Measuring Internal Probabilistic Activity with Variational Distributional Neurons

Yves Ruffenach[1,*]

[1]Conservatoire National des Arts et Métiers, Strasbourg, France

ORCID: 0009-0009-4737-0555

DOI: 10.5281/zenodo.21668942

[*]Correspondence: yves@ruffenach.net

Abstract

Variational distributional neurons make internal uncertainty directly measurable by representing each activation as a learned posterior distribution conditioned on the current input. Elemental Variational Expanse (EVE) implements this principle in a language-model unit that exposes local Kullback–Leibler (KL) activity, posterior scale, posterior mean energy, and active-unit fraction during the forward computation. A controlled next-token experiment compares EVE with a deterministic model and Monte Carlo (MC) Dropout across five shared random seeds, with a five-member deterministic ensemble as an additional predictive reference. Raw EVE attains a mean MC negative log-likelihood of 5.359, compared with 5.620 for the deterministic model and 5.621 for MC Dropout; mean perplexity is 212.743 versus 275.845 and 276.161, mean accuracy is 0.166 versus 0.148 and 0.153, and mean tail-aware CVaR-NLL is 12.713 versus 13.200 and 13.213. Mean expected calibration error remains aligned at 0.042, compared with 0.043 and 0.046. Validation-selected readouts provide explicit calibration-oriented, balanced, and tail-oriented operating regimes. As context-token perturbation increases from 0.05 to 0.30, mean internal KL rises from 0.291 to 0.362, posterior scale from 0.874 to 0.899, and posterior mean energy from 0.052 to 0.075. These coordinated measurements demonstrate that unit-level posterior activity can be quantified alongside predictive behavior in a language-model computation and provide a concrete measurement basis for scaling variational distributional units to larger model architectures.

Keywords: variational distributional neurons; language models; internal probabilistic activity; uncertainty quantification; variational inference; local latent variables; posterior diagnostics; Bayesian deep learning; calibration; tail-aware evaluation; distribution shift; foundation-model architectures; deep ensembles; Monte Carlo Dropout; Gaussian Process Neurons

1 Introduction

Modern language models, including foundation-model architectures, emerge from a long development of neural computation. Early artificial neurons, including threshold units and perceptrons, transformed an input into a deterministic activation. Multilayer networks extended this primitive through trainable compositions, and gradient-based learning made internal representations adaptable to data. Transformer architectures later organized these transformations through self-attention and large-scale pre-training, supporting general-purpose models for language, vision, code, scientific modeling, and multimodal reasoning (Vaswani et al., 2017; Brown et al., 2020; Bommasani et al., 2021).

Probabilistic neural modeling added several complementary locations for uncertainty. Energy-based networks introduced sampled internal states; Bayesian neural networks represented distri-

butions over parameters; latent-variable models represented distributions over hidden variables; Monte Carlo Dropout (MC Dropout) sampled masked network realizations; and deep ensembles represented variability across independently trained models (Gal and Ghahramani, 2016; Lakshminarayanan et al., 2017). Variational inference supplied a scalable framework for learning approximate posterior distributions, while reparameterization enabled stochastic latent states to participate in ordinary gradient-based optimization (Kingma and Welling, 2014; Rezende et al., 2014).

These developments established a hierarchy of probabilistic objects spanning parameters, latent variables, network realizations, model collections, and predictive outputs. Variational distributional neurons add the activation state of the individual computational unit to this hierarchy. The unit carries an input-conditioned posterior state throughout the forward computation, making its probabilistic activity directly observable.

This level of measurement complements established foundation-model evaluation. Scaling laws relate aggregate loss to model size, data, and compute (Kaplan et al., 2020; Hoffmann et al., 2022). Cross-entropy measures the probability assigned to observed targets, perplexity expresses the same quantity on an exponential scale, and accuracy records the proportion of correct predictions. Expected calibration error (ECE) compares predictive confidence with empirical correctness. Predictive entropy describes the dispersion of the output distribution, while disagreement across stochastic samples or ensemble members describes variation among predictions.

Output-level quantities characterize predictive behavior. Internal posterior measurements extend this description by identifying probabilistically active units, posterior concentration, divergence from a reference prior, and changes in local computational state across statistical conditions. Distribution shift denotes a change between the conditions represented in training data and those encountered during validation, testing, or deployment. Output measurements characterize the predictive response to this change, while internal measurements characterize the corresponding evolution of the computation.

Elemental Variational Expanse (EVE) implements this principle at the level of the neural unit. Each EVE unit represents its activation through a learned probability distribution conditioned on the current hidden representation. The unit produces a posterior mean and posterior scale, samples a latent activation through reparameterization, and propagates the sampled state to the next stage of the network.

The posterior mean represents the structured activation associated with the current input. The posterior scale represents the dispersion of possible activation states. The sampled latent activation supplies the state used by the forward computation. The local KL divergence measures the relation between the learned posterior and its reference prior. Together, these quantities form native internal observables.

Established uncertainty methods occupy complementary levels of the probabilistic hierarchy. A Bayesian neural network represents distributions over parameters. MC Dropout represents variability across sampled masks and subnetworks. A deep ensemble represents variability across independently trained models. A probabilistic prediction head represents a distribution over outputs. An EVE unit represents a distribution over the activation generated for the current input. These levels can be studied independently or composed within the same architecture.

Internal and output-level measurements are organized along three complementary dimensions:

1. **Internal probabilistic activity.** Local KL divergence, posterior scale, posterior mean energy, and active-unit fraction characterize the distributional state maintained by the EVE units.
2. **Matched predictive comparison.** A shared teacher-forcing procedure compares EVE with a deterministic model, MC Dropout, and a deterministic deep ensemble. Teacher

forcing supplies the observed preceding tokens as context for prediction of the following token.

3. **Readout and distribution-shift analysis.** Validation-selected probabilistic readouts connect the internal state to predictive likelihood, calibration, accuracy, stochastic disagreement, and tail-aware performance. Controlled changes in input and target distributions reveal the joint evolution of internal and output-level observables.

Output metrics characterize predictive quality and reliability, while unit-level posterior statistics characterize the probabilistic computation from which those predictions emerge. Together, these observables make internal probabilistic activity an explicit and measurable variable in language-model computation and define a unit-level measurement principle that can be extended to larger foundation-model architectures.

2 Related work

2.1 Historical and conceptual lineages of neural computational units

Deterministic threshold units and gradient-based learning. McCulloch and Pitts formalized idealized neurons as binary threshold elements and linked networks of such units with propositional logic (McCulloch and Pitts, 1943). Rosenblatt's perceptron introduced data-driven adaptation of connection weights while retaining a thresholded scalar response at the unit level (Rosenblatt, 1958). Reverse-mode differentiation supplied an efficient basis for propagating derivatives through composed computations, and early neural applications developed this principle for adaptive multilayer systems (Linnainmaa, 1976; Werbos, 1974). Rumelhart, Hinton, and Williams subsequently established backpropagation as a practical and widely adopted procedure for learning distributed representations in multilayer networks (Rumelhart et al., 1986). Sigmoid, hyperbolic-tangent, rectified, gated, recurrent, convolutional, and transformer architectures preserve the same elementary pattern: fixed parameters map the current input state to a deterministic activation value or vector. Transformer feed-forward blocks retain this primitive while self-attention supplies context-dependent interaction among token representations (Vaswani et al., 2017).

Energy-based networks and stochastic states. Hopfield networks introduced an energy-based description of recurrent neural dynamics in which stored patterns correspond to stable attractor states (Hopfield, 1982). Boltzmann machines extended this lineage through stochastic binary units and an energy-based joint probability distribution (Ackley et al., 1985). These systems established sampled internal states as computational objects and connected neural dynamics with statistical mechanics.

Gaussian basis functions and Bayesian parameter uncertainty. Radial basis function (RBF) networks introduced localized hidden-unit responses, commonly through deterministic Gaussian basis functions centered in input space (Broomhead and Lowe, 1988; Moody and Darken, 1989; Park and Sandberg, 1991). The Gaussian form in an RBF unit describes the geometry of a deterministic response surface. Bayesian neural networks (BNNs) introduced a different Gaussian lineage by placing prior and posterior distributions over model parameters (MacKay, 1992; Neal, 1992). Their uncertainty resides in weight space and induces a distribution over network functions.

Directed stochastic networks and learned approximate inference. The Helmholtz machine combined a stochastic generative pathway with a recognition pathway that approximated inference over multilayer latent variables (Dayan et al., 1995). The wake–sleep algorithm supplied alternating learning phases for the generative and recognition models (Hinton et al., 1995). Later stochastic feed-forward networks combined deterministic transformations with sampled hidden variables for conditional distribution modeling (Tang and Salakhutdinov, 2013), while gradient-estimation methods broadened the trainable forms of discrete stochastic computation (Bengio et al., 2013).

Variational latent-variable computation. Variational autoencoders (VAEs) established a scalable pattern for continuous latent-variable learning: an inference network predicts an approximate posterior, a reparameterized sample enters the generative computation, and the evidence lower bound (ELBO) combines data fit with KL regularization (Kingma and Welling, 2014; Rezende et al., 2014). Variational BNNs applied related methods to posterior distributions over weights (Graves, 2011; Blundell et al., 2015). The local reparameterization trick represented sampled weight uncertainty through local activation noise during optimization, improving the statistical efficiency of gradient estimates while retaining a parameter posterior as the underlying probabilistic object (Kingma et al., 2015).

Predictive uncertainty methods. Dropout samples Bernoulli masks during training and regularizes feature co-adaptation (Srivastava et al., 2014). Monte Carlo Dropout retains those masks during evaluation and estimates predictive uncertainty from repeated masked forward passes (Gal and Ghahramani, 2016). Mixture density networks map deterministic hidden states to parameters of a predictive mixture distribution (Bishop, 1994). Input-dependent heteroscedastic heads similarly represent observation-dependent predictive dispersion (Kendall and Gal, 2017). Deep ensembles aggregate predictive distributions from independently optimized models (Lakshminarayanan et al., 2017). These methods place probability at the level of masks, outputs, or model collections.

2.2 Comparative positioning of variational distributional neurons

Activation-state posteriors and Bayesian neural networks. A BNN learns a posterior distribution over weights, conventionally written as $q(W)$, and integrates predictions across parameter configurations. An EVE unit learns an amortized posterior over its activation state, written as $q_\theta(z_u \mid h)$ for unit u and current representation h. BNN KL terms describe divergence in parameter space; EVE KL terms describe divergence in input-conditioned activation space. The two constructions occupy distinct and composable levels: parameter uncertainty can coexist with distributional activation states.

Activation-state posteriors and local reparameterization. Local reparameterization converts uncertainty from a global weight posterior into local activation noise for efficient gradient estimation. In EVE, the local posterior is the modeled computational state itself. Its mean, scale, sampled value, and KL activity remain explicit observables during validation, testing, and distribution-shift analysis. The shared reparameterization operation therefore serves different probabilistic objects: a computational estimator of weight uncertainty in a BNN and an input-conditioned activation distribution in EVE.

Activation-state posteriors and MC Dropout. MC Dropout represents variation across masked subnetworks. Each sampled mask selects a network realization, and predictive statistics

are estimated across repeated realizations. EVE represents a learned continuous posterior within the activation process of a single unit. Posterior mean, posterior scale, local KL, and sample disagreement are native state variables of that unit. MC Dropout and EVE consequently measure complementary forms of stochastic computation.

Activation-state posteriors, ensembles, and predictive heads. Deep ensembles represent diversity across independently trained model functions. Mixture and heteroscedastic heads represent probability at the predictive surface. EVE represents probability inside hidden computation and propagates a sampled local state. An ensemble may contain EVE models, and an EVE network may terminate in a probabilistic prediction head, placing uncertainty at several levels of the same system.

Gaussian processes and Gaussian Process Neurons. Gaussian processes (GPs) define distributions over functions, with posterior uncertainty determined by a prior covariance structure and observed data. Deep Gaussian processes compose multiple GP mappings into hierarchical probabilistic models (Damianou and Lawrence, 2013). Gaussian Process Neurons (GPNs) place a GP prior over neuron-specific activation functions and infer posterior distributions over those functions through sparse variational approximations (Urban et al., 2017). Their stochastic object is the activation function implemented by the neuron. EVE uses deterministic maps to parameterize a posterior over the latent activation produced for the current input. GPNs therefore express function-space uncertainty over activation laws, whereas EVE expresses state-space uncertainty over activations generated under a learned law.

Variational distributional neurons. A variational distributional neuron combines four properties within one computational primitive. It is *stochastic* because the forward state includes a sampled latent variable; *Gaussian* because the local approximate posterior is parameterized as a Gaussian distribution; *variational* because learning includes a prior-relative KL term; and *distributional* because the activation is represented by a posterior state comprising a mean, a scale, and sampled realizations. The defining object is the local posterior $q_\theta(z_u \mid h)$ embedded directly in the forward computation. Its parameters and divergences provide measurable internal variables across inputs, layers, training regimes, and distribution shifts.

Foundation-model measurement. Transformer language models and foundation models are commonly characterized through likelihood, perplexity, calibration, benchmark performance, entropy, and predictive disagreement (Devlin et al., 2019; Brown et al., 2020; Bommasani et al., 2021). Scaling laws connect aggregate loss with model size, data, and compute (Kaplan et al., 2020; Hoffmann et al., 2022). Variational distributional units add unit-level posterior observables to this measurement space. The historical progression links deterministic transformations, energy-based states, Gaussian response geometry, Bayesian parameter distributions, amortized latent posteriors, stochastic predictive methods, and input-conditioned activation distributions within a unified probabilistic hierarchy.

3 Internal probabilistic activity

Let h denote a hidden representation entering a local language-model unit. A deterministic unit computes a transformed activation $a = f_\theta(h)$. In EVE, the unit parameterizes a learned approximate posterior,

$$q_\theta(z \mid h) = \mathcal{N}(\mu_\theta(h), \mathrm{diag}(\sigma^2_\theta(h))), \tag{1}$$

and samples a latent activation through

$$z = \mu_\theta(h) + \sigma_\theta(h) \odot \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I). \tag{2}$$

The prior is a standard normal distribution. For each unit, the local KL divergence is

$$\mathrm{KL}(q_\theta(z \mid h) \| p(z)) = \frac{1}{2} \sum_j \left(\mu_j^2 + \sigma_j^2 - \log \sigma_j^2 - 1 \right). \tag{3}$$

This posterior defines three example-level observables, where U is the number of measured units:

$$A_{\mathrm{KL}}(x) = \frac{1}{U} \sum_{u=1}^{U} \mathrm{KL}_u(x), \tag{4}$$

$$A_\sigma(x) = \frac{1}{U} \sum_{u=1}^{U} \sigma_u(x), \tag{5}$$

$$A_{\mu^2}(x) = \frac{1}{U} \sum_{u=1}^{U} \mu_u(x)^2. \tag{6}$$

Active-unit fraction is computed over the evaluation set. Let N denote the number of evaluated examples and let

$$\overline{\mathrm{KL}}_u = \frac{1}{N} \sum_{i=1}^{N} \mathrm{KL}_u(x_i). \tag{7}$$

A unit is active when $\overline{\mathrm{KL}}_u > \tau_{\mathrm{active}}$, with $\tau_{\mathrm{active}} = 10^{-4}$, and

$$A_{\mathrm{active}} = \frac{1}{U} \sum_{u=1}^{U} \mathbf{1}\left\{ \overline{\mathrm{KL}}_u > \tau_{\mathrm{active}} \right\}. \tag{8}$$

The active-unit fraction therefore records the proportion of units whose evaluation-set mean KL activity exceeds the fixed threshold.

Local reconstruction regularization. Each variational unit includes a neuron-specific decoder. A learned projection of the incoming representation supplies a local target, and the decoder reconstructs that target from the unit's latent state. The target is detached from the reconstruction gradient, and the regularizer is the mean squared reconstruction error averaged across local output dimensions and units. This term encourages the sampled state to retain input-dependent information within each local variational pathway.

Homeostatic control. A homeostatic controller tracks an exponential moving average of posterior mean energy for each unit. The configured target energy is 0.10, with a maintained band from 0.08 to 0.145. After a 200-step warm-up, bounded row-wise rescaling of the posterior-mean projection is applied every 10 optimization steps with update strength 0.22 and a maximum multiplicative step of 1.35. This control maintains a stable distribution of local activation energy while preserving gradient-based learning of the posterior parameters.

The internal observables complement output-level likelihood, calibration, sample disagreement, and tail-aware measurements. Their joint evaluation connects predictive behavior with the posterior states maintained during computation.

4 Materials and Methods

Task and data. A controlled next-token language-modeling experiment uses the RLAIF/WritingPrompts-Filtered dataset, where RLAIF denotes Reinforcement Learning from AI Feedback, with Generative Pre-trained Transformer 2 (GPT-2) tokenization (RLAIF Team, 2025). The experiment uses 1% of the dataset with a fixed 70/15/15 train-validation-test split. Prompt and story sequences are truncated to 96 and 192 tokens, respectively. Teacher-forcing windows use context length 24 and target stride 2. The resulting subset contains 134,086 training windows, 28,662 validation windows, and 28,898 test windows.

Models and training configuration. All models use the same data split, tokenization, teacher-forcing windows, random seeds, four-epoch budget, and batch size. This protocol matching aligns the experimental conditions while reporting architectural capacity explicitly. The deterministic baseline (DET) uses a deterministic language-model unit. Its decoder dropout probability is $p = 0.05$ during training. Monte Carlo (MC) Dropout reuses each trained DET checkpoint and reactivates the same dropout modules with $p = 0.05$ during evaluation. The deterministic ensemble averages five independently trained DET members. EVE replaces the deterministic unit with a local variational unit exposing posterior mean, posterior scale, and local KL activity. The EVE architecture uses 1,024 units, three densely connected variational multilayer perceptron (MLP) stages, a transformer-style block with 12 attention heads, and a gated decoder. Dropout is 0.05 in the variational MLP, attention, residual, and decoder pathways.

The frozen Generative Pre-trained Transformer 2 (GPT-2) embedding matrix contributes 38,597,376 shared non-trainable parameters to each model. DET and MC Dropout contain 20,715,265 trainable parameters, for 59,312,641 parameters per model including the frozen embeddings. EVE contains 134,852,868 trainable parameters, for 173,450,244 parameters including the frozen embeddings. Each deterministic ensemble member has the DET parameter count; the five-member collection therefore comprises five independently optimized DET parameter sets.

Training uses batch size 48, four epochs, mixed-precision computation, weight decay 10^{-4}, and gradient clipping at 1.0. The EVE learning rate is 1.5×10^{-4} and the deterministic learning rate is 2×10^{-4}. The initial KL-controller coefficient is 2×10^{-4}. Local reconstruction regularization begins at the first optimization step, increases linearly over 400 steps to weight 0.03, and uses weight 0.022 from epoch 3. Posterior scale is capped at 7.0 in the first phase and 5.8 from epoch 3. The homeostatic controller uses the settings defined above. EVE uses four latent samples per optimization step. Predictive metrics use one forward pass for DET, four stochastic passes for MC Dropout, four stochastic passes for EVE, and one pass through each of the five deterministic ensemble members. The validation-selected EVE readouts reuse the same four-sample posterior bank and require no additional parameter training. The base five-seed runner—five EVE trainings, five DET trainings, MC Dropout evaluations from the DET checkpoints, and deterministic ensemble evaluation—completed in 1.92 hours on one CUDA device with mixed precision.

Table 1: Model capacity and computational profile. Predictive passes are counted per evaluated item for the reported metrics.

Model	Trainable parameters	Frozen parameters	Training requirement	Predictive passes
DET	20,715,265	38,597,376	One model per seed, four epochs	1
MC Dropout	20,715,265	38,597,376	Reuses the corresponding DET checkpoint	4
EVE	134,852,868	38,597,376	One model per seed, four epochs; four latent samples per step	4
DET Ensemble	20,715,265 per member	38,597,376 per member	Five independently trained DET members	5

Validation-selected readouts. Let $S = 4$ denote the number of stochastic EVE predictions, let ℓ_s be the logit vector from sample s, and let

$$p_s^{(T)} = \mathrm{softmax}(\ell_s / T) \tag{9}$$

be the sample probability vector after temperature scaling. Five readouts transform the same sample bank into one predictive distribution. Mean probabilities use $S^{-1} \sum_s p_s^{(T)}$. Softmax of mean logits uses $\mathrm{softmax}(S^{-1} \sum_s \ell_s / T)$. Median probabilities take the coordinate-wise median across the S probability vectors and renormalize the result to unit mass.

Entropy-weighted and confidence-weighted readouts assign sample-specific weights. For entropy weighting, the entropy of sample s is

$$H_s = -\sum_v p_{s,v}^{(T)} \log p_{s,v}^{(T)}, \tag{10}$$

and the normalized weight is

$$w_s^{(H)} = \frac{\exp(-H_s/\tau_w)}{\sum_{r=1}^{S} \exp(-H_r/\tau_w)}. \tag{11}$$

For confidence weighting, $c_s = \max_v p_{s,v}^{(T)}$ and

$$w_s^{(C)} = \frac{\exp(c_s/\tau_w)}{\sum_{r=1}^{S} \exp(c_r/\tau_w)}. \tag{12}$$

Both weighted readouts use $\tau_w = 0.25$ and compute the final distribution as $\sum_s w_s p_s^{(T)}$, followed by numerical renormalization.

Temperature and readout are selected exclusively on validation data. Lower scores are preferred. The calibration-oriented score is

$$S_{\mathrm{ECE}} = \mathrm{ECE} + 0.01\,\mathrm{NLL} + 0.001\max(0, \text{CVaR-NLL} - \mathrm{NLL}), \tag{13}$$

the tail-oriented score is

$$S_{\mathrm{CVaR}} = \text{CVaR-NLL} + 0.05\,\mathrm{NLL} + 0.25\,\mathrm{ECE}, \tag{14}$$

and the balanced score is

$$S_{\mathrm{bal}} = \mathrm{NLL} + 2\,\mathrm{ECE} + 0.05\max(0, \text{CVaR-NLL} - \mathrm{NLL}). \tag{15}$$

The search covers temperatures $T \in \{0.8, 0.9, 1.0, 1.1, 1.2, 1.35, 1.5, 1.75, 2.0, 2.25, 2.5, 3.0\}$ and the five readouts defined above. The selected pairs are applied unchanged to the test set. The ECE-selected policy uses $T = 1.1$ with mean probabilities, the CVaR-selected policy uses $T = 3.0$ with median probabilities, and the balanced policy uses $T = 1.2$ with confidence-weighted samples.

Metrics. Metrics include cross-entropy (CE), perplexity (PPL), accuracy (Acc.), Monte Carlo negative log-likelihood (MC-NLL), expected calibration error (ECE), mutual information (MI), top-1 flip rate across stochastic predictions, and Conditional Value-at-Risk negative log-likelihood (CVaR-NLL) for the 5% upper-loss tail. CVaR-NLL is the mean negative log-likelihood among the 5% of evaluated examples with the highest individual NLL. Internal EVE measurements include mean KL activity, mean posterior scale, posterior mean energy, and active-unit fraction as defined above.

Distribution-shift evaluation. Four complementary views characterize probabilistic activity across controlled forms of distribution shift: context-token perturbation at rates 0.05, 0.15, and 0.30 with unchanged labels; rare-versus-common target-token slices; ordered evaluation bins that characterize variation across successive portions of the test set; and shifted-validation recalibration, in which the EVE readout and temperature are selected on shifted validation and applied unchanged to shifted test. The multi-axis evaluation includes DET over five deterministic seeds, MC Dropout over the same five deterministic checkpoints with dropout active at evaluation time, and EVE over five variational seeds.

5 Results

Predictive quality and internal activity. Across five shared seeds, raw EVE attains mean MC-NLL 5.359 ± 0.039, compared with 5.620 ± 0.022 for DET and 5.621 ± 0.021 for MC Dropout. Mean perplexity is 212.743 ± 8.288, compared with 275.845 ± 6.015 and 276.161 ± 5.894. Mean accuracy reaches 0.166 ± 0.002, and mean CVaR-NLL is 12.713 ± 0.212, while mean ECE remains aligned at 0.042 ± 0.009. The same runs expose local KL activity, posterior scale, posterior mean energy, and active-unit fraction during validation and testing (Appendices A and B).

Complementary readout regimes. Validation selection identifies three operating regimes derived from the same learned variational state. The calibration-oriented readout emphasizes confidence alignment, the balanced readout combines predictive quality with tail-aware performance, and the CVaR-oriented readout emphasizes the upper-loss portion of the evaluation distribution (Appendices A and B).

Five-seed consistency. Across the five matched seeds, raw EVE records the preferred NLL, CVaR-NLL, and accuracy in every comparison with DET and MC Dropout. ECE is preferred in three of five seeds, and its five-seed mean remains aligned with the comparison methods. The same runs provide the internal posterior measurements specific to EVE (Appendix B).

Response to controlled distribution shift. Increasing context perturbation produces coordinated changes in predictive measurements and internal probabilistic activity. Mean local KL activity, posterior scale, and posterior mean energy follow a coherent dose-response pattern. Target-frequency slices and ordered evaluation bins further characterize the behavior of alternative readouts across distinct statistical conditions (Appendices C and D).

6 Discussion

Internal measurement within predictive modeling. EVE combines predictive uncertainty with directly observable local posterior states. Deterministic models, MC Dropout, and ensembles characterize loss, calibration, and predictive disagreement at complementary levels. EVE adds unit-level mean, scale, KL activity, and participation measurements within a single forward computation.

Scaling the measurement principle. The controlled language-model setting demonstrates a unit-level measurement principle that can be carried into larger language and foundation-model architectures. Internal probabilistic activity can be tracked across training stages, data mixtures, post-training procedures, layers, modalities, and distribution shifts. These measurements support

analysis of how units allocate posterior dispersion, which computational pathways respond to particular data regimes, and how internal changes align with predictive behavior.

Readout spectrum. A single learned posterior state supports several readout criteria. Average likelihood, calibration, accuracy, and tail-aware performance emphasize distinct properties of the predictive distribution. Their joint characterization provides a multidimensional description of how internal probabilistic states are projected into decision-relevant outputs.

7 Conclusion

Internal probabilistic activity is directly measurable within a language-model computation. Across five shared seeds, EVE exposes local KL activity, posterior scale, posterior mean energy, and active-unit fraction while attaining strong likelihood, perplexity, accuracy, calibration, and tail-aware results alongside deterministic, MC Dropout, and ensemble references. Validation-selected readouts define explicit operating regimes, and controlled distribution shifts produce coordinated changes in predictive measurements and internal posterior observables. Variational distributional neurons therefore provide a concrete computational primitive for measuring uncertainty inside language models and a structured path toward unit-level probabilistic measurement in larger foundation-model architectures.

A Detailed Test-Run Numerical Results

Table 2: Detailed test-run results. Lower values indicate stronger performance for CE, MC-NLL, ECE, CVaR-NLL, and top-1 flip rate; higher values indicate stronger accuracy. Internal metrics are defined for EVE readouts.

Model	CE	PPL	Acc.	MC-NLL	ECE	CVaR-NLL	MI	Flip	KL_{int}	σ	μ^2	Active
DET	5.635	280.11	0.147	5.592	0.0460	12.889	0.000	0.000	0.000	0.000	0.000	0.0
MC Dropout	5.595	269.14	0.148	5.595	0.0479	12.915	0.006	0.050	0.000	0.000	0.000	0.0
EVE raw	5.388	218.80	0.166	5.392	0.0552	12.836	0.183	0.308	0.288	0.876	0.055	1.0
EVE-ECE-selected	5.392	219.71	0.161	5.392	0.0498	12.082	0.170	0.306	0.285	0.876	0.055	1.0
EVE-CVaR-selected	7.657	2115.71	0.161	7.657	0.1571	10.456	0.033	0.348	0.289	0.876	0.055	1.0
EVE-balanced	5.445	231.66	0.170	5.445	0.0617	11.940	0.091	0.318	0.291	0.876	0.055	1.0
DET Ensemble	5.455	233.81	0.163	5.455	0.0466	12.761	0.153	0.292	0.000	0.000	0.000	0.0

Table 3: Validation-selected EVE readout policies.

Policy	Temperature	Validation criterion	Readout
EVE-ECE-selected	1.1	ECE-first	Mean probabilities
EVE-CVaR-selected	3.0	CVaR-first	Median probabilities
EVE-balanced	1.2	Balanced score	Confidence-weighted samples

B Five-Seed Robustness

Table 4: Five-seed test summary. Values are mean ± standard deviation.

Model	MC-NLL	PPL	Acc.	ECE	CVaR-NLL	MI	Flip	KL_{int}
DET	5.620 ± 0.022	275.845 ± 6.015	0.148 ± 0.001	0.043 ± 0.007	13.200 ± 0.299	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
MC Dropout	5.621 ± 0.021	276.161 ± 5.894	0.153 ± 0.010	0.046 ± 0.011	13.213 ± 0.290	0.007 ± 0.003	0.061 ± 0.020	0.000 ± 0.000
EVE raw	5.359 ± 0.039	212.743 ± 8.288	0.166 ± 0.002	0.042 ± 0.009	12.713 ± 0.212	0.190 ± 0.011	0.324 ± 0.012	0.280 ± 0.017

Table 5: Seed-level predictive measurements underlying the aggregates in Table 4.

Model	Seed	MC-NLL	PPL	Acc.	ECE	CVaR-NLL
DET	0	5.592484	268.401501	0.142361	0.046035	12.889265
DET	1	5.614031	274.247460	0.151042	0.031154	13.159672
DET	2	5.635453	280.185742	0.161458	0.048854	13.667270
DET	3	5.608967	272.862309	0.147569	0.042176	13.012814
DET	4	5.647310	283.527822	0.164931	0.044604	13.270465
MC Dropout	0	5.595222	269.137263	0.147569	0.047949	12.915292
MC Dropout	1	5.615303	274.596449	0.144097	0.045674	13.183542
MC Dropout	2	5.630275	278.738830	0.161458	0.060052	13.660277
MC Dropout	3	5.611585	273.577553	0.147569	0.029973	13.011932
MC Dropout	4	5.651626	284.754131	0.166667	0.044137	13.291908
EVE raw	0	5.391827	219.604260	0.170139	0.055244	12.836452
EVE raw	1	5.400504	221.518058	0.163194	0.041262	12.374398
EVE raw	2	5.363157	213.397638	0.156250	0.033133	12.634963
EVE raw	3	5.335433	207.562506	0.166667	0.035532	12.855618
EVE raw	4	5.306452	201.633617	0.177083	0.043884	12.865204

Table 6: Seed-level internal probabilistic observables for raw EVE.

Seed	KL_{int}	σ	μ^2	Active	MI	Flip
0	0.287800	0.875682	0.055148	1.000	0.183363	0.308160
1	0.256759	0.863293	0.042177	1.000	0.207476	0.333767
2	0.269573	0.872650	0.049931	1.000	0.189330	0.337674
3	0.286864	0.871480	0.051319	1.000	0.194103	0.319444
4	0.298953	0.876101	0.057048	1.000	0.177926	0.319444

Table 7: Seedwise win rates for raw EVE across five shared seeds; 1.0 denotes a preferred value in all five seeds.

Comparison	NLL	ECE	CVaR-NLL	Accuracy
EVE raw vs. DET	1.0	0.6	1.0	1.0
EVE raw vs. MC Dropout	1.0	0.6	1.0	1.0

C Distribution-Shift Measurements

Table 8: Selected five-seed distribution-shift results. Values are averaged within each scenario and model family.

Scenario	Model	Seeds	NLL	Acc.	ECE	CVaR-NLL	KL_{int}	σ	μ^2
Base test	DET	5	5.478	0.159	0.0376	12.685	0.000	0.000	0.000
Base test	MC Dropout	5	5.478	0.162	0.0452	12.680	0.000	0.000	0.000
Base test	EVE raw	5	5.293	0.166	0.0599	12.339	0.274	0.873	0.051
Base test	EVE-balanced	5	5.403	0.168	0.0750	11.412	0.304	0.873	0.052
Context perturbation 0.15	DET	5	6.309	0.156	0.0597	16.028	0.000	0.000	0.000
Context perturbation 0.15	MC Dropout	5	6.305	0.154	0.0573	15.975	0.000	0.000	0.000
Context perturbation 0.15	EVE raw	5	6.237	0.143	0.0602	16.118	0.302	0.878	0.057
Context perturbation 0.15	EVE-balanced	5	6.176	0.153	0.0792	14.409	0.309	0.879	0.057
Context perturbation 0.30	DET	5	8.167	0.103	0.0925	18.540	0.000	0.000	0.000
Context perturbation 0.30	MC Dropout	5	8.137	0.104	0.0916	18.429	0.000	0.000	0.000
Context perturbation 0.30	EVE raw	5	8.057	0.103	0.0941	18.483	0.362	0.899	0.075
Context perturbation 0.30	EVE-balanced	5	7.734	0.106	0.0728	16.426	0.366	0.899	0.075
Rare targets	DET	5	10.375	0.000	0.1226	15.619	0.000	0.000	0.000
Rare targets	EVE-balanced	5	9.551	0.002	0.0775	14.334	0.313	0.870	0.055
Common targets	DET	5	2.012	0.648	0.4448	4.346	0.000	0.000	0.000
Common targets	EVE raw	5	2.289	0.494	0.3100	5.030	0.268	0.871	0.042

Table 9: Dose-response measurements for raw EVE under controlled context perturbation. Values are mean ± standard deviation across five seeds.

Perturbation rate	NLL	Acc.	ECE	CVaR-NLL	KL_{int}	σ	μ^2
0.05	5.544 ± 0.038	0.167 ± 0.013	0.059 ± 0.020	13.064 ± 0.605	0.291 ± 0.007	0.874 ± 0.005	0.052 ± 0.005
0.15	6.237 ± 0.073	0.143 ± 0.006	0.060 ± 0.010	16.118 ± 0.878	0.302 ± 0.019	0.878 ± 0.006	0.057 ± 0.005
0.30	8.057 ± 0.183	0.103 ± 0.015	0.094 ± 0.022	18.483 ± 0.969	0.362 ± 0.034	0.899 ± 0.008	0.075 ± 0.007

D Result Figures

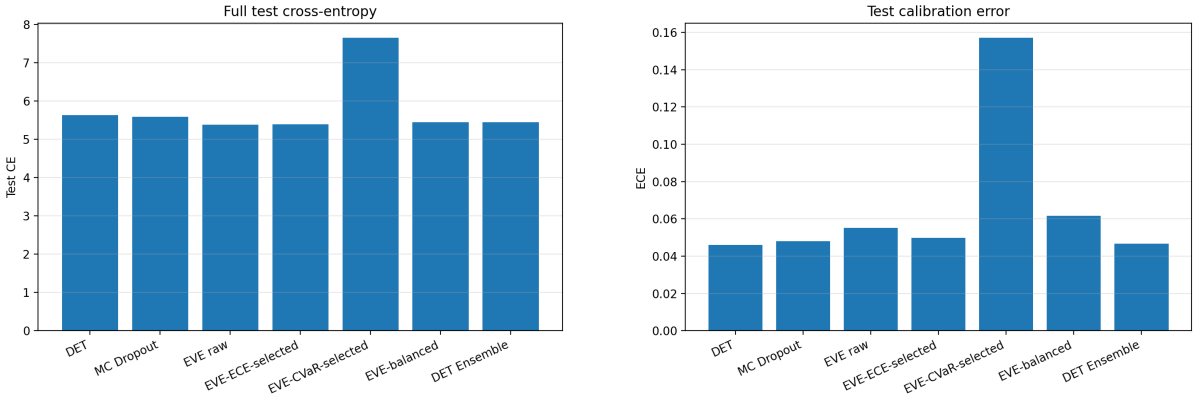

Figure 1: Output-level comparison. Left: test cross-entropy. Right: test expected calibration error.

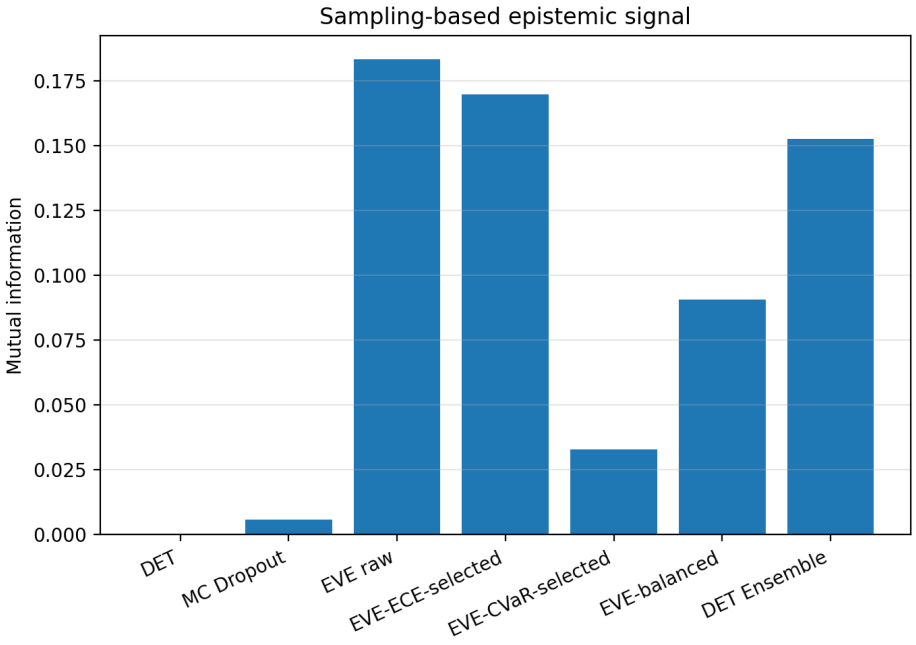

Figure 2: Predictive mutual information across stochastic samples.

Author Contributions

Conceptualization, methodology, software, validation, formal analysis, investigation, data curation, visualization, writing—original draft, and writing—review and editing: Y.R.

Funding

The research was conducted independently.

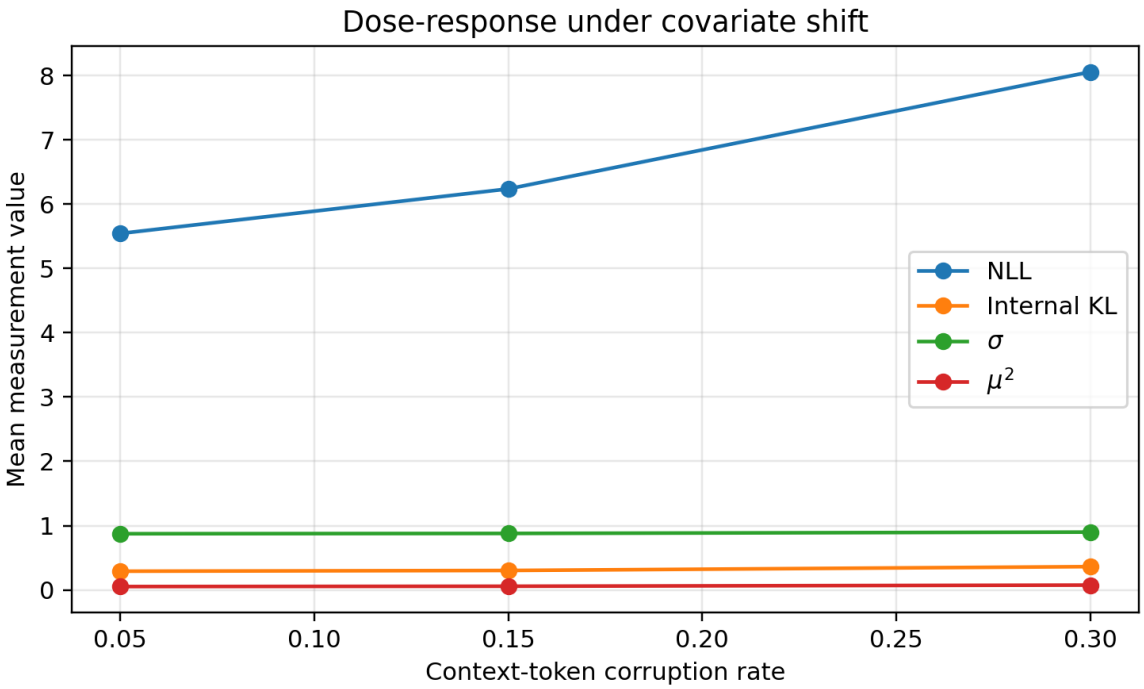

Figure 3: Two-view dose-response for raw EVE under controlled context perturbation. Left: NLL and CVaR-NLL, shown as mean ± standard deviation across five seeds. Right: internal KL activity, posterior scale, and posterior mean energy normalized to their values at perturbation rate 0.05. The coordinated response makes the relationship between output-level change and internal probabilistic activity directly visible.

Institutional Review Board Statement

The research consists exclusively of computational experiments on a public text dataset.

Informed Consent Statement

The research consists exclusively of computational experiments on a public text dataset.

Data Availability Statement

The appendices contain the numerical summaries supporting the reported findings, including the detailed test-run results, individual measurements for all five seeds, aggregate statistics, validation-selected readout definitions and settings, distribution-shift measurements, and result figures.

Acknowledgments

The author thanks the developers and maintainers of the open-source software and datasets used for the experiments.

Declaration of Generative AI and AI-Assisted Technologies

During the preparation of this manuscript, generative artificial intelligence tools were used for language editing, structural revision, and LaTeX formatting. All generated material was reviewed and edited by the authors, who take full responsibility for the scientific content of the manuscript.

Conflicts of Interest

The author declares no conflicts of interest.

References

McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5, 115–133.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386–408.

Werbos, P. J. (1974). Beyond regression: New tools for prediction and analysis in the behavioral sciences. *Ph.D. thesis, Committee on Applied Mathematics, Harvard University*.

Linnainmaa, S. (1976). Taylor expansion of the accumulated rounding error. *BIT Numerical Mathematics*, 16(2), 146–160.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323, 533–536.

Broomhead, D. S. and Lowe, D. (1988). Radial basis functions, multi-variable functional interpolation and adaptive networks. *Royal Signals and Radar Establishment Memorandum 4148*.

Moody, J. and Darken, C. J. (1989). Fast learning in networks of locally-tuned processing units. *Neural Computation*, 1(2), 281–294.

Park, J. and Sandberg, I. W. (1991). Universal approximation using radial-basis-function networks. *Neural Computation*, 3(2), 246–257.

Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8), 2554–2558.

Ackley, D. H., Hinton, G. E., and Sejnowski, T. J. (1985). A learning algorithm for Boltzmann machines. *Cognitive Science*, 9(1), 147–169.

Dayan, P., Hinton, G. E., Neal, R. M., and Zemel, R. S. (1995). The Helmholtz machine. *Neural Computation*, 7(5), 889–904.

Hinton, G. E., Dayan, P., Frey, B. J., and Neal, R. M. (1995). The wake–sleep algorithm for unsupervised neural networks. *Science*, 268(5214), 1158–1161.

Tang, Y. and Salakhutdinov, R. (2013). Learning stochastic feedforward neural networks. *Advances in Neural Information Processing Systems*, 26.

Bengio, Y., Léonard, N., and Courville, A. (2013). Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*.

MacKay, D. J. C. (1992). A practical Bayesian framework for backpropagation networks. *Neural Computation*, 4(3), 448–472.

Neal, R. M. (1992). Bayesian learning via stochastic dynamics. *Advances in Neural Information Processing Systems*, 5.

Graves, A. (2011). Practical variational inference for neural networks. *Advances in Neural Information Processing Systems*, 24.

Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. (2015). Weight uncertainty in neural networks. *Proceedings of the 32nd International Conference on Machine Learning*, 1613–1622.

Kingma, D. P., Salimans, T., and Welling, M. (2015). Variational dropout and the local reparameterization trick. *Advances in Neural Information Processing Systems*, 28.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15, 1929–1958.

Bishop, C. M. (1994). Mixture density networks. *Technical Report NCRG/94/004, Aston University.*

Kendall, A. and Gal, Y. (2017). What uncertainties do we need in Bayesian deep learning for computer vision? *Advances in Neural Information Processing Systems*, 30.

Damianou, A. and Lawrence, N. D. (2013). Deep Gaussian processes. *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics*, 207–215.

Urban, S., Basalla, M., and van der Smagt, P. (2017). Gaussian Process Neurons learn stochastic activation functions. *arXiv preprint arXiv:1711.11059.*

RLAIF Team. (2025). WritingPrompts Filtered Dataset (LitBench Decontaminated). *Hugging Face dataset.*

Bommasani, R., Hudson, D. A., Adeli, E., et al. (2021). On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258.*

Vaswani, A., Shazeer, N., Parmar, N., et al. (2017). Attention is all you need. *Advances in Neural Information Processing Systems.*

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of NAACL-HLT.*

Brown, T. B., Mann, B., Ryder, N., et al. (2020). Language models are few-shot learners. *Advances in Neural Information Processing Systems.*

Kaplan, J., McCandlish, S., Henighan, T., et al. (2020). Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361.*

Hoffmann, J., Borgeaud, S., Mensch, A., et al. (2022). Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556.*

Gal, Y. and Ghahramani, Z. (2016). Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. *International Conference on Machine Learning.*

Lakshminarayanan, B., Pritzel, A., and Blundell, C. (2017). Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in Neural Information Processing Systems.*

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. (2017). On calibration of modern neural networks. *International Conference on Machine Learning*.

Kingma, D. P. and Welling, M. (2014). Auto-encoding variational Bayes. *International Conference on Learning Representations*.

Rezende, D. J., Mohamed, S., and Wierstra, D. (2014). Stochastic backpropagation and approximate inference in deep generative models. *International Conference on Machine Learning*.