Automated Postmortem Action Generation (PAG) using LLM

Priyaranjan Pattnayak[1], **Sanchari Chowdhuri**[1]
[1]Oracle America Inc.
Correspondence: priyaranjanpattnayak@gmail.com

***Abstract*—We present Postmortem Action Generation (PAG), a novel NLP framework for intelligently generating structured Corrective and Preventive Measures (CAPMs) from post-incident analyses in large-scale cloud systems. Unlike prior Root Cause Analysis (RCA) work that identifies why failures occur, PAG focuses on what to do next producing actionable, role-specific remediation steps. Our approach combines hybrid retrieval of similar incidents, persona-conditioned prompting, and LLM-based validation to ensure correctness, actionability, and role fit. Evaluated on 8,000+ real-world cloud incidents, PAG achieves strong performance with factual consistency (0.83), groundedness (85.3%), and expert-rated correctness (4.3/5), surpassing fine-tuned and rule-based baselines across automated and human evaluations. The architecture is lightweight, scalable, and production-ready frozen LLMs paired with retrieval and validation loops delivering reliable, large-scale post-incident decision support.**

I. Introduction

Cloud infrastructure outages are costly and complex. While postmortem analysis helps teams learn from failures, the formulation of CAPMs, the final step in RCA [1], [2], remains largely manual. CAPMs specify *how* to prevent recurrence, tailored to roles such as SREs, PMs, and architects, but their creation is slow, inconsistent, and hard to scale.

We propose a modular, prompt-based system using frozen LLMs with in-context learning (ICL) to generate structured CAPMs without fine-tuning. The pipeline retrieves similar incidents via hybrid semantic and metadata filtering, uses role-specific prompts [3], and applies persona-aware validators to ensure factual grounding, actionability, and role alignment. A frozen LLM here is adapted through prompting and retrieval only, without parameter updates. Evaluated on 8,000+ real-world cloud incidents, the approach shows strong results on automated and human evaluations while meeting cost and latency constraints.

We make the following contributions:

- **Problem Definition**: We introduce *Postmortem Action Generation (PAG)*, a structured generation task focused on producing role-specific corrective actions rather than diagnosing incidents alone.
- **Reliability-Oriented Framework**: A lightweight, prompt-driven system using frozen LLMs, hybrid retrieval, and persona-aware validation to improve the reliability of generated CAPMs.
- **Evaluation of Validation**: Empirical evidence on real and synthetic incidents showing that persona-aware validation improves factual correctness and actionability over generation-only baselines.
- **Production Evidence**: Deployment results demonstrating practical feasibility with low operational cost and latency.

By integrating incident narratives with validated, role-specific actions, our work positions LLMs as practical decision-support tools for post-incident workflows. Anonymous repository with data and all prompts for reproducibility.

II. Related Work

Automating incident management in cloud systems has gained significant attention, with most prior work focusing on *Root Cause Analysis (RCA)* identifying why incidents occur—rather than producing structured, actionable next steps. Existing approaches include supervised and prompt-based methods: FaultProfIT [4] uses hierarchy-guided contrastive learning for fault classification, while Ahmed et al. [1] and Chen et al. [2] leverage fine-tuned and prompt-based LLMs for root cause identification and mitigation suggestions. Similarly, RCACopilot [5] and Zhang et al. [6] demonstrate the effectiveness of LLMs and in-context learning for diagnostic reasoning. However, these approaches largely focus on cause identification and do not extend to generating structured Corrective and Preventive Measures (CAPMs)

Work on instruction/action generation [7], [8], [9] and persona-aware prompting [10] targets low-risk,multi-step or role-based outputs in weakly grounded settings. In contrast, cloud postmortem action generation is high-stakes, demanding factual, role-specific, and operationally feasible outputs.

Prior work shows key gaps: weak support for concrete post-incident actions, little use of historical grounding, no role-aware validation, and limited evaluation of actionability or correctness.

Our work addresses these via *Postmortem Action Generation (PAG)*, a retrieval-augmented, prompt-driven framework that uses frozen LLMs to generate and validate role-specific, incident-grounded CAPMs.

A. Our Contributions

Our key contributions are:

- **New Task:** We introduce *Postmortem Action Generation (PAG)*, an NLP task for generating structured, role-specific *Corrective and Preventive Measures (CAPMs)* from cloud incidents—focusing on *what to do next* rather than *what went wrong*.

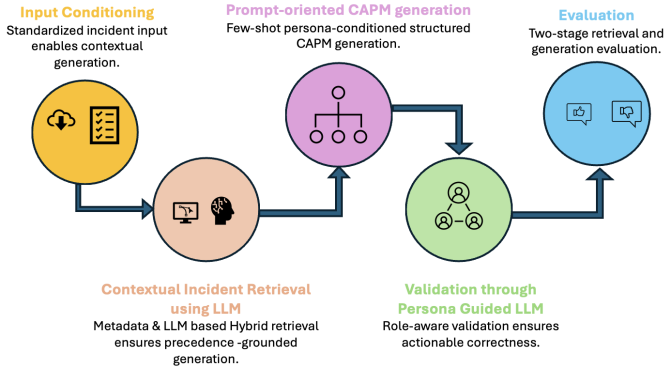

Fig. 1. High Level CAPM Generation Flow

- **Reliability-Oriented Framework:** A lightweight, frozen-LLM pipeline combining retrieval, role-conditioned prompting, and persona-aware validation to improve the reliability of generated actions.
- **Persona-Aware Validation:** We develop persona-aware LLM validators that ensure correctness, actionability, and role fit, improving reliability and trust.
- **Empirical Evaluation:** On 8,000+ real (and synthetic) incidents, PAG achieves improved factual consistency (0.83) and expert-rated correctness (4.3/5), over generation-only baselines.
- **PAG Synthetic Dataset:** We release a synthetic, privacy-safe dataset of 950 incident records mirroring production distributions for open benchmarking of PAG systems.

III. Methodology

We formulate CAPM generation as a structured text generation task: given a postmortem incident summary and relevant historical context, generate actionable, persona-specific remediation steps. This high-stakes task requires outputs that are factually grounded, technically valid, and appropriate for diverse operational roles.

Our system is a modular pipeline built on frozen LLMs with in-context learning (ICL), avoiding fine-tuning for fast adaptation and low maintenance. As shown in Figure 1, the pipeline consists of input conditioning, contextual retrieval, prompt-based CAPM generation, and persona-guided validation, inspired by RAG [11] and LLM self-feedback [12].

Algorithm 1: Postmortem Action Generation (PAG)

1) Normalize incident input (summary + metadata)
2) Retrieve candidate incidents via metadata filtering
3) Rank candidates by semantic similarity; select top-k
4) Construct persona-conditioned prompt with retrieved CAPMs
5) Generate candidate CAPMs using frozen LLM
6) Validate each CAPM for correctness, actionability, and role fit
7) Discard CAPMs rated *No* or *Partial*

A. Input Conditioning

Each task starts with a semi-structured incident input—free-text summary, metadata (e.g., service, timestamp, severity etc).

TABLE I
Retrieval Configuration

Parameter	Value
Retriever model	SBERT MiniLM
Top-k	5
Similarity metric	Cosine
Metadata match	Service, category

We normalize this to create a consistent prompt; CAPMs are generated from the narrative and similar past incidents, without requiring root cause input.

B. Contextual Retrieval of Relevant Incidents

To ground CAPMs in operational history, we retrieve similar incidents from a curated postmortem repository using a hybrid strategy:

- **Metadata Filtering:** Candidates are filtered by service, team, and category.
- **Semantic Ranking:** Incident summaries are encoded using SBERT (all-MiniLM-L6-v2) [13] and ranked by cosine similarity.

The top-k retrieved incidents and their CAPMs are used as in-context references, improving factuality and plausibility.

C. Prompt-Based CAPM Generation

CAPM generation is framed as a few-shot structured task. The LLM receives the incident context, retrieved CAPM examples as in-context demonstrations, and a persona-specific prompt defining structure and granularity (see the anonymized repository for dataset and full prompt templates). For example: *As an SRE, list 2–3 system-level mitigations to prevent recurrence*. We use frozen Cohere-command-R-plus with low temperature ($T = 0.2$) and top-p=0.75 to favor factual consistency without fine-tuning. Persona conditioning mainly constrains scope and granularity of CAPMs: **SRE** prompts bias toward system/network reliability changes (runbooks, monitoring, rollout/traffic controls), while **SDE** prompts target code-level fixes (defensive checks, tests, refactors) and **PM** prompts focus on process/product actions (customer comms, prioritization, requirements/guardrails). The same persona is applied in validation, which explicitly checks technical correctness, actionability, and role fit before accepting CAPMs. Complete prompt templates for generation and validation are provided in the anonymized repository to preserve page limits and confidentiality.

D. Persona-Guided Validation and Reliability Checks

Given the high-stakes nature of CAPMs, we apply a validation layer using frozen LLMs as role-based evaluators. Each generated CAPM is assessed along three dimensions: (1) **Technical Correctness** is the recommendation valid and sound? (2) **Actionability** can it be practically implemented within the service context? (3) **Role Fit** is it appropriate for the specified persona. We enforce operational fit via persona conditioned prompts: SRE outputs are constrained to concrete

system-level reliability mitigations, while PM outputs are constrained to process/product CAPMs.

We support four personas: Site Reliability Engineer (SRE), Product Manager (PM), Software Development Engineer (SDE) and Network Architect—each with zero or few-shot validation prompts. Validators return structured judgments (*Yes*, *Partial*, *No*) following a rubric. CAPMs rated *No* or *Partial* for correctness or actionability are discarded.

Validator outcomes are logged to monitor prompt degradation and detect drift in production.

Validation-Driven Regeneration: We additionally study a simple validation-driven regeneration strategy, where CAPMs that fail validation are regenerated once using validator feedback. This is not intended as a new algorithm, but as a practical extension to improve pass rates under fixed cost and latency constraints.

E. Experiment Setup and Data

Dataset Details and Bias: Our dataset includes over 8,000 resolved cloud incidents collected over 36 months. These incidents come from a production [public cloud] environment supporting multiple service categories (e.g., compute, database, networking, storage), so the reported retrieval and CAPM metrics reflect operational conditions typical of that setting. Incident summaries are cleaned to remove PII and logs, then condensed via prompt-based summarization. We use Cohere-command-R-plus, with low temperature ($T = 0.2$), top-p=0.75 for summarization and CAPM generation with frozen inference. Dataset includes structured incident fields (summary, description, severity, timestamps, root cause, etc.) with free-text PII removed via compliance-conditioned prompts. Human experts (senior SDE/SREs, principal incident managers) were selected for cross-functional experience, incident exposure, and tenure. We analyzed incident distributions across service teams and root cause categories to assess potential dataset skews.

Due to confidentiality, raw data cannot be released. We therefore provide methodological details and release a sanitized synthetic dataset of 950 incidents[1] that mirrors the schema and distributions of production data (see Table IV) , ensuring that evaluations on it closely approximate real-world behavior. This dataset provides a controlled yet representative benchmark for evaluating CAPM generation models across varied root causes and service domains.

Prompt Robustness and Adaptation:

To address prompt and validator drift as incidents and team practices evolve, we (i) monitor persona-conditioned validators over a 30-day window, flagging prompts with $>10\%$ "No/Partial" rates (ii) log all prompt versions with validator outcomes to track performance. This drift detection, versioning, and expert review ensures prompts remain robust and aligned with operational changes.

[1]https://github.com/ppattnayak/PAG_DB

IV. Evaluation

We evaluate PAG along two axes: (i) the quality of contextual retrieval used for grounding CAPM generation, and (ii) the correctness, actionability, and role alignment of generated CAPMs. Given the open-ended nature of postmortem actions, evaluation emphasizes factual grounding and practical utility rather than exact match to a single reference.

A. Contextual Retrieval Evaluation

We assess retrieval quality by encoding incident summaries using SBERT and retrieving the top-k precedent incidents via cosine similarity. Two domain experts annotated the top-5 retrieved incidents for 420 queries based on similarity in failure type, mitigation strategy, and causal structure, using a 3-point relevance scale (Not, Partial, High).

We report Precision@5, Recall@5, and NDCG@5 to measure retrieval accuracy, coverage, and ranking quality. Inter-annotator agreement was measured using Cohen's κ.

B. CAPM Generation Evaluation

Generated CAPMs are evaluated against historical actions with an emphasis on grounding and action quality rather than exact textual overlap.

a) Automated Metrics: We use the following metrics:

Primary Metrics.

- **Factual Consistency (FCS) (0–1)** [14]: Verifiability of CAPM statements against incident context and retrieved precedents.
- **Groundedness Rate (0–1)** [15]: Fraction of CAPMs supported by incident evidence or retrieved context.
- **Adoption Likelihood (AL) (1–5)**: Expert-estimated likelihood that a CAPM would be implemented in practice.

We additionally report **Validation Acceptance Rate (Pass@Val)**, defined as the fraction of generated CAPMs that pass persona-guided validation. Pass@Val is not a standalone accuracy metric, but captures an operational notion of reliability by measuring whether CAPMs survive expert-aligned validation, complementing standard lexical and semantic generation metrics.

Supporting Metrics.

- **BERTScore**: Semantic alignment with historical CAPMs.
- **BLEU-4**: Structural and lexical consistency.
- **Answerability (1–5)**: Completeness and plausibility of CAPMs.
- **Causal Coverage (CCov) (0–1)**: Coverage of key root causes addressed by CAPMs.

C. Human Evaluation

We conduct human evaluation on a subset of 300 generated CAPMs. Two domain experts reviewed each CAPM alongside the incident summary and retrieved precedents, rating the following dimensions on a 5-point Likert scale:

- **Correctness**: Is the action technically valid in context?
- **Faithfulness**: Is the action consistent with the incident description and retrieved context?

- **Novelty**: Does the action go beyond naive reuse?
- **Adoption Likelihood (AL)**: Likelihood that the CAPM would be adopted in practice.

V. Results

We report results across three core dimensions of our PAG framework: (1) contextual retrieval of precedent incidents, (2) quality of CAPM generation, and (3) effectiveness of the overall pipeline using ablation studies.

A. Contextual Retrieval Evaluation

Our SBERT-based retriever achieved high performance across 420 queries: Precision@5 0.78, Recall@5 0.74, and NDCG@5 0.81. Scores were slightly lower for rare or service-specific incidents, while inter-rater reliability was strong (Cohen's $\kappa = 0.72$). These results confirm the retriever's effectiveness in surfacing diverse, contextually relevant precedents, providing a solid factual foundation for CAPM generation.

B. CAPM Generation Evaluation

The CAPM generator aligns closely with historical actions, achieving BERTScore 0.84 and BLEU-4 27.6 (Table II). Groundedness 85.3% and FCS 0.83 show strong factual support, while Causal Coverage 0.74 and Answerability 4.2 indicate that generated CAPMs capture root causes and provide actionable, reliable recommendations for postmortem reviews.

In addition to quality metrics, we report **Validation Acceptance Rate (Pass@Val)**, defined as the fraction of generated CAPMs that pass persona-guided validation without being discarded. On production data, PAG achieves a Pass@Val of **79.1%**, indicating that the majority of generated actions meet correctness and actionability thresholds without requiring intervention.

1) Comparison with Fine-tuned Baseline: We compared our frozen LLM + retrieval pipeline with a fine-tuned Mixtral-8x22B trained on over 8,000 incident→CAPM pairs. As Table II shows, the fine-tuned model had slightly higher BLEU-4 (28.0 vs. 27.6) for lexical overlap and smoother phrasing but lagged in factual grounding (Groundedness 80.2%, FCS 0.78) and causal completeness. This indicates that contextual retrieval and persona conditioning offer better domain generalization and reliability than fine-tuning alone.

2) Distribution Across Services: Service-wise evaluation (Table IV) shows CAPM quality is consistent across domains. Compute & Infrastructure and Storage Services achieve the highest FCS ≈ 0.85) and answerability (4.4), while smaller or specialized categories like Marketplace and Key Management have lower coverage and groundedness due to limited precedent data.

3) Comparison with Rule-Based Baseline: A simple rule-based CAPM generator retrieves the top-3 similar incidents using TF–IDF similarity and fills a fixed summary template. While this deterministic approach ensures factual grounding (FCS 0.62), it yields limited contextual or causal coverage (CCov 0.45), confirming that retrieval and templating alone cannot produce adaptive, role-aligned guidance.

TABLE II
Comparison with Fine-tuned, Rule-based, and Zero-shot Baselines

Metric	Frozen + Retrieval	FT Mixtral 8x22B	Rule-based Heuristic	LLaMA-3-8B (0-shot)
BLEU-4	27.6	**28.0**	18.3	24.9
BERTScore	**0.84**	0.82	0.65	0.78
Grounded (%)	85.3	80.2	**87.2**	74.6
FCS	**0.83**	0.78	0.62	0.72
CCov	**0.74**	0.68	0.45	0.63
Ans. (1–5)	**4.2**	3.8	3.0	3.1
Pass@Val	0.79	0.65	**0.82**	0.58

4) Zero-shot Baseline.: We tested LLaMA-3-8B-Instruct in a zero-shot setting (prompt-only, no retrieval or persona context). While fluent, its outputs were less grounded (*FCS*=0.72, *Grounded*=74.6%) and impractical (*AL*=3.1), reaffirming the value of PAG's retrieval and validation layers.

C. Human Evaluation

Two domain experts rated 300 CAPMs on a 1–5 scale for Correctness, Novelty, Faithfulness, and Adoption Likelihood (AL), with averages of 4.3, 4.1, 4.0, and 4.0, respectively. The 300 samples were stratified across nine services (30-35 each) and four personas ($\sim$ 75 CAPMs per role) for balance. Estimated 95% CI for mean correctness was ±0.1, indicating stable results across domains and roles. Inter-annotator agreement was substantial (Cohen's κ 0.66–0.72, Weighted κ 0.70–0.78). Automated metrics correlated strongly with human judgments (FCS → AL: $\rho = 0.61$, BERT Score→Faithfulness: $\rho = 0.61$; CCov → Correctness: $\rho = 0.65$), supporting the system's ability to generate technically valid, context-specific, and non-redundant CAPMs.

D. Ablation Studies

We ablated retrieval, persona prompting, and validation modules, evaluating each with automatic metrics (BERTScore, Groundedness, FCS, CCov) and human ratings (Correctness, AL). Fifty CAPMs per setup were reviewed by two experts (SRE and incident manager) with strong agreement ($\kappa = 0.76$). Retrieval boosted factual and causal grounding, persona prompting improved contextual relevance, and validation prevented unsafe or hallucinated outputs (Table III).

Without Retrieval Context. Removing retrieved precedents sharply reduced performance: BERTScore 0.75, FCS 0.70, CCov 0.59, groundedness 68.9%, with human scores dropping (Correctness 3.6, AL 3.2), confirming lower factual reliability and actionability.

Without Persona Prompting. Generic prompts weakened role alignment, lowering BERTScore to 0.78, CCov to 0.64, and groundedness to 72.4%. Correctness and AL decreased to 3.7 and 3.4, underscoring the value of persona conditioning for contextual fit.

Without LLM Validator.

Skipping validation sharply reduced system reliability, with Pass@Val dropping from **79%** to **46%**, alongside increased

TABLE III
ABLATION STUDY. BEST RESULTS ARE SHOWN IN BOLD.

Metric	PAG (Ours)	w/o Retrieval Context	w/o Persona Prompting	w/o LLM Validator
BERTScore	**0.84**	0.75	0.78	0.79
Groundedness (%)	**85.3**	68.9	72.4	69.1
FCS	**0.83**	0.70	0.75	0.68
CCov	**0.74**	0.59	0.64	0.61
Correctness (1–5)	**4.3**	3.6	3.7	3.4
AL (1–5)	**4.0**	3.2	3.4	3.3
Pass@Val	**0.79**	0.61	0.66	-

hallucinations and factual drift (FCS fell to 0.68, groundedness to 69.1%, CCov to 0.61, human scores to 3.4 (Correctness) and 3.3 (AL).A single validation-driven regeneration step improves acceptance rates and marginally improves quality, with minimal additional latency.

E. Experiment on Released Synthetic Dataset

To assess reproducibility without private data, we evaluate PAG on a released synthetic dataset (950 incidents) modeled after the production schema. As shown in Figure 2, results closely track production metrics across all six dimensions, indicating that the synthetic corpus preserves the same performance characteristics.

Our model achieves comparable semantic fidelity (**BERTScore = 0.87** vs. 0.84), lexical overlap (**BLEU-4 = 28.5**), and groundedness (**80%** vs. 85%) with only minor variations. Factual Consistency (**0.80**) and Causal Coverage (**0.76**) remain aligned with real-world data, and Answerability (**4/5**) reflects realistic task completeness.

The close correspondence between synthetic and production results demonstrates PAG's robustness to dataset variation and establishes the released dataset as a reliable benchmark for reproducible post-incident action generation.

VI. ERROR ANALYSIS

We evaluated 300 CAPM generations across services and personas, flagging errors via validator disagreement and expert scores ($\leq$3). Twenty-five cases (8.3%) failed validation, grouped into four categories as shown in Table VI.

Most errors involved **incomplete representations (40%)** from vague or underspecified summaries, reducing causal coverage and adoption likelihood. Other issues included **hallucinated actions (32%)**, typically fluent but unsupported suggestions, and **false acceptances (16%)** or **false rejections (12%)** from validator misjudgment. Validator misalignment and persona drift was rare ($< 2\%$) and caused only minor factual inconsistencies, with no safety-critical impact. Most incomplete representations were associated with low-quality retrieval context, while hallucinated actions were disproportionately filtered by persona-guided validation, reinforcing the importance of both components.

Human-in-the-loop validation, persona conditioning, and prompt updates reduce errors by improving grounding and consistency. Refer Table V for Extended error analysis. Most incomplete outputs stem from poor retrieval context, while persona-guided validation effectively filters hallucinated actions, highlighting the importance of both mechanisms.

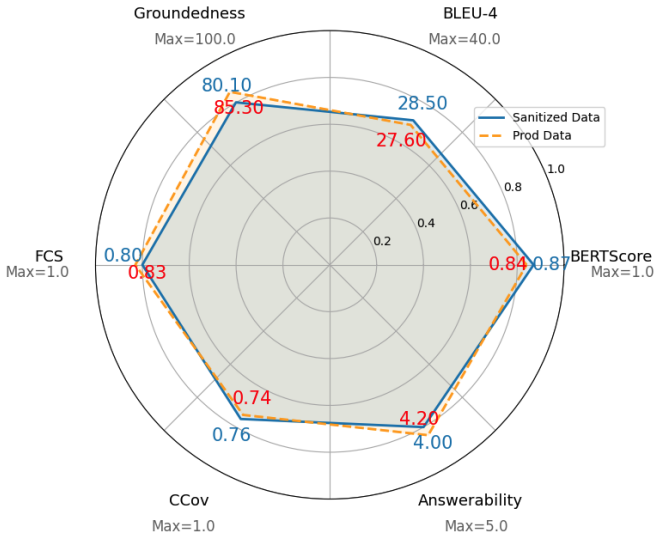

Fig. 2. Comparison of production and synthetic dataset performance across six evaluation dimensions. Synthetic data tracks production behavior, supporting reproducibility

VII. CONCLUSION

We present **Postmortem Action Generation (PAG)**, a lightweight frozen-LLM framework that uses retrieval, persona prompting, and validation to generate grounded, role-specific CAPMs from incident reports without fine-tuning.

Evaluated on over 8,000 real-world incidents, PAG achieves high factual consistency and expert-rated correctness, outperforming fine-tuned and rule-based baselines. Results on released synthetic dataset closely mirror real-world outcomes, demonstrating robustness and enabling reproducible benchmarking beyond proprietary data. PAG thus offers a practical, reliable, and reproducible approach for scalable post-incident decision support in cloud operations. PAG has been piloted in an enterprise incident management workflow, generating CAPMs for high-severity incidents with low latency and stable validation behavior, demonstrating practical feasibility beyond offline evaluation.

VIII. LIMITATIONS

While our framework is practical and scalable, it has practical constraints. Persona prompts can drift as formats change; the system is currently English-only and would need new retrieval indices/role templates for other domains or languages. Validation adds minor latency and may misjudge underspecified edge cases; future work targets adaptive prompts/validation and multilingual generalization

TABLE IV
CAPM GENERATION PERFORMANCE ACROSS SERVICE CATEGORIES. PASS@VAL DENOTES THE PERCENTAGE OF GENERATED CAPMS THAT PASS PERSONA-GUIDED VALIDATION, REFLECTING END-TO-END RELIABILITY RATHER THAN SURFACE SIMILARITY.

Service Category	**Incidents**	**BLEU-4**	**BERTScore**	**Grounded (%)**	**FCS**	**CCov**	**Ans.**	**Pass@Val (%)**	**Synth.**
Compute & Infrastructure	1404	28.7	0.86	87.0	0.85	0.77	4.4	80.4	105
Database Services	895	27.9	0.84	85.6	0.83	0.75	4.2	79.5	108
AI & ML Services	1105	27.4	0.83	84.3	0.82	0.73	4.1	76.8	99
API & Identity Services	997	27.6	0.84	85.0	0.83	0.74	4.2	78.9	118
Networking Services	851	27.1	0.83	84.1	0.82	0.73	4.1	76.2	105
Storage Services	1101	28.2	0.85	86.2	0.84	0.76	4.3	80.0	102
Logging	903	27.0	0.82	83.6	0.81	0.72	4.0	78.9	114
Artifacts & Key Mgmt	499	26.5	0.81	82.8	0.80	0.70	3.9	75.5	87
Marketplace	445	25.9	0.80	81.2	0.78	0.68	3.8	74.7	112
Overall (Weighted Avg.)	**8200**	**27.6**	**0.84**	**85.3**	**0.83**	**0.74**	**4.2**	**79.0**	**950**

TABLE V
EXTENDED ERROR EXAMPLES

Error Category	**Incident Summary**	**Erroneous CAPM**
Incomplete Representation	Authentication latency increased for users in region A.	Improve retry logic for third-party calls.
Incomplete Representation	Health check flaps occurred during canary rollout in zone A.	Implement auto-scaling based on disk usage.
Hallucinated CAPMs	Billing reports failed due to missing CSV upload.	Auto-label bugs using generative AI model.
Hallucinated CAPMs	Third-party DNS provider outage affected name resolution.	Implement blockchain-based audit for deployments.
Persona Misalignment	Product dashboard failed to load metrics for executives.	Provision dedicated node pool for SRE testing.
Persona Misalignment	Monitoring alerts failed to trigger during API downtime.	Add TLS cipher compatibility check in CI.
Validator Misjudgment	Storage latency spike observed for write-heavy workload.	Auto-retry failed alerts with increasing backoff.

TABLE VI
BREAKDOWN OF 25 OBSERVED ERRORS IN CAPM GENERATION.

Error Type	**Incident Count**	**% of Errors**
Incomplete Rep	10	40%
Hallucinated Actions	8	32%
False Acceptances	4	16%
False Rejections	3	12%

REFERENCES

[1] T. Ahmed, S. Ghosh, C. Bansal, T. Zimmermann, X. Zhang, and S. Rajmohan, "Recommending root-cause and mitigation steps for cloud incidents using large language models," in *Proceedings of the 2023 IEEE/ACM International Conference on Software Engineering (ICSE)*, 2023, pp. 1737–1749.

[2] Y. Chen, H. Xie, M. Ma, Y. Kang, X. Gao, L. Shi, Y. Cao, X. Gao, H. Fan, and M. Wen, "Automatic root cause analysis via large language models for cloud incidents," in *Proceedings of the 2024 European Conference on Computer Systems*, 2024, pp. 674–688.

[3] L. Reynolds and K. McDonell, "Prompt programming for large language models: Beyond the few-shot paradigm," in *arXiv preprint arXiv:2102.07350*, 2021.

[4] J. Huang, J. Liu, Z. Chen, Z. Jiang, Y. Li, J. Gu, C. Feng, Z. Yang, Y. Yang, and M. R. Lyu, "Faultprofit: Hierarchical fault profiling of incident tickets in large-scale cloud systems," in *Proceedings of the 46th International Conference on Software Engineering: Software Engineering in Practice*, 2024, pp. 392–404.

[5] Y. Chen, H. Xie, M. Ma, Y. Kang, X. Gao, L. Shi, Y. Cao, X. Gao, H. Fan, M. Wen *et al.*, "Automatic root cause analysis via large language models for cloud incidents," in *Proceedings of the Nineteenth European Conference on Computer Systems*, 2024, pp. 674–688.

[6] X. Zhang, S. Ghosh, C. Bansal, R. Wang, M. Ma, Y. Kang, and S. Rajmohan, "Automated root causing of cloud incidents using in-context learning with gpt-4," 2024. [Online]. Available: https://arxiv.org/abs/2401.13810

[7] S. Yuan, J. Chen, Z. Fu, X. Ge, S. Shah, C. R. Jankowski, Y. Xiao, and D. Yang, "Distilling script knowledge from large language models for constrained language planning," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2023. [Online]. Available: https://aclanthology.org/2023.acl-long.236/

[8] Z. Shi, Y. Xu, M. Fang, and L. Chen, "Self-imitation learning for action generation in text-based games," in *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2023)*. Association for Computational Linguistics, 2023. [Online]. Available: https://aclanthology.org/2023.eacl-main.50/

[9] Y. Shen, X. Li, M. Wang, and L. Liu, "Task-oriented action generation with retrieval-conditioned language models," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

[10] J. Lee, M. Oh, and D. Lee, "P5: Plug-and-play persona prompting for personalized response selection," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP 2023)*. Association for Computational Linguistics, 2023. [Online]. Available: https://aclanthology.org/2023.emnlp-main.1031/

[11] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, A. Fan, V. Chaudhary, T. Rocktäschel, M.-F. Moens *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," in *Advances in Neural Information Processing Systems*, 2020, pp. 9459–9474.

[12] A. Madaan, X. Lin, L. Zettlemoyer, and A. Gupta, "Self-refine: Iterative refinement with self-feedback," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

[13] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 Conference on EMNLP*, 2019.

[14] O. Honovich, L. Choshen, U. Katz, and et al., "True: Re-evaluating factual consistency evaluation," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, 2022, pp. 3902–3920.

[15] P. Manakul, A. Liusie, and M. J. F. Gales, "Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, 2023.