Position: Diagnosing AI Model Jaggedness with Normative Neighborhoods

Anonymous ACL submission

Abstract

Frontier AI systems often exhibit *jaggedness*, abrupt changes in performance or behavior across inputs that appear closely related. Existing work has characterized jaggedness through capability profiles, prompt-sensitivity analyses, and workflow-level evaluations. This position paper reframes jaggedness as a problem of *normative local behavior*: what should remain stable, and what should change across nearby task inputs or task variants? We introduce *normative neighborhoods*, sets of related inputs or variants that specify the expected pattern of model behavior, such as when outputs should remain invariant, when confidence should adjust, and when strategies should change. By separating justified sensitivity from unjustified discontinuity, normative neighborhoods support a diagnostic account of jaggedness; the same surface-level instability may arise from different mechanisms, such as unstable task representation, poor uncertainty handling, or workflow bottlenecks. This perspective specifies which variation is expected, which variation is surprising, and which system component should be tested and/or redesigned, moving AI evaluation beyond describing jaggedness toward diagnosing and addressing it.

1 Introduction

A growing challenge in evaluating frontier AI systems is that their behavior can change abruptly across seemingly similar situations (Mineault et al., 2026; Shu and Wei, 2026). A model may solve a difficult reasoning problem but fail on a nearby variant (Song et al., 2025), answer correctly under one wording but incorrectly under a paraphrase (Razavi et al., 2025), or change its conclusion when the same evidence is reordered (Zhang et al., 2025b). This pattern, often described as *jaggedness*, complicates deployment because model capability does not vary smoothly across tasks, prompts, and contexts (Gans, 2026; Campbell et al., 2025; Morris et al., 2026). Figure 1 depicts this intuition; performance may be high in some task regions but drop sharply in others, producing an uneven capability profile rather than a smooth frontier. Jaggedness challenges the common assumption that intelligence can be understood along a single scale (Chollet et al., 2025; Romero-Alvarado et al., 2026; Voudouris et al., 2026).

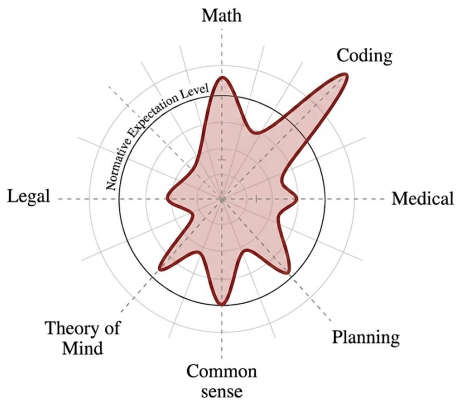

Figure 1: Schematic depiction of jaggedness in AI systems. Each spoke represents a task family, and distance from the center denotes task performance. The uneven frontier shows that a model may exceed a reference level in some regions while falling below it in others, rather than varying smoothly along a single capability scale.

Pervasive deployment of LLMs has made jaggedness increasingly visible. The "jagged technological frontier" describes how AI assistance can improve performance on some tasks while harming performance on nearby tasks outside a system's effective capability boundary (Dell'Acqua et al., 2023). Prior work characterizes jaggedness through capability profiles, prompt sensitivity (Razavi et al., 2025; Ribeiro et al., 2020; Gardner et al., 2020), workflow bottlenecks (Fan et al., 2025), long-horizon failures (Sinha et al., 2025; Guo et al., 2026), confidence failures (Xu et al., 2025; Steyvers et al., 2025), and model-level safety and usability analyses (Morris et al., 2026). Yet this work primarily shows *where* models are uneven; it does not fully explain *when* local variation is mean-

ingful, *why* it arises, or *which* intervention should follow. The key problem is that jaggedness cannot be defined by input similarity alone (Ribeiro et al., 2020). Two inputs may look similar while differing in decision-relevant information; two others may look different while preserving the same underlying decision problem (Sanwal, 2024; Jumelet et al., 2026). A useful theory of jaggedness must specify the local conditions under which model behavior should remain stable, when expressed confidence or response strategy should adjust, and when outputs should change in response to task-relevant differences. We call such units *normative neighborhoods* and define them concretely in Section 2.

This framing separates *justified sensitivity* from *unjustified discontinuity*. Not all local variation is a failure. A small input change may make the task more ambiguous, change the available evidence, alter the consequences of a mistake, or call for a different response strategy. The judgment and decision-making literature offers a useful basis for making these distinctions because it separates normative models of how agents should decide (Wald, 1950; Savage, 2012) from descriptive accounts of how agents actually behave (Kim, 2025). It also provides well-studied examples of structured departures from normative behavior, including framing effects (Tversky and Kahneman, 1981), ambiguity sensitivity (Einhorn and Hogarth, 1986; Fox and Ülkümen, 2011), order effects (Hogarth and Einhorn, 1992), bounded rationality, and miscalibration (Juslin and Olsson, 2007).

This perspective turns jaggedness from a single empirical symptom into a *diagnostic* problem. Our goal is not to propose metrics for jaggedness or to argue that all variation should be penalized. Rather, we propose a diagnostic framework to specify which variation is expected, which is surprising, and which system component should be evaluated or redesigned. This requires evaluating models over neighborhoods rather than isolated examples, measuring whether local behavior changes in the expected direction, and matching interventions to the mechanism that produced the discontinuity.

The paper makes three contributions. First, it distinguishes levels of jaggedness and clarifies the unit of comparison at each level. Second, it introduces normative neighborhoods as a way to specify when local variation should be stable, calibrated, or sensitive. Third, we propose a taxonomy of diagnostic hypotheses that link observed discontinuities to plausible mechanisms and responses.

2 Normative Neighborhoods

The key question in jaggedness is when variation is meaningful (Franceschelli and Musolesi, 2025; Mathur et al., 2026; Shah et al., 2026b). A *normative neighborhood* is a set of inputs, task variants, contexts, or system states over which an idealized decision policy specifies what should remain stable and what is free to vary. Such a neighborhood is not discovered from input similarity alone (Gardner et al., 2020; Sanwal, 2024; Jumelet et al., 2026). It is constructed from a task-specific theory of relevance: an account of which features, constraints, evidence, costs, preferences, or policies should matter for model behavior in a given setting. The source of this expectation may differ across tasks. In some cases, it may come from formal decision theory or task instructions; in others, from expert judgment, benchmark design, deployment policy, user preferences, or safety constraints. These sources are not interchangeable, so a neighborhood specification should make explicit where its normative expectation comes from.

More concretely, a normative neighborhood contains four elements: **(1) a base case, (2) a set of variants, (3) a transformation relating the variants to the base case, and (4) an expected behavioral relation**. The transformation may preserve decision-relevant information, alter uncertainty, or change task-relevant features. The expected behavioral relation specifies what should happen to the model's output, confidence, or strategy across those variants. The neighborhood makes explicit the basis on which nearby cases can be compared. This is a precondition for diagnosing jaggedness, because local instability is only interpretable once the comparison itself is justified.

Example: Clinical Note Generation

Consider an LLM that writes a medical progress note from a therapy-session transcript (Shah et al., 2025).

1. Base case. The transcript says that the client is sleeping worse and feels anxious, but does not clearly state whether the client has taken medication.

2. Variants. Nearby versions of the transcript may rephrase the same facts, say only that the client took "their usual medicine," name a specific medication, or omit the medication detail entirely.

3. Transformation. These variants change different things: some change the wording, some add useful detail, and some leave an important detail unclear.
4. Expected behavior. The note should preserve the same clinical content when only the wording changes, include the medication detail when it is stated, and flag or ask about the missing information when it is unclear.

This example shows that a neighborhood does not require identical notes across all variants. It specifies which changes should be ignored, which should make the model more cautious, and which should change the clinical content.

The aim is to make the capability surface less surprising: smoother where differences are irrelevant, appropriately sloped where uncertainty changes, and protected against cliffs where small irrelevant perturbations produce large behavioral shifts. These expectations give rise to three recurring neighborhood types, which we describe next.

Invariance neighborhoods Invariance neighborhoods capture cases where the transformation (3) preserves the relevant decision structure, so the expected behavioral relation (4) is stable across variants. If two prompts preserve the same relevant information, then a model's answer, confidence, or recommendation should not change sharply simply because the wording, formatting, or order of presentation changed. Paraphrase sets (Liu et al., 2024b), equivalent mathematical formulations (Zhou et al., 2024), and reordered but logically identical task descriptions are examples of such neighborhoods (Zhang et al., 2025b; Song et al., 2026). Failures in these cases are especially informative because they provide negative tests of whether the model has learned the relevant decision structure. This way, invariance neighborhoods connect to robustness, but ask a more specific question: whether the perturbation should have mattered for the task in the first place.

Calibration neighborhoods. Calibration neighborhoods capture cases where the transformation (3) changes the uncertainty of a decision, rather than directly changing the decision-relevant state. Here, the expected behavior (4) is calibrated adjustment: the model should change its confidence or strategy in proportion to changes in ambiguity, missing information, evidential weakness, or risk (Cheng et al., 2024; Bani-Harouni et al., 2025). For example, if an input becomes more ambiguous, underspecified, or evidentially weak, a reliable model should not necessarily give the same answer as before. It may instead lower confidence, ask for clarification (Wu, 2023), retrieve information, or abstain (Machcha et al., 2026). Jaggedness arises when the model's uncertainty response is itself unstable: confident in one underspecified variant, cautious in another equivalent one, or insensitive to a genuine increase in ambiguity.

Sensitivity neighborhoods. These neighborhoods capture cases where the transformation (3) changes a task-relevant feature that should alter the substantive response (4). Some perturbations change the probability of an outcome, the cost of an error, or the user's stated preference. In these cases, model behavior should change, but in a direction and magnitude that can be justified by the decision structure. The expected change is not only a shift in confidence, but a predictable change in the answer, recommendation, plan, or action. Sensitivity neighborhoods therefore support a form of sensitivity analysis: systematically varying task-relevant features while holding irrelevant features fixed, and testing whether the model's response changes in the expected direction (Frank and Goodman, 2025). This converts generalization into a local question (Bai et al., 2025): not only whether a model transfers to new cases, but whether its behavior changes in the way the decision structure requires. A model that gives identical advice after a risk level increases may be too insensitive; a model that reverses its recommendation after a minor, irrelevant change may be too brittle.

The clinical note example illustrates all three cases: paraphrased transcripts test invariance, missing medication information tests calibration, and newly specified medication details test sensitivity. This framing clarifies the relation between jaggedness and familiar evaluation concepts without collapsing into any one of them. Robustness asks whether a system maintains performance under perturbation; normative neighborhoods ask whether the perturbation should have mattered for the task. Calibration asks whether expressed confidence tracks correctness; the neighborhoods ask whether confidence changes appropriately across related variants. Generalization asks whether a system transfers to new cases; the neighborhoods ask what pattern of transfer is expected locally.

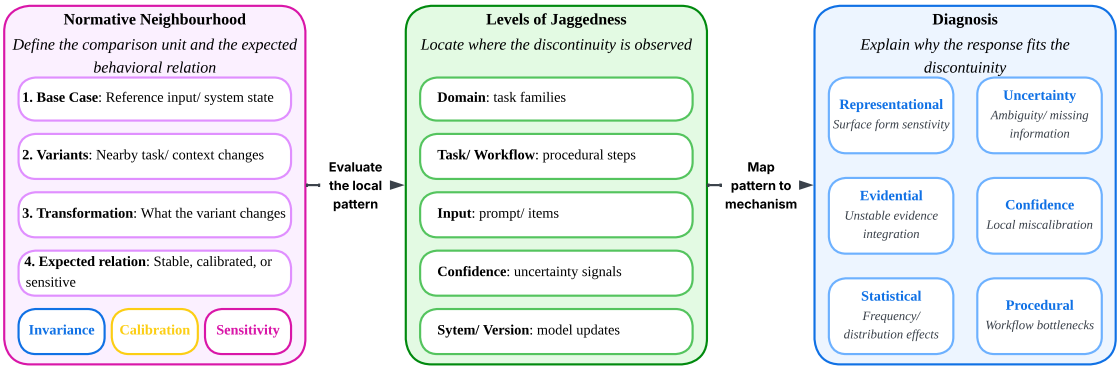

Figure 2: A diagnostic pipeline for tackling jaggedness in AI systems.

Stochasticity concerns random variation in outputs; jaggedness concerns structured discontinuities that persist across repeated or controlled evaluation. Only after the relevant expectation is specified can we determine whether observed variation is justified sensitivity or unjustified discontinuity.

3 Jaggedness Across Levels

Having defined normative neighborhoods, we now ask what counts as "nearby." The answer depends on the level of analysis: the comparison may be between domains of competence, task formulations, workflow steps, confidence reports, or model versions. These levels indicate where a discontinuity is observed (though not yet why it occurs). This section reviews jaggedness across these levels and their relation to neighborhood types (see Table 1). Across these levels, the four elements remain fixed, but their instantiation changes.

Domain-level. At the broadest level, models may show uneven capability across domains (Mathur et al., 2026). A system can perform strongly on programming tasks while being unreliable in legal, medical, or social contexts; see Figure 1. Domain-level analysis is useful because broad labels such as "good at reasoning" often hide differences among task families that rely on different forms of knowledge, evidence, and decision structure.

Task- and workflow-level. Within a domain, jaggedness can arise across the steps of a task or workflow. A model may retrieve relevant information but fail to synthesize it, decompose a problem well but make a local arithmetic error, or solve short tasks while failing over longer horizons (Sinha et al., 2025; Zhang et al., 2025a). In such cases, the system is not uniformly incapable. Rather, a local failure in execution, consistency, or intermediate decision-making can become a bottleneck that dominates total performance. At the workflow level, jaggedness requires evaluation of process structure, not only of final answers. It asks which substep fails, how that failure propagates, and whether decomposition, routing, verification, or tool use can contain it.

Input-level. At a finer scale, models may vary across equivalent or near-equivalent formulations of the same task. This includes paraphrases, changes in formatting, reordered evidence, or small shifts in contextual framing (Shah et al., 2026a). Prompt-sensitivity work shows that model outputs can change substantially under such perturbations (Razavi et al., 2025). This level is especially important because it challenges the assumption that local changes in form preserve local behavior.

Confidence-level. Jaggedness can also appear in expressed uncertainty. A model may give the same answer with very different confidence across paraphrases, express high certainty on brittle or incorrect outputs, or fail to lower confidence when an input becomes ambiguous. This matters because reliability in deployment depends not only on whether a model is correct, but also on whether it knows when it is likely to be wrong (Xu et al., 2025; Steyvers et al., 2025). Confidence-level jaggedness connects performance variation to local calibration: whether expressed confidence changes appropriately across nearby cases.

Version- and system-level. Finally, jaggedness can appear across model versions, tool configurations, and deployment settings. A new model release may improve performance on some abilities while degrading others; adding retrieval, tools, memory, or system prompts may reduce one form of jaggedness while introducing another. This level

Level	Base case / variants	Invariance	Calibration	Sensitivity
Domain	*Base:* a task in one domain. *Variants:* structurally related tasks in nearby domains.	Same decision rule should yield stable behavior.	Domains with poor base rate information or a lack of clear comparison classes should produce greater uncertainty.	Domain-specific constraints, costs, or evidence should alter behavior.
Task/ workflow	*Base:* a task decomposition or execution path. *Variants:* alternative decompositions, tool-use paths, verification steps, or intermediate states.	Equivalent paths should preserve the final outcome.	Uncertain or high-risk steps should trigger verification, routing, or lower confidence.	Different procedural demands should change tools, subtasks, or escalation.
Input	*Base:* a prompt, item, or local task formulation. *Variants:* paraphrases, formatting changes, reordered evidence, or minimally changed inputs.	Surface-only changes should not change the answer.	Ambiguity or weak evidence should elicit hedging, clarification, retrieval, or abstention.	Changed facts, evidence, or preferences should change the response.
Confidence	*Base:* a response with an expressed uncertainty signal. *Variants:* nearby cases with matched or systematically varied evidence, ambiguity, or difficulty.	Matched evidence and difficulty should produce similar confidence.	Confidence should track uncertainty, evidence strength, and likely correctness.	Confidence should shift predictably as risk, ambiguity, or evidential support changes.
Version/ system	*Base:* a model version, tool configuration, prompt policy, or deployed system state. *Variants:* updated models, tools, retrieval, memory, safety policies, or prompt changes.	Unchanged task expectations should preserve behavior.	Updates should preserve or improve uncertainty handling.	New tools, retrieval, memory, or policies should justify behavioral change.

Table 1: Operationalizing normative neighborhoods across levels of jaggedness. Each row identifies the level-specific unit of comparison by specifying a base case and variants; each cell specifies the expected behavioral relation under invariance, calibration, or sensitivity transformations.

of jaggedness differs from workflow-level jaggedness, where the concern is how a fixed system performs across steps within a task. Here, the concern is how behavior shifts when the model or deployment configuration changes. Version- and system-level jaggedness is consequential for real-world systems because users and developers often assume that upgrades monotonically increase capability. In practice, system changes can reshape the capability surface in ways that are difficult to anticipate without regression tests over carefully chosen neighborhoods.

The clinical note example can also be read across levels: paraphrase sensitivity appears at the input level, uncertainty about medication use appears at the confidence level, and failures to retrieve or verify medication information appear at the workflow level. *These levels locate discontinuities. The next section asks what mechanisms may produce them.*

4 A Diagnostic Taxonomy of Jaggedness

Normative neighborhoods specify when local variation is expected or surprising. Diagnosis then asks what produced the surprising cases. Table 2 maps recurring types of jaggedness to the local discontinuity they describe, the likely mechanism behind it, and a candidate intervention. The goal is not to provide an exhaustive ontology, but to determine what kind of local instability is present and what response it calls for.

Representational Jaggedness Representational jaggedness occurs when a model's behavior changes across alternative representations of the same underlying task. This includes sensitivity to paraphrase (Razavi et al., 2025), ordering of answer choices (Pezeshkpour and Hruschka, 2024), or other presentation details that should not alter the relevant decision problem. If two prompts keep the same information and ask for the same judgment, large changes in output suggest that the model has not formed a stable representation of the task.

This form of jaggedness is closely related to invariance. In decision-theoretic terms, equivalent formulations should not produce different preferences or conclusions merely because they are framed differently (Tversky and Kahneman, 1981; Wald, 1950; Savage, 2012). In AI evaluation, the analogous concern is that models may respond to superficial cues rather than task-relevant structure. For example, a model may correctly answer a reasoning problem in a familiar format but fail when the same problem is reworded, or it may shift its recommendation when options are listed in a different order (Kirshner et al., 2025). Such failures are particularly important because they can remain hidden in benchmark settings where each item is evaluated only once (Lalai et al., 2025).

Type	Local discontinuity	Likely mechanism	Candidate intervention
Representational	Equivalent framings, paraphrases, or formats yield different behavior.	Surface-form sensitivity; unstable task representation.	Paraphrase consistency tests, input normalization, representation alignment.
Uncertainty	Ambiguous or underspecified inputs trigger inconsistent strategies.	Failure to distinguish epistemic uncertainty, aleatoric uncertainty, ambiguity, and missing info.	Clarification, retrieval, abstention, uncertainty-aware routing.
Evidential	The same evidence supports different conclusions when reordered or reformatted.	Unstable evidence integration; primacy, recency, or positional effects.	Structured evidence tracking, order-robust evaluation, explicit updating.
Confidence	Confidence shifts without matching changes in evidence, difficulty, or correctness.	Local miscalibration; weak discrimination between reliable and unreliable outputs.	Calibration training, selective prediction, confidence-aware abstention.
Procedural	Multi-step performance collapses at specific substeps or decision points.	Workflow bottlenecks; weak local capabilities; error propagation.	Decomposition, verification, tool use, routing, aggregation.
Statistical	Common and rare variants behave differently despite equivalent task structure.	Sensitivity to training frequency, input statistics, or output likelihood.	Frequency-controlled evaluation, rare/common variant testing, distribution-aware training.

Table 2: A diagnostic taxonomy of jaggedness. The categories identify different sources of local discontinuity and suggest different forms of mitigation.

Uncertainty Jaggedness Uncertainty jaggedness arises when a model responds inconsistently to ambiguity, underspecification, or missing information (Einhorn and Hogarth, 1986; Fox and Ülkümen, 2011; Xiao et al., 2025). The inconsistency is not only in reported confidence, but in the response strategy the model selects. Unlike representational jaggedness, some variation in these cases may be justified. If a prompt becomes underspecified, a reliable model should not necessarily give the same answer as it would for a fully specified prompt. It may need to ask a clarification question, retrieve additional information, hedge, or abstain (Wu, 2023; Kadavath et al., 2022; Zhang et al., 2025c). The jaggedness lies not in the fact that behavior changes, but in whether the change tracks the uncertainty structure of the input.

This distinction is important because many AI failures are not failures of knowledge alone. They are failures to recognize what kind of uncertainty is present. A factual question with a knowable answer calls for a different response than a forecast about an unresolved future event, and both differ from a user request that is ambiguous because key preferences or constraints are missing. Uncertainty jaggedness should therefore be evaluated over uncertainty neighborhoods: well-specified inputs, minimally underspecified variants, ambiguous variants, and variants where additional information resolves the uncertainty. The relevant question is whether the model changes strategy in the right way as uncertainty changes.

Evidential Jaggedness Evidential jaggedness occurs when a model's conclusion depends on irrelevant changes in the order, placement, or presentation of evidence. In many reasoning settings, the same evidence should support the same conclusion regardless of sequence. However, both human and model judgments can be shaped by primacy, recency, salience, or positional effects (Hogarth and Einhorn, 1992; Tversky and Kahneman, 1981). For language models, this problem is especially salient in long-context settings, where evidence may be weighted differently depending on where it appears (Liu et al., 2024a; Zhang et al., 2025b).

This type of jaggedness differs from representational jaggedness because the issue is not merely the surface form of a single prompt. The issue is how the model integrates multiple pieces of information over a structured context. A model may over-weight early evidence, ignore information in the middle of a context, or revise too strongly in response to later claims. In such cases, the failure is better described as unstable evidence integration than as generic prompt sensitivity.

Evaluating evidential jaggedness requires perturbations over the evidence structure. The same evidence can be presented in different orders, grouped differently, or separated by distractors, making order sensitivity a specific evaluation target rather than a generic instance of input variation.

Confidence Jaggedness Jaggedness is not only about what a model answers. It can also concern how certain the model appears. Confidence jaggedness occurs when expressed uncertainty varies sharply across nearby cases without corresponding changes in evidence, difficulty, or correctness (Juslin and Olsson, 2007; Xu et al., 2025; Steyvers

et al., 2025). This differs from uncertainty jaggedness: uncertainty jaggedness concerns ambiguity or missing information in the input, whereas confidence jaggedness concerns the model's self-assessment of its output. For example, a model may give the same answer with high confidence under one paraphrase and low confidence under another, or it may be overconfident on outputs that are locally brittle.

This matters because reliability depends on both accuracy and uncertainty estimation (Guo et al., 2017). A model that is accurate on average can still be difficult to use if its confidence does not help users distinguish reliable from unreliable outputs. Conversely, a model that is somewhat less accurate but well calibrated may be easier to integrate into decision-making workflows because users can identify when to trust it, verify it, or seek additional information. Confidence jaggedness should therefore be evaluated locally, not only globally. Aggregate calibration analyses summarize whether confidence tracks accuracy across many examples, but jaggedness requires asking whether confidence behaves appropriately within all relevant neighborhoods (Naeini et al., 2015).

Statistical Jaggedness This occurs when structurally equivalent variants differ because of their frequency in training data, input statistics, distributional familiarity, or output likelihood rather than because of task-relevant differences (McCoy et al., 2019; Gardner et al., 2020). A model may perform well on a common formulation but fail on a rare or out-of-distribution one, even when both require the same rule or support the same answer (Feng et al., 2026). In such cases, the discontinuity reflects the statistical structure of the model's training and generation process rather than the normative structure of the task (Shu and Wei, 2026).

This form of jaggedness is especially relevant for language models because next-token prediction encourages sensitivity to distributional frequency (Bender et al., 2021; Cheng et al., 2025). Such sensitivity is often useful: common patterns may provide reliable cues about meaning, usage, or likely answers. It becomes problematic when common and rare variants are equivalent for the task at hand, such that the same rule, answer type, or decision structure should apply. Evaluating statistical jaggedness requires frequency-controlled neighborhoods that compare familiar and rare variants while preserving the task-relevant structure.

Procedural Jaggedness Procedural jaggedness arises when performance depends on a sequence of decisions and a local bottleneck dominates downstream success (Fan et al., 2025; Sinha et al., 2025; Guo et al., 2026). The issue is not simply that the model lacks a capability, but that reliability varies across procedural roles: retrieval, decomposition, synthesis, verification, execution, or recovery.

This form of jaggedness is especially important for the utility of models when deployed. A system may appear capable when evaluated on isolated subtasks but fail when those subtasks must be sequenced (Mohammadi et al., 2025; Rudd et al., 2025). The capability surface is therefore jagged not only across inputs, but across procedural roles within a task.

Diagnosing procedural jaggedness requires decomposing workflows into decision points and evaluating where errors arise, identifying which step failed, whether the failure propagated, and whether the system had opportunities to recover (Guo et al., 2026; Fan et al., 2025). As such, procedural evaluation resembles a sensitivity analysis over the workflow: perturbing or isolating intermediate steps to identify where jaggedness poses the greatest downstream risk.

It is important to note that other jaggedness sources may also matter in particular situations. For example, policy or alignment jaggedness may arise when safety rules or instruction hierarchies apply inconsistently across nearby cases. The categories above are therefore not a closed ontology, but a starting point for moving from observed instability to mechanism-matched evaluation and design.

5 From Diagnosis to Mechanism-Matched Evaluation and Design

If jaggedness has different sources (refer to Table 2), then evaluation and mitigation cannot rely on a single generic fix. The diagnostic question should therefore precede the intervention question: *what kind of jaggedness is present, where does the system break under targeted variation, and what would count as reducing or containing it?*

Neighborhood-based evaluation Most benchmarks score independent units (Rudd et al., 2025). A neighborhood-based benchmark instead treats a base case and its controlled variants as a single evaluation unit. Operationally, this means constructing benchmark units as neighborhoods with counterfactual variants that instantiate the levels and

mechanisms in Tables 1 and 2. In the clinical-note example, the unit would be the original transcript together with paraphrased, underspecified, and clinically modified variants, rather than one transcript-note pair. A single benchmark item yields one observation and often frames evaluation as confirmatory: the model either appears to have the capability or not. A neighborhood-based analysis is more diagnostic, and in this sense more disconfirmatory or falsification-based, because it asks whether that apparent capability survives nearby counterfactual cases (Klayman and Ha, 1987; Thomas et al., 2008). Rather than treating one failure as decisive, it characterizes the neighborhood of the failure, making the analysis less brittle and less vulnerable to confirmation bias (Thomas et al., 2014).

This changes what counts as success. A model should not receive credit only for answering one version of a problem correctly. If evaluation is neighborhood-based, metrics should report local smoothness alongside aggregate performance. The target is structured smoothness; low variance where the perturbation is irrelevant, calibrated change where uncertainty changes, and predictable sensitivity where decision-relevant structure changes.

Lastly, evaluation should also weigh jaggedness by downstream consequence, not only by frequency (Fan et al., 2025; Guo et al., 2026). A small inconsistency on a low-stakes preference question is different from an inconsistent medical recommendation, legal interpretation, or multi-step tool action. Reporting where jaggedness is concentrated, which discontinuities are consequential, and which are benign can help prioritize interventions.

Mechanism-matched intervention Once a discontinuity has been located, the intervention should target the mechanism rather than the surface symptom. The same output instability may call for different responses depending on whether it reflects unstable representation, poor uncertainty handling, evidence-integration failure, miscalibration, or a procedural bottleneck. Table 2 summarizes these mechanism-matched responses.

The common point is that smoothing is not the same as forcing identical behavior. Paraphrase consistency is appropriate when two prompts preserve the same relevant information, but not when a rewritten prompt changes the evidence/ user constraints. Mechanism-matched intervention requires specifying what should remain stable, what should change, and which component should be modified.

This also makes strategy selection part of the intervention. Many jagged failures arise because the model answers when it should ask, guesses when it should retrieve, gives a single recommendation when it should expose tradeoffs, or proceeds when it should verify an intermediate step. Systems can therefore include local reliability mechanisms for detecting ambiguity, checking evidence, monitoring confidence,or routing high-risk substeps when constraints or blind spots may be missing. Such designs treat jaggedness as a system property, not merely as a model defect.

Monitoring across system changes Deployment should include regression tests over normative neighborhoods, not only aggregate benchmark scores. Such tests can show whether a model update improves topline performance while making behavior more jagged.

Users experience AI systems longitudinally and build expectations about what a system can do, which inputs are safe, and when verification is needed. If updates shift the capability surface unpredictably, users may overgeneralize from previous behavior. When consequential discontinuities remain, the goal may be containment rather than smoothing: isolating brittle workflow steps, requiring verification before action, routing high-risk cases to tools or humans, or warning users when confidence is unreliable.

6 Conclusion

Jaggedness is often treated as uneven capability, but unevenness alone is not the central issue. The more important question is whether model behavior changes in ways justified by the task's structure. This paper reframed jaggedness as a problem of normative local behavior and introduced *normative neighborhoods* as a unit of analysis for answering that question.

The resulting view shifts jaggedness research from cataloging failures to diagnosing them. By separating the levels at which discontinuities appear from the mechanisms that produce them, the framework clarifies what kind of evaluation or design response is needed. Some jaggedness should be smoothened; other forms must be monitored, routed, verified, or contained. A useful theory of jaggedness should therefore tell us not only where models are uneven, but which unevenness matters, why it arises, and what should be done about it in deployment.

7 Limitations and Scope

This paper proposes a conceptual framework rather than a new benchmark, metric, or empirical measurement of jaggedness. Our goal is to define the unit of analysis for model jaggedness. As a result, the framework should be read as a diagnostic vocabulary rather than as a complete operational recipe. One limitation is that normative neighborhoods require task-specific judgments about what should remain invariant, what should vary, and what costs matter. These judgments may be straightforward in settings with formal task definitions, clear labels, or well-specified decision rules. They are more difficult in open-ended, creative, social, or value-laden settings, where multiple reasonable behavioral expectations may coexist. In such cases, neighborhood specifications may reflect expert assumptions, institutional policies, user preferences, or stakeholder values rather than a single objective standard. The framework does not eliminate these disagreements; it makes them visible. Next, we do not expect the various categories in the paper to be exhaustive or mutually exclusive. Representational, uncertainty, evidential, confidence, statistical, and procedural jaggedness often interact in practice. For example, a model's failure under an underspecified prompt may involve both uncertainty handling and confidence calibration, while a workflow failure may be amplified by unstable evidence integration. Other forms of jaggedness, such as policy, alignment, interactional, or multimodal jaggedness, may require additional categories. The proposed taxonomy should therefore be treated as a starting point for diagnosis rather than a closed ontology of model failures. Additionally, operationalizing the framework also introduces measurement challenges. Constructing high-quality neighborhoods requires ensuring that variants differ only in the intended way. Finally, the framework identifies plausible mechanisms of jaggedness, but it does not by itself establish causal explanations. Confirming the exact mechanism may require additional ablations, access to system components, or controlled deployment studies. Mechanism-matched intervention should therefore be understood as a research agenda rather than a guaranteed mitigation procedure. Despite these limitations, we argue that explicitly specifying normative neighborhoods is preferable to treating all jaggedness as generic brittleness. The framework is most useful when its assumptions are stated clearly: what neighborhood is being tested, what behavioral relation is expected, whose normative standard is being used, and what downstream cost makes the discontinuity matter.

References

Haoyue Bai, Yiyou Sun, Wenjie Hu, Shi Qiu, Maggie Ziyu Huan, Peiyang Song, Robert Nowak, and Dawn Song. 2025. How and why llms generalize: A fine-grained analysis of llm reasoning from cognitive behaviors to low-level patterns. *arXiv preprint arXiv:2512.24063*.

David Bani-Harouni, Chantal Pellegrini, Paul Stangel, Ege Özsoy, Kamilia Zaripova, Nassir Navab, and Matthias Keicher. 2025. Rewarding doubt: A reinforcement learning approach to calibrated confidence expression of large language models. In *The Fourteenth International Conference on Learning Representations*.

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623.

Colin Campbell, Sean Sands, Lucas Whittaker, and Alexis Mavrommatis. 2025. The ai intelligence playbook: Decoding genai capabilities for strategic advantage. *Business horizons*.

Liang Cheng, Zhaowei Wang, and Mark Steedman. 2025. Llms are frequency pattern learners in natural language inference. *arXiv preprint arXiv:2505.21011*.

Qinyuan Cheng, Tianxiang Sun, Xiangyang Liu, Wenwei Zhang, Zhangyue Yin, Shimin Li, Linyang Li, Zhengfu He, Kai Chen, and Xipeng Qiu. 2024. Can ai assistants know what they don't know? In *Proceedings of the 41st International Conference on Machine Learning*, pages 8184–8202.

Francois Chollet, Mike Knoop, Gregory Kamradt, Bryan Landers, and Henry Pinkard. 2025. Arc-agi-2: A new challenge for frontier ai reasoning systems. *arXiv preprint arXiv:2505.11831*.

Fabrizio Dell'Acqua, Edward McFowland III, Ethan R Mollick, Hila Lifshitz-Assaf, Katherine Kellogg, Saran Rajendran, Lisa Krayer, François Candelon, and Karim R Lakhani. 2023. Navigating the jagged technological frontier: Field experimental evidence of the effects of ai on knowledge worker productivity and quality. *Harvard business school technology & operations mgt. Unit working paper*, (24-013).

Hillel J Einhorn and Robin M Hogarth. 1986. Decision making under ambiguity. *Journal of business*, pages S225–S250.

Shengda Fan, Xin Cong, Yuepeng Fu, Zhong Zhang, Shuyan Zhang, Yuanwei Liu, Yesai Wu, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2025. Workflowllm: Enhancing workflow orchestration capability of large language models. In *International Conference on Learning Representations*, volume 2025, pages 24498–24525.

Dylan Feng, Pragya Srivastava, and Cassidy Laidlaw. 2026. Benchmarking and improving monitors for out-of-distribution alignment failure in llms. *Preprint*, arXiv:2605.21602.

Craig R Fox and Gülden Ülkümen. 2011. Distinguishing two dimensions of uncertainty. *Fox, Craig R. and Gülden Ülkümen (2011),"Distinguishing Two Dimensions of Uncertainty," in Essays in Judgment and Decision Making, Brun, W., Kirkebøen, G. and Montgomery, H., eds. Oslo: Universitetsforlaget*.

Giorgio Franceschelli and Mirco Musolesi. 2025. On the creativity of large language models. *AI & society*, 40(5):3785–3795.

Michael C Frank and Noah D Goodman. 2025. Cognitive modeling using artificial intelligence. *Annual Review of Psychology*, 77.

Joshua S Gans. 2026. A model of artificial jagged intelligence. Technical report, National Bureau of Economic Research.

Matt Gardner, Yoav Artzi, Victoria Basmov, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hannaneh Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, and 7 others. 2020. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1307–1323, Online. Association for Computational Linguistics.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. 2017. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR.

Yaxin Guo, Hongye Tan, Ru Li, Xiaoli Li, Xinyi Sun, Pengpeng Qiang, and Hu Zhang. 2026. Problem decomposition guided by reasoning utility for complex reasoning in llms. *Information Processing & Management*, 63(3):104509.

Robin M Hogarth and Hillel J Einhorn. 1992. Order effects in belief updating: The belief-adjustment model. *Cognitive psychology*, 24(1):1–55.

Jaap Jumelet, Leonie Weissweiler, Joakim Nivre, and Arianna Bisazza. 2026. Multiblimp 1.0: A massively multilingual benchmark of linguistic minimal pairs. *Transactions of the Association for Computational Linguistics*, 14:193–216.

Peter Juslin and Henrik Olsson. 2007. Computational models of subjective probability calibration. In *Judgment and Decision Making*, pages 67–95. Psychology Press.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, and 1 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Nancy S Kim. 2025. *Judgment and Decision-making: In the Lab and the World*. Bloomsbury Publishing.

Samuel N Kirshner, Yiwen Pan, and Jason Xianghua Wu. 2025. Prosocial when simple and cold-hearted when complex: How task difficulty shapes llm behavior. *Decision Analysis*.

Joshua Klayman and Young-Won Ha. 1987. Confirmation, disconfirmation, and information in hypothesis testing. *Psychological review*, 94(2):211.

Harsh Nishant Lalai, Raj Sanjay Shah, Jiaxin Pei, Sashank Varma, Yi-Chia Wang, and Ali Emami. 2025. The world according to llms: How geographic origin influences llms' entity deduction capabilities. *arXiv preprint arXiv:2508.05525*.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024a. Lost in the middle: How language models use long contexts. *Transactions of the association for computational linguistics*, 12:157–173.

Qin Liu, Fei Wang, Nan Xu, Tianyi Lorena Yan, Tao Meng, and Muhao Chen. 2024b. Monotonic paraphrasing improves generalization of language model prompting. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 9861–9877.

Sravanthi Machcha, Sushrita Yerra, Sahil Gupta, Aishwarya Sahoo, Sharmin Sultana, Hong Yu, and Zonghai Yao. 2026. Knowing when to abstain: Medical llms under clinical uncertainty. *arXiv preprint arXiv:2601.12471*.

Shray Mathur, J. Anibal Boscoboinik, Esther H. R. Tsai, and Kevin G. Yager. 2026. Llm jaggedness unlocks scientific creativity. *Preprint*, arXiv:2605.10574.

R. Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Patrick J Mineault, Thomas L Griffiths, and Sean Escola. 2026. Cognitive dark matter: Measuring what ai misses. *arXiv preprint arXiv:2603.03414*.

Mahmoud Mohammadi, Yipeng Li, Jane Lo, and Wendy Yip. 2025. Evaluation and benchmarking of llm agents: A survey. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 6129–6139.

Meredith Ringel Morris, Dan Altman, Haydn Belfield, Arthur Goemans, Hasan Iqbal, Ryan Burnell, Iason Gabriel, Samuel Albanie, and Allan Dafoe. 2026. Characterizing model jaggedness supports safety and usability.

Mahdi Pakdaman Naeini, Gregory Cooper, and Milos Hauskrecht. 2015. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29.

Pouya Pezeshkpour and Estevam Hruschka. 2024. Large language models sensitivity to the order of options in multiple-choice questions. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2006–2017.

Amirhossein Razavi, Mina Soltangheis, Negar Arabzadeh, Sara Salamat, Morteza Zihayat, and Ebrahim Bagheri. 2025. Benchmarking prompt sensitivity in large language models. In *European Conference on Information Retrieval*, pages 303–313. Springer.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of nlp models with checklist. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 4902–4912.

Daniel Romero-Alvarado, Fernando Martínez-Plumed, Lorenzo Pacchiardi, Hugo Save, Siddhesh Milind Pawar, Behzad Mehrbakhsh, Pablo Antonio Moreno Casares, Ben Slater, Paolo Bova, Peter Romero, and 1 others. 2026. Capabilities ain't all you need: Measuring propensities in ai. *arXiv preprint arXiv:2602.18182*.

Ethan M Rudd, Christopher Andrews, and Philip Tully. 2025. A practical guide for evaluating llms and llm-reliant systems. *arXiv preprint arXiv:2506.13023*.

Manish Sanwal. 2024. Evaluating large language models using contrast sets: An experimental approach. *arXiv preprint arXiv:2404.01569*.

Leonard J Savage. 2012. *The foundations of statistics*. Courier Corporation.

Raj Sanjay Shah, Jing Huang, Keerthiram Murugesan, Nathalie Baracaldo, and Diyi Yang. 2026a. The unlearning mirage: A dynamic framework for evaluating llm unlearning. *arXiv preprint arXiv:2603.11266*.

Raj Sanjay Shah, Alex Warstadt, Michael Frank, and Sashank Varma. 2026b. On the use of pre-trained language models in cognitive science. *Under Review*.

Raj Sanjay Shah, Lei Xu, Qianchu Liu, Jon Burnsky, Andrew Bertagnolli, and Chaitanya Shivade. 2025. TN-eval: Rubric and evaluation protocols for measuring the quality of behavioral therapy notes. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, pages 179–199, Vienna, Austria. Association for Computational Linguistics.

Wesley Shu and Peng Wei. 2026. Artificial jagged intelligence as uneven optimization energy allocation capability concentration, redistribution, and optimization governance. *arXiv preprint arXiv:2605.01420*.

Akshit Sinha, Arvindh Arun, Shashwat Goel, Steffen Staab, and Jonas Geiping. 2025. The illusion of diminishing returns: Measuring long horizon execution in llms. *arXiv preprint arXiv:2509.09677*.

Peiyang Song, Pengrui Han, and Noah Goodman. 2025. A survey on large language model reasoning failures. In *2nd AI for Math Workshop@ ICML 2025*.

Peiyang Song, Pengrui Han, and Noah Goodman. 2026. Large language model reasoning failures. *arXiv preprint arXiv:2602.06176*.

Mark Steyvers, Heliodoro Tejeda, Aakriti Kumar, Catarina Belem, Sheer Karny, Xinyue Hu, Lukas W Mayer, and Padhraic Smyth. 2025. What large language models know and what people think they know. *Nature Machine Intelligence*, 7(2):221–231.

Rick Thomas, Michael R Dougherty, and Daniel R Buttaccio. 2014. Memory constraints on hypothesis generation and decision making. *Current Directions in Psychological Science*, 23(4):264–270.

Rick P Thomas, Michael R Dougherty, Amber M Sprenger, and J Harbison. 2008. Diagnostic hypothesis generation and human judgment. *Psychological review*, 115(1):155.

Amos Tversky and Daniel Kahneman. 1981. The framing of decisions and the psychology of choice. *science*, 211(4481):453–458.

Konstantinos Voudouris, Mirko Thalmann, Alex Kipnis, José Hernández-Orallo, and Eric Schulz. 2026. Measuring what ai systems might do: Towards a measurement science in ai. *arXiv preprint arXiv:2603.00063*.

Abraham Wald. 1950. Statistical decision functions. In *Breakthroughs in Statistics: Foundations and Basic Theory*, pages 342–357. Springer.

Jie JW Wu. 2023. Large language models should ask clarifying questions to increase confidence in generated code. *arXiv preprint arXiv:2308.13507*.

Jiancong Xiao, Bojian Hou, Zhanliang Wang, Ruochen Jin, Qi Long, Weijie J Su, and Li Shen. 2025. Restoring calibration for aligned large language models: A calibration-aware fine-tuning approach. In *International Conference on Machine Learning*, pages 68364–68390. PMLR.

Chenjun Xu, Bingbing Wen, Bin Han, Robert Wolfe, Lucy Lu Wang, and Bill Howe. 2025. Do language models mirror human confidence? exploring psychological insights to address overconfidence in llms. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 25655–25672.

Alex L Zhang, Tim Kraska, and Omar Khattab. 2025a. Recursive language models. *arXiv preprint arXiv:2512.24601*.

Junhao Zhang, Richong Zhang, Fanshuang Kong, Ziyang Miao, Yanhan Ye, and Yaowei Zheng. 2025b. Lost-in-the-middle in long-text generation: Synthetic dataset, evaluation framework, and mitigation. *arXiv preprint arXiv:2503.06868*.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, and 1 others. 2025c. Siren's song in the ai ocean: A survey on hallucination in large language models. *Computational Linguistics*, 51(4):1373–1418.

Yue Zhou, Yada Zhu, Diego Antognini, Yoon Kim, and Yang Zhang. 2024. Paraphrase and solve: Exploring and exploiting the impact of surface form on mathematical reasoning in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2793–2804.