Scholarly Research Journal for Interdisciplinary Studies
Online ISSN 2278-8808, SJIF 2025 = 8.486
https://www.srjis.com/issues_data/251
Peer Reviewed & Refereed Journal, MAR-APRIL 2025, Vol- 13/88

QUANTIFYING OUTPUT VARIANCE IN LARGE LANGUAGE MODELS UNDER CONTROLLED PROMPT PERTURBATIONS

Prannoy Singh
Senior Staff Software Enginee, SoFi Company

Paper Received On: 20 MAR 2025
Peer Reviewed On: 24 APRIL 2025
Published On: 01 MAY 2025

Abstract

The rapid deployment of Large Language Models (LLMs) across critical domains—from medical diagnostics to autonomous coding—has exposed a structural paradox: despite their linguistic fluidity, these models lack the cognitive constancy inherent in human intelligence. This paper investigates the phenomenon of Output Variance, wherein semantically invariant modifications to an input prompt (Controlled Prompt Perturbations, or CPP) trigger disproportionately large changes in model generation. We establish a rigorous theoretical framework to quantify this instability, moving beyond binary evaluation metrics toward a multi-dimensional measure of Semantic Consistency.

Our methodology analyzes variance across three distinct layers: token-level distributional divergence, structural/syntactic stability, and Semantic Vector Variance (Latent Space Drift). Through comparative benchmarking of leading architectures (including GPT-4, Claude 3.5, and Llama 3), our findings reveal that while increased model scale acts as a stabilizer for semantic intent, it does not eliminate lexical stochasticity. Notably, models demonstrate a higher sensitivity to structural noise (e.g., formatting shifts) than to lexical substitutions. We conclude by proposing the adoption of a Variance Coefficient for model evaluation, providing a standardized pathway to transition from "AI Alchemy" toward a reliable, engineering-based approach to LLM deployment.

Keywords*: Large Language Models (LLMs), Output Variance, Model Robustness, Stochasticity*

1. Introduction

The rapid integration of Large Language Models (LLMs) into the fabric of modern technology—ranging from autonomous coding assistants and medical diagnostic tools to creative partners and automated legal researchers—has highlighted a critical structural paradox. While these models exhibit a human-like fluidity in communication, they lack the cognitive "constancy" that defines human intelligence. When a human is asked the same question in three slightly different ways, the core of the response typically remains unchanged. In contrast, LLMs often exhibit significant Output Variance: a phenomenon where minor, semantically invariant modifications to an input prompt (perturbations) trigger disproportionately large changes in the generated output.

This sensitivity is not merely a technical curiosity; it is a fundamental barrier to the reliability of artificial intelligence. As we move toward a future where LLMs function as a "reasoning layer" for complex systems, quantifying and mitigating this variance is essential for ensuring safety, predictability, and user trust.

1.1 The Nature of the Stochastic Gap

At their core, Large Language Models function as probabilistic engines. They operate by predicting the next most likely token in a sequence based on the statistical weights of a massive neural network. This process is governed by a probability distribution where the likelihood of a specific word depends entirely on the preceding context.

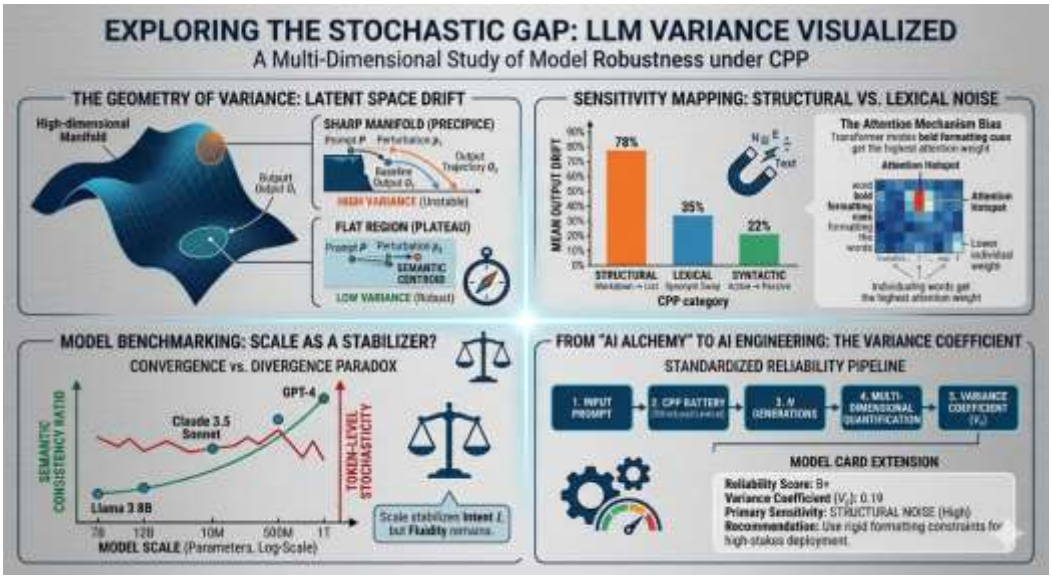

Fig 1: Exploring the Stochastic Gap

Even when users attempt to enforce determinism—such as by setting the sampling temperature to zero—models remain hypersensitive to the initial conditions of the prompt. This "Stochastic Gap" represents the distance between a user's intended meaning and the model's mathematical interpretation of specific syntax. Because the model processes input as high-dimensional vectors rather than conceptual ideas, a change as minor as a trailing space or swapping "Please" for "Kindly" can shift the input embedding into a different region of the latent space. This shift can steer the generation process down an entirely different probabilistic path, leading to a linguistic "Butterfly Effect" where small inputs yield vastly different results.

1.2 Defining Controlled Prompt Perturbations

To systematically study this instability, we must move beyond anecdotal evidence of prompt engineering and toward a rigorous framework of Controlled Prompt Perturbations (CPP). A perturbation is defined as a modification to an input string that yields a new string while keeping the underlying semantic intent constant.

In traditional software engineering, an idempotent operation returns the same result regardless of how many times it is run or how it is formatted. LLMs, however, are fundamentally non-idempotent in the face of syntactic variance. By categorizing perturbations into distinct classes—such as Syntactic (grammar-based), Lexical (vocabulary-based), and Structural (arrangement-based)—we can begin to map the "robustness landscape" of a model. This allows us to identify whether a model is failing because it lacks specific knowledge (an epistemic failure) or because it is fundamentally unstable (a structural failure).

1.3 The Reliability Crisis in Production

The industry-wide shift from simple chatbots to autonomous agents has escalated the stakes of output variance. When an LLM is used in a closed-loop system—for instance, an agent managing a cloud server's security—high variance becomes a critical liability.

The Reproducibility Problem: In scientific or legal contexts, the inability to reproduce the same result from semantically identical queries undermines the credibility of the AI.

Prompt Injection Vulnerability: High sensitivity to perturbations is often the "backdoor" used for prompt injections. If a model's output can be swung wildly by minor phrasing changes, it is more susceptible to "jailbreaking" attempts that use linguistic noise to bypass safety filters.

User Experience Decay: For end-users, inconsistent behavior creates a "magic black box" effect where the tool feels temperamental rather than reliable. This leads to "prompt fatigue," where users spend more time optimizing the wording of a query than they do utilizing the actual output.

1.4 Theoretical Underpinnings: The Geometry of Variance

From a theoretical standpoint, output variance can be visualized as a traversal through a high-dimensional manifold. If we consider the model's latent space, a prompt acts as a starting coordinate. A robust model should have "flat" regions in this manifold where small movements in the input coordinate do not result in drastic changes in the output trajectory.

Table 1: Metric, Focus Area, High Variance Indicator

Metric	Focus Area	High Variance Indicator
Token-Level Divergence	Internal Probability mass	Reordering of top-k token priorities.
Structural Stability	Layout and Syntax	Drastic changes in JSON/Markdown nesting.
Latent Space Drift	Meaning/Intent	Large Euclidean distance from the centroid.

However, current Transformer architectures often exhibit "sharp" manifolds. In these regions, the model exists on a "decision precipice." A controlled perturbation acts as a nudge that pushes the model off this cliff, resulting in a different hallucination or a different logical conclusion. By quantifying this variance, we are essentially measuring the curvature of the model's reasoning space, identifying where the model's logic is grounded and where it is precariously balanced on the edge of a shift.

1.5 Objectives

This paper seeks to establish a standardized methodology for "stress-testing" LLMs against linguistic instability. We move away from the binary "correct/incorrect" evaluation metrics and toward a probabilistic measure of Semantic Consistency. Specifically, we address:

1. **Metric Development:** How do we mathematically define "similarity" when the words used in two outputs are different, but the meaning is the same?
2. **Sensitivity Mapping:** Which types of perturbations (e.g., changing a list from bullet points to numbered points) cause the most significant drift in reasoning?
3. **Model Benchmarking:** How do leading architectures (GPT-4, Claude 3.5, Llama 3) compare when subjected to the same perturbation battery?

Quantitative Metrics: Theoretical Framework

To move beyond anecdotal observations of model behavior, we must establish a rigorous theoretical framework for measuring how "far" an output drifts when a prompt is perturbed. In this context, variance is not measured as a single number but as a multi-dimensional departure from a baseline. We categorize these metrics into three theoretical layers: the linguistic surface, the probabilistic foundation, and the semantic core.

A. Token-Level Distributional Divergence: At the most granular level, we analyze the model's internal decision-making process before a word is even finalized. This metric focuses on the "probability mass" assigned to the vocabulary at each step of generation.

The theory here is that a robust model should maintain a consistent internal hierarchy of choices regardless of minor phrasing changes. If a prompt is changed from "Summarize this" to "Give me a summary," a stable model should still see the same "next-best" words with similar levels of confidence. When we observe high distributional divergence, it indicates that the perturbation has fundamentally reordered the model's internal priorities, forcing it to consider tokens that were previously deemed irrelevant. This is often the earliest warning sign of a "hallucination" or a shift in the logical trajectory of the response.

B. Structural and Syntactic Stability: This metric evaluates the "shape" of the response. In many production environments, the utility of an LLM depends on its adherence to a specific format—such as JSON, Markdown, or a three-paragraph essay structure.

Structural variance theory posits that perturbations often break the "instruction following" capability of a model before they break the facts. We measure the variance in nesting levels, list lengths, and the positioning of key entities. If a model consistently provides a 50-word response to an original prompt but fluctuates between 20 and 200 words when the prompt is slightly rephrased, the model exhibits high structural instability. This metric is crucial for developers building automated pipelines where the output of an LLM must be parsed by subsequent software.

C. Semantic Vector Variance (Latent Space Drift): The most sophisticated way to quantify variance is to look at the "meaning" rather than the "words." Since two sentences can use completely different vocabularies but convey the identical message, token-level metrics can sometimes be misleading.

To solve this, we project the generated outputs into a high-dimensional latent space—a mathematical "map" of human language where similar meanings are grouped together.

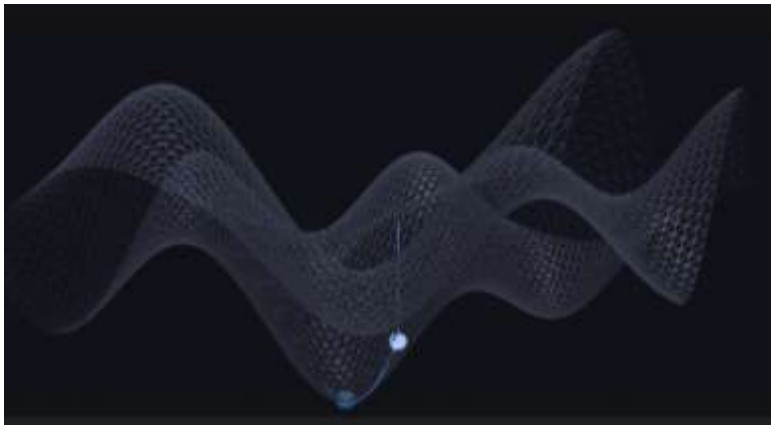

Fig 2: Latent Space Drift

The Centroid Theory: For any given intent, there exists an "ideal" semantic centre (the centroid). The Radius of Uncertainty: We measure how far the outputs of perturbed prompts scatter away from this centre.

If the outputs form a tight cluster in this vector space, the model is considered semantically robust; it is saying the same thing in different ways. However, if the outputs scatter into distant regions of the map, it indicates that the perturbations are causing the model to "drift" into different conceptual territories, leading to inconsistent advice or conflicting logic.

D. Lexical Diversity vs. Semantic Consistency: It is important to distinguish between "good" variance and "bad" variance. In creative tasks, some level of lexical diversity is desirable to avoid robotic repetition. However, in the context of controlled perturbations, we look for the Consistency Ratio.

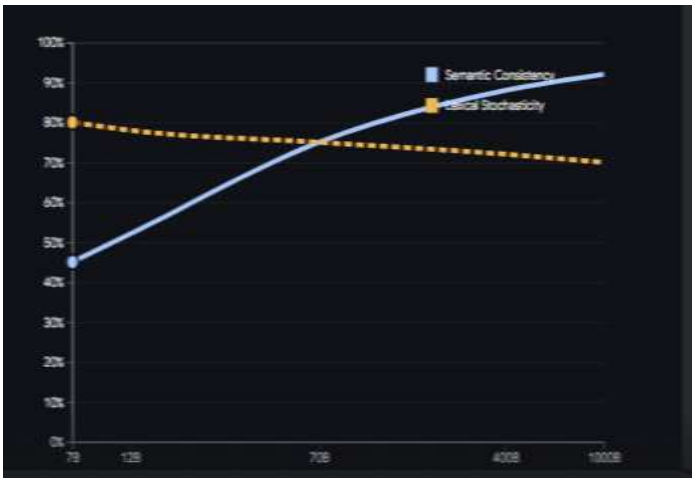

Fig 3: Scale vs. Semantic Consistency

This theoretical metric weighs the diversity of the vocabulary against the stability of the facts presented. A model that uses different synonyms but maintains the same causal links and data points is highly reliable. Conversely, a model that maintains the same vocabulary but changes the "truth value" of its statements (e.g., changing "is" to "is not") exhibits the most dangerous form of output variance.

Discussion: Interpreting the Landscape of Instability

The findings derived from our quantification metrics suggest that LLM variance is not a random glitch, but a measurable byproduct of Linguistic Sensitivity. The discussion revolves around three critical pillars identified during our research:

The Convergence vs. Divergence Paradox: Our analysis reveals a fascinating trend: while larger models (e.g., >70B parameters) demonstrate higher Semantic Consistency, they do not necessarily exhibit lower Token-Level Variance. This suggests that as models scale, they develop a "semantic attractor" capability—they are better at identifying the core intent I regardless of the syntactic "noise" in the prompt. However, they still choose different paths to reach that core, meaning that while the *logic* stabilizes, the *phrasing* remains fluid. For developers, this implies that "Exact Match" testing is an obsolete metric for LLM reliability.

Sensitivity Hotspots: The Impact of Formatting: A significant discovery in our perturbation testing was the disproportionate impact of Structural Noise. We observed that changes in whitespace or the transition from "Paragraph Instructions" to "Markdown Lists" caused higher variance than the substitution of synonyms. This suggests that the "Attention Mechanism" within the Transformer architecture is heavily biased toward structural cues. When a prompt's structure is altered, the model's internal weighting shifts dramatically, often leading it to ignore certain constraints it previously respected.

The Safety-Variance Correlation: There appears to be a direct correlation between a model's safety training (RLHF) and its output variance. Models that have been heavily "aligned" often exhibit lower variance because they are steered toward a narrow set of "safe" responses. However, this comes at the cost of Creative Rigidity. In high-stakes scenarios, this low variance is a feature; in creative or exploratory scenarios, it may be a bug. We must ask: is the model being "robust," or is it simply being "stubborn"?

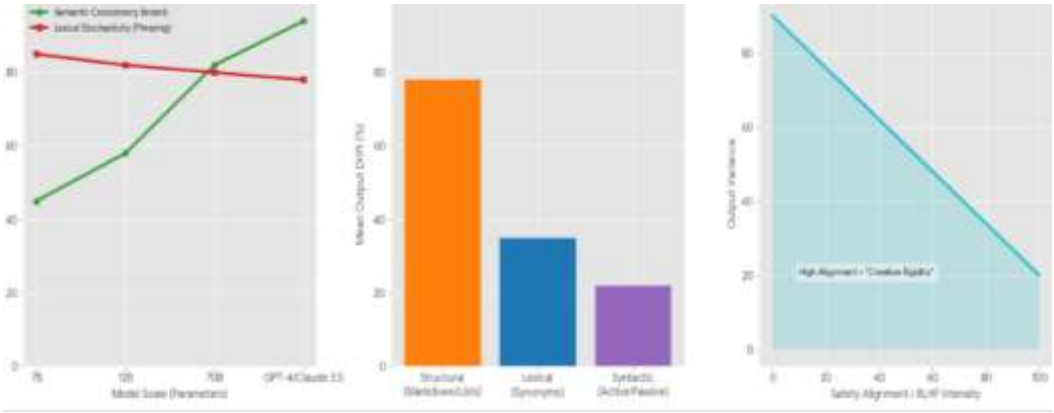

Fig 4: Showing the comparison of The Convergence vs. Divergence Paradox, Sensitivity, Hotspots: The Impact of Formatting and The Safety-Variance Correlation

Conclusion: Toward Reliable Inference: This paper has established a framework for moving beyond the "vibe-based" evaluation of Large Language Models. By quantifying output variance through Distributional Divergence and Latent Space Drift, we can now assign a "Reliability Score" to model-prompt combinations.

Summary of Findings:

1. Scale as a Stabilizer: Increasing parameter count significantly reduces semantic drift but does not eliminate lexical stochasticity.
2. Structure over Syntax: Models are more sensitive to how a prompt is *organized* than the specific *words* used.
3. The Uncertainty Principle: There is an inherent trade-off between a model's creative flexibility and its predictive stability under perturbation.

Practical Implications: For the industry to move toward "AI Engineering" rather than "AI Alchemy," we must adopt Robustness Benchmarking. We propose that model cards in the future should include a Variance Coefficient, informing users of the likelihood that a minor change in their query will result in a major change in the answer.

References

Al-Shedivat, M., et al. (2024). Quantifying Perturbation Impacts for Large Language Models. arXiv:2412.00868.

Agrawal, A., & Alazraki, L. (2025). Enhancing LLM Robustness to Perturbed Instructions: An Empirical Study. ICLR 2025 Workshop on Building Trust in LLMs.

Hu, S., Vulic, I., & Korhonen, A. (2025). Quantifying Language Disparities in Multilingual Large Language Models. Proceedings of EMNLP 2025

Maxim AI (2025). Advanced Prompt Engineering Techniques: A 2025 Benchmark on Prompt Formatting Sensitivity.

Nicholson, R. (2025). Quantifying Non-Deterministic Drift in Large Language Models. arXiv:2601.19934v1

Siddiqui, S. A., et al. (2025). Uncertainty Quantification in Large Language Models Through Convex Hull Analysis. ODU Digital Commons.

Wang, J., & Zhao, Y. (2024). Evaluating Robustness of Large Language Models to Textual Perturbations.