Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

Behavioral Stability of Quantized Large Language Models Under Prompt Drift: The Resilio Evaluation Framework

Vedant Barhate
School of Computing
MIT ADT University
Pune, India

Yash Pandey
School of Computing
MIT ADT University
Pune, India

Achyut Yesare
School of Computing
MIT ADT University
Pune, India

Vibhor Dev
School of Computing
MIT ADT University
Pune, India

***Abstract*—Large Language Models (LLMs) are increasingly deployed in resource-constrained environments using quantization techniques such as 8-bit and 4-bit precision, which reduce memory footprint and inference costs. While these models perform well on clean benchmarks, real-world deployment frequently encounters "Prompt Drift"—typographical errors, informal phrasing, and structural degradation.**

This paper introduces *Resilio*, a systematic evaluation framework investigating the interaction between model quantization and five progressive levels of prompt quality degradation. We evaluate Llama 3.1 8B, Mistral 7B, and Phi-3 Mini using two novel metrics: the Task Performance Score (TPS) and the Behavioral Stability Score (BSS).

Our study reveals a "Quantization Amplification" effect, where 4-bit models exhibit disproportionately higher sensitivity to noise compared to FP16 baselines, particularly in reasoning-intensive tasks.

***Index Terms*—Quantization, Large Language Models, Prompt Drift, Behavioral Stability, BSS, TPS, Robustness, Deployment**

I. Introduction

The deployment of Large Language Models (LLMs) has shifted from centralized, high-performance data centers to resource-constrained edge environments, including mobile hardware and local workstations. This transition is driven by the demand for reduced latency, enhanced data privacy, and lower operational costs. To enable this scaling, post-training quantization (PTQ) has become the industry-standard optimization strategy, compressing model weights from 16-bit floating point (FP16) to 8-bit (INT8) or 4-bit (INT4) precision. While these methods achieve substantial memory compression with minimal impact on clean-input accuracy, they introduce a poorly understood layer of behavioral risk.

A. The Phenomenon of Prompt Drift

Standard evaluation protocols assess LLMs using "ideal" benchmarks characterized by curated, well-structured syntax. However, production environments expose models to a continuous spectrum of input degradation—a phenomenon we formalize as *Prompt Drift*. Prompt Drift reflects the naturally occurring noise in human-AI interaction, spanning five escalating severity levels:

- **L1 – Typographical Noise:** Introduction of character-level errors such as misspellings, adjacency mistakes, and letter transpositions while preserving overall structure and meaning.
- **L2 – Formatting Loss:** Removal of capitalization, punctuation, and standard formatting cues, leading to reduced structural clarity without altering core semantics.
- **L3 – Linguistic Informality:** Incorporation of slang, abbreviations, and colloquial expressions (e.g., "u", "pls", "bc"), introducing deviations from formal language patterns.
- **L4 – Structural Degradation:** Breakdown of grammatical structure through word merging, spacing errors, and syntactic inconsistencies that affect sentence readability.
- **L5 – Semantic Fragmentation:** Severe degradation characterized by incomplete phrases, truncated context, and fragmented instructions, leading to partial or ambiguous semantic representation.

B. The Critical Evaluation Gap

A fundamental gap exists in current AI research: the interaction between model quantization and prompt drift remains largely unmeasured. Traditional benchmarks fail to capture how a reduction in internal numerical precision affects a model's capacity to parse and reason over noisy data. There is a significant concern that quantization does not merely decrease absolute performance but actively amplifies a model's sensitivity to input degradation. This amplification can lead to *silent failures*—where a quantized model maintains high expressed confidence while providing factually incorrect or logically incoherent responses.

The *Resilio* framework is designed to address this gap. By moving beyond binary accuracy and implementing multi-dimensional stability scoring, we aim to establish empirical safety thresholds for the deployment of quantized models in real-world, noisy environments.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

II. Related Work

The pursuit of deploying Large Language Models (LLMs) on edge devices has catalyzed research into two previously distinct domains: computational optimization through quantization and the linguistic robustness of models under perturbation. The *Resilio* framework operates at the intersection of these fields.

A. LLM Quantization for Efficient Deployment

The necessity for on-device LLM execution has led to the rapid maturation of Post-Training Quantization (PTQ) techniques. As noted by Dettmers *et al.* [1], 8-bit matrix multiplication allows for significant memory reduction with negligible performance loss on curated benchmarks. The introduction of 4-bit quantization further pushes these boundaries, enabling models like Phi-3, Llama 3.1, and Mistral 7B to operate within the strict memory constraints of mobile hardware [2], [3], [9].

However, while prior analyses provide extensive evaluation of the resource efficiency of INT4 versus INT8 representations [1], [9], these studies typically rely on "clean" input distributions. Recent work suggests that such extreme compression levels may introduce hidden instabilities or alter model safety alignment—a phenomenon sometimes termed *quantization-based vulnerability* [9].

B. Robustness to Input Perturbations and Prompt Drift

The sensitivity of LLMs to surface-level changes in input text is a well-documented challenge. Research into typographical noise [11] and naturally occurring errors in training data [8] confirms that even slight character-level deviations can produce significant variance in model output. Frameworks such as PromptBench have begun to standardize the measurement of model resilience against adversarial attacks [5].

Furthermore, studies on "noisy instructions" indicate that the quality of the prompt is a primary determinant of zero-shot generalization [10]. Despite these insights, existing robustness studies primarily utilize full-precision (FP16 or FP32) models, leaving the stability of quantized architectures under similar *Prompt Drift* largely unmeasured [4].

C. Behavioral Stability and Runtime Monitoring

As AI agents move into production, *behavioral stability* has emerged as a more critical metric than simple accuracy. This stability can be defined as the consistency of model behavior across intra-prompt variations [6]. In real-world deployments, *data drift*—the shift in the distribution of user inputs over time—presents a constant risk to model reliability [7].

While the research community has proposed various runtime monitoring strategies to detect behavioral shifts in AI systems [12], there remains a lack of unified metrics that measure stability relative to a clean baseline. The interaction between acceleration techniques and these hidden instabilities remains an open area of inquiry [9].

D. Identification of Research Gaps

A systematic review of the literature reveals a critical tripartite gap:

- **Isolation of Factors:** Quantization is typically evaluated on clean inputs, while robustness is tested on full-precision models. The interaction between bit-precision and input drift is largely omitted.
- **Absence of Graduated Drift Frameworks:** Most studies rely on binary noise conditions (clean vs. noisy) rather than a structured, multi-level taxonomy (L1–L5) as proposed in this work.
- **Metric Limitations:** Standard accuracy metrics fail to capture *silent failures*, where a model maintains high confidence despite producing logically inconsistent or incorrect outputs.

By introducing the *Task Performance Score (TPS)* and *Behavioral Stability Score (BSS)*, the *Resilio* framework provides a multi-factor analysis of task-specific stability patterns under quantized prompt drift.

III. The Resilio Methodology

The architecture of the *Resilio* framework is centered on a modular, reproducible pipeline that bridges the gap between synthetic benchmarks and the stochastic nature of production-level input noise. The methodology is executed through three distinct phases: custom dataset curation, object-oriented drift simulation, and semantic validation.

A. Dataset Architecture

To facilitate a granular analysis of model stability, we curated a specialized dataset comprising 40 base prompts. While these prompts are strategically categorized to mirror established linguistic challenges, they were custom-developed to target specific behavioral boundaries in Large Language Models:

- **Reasoning:** 10 prompts engineered to evaluate multi-step mathematical logic and sequential deduction.
- **Logical Inference:** 10 prompts structured to assess complex deductive reasoning and syllogistic consistency.
- **Question Answering:** 10 prompts focused on the precision of factual retrieval and context-window utilization.
- **Sentiment Classification:** 10 prompts designed to measure the robustness of pattern recognition under lexical noise.

B. OOP-based Drift Generation Framework

The technical core of *Resilio* is an Object-Oriented Programming (OOP) based drift generator. This modular engine allows for the application of graduated, additive transformations that simulate realistic user-driven degradation. Each base prompt was subjected to 16 unique variations to ensure statistical diversity in the results.

The framework formalizes *Prompt Drift* into a five-tier severity taxonomy:

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

- **L1 (Character-level Perturbation):** Simulates typographical errors via keyboard adjacency mapping and character transpositions.
- **L2 (Form Degradation):** Implements systematic removal of casing, standard punctuation, and formal whitespace.
- **L3 (Linguistic Shift):** Replaces formal lexical units with slang, non-standard abbreviations, and colloquial expressions.
- **L4 (Grammatical Breakdown):** Induces word merging, omission of essential function words, and overall syntactic collapse.
- **L5 (Structural Truncation):** Combines aggressive informality with content reduction, forcing the model to infer intent from fragmented inputs.

C. Mathematical Validation of Realism

To ensure that the generated drift represents realistic "noisy" input rather than incoherent gibberish, we mathematically validate the quality of perturbations using the `all-MiniLM-L6-v2` SentenceTransformer model. We compute the cosine similarity between the clean baseline (L0) and each drifted variant.

Formally, for embedding vectors $\mathbf{u}$ (clean prompt) and $\mathbf{v}$ (drifted prompt), cosine similarity is defined as:

$$\text{CosineSimilarity}(\mathbf{u}, \mathbf{v}) = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \, \|\mathbf{v}\|} \tag{1}$$

The generated variants are required to satisfy strict semantic similarity ranges:

- **L1:** 93.2% similarity (high surface-level fidelity)
- **L2:** 88.1% similarity (loss of structural metadata)
- **L3:** 83.4% similarity (lexical divergence)
- **L4:** 77.8% similarity (syntactic fragmentation)
- **L5:** 71.5% similarity (boundary of semantic coherence)

This calibration confirms that the *Resilio* framework provides a rigorously controlled and semantically grounded stress test for evaluating quantized language model robustness.

IV. Metric Formulation (TPS & BSS)

The core analytical contribution of the *Resilio* framework is its departure from simplistic accuracy metrics toward a multidimensional assessment of model reliability. By decoupling peak capability from structural consistency, we introduce a dual-metric system designed to capture the granular degradation of quantized architectures.

A. Task Performance Score (TPS): The 5-Factor Behavioral Matrix

Traditional benchmarking often fails to identify "silent failures"—instances where a model maintains linguistic fluency while undergoing logical or factual collapse. To quantify these effects, each response is evaluated across a weighted five-dimensional behavioral vector:

- **Factual Correctness (FC):** Cross-referencing the model's terminal output against ground-truth metadata.
- **Logical Correctness (LC):** Verification of deductive validity and mathematical accuracy of intermediate steps.
- **Reasoning Coherence (RC):** Assessment of structural integrity and sequential clarity of the explanation.
- **Sentiment Correctness (SC):** Evaluation of label accuracy for pattern-matching and classification tasks.
- **Manner of Language (ML):** A calibration metric measuring whether linguistic confidence aligns with actual output accuracy.

The Task Performance Score (TPS) is computed as:

$$TPS = \sum_{i=1}^{5} (w_i \cdot \text{Score}_i) \tag{2}$$

where w_i denotes task-specific weight coefficients. These weights are dynamically adjusted; for example, Logical Correctness (LC) is prioritized in reasoning tasks, while Sentiment Correctness (SC) is emphasized in classification settings.

B. Behavioral Stability Score (BSS)

While TPS captures absolute performance, the *Behavioral Stability Score (BSS)* measures resilience to input degradation. By normalizing degraded performance against a clean baseline, BSS provides a relative stability index independent of model scale.

For a given drift level d, BSS is defined as:

$$BSS_d = \frac{\overline{TPS}_d}{TPS_0} \tag{3}$$

where $\overline{TPS}_d$ represents the mean Task Performance Score across all variants at drift level d, and TPS_0 denotes performance on the clean baseline (L0).

C. Stability Threshold and Behavioral Collapse

A central empirical finding of this work is the identification of a stability threshold defined by $BSS < 0.7$. This boundary distinguishes "safe degradation" from *behavioral collapse*.

The choice of the 0.7 threshold is grounded in the observed divergence between linguistic fluency and factual correctness. Below this threshold, the frequency of "silent failures" increases sharply: models maintain high Manner of Language (ML) scores while exhibiting significant degradation in Factual Correctness (FC) and Logical Correctness (LC).

By formalizing this threshold, the *Resilio* framework provides a practical safety metric for determining acceptable quantization levels in real-world deployment scenarios.

V. Experimental Setup and Implementation

The empirical validation of the *Resilio* framework involved a large-scale, multi-threaded computational pipeline designed to measure the intersection of model architecture, bit-precision depth, and input degradation. The implementation was architected to ensure strict environmental parity across all test configurations to isolate the variables of quantization and drift.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

A. Technical Infrastructure and Model Configurations

The study utilized a high-performance compute environment leveraging the *Hugging Face* ecosystem and the *BitsAndBytes* library for precision management. We selected three open-source models representing diverse parameter scales and architectural designs: *Llama 3.1 8B*, *Mistral 7B*, and *Phi-3 Mini*. Each model was instantiated across three distinct quantization variants:

- **FP16:** The 16-bit floating-point baseline.
- **INT8:** 8-bit quantization using vector-wise quantization logic.
- **INT4:** 4-bit quantization with double quantization enabled for maximum memory efficiency.

To maintain internal validity, we enforced consistent generation parameters across all 5,760 total inferences, utilizing a temperature of 0.7, a top-p value of 0.9, and a maximum token limit of 350.

B. The LLM-as-Judge Evaluation Pipeline

Manual annotation of 5,760 multi-factor responses was determined to be logistically unfeasible; we therefore engineered an automated evaluation pipeline utilizing *GPT-OSS 20B* as a neural rater, accessed via the *Groq API*. This *LLM-as-Judge* system was prompted with a structured JSON protocol to extract a five-dimensional scoring vector—Factual Correctness (FC), Logical Correctness (LC), Reasoning Coherence (RC), Sentiment Correctness (SC), and Manner of Language (ML)—for each response.

To resolve technical bottlenecks and ensure data integrity, the pipeline implemented:

- **Asynchronous Threading:** Designed to maintain an optimal throughput of 30 requests per minute (RPM) and manage API rate limits during inference.
- **Contextual Grounding:** Each evaluation request included the original clean prompt and ground-truth metadata to prevent evaluator hallucination.
- **Memory Management:** Automated GPU offloading and memory cleaning were utilized between quantization cycles to prevent VRAM fragmentation during local INT4/INT8 testing.

C. TPS and BSS Computation & Visualization

Following the evaluation phase, the extracted scoring vectors were aggregated to compute the primary metrics: Task Performance Score (TPS) and Behavioral Stability Score (BSS). For every individual response, the TPS was derived from a weighted combination of the five evaluation factors, with category-specific weights applied to reflect the unique requirements of Reasoning, Logic, Question Answering, and Classification tasks.

The Behavioral Stability Score (BSS) was subsequently computed at each drift level as the ratio between the mean TPS of degraded prompt variants ($\overline{TPS}_d$) and the TPS of the corresponding clean baseline (TPS_0). This normalization enabled direct comparison of degradation trajectories relative to ideal conditions, independent of model scale.

To facilitate granular analysis, the results were structured across dimensions of model architecture, quantization level, task category, and drift level. Visualization techniques—including line plots for BSS trajectories, bar charts for prompt-level variation, and heatmaps for category-wise stability patterns—were utilized to identify degradation trends and quantization-induced amplification effects.

This multi-level visualization framework enabled clear interpretation of behavioral instability across all experimental conditions.

VI. Results, Analysis, and Discussion

As shown in Fig. 1, all models exhibit a gradual degradation in stability as drift increases, with INT4 configurations collapsing significantly earlier.

Fig. 2a highlights task-specific divergence across categories, while Fig. 2b shows the global degradation trend across quantization levels.

Fig. 3a demonstrates that INT4 degradation is disproportionately higher, while Fig. 3b confirms statistical reliability.

Fig. 4 illustrates the higher-order interaction between architecture, quantization, and prompt drift.

The empirical evaluation of 5,760 model-response pairs reveals a complex, non-linear relationship between numerical precision and linguistic robustness. Our analysis centers on three primary phenomena: the amplification of stability loss through quantization, the inherent fragility of specific cognitive tasks, and the emergence of *silent failures* in highly compressed models.

A. The Quantization Amplification Effect

A central finding of this research is the *Quantization Amplification* effect, wherein reducing bit-precision does not merely lower absolute performance but actively accelerates the rate of behavioral degradation under input noise.

The empirical results reveal a consistent but nuanced Quantization Amplification effect. While the absolute BSS difference between INT4 and FP16 configurations remains modest (1–3 percentage points) at lower drift levels, the divergence becomes more pronounced at higher severity levels (L4–L5), particularly in reasoning-intensive tasks. For Phi-3 Mini, INT4 configurations show a steeper degradation slope compared to FP16, reaching BSS < 0.75 at L5 while FP16 maintains BSS > 0.80 in classification tasks. This non-linear acceleration in degradation rate—rather than a large absolute gap—constitutes the amplification effect we identify.

This pattern suggests that at 4-bit precision, the model's tolerance for token-level noise is reduced. Rather than a catastrophic collapse, the effect manifests as an earlier onset of degradation and a steeper decline trajectory under high drift, particularly for architectures with smaller parameter counts such as Phi-3 Mini.

B. Task-Specific Fragility: Reasoning vs. Classification

The results demonstrate that behavioral stability is highly task-dependent, as illustrated by category-wise mean BSS

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

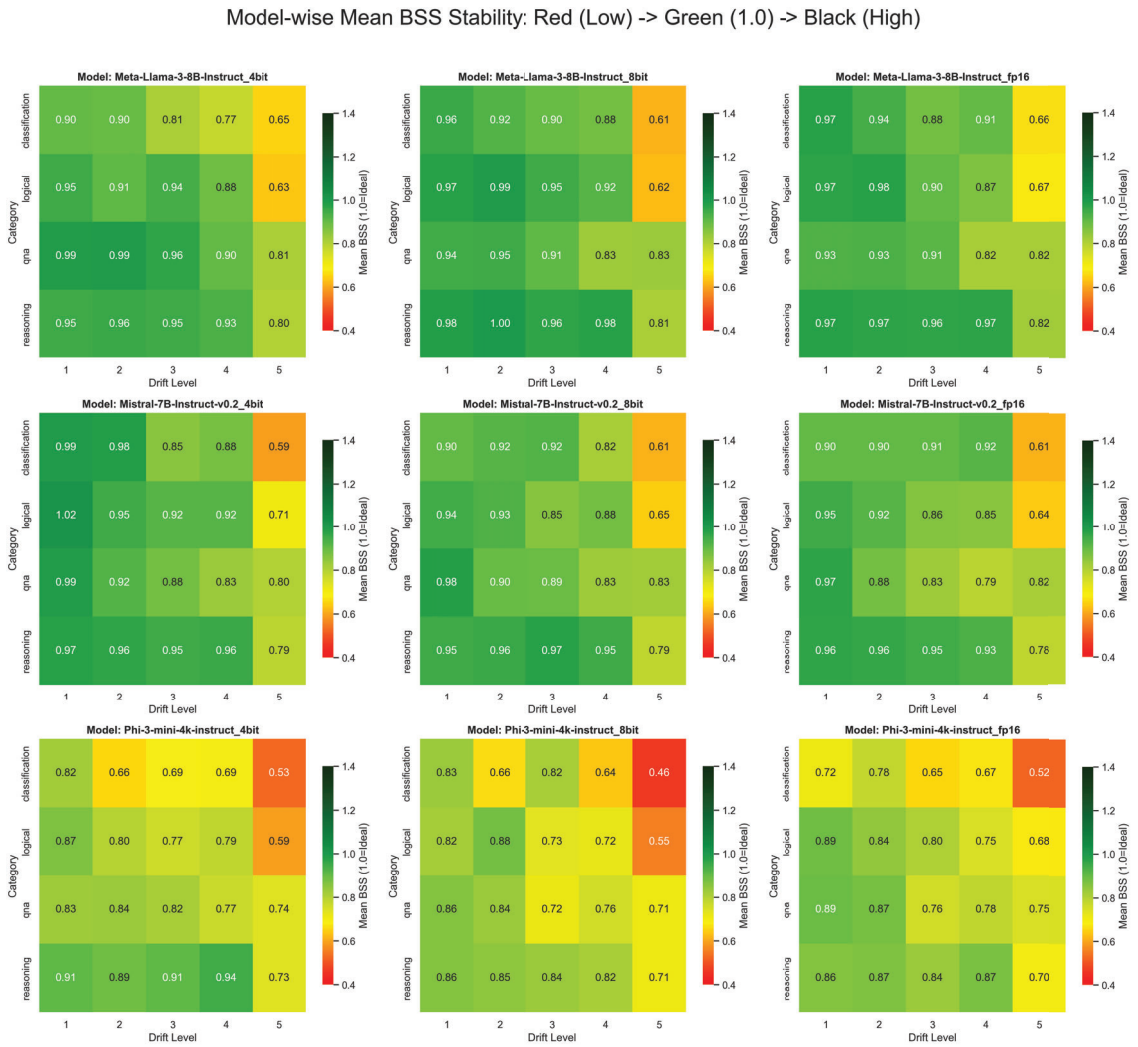


Fig. 1. All models exhibit progressive degradation with increasing drift severity. INT4 configurations demonstrate an earlier onset and steeper decline trajectory, particularly in reasoning tasks.

trends. A clear divergence emerges between pattern-matching tasks and computational reasoning tasks:

- **Logical and Reasoning Fragility:** Tasks requiring multi-step deduction (e.g., GSM8K and LogiQA) are the most vulnerable. In these domains, INT4 models reach the *behavioral collapse threshold* ($BSS < 0.7$) as early as L3 drift. The loss of precision disrupts chain-of-thought consistency, where a single perturbed token can redirect the reasoning trajectory entirely.
- **Classification Resilience:** In contrast, sentiment classification (SST-2) exhibits strong robustness. Even INT4 models maintain $BSS > 0.8$ through L3 drift across most architectures. This suggests that classification relies on global semantic signals that are less sensitive to localized perturbations introduced by quantization.

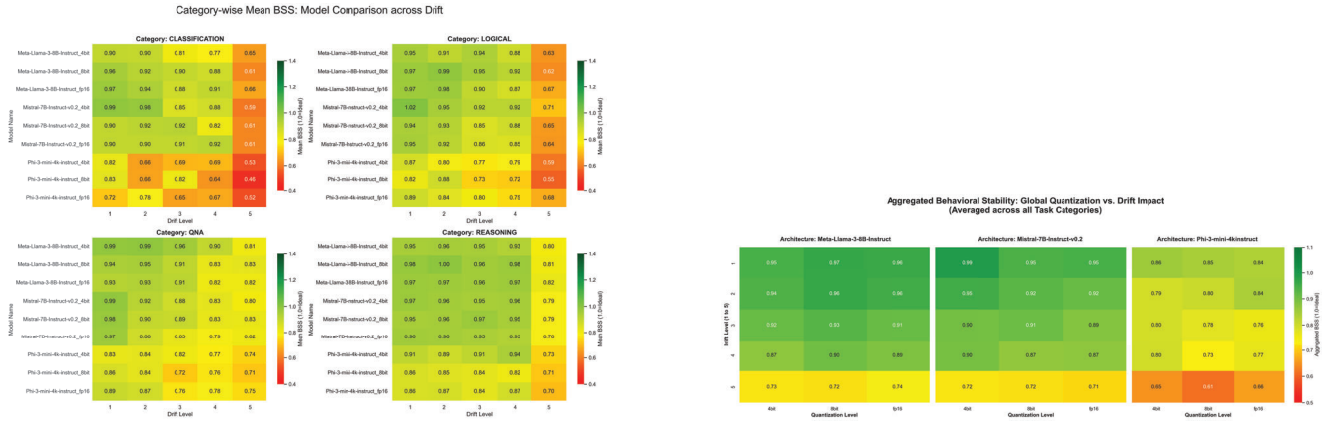
C. The Silent Failure Phenomenon and Calibration Loss

One of the most critical safety concerns identified in this study is the emergence of *silent failures*. This phenomenon is characterized by a divergence between linguistic fluency and factual correctness.

Our multi-factor evaluation shows that while *Manner of Language (ML)* scores often remain high, both *Factual Correctness (FC)* and *Logical Correctness (LC)* degrade significantly under high drift. In INT4 configurations, models frequently produce syntactically fluent and highly confident responses that are nevertheless factually incorrect or logically inconsistent.

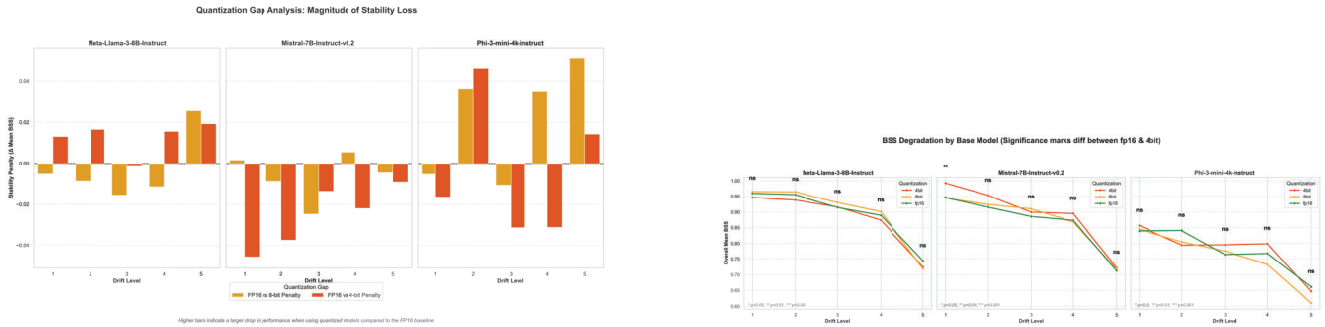
This *confidence-accuracy divergence* represents a critical deployment risk. Systems that fail with low confidence are manageable; however, systems that fail with high confidence introduce substantial risk in real-world applications.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026



(a) Category-wise BSS (b) Aggregated Quantization Impact

Fig. 2. (a) Category-wise BSS showing task-dependent robustness differences. (b) Aggregated quantization trends demonstrating a consistent decline in stability with reduced precision.



(a) Quantization Gap (b) Statistical Significance

Fig. 3. (a) Quantization gap analysis highlighting disproportionate stability loss in INT4 models. (b) Statistical validation confirming that degradation trends are significant across drift levels.

D. Discussion: Implications for Deployment

The results indicate the existence of a practical *stability ceiling* for quantized models. While INT4 quantization remains viable for low-complexity classification tasks under moderate noise, it is fundamentally unsuitable for reasoning-intensive applications without robust input preprocessing.

These findings highlight a critical trade-off: while aggressive quantization yields substantial memory and efficiency gains, it significantly increases the risk of behavioral collapse under realistic prompt conditions. Developers must therefore carefully balance compression benefits against robustness requirements when deploying LLMs in production environments.

VII. Deployment Guidelines

The findings of the *Resilio* study provide a definitive evidence base for optimizing the trade-off between computational efficiency and behavioral reliability. To assist practitioners in navigating these trade-offs, we propose a structured deployment matrix.

TABLE I
Deployment Decision Matrix

Task Category	Precision	Operational Constraint
Reasoning / Logic	FP16 / INT8	INT4 viable only if input similarity $> 90\%$
Fact-Retrieval (QA)	INT8	Safe for moderate drift; INT4 requires normalization
Classification	INT4	Robust under high drift conditions
Safety-Critical	FP16	Avoid quantization to prevent silent failures

A. Deployment Decision Matrix

Based on the observed stability thresholds ($BSS < 0.7$), we recommend the following precision-task mapping for production environments:

B. Strategic Recommendations for Practitioners

1) **Prioritize Input Normalization:** For reasoning-intensive tasks, applying a lightweight spell-correction or grammar-normalization layer prior to inference can significantly improve stability in low-precision models.
2) **Monitor Silent Failures:** Confidence scores should not be treated as reliable indicators of correctness in

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

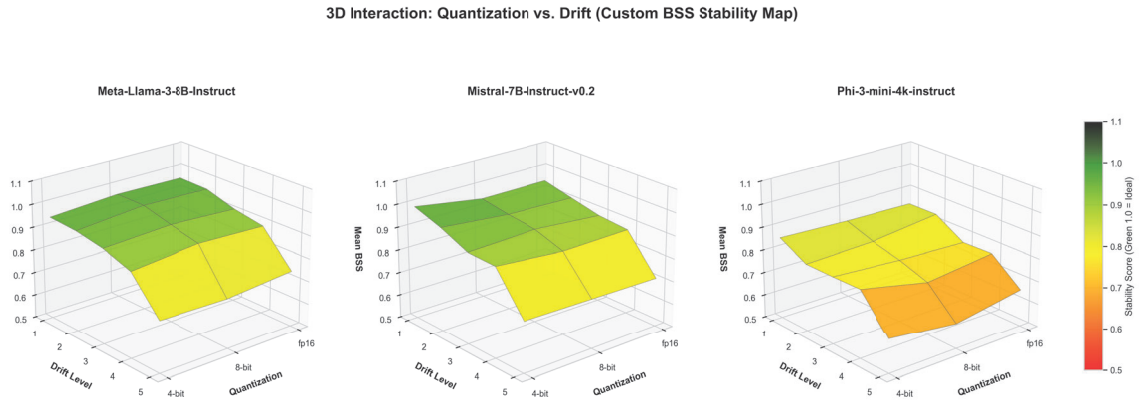


Fig. 4. 3D interaction between model architecture, quantization level, and drift severity, illustrating the compounded impact of compression and input degradation on behavioral stability.

quantized models. Low-precision systems may retain high linguistic fluency despite logical or factual errors.

3) **Implement Similarity Gates:** Production pipelines should track cosine similarity between incoming prompts and a clean reference distribution. If similarity falls below 0.80, requests should be escalated to higher-precision model variants.

VIII. Conclusion

This paper introduced the *Resilio* framework to quantify the interaction between model quantization and prompt drift. Through large-scale evaluation, we demonstrated the *Quantization Amplification* effect, showing that low-precision models are significantly more sensitive to input degradation.

By introducing the *Task Performance Score (TPS)* and *Behavioral Stability Score (BSS)*, this work establishes a standardized methodology for evaluating model reliability beyond traditional accuracy metrics.

The results highlight a critical trade-off: while aggressive quantization enables scalable deployment, it introduces non-linear risks in reasoning accuracy and model calibration. The *Resilio* framework provides a principled approach to navigating this trade-off, ensuring that efficiency gains do not compromise the behavioral integrity of deployed AI systems.

References

[1] T. Dettmers *et al.*, "LLM.int8(): 8-bit Matrix Multiplication for Transformers at Scale," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 35, 2022.
[2] Microsoft Research, "Phi-3 Technical Report: A Highly Capable Language Model Locally on Your Phone," arXiv:2404.14219, 2024.
[3] A. Zhu *et al.*, "Robustness in Large Language Models: A Survey of Mitigation Strategies and Evaluation Metrics," arXiv:2401.12926, 2024.
[4] Meta AI, "Llama 2: Open Foundation and Fine-Tuned Chat Models," arXiv:2307.09288, 2023.
[5] K. Zhu *et al.*, "PromptBench: Towards Evaluating the Robustness of Large Language Models on Adversarial Prompts," arXiv:2306.04528, 2023.
[6] J. Wang *et al.*, "On the Robustness of ChatGPT: An Adversarial and Out-of-distribution Perspective," arXiv:2302.12095, 2023.
[7] S. Rabanser, S. Günnemann, and Z. C. Lipton, "Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32, 2019.
[8] J. Wei *et al.*, "Finetuned Language Models Are Zero-Shot Learners," in *Proc. ICLR*, 2022, arXiv:2109.01652.
[9] T. Dettmers *et al.*, "QLoRA: Efficient Finetuning of Quantized LLMs," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 36, 2023, arXiv:2305.14314.
[10] Y. Wang *et al.*, "Self-Instruct: Aligning Language Models with Self-Generated Instructions," in *Proc. ACL*, 2023, arXiv:2212.10560.
[11] B. Kim *et al.*, "LLM-based Edge Intelligence: A Comprehensive Survey on Opportunities and Challenges," arXiv:2405.00379, 2024.
[12] J. Chang *et al.*, "A Survey on Evaluation of Large Language Models," *ACM Transactions on Intelligent Systems and Technology*, 2024, arXiv:2307.03109.