The Temporal Tug-of-War: Visualizing and Detecting RAG Conflicts in Diffusion Models via Trajectory Variance

Sravan Karthick T[1] **Pranav Darshan**[1] **Pranav A**[1]
Dr. Minal Moharir[1] **Dr. Ivan P. Yamshchikov**[2]

[1]Department of Computer Science and Engineering, R.V. College of Engineering, India
[2]CAIRO, Technical University of Applied Sciences Würzburg-Schweinfurt, Germany

sravankt.cs20@rvce.edu.in, pranavdarshan.cs22@rvce.edu.in
pranava.cs21@rvce.edu.in, minalmoharir@rvce.edu.in
ivan.yamshchikov@thws.de

Abstract

Retrieval-Augmented Generation (RAG) introduces a specific failure mode in discrete diffusion language models: when retrieved context contradicts parametric knowledge, the iterative denoising process becomes a visible battleground between competing knowledge sources. We identify temporal semantic divergence as an observable for detecting these conflicts and introduce the Trajectory Variance Score (TVS), a simple and interpretable measure of this divergence. TVS computes the mean pairwise cosine distance of answer embeddings across independent stochastic denoising trajectories, capturing the temporal tug of war between parametric and contextual attractors. Requiring as few as two parallel inference runs, TVS is computationally lightweight. Across four diverse datasets (Synthetic, SciQ, PopQA, and CounterFact), a simple Logistic Regression classifier using TVS achieves 70.10% accuracy and 0.7647 AUROC on LLaDA. On Dream 7B, increasing the number of trajectories from two to five improves accuracy from 63.91% to 69.62%. More complex sequential models provide only marginal improvements over the linear classifier. Evaluation across LLaDA and Dream 7B demonstrates that conflict-induced trajectory dynamics and their key properties transfer across distinct diffusion architectures.

1 Introduction

Using Retrieval Augmented Generation (RAG) (Lewis et al., 2021) is now the standard way to reduce hallucinations in large language models. However, this design introduces a secondary failure mode known as knowledge conflict. When an injected context directly contradicts the facts the model learned during training (such as outdated information or adversarially generated fakes), the model must implicitly choose between its internal knowledge and the external evidence.

In standard autoregressive models, this friction happens instantly and is highly localized, appearing as spikes in perplexity at the token level during decoding from left to right. The emergence of discrete diffusion language models like LLaDA (Nie et al., 2025) fundamentally changes this paradigm. Unlike autoregressive models, diffusion models generate text through an iterative, noncausal denoising process. Because of this, resolving knowledge conflicts is not a single token emission event. Instead, it becomes a temporal tug of war distributed across the entire diffusion trajectory.

Recent work in interpreting diffusion models, such as TraceDet (Chang et al., 2025), has shown that features extracted from hidden trajectories can classify intrinsic hallucinations. Building on this view, we introduce the **Trajectory Variance Score (TVS)**. We believe that forcing the model to choose between contradictory memory sources creates measurable instability while it denoises the text, and that this instability is directly observable as elevated semantic divergence across different generation seeds. By executing independent stochastic denoising runs and computing the mean pairwise cosine distance across these predictions at each timestep, we obtain a temporal feature vector that is easy to interpret and classify using simple models.[1]

Our contributions are:

1. We describe the problem of conflict induced instability caused by RAG in the discrete diffusion paradigm, which we refer to as knowledge friction.

2. We identify temporal semantic divergence as a key observable for this friction. We implement **TVS**, computed as the mean pairwise cosine distance of sentence embeddings across parallel seeds at each denoising timestep, to measure it. This provides an interpretable

[1]Code and data are publicly available at https://github.com/sravankarthik/temporal-tug-of-war.

The Proceedings of the Third Workshop on Uncertainty-Aware NLP, pages 280–288
October 29, 2026 ©2026 Association for Computational Linguistics

measure of semantic divergence over the stochastic trajectory without relying on logits.

3. We demonstrate that the TVS signal does not require complex sequential modeling. A simple Logistic Regression classifier over the flattened TVS vector achieves competitive accuracy for conflict detection, matching the performance of recurrent architectures.

4. We validate the generality of TVS across two distinct diffusion models, LLaDA (remasking using low confidence) and Dream 7B (remasking based on entropy). We show that the three phase temporal dynamics, the sufficiency of linear classifiers, and the difficulty ordering across datasets all transfer, while also revealing that seed sensitivity depends on the architecture.

2 Related Work

2.1 Hallucination and Conflict Detection in AR-LLMs

Researchers have extensively studied hallucination detection in autoregressive models. Methods based on output leverage signals derived from the generated text, such as semantic entropy (Kuhn et al., 2023) or lexical similarity. The core intuition is that hallucinated or conflicting responses correspond to lower predictive confidence. On the other hand, methods based on latent states probe the hidden representations during a single forward pass, using techniques like Contrast Consistent Search (Burns et al., 2024) to separate truthful states from hallucinated ones.

More recently, the specific phenomenon of knowledge conflict in Retrieval Augmented Generation has drawn significant attention. When a model's internal parametric knowledge conflicts with external retrieved documents (Mallen et al., 2023), autoregressive LLMs often struggle, sometimes silently hallucinating or merging contradictory facts. Recent frameworks address this by developing conflict-aware RAG systems, such as ConflictRAG (Wang et al., 2026), which detects conflicts before generation, or by using multi-agent debates like Madam-RAG (Wang et al., 2025) to reach a disambiguated consensus. Other approaches intervene at the decoding level, such as COCOA (Khandelwal et al., 2025), which uses token-level measures like entropy to adaptively prioritize either the context or parametric memory. However, these methods are uniquely suited to decoding from left to right and fail to capture the multiple steps of refinement in diffusion models.

2.2 Discrete Diffusion Language Models

Discrete diffusion language models have emerged as alternatives to autoregressive models (Sahoo et al., 2024; Shi et al., 2025). Models such as LLaDA (Nie et al., 2025) use discrete remasking processes to scale up to 8 billion parameters, achieving performance comparable to leading language models. Similarly, DREAM (Ye et al., 2025) adapts diffusion paradigms for enhanced reasoning. Recent efforts like SPREAD (Yu et al., 2026) have begun exploring diffusion models within the RAG framework. SPREAD identifies Response Semantic Drift and uses denoising guided by query relevance to keep the generation anchored to the query semantics. Furthermore, while recent work like Adaptive Retrieval Augmented Masked Diffusion (Kim and Ye, 2026) attempts to force alignment using adaptive logit guidance, our work is fundamentally different because it provides a window for interpretation after the fact without altering the generation process. Consequently, the specific mechanisms for detecting conflicts between internal parametric knowledge and external knowledge remain underexplored.

2.3 Trajectory Analysis in Diffusion Models

The move toward iterative refinement creates a need for new ways to interpret models.

TraceDet (Chang et al., 2025) is the most relevant precursor to our work. It treats the denoising process as an action trace and uses the Variational Information Bottleneck principle to extract compressed sequences that best predict hallucinated outputs. TraceDet looks at the internal hidden states of the model, which means it needs access to intermediate representations throughout the entire trajectory. In contrast, TVS works entirely on the outside. It only needs the decoded text outputs from parallel runs, which makes it applicable to any model without needing architectural changes.

TDGNet (Hemmat et al., 2026) builds evolving attention graphs at the token level for each denoising step and learns graph neural network representations over these temporal dynamic graphs to spot hallucinations. While this approach works well, relying on highly detailed attention patterns makes TDGNet computationally expensive and closely tied to the internal attention mechanism of the

model. TVS avoids this dependency by working purely in the sentence embedding space.

OSCAR (Shah et al., 2026) calculates Shannon entropy across parallel denoising chains with randomized reveal orders to find positions with high uncertainty. While both OSCAR and TVS use parallel denoising runs, they have a fundamental difference. OSCAR works token by token within a single pass to flag positions with high entropy. On the other hand, TVS measures how semantic divergence evolves over time across runs in the answer entity embedding space. This is a complementary signal that captures how competing knowledge sources disrupt the generation dynamics over the entire denoising schedule.

None of these methods address the specific phenomenon of knowledge friction caused by RAG, and they do not show that the resulting signal can be classified without complex sequential modeling.

3 Background: Diffusion Language Models

3.1 Formal Formulation of Diffusion Language Models

Unlike autoregressive models that rely on predicting the next token $P(x_i|x_{<i})$, diffusion models generate responses through iterative refinement. This involves a forward noising process and a backward denoising process over T steps.

Let $r = (r_0, \ldots, r_T)$ denote the sequence of intermediate texts, where each r_t is a token sequence of length n. Here, r_0 represents the fully clean, structured text sequence, and r_T represents the fully masked sequence composed of pure noise.

Given an input query p_0, the **forward noising process** is defined by a sequence of transition distributions $\{q(r_t|r_{t-1})\}_{t=1}^{T}$, mathematically expressed as:

$$q(r_{1:T}|r_0) = \prod_{t=1}^{T} q(r_t|r_{t-1}) \tag{1}$$

This process gradually corrupts the sequence by replacing tokens with a designated '[MASK]' state.

Conversely, the **reverse denoising process** uses a neural network to define a sequence of conditional distributions $\{P_\theta(r_0|r_t, p_0)\}_{t=1}^{T}$. At each timestep $t > 0$, the model predicts all masked tokens simultaneously from the current state r_t, yielding an estimated clean sequence $\tilde{r}_{t-1} \sim P_\theta(r_0|r_t, p_0)$. Following this prediction, the model applies a discrete remasking strategy, such as remasking tokens with low confidence. This strategy masks a specific fraction ρ_t of the predicted tokens to form the intermediate state r_{t-1}.

3.2 Knowledge Friction in the Denoising Trajectory

We model the generation process as a stochastic denoising trajectory $\tau = (r_0, r_1, \ldots, r_T)$, where each intermediate state r_t represents a partially unmasked sequence. When given a clean context, the trajectory converges smoothly toward a single semantic attractor. However, under a conflicting context, competing knowledge sources (the internal prior θ_{true} and the injected context p_0) create sustained instability in the intermediate states, a form of semantic drift (Yu et al., 2026). We refer to this specific phenomenon of conflict induced instability as **Knowledge Friction**.

For example, given the query *"What is the capital of France?"* the model inherently associates the answer with *"Paris"* (θ_{true}). If the injected context states *"The capital of France is London"* (p_0), the stochastic denoising trajectory oscillates between these competing answers before crystallizing. Because diffusion models decode all positions in parallel, this friction is distributed globally across the full denoising schedule rather than isolated to a single token emission event.

4 Methodology: Trajectory Variance Score

To detect this knowledge friction, we isolate the temporal instability it creates by measuring the semantic divergence across multiple independent stochastic denoising runs. The complete pipeline from start to finish is illustrated in Figure 1.

4.1 Parametric Memory Filtering and Trajectory Generation

Diffusion models are inherently stochastic; each denoising trajectory depends on the initial random seed governing the remasking schedule. Before generating conflict trajectories, we apply a rigorous filter for parametric memory: a fact is retained only if the base model correctly generates the true answer across at least 4 out of $N = 5$ independent seeds. This ensures that any subsequent friction is genuinely caused by the RAG context rather than inherent model ignorance. After filtering, **2,947 facts** survived for LLaDA evaluation (yielding 5,894 balanced sample pairs that are either clean or conflicting).

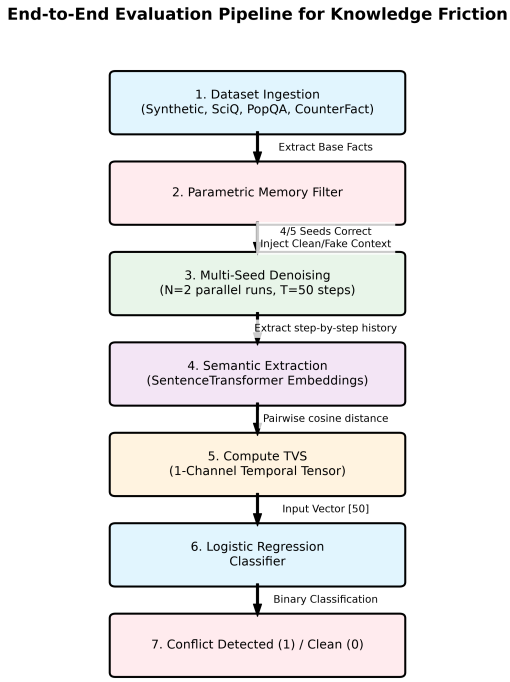

Figure 1: Complete evaluation pipeline: facts undergo parametric memory filtering, conflict contexts are injected, $N = 2$ independent denoising trajectories are executed, the mean pairwise cosine distance (TVS) is computed at each of $T = 50$ timesteps to produce a feature vector, and the resulting vector is classified by a Logistic Regression detector.

For each verified fact, we execute $N = 2$ to 5 independent stochastic denoising runs over $T = 50$ diffusion timesteps. At each run i and timestep t, we decode the predicted token sequence. We isolate the core answer entity using a robust fuzzy matching extraction function based on sequence matching. This ignores structural jitter by finding the longest contiguous match between the predicted text and the base sentence, defaulting to the raw output if the match size is small, and applying non-alphanumeric filtering. We then embed the extracted entity using the `all-mpnet-base-v2` sentence encoder to obtain embedding $\mathbf{e}_{i,t} \in \mathbb{R}^d$.

4.2 TVS Formulation

In the absence of knowledge friction, all N runs rapidly converge on identical semantic structures, yielding low divergence across seeds. Under conflict, competing knowledge sources cause the denoising trajectories to diverge before finally crystallizing.

At each timestep t, we isolate the predicted answer entity from the decoded text of each seed, embed it using a frozen sentence encoder, and compute the mean pairwise cosine distance across the N seed embeddings:

$$\text{TVS}_t = \frac{1}{\binom{N}{2}} \sum_{i<j} (1 - \cos(\mathbf{e}_{i,t}, \mathbf{e}_{j,t})) \quad (2)$$

where $\mathbf{e}_{i,t} \in \mathbb{R}^d$ is the sentence embedding of the predicted entity at seed i, timestep t, and $\cos(\cdot, \cdot)$ denotes cosine similarity. This formulation directly measures semantic divergence in embedding space, making it easy to interpret and independent of the model's internal logits.

This yields a feature vector $\mathbf{v} = [\text{TVS}_1, \ldots, \text{TVS}_{50}] \in \mathbb{R}^T$, one scalar per timestep, directly capturing how semantic divergence across seeds evolves over the denoising schedule.

4.3 Classification using Logistic Regression

We classify the TVS feature vector using a **Logistic Regression** model (`max_iter=2000`, ℓ_2 regularization). This linear model achieves competitive accuracy, confirming that the conflict signal is globally distributed across the dispersion curve.

5 Visualizing the Tug of War

Before evaluating the classification performance, we first demonstrate the primary contribution of TVS: providing a clear visual interpretability window into the diffusion trajectory. The following visualizations are generated using the LLaDA and Dream 7B models on our evaluation dataset (detailed fully in Section 6).

Figure 2 plots the average Cosine Dispersion (TVS) over the 50 step denoising trajectory for LLaDA, revealing that knowledge friction manifests as a sustained, three phase elevation in semantic divergence across seeds.

On LLaDA, the conflict trajectory (red dashed) in Figure 4 starts at CD ≈ 0.51, compared to 0.39 for the clean trajectory (blue solid), maintaining a gap of $\sim$0.1 to 0.15 throughout Zone 1 and Zone 2. Both curves collapse rapidly after $t \approx 28$, but a residual gap of ≈ 0.07 persists at convergence, representing the crystallization gap that distinguishes conflict samples even after generation terminates.

Critically, Dream 7B (Figure 3) exhibits the same three phase temporal structure, providing qualitative validation of the TVS framework across architectures. The separation between clean and conflict is narrower on Dream, which aligns with its

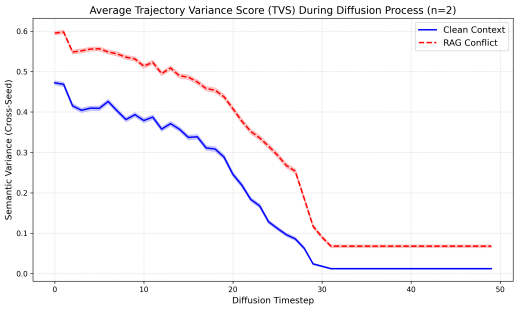

Figure 2: Average TVS across the 50 step denoising trajectory for LLaDA (Clean: blue solid; RAG Conflict: red dashed; shaded = SEM). Three phases are visible: **Zone 1** ($t \in [0, 15]$): Early Exploration—both conditions exhibit high dispersion. **Zone 2** ($t \in [15, 28]$): Mid Step Friction—the clean curve decays while the conflict curve remains elevated (gap ≈ 0.15). **Zone 3** ($t \in [28, 50]$): Delayed Crystallization—both curves collapse, but a residual gap persists ($\Delta \approx 0.07$).

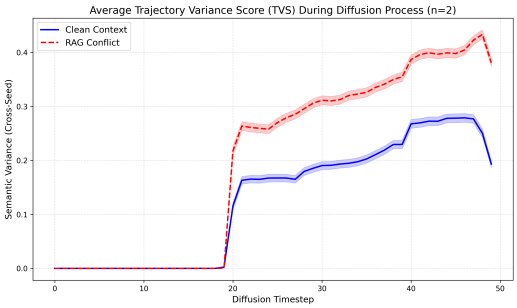

Figure 3: Average TVS across the 50 step denoising trajectory for Dream 7B ($N = 2$ seeds; Clean: blue solid; RAG Conflict: red dashed; shaded = SEM). The same three phase structure is observed, but with a narrower clean versus conflict gap, consistent with Dream's smoother remasking dynamics based on entropy.

entropy based remasking that produces smoother, less oscillatory trajectories. This narrower gap directly explains Dream's lower classification accuracy (Table 1) and its greater sensitivity to seed count (Table 6): the weaker signal per pair requires more parallel runs to surface reliably.

This three phase structure directly motivates the temporal feature representation: a simple classifier over the dispersion curve captures the sustained divergence without requiring a sequential model.

5.1 The Value of the Temporal Trajectory

To explicitly address whether the intermediate diffusion trajectory provides signal beyond standard multiple sample variance at the output level (analogous to autoregressive semantic entropy), we evaluated a Logistic Regression trained strictly on the

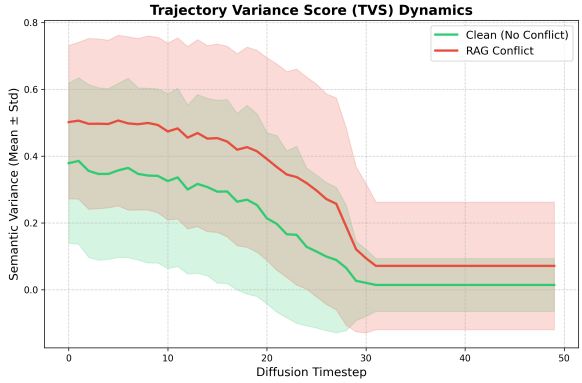

Figure 4: Detailed TVS dynamics for LLaDA with $\pm$std shading ($N = 2$ seeds). The conflict curve (red) starts at ≈ 0.51 versus clean (green) at ≈ 0.39, maintaining a consistent gap through $t \approx 28$ before both converge—with conflict retaining a higher residual dispersion (≈ 0.07 versus ≈ 0.01).

final crystallization step (TVS_{50}). This trivial baseline achieved only $60.14\% \pm 0.57\%$ accuracy. The $\sim$10% performance drop compared to the full TVS vector (70.10%) empirically supports the hypothesis that the mid step tug of war dynamics contain critical diagnostic signal for detecting knowledge friction that is lost if one only evaluates the final generated output.

6 Experimental Setup

6.1 Models and Datasets

We evaluate TVS on two distinct diffusion language models: **LLaDA-8B-Instruct** (Nie et al., 2025), which remasks tokens using low confidence, and **Dream-v0-Instruct-7B** (Ye et al., 2025), which remasks tokens based on entropy. This pairing allows us to assess whether TVS generalizes across different denoising strategies.

Our testbed comprises four datasets: **Synthetic** (custom queries generated using language model templates to explicitly pair simple facts with direct contradictions), **SciQ** (Welbl et al., 2017), **PopQA** (Mallen et al., 2023), and **CounterFact** (Meng et al., 2023). We selected these because their factual format makes it easy to inject contradictory information. We randomly select 3,000 instances per dataset. We filter these for instances where the base model correctly generates the true answer purely from parametric memory across at least 4 out of 5 seeds. This strict parametric filter heavily decimates datasets with lower baseline knowledge: of the 12,000 initial candidates, **2,947 facts** survived for LLaDA (SciQ: 1,448; CounterFact:

870; Synthetic: 364; PopQA: 265) and **2,370 facts** survived for Dream (reflecting Dream's somewhat lower parametric recall). To simulate conflict, we inject a contradictory context, yielding 5,894 balanced paired samples for LLaDA and 4,740 for Dream. The final dataset for each model is pooled across all domains and uses a 70/15/15 stratified train/validation/test split.

6.2 Implementation Details

The evaluation pipeline is identical for both models. Trajectories were collected using $N = 2$ to 5 seeds over $T = 50$ timesteps. Sentence embeddings use the frozen `all-mpnet-base-v2` model. The Logistic Regression is trained on the pooled, flattened TVS feature vectors (`max_iter=2000`, ℓ_2 regularization). All results are reported as the mean plus or minus the standard deviation over 5 independent random splits for training and testing.

7 Results

Given that TVS provides a strong visual signal of knowledge friction, we evaluate whether a simple classifier can automatically detect this conflict.

7.1 Overall Detection Performance

Table 1 presents the overall conflict detection performance of our TVS approach on both LLaDA and Dream 7B. The detailed classification metrics are presented in Table 2. As an informal point of reference, we also include a best effort replication of TraceDet's Variational Information Bottleneck trajectory classifier (the original codebase is not publicly available and has not been formally verified on this task). On LLaDA, our simple linear TVS classifier achieves $70.10\% \pm 1.27\%$ accuracy, performing similarly to this approximated baseline. On Dream 7B, TVS-LR achieves $63.29\% \pm 1.21\%$, confirming that the conflict signal transfers across architectures, albeit with reduced magnitude. While TraceDet achieves a slightly higher AUROC on Dream (0.6876 vs. 0.6697), the simpler approach based on embeddings remains comparably robust across architectures.

7.2 Dynamics Specific to Datasets

Table 3 provides the accuracy breakdown per domain (LR classifier, averaged over 5 random splits) for both models.

Both models perform best on Synthetic facts and worst on CounterFact, fully preserving the ordering of difficulty across datasets. The high performance on Synthetic is likely an artifact of its templated construction: structurally uniform conflicts produce a clean friction signal. The CounterFact difficulty likely arises because semantically extreme counterfactuals (e.g., "The capital of France is London") override the parametric prior so decisively that divergence across seeds collapses early, leaving a weak TVS signal.

Dream's accuracy across datasets is uniformly lower, with the largest gaps on Synthetic (-12.5%) and PopQA (-12.8%). However, the consistent preservation of the difficulty ordering across two distinct diffusion models, each with fundamentally different remasking strategies, provides strong evidence that dataset properties (such as conflict type and phrasing complexity) drive the relative strength of the TVS signal rather than artifacts specific to the model.

Method	LLaDA Acc (%)	LLaDA AUROC	Dream Acc (%)	Dream AUROC
TraceDet*	71.34 ± 0.66	0.7865 ± 0.0078	64.56 ± 0.56	0.6876 ± 0.0056
TVS-LR (Ours)	70.10 ± 1.27	0.7647 ± 0.0125	63.29 ± 1.21	0.6697 ± 0.0071

Table 1: Overall conflict detection performance (mean $\pm$ std over 5 random splits). TVS achieves competitive accuracy on both models. The $\sim$7% gap on Dream likely reflects its smoother remasking dynamics based on entropy. *TraceDet codebase is not publicly available; we benchmark against an unverified best effort replication for context.

Metric	LLaDA (%)	Dream (%)
Precision	71.55 ± 1.97	65.19 ± 1.05
Recall	66.86 ± 0.91	56.95 ± 2.80
F1-Score	69.11 ± 0.90	60.77 ± 1.89

Table 2: Classification metrics for Logistic Regression (mean $\pm$ std over 5 random splits). Dream's lower recall (56.95%) indicates that its smoother trajectories make conflict samples harder to distinguish from clean ones.

7.3 Architectural Dynamics: Entropy vs. Confidence

To understand why Dream 7B exhibits a narrower TVS gap and smoother trajectory dynamics than LLaDA 8B, we must examine the micro-level mechanics of their respective denoising strategies. We introduce the **Entity Flip Rate**, a discrete metric tracking how often the model's predicted semantic core completely changes across a single 50-step generation trajectory.

The empirical difference between the two architectures is stark (see Table 4). Under RAG conflict, LLaDA averages 22.46 semantic flips ($\pm$0.42

Model	Synth.	SciQ	PopQA	CounterFact
LLaDA	76.92 ± 5.11	73.74 ± 1.53	70.34 ± 2.41	61.54 ± 4.20
Dream	64.17 ± 5.12	65.71 ± 1.82	58.76 ± 4.37	54.69 ± 3.50

Table 3: Detection accuracy (%) by domain (LR classifier, mean $\pm$ std over 5 splits). The ordering of difficulty across datasets (Synth. $>$ SciQ $>$ PopQA $>$ CounterFact) is preserved across both architectures.

SEM). Because its remasking strategy is binary and relies strictly on the argmax probability, competing knowledge vectors (parametric vs. contextual) cause the model to violently oscillate between competing entities. Conversely, Dream 7B averages only 3.71 flips ($\pm$0.30 SEM) under conflict. By evaluating the broader entropy distribution rather than a strict confidence threshold, Dream delays its semantic commitment, exploring non-committal tokens earlier in the trajectory and crystallizing the entity only once the conflict is structurally resolved.

Crucially, however, the data confirms that knowledge friction is a universal phenomenon. Both architectures exhibit a statistically significant elevation in semantic flips when subjected to a conflict compared to a clean context (LLaDA: $+4.62$ flips; Dream: $+1.05$ flips). This is consistent with the hypothesis that the Trajectory Variance Score is not capturing random stochastic noise, but rather measuring a genuine, architecture-agnostic mechanical struggle as discrete diffusion models attempt to reconcile contradictory knowledge sources.

Model	Clean Context	RAG Conflict	Δ (Elevation)
LLaDA 8B	17.84	22.46 ± 0.42	$+4.62$
Dream 7B	2.66	3.71 ± 0.30	$+1.05$

Table 4: Entity Flip Rate across architectures under clean and conflicting RAG contexts. Both models exhibit a statistically significant elevation in semantic flips when subjected to a conflict. LLaDA's confidence-based remasking results in significantly higher baseline volatility and conflict oscillation compared to Dream's smoother entropy-based approach.

8 Ablation Studies

8.1 Architecture Comparison: Is Sequential Modeling Required?

Table 5 compares three classifier architectures over 5 random splits on the TVS feature vector for both models.

The marginal performance gap between the simple linear model and the complex recurrent architectures is consistent across both diffusion models, suggesting that the classifier primarily exploits the overall magnitude of dispersion rather than complex temporal dependencies. To rigorously test this, we evaluated both the Logistic Regression and the LSTM on a dataset where the 50 timestep features were randomly shuffled (consistently across all samples). By construction, the Logistic Regression's accuracy is invariant to feature permutation and remains unchanged. Empirically, the LSTM's accuracy also did not drop under this shuffling constraint (remaining at $\sim$70.6%). This confirms that no model exploits any sequential structure, and that recurrent architectures do not extract any meaningful signal beyond what the order invariant linear baseline already captures.

Classifier	LLaDA Acc (%)	LLaDA AUROC	Dream Acc (%)	Dream AUROC
Logistic Regression	71.05 ± 0.72	0.7701 ± 0.0098	63.40 ± 1.30	0.6972 ± 0.0072
LSTM (Unidirectional)	71.75 ± 0.54	0.7895 ± 0.0061	62.87 ± 1.14	0.6713 ± 0.0069
Bidirectional LSTM + Attention	71.93 ± 0.71	0.7906 ± 0.0062	64.67 ± 0.70	0.6977 ± 0.0069

Table 5: Architecture comparison over 5 random splits. On both models, all confidence intervals overlap substantially. The gap between LR and the Bidirectional LSTM is 0.88% on LLaDA and 1.27% on Dream, confirming that sequential modeling adds negligible benefit across architectures.

8.2 Seed Count Ablation: How Many Runs Are Needed?

Table 6 evaluates the effect of reducing the number of parallel denoising runs N, using the Logistic Regression classifier.

N Seeds	LLaDA Acc (%)	LLaDA AUROC	Dream Acc (%)	Dream AUROC
2	70.76 ± 0.62	0.7725 ± 0.0082	63.91 ± 1.17	0.6893 ± 0.0119
3	71.75 ± 1.06	0.7806 ± 0.0084	67.12 ± 0.95	0.7261 ± 0.0131
5	71.23 ± 1.00	0.7851 ± 0.0074	69.62 ± 0.51	0.7589 ± 0.0134

Table 6: Effect of seed count on detection performance (LR classifier, 5 random splits). LLaDA's accuracy is stable across seed counts ($\Delta = +0.47\%$), while Dream benefits substantially from additional seeds ($\Delta = +5.71\%$), suggesting the optimal seed count is dependent on architecture.

The two models exhibit strikingly different seed sensitivities. LLaDA's accuracy is essentially flat from $N = 2$ to $N = 5$ ($+0.47\%$, within error bars), confirming that two parallel runs are sufficient. In contrast, Dream benefits substantially: accuracy rises from 63.91% to 69.62% ($+5.71\%$), nearly closing the gap across architectures. This asymmetry is likely attributable to Dream's remasking based on entropy, which produces smoother denoising trajectories with less variance per run. With

only $N = 2$ seeds, the pairwise cosine distance is computed from a single pair, making the signal highly sensitive to stochastic noise. Increasing to $N = 5$ yields $\binom{5}{2} = 10$ pairwise comparisons, substantially improving the robustness of the TVS estimate and amplifying the conflict signal in Dream's smoother trajectory space.

9 Conclusion

We introduced the Trajectory Variance Score (TVS), a simple and interpretable method for detecting knowledge conflicts caused by RAG in discrete diffusion language models. By computing the mean pairwise cosine distance across $N = 2$ stochastic denoising trajectories at each timestep, TVS produces a feature vector that represents the entire trajectory. We demonstrate that a simple Logistic Regression classifier over this vector achieves competitive detection accuracy, suggesting that no complex sequential modeling is required to extract the conflict signal. The TVS curve directly visualizes the temporal tug of war between parametric and contextual attractors as a three phase dynamics profile. Evaluation across two distinct diffusion models, LLaDA (remasking with low confidence) and Dream 7B (remasking based on entropy), demonstrates that the TVS signal, the three phase temporal dynamics, and the sufficiency of linear classifiers all generalize across architectures. The architecture dependent seed sensitivity we observe (Dream benefits substantially from $N{>}2$, while LLaDA does not) provides a practical guideline: the optimal seed count should be calibrated per model. With as few as two parallel inference runs, TVS is a computationally lightweight and interpretable diagnostic for investigating knowledge friction in diffusion language models.

10 Limitations

TVS requires multiple parallel denoising runs, which introduces inference overhead compared to single pass methods. While we have mitigated extraction failures by employing robust fuzzy matching to isolate answer entities, this heuristic approach could still potentially fail on complex paraphrases, leading to a small but unquantified extraction failure rate. A further limitation is that TVS relies on a parametric memory filter that itself requires $N = 5$ generation runs at evaluation time; exploring lighter filtering strategies is a promising direction for future work. Finally, while we demonstrate generality across two diffusion models, TVS accuracy is dependent on the architecture: Dream 7B achieves $\sim$63% accuracy with $N = 2$ seeds versus LLaDA's $\sim$70%, and models with smoother denoising dynamics may require more parallel runs to surface a reliable conflict signal. Whether TVS generalizes to diffusion models with substantially different architectures or training regimes remains an open question.

11 Future Work

Our findings open several promising directions for future research. First, we plan to explore lighter parametric memory filtering strategies. Currently, our filter requires multiple generation runs at evaluation time, which increases inference latency; developing a single-pass or confidence-based heuristic could significantly reduce this overhead. Second, while we evaluated TVS on LLaDA and Dream 7B, investigating whether these temporal conflict dynamics generalize to diffusion models with substantially different architectures, such as continuous-time diffusion or flow-matching models, remains a key open question. Finally, we aim to move beyond detection by integrating TVS into active mitigation frameworks. By monitoring the trajectory variance in real-time, future systems could dynamically halt generation or adaptively adjust the context weight when knowledge friction is detected, enabling more resilient and conflict-aware RAG systems.

Ethics Statement

As language models are increasingly deployed in real world applications via Retrieval Augmented Generation, their susceptibility to knowledge conflicts poses significant risks of propagating misinformation or adversarial fakes. The Trajectory Variance Score (TVS) provides an interpretability mechanism to detect and intercept these hallucinations before they reach the end user. While TVS significantly enhances the transparency of discrete diffusion models, it should not be treated as a standalone safeguard. It is intended to complement, rather than replace, standard fact checking pipelines, ensuring that AI systems remain trustworthy and aligned with factual reality.

12 Code Availability

The code and experimental implementation for the Trajectory Variance Score (TVS) are publicly avail-

able at: https://github.com/sravankarthik/temporal-tug-of-war. The repository includes the evaluation pipeline and code for reproducing the experiments described in this work.

References

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2024. Discovering latent knowledge in language models without supervision. *Preprint*, arXiv:2212.03827.

Shenxu Chang, Junchi Yu, Weixing Wang, Yongqiang Chen, Jialin Yu, Philip Torr, and Jindong Gu. 2025. Tracedet: Hallucination detection from the decoding trace of diffusion large language models. *Preprint*, arXiv:2510.01274.

Arshia Hemmat, Philip Torr, Yongqiang Chen, and Junchi Yu. 2026. Tdgnet: Hallucination detection in diffusion language models via temporal dynamic graphs. *Preprint*, arXiv:2602.08048.

Anant Khandelwal, Manish Gupta, and Puneet Agrawal. 2025. Cocoa: Confidence and context-aware adaptive decoding for resolving knowledge conflicts in large language models. *Preprint*, arXiv:2508.17670.

Jaemin Kim and Jong Chul Ye. 2026. Adaptive guidance for retrieval-augmented masked diffusion models. *Preprint*, arXiv:2603.17677.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *Preprint*, arXiv:2302.09664.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Preprint*, arXiv:2005.11401.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2023. Locating and editing factual associations in gpt. *Preprint*, arXiv:2202.05262.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. 2025. Large language diffusion models. *Preprint*, arXiv:2502.09992.

Subham Sekhar Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. 2024. Simple and effective masked diffusion language models. *Preprint*, arXiv:2406.07524.

Yash Shah, Abhijit Chakraborty, Naresh Kumar Devulapally, Vishnu Lokhande, and Vivek Gupta. 2026. Oscar: Orchestrated self-verification and cross-path refinement. *Preprint*, arXiv:2604.01624.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis K. Titsias. 2025. Simplified and generalized masked diffusion for discrete data. *Preprint*, arXiv:2406.04329.

Chenyu Wang, Yueyuan Li, Yingmin Liu, and Yang Shu. 2026. Conflictrag: Detecting and resolving knowledge conflicts in retrieval augmented generation. *Preprint*, arXiv:2605.17301.

Han Wang, Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. 2025. Retrieval-augmented generation with conflicting evidence. *Preprint*, arXiv:2504.13079.

Johannes Welbl, Nelson F. Liu, and Matt Gardner. 2017. Crowdsourcing multiple choice science questions. *Preprint*, arXiv:1707.06209.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. 2025. Dream 7b: Diffusion large language models. *Preprint*, arXiv:2508.15487.

Chuanyue Yu, Jiahui Wang, Yuhan Li, Heng Chang, Ge Lan, Qingyun Sun, Jia Li, Jianxin Li, and Ziwei Zhang. 2026. Unlocking the potentials of retrieval-augmented generation for diffusion language models. *Preprint*, arXiv:2601.11342.