Improve Predictive Model Calibration for Safety-Critical Domains

Le Van Tuan Anh*
anh.lvt2416657@sis.hust.edu.vn

Nguyen Tran Minh Tri*
tri.ntm2400116@sis.hust.edu.vn

Le Xuan Nhat Khoi*
khoi.lxn2416711@sis.hust.edu.vn

Nguyen Duy Tung*
tung.nd2416762@sis.hust.edu.vn

Abstract

Reliable uncertainty estimation is crucial for deploying modern neural networks in high-stakes scenarios, yet standard deterministic models trained with conventional loss functions often become systematically overconfident in their predictions. In this work, we propose a Bayesian neural network (BNN) framework that improves model calibration by replacing traditional point-estimate training with optimization of the evidence lower bound (ELBO). Our approach learns a distribution over network weights, enabling principled uncertainty quantification and mitigating the overconfidence commonly induced by cross-entropy–based training. Empirical results demonstrate that our BNN formulation yields significantly better calibration, measured via expected calibration error and negative log-likelihood, while maintaining competitive predictive accuracy.

1 Introduction

Neural networks have achieved remarkable performance across a wide range of predictive tasks, yet their reliability remains a significant challenge in practical deployment. Standard deterministic models trained with conventional loss functions, most notably cross-entropy, tend to produce overly confident predictions, even when faced with ambiguous inputs or samples far outside the training distribution. This systematic overconfidence not only impairs decision-making but also limits the applicability of modern deep learning systems in safety-critical domains such as medical diagnosis, autonomous navigation, and scientific modeling.

Bayesian neural networks (BNNs) offer a principled framework for addressing this limitation by replacing point estimates of network weights with full posterior distributions. Instead of committing to a single set of parameters, BNNs represent uncertainty directly through the variability of sampled weight configurations, enabling more calibrated predictive distributions. However, training BNNs remains challenging due to the intractability of the exact posterior, which necessitates efficient and scalable approximations.

In this work, we revisit variational Bayesian learning as a practical solution to these challenges. We propose a BNN framework trained via maximization of the evidence lower bound (ELBO), which balances model fit with posterior regularization through a Kullback–Leibler divergence term. Unlike traditional losses that focus solely on maximizing predictive accuracy, the ELBO encourages the model to capture epistemic uncertainty by constraining the posterior to remain close to a meaningful prior. This leads to better uncertainty estimates, improved robustness to distribution shift, and reduced overconfidence in regions where data are sparse or ambiguous.

*Equal contribution

To assess the effectiveness of our approach, we conduct extensive empirical evaluations across classification tasks and calibration benchmarks. Our results show that ELBO-based training consistently improves calibration metrics, including expected calibration error and maximum calibration error, while maintaining competitive accuracy. These findings highlight the value of Bayesian training objectives in producing trustworthy and informative predictions.

2 Bayesian Inference

Bayesian inference provides a principled mathematical framework for reasoning under uncertainty by treating model parameters as random variables rather than fixed quantities. Given observed data $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ and model parameter θ, Bayes' rule defines the posterior distribution as:

$$p(\theta \mid \mathcal{D}) = \frac{p(\theta)p(\mathcal{D} \mid \theta)}{p(\mathcal{D})}$$

where $p(\theta)$ is a prior encoding assumptions about the parameters, $p(\mathcal{D} \mid \theta)$ is the likelihood of the data under the model, and $p(\mathcal{D})$ is the marginal likelihood (the evidence). The posterior captures all uncertainty about the parameters after observing data, and predictions are obtained by marginalizing over this uncertainty:

$$p(y^* \mid x^*, \mathcal{D}) = \int p(y^* \mid x^*, \theta)p(\theta \mid \mathcal{D})d\theta$$

2.1 Bayesian Neural Networks

A Bayesian Neural Network (BNN) applies Bayesian inference to deep learning by placing distributions over the network weights instead of learning deterministic parameters. Let θ represent all weights and biases of the network. In a BNN, we specify:

- A **prior** $p(\theta)$, typically chosen as a factorized Gaussian with zero mean
- A **likelihood** $p(\mathcal{D} \mid \theta)$, which depends on the model output (e.g., softmax likelihood for classification)

Because the neural network likelihood is highly nonlinear and the parameter space is high-dimensional, the posterior is **intractable**, we cannot compute it analytically or normalize it exactly. Likewise, exact predictive inference requires evaluating an integral over all possible weight values, which is computationally prohibitive.

2.2 Probabilistic Inference

To address this intractability, BNNs rely on approximate inference methods. The two primary families are:

- **Markov chain Monte Carlo (MCMC)**: Draws samples from the true posterior using iterative sampling schemes (e.g., Hamiltonian Monte Carlo). While theoretically accurate, MCMC scales poorly with deep architectures and large datasets.
- **Variational Inference (VI)**: Approximates the posterior by a simpler distribution $q(\theta)$ and selects $q(\theta)$ to minimize the divergence from the true posterior. VI is computationally efficient, scalable to large datasets, and compatible with gradient-based optimization.

Variational inference forms the foundation of most practical BNN implementations, as it enables amortized training through stochastic optimization and mini-batching.

2.3 Predictive Uncertainty

BNNs provide two forms of uncertainty:

- **Epistemic Uncertainty**: Arising from limited data or model ambiguity, captured through variability in $p(\theta \mid \mathcal{D})$

- **Aleatoric Uncertainty**: Representing inherent noise in the data, captured through the likelihood $p(y \mid x, \theta)$

Integrating over the weight posterior naturally combines both sources:

$$\mathbb{E}_{\theta \sim q}[p(y^* \mid x^*, \theta)]$$

3 Variational Inference and the Evidence Lower Bound

As discussed earlier, exact Bayesian inference in neural networks is intractable due to the high-dimensional and nonlinear nature of the weight posterior $p(\theta \mid \mathcal{D})$. Variational Inference (VI) provides a scalable alternative by reframing posterior inference as an optimization problem. Instead of computing the exact posterior, VI introduces a tractable family of distributions $q(\theta)$ and finds the member of this family that best approximates the true posterior.

3.1 Variational Approximation

We choose a variational distribution $q(\theta)$, typically a factorized Gaussian over weights:

$$q(\theta) = \prod_j \mathcal{N}(\theta_j \mid \mu_j, \sigma_j^2)$$

The goal is to make $q(\theta)$ close to the true posterior. VI does this by minimizing the reverse Kullback–Leibler divergence:

$$q^\star = \arg\min \mathbb{KL}[q \parallel p(\cdot \mid \mathcal{D})]$$

However, the posterior $p(\theta \mid \mathcal{D})$ is inaccessible, so this KL divergence cannot be evaluated directly. Instead, we rewrite it in a tractable form.

3.2 Deriving the ELBO

We begin with the reverse KL-divergence:

$$\begin{aligned}
\mathbb{KL}[q(\theta) \parallel p(\theta \mid \mathcal{D})] &= \mathbb{E}_{\theta \sim q}\left[\log \frac{q(\theta)}{p(\theta \mid \mathcal{D})}\right] \\
&= \mathbb{E}_{\theta \sim q}\left[\log \frac{p(y_{1:n} \mid x_{1:n}) q(\theta)}{p(y_{1:n}, \theta \mid x_{1:n})}\right] \\
&= \log p(y_{1:n} \mid x_{1:n}) \underbrace{-\mathbb{E}_{\theta \sim q}[\log p(y_{1:n}, \theta \mid x_{1:n})] - \mathrm{H}[q]}_{-\mathcal{L}(q,p;\mathcal{D}_n)}
\end{aligned}$$

Because KL divergence is always non-negative, the ELBO forms a lower bound:

$$\mathcal{L}(q, p; \mathcal{D}_n) \leq \log p(y_{1:n} \mid x_{1:n})$$

Thus maximizing the ELBO is equivalent to minimizing the reverse KL divergence. We can also express the ELBO in other forms:

$$\begin{aligned}
\mathcal{L}(q, p; \mathcal{D}_n) &= \mathbb{E}_{\theta \sim q}\left[\log p(y_{1:n}, \theta \mid x_{1:n}) - \log q(\theta)\right] \\
&= \mathbb{E}_{\theta \sim q}\left[\log p(y_{1:n} \mid x_{1:n}, \theta) + \log p(\theta) - \log q(\theta)\right] \\
&= \mathbb{E}_{\theta \sim q}\left[\log p(y_{1:n} \mid x_{1:n}, \theta)\right] - \mathbb{KL}[q \parallel p(\cdot)]
\end{aligned}$$

3.3 Stochastic Optimization and the Reparameterization Trick

Computing the gradients of the ELBO requires sampling from $q(\theta)$. VI employs the **reparameterization trick** to make this sampling differentiable:

$$\theta = \mu + \sigma \odot \epsilon$$

This transforms stochastic sampling into a deterministic function of θ and allows backpropagation through expectations:

$$\nabla \mathbb{E}\left[f(\theta)\right] = \mathbb{E}_{\epsilon \sim \mathcal{N}(0,(I))}\left[\nabla f(\mu + \sigma \odot \epsilon)\right]$$

As a result, ELBO maximization can be done efficiently using SGD or Adam, making variational BNNs scalable to large datasets.

3.4 Why ELBO Improves Calibration

Traditional losses such as cross-entropy encourage the logits of the predictive distribution to grow without bound, often producing overconfident outputs even for uncertain inputs.

In contrast, ELBO-based training:

- Enforces a distribution over weights, naturally reflecting epistemic uncertainty
- Constrains the posterior through the KL term, preventing extreme parameter values
- Averages predictions over multiple weight samples, smoothing the predictive distribution
- Penalizes overconfident predictions that are unsupported by data

This results in models that know when they don't know, leading to better-calibrated probabilities and more trustworthy predictions.

4 Architechture

Our Bayesian neural network framework builds upon a standard feed-forward architecture, but replaces deterministic weights with variational distributions. This section describes the overall model structure, the parameterization of the variational layers, and the modifications required to support ELBO-based training.

4.1 Overall Structure

The proposed model follows a conventional multilayer perceptron (MLP) architecture composed of L hidden layers:

$$f_\theta = f^{(L)} \circ f^{(L-1)} \circ \cdots \circ f^{(1)}(x)$$

where each layer consists of an affine transformation followed by a nonlinear activation.

In the deterministic case, each layer uses weights $W^{(\ell)}$ and biases $b^{(\ell)}$.

In our Bayesian formulation, these parameters are replaced by distributions:

$$W^{(\ell)} \sim q(W^{(\ell)}), \qquad b^{(\ell)} \sim q(b^{(\ell)})$$

The output layer uses a task-appropriate likelihood:

- **Softmax likelihood** for classification
- **Gaussian likelihood** for regression

4.2 Variational (Bayesian) Layers

Each weight tensor $W^{(\ell)}$ is modelled using a mean-field Gaussian distribution is modeled using a mean-field Gaussian distribution:

$$q(W^{(\ell)}) = \mathcal{N}\left(W^{(\ell)} \mid \mu_W^{(\ell)}, \sigma_W^{(\ell)2}\right)$$

and similarly for $b^{(\ell)}$. The parameter $\{\mu, \sigma\}$ are optimized during training.

During forward passes, weight samples are generated using the reparameterization trick:

$$W^{(\ell)} = \mu_W^{(\ell)} + \sigma_W^{(\ell)} \odot \epsilon^{(\ell)}, \quad \epsilon^{(\ell)} \sim \mathcal{N}(0, I)$$

Thus each model evaluation corresponds to a single draw from the approximate posterior.

4.3 Forward Pass and Monte Carlo Predictive Distribution

A single forward pass yields a stochastic prediction due to sampling from the posterior.

For calibrated predictions, we compute the predictive distribution by averaging over multiple samples:

$$p(y^* \mid x^*) \approx \frac{1}{S} \sum_{s=1}^{S} p(y^* \mid x^*, \theta_s)$$

In practice, using $S = 10$ - 20 balances predictive stability and runtime.

5 Experiments

5.1 Experiments Setup

We evaluate BNN Framework on prediction tasks on two famous image dataset, CIFAR-10 and CIFAR-100, with the following experimental set-up:

- Networks architecture: We use a basic CNN structure for image classification task, but due to a resource constraint, we will only evaluate a hybrid architecture, which will replace a baseline original classifier head, with a Bayesian Linear layer. This, even though limiting, will still improve calibration drastically.
- Baseline: We compare the propose framework with the same CNN architecture, without the Bayesian classifier, to see the difference one Bayes layer makes in model calibration.
- Evaluations: We consider the three following metrics: Accuracy (ACC), Expected Calibration Error (ECE) and Maximum Calibration Error (MCE). We will also be reporting a reliability diagram, showing where and when the model is miscalibrated.

5.2 Experimental Results and Analysis

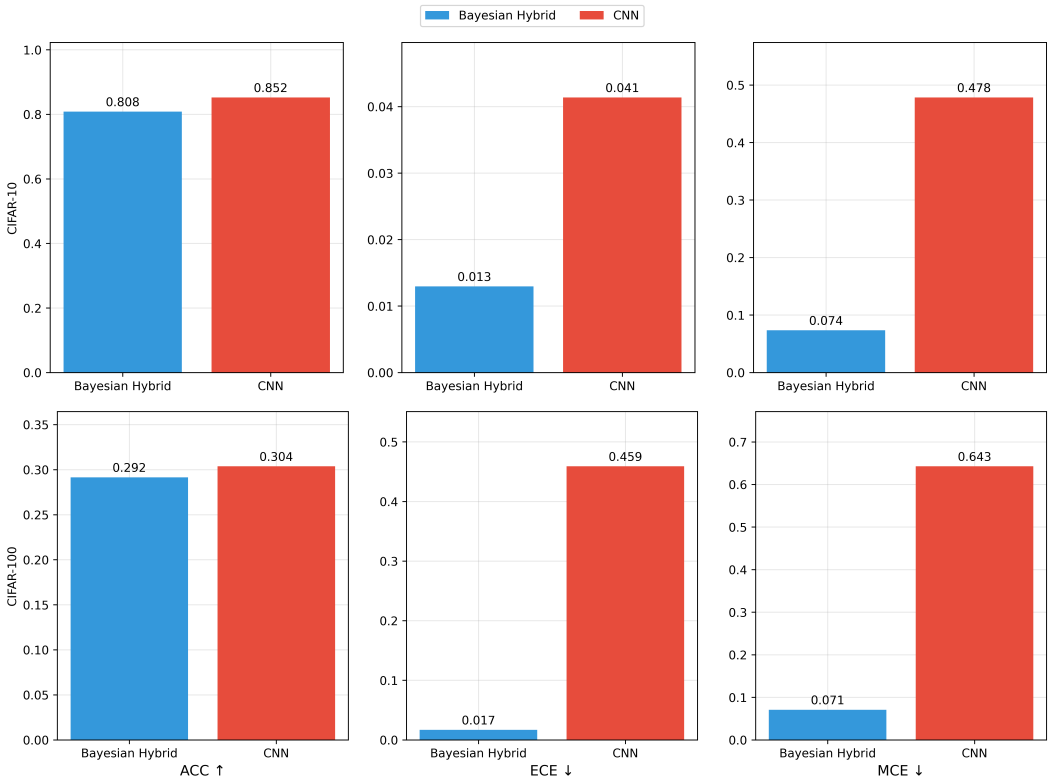

Figure 1: Test accuracy & calibration metrics of CNN or BNN-Hybrid trained on CIFAR-10 (1st row) or CIFAR-100 (2nd row)

The results demonstrate that the Bayesian Hybrid model successfully maintains a high level of predictive power while significantly improving the reliability of those predictions.

- **Accuracy (ACC)**: On both datasets, the standard CNN slightly outperforms the hybrid model in raw accuracy (e.g., 0.852 vs. 0.808 on CIFAR-10). This "accuracy tax" is a common phenomenon when introducing Bayesian layers, as the model prioritizes representing uncertainty over fitting a single deterministic point estimate.

- **Calibration Improvement**: The most striking finding is the drastic reduction in calibration error. On CIFAR-100, the CNN's ECE is 0.459, indicating a massive "confidence-accuracy gap." In contrast, the Bayesian Hybrid reduces this to just 0.017, an almost 95% reduction.

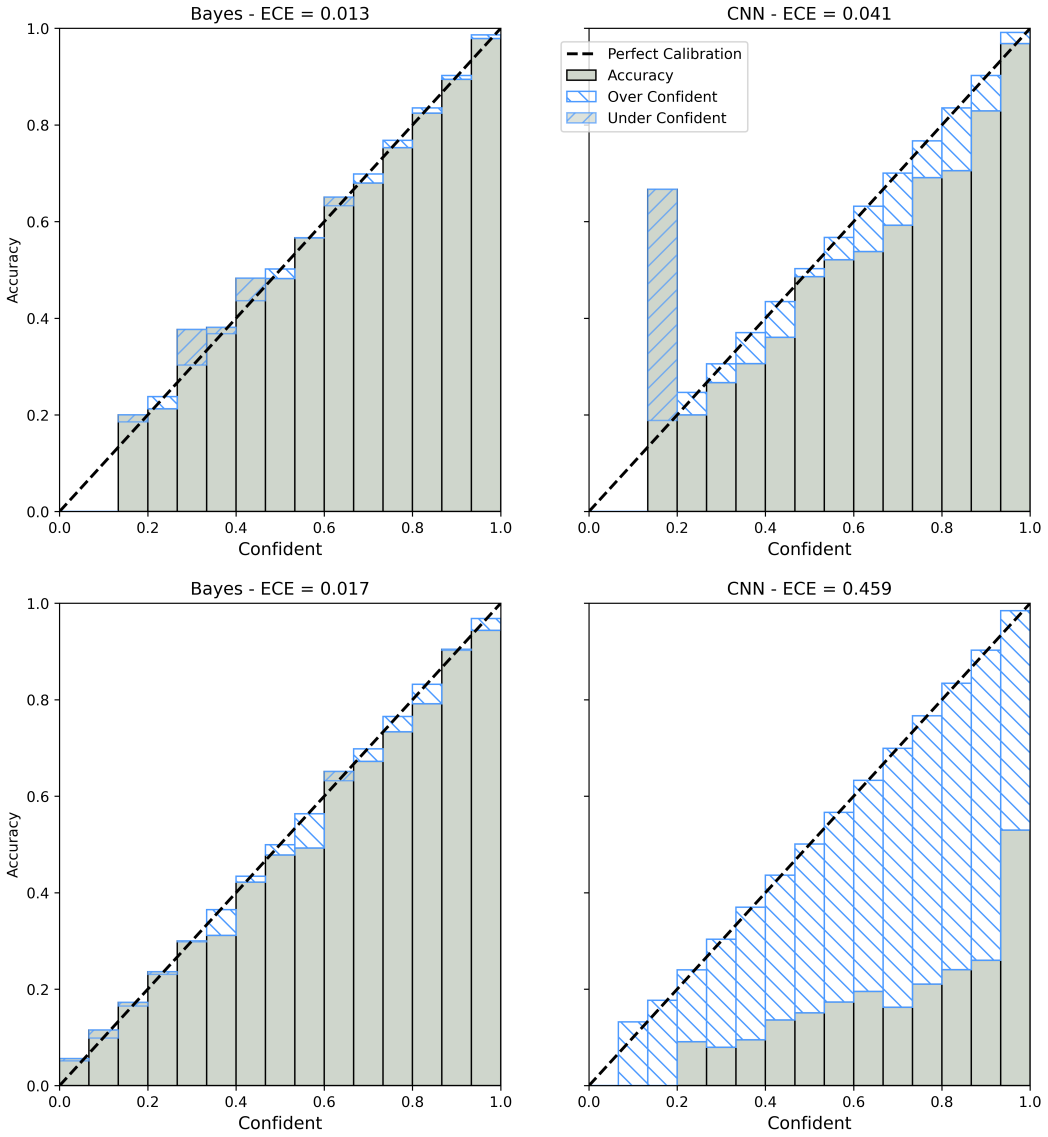

Figure 2: Reliability diagram of CNN or BNN-Hybrid on CIFAR-10 (1st row) or CIFAR-100 (2nd row)

On the simpler CIFAR-10 task, the CNN is already relatively well-calibrated but exhibits notable overconfidence in the high-confidence regime (0.6 to 1.0). The Bayesian Hybrid successfully "pulls" these predictions toward the identity line, reducing the ECE from 0.041 to 0.013. The resulting bins show a nearly seamless alignment between how sure the model is and how often it is correct.

The benefits of the hybrid approach are most striking in the high-complexity CIFAR-100 environment:

- **CNN Failure**: The standard CNN displays a severe calibration failure, with large "overconfident" gaps across the entire spectrum. Even when the model reports nearly 100% confidence, its actual accuracy is approximately 50%.

- **Bayesian Robustness**: Despite the lower overall accuracy inherent in the 100-class task, the Bayesian Hybrid remains exceptionally well-calibrated (ECE = 0.017). Even as the classes increase, the model effectively "tames" the overconfidence that typically plagues deep learning models, ensuring that its confidence scores remain a trustworthy proxy for actual performance.

These results confirm that the Hybrid framework, keeping the CNN backbone deterministic and only making the classifier head Bayesian, is a highly effective strategy. It provides the uncertainty quantification benefits of Bayesian Neural Networks while avoiding the massive computational costs of a fully Bayesian architecture, making it viable for training on standard hardware.

6 Bayesian Hybrid Framework for Medical Imaging

The results from the CIFAR-10 and CIFAR-100 experiments validate that the Bayesian Hybrid framework effectively mitigates overconfidence while maintaining competitive accuracy. However, in the context of general image recognition, a miscalibrated prediction (e.g., mistaking a "cat" for a "dog" with 99% confidence) results in a harmless error.

In contrast, the stakes are significantly higher in Medical Imaging. A deep learning model used for diagnostics must not only be accurate but also "know when it is uncertain." An overconfident misclassification of a malignant tumor as benign could lead to delayed treatment and adverse patient outcomes. Conversely, a well-calibrated model that expresses low confidence in ambiguous cases can signal a human radiologist to perform a manual review, thereby acting as a reliable safety net.

In this section, we apply the Bayesian Hybrid framework to a Brain Tumor Classification task. We aim to demonstrate that the calibration benefits observed in the CIFAR benchmarks translate to a real-world clinical setting, providing a more robust and trustworthy tool for medical decision support.

7 Conclusion