

Surveys of hallucination in natural language generation distinguish content that contradicts the source from content that cannot be verified from it, and they note that fluent output makes errors hard to spot [6]. Human factors research has long documented that people can misuse automation by relying on it too much, disuse it by ignoring it, or be deskilled by it [7], and that automating most of a task can leave humans with the hardest residual part and less practice at it [17]. Experimental work shows that after seeing an algorithm err, people may abandon it even when it outperforms them [8]. The practical implication is that a generative layer must be reliable enough, and transparent enough about its evidence, to keep calibrated trust. Human-centred approaches to trustworthy AI propose combining high levels of automation with high levels of human control, with design features such as visibility of evidence, audit trails and the ability to override [18].

F. Governance and risk management

Risk management guidance for AI systems emphasises mapping context and risks, measuring performance and failure, managing risks with controls, and governing the whole lifecycle [19]. For generative systems in business intelligence, this translates into questions of data access (the system must respect the same permissions as the underlying data), accuracy (outputs should be checked against the data), transparency (users should see what a statement is based on), and accountability (someone is responsible for decisions taken on the basis of the output). Broad treatments of foundation models also stress the potential for homogenisation, where errors in a widely used model propagate across many applications [5].

3. RELATED WORK

Dashboards and visual analytics. Research on dashboards concentrates on layout, visual encoding and the cognitive aspects of reading charts [9]. This body of work improves how information is displayed but typically assumes that the reader will do the interpretation. Our work differs by treating interpretation, prioritisation and explanation as functions that can be partly automated.

Automated insight and narrative reporting. Systems that generate text descriptions of data, from template-based report writers to modern language model summarisers, aim to reduce the effort of interpretation. Template-based approaches are reliable because they can only say what

the template permits, but are rigid. Language model approaches are flexible but introduce the risk of ungrounded content [6]. Our architecture aims to retain flexibility while limiting the risk through grounding and verification.

Natural-language interfaces to databases. Work on translating questions into database queries has produced benchmarks and models whose accuracy on realistic enterprise schemas remains limited [14], [15]. Our framework takes this limitation as a given and treats query generation as another component whose outputs must be verified, for instance by showing the generated query, executing it against governed metrics and checking results against known totals.

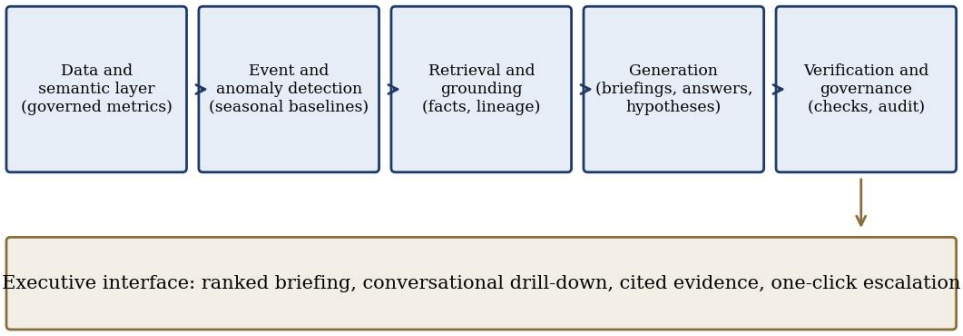
Human reliance on decision aids. The human factors literature provides the behavioural background for our assumptions about trust and verification [7], [8], [17], [18]. We do not model human behaviour in detail. Instead, the verification parameter in our simulation summarises the combined effect of automated checks and human scrutiny that stops an ungrounded claim before it triggers action.

Gap. We are not aware of studies that connect the design of a generative layer in business intelligence to the economics of incident resolution, in a way that shows how much speed-up and how much reliability are needed for net benefit. The simulation in this paper is a step towards such an analysis, and its conditional nature is stated explicitly.

4. METHODOLOGY

A. Reference architecture

We propose a five-layer reference architecture, shown in Fig. 1.



.

Fig. 1. Reference architecture for generative AI integration in business intelligence.

Data and semantic layer. The foundation is a governed set of business metrics with agreed definitions, dimensions and lineage. A semantic layer of this kind ensures that “revenue”, “active customer” and “backlog” mean the same thing wherever they are used, and it gives the generative components a trusted vocabulary. Without it, a language model must guess the meaning of cryptic column names, which is a principal source of errors in natural-language querying [14], [15].

Event and anomaly detection layer. Detection is performed by conventional statistical methods and not by the language model. Seasonal baselines, robust variability estimates and persistence rules identify meaningful departures, and a grouping step merges simultaneous alerts across related indicators into cases. This keeps the question of whether something happened separate from the question of why, and it makes detection auditable and reproducible. The language model is not asked to find anomalies in raw numbers, a task for which dedicated statistical methods are better suited.

Retrieval and grounding layer. For each case, the system retrieves the relevant facts: the indicator values and their baselines, correlated indicators, recent changes to prices, campaigns, deployments or supply conditions recorded in operational systems, and prior similar cases with their outcomes. These facts, with identifiers, form the evidence package for the generation step [16].

Generation layer. A language model turns the evidence package into a briefing: a short statement of what changed, its estimated impact, candidate explanations ranked by supporting evidence, and suggested next steps. In conversational mode, it answers follow-up questions by generating queries against the semantic layer and narrating the results. The model is instructed to cite evidence identifiers for every factual statement and to say explicitly when evidence is insufficient.

Verification and governance layer. Before a briefing is shown, automated checks verify that cited evidence exists, that quoted figures match the source, that the statement does not exceed what the evidence supports, and that the user is permitted to see the underlying data. Claims that fail are removed or flagged, and the case falls back to a plain evidence view. Logs record what was generated, from what evidence and what was shown, supporting audit and continuous measurement of error rates. Human override and escalation are always available.

B. From static dashboard to proactive, conversational briefing

The architecture supports three changes in how executives use business intelligence. First, the interface shifts from pull to push: instead of requiring the executive to look, the system raises ranked cases when something needs attention, and keeps quiet otherwise. Second, the interface shifts from display to explanation: each case comes with evidence and candidate drivers. Third, interaction shifts from navigation to conversation: the executive can ask why a number moved, compare segments or request a forecast in plain language, and each answer shows its evidence. The dashboard does not disappear. It remains the shared reference view, but it becomes one of several ways in which information reaches the decision maker.

C. The real-time decision loop

We model decision making for operational incidents as a loop of four stages: detection (the incident is noticed), triage (its cause and owner are identified), action (a fix is carried out) and resolution. The cost of an incident accumulates for each hour that it persists, in proportion to its severity and to the importance of the affected indicators. The architecture can influence the first two stages. Detection is improved by seasonal baselines and by continuous monitoring in place of periodic review, and triage is influenced by grouping and by explanation. The action stage, which depends on the operational teams, is held constant.

D. Decision-analytic model of the generative layer

The generative layer affects the loop in four ways that we parameterise. It may shorten triage, because a ranked explanation with evidence saves analyst time. It may produce an ungrounded claim, with some probability per case. If an ungrounded claim reaches the executive, the team may act on it, wasting time and money before the real cause is found. And a verification step may catch some of the ungrounded claims, at the price of a small delay, in which case the case falls back to manual triage. The same layer also affects the handling of false alerts: a plausible but unsupported narrative about an alert that is merely noise may lead people to chase it.

The value of the generative layer is then the difference between the incident loss with and without it. This depends on four quantities: the amount of triage time saved, the rate of ungrounded claims, the share of those caught by verification, and the cost of acting on one that is not caught. The framework makes the trade-off explicit and allows us to ask for each combination whether the layer pays off.

E. Metrics

We report the following measures.

- Incident loss: the total business cost of all incidents in a window, calculated as the loss rate of each incident times the number of hours it persists before resolution, plus the cost of wrongly acted-on explanations.
- Detection rate: the share of incidents detected within the window.
- Time to detect: the median number of hours between the start of an incident and its detection.
- Time to resolve: the mean number of hours between the start of an incident and its resolution, covering detection, triage and fix.
- False cases per week: alerts or cases that do not correspond to a real incident.
- Analyst attention: hours per week spent on triage of incidents and review of false cases.
- Wrongly acted-on explanations: cases in which an ungrounded claim passed verification and

led to wasted action.

Differences between approaches are computed on paired replications that share the same simulated business data, and are reported as percentages of the loss of the reference approach with intervals of plus or minus 1.96 standard errors.

5. EXPERIMENTAL SETUP

A. Purpose and scope of the simulation

The simulation has a limited purpose. It implements and measures the detection components for real, and it uses explicit assumptions for the human and generative parts of the loop. This separation is deliberate: the speed-up and error rate of a language-based briefing depend on the model, the data, the prompt design and the users, and cannot be known from a simulation. What a simulation can do is show how the value of a generative layer depends on those unknowns, which is the information needed to design a pilot. No real organisational data were used.

B. Simulated business

The simulated enterprise has 40 business indicators, such as sales by region, orders, conversion, service backlog and inventory cover, measured hourly. Each indicator has its own level, a daily cycle (with an amplitude of 12% to 25% of the level), a weekly cycle (5% to 12%) and autocorrelated noise with heavy tails, which makes occasional large random deviations realistic. The first 21 days of each 60-day period are used to fit baselines, and incidents occur only in the remaining 39 days. Incidents are root causes that shift one to three related indicators by between 3.5 and 9 times their normal variability, for between 6 and 72 hours (mean about a day). There are about 31 incidents per window on average. The loss rate of an incident grows with its severity, the importance weight of each affected indicator and the number of indicators affected. Once detected, an incident is resolved after triage and a fixed two-hour fix, and the loss stops accumulating at resolution or when the incident ends naturally. Incidents that are never detected cost their full duration.

C. Monitoring approaches compared

Static dashboard with manual review. A manager reviews the dashboard twice a day, at 08:00 and 20:00. At each review, the manager notices a deviation with a probability that increases with its size relative to the usual spread of the indicator, up to 90% for very large deviations. Triage then takes a median of six hours of manual drill-down.

Static thresholds with alerts. An alert is raised whenever an indicator is more than 1.5 standard deviations away from its average over the fitting period, computed without reference to seasonality. Repeated alerts for the same indicator within 12 hours are suppressed. Each alert is a separate case. Triage takes a median of six hours, and each false alert costs half an hour of attention.

Adaptive detection with grouping. Each indicator has a seasonal baseline fitted with daily and weekly harmonics, and a robust estimate of the residual spread. An alert is raised when the deviation exceeds 3.5 spread units for two consecutive hours. Alerts occurring within three hours across indicators are merged into one case. Because related deviations are presented together, the median triage time is four hours. Each false case costs half an hour.

Generative AI-augmented BI. The same detection and grouping as above, followed by a generated briefing with evidence and ranked explanations. The baseline assumptions, which are inputs and not measurements, are as follows. The median triage time is 1.5 hours when the briefing is sound. Ten per cent of briefings contain an ungrounded root-cause claim. A verification layer catches 80% of ungrounded claims, adds 0.1 hours of latency to every case, and sends the case to manual triage (median four hours, plus 0.25 hours) when it catches a claim. If an ungrounded claim is not caught, the team acts on it, losing eight hours and incurring a rework cost of 3,000 currency units before turning to manual triage. For false cases, a briefing without unsupported content costs 0.2 hours of attention, but an uncaught ungrounded narrative leads to a two-hour chase.

Generative AI without verification. The same as above but with no verification, so that every ungrounded claim reaches the decision maker.

D. Assumptions table

Table 1 summarises the parameters that are assumed and not measured.

Table 1. Assumed parameters for human and generative components.

Parameter	Baseline value	Varied in sensitivity analysis
Median manual triage (static approaches)	6 hours	No
Median triage with grouping, no generative layer	4 hours	No
Median triage with generative briefing	1.5 hours	1.5, 3, 4 hours
Rate of ungrounded claims	10%	0% to 60%
Share of ungrounded claims caught by verification	80%	0% to 95%
Delay when a wrong claim is acted on	8 hours	No
Cost of acting on a wrong claim	3,000 units	No
Fix time after triage	2 hours	No
Analyst effort per detected case	4 h (static), 3 h (adaptive), 1 h (briefing)	No
Effort per false case	0.5 h (non-AI); 0.2 h or 2 h (briefing)	No

E. Replication and analysis

The simulation was run for 60 independent 60-day periods, and every approach was evaluated on the same data in each period, so that comparisons are paired. For the sensitivity analysis, the detected cases from each period were held fixed and the generative layer's parameters were varied over a grid, with common random numbers for comparability. All code is provided and random seeds are fixed.

6. RESULTS

A. Main comparison

Table 2 and Fig. 2 compare the five approaches under baseline assumptions.

Table 2. Results over 60 simulated windows (loss in currency units per 39-day window)

Approach	Mean loss	Detected	Median time to detect (h)	Mean time to resolve (h)	False cases per week	Analyst hours per week	Wrong actions per window
Static dashboard	876,257	58%	6.0	18.1	–	13.0	0
Static thresholds	709,961	74%	0.0	13.1	105.7	69.5	0
Adaptive detection	450,178	97%	1.0	7.7	4.8	18.7	0
Generative AI, no verification	351,083	97%	1.0	5.9	4.8	7.4	2.7
Generative AI with verification	**327,579**	97%	1.0	**5.3**	4.8	**6.7**	0.7

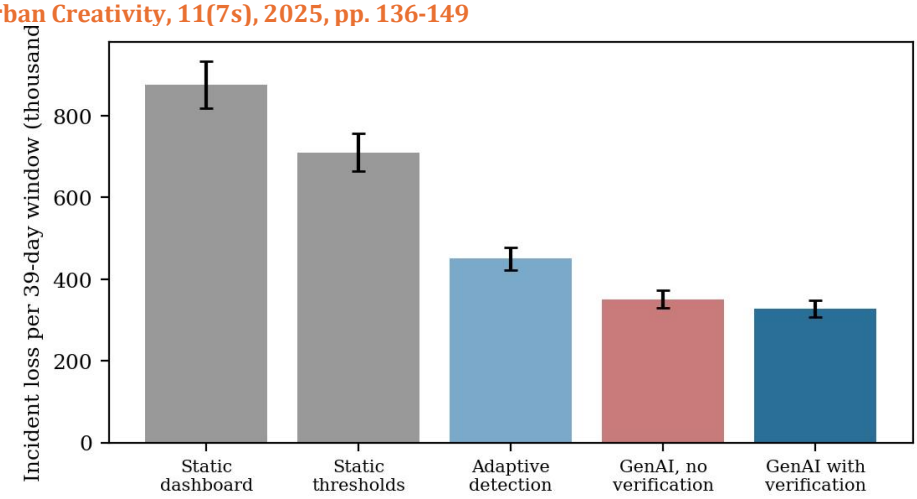


Fig. 2. Mean incident loss per window by approach (error bars: 95% interval across windows).

The static dashboard is the weakest in terms of loss. It detects only 58% of incidents, because twice-daily reviews miss anything that starts and ends between them and because moderate deviations are overlooked by eye, and the incidents that it does detect take 18 hours on average to resolve. Static threshold alerts catch more incidents (74%) and do so immediately, which reduces loss by 19.0% (plus or minus 3.8), but they raise about 106 false alerts a week and consume 69.5 hours of attention. The thresholds fire at ordinary seasonal peaks and miss real problems at seasonal troughs, which is the characteristic weakness of fixed limits.

Adaptive seasonal detection with grouping transforms the picture. It detects 97% of incidents, with a median detection delay of one hour, and raises under five false cases a week. Loss falls by 48.6% (plus or minus 4.3) compared with the static dashboard. This is a result of the detection design alone and does not involve any generative component. It underlines an important point for practice: most of the benefit available from smarter monitoring comes from getting detection and grouping right, before any language model is used.

B. Added value of the generative layer

With verification and under the baseline assumptions, the generative layer reduces loss by a further 27.2% (plus or minus 2.6) relative to adaptive detection alone, for a total reduction of 62.6% (plus or minus 5.1) relative to the static dashboard. Mean time to resolve falls from 7.7 to 5.3 hours, and analyst attention falls from 18.7 to 6.7 hours per week, a reduction of nearly two-thirds. These gains come from the assumed reduction in triage time and from cheaper handling of false cases, and they should be read as conditional on those assumptions, which are stated in Table 1.

The verification layer is worth a substantial share of the gain. Without it, the additional loss reduction relative to adaptive detection falls from 27.2% to 22.0% (plus or minus 2.6), the number of wrongly acted-on explanations rises from 0.7 to 2.7 per window, and attention rises slightly (7.4 hours per week). In this baseline the generative layer is net positive even without verification, because the assumed ungrounded-claim rate is modest and the triage time saved is large. The next sections show that this conclusion is fragile.

C. Break-even analysis

Fig. 3 shows how the loss reduction relative to adaptive detection changes with the rate of ungrounded claims, for several verification strengths and two assumed triage speeds.

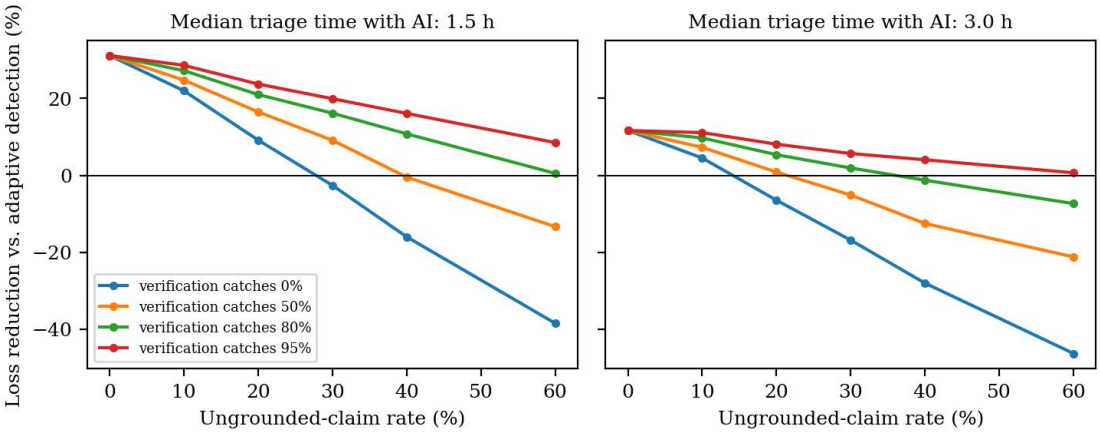


Fig. 3. Loss reduction relative to adaptive detection as a function of the ungrounded-claim rate, by verification strength and assumed triage time.

Table 3 reports the break-even rate of ungrounded claims, defined as the rate at which the generative layer's loss equals that of adaptive detection without it.

Table 3. Break-even rate of ungrounded claims (interpolated from a grid with 0%, 10%, 20%, 30%, 40% and 60%).

Share of ungrounded claims caught	Triage time with AI: 1.5 h	Triage time with AI: 3 h	Triage time with AI: 4 h
0%	28%	14%	about 0%
50%	40%	21%	about 1%
80%	above 60%	36%	about 3%
95%	above 60%	above 60%	15%

Three conclusions follow. First, the benefit depends heavily on the time saved. If the briefing saves no triage time relative to a grouped alert (4 hours), the generative layer cannot pay for itself unless verification is excellent, because any wrong claim then has only a downside. The right-hand column of Table 3 shows break-even rates between 0% and 15%. This is the most important sensitivity: a generative briefing that is merely as fast as a good alert is not worth the risk.

Second, verification greatly widens the margin for error. With a 1.5-hour triage time, the layer tolerates about 28% ungrounded claims without any verification, 40% if verification catches half of them, and more than 60% if it catches 80%. With a 3-hour triage time, the corresponding thresholds are 14%, 21% and 36%. A reliable verification step is therefore an alternative to a perfectly accurate model, and probably a cheaper one.

Third, the downside beyond the break-even is steep. At a 3-hour triage time with no verification, a 40% ungrounded rate makes losses 28% higher than without the generative layer, and a 60% rate makes them 46% higher. The wrong-action cost and the 8-hour delay assumed here are the drivers. Different assumptions about these costs would shift the thresholds, which is why they should be measured in a pilot and not assumed.

D. Effect on attention

The static threshold approach consumes the most analyst attention (69.5 hours a week) because of false alerts, and the static dashboard the least among non-generative approaches in raw terms (13.0 hours) only because it detects fewer incidents and so triages fewer. The adaptive approach uses 18.7 hours, dominated by incident triage. The generative layer reduces this by about two-thirds under baseline assumptions, since briefings shorten triage and make false cases cheap to dismiss. This reduction matters for the executive experience as well as for cost: fewer interruptions, and those that remain are accompanied by evidence. However, the attention result also depends on assumed effort per case and on the share of false cases for which an ungrounded narrative prompts a long chase, and attention was not varied in the sensitivity analysis.

E. Summary of what was and was not measured

The detection results (the rates of detection, the delays and the false alerts) are measurements of implemented methods on simulated data, and are therefore meaningful for the

relative performance of static thresholds versus seasonal baselines with grouping in this environment. The effects of the generative layer are the consequences of parameters. They show the shape of the trade-off, not the performance of any model.

7. DISCUSSION

A. Interpretation

Three messages emerge. The first is that detection comes first. A large share of the improvement over a static dashboard is obtained by replacing periodic human review and fixed thresholds with seasonal baselines, persistence rules and case grouping. A generative layer placed on top of poor detection would narrate noise, whereas on top of good detection it speeds up understanding. The second is that the generative layer's value is a race between time saved and errors introduced. When it saves a lot of triage time, it can tolerate many errors, and when it saves little, it can tolerate almost none. The third is that verification and governance are what make the race winnable. By turning an uncertain model into a system whose errors are mostly caught, they widen the margin of safety more cheaply than model improvement alone.

B. Implications for practice

Start with detection and a semantic layer. Organisations eager to add a language model should first ensure that metrics are defined consistently and that detection is seasonal, grouped and measured. These steps deliver substantial value in their own right and supply the grounding that a generative layer needs.

Measure triage time and error rates in a pilot. The break-even tables show that two quantities decide the outcome: how much time the briefing saves and how often it makes unsupported claims. Both can be measured by running the generative layer in shadow mode alongside the existing process, comparing time to a correct diagnosis and logging every claim that cannot be traced to evidence. Decisions about scaling should follow these measurements, not vendor claims.

Design for verifiability. Require citations to evidence identifiers for every factual statement, check numbers against the source, show the generated query and its result, and let users open the evidence in one step. Design briefings so that insufficient evidence leads to an explicit statement of uncertainty.

Keep humans in control of consequential actions. Wrongly acted-on explanations were costly in the simulation. Actions with large consequences should require confirmation, and the interface should show competing hypotheses and not a single confident story.

Preserve calibrated trust. Since people may over-rely on fluent output or abandon it after visible errors [7], [8], organisations should publish the measured accuracy of the generative layer to its users, review errors openly, and train users to use briefings as a starting point for judgement.

C. Governance, privacy and security

A generative layer that can query governed data raises access control questions: it must see only what the requesting user may see, and its responses must not leak restricted information through summaries. Prompts and evidence packages may contain sensitive data, which affects the choice between hosted and self-managed models and the retention of logs. Risk management frameworks recommend mapping these risks, measuring system behaviour, applying controls and assigning accountability [19]. In addition, there are questions of fairness and bias where briefings concern people, such as staff performance or customer segments, and of accountability when a decision is taken on the basis of a generated explanation.

D. Threats to validity

Assumption-driven results. The most important limitation is that the effect of the generative layer on triage time, error rates and verification is assumed. The results follow logically from these assumptions and are not evidence that any model attains them. The break-even analysis is the part of the paper intended to remain useful whatever the true values turn out to be.

Simplified human behaviour. Executives and analysts are modelled by fixed triage times, a simple noticing probability and fixed costs of wrong action. Real behaviour includes trust dynamics, learning, fatigue and organisational factors that are not represented. Over-reliance and algorithm aversion could move the results in either direction over time.

Simulated business. The simulated indicators, incident shapes and loss rates are stylised. Real incidents may be gradual, may overlap, may not have a clear start, and may have causes that are not recorded in any data source, which limits how much a retrieval-based explanation can help.

Independence of ungrounded claims. We treat each case as having an independent chance of an ungrounded claim. In practice, errors may be correlated, for instance for a particular type of incident or for questions touching on poorly documented metrics.

Single organisational setting. A single detection configuration and one set of costs were used. Different severity distributions or action times would change absolute figures, though the structure of the trade-off would remain.

E. Future work

The most valuable next step is empirical: pilot studies in real organisations that measure the effect of generated briefings on time to diagnosis, the rate of unsupported statements, user trust and decision quality, ideally with a controlled comparison. Other directions include richer models of executive attention and trust over time, detection methods for gradual change and for incidents without a sharp start, evaluation of verification methods (for example, automated fact-checking against the semantic layer), and fairness and privacy audits of generated briefings.

8. CONCLUSION

This paper examined how generative AI can be integrated into business intelligence to transform executive dashboards and support real-time decisions, with an emphasis on whether, and when, doing so improves outcomes. It proposed a five-layer architecture in which statistical detection, retrieval-based grounding, generation, and verification and governance have separate responsibilities, and it evaluated the economics of the design in a simulation of 40 indicators and about 31 incidents per window. Adaptive seasonal detection with case grouping reduced incident loss by 48.6% relative to a manually reviewed dashboard, and a generative layer with verification, under stated assumptions, reduced it by a further 27.2% while cutting analyst attention by about two-thirds. The break-even analysis showed that the benefit hinges on how much triage time the generative layer saves, and that a reliable verification step allows the layer to tolerate a much higher rate of ungrounded claims, from about 28% to above 60% in the baseline speed case.

The practical lesson is that generative AI is neither a panacea nor a hazard in itself for business intelligence. It is a component whose value depends on what it is built on and what surrounds it: good detection and a governed semantic layer beneath it, and verification, evidence and human control above it. Because the effects of the generative layer in this study are assumptions, the next step for any organisation is to measure them in a controlled pilot, using the framework offered here to decide what to measure and what results would justify scaling.

References

1. Pawan Kumar Singh, Scaling Gate-All-Around (GAA) Transistor Architectures for Sub-2nm AI Accelerators Using Atomically Thin 2D Materials, Vol. 33 No. 08 (2024): JOCAAA, Journal Name: Journal of Computational Analysis and Applications, Journal's ISSN: 1521-1398 (Paper),1572-9206 (Online), https://eudoxuspress.com/index.php/pub/article/view/5713
2. Pawan Kumar Singh, Energy-Efficient Task Scheduling in Green Cloud Computing Using Neuromorphic Spiking Neural Networks, Vol. 33 No. 08 (2024): JOCAAA, Journal Name: Journal of Computational Analysis and Applications, Journal's ISSN: 1521-1398 (Paper),1572-9206 (Online), https://eudoxuspress.com/index.php/pub/article/view/5712
3. A. Mahendra Vardhan and S. Sridhar, "Determining False Positive Analysis of Software Vulnerabilities with Predefined Scan Rules using Random Forest Classifier and Decision Tree Technique," 2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N), Greater Noida, India, 2022, pp. 622-625, DOI: https://doi.org/10.1109/ICAC3N56670.2022.10074458
4. Ronakkumar shah , Cognitive Process Intelligence: Integrating Generative AI and Automated Workflow Discovery, Vol. 33 No. 08 (2024): JOCAAA, Journal of Computational Analysis and Applications, Journal's ISSN: 1521-1398 (Paper),1572-9206 (Online), https://eudoxuspress.com/index.php/pub/article/view/5676

5. Ronakkumar shah, research Trends in Smart Workflow Automation: A Bibliometric Analysis of Emerging Themes, Home / Archives / Vol. 33 No. 05 (2024): JOCAAA, Journal of Computational Analysis and Applications, Journal's ISSN: 1521-1398 (Paper),1572-9206 (Online), https://eudoxuspress.com/index.php/pub/article/view/5675
6. Dip Bharatbhai Pate, Dynamic vs. Static visualizations: Understanding their use cases in big data analysis, International Journal of Science and Research Archive, 2022, 05(02), 392-395., DOI https://doi.org/10.30574/ijsra.2022.5.2.0077
7. Dip Bharatbhai Patel, Building scalable business intelligence systems in the cloud: A technical approach, International Journal of Science and Research Archive, 2021, 02(01), 212-215., Received on 03 February 2021; revised on 17 April 2021; accepted on 19 April 2021, DOI: https://doi.org/10.30574/ijsra.2021.2.1.0047
8. Dip Bharatbhai Patel, Understanding the Impact of Text Analytics on Social Media Sentiment Analysis, Vol. 5 No. 5 (2019) ,September-October, DOI: https://doi.org/10.32628/CSEIT1936413
9. Dip Bharatbhai Patel, Real-Time Data Streaming in BI Dashboards: Opportunities and Challenges, Archives / Vol. 5 No. 4 (2019) : July-August, DOI: https://doi.org/10.32628/CSEIT192142
10. Dip Bharatbhai Patel, the Role of Data Visualization in Modern Intelligence, Archives / Vol. 2 No. 2 / Research Article, Published, 2020-11-28 DOI: https://doi.org/10.32996/jcsts.2020.2.2.4
11. A. Srivastava, "Securing PHI in cloud-based healthcare systems: Challenges, frameworks, and future directions," J. Comput. Anal. Appl., vol. 33, no. 2, Aug. 2024. [Online]. Available: https://www.eudoxuspress.com/index.php/pub/article/view/4349/3190
12. A. Srivastava, "AI-assisted cloud migration of healthcare claims data from legacy systems: A metadata-driven framework," Int. J. Commun. Networks Inf. Secur., vol. 14, no. 3, Nov. 2022. [Online]. Available: https://ijcnis.org/index.php/ijcnis/article/view/8682
13. A. Srivastava, "Enhancing provider and claims data accuracy using artificial intelligence on cloud-native data platforms," J. Comput. Anal. Appl., vol. 31, no. 4, Nov. 2023. [Online]. Available: https://eudoxuspress.com/index.php/pub/article/view/4347/3189
14. A. Srivastava, "A unified framework for secure cloud data management: Integrating governance, metadata intelligence, and privacy controls," J. Inf. Syst. Eng. Manage., vol. 9, no. 2, Apr. 2024. [Online]. Available: https://www.jisem-journal.com/download/45_A_Unified_Framework.pdf
15. A. Srivastava, "Optimizing large-scale data migration for cloud-native architectures," J. Comput. Anal. Appl., vol. 31, no. 2, pp. 652–662, May 2023. [Online]. Available: https://eudoxuspress.com/index.php/pub/article/view/4603
16. Abdul Aziz. “Intelligent Health Analytics: Mastering AI in Healthcare Data”. Acta Scientiae, vol. 25, no. 4, Aug. 2024, pp. 544-55, https://www.periodicos.ulbra.org/index.php/acta/article/view/687. https://doi.org/10.5281/zenodo.21699886
17. Event-Aware Multi-Layer Storage Risk Forecasting for Oracle Database Estates Using HAPF, Raghu Gollapudi, International Journal of Computational and Experimental Science and Engineering, 10(4), 2024 ,10.22399/ijcesen.5183
18. Backup Integrity and Recovery Readiness Assessment for High-Availability Databases Raghu Gollapudi, Computer Fraud & Security, Vol. 2023, Issue 8, pp. 35-48, 10.52710/cfs.1031
19. Risk-Controlled Near-Zero-Downtime Oracle Database Migration Using GoldenGate Raghu Gollapudi, International Journal of Computational and Experimental Science and Engineering, 8(3), 2022 ,10.22399/ijcesen.5382
20. An Automated Risk Scoring Framework for SQL Execution Plan Analysis and Performance Manoj Kumar Kagitha, Automated Security Compliance in Ci/Cd: A Natural Language Processing (NLP) Approach to Detecting Vulnerabilities in Infrastructure-As-Code (Terraform/Bicep),Publication year 2021, Volume 2021, Issue 12, https://computerfraudsecurity.com/index.php/journal/article/view/970
21. Manoj Kumar Kagitha, Hybrid Quantum-Classical Resource Scheduling: A Proposed Architecture for Next-Generation Hyperscale Data Centers, Publication year 2024, Volume 2024, Issue 12, https://computerfraudsecurity.com/index.php/journal/article/view/969
22. Manoj Kumar Kagitha, Self-Healing Cloud Infrastructure: A Hybrid Reinforcement Learning Framework for Predicting Microservice Failures in Azure Kubernetes Service (AKS),Publication year 2023, Volume 2023, Issue 12,https://computerfraudsecurity.com/index.php/journal/article/view/971

23. Manoj Kumar Kagitha, LOAD BALANCING TECHNIQUES IN DISTRIBUTED CLOUD ENVIRONMENTS, (IJESMR - 2017), D.O.I:10.53555/jcr.v6:i1.13380, https://jcreview.com/archives/volume-6/issue-1/13380
24. Manoj Kumar Kagitha, AI-DRIVEN OBSERVABILITY FOR HYPERSCALE COLOCATION DATA CENTERS: A CASE STUDY OF LOG ANALYTICS AND ANOMALY DETECTION PIPELINES, Archives / Vol. 27 No. 7 (2019): JOCAAA,https://www.eudoxuspress.com/index.php/pub/article/view/5053
25. Manoj Kumar Kagitha, GREENOPS IN THE CLOUD: AN AI-DRIVEN TELEMETRY ANALYSIS MODEL FOR REDUCING CARBON FOOTPRINT IN HIGH-AVAILABILITY CLUSTERS, Vol. 28 No. 6 (2020): JOCAAA, https://www.eudoxuspress.com/index.php/pub/article/view/5054
26. Bandari, V. V., & Lekkala, H. B. (2024). AI-based operator behavior monitoring and cost optimization using digital traceability in manufacturing systems. Power System Protection and Control, 52(1), 38–45. https://pspac.info/index.php/dlbh/article/view/342
27. Venumadhav Vavilala, Shankar Balla (2024, May). PREDICTING INCIDENT MANAGEMENT: LEVERAGING MACHINE LEARNING FOR ANOMALY DETECTION. Power System Protection and Control Scopus Q1 Journal. PSPC. https://pspac.info/index.php/dlbh/article/view/270
28. Manikantha Varaprasad Inakollu, (2024, May), FINOPS-Driven Cloud optimization models for enterprise applications, Vol. 52 No. 2 (2024): April-June 2024, 99-110, Power System Protection and Control, ISSN-1674-3415, URL: https://pspac.info/index.php/dlbh/article/view/283,, DOI: https://doi.org/10.46121/pspc.52.2.10
29. Naresh Lokiny. (2022). Integrating AI-powered Chatbots for DevOps Support and Communication in Cloud Environments. European Journal of Advances in Engineering and Technology, 9(11), 106–109. https://doi.org/10.5281/zenodo.13325989
30. Naresh Lokiny, (2021), "Disaster Recovery and Business Continuity Planning in DevOps Cloud with AI", International Journal of Science and Research (IJSR), 10(3), 2024-2027. https://dx.doi.org/10.21275/SR24724151733, https://www.ijsr.net/getabstract.php?paperid=SR24724151733
31. Naresh Lokiny, & Ranganath Nandanampati. (2020). DevSecOps: Integrating Security into DevOps with AI in Cloud. Journal of Scientific and Engineering Research, 7(10), 239–242. https://doi.org/10.5281/zenodo.13348695
32. Naresh Lokiny, & Pradip Reddy. (2021). Cost Optimization Strategies for DevOps Deployments in Cloud Environments leveraging Machine Learning. European Journal of Advances in Engineering and Technology, 8(3), 69–72. https://doi.org/10.5281/zenodo.13325845
33. Jayanth Para, 6G Internet Technology Cyber Threat Notification & Alert System, Vol. 52 No. 4 (2024): October-December 2024, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/170 , DOI: https://doi.org/10.46121/pspc.52.4.9
34. Jayanth Para, AI Based Cloud Computation Observational Method & Process, Vol. 51 No. 4 (2023): October-December 2023, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/171 , DOI: https://doi.org/10.46121/pspc.51.4.3
35. Kaleshwar Aryasomayajula, PREVENTING BIAS IN AI-BASED DECISION-MAKING: ANALYZING TECHNIQUES TO REMOVE UNFAIR PREJUDICE IN ALGORITHMIC OUTCOMES, Vol. 50 No. 2 (2022): April-June 2022, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/294
36. Kaleshwar Aryasomayajula, MODERN DEVELOPMENT PRACTICES: INVESTIGATIONS INTO SERVERLESS COMPUTING, LOW-CODE/NO-CODE PLATFORMS, AND DEVELOPER PRODUCTIVITY IN REMOTE ENVIRONMENTS, Vol. 50 No. 4 (2022): October-December 2022, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/364
37. Kaleshwar Aryasomayajula, Micro services: “A Comparative Analysis of Monolithic vs. Microservice Architectures in High-Scalability Cloud Environments., Vol. 51 No. 2 (2023): April-June 2023 , Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/374
38. Vikas Suresh Bagora, KERNEL-LEVEL PERFORMANCE OPTIMIZATION FOR CARRIER-GRADE BROADBAND AND HIGH-SPEED NETWORKING SYSTEMS, Vol. 47 No. 1 (2019), Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/359

39. Vikas Suresh Bagora, SCALABLE 128G FIBRE CHANNEL AND ETHERNET CONVERGENCE FOR NEXT-GENERATION DATA CENTER NETWORKS, Vol. 50 No. 4 (2022): October-December 2022, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/362
40. Vikas Suresh Bagora, SECURE EMBEDDED NETWORKING ARCHITECTURES USING TRUSTED EXECUTION ENVIRONMENTS IN BROADBAND GATEWAYS, Vol. 52 No. 2 (2024): April-June 2024, Power System Protection and Control, ISSN-1674-3415, https://pspac.info/index.php/dlbh/article/view/363
41. Santhosh Chitraju Gopal Varma, The Evolution of Cloud-Native Architectures: Exploring the Synergy between Kubernetes and Microservices, DOI: https://doi.org/10.63282/3050-9246.IJETCSIT-V1I4P104, Vol. 1 No. 4 (2020), https://ijetcsit.org/index.php/ijetcsit/article/view/108
42. Santhosh Chitraju Gopal Varma, Artificial Intelligence in Cloud Computing: Building Intelligent, Distributed, and Fault-Tolerant Systems, Vol. 3 No. 1 (2022), DOI: https://doi.org/10.63282/3050-9416.IJAIBDCMS-V3I1P105, https://ijaibdcms.org/index.php/ijaibdcms/article/view/78
43. Sunil Karmakar, AI driven Identity & Access Management with Oracle Cloud security, Vol. 30 No. 2 (2022): JOCAAA, https://eudoxuspress.com/index.php/pub/article/view/4118
44. Rakshitha Kiran P., Samitha Khaiyum, Aparna R. Palandye, Apoorva S. D., Arjun R., Akshaya S. A., Annapoorna S., Leveraging LLaMA3 and LangChain for Rapid AI Application Development, Vol. 20 No. 10s (2024), https://doi.org/10.52783/jes.5539, https://journal.esrgroups.org/jes/article/view/5539.
45. Kancharla Vijayalakshmi, Bethapudi Samuel Anand Andrews, Estimation of Cyproheptadine Hydrochloride and Tricholine Citrate Simultaneously in Syrup Dose by Applying Stability Indicating HPLC Methodology, December 31, 2020, Vol. 11, Issue 4, Pages 1813–1819, DOI: 10.26452/ijrps.v11iSPL4.4383, https://ijrps.com/v11si4/3683
46. Vijayalakshmi Kancharla ,Kumar Abbaraju V D N, Andrews B S A, Nagaraju Dasari, Nagu Korupolu, Melphan as Anticancer and New Insights of Drug as Parenteral Dosage form against SARS-CoV-2 by Using Development and Validation , November 18, 2020, Vol. 11, Issue 1, Pages 1437–1446
DOI: 10.26452/ijrps.v11iSPL1.3686, https://ijrps.com/v11si1/2911