Is Uncertainty Quantification in Deep Learning Sufficient for Out-of-Distribution Detection?

Adrian Schwaiger , **Poulami Sinhamahapatra** , **Jens Gansloser** , **Karsten Roscher**
Fraunhofer IKS, Fraunhofer Institute for Cognitive Systems
{firstname.lastname}@iks.fraunhofer.de

Abstract

Reliable information about the uncertainty of predictions from deep neural networks could greatly facilitate their utilization in safety-critical applications. Current approaches for uncertainty quantification usually focus on in-distribution data, where a high uncertainty should be assigned to incorrect predictions. In contrast, we focus on out-of-distribution data where a network cannot make correct predictions and therefore should always report high uncertainty. In this paper, we compare several state-of-the-art uncertainty quantification methods for deep neural networks regarding their ability to detect novel inputs. We evaluate them on image classification tasks with regard to metrics reflecting requirements important for safety-critical applications. Our results show that a portion of out-of-distribution inputs can be detected with reasonable loss in overall accuracy. However, current uncertainty quantification approaches alone are not sufficient for an overall reliable out-of-distribution detection.

1 Introduction

Many state-of-the-art methods for solving perceptual tasks are based on *Deep Neural Networks (DNNs)*. However, the lack of interpretability of these networks is still a problem when DNNs are employed in safety-critical applications, e.g., for autonomous driving or in medical diagnosis. In these domains, mistakes are not just a minor annoyance but can have severe consequences. Therefore, thorough safety analysis and argumentation are an integral part of the development of such systems. Unfortunately, the black-box nature of DNNs and the fact that already slight changes in the input can have drastic effects on the output make this task almost impossible for complex DNN-based computer vision pipelines.

One approach to address this problem is quantifying the predictive uncertainty of a DNN for each given input. Reliable uncertainty estimates can be utilized by a safety-envelope [Weiss *et al.*, 2018] that encapsulates the high-performance DNN. Whenever the uncertainty of a prediction is high, the result of the DNN is discarded and the prediction of a verified, lower-performance safety path is used instead. In this context, the performance of different *Uncertainty Quantification (UQ)* approaches for DNNs have already been investigated on *In-Distribution (ID)* data, i.e., data that is conceptually similar to the data the network has been trained on [Henne *et al.*, 2020]. However, the viability of UQ to detect *Out-of-Distribution (OOD)* inputs, i.e., data that differs strongly from the training data, is still an open question. The detection of such inputs is important, as DNNs are not able to provide a correct prediction for them. For instance, a network trained to distinguish between cats and dogs will always output one or the other, and very often with high confidence, even when challenged with an OOD sample, e.g., with the image of a car. As it is not feasible to construct a dataset that guarantees the coverage of all relevant concepts in sufficient quantity for open world scenarios, approaches to detect OOD inputs are important to ensure the safety of the overall system and to detect violations of its operational design domain.

In this paper, we investigate several state-of-the-art methods for UQ in combination with popular DNN architectures for image classification. We use three datasets from different application domains to train the models and apply them to test sets containing in- and out-of-distribution samples. We focus on the trade-off between remaining accuracy and remaining error under the assumption that inputs with uncertain predictions are handled by a fallback mechanism and therefore account to neither of them. Since the acceptable remaining error or minimal performance may vary from application to application we highlight the relationship between the two instead of assuming arbitrary limits.

2 Related Work

Machine Learning in Safety-Critical Domains Arguing the safety of *Machine Learning (ML)* algorithms for complex tasks still remains an open research question. Insufficiencies of DNNs on perception tasks include, e.g., susceptibility towards distributional shifts and lack of interpretability, and general mitigation strategies for e.g., the incorporation of uncertainty and proper specification of the data acquisition pro-

cess, as discussed in [Willers *et al.*, 2020]. One way to argue the safety is by formulating confidence arguments to gather evidence for the performance of an ML system [Burton *et al.*, 2019]. To aid the formulation of such arguments, the authors provide an overview of the most common failure cases and propose assurance claim points to further break down the task. Another direction in the domain of autonomous vehicles to assure the safety is the creation of a specification based on formal rules and physical constraints, as it is done within RSS [Shalev-Shwartz *et al.*, 2018]. As this approach implicitly requires perfect perception, which in a real-world scenario is unattainable, PURSS [Salay *et al.*, 2020] has been proposed as an extension to allow the integration of perceptual uncertainty into the otherwise rigid specifications.

Interpretability The lack of interpretability of DNNs is a hindrance for using them in safety-critical applications, as it makes thorough safety analyses almost impossible. One approach to address this problem is the visualization of the learned features and their interplay with each other [Olah *et al.*, 2020]. While this only enables qualitative analyses, the authors suggest that it can aid in gaining a better understanding of DNNs and facilitate other work in this domain. A different direction is the formation of human-understandable features in DNNs. For instance, by specifying desired concepts a network can be incentivized to learn corresponding features which in turn can be used for quantitative analyses [Kim *et al.*, 2018].

Verification of DNNS The ability to verify DNNs would facilitate any safety argumentation greatly. Approaches concerning the verification include linear approximations of the learned function in order to subsequently solve them using existing verification tools [Katz *et al.*, 2017]. The problems of scalability and the definition of proper specifications, however, prevent their application to complex perception tasks. Nevertheless, it is an active field of research and promising approaches exist, e.g., for the verification of direct perception utilizing an input property characterizer and an approach to verification based on assumed guarantees [Cheng *et al.*, 2019].

Out-of-Distribution Detection In real-world machine learning applications, the importance of detecting OOD samples in the test data, which basically indicates distributional shift from training data, is paramount. It has been recognized as an important problem for AI safety [Amodei *et al.*, 2016]. Neural Network classifiers tend to incorrectly classify OOD samples with high confidence. The high-confidence predictions are often the result from the softmax functions, since these probabilities are computed with the fast-growing exponential function, where minor input addition can lead to substantial increase in output. In this direction, [Hendrycks and Gimpel, 2018] proposed a baseline method to detect OOD samples based on an observation that a well-trained neural network tends to assign higher softmax scores to ID samples than OOD samples. This approach was further extended in ODIN [Liang *et al.*, 2018] by using temperature scaling in the softmax function [Guo *et al.*, 2017], and adding small controlled perturbations to inputs such that the softmax score gap between ID and OOD samples is further enlarged. Here, while the network is trained with the default softmax, during test phase the tempered softmax forces the network to be sure with its decisions. In [DeVries and Taylor, 2018], the authors propose Learned Confidence estimates to classify a sample as ID or OOD sample by appending a confidence estimation branch to the network. Similar to this, Metric Learning [Masana *et al.*, 2018] adds an additional output branch and maps it into a manifold where the Euclidean distance from such manifolds is used as a measure of detecting possible OOD samples. A probabilistic approach given in [Lee *et al.*, 2018] uses features (lower and upper level) from any pre-trained classifier and maps them into class conditional Gaussian distributions under Gaussian discriminant analysis, which result in a confidence score based on the Mahalanobis distance. Finally, the most popular method of computing probabilistic statistics uses the ensembles of predictions of discriminative classifiers trained on ID data, as proposed by [Lakshminarayanan *et al.*, 2017]. It has emerged as popular non-Bayesian approach for predictive UQ, also used for detecting OOD samples during inference. An alternative direction of approaching the OOD detection problem is the use of generative model-based methods, which are appealing as they do not require labeled data and directly model the input distribution. These methods fit a generative model $p(x)$ to the ID data, and then evaluate the likelihood of new OOD inputs under that model as in [Ren *et al.*, 2019], [Serrà *et al.*, 2019]. Moreover, many self-supervised approaches [Hendrycks *et al.*, 2019], [Mohseni *et al.*, 2020], which also do not need labeled data, have shown promise in OOD detection, often with accuracy comparable to supervised methods.

3 Uncertainty Quantification for OOD Detection

In the previous section, dedicated OOD detection techniques have been presented. However, it is reasonable to investigate the usage of UQ for this task as well. The idea is that a DNN should assign a high uncertainty to OOD inputs, as nothing comparable has been encountered before.

In [Osawa *et al.*, 2019] the authors, i.a., compare Bayesian UQ methods wrt. to their performance of detecting OOD samples. Although their results are promising, the chosen task is not as complex, because the defined datasets for ID and OOD are very dissimilar. The performance of different uncertainty quantifiers to distinguish samples from more similar distributions have been investigated in [Pawlowski *et al.*, 2017]. Their findings are promising and also encourage further research in that area.

3.1 Predictive Uncertainty Quantification of DNNs

A common approach to probabilistic UQ for neural networks is to rely on Bayesian methods (e.g., variational Bayes or Markov chain Monte Carlo), where the posterior distribution over the network parameters is computed. However, exact Bayesian inference is usually intractable, thus the posterior can only be computed approximately. Recently, non-Bayesian methods gained in popularity, which often allow for simpler implementation and faster training. In this work, we focus on methods for predictive UQ that are fast to train, rea-

sonably easy to implement and suitable for large-scale problems often seen in image classification tasks.

A straightforward approach to UQ is to interpret the classification scores as probabilities, e.g., by applying the softmax function to the prediction scores. However, modern DNNs tend to be not well calibrated, i.e., the predicted probability for an input sample does not represent the true accuracy of the network. This is especially true for DNNs with high model capacity and lack of regularization [Guo *et al.*, 2017]. One approach for DNN calibration is to learn a scaling of the predicted probabilities using a validation set, where the parameters of the DNN are fixed.

In addition to that, softmax probabilities viewed alone are often overconfident for OOD samples [Gal and Ghahramani, 2016]. Nevertheless, for a given network ID samples tend to have greater softmax values than OOD samples, which can be used as a baseline for OOD detection [Hendrycks and Gimpel, 2018].

Deep Ensembles Ensembles of deep neural networks, i.e. deep ensembles, is a well-known method to improve prediction accuracy. However, deep ensembles can also be used as a non-Bayesian uncertainty estimator [Lakshminarayanan *et al.*, 2017]. A number of randomly initialized neural networks are trained independently on the same training data. To compute the predictive distribution, the individual prediction probabilities of all neural networks in the ensemble are averaged. Additionally, [Lakshminarayanan *et al.*, 2017] propose to use proper scoring functions as loss functions and adversarial training to smooth the predictive distributions.

Monte-Carlo Dropout MC-Dropout can be interpreted as a form of ensembles with shared network parameters or alternatively, as approximate Bayesian inference [Gal and Ghahramani, 2016]. Usually, dropout is used during training for regularization to prevent overfitting. However, dropout can also be used during inference to estimate the predictive distribution. The empirical predictive mean and variance are calculated from multiple stochastic forward passes, where each forward pass can be seen as sampling from a posterior distribution over the network weights. Since MC-Dropout does not require any change in the network architecture, it is easy to implement and to use with existing architectures.

Learned Confidence A different, sampling-free approach to estimate uncertainty is proposed in [DeVries and Taylor, 2018] where the network learns an explicit confidence score as second optimization objective. A confidence layer is added after the last network layer, in parallel to the class prediction layer. The optimization objective is then the sum of the classification loss and the confidence loss.

Evidential Deep Learning Evidential Deep Learning [Sensoy *et al.*, 2018] is inspired by the Dempster-Shafer theory and another sampling-free approach. For classification tasks the parameters of a Dirichlet distribution are learned, from which the total evidence for each of the classes and the epistemic uncertainty regarding the prediction as a whole can be calculated. The authors also conducted some experiments regarding OOD detection and showed that their method generally assigned higher uncertainties to OOD inputs.

4 Evaluation

In the following, the previously presented UQ methods, Deep Ensembles (DE), Monte-Carlo Dropout (MCDO), Learned Confidence (LC), and Evidential Deep Learning (EDL), are compared to each other and to the default softmax confidences, which serve as a baseline. The task, hereby, is to classify images correctly and confidently.

4.1 Experimental Setup

To provide a comprehensive comparison, we trained each of the UQ methods on three different model architectures. VGG16 [Simonyan and Zisserman, 2015] as a standard network architecture, SqueezeNet [Iandola *et al.*, 2016] for its small size and suitability for embedded systems, and the recently introduced EfficientNet [Tan and Le, 2019] as a high-performing and efficient architecture. The model variant B0 for EfficientNet was adopted for our use-cases. All models use dropout regularization to allow the application of MCDO. Each deep ensemble consists of 5 networks and the number of sampling steps for MCDO has been set to 50. Increasing the number of members or sampling steps further lead only to minor improvements. For LC the last dense layer of each model is replaced by a prediction and a confidence branch, which then are concatenated again to form the final prediction, as in [DeVries and Taylor, 2018]. Additionally we set the hyperparameters for the loss function of LC to $\lambda = 0.1$ and $\beta = 0.3$, which generally showed the best results in our experiments. For EDL, using softplus as evidence function in combination with the expected cross entropy loss employing the digamma function, as described in [Sensoy *et al.*, 2018], yielded the best results and is used in all experiments presented in this paper.

As training datasets we used CIFAR-10, German Traffic Sign Recognition Benchmark (GTSRB) [Stallkamp *et al.*, 2011], and NWPU-RESISC45 [Cheng *et al.*, 2017]. CIFAR-10 contains small images separated into 10 different classes, e.g., automobile, truck or dog. GTSRB is a collection of German traffic signs. The number of classes amounts to 43. NWPU-RESISC45 has larger aerial images which are categorized into 45 different classes, e.g., forest, freeway or railway station. Additionally, we used images from CIFAR-100 as OOD samples for CIFAR-10 and Belgium Traffic Signs (BTSRB) [Timofte *et al.*, 2014] as OOD samples for GTSRB. While CIFAR-100 and CIFAR-10 already have distinct classes, for BTSRB we only included classes that had no equivalent in GTSRB. As we found no suitable OOD datasets for NWPU-RESISC45, we split it into two datasets. The OOD dataset includes 9 classes, airplane, airport, beach, harbor, island, lake, river, sea ice, and ship. These are semantically separated from the remaining 39 classes used as ID dataset. Overall, the ID and OOD dataset pairs are quite similar to each other, which makes the task of OOD detection more difficult. This was done purposefully, as it transfers better to safety-critical applications, where OOD inputs must be detected, coming from the exact same sensor in similar environments.

We trained the models from scratch using random initializations and used Adam as optimizer. Early stopping has been applied if the validation loss did not change for several epochs

to prevent overfitting whilst ensuring fully trained networks. Augmentations have not been applied, to rule out potential side effects introduced by the specific configuration used.

4.2 Evaluation Metrics

Following the cue of our previous work in [Henne *et al.*, 2020], similar evaluation metrics have been used in this paper. It constitutes of maximizing *Remaining Accuracy Rate (RAR)* along with minimizing the *Remaining Error Rate (RER)*. RAR takes into account the number of samples which have been correctly classified by the classifier as well as declared confident ("certain" and "correct") for a given threshold by the respective UQ method. RER on the other hand is the fraction of inputs that is classified incorrectly but with a high confidence ("certain" and "incorrect").

All trained networks were evaluated first on a test set with only ID data. Subsequently, the same model was tested on a second test set where OOD samples corresponding to 17.65% of the size of the ID data were added, to obtain a dataset with 85% ID and 15% OOD samples. The amount of ODD samples was chosen arbitrarily to improve the visual presentation of the plots. However, it has no impact on the overall observations since we focus on the relative performance between best and worst case.

4.3 Results and Discussion

Remaining Accuracy and Error

The results are shown in Figure 1. Due to space restrictions, the graphs for VGG16 could not be included. Each curve consists of the RAR and RER plotted for each threshold $t \in [0; 1]$ with a sampling step size of 0.001. The blue curves represent the performance on the ID dataset, the green curves show the performance on the dataset with combined ID and OOD samples. Furthermore, the green curves have been normalized regarding the RAR by the amount of OOD samples. Thereby, the influence on the accuracy due to additional OOD samples is eliminated and only the error introduced by them is factored in. For one, this better represents the application case, as the DNNs are not supposed to classify OOD samples correctly and only have to detect them. Second, due to the normalization, the behavior regarding the OOD detection can be better interpreted visually. Given a perfect OOD detection method, both curves would be the same, as all OOD samples would be rejected. The black curves show the worst case, i.e., if none of the OOD samples are rejected. They also have been normalized like the green curves.

On GTSRB, DE can detect most of the OOD samples, with a minor loss in accuracy. Although SqueezeNet has a slightly lower base accuracy, it is slightly better in rejecting OOD samples. For GTSRB, this also holds for MCDO and softmax. EDL on the other hand shows a better OOD discrimination ability in the other two architectures for higher RER, but can reduce the error almost completely with the highest accuracy left. LC performs sub-par with SqueezeNet, which might be due to the low number of parameters, as we already noticed in [Henne *et al.*, 2020].

On CIFAR-10 using EfficientNet, all but DE perform equally with only minor differences. An exception to this are softmax and MCDO, which for an RER of $< 3.5\%$ drop in a straight line, suggesting that there are no thresholds which can produce error rates in that range. For error rates $< 0.5\%$, all but softmax show the same accuracy. Upon further investigation, we noticed the distribution of classes among the undetected OOD samples were similar, hinting towards samples that are universally hard to reject. For SqueezeNet, DE significantly shows the best performance, followed by EDL. Softmax and MCDO perform equally and LC again performs the worst using this architecture. Using VGG16, DE still outperforms the other methods but the difference is much less significant. EDL and MCDO more or less perform equally, with EDL being slightly better for high RER and MCDO being slightly better for really low RER. LC and softmax also show similar performance. Softmax again is not able to produce different RER in lower ranges, however, this can mostly be attributed to the ID samples.

On NWPU-RESISC45, DE performs the best for EfficientNet in terms of maximum RAR achieved. Next, softmax and MCDO behave similarly but with a slight decrease in RAR. For RER $< 5\%$, both of these methods show a slight kink in the curve showing their sensitivity to certain range of thresholds. But even in this range, DE clearly achieves much better RAR at the cost of $< 1\%$ RER. LC tries to achieve close to 80% RAR, but at the cost of much higher RER. Finally, EDL performs similarly as others in RER $< 5\%$, but is vastly outperformed in terms of overall RAR. For SqueezeNet, DE again performs the best followed by MCDO, EDL and softmax in close proximity. Nonetheless, LC as pointed out earlier performs the worst with this architecture. For VGG16, all the methods perform sub-optimally in comparison to other architectures with maximum RAR of nearly 75% achieved by DE. Similar to the trend above, DE is followed by EDL with comparable RAR, as DE, in RER $< 5\%$. Softmax and MCDO follow them, but spread over larger RER. LC is not again able to produce RER in lower ranges and has much larger RER as compared to similar RAR achieved by other UQ methods.

Based on the observations above, DE performs best across all methods and datasets. LC had been originally proposed as an OOD detection method rather than being an UQ method. However, LC has shown consistent sub-optimal performance in almost all the scenarios above, particularly with smaller architectures like SqueezeNet or higher resolution dataset, like NWPU-RESISC45. MCDO and softmax perform averagely in most cases. Most UQ methods including EDL tend to have quite competent RAR for lower RER ranges, but on initial investigation it has been also observed there always exist some harder sample categories which are almost too difficult to certainly reject for most UQ methods.

Quality of Uncertainty Estimation

To further assess the novelty detection capabilities of the methods, we show the ratio of inputs marked as uncertain for a given threshold. We, thereby, show the comparison for the three possible cases: ID inputs predicted correctly, ID inputs predicted incorrectly and OOD inputs. Corresponding to each of the three cases we plot, over all thresholds, the fraction of samples having high uncertainty. An ideal method, for some given threshold, is certain for all correct predictions and

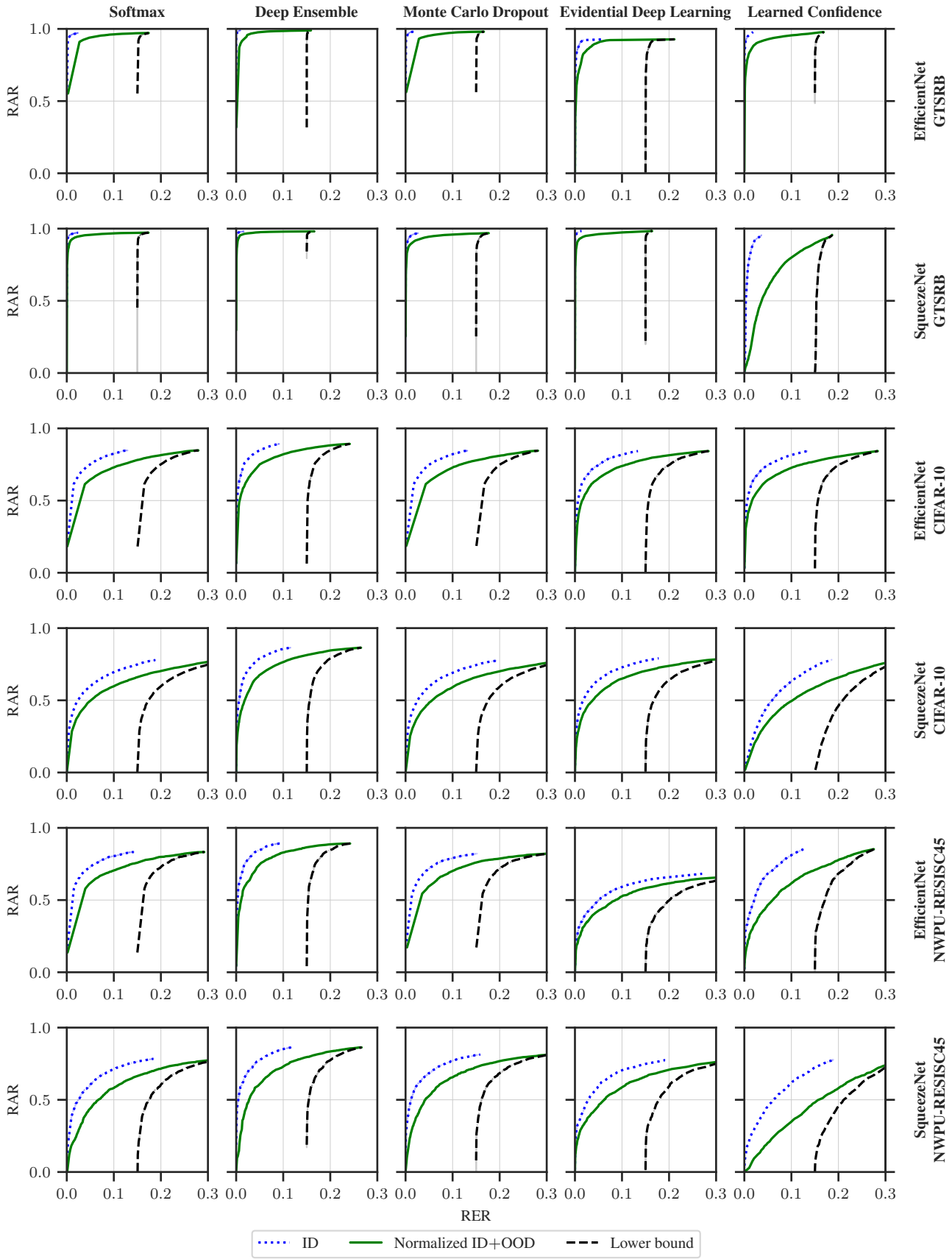

Figure 1: Remaining Error Rate (RER) vs. Remaining Accuracy Rate (RAR) for EfficientNet and SqueezeNet on the GTSRB, CIFAR-10 and NWPU-RESISC45 datasets. The plots show the performances first on the ID dataset(blue), then on dataset consisting of the ID and OOD samples(green). The lower bound (black) represents the worst-case scenario where the network fails to reject none of the OOD sample.

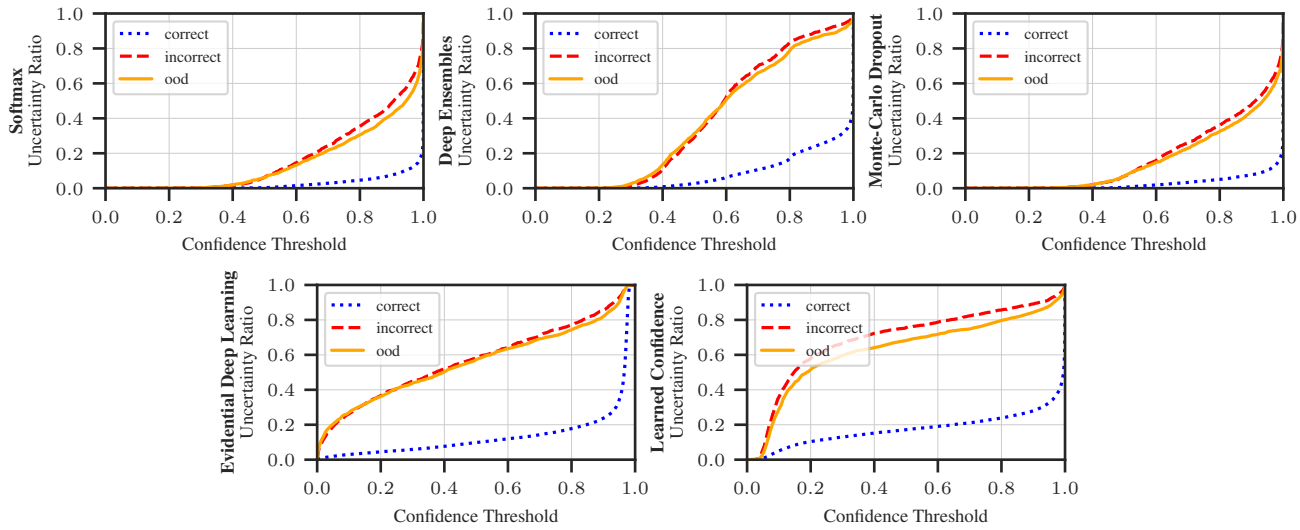

Figure 2: The ratio of inputs marked as uncertain for the three cases — correctly classified, incorrectly classified, and OOD inputs — over the range of thresholds for EfficientNet on CIFAR-10.

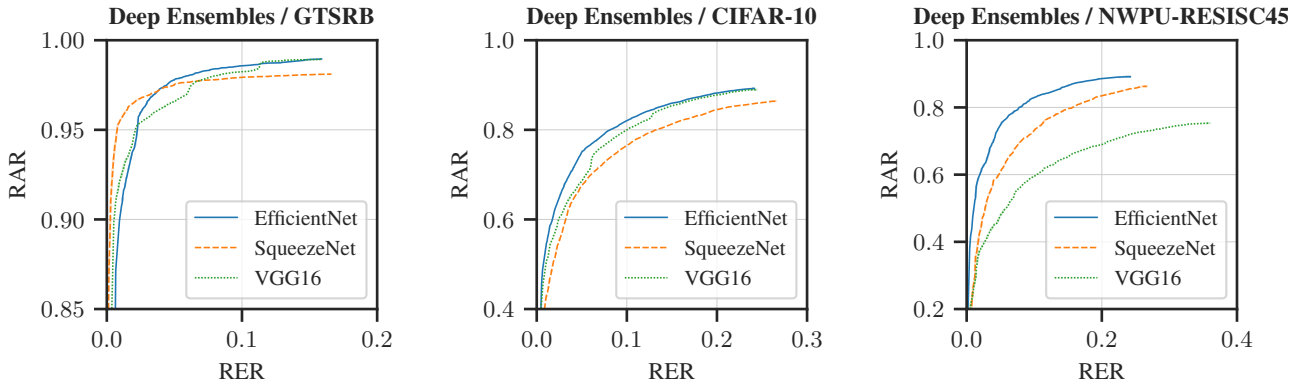

Figure 3: Remaining Error Rate (RER) vs. Remaining Accuracy Rate (RAR) for Deep Ensembles on the normalized ID+OOD dataset trained on GTSRB, CIFAR-10 and NWPU-RESISC45 with EfficientNet, SqueezeNet and VGG16.

uncertain for all incorrect predictions as well as predictions for OOD inputs. In Figure 2, the uncertainty ratios are shown only for CIFAR-10 and EfficientNet, but we observe the same findings in our other considered configurations. Most interestingly, the curves for incorrectly classified and ood inputs match very closely. Obviously, this raises the question: How correlated are these two categories and will better UQ methods be able to better detect novel inputs? This could be subject for future research. The plots again indicate that very low error rates can only be achieved at the cost of sacrificing a lot of accuracy. It is also worth mentioning, that EDL and LC exhibit a smoother behavior over the range of thresholds, especially compared to Softmax and MCDO, and therefore, are less sensitive towards small changes in the choice of a threshold.

Influence of Model Architecture

While the choice of architecture is important for the performance wrt. accuracy, its influence on the OOD detection ability is not as significant, visually represented by how close the blue and green curves match. An exception to this are some configurations with LC, especially with SqueezeNet. On CIFAR-10, all architectures perform mostly the same regarding their novelty detection ability and on the easier dataset GTSRB SqueezeNet has a slight edge. For NWPU-RESISC45, VGG16 rejects OOD samples slightly better, however, its baseline accuracy is about 20% lower for all UQ methods. Figure 3 shows the overall performance of DE for all architectures on the combined ID + OOD datasets.

5 Conclusion and Future Work

In this paper, we investigated the question, whether uncertainty quantification is sufficient for detecting out-of-distribution inputs. To that end, we applied different state-of-the-art methods and network architectures to three image classification tasks. While all tested UQ methods assign high uncertainty to some of the ODD samples, their rejection capabilities will not suffice for most safety-critical applications,

especially considering that in the real-world even more difficult OOD inputs can occur. If UQ should be applied, deep ensembles consistently showed the best trade-off between performance and remaining error, but mostly due to its better accuracy baseline to begin with.

A closer look at our results revealed that in many cases all methods fail on ODD inputs from the same classes. This hints at the possibility that certain ODD inputs are conceptually harder (or even impossible) to identify either by UQ methods or even in general. However, further research is needed to provide more evidence. In addition, many novelty detection approaches have been proposed in recent years. It would be interesting to see how they perform compared to the UQ methods presented here. Furthermore, their error patterns may provide additional insights into the difficulties of OOD detection in general.

Additionally, it is worthwhile investigating, whether our findings also transfer to other tasks, e.g., object detection or instance segmentation, and to other types of input data, for instance, radar or lidar point clouds. While there are similar base components at play —object detectors even use the investigated networks as feature extractors—, the transferability of our results is not guaranteed.

Acknowledgments

This work was partially supported by the Bavarian Ministry of Economic Affairs, Regional Development and Energy through the Center for Analytics—Data—Applications (ADA-Center) within the framework of "BAYERN DIGITAL II" and within the Intel Collaborative Research Institute—Safe Automated Vehicles.

References

[Amodei *et al.*, 2016] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete Problems in AI Safety. *ArXiv160606565 Cs*, July 2016.

[Burton *et al.*, 2019] Simon Burton, Lydia Gauerhof, Bibhuti Bhusan Sethy, Ibrahim Habli, and Richard Hawkins. Confidence Arguments for Evidence of Performance in Machine Learning for Highly Automated Driving Functions. In *Computer Safety, Reliability, and Security*, LNCS, pages 365–377, Cham, 2019. Springer International Publishing.

[Cheng *et al.*, 2017] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote Sensing Image Scene Classification: Benchmark and State of the Art. *Proc. IEEE*, 105(10):1865–1883, October 2017.

[Cheng *et al.*, 2019] Chih-Hong Cheng, Chung-Hao Huang, Thomas Brunner, and Vahid Hashemi. Towards Safety Verification of Direct Perception Neural Networks. *ArXiv190404706 Cs*, November 2019.

[DeVries and Taylor, 2018] Terrance DeVries and Graham W. Taylor. Learning Confidence for Out-of-Distribution Detection in Neural Networks. *ArXiv180204865 Cs Stat*, February 2018.

[Gal and Ghahramani, 2016] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proc. ICML 2016*, volume 48, pages 1050–1059. PMLR, June 2016.

[Guo *et al.*, 2017] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On Calibration of Modern Neural Networks. In *Proc. ICML 2017*, pages 1321–1330. JMLR.org, August 2017.

[Hendrycks and Gimpel, 2018] Dan Hendrycks and Kevin Gimpel. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. In *Proc. ICML 2017*. JMLR.org, October 2018.

[Hendrycks *et al.*, 2019] Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using Self-Supervised Learning Can Improve Model Robustness and Uncertainty. In *Advances in Neural Information Processing Systems 32*, pages 15663–15674. Curran Associates, Inc., October 2019.

[Henne *et al.*, 2020] Maximilian Henne, Adrian Schwaiger, Karsten Roscher, and Gereon Weiss. Benchmarking Uncertainty Estimation Methods for Deep Learning With Safety-Related Metrics. In *Proc. SafeAI@AAAI 2020*, volume 2560 of *CEUR Workshop Proceedings*, pages 83–90, 2020.

[Iandola *et al.*, 2016] Forrest N. Iandola, Matthew W. Moskewicz, Khalid Ashraf, Song Han, William J. Dally, and Kurt Keutzer. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <1MB model size. *CoRR*, abs/1602.07360, 2016. _eprint: 1602.07360.

[Katz *et al.*, 2017] Guy Katz, Clark Barrett, David L. Dill, Kyle Julian, and Mykel J. Kochenderfer. Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks. In *Computer Aided Verification*, LNCS, pages 97–117, Cham, 2017. Springer International Publishing.

[Kim *et al.*, 2018] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV). In *Proc. ICML 2018*, pages 2668–2677, July 2018.

[Lakshminarayanan *et al.*, 2017] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems 30*, pages 6402–6413. Curran Associates, Inc., 2017.

[Lee *et al.*, 2018] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A Simple Unified Framework for Detecting Out-of-Distribution Samples and Adversarial Attacks. *ArXiv180703888 Cs Stat*, October 2018.

[Liang *et al.*, 2018] Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks. In *arXiv:1706.02690 [Cs, Stat]*, February 2018.

[Masana *et al.*, 2018] Marc Masana, Idoia Ruiz, Joan Serrat, Joost van de Weijer, and Antonio M. Lopez. Metric Learn-

ing for Novelty and Anomaly Detection. In *Proc. BMVC 2018*, August 2018.

[Mohseni *et al.*, 2020] Sina Mohseni, Mandar Pitale, JBS Yadawa, and Zhangyang Wang. Self-Supervised Learning for Generalizable Out-of-Distribution Detection. In *Proc. AAAI 2020*, page 8, 2020.

[Olah *et al.*, 2020] Chris Olah, Nick Cammarata, Ludwig Schubert, Gabriel Goh, Michael Petrov, and Shan Carter. Zoom In: An Introduction to Circuits. *Distill*, 5(3):10.23915/distill.00024.001, March 2020.

[Osawa *et al.*, 2019] Kazuki Osawa, Siddharth Swaroop, Mohammad Emtiyaz E Khan, Anirudh Jain, Runa Eschenhagen, Richard E Turner, and Rio Yokota. Practical Deep Learning with Bayesian Principles. In *Advances in Neural Information Processing Systems 32*, pages 4287–4299. Curran Associates, Inc., 2019.

[Pawlowski *et al.*, 2017] Nick Pawlowski, Miguel Jaques, and Ben Glocker. Efficient variational Bayesian neural network ensembles for outlier detection. In *Proc. ICLR 2017*. OpenReview.net, 2017.

[Ren *et al.*, 2019] Jie Ren, Peter J. Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. Likelihood ratios for out-of-distribution detection. In *Advances in Neural Information Processing Systems 32*, pages 14707–14718. Curran Associates, Inc., 2019.

[Salay *et al.*, 2020] Rick Salay, Krzysztof Czarnecki, Maria Soledad Elli, Ignacio J. Alvarez, Sean Sedwards, and Jack Weast. PURSS: Towards Perceptual Uncertainty Aware Responsibility Sensitive Safety with ML. In *Proc. SafeAI@AAAI 2020*, volume 2560 of *CEUR Workshop Proceedings*, pages 91–95, 2020.

[Sensoy *et al.*, 2018] Murat Sensoy, Lance Kaplan, and Melih Kandemir. Evidential Deep Learning to Quantify Classification Uncertainty. In *Advances in Neural Information Processing Systems 31*, pages 3179–3189. Curran Associates, Inc., 2018.

[Serrà *et al.*, 2019] Joan Serrà, David Álvarez, Vicenç Gómez, Olga Slizovskaia, José F. Núñez, and Jordi Luque. Input Complexity and Out-of-distribution Detection with Likelihood-based Generative Models. In *Proc. ICLR 2020*, September 2019.

[Shalev-Shwartz *et al.*, 2018] Shai Shalev-Shwartz, Shaked Shammah, and Amnon Shashua. On a Formal Model of Safe and Scalable Self-driving Cars. *ArXiv170806374 Cs Stat*, October 2018.

[Simonyan and Zisserman, 2015] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *Proc. ICLR 2015*, 2015.

[Stallkamp *et al.*, 2011] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The German Traffic Sign Recognition Benchmark: A multi-class classification competition. In *The 2011 International Joint Conference on Neural Networks*, pages 1453–1460, July 2011.

[Tan and Le, 2019] Mingxing Tan and Quoc Le. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In *Proc. ICML 2019*, volume 97 of *Proceedings of Machine Learning Research*, pages 6105–6114. PMLR, June 2019.

[Timofte *et al.*, 2014] Radu Timofte, Karel Zimmermann, and Luc Van Gool. Multi-view traffic sign detection, recognition, and 3D localisation. *Machine Vision and Applications*, 25(3):633–647, April 2014.

[Weiss *et al.*, 2018] Gereon Weiss, Philipp Schleiss, Daniel Schneider, and Mario Trapp. Towards integrating undependable self-adaptive systems in safety-critical environments. In *Proc. SEAMS 2018*, pages 26–32. ACM, May 2018.

[Willers *et al.*, 2020] Oliver Willers, Sebastian Sudholt, Shervin Raafatnia, and Stephanie Abrecht. Safety Concerns and Mitigation Approaches Regarding the Use of Deep Learning in Safety-Critical Perception Tasks. *ArXiv200108001 Cs Stat*, January 2020.