Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Distributional Neurons: Making Uncertainty a Unit of Computation

YVES RUFFENACH

yves@ruffenach.net

Conservatoire National des Arts et Métiers https://orcid.org/0009-0009-4737-0555

Research Article

Keywords: Distributional neurons, variational inference, uncertainty quantification, stochastic neural networks, probabilistic deep learning, heteroscedastic regression, deep neural network composition, Transformers

Posted Date: July 21st, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10401116/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: The authors declare no competing interests.

Distributional Neurons: Making Uncertainty a Unit of Computation

Yves Ruffenach
Conservatoire National des Arts et Métiers
Strasbourg, France
yves@ruffenach.net

Abstract

Neural networks usually propagate point-valued hidden activations. Uncertainty is commonly introduced through weights, outputs, ensembles, stochastic perturbations, or global latent variables. We study a finer granularity: the neuron as a distributional computational unit. We introduce EVE, a variational neuron with an input-dependent posterior, local prior, reparameterized sampling, neuron-level KL regularization, and internal diagnostics. EVE computes through sampled local latent states rather than only parameterizing uncertainty. A single-unit microscope isolates neuron-level latent computation from depth, width, and composition. Against deterministic and capacity-matched deterministic controls, EVE preserves comparable point accuracy and improves NLL, CRPS, and alignment with true heteroscedastic structure. A direct heteroscedastic output control provides a strong output-level uncertainty reference; EVE adds neuron-level posterior states and local KL diagnostics. We then test true dense 128×128 MLP composition on Energy Efficiency. EVE improves NLL, CRPS, and pinball loss on all five test seeds; MSE and MAE remain close but are not uniformly improved on test. Finally, a PG-19 Transformer feed-forward bridge shows improved validation CE, perplexity, accuracy, and Monte Carlo NLL over a matched deterministic Transformer, with non-degenerate Monte Carlo uncertainty signals. The results support neuron-level uncertainty as a feasible computational primitive that can be isolated at the unit level, composed in deep dense networks, and inserted into Transformer feed-forward computation.

1 Introduction

Neural networks usually propagate point-valued hidden activations. Uncertainty is commonly placed in weights, outputs, ensembles, stochastic perturbations, or global latent variables. The hidden unit itself often remains a deterministic scalar object.

We study uncertainty at the unit level. A neuron is treated as a local variational object with a posterior, prior, sampled latent state, variational regularizer, and diagnostics. The change is local. The external topology is preserved. A deterministic hidden unit is replaced by a distributional hidden unit. The forward pass then depends on sampled neuron-level latent states.

We introduce EVE, a distributional neuron. Given a preactivation, EVE defines an input-dependent posterior, samples a local latent state by reparameterization, emits an activation through a local decoder, and regularizes the posterior against a standard prior. Uncertainty becomes part of the forward computation.

We evaluate EVE in three stages. First, a single-unit microscope isolates the computational unit from depth, width, and composition. Second, a true dense 128×128 MLP tests deep composition. Third, a Transformer feed-forward replacement tests insertion into a modern sequence architecture. The target property is consistent across stages: neuron-level latent computation should remain trainable, measurable, and diagnostically meaningful.

Our contribution is threefold. First, we formulate a distributional neuron as a local variational computational unit. Second, we isolate this unit-level change from deterministic local capacity and direct output-level uncertainty through deterministic, capacity-matched, and heteroscedastic controls. Third, we test the same unit under composition: first in a true 128×128 dense MLP on Energy Efficiency, then as a Transformer feed-forward replacement on PG-19 language modeling. This evaluates neuron-level uncertainty

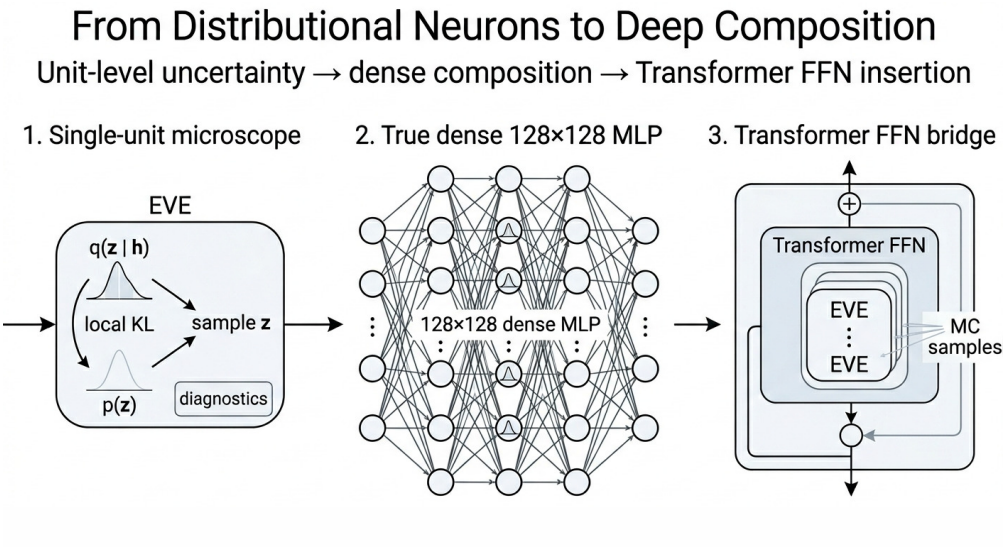


Figure 1: Three-stage evaluation of EVE. We first isolate the local variational neuron, then compose it in a true dense 128×128 MLP, and finally insert it into Transformer feed-forward computation.

as a computational primitive: whether it can be trained, measured, controlled, and composed while preserving competitive predictive performance.

2 Related Work

Artificial neurons as computational units. The artificial neuron is usually a simple computational abstraction. The McCulloch–Pitts neuron formalized neural activity as an all-or-none logical unit (McCulloch and Pitts, 1943). The perceptron introduced a trainable threshold unit for information storage and organization (Rosenblatt, 1958). Back-propagation made multilayer networks of neuron-like units trainable by differentiating through their computations (Rumelhart et al., 1986). Universal approximation results established that networks of simple scalar nonlinear units can approximate broad classes of functions (Cybenko, 1989). This defines the standard view: a neuron computes a scalar activation; expressivity emerges from composition.

Richer deterministic units. Later work enriched the deterministic neuron while preserving a deterministic hidden state. Activation-function design modifies the scalar nonlinearity. Maxout units use the maximum over learned affine responses (Goodfellow et al., 2013). Parametric rectifiers learn the negative-region slope (He et al., 2015). GELU and Swish introduce smooth value-dependent activation rules that improve optimization and performance (Hendrycks and Gimpel, 2016; Ramachandran et al., 2017). EVE changes the unit state itself: a point activation becomes a sampled local latent variable.

Stochastic neurons and stochastic computation. Stochastic units have a long history. Boltzmann machines use stochastic binary units and statistical-mechanics-inspired learning rules (Ackley et al., 1985). Stochastic neurons and hard nonlinearities support conditional computation but require gradient estimators (Bengio et al., 2013). Concrete relaxations give differentiable approximations to discrete stochastic nodes (Maddison et al., 2017). EVE makes each hidden unit an amortized variational object with posterior, prior, reparameterized sample, KL regularizer, and diagnostics.

Uncertainty in neural networks. Most uncertainty-aware methods place uncertainty outside the hidden unit. Bayesian neural networks place distributions over parameters (Blundell et al., 2015). Ensembles approximate predictive uncertainty with multiple models (Lakshminarayanan et al., 2017). Dropout injects

stochasticity into activations or weights (Gal and Ghahramani, 2016; Srivastava et al., 2014). Heteroscedastic heads predict input-dependent output variance (Kendall and Gal, 2017). Variational dropout and the local reparameterization trick translate parameter uncertainty into local noise for efficient stochastic-gradient variational inference (Kingma et al., 2015). These are strong uncertainty methods. EVE assigns posterior, prior, sampling, KL, and diagnostics to the individual hidden unit.

Latent-variable models. Variational autoencoders introduce learned distributions into neural computation (Kingma and Welling, 2014). Their latent variables are usually attached to inputs, representations, or global stochastic states. EVE changes the granularity: each neuron becomes a local latent-variable computation.

Transformer feed-forward computation. Transformers combine self-attention with deterministic feed-forward blocks (Vaswani et al., 2017). In language modeling, uncertainty is usually read from the final predictive distribution or from external stochastic procedures. Our Transformer experiment preserves self-attention, embeddings, normalization, and the language-model head. It replaces deterministic feed-forward computation by local variational neurons.

Positioning of EVE. EVE combines unit-level locality, variational structure, stochastic computation, and diagnostics. The latent distribution is attached to the neuron. Each unit has a posterior, prior, KL term, sampled latent state, posterior energy, posterior variance, local KL, and band occupancy. This positions EVE as a distributional computational unit rather than only an activation function, stochastic regularizer, output-variance head, or global latent-variable model.

3 Distributional Neurons

3.1 From Point Activations to Local Latent Computation

A deterministic neuron maps a preactivation h_i to a scalar activation

$$a_i = \phi(h_i).$$

The unit propagates one point value.

A deterministic mean-variance unit can output

$$(\mu_i, \sigma_i) = f_\theta(h_i).$$

This parameterizes a distribution, while the propagated computation can remain deterministic.

EVE replaces this unit by a local variational computation. Given h_i, the unit defines

$$q_i(z_i \mid h_i) = \mathcal{N}\left(\mu_i(h_i), \operatorname{diag}(\sigma_i^2(h_i))\right),$$

with local prior

$$p(z_i) = \mathcal{N}(0, I).$$

The local latent state is sampled by

$$z_i = \mu_i(h_i) + \sigma_i(h_i) \odot \epsilon_i, \qquad \epsilon_i \sim \mathcal{N}(0, I).$$

The emitted activation is

$$a_i = g_\theta(z_i, h_i).$$

The unit computes through a sampled local latent state.

3.2 Why This Is Not a Deterministic Mean-Variance Unit

The distinction is computational. A deterministic mean-variance unit describes a distribution through deterministic parameters. EVE samples a local latent state and uses that state in the forward pass.

The hidden unit therefore has internal stochastic state. The model can measure and regulate it locally. The unit exposes posterior mean energy, posterior variance, and local KL. These are unit-level variational diagnostics.

3.3 Training Objective

For a network with EVE units, the task loss is augmented by neuron-level variational regularization:

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \beta \sum_i D_{\text{KL}} \left(q_i(z_i \mid h_i) \| p(z_i) \right).$$

The KL term is local to the unit. It regularizes each posterior toward the standard prior and makes the latent state measurable during training.

Additional local regularizers can control the operating regime. For posterior mean energy

$$e_i = \|\mu_i(h_i)\|_2^2,$$

a soft band penalty is

$$\mathcal{L}_{\text{band}} = \sum_i \left[\max(0, \ell - e_i)^2 + \max(0, e_i - u)^2 \right],$$

where $[\ell, u]$ is the target energy band. The full objective is

$$\mathcal{L} = \mathcal{L}_{\text{task}} + \beta \mathcal{L}_{\text{KL}} + \lambda_{\text{band}} \mathcal{L}_{\text{band}}.$$

3.4 Unit-Level Diagnostics

EVE makes the hidden unit observable. For each unit, we track

$$D_{\text{KL}}(q_i(z_i \mid h_i) \| p(z_i)),$$

$$\|\mu_i(h_i)\|_2^2,$$

$$\sigma_i^2(h_i),$$

and the fraction of units inside a target energy interval.

These diagnostics describe the internal operating state of the computational unit. They identify collapse, drift, inactive units, over-dispersion, and layer-wise imbalance. Neuron-level uncertainty therefore exposes where uncertainty appears inside the computation.

4 Experiments

Reproducibility note. The main text reports datasets, models, metrics, seeds, evaluation protocols, selection criteria, architectures, and subset sizes. The experimental implementations specify executable configurations, including data preparation, splits, optimizers, learning rates, batch sizes, training loops, seeds, Monte Carlo settings, evaluation code, and figure generation.

4.1 Stage I: Single-Unit Microscope

We first evaluate EVE in a single-unit microscope. The experiment isolates the effect of replacing a deterministic unit by a local variational unit. All results are averaged over five seeds.

We use controlled heteroscedastic regression with input-dependent conditional noise. We compare four unit types: DET, a deterministic scalar unit; DET-cell matched, a deterministic unit with comparable local

encoder-decoder capacity but no posterior, prior, sampling, or KL; STOCH-hetero, a direct heteroscedastic output control without neuron-level latent computation; and EVE.

We report MSE, NLL, CRPS, 90% coverage, 90% interval width, and correlation between predicted standard deviation and true heteroscedastic noise. MSE measures point prediction. NLL and CRPS measure predictive distribution quality. The controls isolate local deterministic capacity, direct output-level uncertainty, and neuron-level latent computation.

Table 1: Single-unit microscope on controlled heteroscedastic regression. Values are mean ± standard deviation over five seeds. EVE keeps MSE comparable to the capacity-matched deterministic control while improving probabilistic metrics and uncertainty alignment. The direct heteroscedastic output control provides the strongest direct output-uncertainty reference.

Model	Params	MSE ↓	NLL ↓	CRPS ↓	Cov90 ↑	Width90 ↓	Corr. ↑
DET	13	0.1042 ± 0.0110	0.2877 ± 0.0532	0.1764 ± 0.0088	0.8911 ± 0.0110	1.0573 ± 0.0722	0.0021 ± 0.0196
DET-cell matched	2596	0.0955 ± 0.0065	0.2464 ± 0.0320	0.1700 ± 0.0052	0.8965 ± 0.0161	1.0272 ± 0.0187	0.0100 ± 0.0161
STOCH-hetero	3748	$\mathbf{0.0954 \pm 0.0065}$	$\mathbf{0.0668 \pm 0.0149}$	$\mathbf{0.1614 \pm 0.0045}$	0.8864 ± 0.0085	$\mathbf{0.9316 \pm 0.0106}$	$\mathbf{0.9867 \pm 0.0036}$
EVE	2662	0.0958 ± 0.0064	0.1144 ± 0.0111	0.1635 ± 0.0043	$\mathbf{0.9123 \pm 0.0109}$	0.9948 ± 0.0107	0.9390 ± 0.0081

Table 1 separates three effects. DET tests point-valued computation. DET-cell matched tests deterministic local capacity. STOCH-hetero tests direct output-level uncertainty. EVE tests uncertainty inside the unit.

The direct heteroscedastic output control gives the strongest direct output-uncertainty scores: best NLL, CRPS, interval width, and correlation with true noise. This matches the synthetic design, where the target noise is explicitly heteroscedastic and optimized at the output.

EVE adds neuron-level latent computation while preserving competitive point accuracy. Compared with DET, EVE reduces MSE from 0.1042 to 0.0958, NLL from 0.2877 to 0.1144, and CRPS from 0.1764 to 0.1635. It increases uncertainty alignment from 0.0021 to 0.9390.

The capacity-matched comparison isolates deterministic capacity. DET-cell matched has comparable local encoder-decoder capacity but no posterior, prior, sampling, or KL. EVE keeps point accuracy comparable while reducing NLL from 0.2464 to 0.1144, reducing CRPS from 0.1700 to 0.1635, and increasing uncertainty alignment from 0.0100 to 0.9390. Across five seeds, EVE improves NLL and CRPS over DET-cell matched in every run. The MSE difference is negligible.

Figure 2 carries the main uncertainty-alignment comparison. Deterministic controls show near-zero alignment with the true noise; EVE tracks the heteroscedastic structure and retains neuron-level KL diagnostics through its local variational objective.

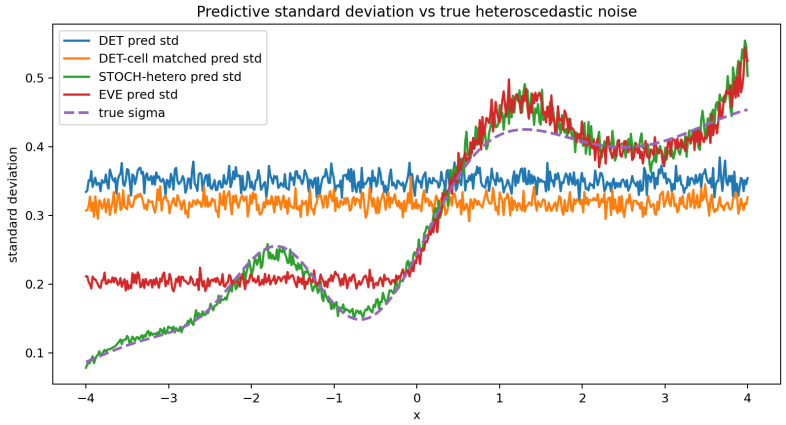


Figure 2: Predicted standard deviation against true heteroscedastic noise. Deterministic controls show near-zero alignment. EVE tracks heteroscedastic structure; the direct heteroscedastic output control remains closest to true noise.

4.2 Stage II: True Dense MLP Composition

The microscope isolates the unit. We next test deep composition on UCI Energy Efficiency (Tsanas and Xifara, 2012). The notation 128×128 denotes 128 sequential fully connected hidden layers of width 128. The deterministic baseline uses standard point-valued hidden units. EVE preserves the external topology and replaces hidden units by local variational units. Both models are evaluated over five seeds.

This is a composition test. It evaluates whether neuron-level latent computation remains trainable, measurable, and active under depth. It also tests whether unit-level probabilistic gains persist when many distributional units interact across layers. Both models use geometry 128×128, 48 epochs, and the same dataset protocol.

Table 2: True dense 128×128 MLP composition on Energy Efficiency. Values are mean $\pm$ standard deviation over five seeds. EVE improves uncertainty-sensitive metrics on all five seeds. MSE and MAE remain close but do not uniformly improve on test.

Test metric ↓	DET	EVE	Gain	EVE wins
MSE	0.9802 ± 0.0820	1.0170 ± 0.1162	-3.75%	2/5
NLL	1.6699 ± 0.0993	1.3557 ± 0.0374	$+18.82\%$	5/5
CRPS	0.6171 ± 0.0332	0.5889 ± 0.0397	$+4.57\%$	5/5
MAE	0.8770 ± 0.0400	0.8848 ± 0.0595	-0.90%	2/5
Pinball	0.2693 ± 0.0165	0.2568 ± 0.0152	$+4.65\%$	5/5

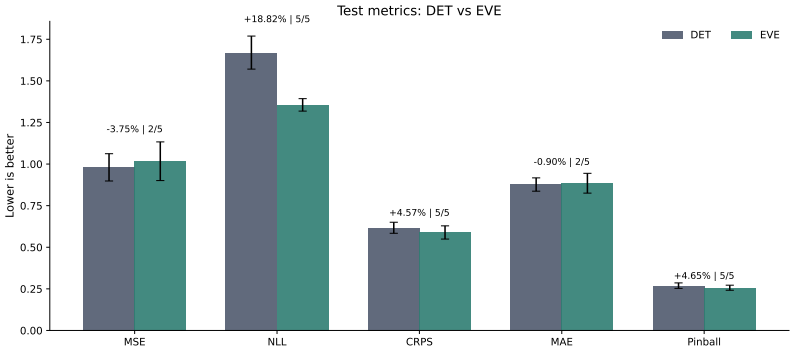


Figure 3: True dense 128×128 MLP test metrics on Energy Efficiency. EVE improves uncertainty-sensitive metrics under deep dense composition, with 5/5 seed wins on NLL, CRPS, and pinball loss. MSE and MAE remain close but do not uniformly improve on test.

Table 3: Validation metrics for true dense 128×128 MLP composition on Energy Efficiency. Values are mean $\pm$ standard deviation over five seeds. EVE improves all reported validation metrics on average, with 5/5 wins on NLL, CRPS, and pinball loss.

Validation metric ↓	DET	EVE	Gain	EVE wins
MSE	0.9909 ± 0.1399	0.9703 ± 0.1474	$+2.08\%$	4/5
NLL	1.6832 ± 0.1713	1.2916 ± 0.0838	$+23.26\%$	5/5
CRPS	0.6215 ± 0.0471	0.5734 ± 0.0448	$+7.74\%$	5/5
MAE	0.8845 ± 0.0476	0.8615 ± 0.0566	$+2.60\%$	4/5
Pinball	0.2719 ± 0.0243	0.2493 ± 0.0170	$+8.32\%$	5/5

Tables 2 and 3 show the composition behavior. On validation, EVE improves all reported metrics on average. It wins NLL, CRPS, and pinball loss in all five seeds, and MSE and MAE in four of five seeds.

On test, gains concentrate on uncertainty-sensitive metrics. EVE improves NLL by 18.82%, CRPS by 4.57%, and pinball loss by 4.65%, with 5/5 seed wins for all three. MSE changes from 0.9802 to 1.0170. MAE

changes from 0.8770 to 0.8848. The dense MLP result therefore supports probabilistic improvement under deep composition, with close point-error metrics. Figure 3 visualizes the same pattern.

Table 4: EVE diagnostics in the true dense 128×128 MLP. The latent state remains active under depth, with measurable KL, nonzero posterior energy, and controlled high-activity mass.

Diagnostic	Value
KL mean	48.18
Posterior energy μ^2 mean	0.2106
Inside-band mass	0.5156
Fraction too high	0.0047
Fraction too low	0.4797
Valid rate	1.0000

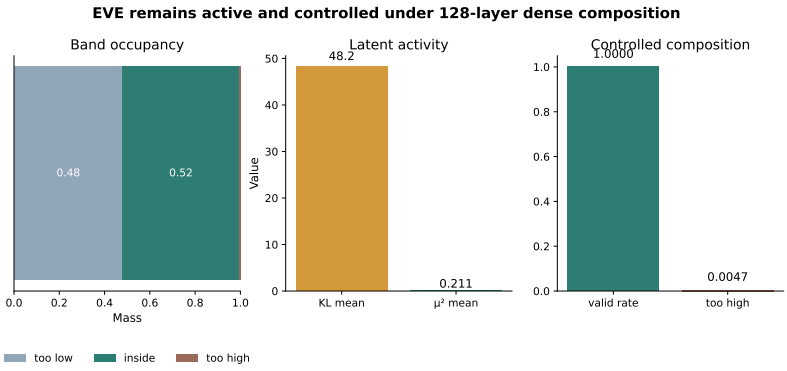


Figure 4: EVE diagnostics under true dense 128×128 composition. The latent state remains active under depth, with measurable KL, nonzero posterior energy, and controlled high-activity mass. The diagnostic profile supports the claim that the unit remains active and measurable when composed through 128 fully connected layers.

Table 4 reports internal EVE diagnostics. The latent state remains active: KL is nonzero, posterior energy is measurable, and the valid rate is 1.0. High-activity mass is small, with 0.0047 of units above the upper band. Most out-of-band mass is low-side, indicating conservative latent use rather than high-activity collapse. Figure 4 summarizes the operating regime.

The composition gain has nontrivial computational cost. In this implementation, EVE is approximately 7.1× slower than the deterministic baseline, reflecting local posterior computation, sampling, and probabilistic evaluation.

4.3 Stage III: Transformer Feed-Forward Replacement on PG-19

The dense MLP experiment tests deep fully connected composition. We next use PG-19 as a Transformer feed-forward bridge. The stage evaluates whether the same distributional neuron can be inserted into a modern sequence architecture while preserving self-attention, embeddings, normalization, and the language-model head.

We replace the feed-forward computation of a compact Transformer by local variational neurons and evaluate next-token prediction on PG-19 (Rae et al., 2020). The stochastic computation is localized to the feed-forward path. This is a capped feed-forward composition run. It tests trainability, predictive utility, and measurable Monte Carlo uncertainty under the reported budget.

We use a capped streamed PG-19 subset with 16k prompt-story segments, corresponding to 1,241,600 training tokens and 244,440 validation tokens in the reported run. Both DET and EVE are trained for four epochs with the same data protocol. Checkpoint selection uses validation cross-entropy.

Table 5 reports the PG-19 result. EVE improves validation CE from 5.2456 to 5.0872, perplexity from 189.74 to 161.94, and accuracy from 0.1938 to 0.2064. It improves Monte Carlo predictive NLL from 5.2843

Table 5: PG-19 Transformer feed-forward replacement. EVE improves the matched deterministic Transformer on task-level metrics and Monte Carlo predictive NLL. Lower is better for CE, PPL, MC NLL, MC PPL, ECE, and CVaR-NLL. Higher is better for accuracy, mutual information, and top-1 Monte Carlo flip rate.

Metric	DET	EVE	Better
Validation CE ↓	5.2456	**5.0872**	EVE
Validation PPL ↓	189.74	**161.94**	EVE
Validation accuracy ↑	0.1938	**0.2064**	EVE
MC NLL ↓	5.2843	**5.1166**	EVE
MC PPL ↓	197.22	**166.77**	EVE
ECE ↓	**0.0256**	0.0442	DET
CVaR-NLL ↓	**12.1046**	12.1769	DET
Mutual information ↑	≈ 0	**0.1214**	EVE
Top-1 MC flip rate ↑	0.0000	**0.2474**	EVE

to 5.1166. DET is better on ECE and CVaR-NLL in this run. The main signal is task-level predictive improvement plus non-degenerate Monte Carlo uncertainty.

Figure 5 shows the optimization pattern. The deterministic Transformer reaches its best validation point at epoch 2 and then degrades. EVE continues improving through epoch 4. Under this controlled budget, neuron-level variational computation provides a more stable validation trajectory than the matched deterministic baseline.

Table 6: EVE internal diagnostics on the PG-19 Transformer run. The global latent regime is active and measurable, but the selected checkpoint remains dominated by the first variational layer. This makes the run useful as a Transformer feasibility result, while leaving balanced deep latent usage as an optimization target.

Diagnostic	Value
Global posterior energy μ^2 mean	0.1262
Fraction too low	0.3018
Inside-band mass	0.5771
Fraction too high	0.1211
Layer 2 weight	0.0702
Layer 3 weight	0.0669
Layer 2 μ^2	0.00020
Layer 3 μ^2	0.00811

Table 6 reports internal diagnostics. The global latent regime is active and measurable. Inside-band mass is 0.5771, and the fraction of too-low units is 0.3018. The selected checkpoint remains first-layer dominated. Layer 2 and layer 3 have weak energy and limited final weight. The run supports Transformer feasibility and predictive benefit; balanced deep latent usage remains an optimization target.

5 Discussion

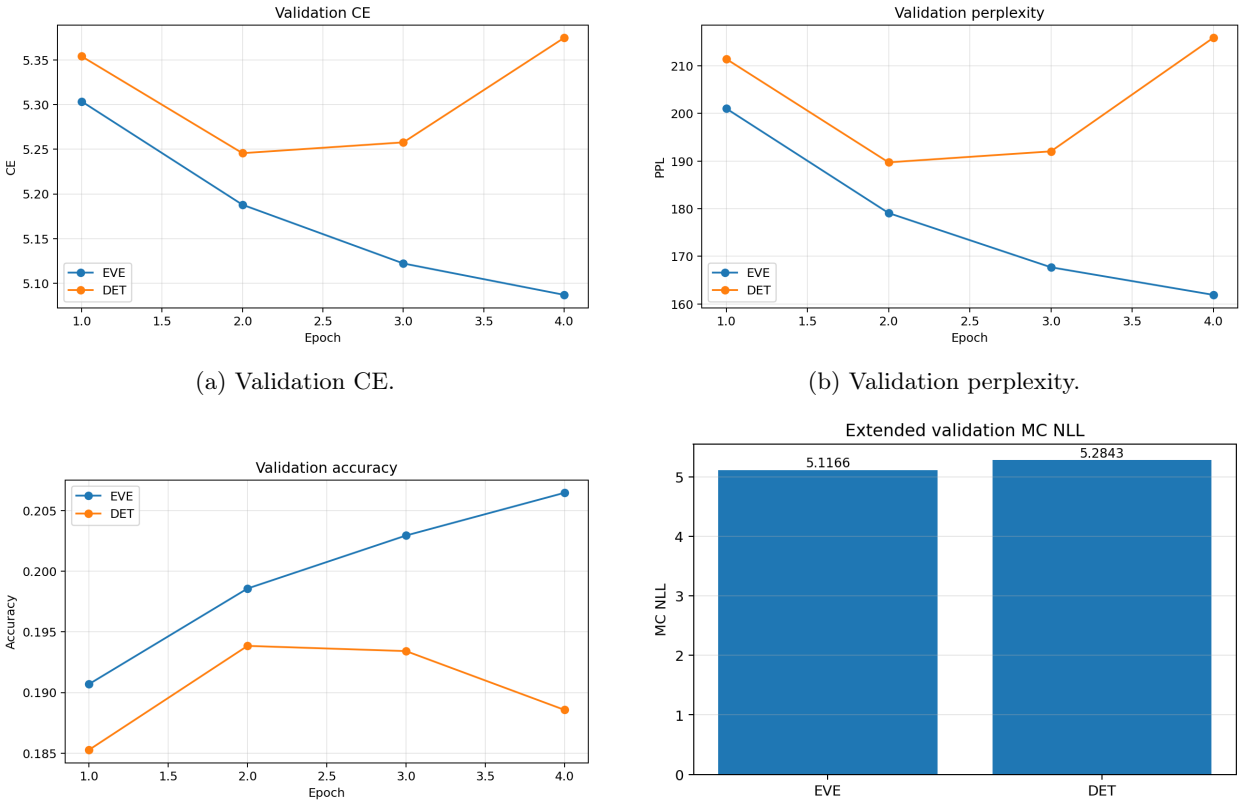
The results separate output-level uncertainty from neuron-level uncertainty. The heteroscedastic output head is strongest when the task directly exposes output noise. EVE instead represents, samples, regularizes, and measures uncertainty inside the computational unit.

A deterministic unit exposes an activation. An EVE unit exposes a posterior, prior mismatch, sampled latent state, local KL, posterior energy, posterior variance, and occupancy statistics.

The DET-cell matched comparison is central. Matched deterministic capacity does not reproduce EVE's uncertainty alignment. The relevant structure is posterior, prior, sampling, KL, and diagnostics.

The dense 128×128 MLP extends the microscope. EVE remains active under deep composition. It improves NLL, CRPS, and pinball loss on all five test seeds. MSE and MAE remain close but are not uniformly improved.

The PG-19 Transformer experiment tests FFN insertion. EVE improves CE, perplexity, accuracy, and MC NLL under the reported budget. It also produces non-degenerate Monte Carlo uncertainty signals. The



(a) Validation CE.

(b) Validation perplexity.

(c) Validation accuracy.

(d) Extended MC NLL.

Figure 5: PG-19 Transformer feed-forward replacement. The deterministic Transformer reaches its best validation CE early and then degrades, while EVE continues improving across the four-epoch run. EVE also improves MC NLL, showing that the stochastic feed-forward path contributes to predictive distribution quality.

selected checkpoint remains first-layer dominated.

Overall, output-level uncertainty says that a prediction is uncertain. Neuron-level uncertainty indicates where uncertainty appears inside the computation.

6 Limitations

This work studies a change in computational granularity under controlled settings.

The single-unit microscope is synthetic by design. It isolates the mechanism but does not replace broader architectural and dataset-level evaluation. The heteroscedastic output head is the strongest direct output-noise reference in this setting.

The true dense 128×128 MLP tests deep composition. It supports probabilistic gains under depth, but MSE and MAE are not uniformly improved on test. EVE also adds runtime overhead through posterior computation, sampling, and probabilistic evaluation.

The PG-19 Transformer experiment is a capped feed-forward bridge. It supports insertion into Transformer FFN blocks, but calibration and tail-risk are mixed. The selected checkpoint remains first-layer dominated. Balanced deep latent usage remains open.

The baseline set is focused. Broader comparisons should include deep ensembles, Monte Carlo dropout, Bayesian weight-space baselines, and tuned Gaussian output heads.

Larger Transformer runs, stronger baselines, broader architectures, and improved controller policies are natural next steps.

7 Conclusion

We introduced EVE, a distributional neuron that makes uncertainty part of the computational unit. It defines a local posterior, samples a latent state, regularizes it against a prior, and exposes unit-level diagnostics. The single-unit microscope isolates this mechanism. EVE matches the capacity-matched deterministic control on point prediction while improving NLL, CRPS, and uncertainty alignment. The heteroscedastic output control remains the strongest direct output-uncertainty reference. The true dense 128×128 MLP tests composition. EVE improves NLL, CRPS, and pinball loss on all five test seeds. Its latent state remains active under 128-layer composition. MSE and MAE remain close but are not uniformly improved. The PG-19 Transformer experiment tests FFN insertion. EVE improves validation CE, perplexity, accuracy, and Monte Carlo NLL over a matched deterministic Transformer. It also produces non-degenerate Monte Carlo uncertainty signals. Overall, EVE supports a staged view of distributional computation: isolate the neuron, compose it in deep dense networks, and insert it into Transformer feed-forward computation. Across these stages, EVE remains trainable, measurable, and useful for uncertainty-sensitive prediction.

References

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

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

Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D. Weight uncertainty in neural networks. In *International Conference on Machine Learning*, 2015.

Cybenko, G. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2:303–314, 1989.

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

Goodfellow, I. J., Warde-Farley, D., Mirza, M., Courville, A., and Bengio, Y. Maxout networks. In *International Conference on Machine Learning*, 2013.

He, K., Zhang, X., Ren, S., and Sun, J. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In *IEEE International Conference on Computer Vision*, 2015.

Hendrycks, D. and Gimpel, K. Gaussian error linear units. arXiv preprint arXiv:1606.08415, 2016.

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

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

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

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

Maddison, C. J., Mnih, A., and Teh, Y. W. The Concrete distribution: A continuous relaxation of discrete random variables. In *International Conference on Learning Representations*, 2017.

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

Rae, J. W., Potapenko, A., Jayakumar, S. M., and Lillicrap, T. P. Compressive transformers for long-range sequence modelling. In *International Conference on Learning Representations*, 2020.

Ramachandran, P., Zoph, B., and Le, Q. V. Searching for activation functions. arXiv preprint arXiv:1710.05941, 2017.

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

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

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

Tsanas, A. and Xifara, A. Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. *Energy and Buildings*, 49:560–567, 2012.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

A Reproducibility Information

The reported experiments were implemented in notebooks covering the single-unit microscope, true dense 128×128 MLP composition, and PG-19 Transformer feed-forward replacement. The experimental implementations include model definitions, dataset preparation, training loops, evaluation code, random seeds, metrics, and figure generation commands.

All reported experiments use public or synthetic data. The single-unit microscope uses a synthetic controlled heteroscedastic regression task. The dense MLP composition experiment uses the UCI Energy Efficiency dataset. The Transformer feasibility experiment uses PG-19, a public language-modeling benchmark derived from Project Gutenberg books.

B Datasets and Licenses

The UCI Energy Efficiency dataset is credited to Tsanas and Xifara and distributed by the UCI Machine Learning Repository under a Creative Commons Attribution 4.0 International license. PG-19 is credited to Rae et al. and the DeepMind PG-19 benchmark repository, which identifies the dataset license as Apache License 2.0. The synthetic heteroscedastic regression task is generated by the experimental code and introduces no external dataset.

C Compute Resources

For the true dense 128×128 MLP composition experiment, EVE takes 210.6 ± 2.1 seconds per run versus 29.5 ± 0.2 seconds for DET, corresponding to a $7.1\times$ runtime overhead in the current implementation. The overhead reflects local posterior computation, sampling, and probabilistic evaluation. The experimental configurations specify the software environment, package versions where available, random seeds, and execution order.

D Broader Impacts

This work is foundational research on neural-network computational units. Potential positive impacts include improved uncertainty-sensitive prediction and more inspectable internal diagnostics. Potential negative impacts follow from general use of improved neural models in downstream systems, especially if uncertainty

estimates are misinterpreted or deployed without calibration, monitoring, or domain validation. The reported experiments do not release a high-risk pretrained model and are not tied to a deployed decision system.

Competing Interests

The author declares no competing interests.