Reviewer 1:

Comments:

1. Full-precision neural networks, binarized neural networks and their gap (e.g., quantization) are not well-defined, so that the correctness of the proposed approach cannot be evaluated.
2. Thus, the loss function only directly affects the update of parameters of full-precision neural networks which are in turn quantized into binarized values… It is impossible to train a binarized neural network using the proposed approach to preserve claimed upper bound of robustness.
3. All the experiments are based on testing results instead of verification, there is a lack of formal guarantees.
4. There are lots of formal verification works on both binarized and quantized neural networks, the authors did not mention any of them (some of them are listed below).
5. The writing of the paper should be significantly improved, there are too many typos and errors.

Response:

Thank you very much for your constructive comments. **Comments 1, 3 and 4** are about the relationship to formal verification. Our work aims to provide an effective training framework for enhancing the robustness of Deep BNNs against general input noise perturbations by incorporating an L1-infinity norm constraint during training process, hence it is orthogonal to the topic of formal verification for verifying the correctness of the resulting quantized/binary NN, as illustrated in Figure 1 below:

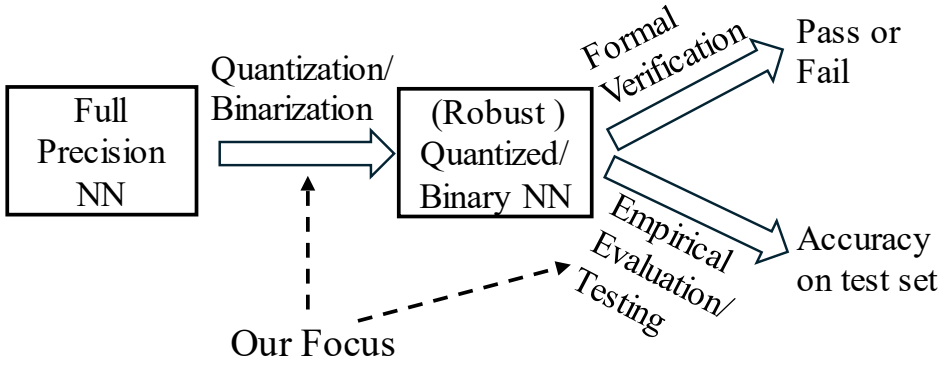

Figure 1. Workflow of quantization/binarization and performance evaluation.

In these remarkable verification researches [A-J], such researches only answer the Pass/Fail verification questions on the robustness property of well-established models (i.e., whether the trained model is pass or fail). If the given models do not pass the verification, we need to reconduct the training process. Such process is not efficient in practice.

The reviewer mentioned many papers on formal verification of quantized/binarized NNs [A]-[J], which can verify a given NN against a specific property, e.g., local robustness verification and maximum robustness radius computation in QVIP [F]. Our

work aims to produce a more robust Binary NN, with empirical evaluation based on testing on a test dataset, similar to our comparison baselines [18, 21-22]. Note that the "upper bound" derived in our paper refers to the Lipschitz constant of the NN, a metric that measures the maximum rate of change of the network's output with respect to changes in its input, but we do not address the formal verification problem (as we mentioned above). In real-world scenarios, the input noise may be unpredictable and encompass various types of noise unseen during the training phase, and without any upper bound on its radius or norm. Our work proposes a robust training algorithm for enhancing robustness against general input noise perturbations during the testing phase, and empirical experiments confirm its effectiveness.

The reason that our work and comparison baselines do not use formal verification is due to the well-known scalability issue that limits the size of NNs that can be formally verified with both soundness and completeness. Here is an excerpt from a recent survey paper: Adversarial Robustness of Deep Neural Networks: A Survey from a Formal Verification Perspective, CoRR abs/2206.12227 (2022).
"4.2 Trade-off in Practice
Verifying deep neural network models is shown as an NP hard problem in [26] and therefore it is challenging to implement verification that is simultaneously sound, complete, and able to terminate in a reasonable time. In practice, one of those three requirements are usually put aside to guarantee the other two characters. Some approaches (e.g., [26], [47]) put aside termination to ensure both soundness and completeness are preserved, which is widely adopted in the early stage of neural network verification due to the size of neural networks is comparably small, becomes a critical factor in the verification of deeper and more complicated neural networks. Considering that soundness is more essential in program analysis (to prevent what the system claims to prevent), recent verification solutions (e.g., [48], [49]) commonly choose to sacrifice completeness to secure soundness and scalability. Unlike the complete verification that comes with an exponential time complexity for the worst case, incomplete verifiers usually come with much better scalability but risk the verification from precision loss due to over-approximation. Since the precision loss is accumulated layer by layer, in the worst case, an incomplete verifier may fail to certify the robustness even though it terminates quickly."

Finally, we agree with the Reviewer1 that related works on formal verification should be discussed, and their relationship with our work should be clarified.

Regarding **Comment 2**, the reviewer's statement that "binarized neural networks do not have the same robustness of full-precision neural networks" is not 100% accurate. In fact, quantization/binarization can either improve or hurt model robustness compared to the original full precision NN. Here is an excerpt from the response of the search engine perplexity.ai on the relationship between quantization and robustness:
"Impact of Quantization on Adversarial Robustness. The effect of quantization on adversarial robustness is not straightforward and can vary depending on the attack

strength and quantization method used:
1. For simple gradient-based attacks with small perturbations, quantization can sometimes improve adversarial robustness compared to full-precision networks [3]. This is because quantization introduces non-differentiable operations that can obfuscate gradients and make it harder for the attack to find adversarial perturbations.

2. However, for stronger attacks with larger perturbations, quantized networks can be more vulnerable than full-precision networks [1-3]. The limited representational capacity of low-bit quantized networks makes them less robust to larger perturbations.

3. The robustness of quantized networks also depends on the quantization method used. Methods like quantization-aware training (QAT) that simulate quantization during training can improve robustness compared to post-training quantization [1].

[1] Moran Shkolnik, Brian Chmiel, Ron Banner, Gil Shomron, Yury Nahshan, Alex M. Bronstein, Uri C. Weiser. Robust Quantization: One Model to Rule Them All. NeurIPS 2020.
[2] Haowen Lin, Jian Lou, Li Xiong, Cyrus Shahabi. Integer-arithmetic-only Certified Robustness for Quantized Neural Networks. ICCV 2021: 7808-7817.
[3] Micah Gorsline, James Smith, Cory E. Merkel. On the Adversarial Robustness of Quantized Neural Networks. ACM Great Lakes Symposium on VLSI 2021: 189-194”

Regarding **Comment 5**, we will thoroughly check the grammar of our manuscript with tools such as Grammarly and proof-checking by professional services.

Reviewer 2:
Comments:
1. In terms of the expected runtime required to verify/analyze robustness or enhance robustness during training.
2. Moreover, there are typos that should be addressed.

Response:
Please refer to “Response for Reviewer1” on the quantization/binarization and performance evaluation. Our focus is on enhancing robustness of the DBNNs, not the original full-precision model, hence we cannot compare the runtime of analyzing or enhancing robustness of the full-precision NN.
Related to the reviewer’s question, we had conducted experiments before on the training and testing time for both our method and the comparison baseline Lipschitz continuity retained (LCR) [22] during the training and testing stages, as shown in Table below. Due to space limitations, this table was not included in the original submission, but it may be added in the final version. The training costs of our proposed method are reduced by 16% compared to LCR [22], as our $L_{1, \infty}$-norm constraint eliminates the need for approximation computing.

Table D2 Computational Overhead comparison between the our method and SOTA method on the ImageNet dataset. The notation (mm:ss) represents the unit of minutes and seconds in this table.

Methods	Training Time / Epoch (mm:ss)	Test Time/ Epoch (mm:ss)
Our	**56:08**	**4:49**
LCR [22]	66:37	6:30

Thank you for pointing out these typos, and we will correct them in the final version.

Reviewer 3:
Comment 1: This might limit the technical novelty of the work.

Response:
The technical novelty of our paper lies in the proposed L_1,∞ norm constraint, which surpasses the spectral norm constraint in enhancing the robustness of DBNNs under general input noise perturbations through quantitative analysis of a tighter upper bound on the Lipschitz constant compared to related work.

Comment 2: One may doubt the practical impact of these gains in real-world applications.

Response:
The improvements in robustness are in fact not insignificant in the context of related works on this topic, and related works on Deep Learning (e.g., image classification tasks) in general, where an improvement of a few percentages often represents significant progress or even a breakthrough. Please refer to the performance results of our comparison baselines [18, 21-22], which have similar scales of improvement, ranging from less than 1 percent to a few percent [18].

[18] Qin H, et al (2020) Forward and backward information retention for accurate binary neural networks. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)
[21] Qin H, et al (2023) Bibench: Benchmarking and analyzing network binarization. In: Proceedings of the International Conference on Machine Learning (ICML)
[22] Shang Y, et al (2022) Lipschitz continuity retained binary neural network. In: Proceedings of the European Conference Computer Vision (ECCV)

Table 3: Accuracy comparison with SOTA methods on CIFAR-10.

Topology	Method	Bit-width (W/A)	Acc.(%)
ResNet-18	FP	32/32	93.0
	RAD	1/1	90.5
	Ours[1]	1/1	**91.5**
ResNet-20	FP	32/32	91.7
	DoReFa	1/1	79.3
	DSQ	1/1	84.1
	Ours[1]	1/1	**85.4**
	Ours[2]	1/1	**86.5**
	FP	32/32	91.7
	DoReFa	1/32	90.0
	LQ-Net	1/32	90.1
	DSQ	1/32	90.2
	Ours[1]	1/32	**90.8**
VGG-Small	FP	32/32	91.7
	LAB	1/1	87.7
	XNOR	1/1	89.8
	BNN	1/1	89.9
	RAD	1/1	90.0
	Ours	1/1	**90.4**

Top-1 accuracy of IR-Net is also significantly better than that of the SOTA methods (*e.g.*, 58.1% vs. 56.4% for ResNet-18). The experimental results prove that our IR-Net is more competitive than the existed methods.

Table 4: Accuracy comparison with SOTA methods on ImageNet.

Topology	Method	Bit-width (W/A)	Top-1(%)	Top-5(%)
ResNet-18	FP	32/32	69.6	89.2
	ABC-Net	1/1	42.7	67.6
	XNOR	1/1	51.2	73.2
	BNN+	1/1	53.0	72.6
	DoReFa	1/2	53.4	–
	Bi-Real	1/1	56.4	79.5
	XNOR++	1/1	57.1	79.9
	Ours[2]	1/1	**58.1**	**80.0**
	FP	32/32	69.6	89.2
	SQ-BWN	1/32	58.4	81.6
	BWN	1/32	60.8	83.0
	HWGQ	1/32	61.3	83.2
	TWN	2/32	61.8	84.2
	SQ-TWN	2/32	63.8	85.7
	BWHN	1/32	64.3	85.9
	Ours[1]	1/32	**66.5**	**86.8**
ResNet-34	FP	32/32	73.3	91.3
	ABC-Net	1/1	52.4	76.5
	Bi-Real	1/1	62.2	83.9
	Ours[2]	1/1	**62.9**	**84.1**
	FP	32/32	73.3	91.3
	Ours[1]	1/32	**70.4**	**89.5**

Table 5: Comparison of time cost of ResNet-18 with different bits

Comment 3: Then, why not present a comparison of the implementation cost between this method and that of [22] in further experiments?

Response:

We had conducted experiments before on the training and testing time for both our method and the comparison baseline Lipschitz continuity retained (LCR) [22] during the training and testing stages. Due to space limitations, this table (Please refer to **response** reviewer2) was not included in the original submission, but it may be added in the final version.

Minor Comments and Suggestions:

1. This paper provides a descriptive example in Figure 2; however, the operational rules shown seem to differ from those of general neural networks. In particular, why isn't the value of a hidden neuron represented as the sum of all the neurons in the previous layer?

Response:

We realize that our drawing is not accurate, and the weights w11, w12… should be annotations on the edges, not on the nodes. We will correct the figure to reflect the operation of an NN.

2. If the aim is to demonstrate a tighter bound, the ratio should ideally be less than 1. However, the authors mentioned the right-hand side of Equation (17) grows rapidly as n increasing.

Response:

We realize that our drawing is not accurate, and the weights w11, w12… should be

annotations on the edges, not on the nodes. We will correct the figure to reflect the operation of an NN. We realize that there may be some ambiguity in the discussion. Here, the symbol n denotes the dimension of the binary convolution weights. In fact, this is a definite value and does not change as the number of network layers increases. In addition, the upper bound of LCR [22] increases exponentially and its value is much larger (due to the value of hyperparameters $\gamma>1$ in the LCR method) than the upper bound of our work, which means that the denominator of Eq. 17 is always greater than the numerator, that is to say, the ratio is strictly less than 1.

3. In this paper, where quantization reduces weights to {-1,1}, would the Lipschitz constant derived from the L-infinity Norm exceed 1? Could perturbations indeed expand with more layers, potentially compromising robustness? These critical points deserve further discussion and investigation.

Response:
We express our gratitude to the reviewer for your constructive suggestions. Yes. When multiplied by the L-infinity norm, this value will be greater than 1. We will discuss the effect of different intervals of this value on the robustness in the future.