STEPWISE SEMANTIC UNCERTAINTY IN LONG-FORM LANGUAGE GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Semantic uncertainty of large language models (LLMs), i.e., their lack of confidence in the meaning of the generated output, is a well-established indicator of the reliability of short responses. However, in long-form generation, a single response-level estimate cannot distinguish reliable from unreliable units of meaning. Thus, we propose to represent and quantify the semantic uncertainty stepwise, i.e., induced by the next-token distribution at each step during generation, without additional post hoc response decomposition or access to internal model states that existing approaches rely on. We introduce the concept of an attribute as the topic a unit of meaning refers to, which can be identified as soon as it emerges. By representing the alternative tokens at each step jointly by both their meaning and attribute, we can separate uncertainty due to variation across attributes from uncertainty due to contradictory meanings within attributes. Given that the latter primarily indicates potential factual errors, we leverage our representation to construct an attribute-conditional, kernel-based von Neumann entropy that specifically quantifies this kind of uncertainty. Empirically, we show this is more predictive of token- and unit-wise accuracy in long-form language generation than existing baselines. Since our representation is available before a token is committed, we further utilize it to steer decoding and improve the factual correctness of generated responses.

1 INTRODUCTION

Large language models (LLMs) are deployed across a wide range of critical tasks (Singhal et al., 2023; Katz et al., 2024; Kasneci et al., 2023) and in increasingly autonomous settings (Luo et al., 2025), yet assessing the reliability of their outputs remains a central concern. Semantic uncertainty, i.e., an LLM's predictive uncertainty w.r.t. the meaning of a generated output rather than the phrasing of the output (Kuhn et al., 2023), has been shown to be an indicator for the factual correctness of short responses. However, in long-form generation, a single response consists of multiple individual units of meaning, such that a single, response-level semantic uncertainty estimate cannot separate reliable from unreliable ones. Some finer-grained approaches, therefore, quantify semantic uncertainty per unit of meaning, effectively distinguishing uncertainty about a unit's meaning from uncertainty about which unit to generate (Fadeeva et al., 2024). For assessing unit-wise reliability, only the former is usually considered relevant, while variation across possible units is interpreted as benign stylistic variation of free-form generation. However, these approaches usually require a decomposition of a completed response into its units of meaning, introducing additional postprocessing overhead and making them inherently post hoc and thus unsuitable to steer generation itself.

We argue that the distinction between within-unit contradiction and across-unit variability does not inherently rely on fully completed outputs. More specifically, at any generation step, the distribution across the alternative next tokens not only carries latent meaning but also information about attributes, i.e., the topics these meanings instantiate. Given a suitable representation, one can then separate within-attribute contradiction from cross-attribute variability at any step of the generation trajectory, regardless of output length. Based on this, we make three contributions. Firstly, we introduce an uncertainty representation that jointly captures both meanings and attributes within any next-token distribution. Leveraging kernel-based covariance operators, this representation not only preserves the graded nature of meaning and attribute similarity, but it also only requires the respective next-token distribution and two encoding functions, making it by design LLM-agnostic and

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Table 1: Systematic overview of approaches to uncertainty quantification in LLMs relevant to our work.

Method	Quant. Level	Report. Level	Signal Type	Struct. aware	On-the-fly	LLM-agnostic
TokenSAR (Duan et al., 2024)	Token	Sentence	Prob.+ Sem.	No	No	Yes
SE (Kuhn et al., 2023)	Response	Response	Prob. + Sem. (NLI)	No	No	Yes
KLE (Nikitin et al., 2024)	Response	Response	Prob. + Sem. (Prob.)	No	No	Yes
SD (Qiu & Miikkulainen, 2024)	Response	Response	Prob. + Sem. (Prob.)	No	No	Yes
CCP (Fadeeva et al., 2024)	Word	Claim	Prob. + Sem. (NLI)	Claim	No	Yes
Spectral (Walha et al., 2025)	Response	Response	Prob. + Sem. (Emb.)	No	No	Yes
STC (Cao et al., 2026)	Token	Response	Prob. + Sem. (Clust.)	No	No	Yes
EntityProbes (Obeso et al., 2026)	Token	Entity	Internal	NA	Yes	No
$\text{SE}_{M\|A}$ (ours)	Token	Token, Entity	Prob. + Sem. (NLI)	Attribute	Yes	Yes
$\text{VNE}_{M\|A}$ (ours)	Token	Token, Entity	Prob. + Sem. (Emb.)	Attribute	Yes	Yes

applicable on the fly. Secondly, we use this representation to separate contradictory meanings within attributes from uncertainty over which attribute to generate at any step of the generation trajectory. Specifically, we derive an attribute-conditional kernel-based von Neumann entropy, which quantifies the step-specific expected degree of within-attribute contradiction. Empirically, this measure has a higher predictive accuracy for incorrect tokens than existing uncertainty baselines. Thirdly, as the representation is available before a token is committed, we use it to steer generation itself. We define an empirical attribute-conditional risk for each candidate token and incorporate this risk as an additional pruning criterion in beam search, which empirically improves factuality in the generated response compared to existing decoding baselines.

2 Related work

We organize prior work according to properties that determine whether a measure can localize within-unit contradictions during generation. First, we distinguish between the *quantification level*, i.e., the granularity at which the uncertainty estimate is initially calculated, and the *reporting level*, i.e., the granularity at which the estimates are then aggregated. The two do not need to coincide, as several methods initially quantify at the token level but report and thus evaluate an aggregated estimate at another granularity. The *signal type* describes the source of information used to represent uncertainty. We distinguish between probabilistic measures derived from a model's predictive distribution, semantic measures that capture differences in meaning (via NLI-, embedding-, or clustering-based representations), and internal-state-based measures that rely on the model's hidden representations. The *structure awareness* (claim, attribute, no) is a complementary property only applicable to semantic methods, indicating whether these methods discriminate between across-unit variability (i.e., which unit to generate) and within-unit contradiction (i.e., contradictory meanings within a unit) as uncertainty signals. *On-the-fly* separates post hoc methods from on-the-fly methods. Finally, *LLM-agnostic* indicates whether a method transfers to a new LLM without LLM-specific training. Table 1 compares representative approaches and ours along these axes. This indicates that the closest competitors are claim-conditioned probability (CCP) (structure-aware, word-level quantification but post hoc) and semantic token clustering (STC) (post hoc, token-level but structure-agnostic), neither of which is designed to localize within-unit contradiction at the token level during generation.

3 Stepwise semantic uncertainty

In this work, we focus on the task of long-form generation. Given a finite, discrete token space $\mathcal{V}$ and an input prompt $\mathbf{x} = (x_1, \ldots, x_L), x_l \in \mathcal{V}$, an LLM generates an output response $\mathbf{y} = (y_1, \ldots, y_T)$ in an autoregressive manner, such that at each step t a token $y_t \in \mathcal{V}$ is drawn from the conditional distribution $p(\cdot \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})$, resulting in $p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta}) = \prod_{t=1}^{T} p(y_t \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})$ for a fully generated response. Due to the natural language domain, $\mathbf{y}$ is not merely a sequence of tokens, but also carries an underlying, latent meaning and an underlying structure. More specifically, within long-form generation, any output $\mathbf{y}$ can be decomposed into atomic claims, i.e., the smallest units of independent meaning (Min et al., 2023), which also serve as the units for assessing correctness.

Due to the autoregressive factorization, an LLM's predictive uncertainty, i.e., its uncertainty w.r.t. the predicted output, can be represented at different granularities. The most common approach is to focus on the response level, so the uncertainty represented by $p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta})$. However, $p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta})$ is a distribution over surface forms and thus disregards the underlying meaning, which is why response-level semantic uncertainty (Kuhn et al., 2023; Nikitin et al., 2024; Walha et al., 2025; Qiu & Miikkulainen, 2024) replaces $p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\theta})$ with a distribution over meanings, thereby not accounting for the spurious uncertainty due to differently phrased, yet meaning-wise equivalent outputs. Whilst this suits short-form texts, this response-level approach is invariant to the underlying units of meaning, which structure the meaning of longer outputs. This requires a finer granularity, for which claim-level methods (Fadeeva et al., 2024) refine the representation by conditioning the distribution over meaning classes on each generated claim. Thereby, they not only disregard the uncertainty due to different, yet meaning-wise equivalent phrasings per claim, but also the uncertainty about which claim to generate in the first place. This assumes that these two disregarded uncertainties simply reflect the stylistic variability during long-form generation, whereas only uncertainty about the meaning within individual claims is relevant for identifying potential factual errors. However, claims can only be identified once they have fully materialized (Obeso et al., 2026), so these methods require a post hoc decomposition of a complete response into its claims, which not only introduces an additional post-processing overhead but also prohibits intervening in generation itself.

Based on these considerations, we align the uncertainty representation with the individual steps of autoregressive factorization itself. More specifically, we aim for an uncertainty representation that (i) operates at the *token-level*, i.e., is induced at each step t by the next-token distribution $p(\cdot \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})$, thereby being available during generation; (ii) is *semantic*, i.e., reflects the different meanings present across alternative tokens at any step t, such that tokens that yield consistent meanings at that step t do not contribute to uncertainty; (iii) is *structure-aware*, i.e., not only reflects meaning but also the corresponding units of meaning at a generation point t, s.t. uncertainty due to contradiction within a unit can be separated from uncertainty about which unit to focus on next at any step t.

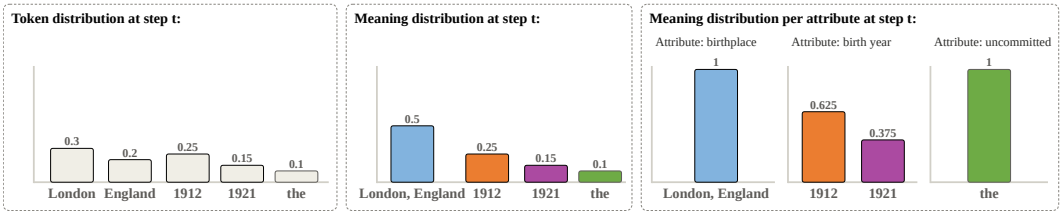

Figure 1: Exemplary token alternatives at step t. Input: "Who was Alan Turing?"; Generated prefix until time point $t-1$: "Alan Turing was a scientist born in".

Figure 1 shows three potential uncertainty representations at a specific step t, each refining the previous one along the above established desiderata. The first is the next-token distribution $p(\cdot \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})$ itself, which satisfies desideratum (i), but conflates uncertainty w.r.t. token-level surface forms with uncertainty w.r.t. meaning, thereby violating desideratum (ii). The second transfers the intuition behind response-level approaches to each step t, and uses the distribution over meanings induced by $p(\cdot \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})$ as an uncertainty representation. However, this then disregards the underlying units of meaning present at step t. Thus, to enable the identification of such units of meaning in our stepwise setting, we shift from claims as post hoc units of meaning to an even finer structure that can be identified as soon as it emerges. Within an atomic claim, entities are defined as its meaningful units, i.e., as the tokens defining the meaning of a claim (Yeh et al., 2025; Obeso et al., 2026). We take this definition further and distinguish explicitly between the specific attribute such an entity is targeting (e.g., a person's birth year, a person's birthplace) and the actual meaning (e.g., "1912"). Consequently, at each step, a token can either (further) instantiate an attribute or not. Thus, as Figure 1 shows, representing alternatives using both their meanings and attributes jointly yields a representation that satisfies all three desiderata. It is induced by $p(\cdot \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})$, it is semantic, and it encodes how meanings relate when additionally considering their corresponding attribute.

4 ATTRIBUTE-CONDITIONAL SEMANTIC UNCERTAINTY

In the following, we firstly formalize the above established concepts by lifting the next-token distribution into a meaning- and attribute-aware representation using covariance operators. Secondly, we quantify the within-attribute semantic uncertainty encoded in this representation using an attribute-conditional von Neumann entropy. Thirdly, we use this representation to derive an uncertainty-informed decoding procedure.

4.1 REPRESENTING LOCAL MEANING AND LOCAL ATTRIBUTES

At a generation step t, we define $\mathbf{y}_{\leq t} = \mathbf{y}_{<t} \circ y$ as the prefix extended by any token y. For each such single-token continuation, we define the *local attribute* as the attribute it addresses and the *local meaning* as the value of the respective attribute. In our example in Figure 1, the local attribute of token "1912" concerns the year of birth, with the corresponding local meaning being 1912. Note that each token that does not instantiate an attribute is interpreted as referring to its own distinct and uncommitted attribute with a distinct and undetermined meaning.

Formalization We formalize local meaning and local attribute using a latent meaning space $\mathcal{M}$ and a latent attribute space $\mathcal{A}$, both instantiated as d-dimensional Euclidean spaces with unit-norm embeddings. The encoding functions $f_M : \mathcal{Y}_{\leq t} \to \mathcal{M}$ and $f_A : \mathcal{Y}_{\leq t} \to \mathcal{A}$ map any (partially evolved) sequence into the respective latent space. Together with $Y_t \sim p(\cdot \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})$, we can thus define the local meaning and the local attribute as the random variables

$$M_t := f_M(\mathbf{y}_{<t} \circ Y_t), \qquad A_t := f_A(\mathbf{y}_{<t} \circ Y_t). \tag{1}$$

Importantly, the corresponding distributions are pushforward measures of $p(\cdot \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})$, s.t. the probability mass assigned to each token is not changed, but its representation is relocated to its meaning and its attribute embedding.

Covariance operators To arrive at a reweighted representation, we follow Bach (2022) and transfer the concept of a density operator from the quantum setting to a probability distribution. For any distribution p on a space $\mathcal{Y}$, with a normalized, p.s.d. kernel $k : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ with RKHS $\mathcal{H}$, and an associated feature map $\phi : \mathcal{Y} \to \mathcal{H}$, Bach (2022) defines the covariance operator as $\Sigma_Y = \mathbb{E}_{Y \sim p}[\phi(Y) \otimes_{\mathcal{H}} \phi(Y)]$, which is the expected outer product of the feature map. By design, it is a density operator in the quantum sense, i.e., a mixture of pure states weighted by p.

In our setting, let k_M and k_A be continuous, normalized, p.s.d. kernels on $\mathcal{M}$ and $\mathcal{A}$ with corresponding feature maps $\phi_M : \mathcal{M} \to \mathcal{H}_M$ and $\phi_A : \mathcal{A} \to \mathcal{H}_A$. To express the covariance operators in terms of the next-token distribution, we additionally define

$$\psi_M(y) := \phi_M\big(f_M(\mathbf{y}_{<t} \circ y)\big), \qquad \psi_A(y) := \phi_A\big(f_A(\mathbf{y}_{<t} \circ y)\big). \tag{2}$$

Thereby, we can state the covariance operators of local meaning and local attributes as

$$\Sigma_{M_t} = \sum_{y \in \mathcal{V}} p_t(y) \left(\psi_M(y) \otimes_{\mathcal{H}_M} \psi_M(y)\right) \text{ and } \Sigma_{A_t} = \sum_{y \in \mathcal{V}} p_t(y) \left(\psi_A(y) \otimes_{\mathcal{H}_A} \psi_A(y)\right), \tag{3}$$

with $p_t(y) := p(y \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})$ for notational simplicity. Intuitively, these represent the meaning and attribute spread present within $p(\cdot \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})$. Alternatives with similar meanings/attributes point in nearly the same direction in the respective RKHS, s.t. their probabilities accumulate in this direction. In contrast, alternatives with dissimilar meanings/attributes point into orthogonal directions of the respective RKHS, thereby increasing the spread. Consequently, Σ_{M_t} corresponds to a graded version of the second representation in Figure 1. To arrive at the third, joint representation of (M_t, A_t), we define a joint state covariance operator by combining the meaning- and attribute-specific feature maps. Analogously to Bach (2022), we consider the product set $\mathcal{M} \times \mathcal{A}$, the feature space $\mathcal{H} = \mathcal{H}_M \otimes \mathcal{H}_A$ corresponding to the product kernel $k((m, a), (m', a')) = k_M(m, m')k_A(a, a')$ with the feature map $\phi(m, a) = \phi_M(m) \otimes \phi_A(a)$. Writing $\psi(y) := \psi_M(y) \otimes \psi_A(y)$, we define the joint covariance operator as

$$\Sigma_{M_t A_t} = \sum_{y \in \mathcal{V}} p_t(y) \left(\psi(y) \otimes_{\mathcal{H}} \psi(y)\right). \tag{4}$$

As this is a convex combination of product pure states, $\Sigma_{M_t A_t}$ is by construction separable (Cerf & Adami, 1999). Analogously to the interpretation of Σ_{M_t} and Σ_{A_t}, $\Sigma_{M_t A_t}$ measures the joint spread of meanings and attributes, s.t. two alternatives only share the same direction if they have similar meanings and attributes, whereas a difference in either of them yields orthogonal directions in the RKHS. Compared to Σ_{A_t}, $\Sigma_{M_t A_t}$ only adds spread if two alternatives share an attribute but differ in meaning. Alternatives with different attributes already separate in Σ_{A_t}, and thus in $\Sigma_{M_t A_t}$. Therefore, by removing the attribute spread from the joint spread, we can extract the within-attribute spread. In the following section, we turn this intuition into a quantitative measure.

4.2 Attribute-conditional kernel-based von Neumann entropy

As the covariance operators are density operators (Bach, 2022), measures that operate on these density operators can be transferred to these covariance operators. More specifically, for any covariance operator Σ_Y, Bach (2022) defines the kernel-based von Neumann entropy as

$$H_{VN}(Y) := -\operatorname{tr}[\Sigma_Y \log \Sigma_Y]. \tag{5}$$

Intuitively, this kernel-based von Neumann entropy is a generalization of the Shannon entropy, measuring the spread of the pure states in the RKHS. In our setting, at any step t, we can thus quantify the meaning spread using $H_{VN}(M_t) := -\operatorname{tr}[\Sigma_{M_t} \log \Sigma_{M_t}]$, the attribute spread $H_{VN}(A_t) := -\operatorname{tr}[\Sigma_{A_t} \log \Sigma_{A_t}]$, and the joint spread $H_{VN}(M_t, A_t) := -\operatorname{tr}[\Sigma_{M_t A_t} \log \Sigma_{M_t A_t}]$.

Furthermore, we can use these quantities to quantify uncertainty arising from distinct attribute meanings. To this end, we leverage the conditional von Neumann entropy (Cerf & Adami, 1997) as

$$H_{VN}(M_t \mid A_t) = H_{VN}(M_t, A_t) - H_{VN}(A_t). \tag{6}$$

Intuitively, $H_{VN}(M_t \mid A_t)$ quantifies the remaining semantic spread after controlling for the spread that can be allocated to different attributes, which thus corresponds to the expected within-attribute semantic uncertainty at step t before committing to any specific token at t. Importantly, as each token that does not instantiate an attribute is interpreted as referring to its own distinct and uncommitted attribute with a distinct and undetermined meaning, these uncommitted meanings do not contribute to $H_{VN}(M_t \mid A_t)$.

In the general quantum setting, the conditional von Neumann entropy is not necessarily non-negative (Cerf & Adami, 1997). However, this issue does not arise here. By Equation (4), $\Sigma_{M_t A_t}$ is a convex combination of product states and is therefore separable by construction (Cerf & Adami, 1999). As separability implies $H_{VN}(M_t, A_t) \geq \max(H_{VN}(M_t), H_{VN}(A_t))$ (Cerf & Adami, 1997), the attribute-conditional VNE in Equation (6) is guaranteed to be non-negative.

A plug-in estimator In practice, the VNE is estimated from a finite, weighted set of alternatives. Given N alternatives with non-negative weights $\boldsymbol{\eta}$ and $\sum_{j=1}^{N} \eta_j = 1$, the empirical kernel matrix $K \in \mathbb{R}^{N \times N}$, with entries given by the kernel evaluated on the i-th and j-th alternative, is scaled as $\tilde{K} = \operatorname{Diag}(\boldsymbol{\eta})^{1/2} K \operatorname{Diag}(\boldsymbol{\eta})^{1/2}$, and the kernel-based VNE is estimated as

$$\hat{H}_{VN} = -\operatorname{tr}[\tilde{K} \log \tilde{K}] = -\sum_{i=1}^{N} \hat{\lambda}^i \log \hat{\lambda}^i, \tag{7}$$

where $\hat{\lambda}^i$ are the eigenvalues of $\tilde{K}$ (Bach, 2022). Response-level methods instantiate Equation (7) with N full responses sampled from the LLM as alternatives and uniform weights $\boldsymbol{\eta} = \frac{1}{N}\mathbf{1}_N$. We instead select, at each step t during generation, the top-N alternative tokens of $p(\cdot \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})$ and renormalize their probabilities to obtain $\boldsymbol{\eta}$. We map the resulting set of single-token continuations $\{\mathbf{y}_{<t} \circ y_t^j\}_{j=1}^N = \{\mathbf{y}_{\leq t}^j\}_{j=1}^N$ into the latent meaning and attribute spaces using the encoding functions f_M and f_A, resulting in $\{\mathbf{m}_t^j\}_{j=1}^N = \{f_M(\mathbf{y}_{\leq t}^j)\}_{j=1}^N$ and $\{\mathbf{a}_t^j\}_{j=1}^N = \{f_A(\mathbf{y}_{\leq t}^j)\}_{j=1}^N$. The empirical kernel matrices $[K_{M_t}]_{ij} = k_M(\mathbf{m}_t^i, \mathbf{m}_t^j)$, $[K_{A_t}]_{ij} = k_A(\mathbf{a}_t^i, \mathbf{a}_t^j)$ and $K_{M_t A_t} = K_{M_t} \odot K_{A_t}$, with $\odot$ the elementwise product, are scaled using $\boldsymbol{\eta}$, and by virtue of Equation (6), the estimated attribute-conditional VNE is then simply

$$\hat{H}_{VN}(M_t \mid A_t) = \hat{H}_{VN}(M_t, A_t) - \hat{H}_{VN}(A_t) = -\sum_{i=1}^{N} \hat{\lambda}_{M_t A_t}^i \log \hat{\lambda}_{M_t A_t}^i + \sum_{i=1}^{N} \hat{\lambda}_{A_t}^i \log \hat{\lambda}_{A_t}^i, \tag{8}$$

with $\hat{\lambda}_{M_t A_t}^i, \hat{\lambda}_{A_t}^i$ the eigenvalues of $\tilde{K}_{M_t A_t}$ and $\tilde{K}_{A_t}$.

4.3 ATTRIBUTE-CONDITIONAL RISK FOR DECODING

Inspired by approaches such as Tomov et al. (2026), which use a latent response-level meaning space to guide response selection, we leverage our representation to guide decoding toward tokens that exhibit little contradiction with alternatives addressing the same attribute. Specifically, at each step during generation, we assign each candidate token an attribute-conditional semantic risk quantifying this within-attribute contradiction and incorporate it into the decoding objective. To this end, we define the attribute- and meaning-aware dissimilarity between two meaning-attribute pairs $(\mathbf{m}^j, \mathbf{a}^j), (\mathbf{m}^i, \mathbf{a}^i)$ as

$$d\big((\mathbf{m}^j, \mathbf{a}^j), (\mathbf{m}^i, \mathbf{a}^i)\big) \;=\; k_A(\mathbf{a}^j, \mathbf{a}^i)\,\big(1 - k_M(\mathbf{m}^j, \mathbf{m}^i)\big). \tag{9}$$

This dissimilarity specifically penalizes within-attribute contradiction, as only pairs that have low meaning similarity and high attribute similarity are assigned a high distance. Using this, one way to express the risk of within-attribute contradiction for a specific candidate token y^j with corresponding $(\mathbf{m}^j, \mathbf{a}^j)$ would be to use $\sum_{i=1}^{N} \eta_i \, [K_{A_t}]_{ji} \, (1 - [K_{M_t}]_{ji})$, i.e., the candidate's average degree of within-attribute contradiction across alternatives. This empirical risk, however, ignores how the attribute mass is distributed. To mitigate this, we normalize by the attribute mass and define the attribute-conditional empirical risk as

$$\hat{R}(\mathbf{m}^j, \mathbf{a}^j) \;=\; \frac{\sum_{i=1}^{N} \eta_i [K_{A_t}]_{ji} \left(1 - [K_{M_t}]_{ji}\right)}{\sum_{i=1}^{N} \eta_i [K_{A_t}]_{ji}}. \tag{10}$$

We use this risk during a beam-search-inspired decoding procedure. In traditional beam search, we maintain a set of B alternative hypotheses, i.e., a set of partially evolved sequences $\{\mathbf{y}^b_{<t}\}_{b=1}^{B}$. At each step, every hypothesis is expanded by its top-N next-token candidates. During pruning, only the B single-token continuations with the highest cumulative log-likelihood are retained from the pool of $B \cdot N$ continuations (Freitag & Al-Onaizan, 2017). We incorporate the attribute-conditional empirical risk into pruning, penalizing continuations that show a high risk of within-attribute contradiction. We extend the pruning score of traditional beam search, s.t. for each continuation $\mathbf{y}^b_{<t} \circ y^j_t$, we recursively calculate

$$S(\mathbf{y}^b_{<t} \circ y^j_t) := S(\mathbf{y}^b_{<t}) + \log p(y^j_t \mid \mathbf{y}^b_{<t}, \mathbf{x}, \boldsymbol{\theta}) - \rho \, \hat{R}(\mathbf{m}^j_t, \mathbf{a}^j_t), \tag{11}$$

where $\rho \geq 0$ trades off likelihood against within-attribute agreement and $\rho = 0$ recovers standard beam search. Since likelihood already enters S through the log-probability term, we use uniform weights rather than the renormalized token probabilities within $\hat{R}(\mathbf{m}^j_t, \mathbf{a}^j_t)$.

5 DOWNSTREAM APPLICATIONS

We are interested in whether the attribute-conditional, kernel-based VNE can be used as a signal within downstream applications, and how its performance compares to other uncertainty measures. More specifically, we evaluate on:

- A correctness prediction task, where we evaluate the performance of the attribute-conditional, kernel-based VNE as a hallucination predictor.
- A decoding task, where we evaluate whether leveraging attribute-conditional risk can improve generation quality.

5.1 HALLUCINATION DETECTION

In principle, we are interested in how uncertainty scores relate to incorrect entities, i.e., the spans of tokens causing a claim to be incorrect. Firstly, we compare the uncertainty estimates for incorrect tokens (i.e., from incorrect entities) with those of all other tokens. This focuses on how well incorrect tokens can be distinguished from correct and non-entity tokens. Secondly, we focus on how well the uncertainty measures can discriminate correct entities from incorrect ones, reducing the set of compared tokens to the ones associated with an entity. We argue that both aspects are relevant in our stepwise setting, since an uncertainty measure should neither report a high score for benign tokens (both correct and non-entity tokens) nor a low score for incorrect ones.

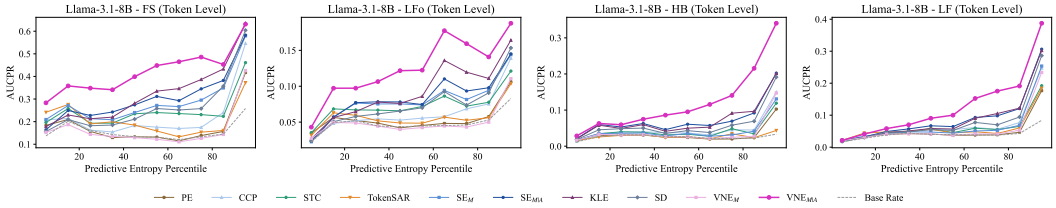

Figure 2: Improvement of $\text{VNE}_{M|A}$ over uncertainty baselines vs. predictive entropy bins (Llama-3.1-8B). The combination of gradedness and structure awareness is crucial for the predictive accuracy of methods.

Models, datasets and metrics We evaluate on instruction-tuned versions of Mistral-7B-v0.3 (Jiang et al., 2023), Gemma-4-12B (Gemma Team, 2026), and Llama-3.1-8B (Grattafiori et al., 2024). To encode local meaning and local attributes, we use two fine-tuned versions of EmbeddingGemma-300M (Vera et al., 2025) (see Appendix C.1). We evaluate on four long-form datasets. We use the FActScore (FS) biography dataset (Min et al., 2023), as well as the LongFact objects (LFo) dataset (Wei et al., 2024). Here, we generate responses per LLM and identify incorrect and correct entities, along with their corresponding tokens, using extended FActScore (Min et al., 2023) and SAFE (Wei et al., 2024) evaluation pipelines (see Appendix B.3). We also evaluate on two labeled datasets: annotated versions of LongFact (LF) (Obeso et al., 2026; Wei et al., 2024) and HealthBench (HB) (Obeso et al., 2026; Arora et al., 2025), which provide spans of incorrect and correct entities within pre-generated responses. However, as these datasets contain on-policy responses only for Llama-3.1-8B, the performances on Mistral-7B-v0.3 and Gemma-4-12B should be read more as an indicator of off-policy fact-checking capabilities (see Appendix B.3). Due to label imbalance (see base rate, i.e., the fraction of incorrect tokens or entities in Table 2), we report the Area Under the Precision-Recall curve (AUPR), with the Area Under the Receiver Operating Characteristic curve (AUROC) in Appendix A.

Baselines and methods We compare our method to other LLM-agnostic uncertainty baselines, specifically predictive entropy (PE), CCP, STC, TokenSAR, SE, KLE, and SD, as mentioned in Section 2. Since we evaluate based on token-level scores, we need baselines that yield uncertainty estimates at this granularity. Whilst token-level methods provide this by design, we introduce localized versions for methods that quantify at higher levels. Further details about localization, baselines, and baseline-specific parameters can be found in Appendix B.1. Furthermore, we include a meaning-kernel VNE (VNE_M), which corresponds to the second approach described in Figure 1, and an attribute-conditional version of semantic entropy ($\text{SE}_{M|A}$), which is the hard-kernel variant of the attribute-conditional, kernel-based VNE ($\text{VNE}_{M|A}$). For $\text{VNE}_{M|A}$ and VNE_M, we use the Radial Basis Function (RBF) kernel. To ensure a fair comparison to baselines that are all evaluated using the same hyperparameters across settings, we use $\gamma_M = 1, \gamma_A = 7$ as default parameters for the respective meaning and attribute kernels. A further discussion on the choice of these parameters can be found in Appendix A. At each step, we use $N = 10$ alternative tokens, with ablations on N in Appendix A. As the baselines differ w.r.t. their inherent aggregation strategies, we report the aggregation strategy that yields the best result per baseline and method on the entity level.

Results Table 2 reports both the token- and entity-level AUPR, together with the respective base rate across models and datasets. At the token level, $\text{VNE}_{M|A}$ performs best across all 12 model-dataset combinations. At the entity level, it remains the best-performing method in 8 out of the 12 settings, despite smaller performance margins. We discuss this decrease in Appendix A.

To examine what drives the overall performance gain of $\text{VNE}_{M|A}$, Figure 2 shows the token-level AUPR as a function of predictive entropy for Llama-3.1-8B. The advantage of $\text{VNE}_{M|A}$ is particularly pronounced in the mid- to high-entropy regime, where the next-token distribution provides sufficient variation for the semantic structure of the alternatives to become informative. Moreover, these gains appear to rely on the combination of gradedness and attribute conditioning. Neither its hard-clustering counterpart $\text{SE}_{M|A}$ nor KLE, the strongest graded but purely semantic baseline, outperforms $\text{VNE}_{M|A}$ in any of these regimes. This pattern is consistent across models and datasets (see Appendix A).

Table 2: Token-level and entity-level AUPR across different methods, models, and datasets. Gold indicates best, silver second-best performance in the given setting.

		Token Level				Entity Level			
Model	Method	FS	LFo	HB	LF	FS	LFo	HB	LF
Mistral-7B-v0.3	Base rate	0.266	0.047	0.027	0.017	0.602	0.132	0.269	0.142
	PE	0.315	0.048	0.041	0.020	0.744	0.148	0.476	0.375
	CCP	0.425	**0.066**	0.041	0.031	0.773	0.166	0.451	0.339
	STC	0.312	0.048	0.036	0.025	0.745	0.151	0.446	0.333
	TokenSAR	0.316	0.045	0.027	0.022	0.756	0.141	0.381	0.357
	KLE	0.380	0.059	0.051	0.039	**0.790**	0.177	0.510	0.427
	SD	0.363	0.054	0.044	0.040	0.785	0.171	0.467	0.372
	SE_M	0.410	0.061	0.041	0.029	0.780	0.174	0.480	0.398
	$SE_{M\|A}$	**0.428**	0.066	**0.057**	**0.050**	0.766	**0.183**	**0.515**	**0.429**
	VNE_M	0.314	0.047	0.044	0.023	0.756	0.151	0.486	0.387
	$VNE_{M\|A}$	**0.483**	**0.095**	**0.109**	**0.088**	**0.785**	**0.202**	**0.559**	**0.494**
Gemma-4-12B	Base rate	0.137	0.051	0.028	0.026	0.432	0.164	0.269	0.217
	PE	0.195	0.073	0.030	0.029	0.509	0.203	0.468	**0.517**
	CCP	0.214	0.074	0.039	0.038	0.523	0.201	0.397	0.371
	STC	**0.241**	**0.092**	0.035	0.041	0.500	0.203	0.392	0.456
	TokenSAR	0.185	0.073	0.027	0.040	**0.541**	**0.209**	0.343	0.478
	KLE	0.232	0.077	0.045	0.053	**0.540**	0.205	**0.480**	0.517
	SD	0.211	0.072	0.041	0.055	0.524	0.198	0.407	0.418
	SE_M	0.223	0.072	0.037	0.040	0.524	0.196	0.460	0.511
	$SE_{M\|A}$	0.220	0.073	**0.056**	**0.070**	0.512	0.194	0.476	0.482
	VNE_M	0.185	0.070	0.035	0.036	0.511	**0.206**	0.475	**0.521**
	$VNE_{M\|A}$	**0.288**	**0.105**	**0.090**	**0.092**	0.535	0.205	**0.531**	0.483
Llama-3.1-8B	Base rate	0.156	0.048	0.025	0.041	0.383	0.131	0.269	0.340
	PE	0.209	0.066	0.036	0.078	0.571	0.171	0.478	0.657
	CCP	0.258	0.077	0.048	0.102	0.617	0.203	**0.521**	0.660
	STC	0.270	0.084	0.041	0.086	0.583	0.168	0.503	0.647
	TokenSAR	0.196	0.065	0.028	0.080	0.612	0.176	0.436	0.660
	KLE	0.298	0.086	0.054	0.112	**0.631**	**0.207**	0.503	0.645
	SD	0.276	0.080	0.053	0.109	0.622	0.205	0.507	0.644
	SE_M	0.305	0.083	0.042	0.097	0.623	0.204	0.484	0.642
	$SE_{M\|A}$	**0.325**	**0.086**	**0.062**	**0.123**	0.612	0.199	0.517	0.614
	VNE_M	0.210	0.066	0.042	0.090	0.598	0.185	0.489	**0.666**
	$VNE_{M\|A}$	**0.381**	**0.121**	**0.108**	**0.166**	**0.634**	**0.225**	**0.566**	**0.718**

5.2 Improved Decoding

To evaluate whether our representation can improve the factual correctness of generated responses, we focus on the FActScore biography dataset (Min et al., 2023), as models have exhibited the highest hallucination rates and thus have the most room for improvement. Otherwise, we use the same experimental setup as in Section 5.1. We compare our attribute-conditional beam search against greedy decoding, sampling, traditional beam search, contrastive search (Su et al., 2022), and DoLa (Chuang et al., 2024) as baselines (see Appendix B.2). For both traditional beam search and our attribute-conditional beam search, we use six beams. Additionally, we use $\rho = 3$ within the attribute-conditional beam search and $\gamma_M = 1, \gamma_A = 7$ as parameters for the meaning and attribute kernels across settings.

Results In Table 3, we report the number of correct entities across responses, the proportion of extracted entities that are correct, and the mean response length in tokens. Attribute-conditional beam search improves factuality compared to all baselines. As standard beam search with the same width and length penalty yields only marginal gains over greedy, we argue that the improvement of the attribute-conditional beam search can be mainly attributed to the attribute-conditioning and not the exploration width itself. In addition, attribute-conditional beam search does not yield substantially longer or shorter outputs than the majority of baselines across models, whilst simultaneously

Table 3: Comparison of different decoding methods on the FActScore biography dataset. Gold indicates best, silver second-best performance in the given setting.

Model	Method	Responses	Entities	Correct	Avg. tokens	Frac. correct
Mistral-7B-v0.3	Greedy	200	11462	4596	381.1	0.401
	Sampling	200	11108	4416	380.6	0.398
	Beam	200	10820	4507	385.4	0.417
	Contrastive	200	10525	6106	376.5	0.580
	DoLa	200	10778	6255	364.5	**0.581**
	Attr. Beam	200	10744	6615	370.3	**0.616**
Llama-3.1-8B	Greedy	200	8230	5072	249.7	0.616
	Sampling	200	7671	4455	224.3	0.581
	Beam	200	8048	5135	247.3	0.638
	Contrastive	200	8638	6338	273.2	0.734
	DoLa	200	8863	6511	254.3	**0.735**
	Attr. Beam	200	8063	6690	238.4	**0.830**
Gemma-4-12B	Greedy	200	13840	7590	371.3	0.548
	Sampling	200	13912	7716	371.1	0.555
	Beam	200	14284	8135	379.7	0.570
	Contrastive	200	13864	10137	376.7	**0.732**
	DoLa	200	15174	10819	395.6	0.713
	Attr. Beam	200	14161	10525	383.9	**0.743**

producing the largest or second largest number of correct entities, thus focusing on generating more reliable attributes.

6 Conclusion

We introduced a stepwise semantic uncertainty framework for long-form generation that represents each next-token distribution jointly in terms of its local meaning and the attributes these meanings instantiate. Using kernel-based covariance operators allows this representation to preserve graded similarities in both dimensions and, crucially, to distinguish uncertainty over which attribute to generate from contradictory meanings within an attribute. Building on this representation, we proposed an attribute-conditional kernel-based von Neumann entropy that specifically quantifies the latter. Across models and datasets, this measure is more predictive of token- and entity-level factual errors than the majority of existing uncertainty baselines, with the results indicating that structure awareness is particularly effective when combined with graded representations. In addition, we used our representation to derive an attribute-conditional risk for individual candidate tokens at any step during decoding and used it to steer decoding toward within-attribute agreement, improving factuality across the evaluated models.

Limitations and future work Conceptually, attribute conditioning rests on the assumption that variation across plausible attributes is largely benign, whereas contradictory meanings within an attribute are informative of factual errors. Our experiments support this distinction on the evaluated factoid-heavy long-form datasets. However, this might not hold for other tasks, where, for example, the ground-truth is inherently ambiguous. Practically, both the value of the representation to downstream applications and its computational efficiency depend on the meaning and attribute encoders. Therefore, a highly relevant future direction would be to either train two small yet highly specialized meaning and attribute encoders, thereby avoiding any efficiency vs. performance tradeoff, or to use a nested or hierarchical embedding space such that a single encoder can represent both attribute and meaning jointly. In addition, selective representation and quantification, for instance by evaluating semantic uncertainty only at steps identified by a cheaper preliminary signal, could reduce some computational overhead that stepwise methods carry by design.

AI USE STATEMENT

In this work, we used generative AI tools for formulating mathematical claims, more specifically, for cross-checking the methodological validity of the joint covariance operator construction and the non-negativity claim. We also used generative AI tools for general feedback about the proposed representation, quantification, and decoding methods. In addition, we used it for writing experiment code, more specifically to streamline a manual implementation of our methods, to include (localized) baselines into our pipeline, to increase the efficiency of existing evaluation pipelines, and to set up the fine-tuning procedure. Lastly, we have used generative AI for cross-checking our interpretation of experiment results. We have not used generative AI tools for developing theoretical models or conceptual frameworks, providing critical ingredients for proving mathematical claims, and proposing or refining hypotheses. Generating synthetic data, assisting in the writing of proofs, assisting with translation, cleaning and reformatting datasets, or supporting qualitative and thematic data analysis do not apply to this work. Additionally, we used generative AI tools for creating scientific figures, creating and editing software code, and editing the paper to improve readability. We have reviewed all AI-assisted work. We verified the correctness of LLM-generated code, claims, and analysis. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Rahul K. Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero-Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, Johannes Heidecke, and Karan Singhal. Healthbench: Evaluating large language models towards improved human health, 2025. URL https://arxiv.org/abs/2505.08775.

Francis Bach. Information theory with kernel methods, 2022. URL https://arxiv.org/abs/2202.08545.

Qi Cao, Andrew Gambardella, Takeshi Kojima, Yutaka Matsuo, and Yusuke Iwasawa. Semantic token clustering for efficient uncertainty quantification in large language models. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 682–696, Rabat, Morocco, March 2026. Association for Computational Linguistics. ISBN 979-8-89176-381-4. doi: 10.18653/v1/2026.eacl-short.49. URL https://aclanthology.org/2026.eacl-short.49/.

N. J. Cerf and C. Adami. Negative entropy and information in quantum mechanics. *Physical Review Letters*, 79(26):5194–5197, December 1997. ISSN 1079-7114. doi: 10.1103/physrevlett.79.5194. URL http://dx.doi.org/10.1103/PhysRevLett.79.5194.

N. J. Cerf and C. Adami. Quantum extension of conditional probability. *Physical Review A*, 60(2): 893–897, August 1999. ISSN 1094-1622. doi: 10.1103/physreva.60.893. URL http://dx.doi.org/10.1103/PhysRevA.60.893.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James Glass, and Pengcheng He. Dola: Decoding by contrasting layers improves factuality in large language models, 2024. URL https://arxiv.org/abs/2309.03883.

Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models, 2024. URL https://arxiv.org/abs/2307.01379.

Ekaterina Fadeeva, Aleksandr Rubashevskii, Artem Shelmanov, Sergey Petrakov, Haonan Li, Hamdy Mubarak, Evgenii Tsymbalov, Gleb Kuzmin, Alexander Panchenko, Timothy Baldwin, Preslav Nakov, and Maxim Panov. Fact-checking the output of large language models via token-level uncertainty quantification, 2024. URL https://arxiv.org/abs/2403.04696.

Markus Freitag and Yaser Al-Onaizan. Beam search strategies for neural machine translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pp. 56–60. Association for Computational Linguistics, 2017. doi: 10.18653/v1/w17-3207. URL http://dx.doi.org/10.18653/v1/W17-3207.

Gemma Team. Gemma 4 technical report, 2026. URL https://arxiv.org/abs/2607.02770.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=XPZIaotutsD.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. URL https://arxiv.org/abs/2310.06825.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension, 2017. URL https://arxiv.org/abs/1705.03551.

Enkelejda Kasneci, Kathrin Seßler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günnemann, Eyke Hüllermeier, et al. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and individual differences*, 103, 2023.

Daniel Martin Katz, Michael James Bommarito, Shang Gao, and Pablo Arredondo. Gpt-4 passes the bar exam. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, 382, 2024.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=VD-AYtP0dve.

Edward Loper and Steven Bird. Nltk: The natural language toolkit, 2002. URL https://arxiv.org/abs/cs/0205028.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. URL https://arxiv.org/abs/1711.05101.

Junyu Luo, Weizhi Zhang, Ye Yuan, Yusheng Zhao, Junwei Yang, Yiyang Gu, Bohan Wu, Binqi Chen, Ziyue Qiao, Qingqing Long, Rongcheng Tu, Xiao Luo, Wei Ju, Zhiping Xiao, Yifan Wang, Meng Xiao, Chenwu Liu, Jingyang Yuan, Shichang Zhang, Yiqiao Jin, Fan Zhang, Xian Wu, Hanqing Zhao, Dacheng Tao, Philip S. Yu, and Ming Zhang. Large language model agent: A survey on methodology, applications and challenges, 2025. URL https://arxiv.org/abs/2503.21460.

George A. Miller. WordNet: A lexical database for English. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*, 1994. URL https://aclanthology.org/H94-1111/.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation, 2023. URL https://arxiv.org/abs/2305.14251.

Alexander Nikitin, Jannik Kossen, Yarin Gal, and Pekka Marttinen. Kernel language entropy: Fine-grained uncertainty quantification for llms from semantic similarities, 2024. URL https://arxiv.org/abs/2405.20003.

Oscar Obeso, Andy Arditi, Javier Ferrando, Joshua Freeman, Cameron Holmes, and Neel Nanda. Real-time detection of hallucinated entities in long-form generation, 2026. URL https://arxiv.org/abs/2509.03531.

Xin Qiu and Risto Miikkulainen. Semantic density: Uncertainty quantification for large language models through confidence measurement in semantic space, 2024. URL https://arxiv.org/abs/2405.13845.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL http://arxiv.org/abs/1908.10084.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, 620, 2023.

Yixuan Su, Tian Lan, Yan Wang, Dani Yogatama, Lingpeng Kong, and Nigel Collier. A contrastive framework for neural text generation, 2022. URL https://arxiv.org/abs/2202.06417.

Tim Tomov, Dominik Fuchsgruber, and Stephan Günnemann. Task-awareness improves llm generations and uncertainty, 2026. URL https://arxiv.org/abs/2601.21500.

Roman Vashurin, Ekaterina Fadeeva, Artem Vazhentsev, Lyudmila Rvanova, Daniil Vasilev, Akim Tsvigun, Sergey Petrakov, Rui Xing, Abdelrahman Sadallah, Kirill Grishchenkov, Alexander Panchenko, Timothy Baldwin, Preslav Nakov, Maxim Panov, and Artem Shelmanov. Benchmarking uncertainty quantification methods for large language models with lm-polygraph. *Transactions of the Association for Computational Linguistics*, 13:220–248, 03 2025. ISSN 2307-387X. doi: 10.1162/tacl_a_00737. URL https://doi.org/10.1162/tacl_a_00737.

Henrique Schechter Vera, Sahil Dua, Biao Zhang, Daniel Salz, Ryan Mullins, Sindhu Raghuram Panyam, Sara Smoot, Iftekhar Naim, Joe Zou, Feiyang Chen, et al. Embeddinggemma: Powerful and lightweight text representations, 2025. URL https://arxiv.org/abs/2509.20354.

Nassim Walha, Sebastian G. Gruber, Thomas Decker, Yinchong Yang, Alireza Javanmardi, Eyke Hüllermeier, and Florian Buettner. Fine-grained uncertainty decomposition in large language models: A spectral approach, 2025. URL https://arxiv.org/abs/2509.22272.

Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, Cosmo Du, and Quoc V. Le. Long-form factuality in large language models, 2024. URL https://arxiv.org/abs/2403.18802.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Huggingface's transformers: State-of-the-art natural language processing, 2020. URL https://arxiv.org/abs/1910.03771.

Min-Hsuan Yeh, Max Kamachee, Seongheon Park, and Yixuan Li. Halluentity: Benchmarking and understanding entity-level hallucination detection, 2025. URL https://arxiv.org/abs/2502.11948.

A ADDITIONAL EXPERIMENTS AND ABLATIONS

Additional performance metrics We report the AUROC across the different datasets and model combinations in the hallucination detection experiment in Table 4. Overall, the picture remains similar to Table 2, with attribute-conditional VNE outperforming all other baselines in the token-level evaluation. In addition, once again, the combination of gradedness and structure awareness seems to be the decisive factor in increasing $\text{VNE}_{M|A}$ performance over graded yet structurally unaware measures (VNE_M) and non-graded yet structurally aware measures ($\text{SE}_{M|A}$).

Table 4: Token-level and entity-level AUROC across different methods, models, and datasets. Gold indicates best, silver second-best performance in the given setting.

		Token Level				Entity Level			
Model	Method	FS	LFo	HB	LF	FS	LFo	HB	LF
Mistral-7B-v0.3	PE	0.489	0.480	0.476	0.512	0.655	0.531	0.672	0.763
	CCP	**0.671**	**0.613**	**0.603**	**0.614**	0.674	0.546	0.663	0.768
	STC	0.492	0.480	0.589	0.596	0.654	0.529	0.655	0.751
	TokenSAR	0.491	0.476	0.476	0.528	0.665	0.521	0.602	0.754
	KLE	0.520	0.496	0.506	0.551	**0.692**	0.558	0.682	0.784
	SD	0.511	0.484	0.508	0.551	0.685	0.547	0.665	0.776
	SE_M	0.631	0.558	0.503	0.542	0.675	0.548	0.659	0.770
	$\text{SE}_{M\|A}$	0.651	0.579	0.561	0.600	0.650	**0.564**	**0.690**	**0.785**
	VNE_M	0.492	0.478	0.478	0.524	0.662	0.541	0.678	0.769
	$\text{VNE}_{M\|A}$	**0.700**	**0.664**	**0.629**	**0.653**	**0.686**	**0.574**	**0.728**	**0.828**
Gemma-4-12B	PE	0.576	0.588	0.460	0.474	0.575	**0.562**	0.658	0.768
	CCP	0.615	0.595	**0.586**	0.583	0.582	0.560	0.632	0.708
	STC	**0.702**	**0.691**	0.583	0.595	0.566	0.558	0.613	0.761
	TokenSAR	0.576	0.587	0.465	0.521	**0.600**	0.562	0.582	**0.773**
	KLE	0.596	0.598	0.494	0.524	**0.586**	0.562	0.661	**0.777**
	SD	0.585	0.588	0.500	0.532	0.577	0.555	0.637	0.747
	SE_M	0.612	0.578	0.501	0.521	0.562	0.546	0.641	0.764
	$\text{SE}_{M\|A}$	0.608	0.582	0.566	**0.606**	0.541	0.536	**0.672**	0.766
	VNE_M	0.569	0.592	0.472	0.501	0.573	**0.566**	0.665	0.772
	$\text{VNE}_{M\|A}$	**0.711**	**0.703**	**0.615**	**0.646**	0.583	0.559	**0.710**	0.772
Llama-3.1-8B	PE	0.521	0.561	0.518	0.591	0.667	0.577	0.662	0.776
	CCP	0.593	0.602	**0.592**	**0.638**	0.684	**0.608**	0.679	**0.782**
	STC	0.665	**0.668**	0.579	0.617	0.672	0.569	0.668	0.770
	TokenSAR	0.531	0.562	0.512	0.595	**0.703**	0.581	0.636	0.780
	KLE	0.565	0.583	0.537	0.610	**0.693**	0.605	0.669	0.766
	SD	0.552	0.572	0.536	0.604	0.687	0.595	0.673	0.770
	SE_M	0.640	0.604	0.548	0.605	0.682	0.595	0.658	0.761
	$\text{SE}_{M\|A}$	**0.666**	0.612	0.583	0.632	0.666	0.588	**0.680**	0.740
	VNE_M	0.525	0.562	0.517	0.595	0.681	0.588	0.670	0.781
	$\text{VNE}_{M\|A}$	**0.729**	**0.706**	**0.617**	**0.664**	0.688	**0.613**	**0.727**	**0.814**

Extending Figure 2, we provide the results across all models and datasets within Figure 3. As mentioned before, the advantage of the combination of gradedness and structure awareness persists across all settings.

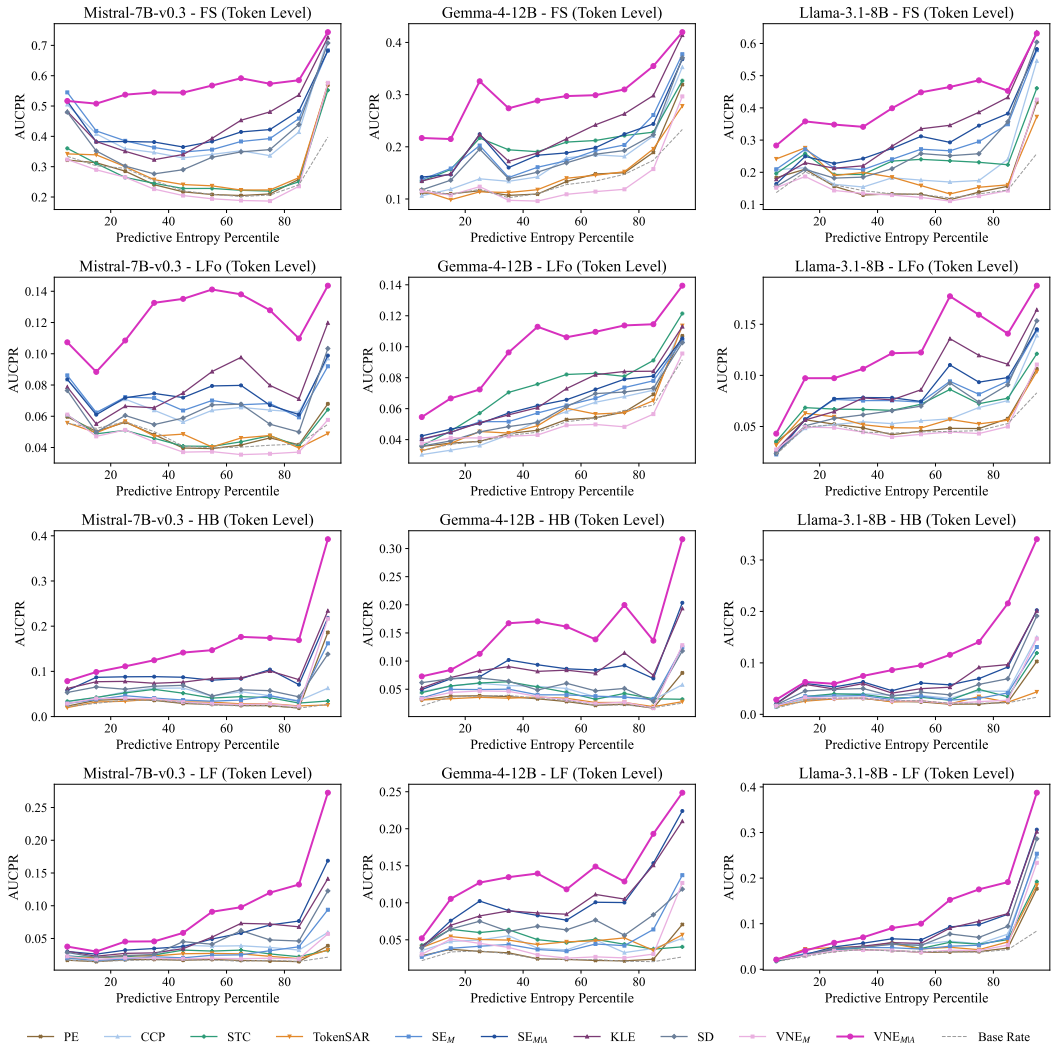

Figure 3: Improvement of VNE$_{M|A}$ over uncertainty baselines vs. predictive entropy across all models. The combination of gradedness and structure awareness is crucial for the predictive accuracy of methods.

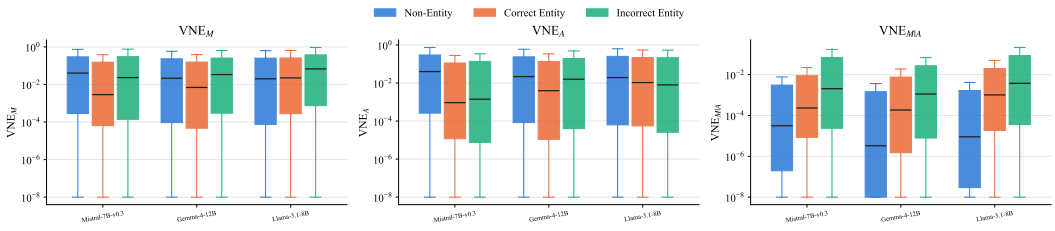

Figure 4: VNE$_M$, VNE$_A$, VNE$_{M|A}$ across different token types.

Token- vs. entity-level performance As already mentioned in Section 5.1, the performance gain of VNE$_{M|A}$ is less pronounced on the entity level compared to the token level. This decrease may

partly reflect the different evaluation setups. On the token level, we evaluate the discrimination capabilities w.r.t. incorrect tokens vs. correct and non-entity tokens, whereas on the entity level, we restrict the comparison to only entity tokens. Because non-entity tokens tend to have high attribute uncertainty (see Figure 4), non-structure-aware token-level methods report many false positives at these positions, whereas structure-aware measures suppress this signal by design. Restricting to only entity tokens removes this source of false positives for non-structure-aware measures and therefore narrows the advantage of structure-aware methods.

Top-N ablation We analyze the effect of the number of alternative tokens on the performance of attribute-conditional, kernel-based VNE across the FActScore and the LongFact objects datasets. As displayed in Figure 5, both the token-level and entity-level AUPR remain relatively stable across all N. On the entity level, performance improves with N, but tends to saturate after $N = 10$. One reason for this low sensitivity to N could be the relatively light tails of next-token distributions, s.t. additional, low-probability tokens contribute basically nothing to the scaled kernel matrices. Consequently, a small to moderate N is sufficient, keeping the computational overhead of the method low and making it practical for real-time use.

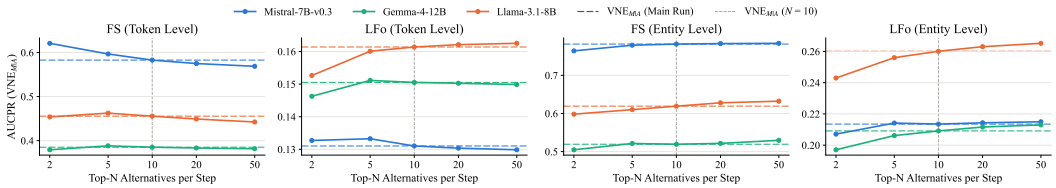

Figure 5: Effect of the considered number of alternatives N per step on the performance (AUPR) of the proposed method across two datasets and different models.

Performance vs. computational efficiency Since stepwise methods estimate uncertainty at every generation step, computational overhead is an important practical consideration. Figure 6 relates token-level AUPR to the average computation time per generated token for Mistral-7B-v0.3 on FS and LFo. Pairwise NLI-based methods (SE$_M$, SE$_{M|A}$, KLE, SD) are the most computationally expensive baselines, as they require $\mathcal{O}(N^2)$ NLI model passes. Methods relying on a linear number of online semantic comparisons or embeddings (VNE$_M$, CCP, TokenSAR) operate in a similar cost range as VNE$_{M|A}$ at $N = 10$, but consistently achieve lower predictive performance. STC is substantially more efficient than the other semantic methods by relying on precomputed token clusters, but provides only limited predictive performance in the token-level setting. Compared with VNE$_M$, VNE$_{M|A}$ incurs approximately twice the encoder cost due to the additional attribute encoder. However, with a shared hierarchical encoder or selective quantification, this computational overhead might be further reducible.

The effect of kernel parameters We analyze the effect of the γ parameter of the RBF-kernel on the performance of the kernel-based VNE$_{M|A}$. As Figure 8 shows, on the token level, the performance is generally highest when the attribute and meaning kernels operate at different scales, with a relatively large attribute γ_A and a smaller meaning γ_M. A large γ_A leads to a sharp separation of distinct attributes, whereas the small γ_M keeps the meaning space coarse, so only substantive semantic disagreement registers as meaning spread. However, the sensitivity to the kernel parameters depends on the model. Within Mistral-7B-v0.3 and Llama-3.1-8B, the kernel-based VNE$_{M|A}$ exhibits a relatively clear performance degradation with an increasing γ_M or a decreasing γ_A, whereas within Gemma-4-12B the performance is comparatively robust across a broad range of settings. Importantly, on the token level, a single shared setting of $\gamma_A = 7$ and $\gamma_M = 0.1$ is optimal in 5 of the 6 panels. On the entity level, a similar, yet less pronounced picture as on the token level arises, suggesting that the ideal balance between γ_A and γ_M can vary across models more decisively than at the token level. Based on these considerations, we use $\gamma_A = 7, \gamma_M = 1$ as defaults across experiments to account for token- and entity-level capabilities.

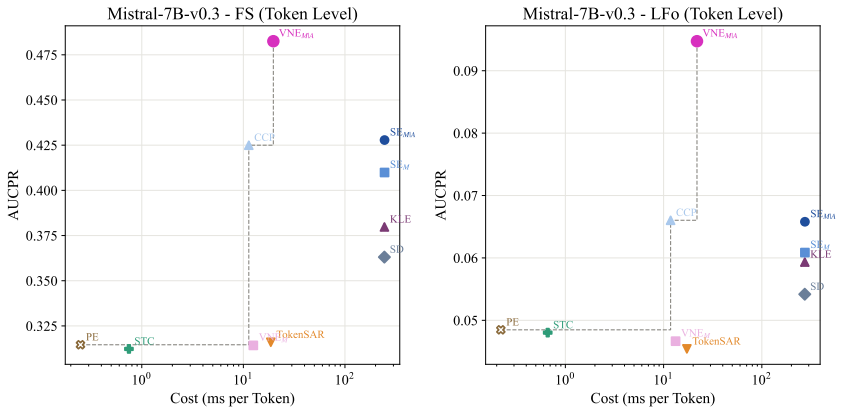

Figure 6: Predictive performance versus computational efficiency. Token-level AUCPR as a function of average computation time per generated token for Mistral-7B-v0.3 on FS and LFo. Lower cost and higher AUCPR are preferable. The dashed lines represent the Pareto front.

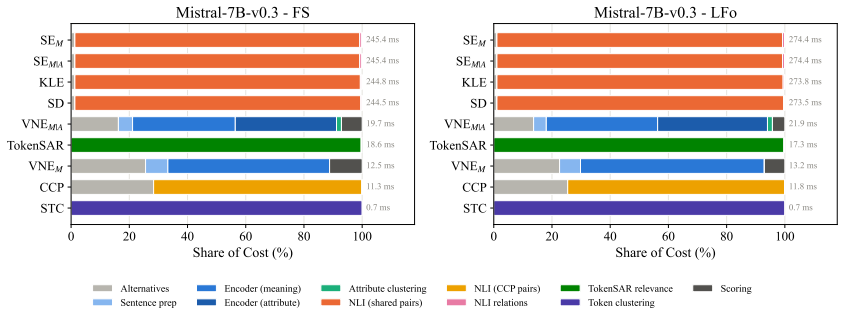

Figure 7: Computational cost breakdown of uncertainty methods on Mistral-7B-v0.3 for FS and LFo. Bars show each component's share of total cost, and the labels on the right give absolute runtime. NLI-based methods (SE, KLE, SD) spend almost all their time on pairwise NLI. VNE's cost is dominated by the encoder passes.

B Experiment Setup

B.1 Uncertainty Baselines

For all uncertainty baselines, we use the official implementations provided by the respective papers. To derive the localized versions of all baselines that do not quantify at the token level by design (see Table 1), we replace the sample of fully generated responses with the top-N single-token continuations. As an NLI model, we use DeBERTa-Large-MNLI (He et al., 2021). Whenever a baseline requires a set of alternative tokens, we use $N = 10$ as the number of alternative tokens provided at any step to the baselines. Additional method-specific parameters are reported per baseline.

Predictive entropy Predictive entropy (PE) quantifies the uncertainty of the next-token distribution directly at the level of surface forms. At each step t, we renormalize the probabilities of the top-N tokens as

$$\eta_j = \frac{p(y_t^j \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})}{\sum_{i=1}^{N} p(y_t^i \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})}. \tag{12}$$

The predictive entropy at step t is then the Shannon entropy of this renormalized distribution,

$$\mathrm{PE}_t = -\sum_{j=1}^{N} \eta_j \log \eta_j. \tag{13}$$

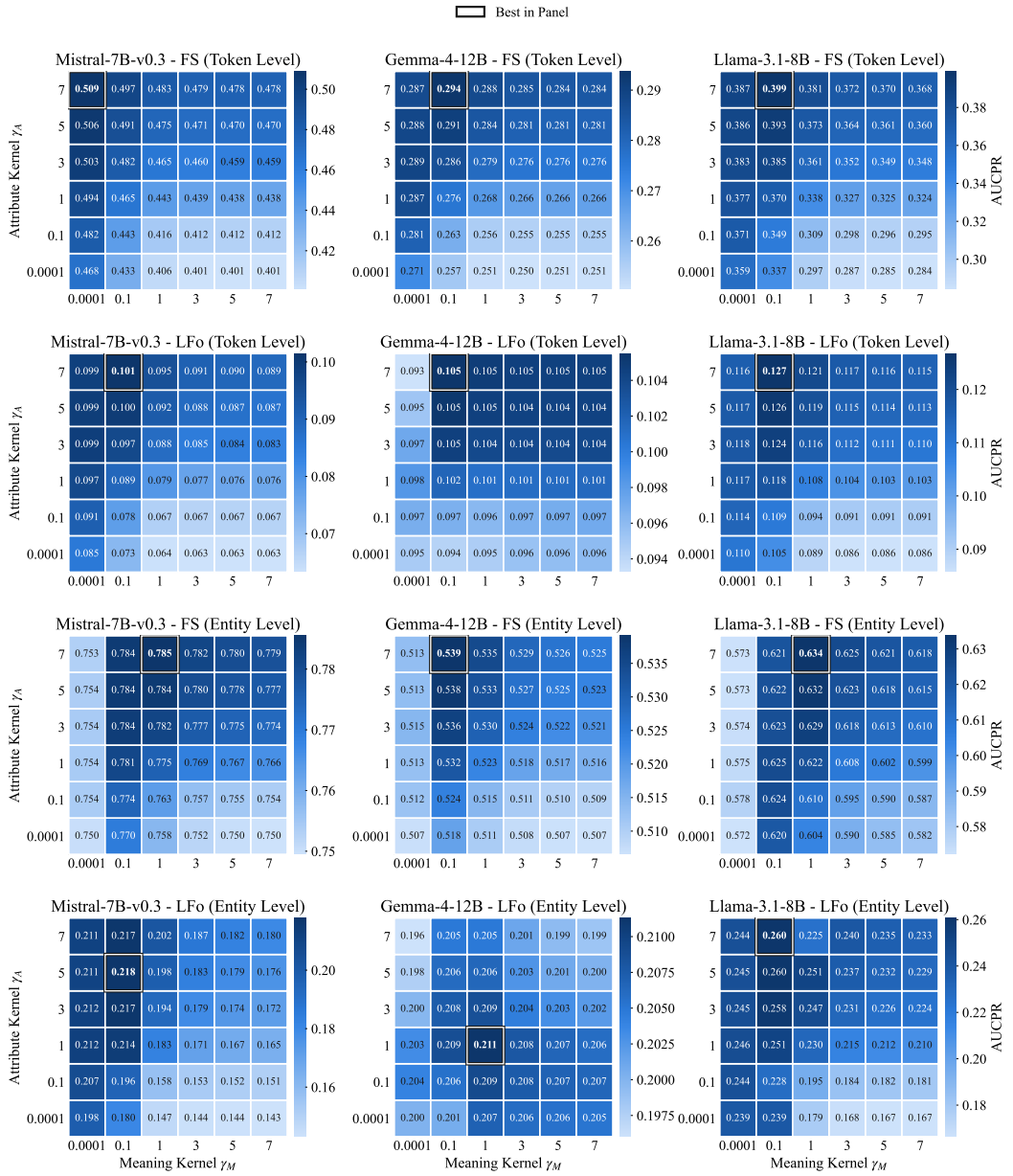

Figure 8: Token- and entity-level performance sensitivity to attribute- and meaning-kernel parameters.

PE thus corresponds to the first representation in Figure 1. It is also equivalent to the kernel-based VNE with an identity kernel matrix $K = I_N$, i.e., when every alternative is treated as semantically distinct from all others.

Claim-conditioned probability As already stated, Fadeeva et al. (2024) decompose a fully evolved answer into its individual claims. Within each claim, they then calculate the claim-conditioned probability for each chosen, non-function word w_t at position t as

$$\mathrm{CCP}_{\mathrm{word}}(w_t) = \frac{\sum_{n\in[N]:\,\mathrm{NLI}(\mathbf{w}_t^n,\,\mathbf{w}_t)=\mathrm{e}} p(w_t^n \mid \mathbf{w}_{<t})}{\sum_{n\in[N]:\,\mathrm{NLI}(\mathbf{w}_t^n,\,\mathbf{w}_t)\in\{\mathrm{e},\,\mathrm{c}\}} p(w_t^n \mid \mathbf{w}_{<t})}, \tag{14}$$

where $w_t^1, \ldots, w_t^N$ are the top-N tokens at position t, $\mathbf{w}_t^n = w_t^n \circ \mathbf{w}_{<t}$ is the concatenation of the word w_t^n and the prefix of the claim $\mathbf{w}_{<t}$and $\mathrm{NLI}(\mathbf{w}_t^n, \mathbf{w}_t) \in \{\mathtt{e}, \mathtt{c}, \mathtt{n}\}$ is the label of a natural language inference model indicating whether the alternative entails (e), contradicts (c), or is neutral towards (n) the generated word. Neutral alternatives are discarded, as they are taken to reflect uncertainty about *what* to claim rather than about the claim itself. Function words are assigned $\mathrm{CCP}_{\mathrm{word}} = 1$. The claim-level uncertainty for a claim C is then aggregated as

$$\mathrm{CCP}_{\mathrm{claim}}(C) = 1 - \prod_{w_t \in C} \mathrm{CCP}_{\mathrm{word}}(w_t). \tag{15}$$

Within our experiments, we use the default "no-context" setting within the provided implementation (Vashurin et al., 2025).

TokenSAR Duan et al. (2024) reweigh the token-level negative log-likelihood by each token's semantic relevance when the token is removed from the sentence:

$$R(y_t, \mathbf{y}, \mathbf{x}) = 1 - \left| g\big(\mathbf{x} \cup \mathbf{y},\ \mathbf{x} \cup \mathbf{y} \setminus \{y_t\}\big) \right|. \tag{16}$$

Here, $g(\cdot, \cdot)$ denotes a semantic similarity score obtained from an encoder model. The relevance scores are normalized across the complete generation,

$$\tilde{R}(y_t, \mathbf{y}, \mathbf{x}) = \frac{R(y_t, \mathbf{y}, \mathbf{x})}{\sum_{t'=1}^{T} R(y_{t'}, \mathbf{y}, \mathbf{x})}, \tag{17}$$

and used to reweigh the token-level negative log-likelihood, yielding the token-level uncertainty

$$\mathrm{TokenSAR}_t = -\log p(y_t \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})\, \tilde{R}(y_t, \mathbf{y}, \mathbf{x}), \tag{18}$$

which is summed over all tokens to obtain the sentence-level estimate. As an encoder model, we use a RoBERTa-large cross-encoder (Reimers & Gurevych, 2019), which yields the highest performance in the original TokenSAR implementation.

Semantic token clustering Cao et al. (2026) use precomputed token clusters to determine the probability mass of the semantic cluster of the generated token. To this end, all tokens of the vocabulary $\mathcal{V}$ are grouped using agglomerative clustering of their embeddings, yielding a map $\mathrm{cluster}(\cdot)$. At step t, the set of tokens considered semantically consistent with the generated token y_t is

$$\mathcal{T}_t = \underbrace{\{y \in \mathcal{V} : \mathrm{cluster}(y) = \mathrm{cluster}(y_t)\}}_{\mathcal{T}_t^e} \cup \underbrace{\{y \in \mathcal{V} : \mathrm{norm}(y_{t:}) \text{ starts with } \mathrm{norm}(y)\}}_{\mathcal{T}_t^p}, \tag{19}$$

where $\mathcal{T}_t^e$ contains the tokens sharing an embedding cluster with y_t, $\mathcal{T}_t^p$ contains the tokens that are a prefix of the remaining generation $y_{t:}$, and $\mathrm{norm}(\cdot)$ denotes case- and whitespace-insensitive normalization. The clustered probability is then

$$\hat{p}_c(y_t \mid \mathbf{y}_{<t}, \mathbf{x}) = \sum_{y \in \mathcal{T}_t} p(y \mid \mathbf{y}_{<t}, \mathbf{x}), \tag{20}$$

and the sequence-level uncertainty of a generation of length T is

$$\mathrm{S}(\mathbf{y}, \mathbf{x}) = 1 - \prod_{t=1}^{T} \hat{p}_c(y_t \mid \mathbf{y}_{<t}, \mathbf{x}). \tag{21}$$

In our experiments, we use the code and hyperparameters provided by the paper to obtain the LLM-tokenizer-specific clustering.

Semantic entropy Kuhn et al. (2023) quantify and report semantic uncertainty at the response level. Given a prompt $\mathbf{x}$, they sample N responses $\mathbf{y}^1, \ldots, \mathbf{y}^N$ and group them into semantic equivalence classes $c \in \mathcal{C}$, where two responses share a class if they entail each other in both directions according to an NLI model. The probability of a meaning class is obtained by summing over its members,

$$p(c \mid \mathbf{x}) = \sum_{n \in [N]:\ \mathbf{y}^n \in c} p(\mathbf{y}^n \mid \mathbf{x}), \tag{22}$$

and the semantic entropy is the Shannon entropy of the resulting distribution over meanings

$$\mathrm{SE}(\mathbf{x}) = -\sum_{c \in \mathcal{C}} p(c \mid \mathbf{x}) \log p(c \mid \mathbf{x}). \tag{23}$$

Kernel language entropy Nikitin et al. (2024) generalize semantic entropy by replacing the hard clustering of responses with a graded semantic kernel. Given a prompt $\mathbf{x}$, they sample N responses $\mathbf{y}^1, \dots, \mathbf{y}^N$ and build a weighted semantic graph over them, with edge weights derived from the NLI predictions in both directions,

$$W_{ij} = f\big(\mathrm{NLI}(\mathbf{y}^i, \mathbf{y}^j),\, \mathrm{NLI}(\mathbf{y}^j, \mathbf{y}^i)\big), \tag{24}$$

where f is a weighted sum of the predicted labels (1 for entailment, 0.5 for neutral and 0 for contradiction, summed over both directions). With the degree matrix $D_{ii} = \sum_j W_{ij}$ and the graph Laplacian $L = D - W$, a semantic kernel is obtained as the heat kernel

$$K_{\mathrm{heat}} = \exp(-\tau L), \tag{25}$$

with lengthscale $\tau > 0$. The kernel language entropy is the von Neumann entropy of the rescaled kernel matrix,

$$\mathrm{KLE} = \mathrm{VNE}(K_{\mathrm{heat}}) = -\operatorname{tr}\left[K_{\mathrm{heat}} \log K_{\mathrm{heat}}\right] = -\sum_{i=1}^{N} \lambda_i \log \lambda_i, \tag{26}$$

where $\lambda_1, \dots, \lambda_N$ are the eigenvalues of K_{heat}. In our implementation, we use $\tau = 0.3$, which aligns with the value provided by the paper.

Semantic density Qiu & Miikkulainen (2024) estimate the response-wise confidence for a specific target response $\mathbf{y}^*$ using a kernel density estimate in a semantic space. Given a prompt $\mathbf{x}$, they sample N unique reference responses $\mathbf{y}^1, \dots, \mathbf{y}^N$ with diverse beam search and define

$$\mathrm{SD}(\mathbf{y}^* \mid \mathbf{x}) = \frac{\sum_{i=1}^{N} p(\mathbf{y}^i \mid \mathbf{x})\, K(\mathbf{y}^*, \mathbf{y}^i)}{\sum_{i=1}^{N} p(\mathbf{y}^i \mid \mathbf{x})}, \tag{27}$$

where $p(\mathbf{y}^i \mid \mathbf{x})$ is the length-normalized sequence probability. The kernel is a dimension-invariant Epanechnikov kernel on the squared semantic distance d^2 between two responses,

$$K(\mathbf{y}^*, \mathbf{y}^i) = \big(1 - d^2(\mathbf{y}^*, \mathbf{y}^i)\big)\, \mathbf{1}\big[d(\mathbf{y}^*, \mathbf{y}^i) \le 1\big], \tag{28}$$

where the distance is not computed from explicit embeddings but estimated as its expectation under the class probabilities of an NLI model,

$$d^2(\mathbf{y}^*, \mathbf{y}^i) = p_{\mathrm{c}}(\mathbf{y}^*, \mathbf{y}^i \mid \mathbf{x}) + \tfrac{1}{2}\, p_{\mathrm{n}}(\mathbf{y}^*, \mathbf{y}^i \mid \mathbf{x}), \tag{29}$$

with p_{c} and p_{n} denoting the predicted probabilities of contradiction and neutrality, averaged over both directions. Equivalent, unrelated, and contradictory references thus contribute kernel values of 1, $\frac{1}{2}$, and 0, respectively.

B.2 Decoding baselines

For all decoding baselines, we use the official implementations provided by the Hugging Face `transformers` library (Wolf et al., 2020) via its `generate` interface (version 5.15.1). Unless stated otherwise, we keep the library's default hyperparameters and only set the method-specific parameters reported below.

Greedy decoding Greedy decoding deterministically selects the most likely token at each step,

$$y_t = \arg\max_{y \in \mathcal{V}} p(y \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta}), \tag{30}$$

until an end-of-sequence token or the maximum length is reached.

Sampling In sampling, each token is drawn from the (temperature-scaled) next-token distribution,

$$y_t \sim p_\tau(\cdot \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta}), \qquad p_\tau(y \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta}) \propto p(y \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta})^{1/\tau}, \tag{31}$$

with temperature $\tau > 0$.

Beam search Beam search maintains a set of B partial hypotheses $\{\mathbf{y}^b_{<t}\}_{b=1}^{B}$. At each step, every hypothesis is expanded by its top-N next-token candidates, and each continuation is scored by its cumulative log-likelihood,

$$S(\mathbf{y}^b_{<t} \circ y^j_t) = S(\mathbf{y}^b_{<t}) + \log p(y^j_t \mid \mathbf{y}^b_{<t}, \mathbf{x}, \boldsymbol{\theta}). \tag{32}$$

Only the B highest-scoring continuations out of the $B \cdot N$ candidates are retained. Completed hypotheses are ranked by their length-normalized score $S(\mathbf{y})/|\mathbf{y}|^\alpha$, where α is the length penalty. In our experiments, we use $B = 6$ and $\alpha = 1$.

Contrastive search Su et al. (2022) penalize token candidates whose representation is similar to the already generated context. At each step, the next token is chosen from the top-N candidates $\mathcal{V}^N$ as

$$y_t = \arg\max_{y \in \mathcal{V}^N} \Big\{ (1-\alpha)\, p(y \mid \mathbf{y}_{<t}, \mathbf{x}, \boldsymbol{\theta}) - \alpha \max_{1 \le j \le t-1} s\big(h_y, h^j_y\big) \Big\}, \tag{33}$$

where h_y denotes the last-layer hidden state of the model for the sequence $\mathbf{y}_{<t} \circ y$, h^j_y the hidden state of the j-th context token, and $s(\cdot,\cdot)$ the cosine similarity. The first term, the model confidence, favors likely candidates, while the second term, the degeneration penalty, discourages candidates that are close to the previous context. In our experiments, we use $N = 4$ and $\alpha = 0.6$, based on the settings provided by the paper. Note that, unlike our approach, contrastive search requires access to the internal states of the LLM.

DoLa Decoding by contrasting layers (DoLa) (Chuang et al., 2024) builds on the observation that semantic information tends to emerge within later transformer layers. It therefore contrasts the next-token distribution of the final layer L with that of an earlier layer. For any layer l, an early-exit distribution $p_l(\cdot \mid \mathbf{y}_{<t}) = \mathrm{softmax}\big(\varphi(h^l_t)\big)$ is obtained by applying a language head φ directly to the hidden state h^l_t. The premature layer M is selected dynamically at each step as the candidate layer whose distribution differs most from the final one,

$$M = \arg\max_{l \in \mathcal{L}} \mathrm{JSD}\big(p_L(\cdot \mid \mathbf{y}_{<t}) \,\|\, p_l(\cdot \mid \mathbf{y}_{<t})\big), \tag{34}$$

where $\mathcal{L}$ is a predefined set of candidate layers and $\mathrm{JSD}(\cdot,\cdot)$ refers to the Jensen-Shannon divergence. The next-token scores are then given by the log-ratio of both distributions,

$$\mathcal{F}(y_t) = \begin{cases} \log \dfrac{p_L(y_t)}{p_M(y_t)} & \text{if } y_t \in \mathcal{V}_{\text{head}}\,(y_t|\mathbf{y}_{<t}), \\ -\infty & \text{otherwise,} \end{cases} \tag{35}$$

with the adaptive plausibility constraint $\mathcal{V}_{\text{head}} = \{y \in \mathcal{V} : p_L(y) \ge \alpha \max_w p_L(w)\}$, which prevents implausible tokens from receiving high scores merely due to low premature-layer probability. The next token is selected greedily w.r.t. $\mathcal{F}$. Like contrastive search, DoLa relies on access to the model's internal states. In our experiments, we use the recommended settings for long-form generation, including $\alpha = 0.1$, and use the "low" mode of the implementation to consider the lower half of the layers to contrast with the final layer.

B.3 Evaluation

Handling off-policy datasets As stated in Section 5, within both prelabeled datasets, no on-policy run existed for Mistral-7B-v0.3 and Gemma-4-12B. Thus, to recover the stepwise token distributions for these two models, we used a teacher-forcing approach, using the pre-generated prefix from the dataset and prompt as inputs at each step t. To decrease the distribution shift as much as possible, we choose the closest available LLM that originally generated the response as a counterpart to our LLMs (Mistral-Small-24B-Instruct-2501 for Mistral-7B-v0.3, Gemma-2-9b-it for Gemma-4-12B).

Evaluation pipeline We use the FActScore pipeline (Min et al., 2023) for splitting the generated outputs into their individual claims. For the FActScore biography dataset, we use the FActScore fact-checking functionality to assess claim-level correctness, whereas for the LongFact dataset, we use the SAFE implementation (Wei et al., 2024). In addition, for each identified claim, we query another LLM to identify the defining entity and its corresponding tokens within that claim.

Entity Annotation System Prompt

```
You are a meticulous annotator in an evaluation pipeline for entity-level hallucination
    detection.

You receive:
1. A STATEMENT: one atomic fact extracted from a longer generated answer (a biography, a
    description ...). It may be true or false; that is irrelevant here.
2. The QUESTION in which context the answer was generated.
3. A list of CANDIDATE WORDS: the lowercased words of the sentence the statement was
    extracted from. That sentence usually packs several atomic facts, so most candidates
    belong to other statements.

Your task: decide which of the CANDIDATE WORDS form the ENTITY of this statement -- the
    most informative unit the statement asserts about its subject: a date, a place, a
    name, a nationality, a profession or role, a title or work, an organisation, a
    quantity or statistic, an award, a citation or reference, a descriptive attribute.
    It is the span that would be the hallucination if the statement were false, and the
    span that is confirmed if it is true.

Rules:
- Select ONLY from the provided CANDIDATE WORDS. Never introduce a word that is not in
    the list. Copy each selected word verbatim.
- Keep a unit whole: select every word of a multi-word entity ("december", "18", "1949";
    "cleveland", "ohio"; "television", "shows"; "acting", "career"; "general",
    "hospital").
- Do NOT select the subject of the statement (the person or thing it is about) or a
    pronoun standing for it. Those are given, not asserted.
- Do NOT select words the statement shares with the other facts of the same sentence and
    that carry no information of their own ("is", "was born", "appeared", "has"), unless
    the verb itself is the fact (e.g. "died", "retired", "married").
- Do NOT select articles, prepositions, conjunctions, auxiliaries or punctuation ("a",
    "an", "in", "on", "and", ","), not even between the words of a unit.
- If the statement asserts more than one unit (a date and a place), select all of them.
- If no candidate word forms the entity, return an empty list.

First think briefly in the `reasoning` field, then commit your selection.
```

Entity Annotation Few-Shot Examples

```
Example 1
STATEMENT: "Lanny Flaherty is an American."
CANDIDATE WORDS: ["lanny", "flaherty", "is", "an", "american", "actor", "born", "on",
     "december", "18", ",", "1949", ",", "in", "pensacola", ",", "florida", "."]
Correct selection: ["american"]   (the nationality is what this fact asserts; the name is
     the subject)

Example 2
STATEMENT: "Lanny Flaherty is an actor."
CANDIDATE WORDS: ["lanny", "flaherty", "is", "an", "american", "actor", "born", "on",
     "december", "18", ",", "1949", ",", "in", "pensacola", ",", "florida", "."]
Correct selection: ["actor"]

Example 3
STATEMENT: "Lanny Flaherty was born on December 18, 1949."
CANDIDATE WORDS: ["lanny", "flaherty", "is", "an", "american", "actor", "born", "on",
     "december", "18", ",", "1949", ",", "in", "pensacola", ",", "florida", "."]
Correct selection: ["december", "18", "1949"]   (the date is one unit; "born" and "on"
     are shared with the birthplace fact and carry nothing on their own)

Example 4
STATEMENT: "Lanny Flaherty was born in Pensacola, Florida."
CANDIDATE WORDS: ["lanny", "flaherty", "is", "an", "american", "actor", "born", "on",
     "december", "18", ",", "1949", ",", "in", "pensacola", ",", "florida", "."]
Correct selection: ["pensacola", "florida"]   (the place is one unit, without the comma)

Example 5
STATEMENT: "She began her acting career in the late 1960s."
CANDIDATE WORDS: ["she", "began", "her", "acting", "career", "in", "the", "late",
     "1960s", ",", "appearing", "in", "various", "television", "shows", "and", "films",
     "."]
Correct selection: ["late", "1960s"]   ("acting career" is what the whole sentence is
     about; the time is what this fact asserts)

Example 6
STATEMENT: "She has appeared in various television shows."
CANDIDATE WORDS: ["she", "began", "her", "acting", "career", "in", "the", "late",
     "1960s", ",", "appearing", "in", "various", "television", "shows", "and", "films",
     "."]
Correct selection: ["television", "shows"]   (one unit; "appearing" is shared with the
     films fact)

Example 7
STATEMENT: "He portrayed Joe Kelly on General Hospital."
CANDIDATE WORDS: ["he", "is", "best", "known", "for", "his", "roles", "in", "soap",
     "operas", ",", "including", "his", "portrayal", "of", "joe", "kelly", "on",
     "general", "hospital", "and", "ben", "gibson", "on", "knots", "landing", "."]
Correct selection: ["joe", "kelly", "general", "hospital"]   (two units, the role and the
     show; "ben gibson" and "knots landing" belong to another fact)
```

C IMPLEMENTATION DETAILS

Function word masking To handle both partial subword tokens and evolving meaning more robustly during encoding, we apply masks to the empirical kernel matrices. Following Fadeeva et al. (2024), we mask function words by assigning them low attribute and low meaning similarity to all other tokens. This follows the intuition that function words do not determine the meaning of an attribute and should therefore be considered neutral. Thereby, the function word mask ensures that both function words and punctuation contribute equally to the spread of the attribute and joint kernel matrices, thus contributing nothing to the attribute-conditional, kernel-based entropy.

C.1 FINE-TUNING

Fine-tuning data For fine-tuning of the embedding models, we need labeled sequence pairs, where the sequences should reflect the partially evolved, only differing in the last token setting that we encounter during generation, and where the labels should represent the meaning- or attribute-wise similarity. To obtain pairs, we truncate texts from the news and fiction categories of the Brown corpus, accessed via NLTK (Loper & Bird, 2002), at a given word. This truncated sentence then

Table 5: Substitution rules across word types.

Word Type	Equivalent	Neutral	Contradictory
Functional	Curated synonyms for prepositions and degree adverbs	Same-subtype swap	Curated antonyms for prepositions and degree adverbs
Noun	WordNet synonyms	Noun–adjective swap after an article	WordNet antonyms; truncated antonym / co-hyponym pairs
Adjective	WordNet synonyms	Adjective–noun swap after an article	WordNet antonyms
Adverb	WordNet synonyms	–	WordNet antonyms
Verb	WordNet synonyms	–	WordNet antonyms
Number	–	–	Last-digit change

serves as the first element of the pair. To construct the second one, we replace the last word of that truncated sequence with either an equivalent, a neutral, or a contradictory word using established relations from WordNet (Miller, 1994). The substitution rules depend on the word type listed in Table 5. The labels are then constructed using the underlying semantic relation between the original word and the chosen alternative. For the meaning-similarity, equivalent pairs receive a similarity of 1; everything else receives a similarity of 0. For the attribute-similarity, both equivalent and contradictory pairs receive a similarity of 1, whereas neutral alternatives receive a similarity of 0, thereby reflecting the theoretical considerations from Section 4. To account for subword tokenization, i.e., tokens that correspond to word fragments rather than complete words, we additionally truncate the last word in noun antonym pairs, mimicking partially generated words. Finally, to obtain additional training data that matches the structures seen during generation as closely as possible, we generate answers to questions from the TriviaQA (Joshi et al., 2017) training split using Mistral-7B-v0.3 and, at each generation step, construct training pairs from the top-N alternative tokens. These pairs are labeled with DeBERTa-Large-MNLI (He et al., 2021), and the resulting labels are manually verified. In the end, we thus obtained 34341 training pairs, 9814 test pairs, and 4906 validation pairs, which are then used during fine-tuning. Note that the data used during fine-tuning is disjoint from all downstream evaluation data used in Section 5.

Fine-tuning details As a base for both the attribute encoder f_A and the meaning encoder f_M, we use the EmbeddingGemma-300M (Vera et al., 2025), and apply the same fine-tuning procedure for both. More specifically, we insert LoRA (Hu et al., 2021) adapters into the query, key, and value of every attention layer, thereby only training the adapter weights and freezing the rest of the parameters. For any training pair $(\mathbf{y}^j_{\leq t}, \mathbf{y}^i_{\leq t})$ with corresponding attribute label r_a^{ji} and meaning label r_m^{ji}, we minimize the cosine loss

$$\mathcal{L}_{\text{emb},a} = \left(\cos\left(f_A(\mathbf{y}^j_{\leq t}), f_A(\mathbf{y}^i_{\leq t})\right) - r_a^{ji}\right)^2, \quad \mathcal{L}_{\text{emb},m} = \left(\cos\left(f_M(\mathbf{y}^j_{\leq t}), f_M(\mathbf{y}^i_{\leq t})\right) - r_m^{ji}\right)^2. \tag{36}$$

As an optimizer, we use AdamW (Loshchilov & Hutter, 2019) (learning rate: 2×10^{-5}, warmup schedule: 10%) and fine-tune for three epochs with a batch size of 64. We employ hyperparameter tuning for the LoRA rank (16, 32, 64), where the final model is selected using the independent, held-out test set (selected rank for both f_A, f_M : 16). Both the validation metric (computed on the validation set; used for returning the best model during fine-tuning) and model selection (computed on the held-out test set) are, however, not based on the training loss, but on a margin criterion. More specifically, we use the separation that the resulting embedding space induces between equivalent and different-meaning (neutral and contradictory) pairs, or between same-attribute (equivalent meaning and contradictory meaning) and different-attribute pairs (neutral pairs), with higher values indicating better separation.

Compute resources All experiments, including the generation of responses, the stepwise uncertainty computation, the decoding experiments, and the fine-tuning of both encoders were run on a single NVIDIA A100 GPU (80 GB).