Uncertainty Quantification and Deep Ensembles

(Seminar: Uncertainty Quantification in Machine Learning)

Gautham Prasad Kundapura
Technische Universität Dortmund
Dortmund, Germany
gauthamprasad.kundapura@tu-dortmund.de

ABSTRACT

A seminar report summarizing the work done on the topic "Uncertainty Quantification and Deep Ensembles" [4] as part of the MSc in Data Science study curriculum at TU Dortmund. This research work was published by the original authors of the paper. The research focuses on calibration issues in deep learning methods, particularly deep ensembles, and a proposed solution to resolve the issue. This report summarizes the proposed solution, methods utilized in the solution, comparison of the methods utilized with other known methods, and possible improvements to the approach presented in the paper.

KEYWORDS

deep ensembles, mixup data augmentation, temperature scaling, pool-then-calibrate

1 INTRODUCTION

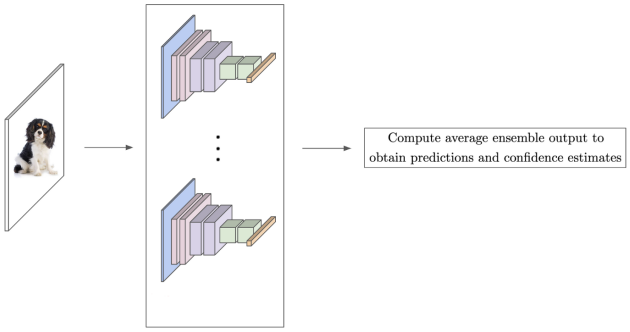

Figure 1: Deep ensemble learning where the data input is very less (Image source: [7])

The amount of data fed into a deep learning algorithm determines the quality of the output. When the data supply is limited (for reference Figure 1), deep learning algorithms are prone to calibration issues. Even though the network model appears to be extremely accurate, it is only addressing a subset of the possible outcomes, and hence the model fails to address uncertainties in the unknown data. This is mostly due to the trained deep model's overfitting, which is caused by overconfidence. A common deep ensemble network can be utilized to alleviate the issue, however, this is most probable in the event of enormous data, making the calibration issue less significant. However, deep ensemble implementation is computationally expensive when the dataset is huge.

The deep ensemble (for reference Figure 2) is a simple approach that performs well with any quantity of data, although the calibration issue exists when the data is small. When paired with two additional strategies, data augmentation and post-processing calibration, it reduces overconfidence and thereby overcomes the calibration issue. The authors of the study emphasize the sequence in which the aggregation of estimations and calibrations are performed, which is referred to as "Pool-then-Calibrate."

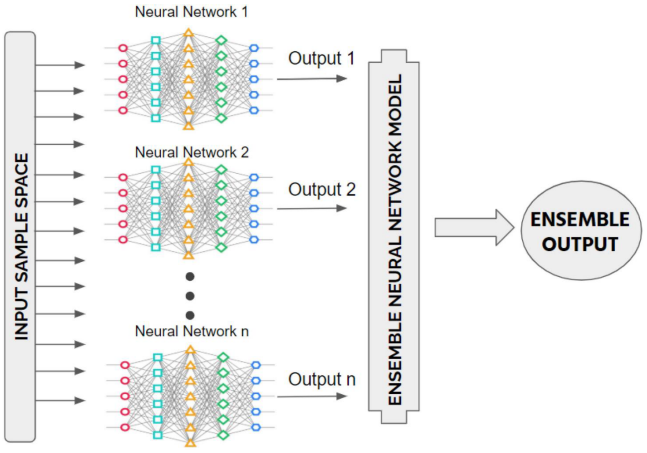

Figure 2: Deep Ensembles model (Image source: [2])

1.1 Data and Setup

Multiple datasets were employed in the study, each having images of different classes since the aim is image classification. Two datasets were primarily utilized for estimate verification and calibration performance. The first is the CIFAR-10 dataset [3], which comprises 60000 32x32 color images of ten different classes, as seen in the Figure 3. Only 1000 images were collected to train the models (also containing validation images). The second dataset is CIFAR-100, which contains images comparable to CIFAR-10 but with images belonging to 100 classes [3]. There are 5000 photos in the training set (along with validation images). RefNet18 is the neural network architecture used to train these datasets.

Imagenette and Imagewoof using ResNet34 network architecture with 5000 training images were utilized to compare the estimated outcomes. MNIST and Diabetic Retinopathy datasets were also utilized for comparisons. Since the goal is to enhance calibration

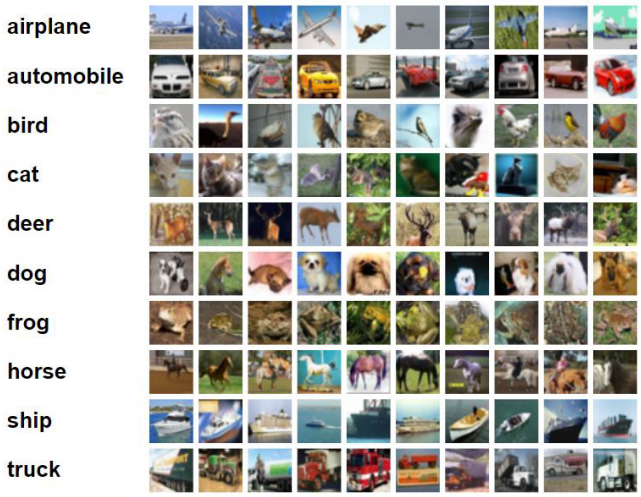

Figure 3: CIFAR-10 dataset having images of 10 different classes (Image source: [3])

in the low-data environment, training data utilized is quite limited. The deep ensembles is implemented with $K = 30$ network models.

2 METHODS IN THE PROPOSED SOLUTION

According to the study, deep ensembles, in combination with the other two approaches employed at the network's input and output sides, have a significant impact on the accuracy and calibration ability of the network model. As shown in the Figure 6, the two approaches employed are data augmentation using the Mixup Augmentation methodology and model calibration with Temperature scaling as a post-processing method. Expected calibration error (ECE) is used to check the extent of calibration in the model.

2.1 Data Augmentation: Mixup

When there is a scarcity of data, some strategies may be utilized to improve the quantity of data available for training the model. One such way is data augmentation. Mixup is a data augmentation strategy that takes convex combinations of training images with randomly chosen weights, and those weights are taken from the beta distribution with identical values for both parameters (Beta(α, α))[7]. This process is shown in the Figure 4.

The formulas depicted in the illustration are as follows:

The weight, λ, is taken from the beta distribution for which the hyperparameter, α, is given by the experimenter. The image x_{mixup} is the result of the convex combination of two images x_1 and x_2 with weights. After combining the labels of images x_1 and x_2, the augmented label for the new image is y_{mixup}. The amount of mixup data augmentation will result in an increase in entropy, decrease in over-confidence of the model, also the decrease in the value of negative log-likelihood and higher model accuracy [4] (discussed in the next sections).

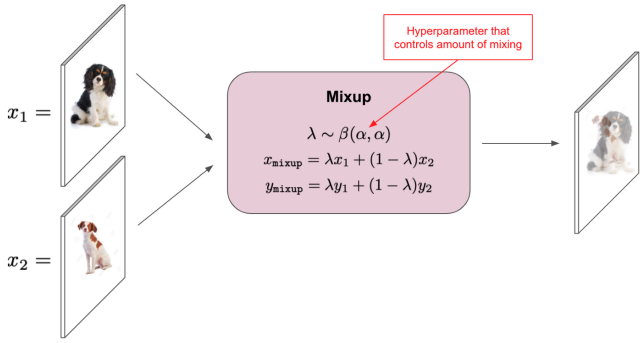

Figure 4: Mixup data augmentation with hyperparameter α from the beta distribution (Image source: [7])

2.2 Post-processing Calibration: Temperature Scaling

In the post-processing phases, the calibration methods are used to recalibrate the model's naive probabilities in order to provide a predicted confidence score $p(x)$. Temperature scaling is one of the calibration methods and is a single parameter *PlattScaling* approach [1].

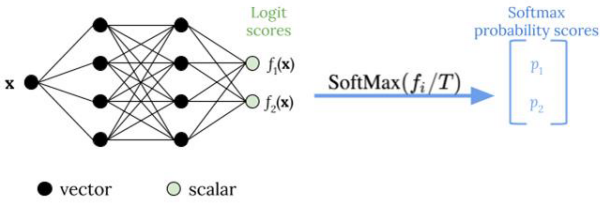

Figure 5: Temperature scaling using the optimized parameter τ (Image source: [1])

It converts the probabilistic outputs, $f(x)$, from model averaging into temperature scaled outputs, $p(x)$, determined by the scaling function, as illustrated in the equation below.

$$Scale(f, \tau) = \sigma_{SM}(log\, f/\tau) = \frac{1}{Z}(p_1^{1/\tau}, ..., p_C^{1/\tau}) \in \Delta_C,$$

where σ_{SM} is the softmax function, $Z > 0$ is the normalization scaler, τ is the optimal parameter found by minimizing the negative log-likelihood score on the validation set and Δ_C is the probabilistic predictions for all the classes C. The temperature paremeter $\tau > 0$ is the only parameter that is optimized here [4]. The Figure 5 explains the temperature scaling.

2.3 Calibration Metric: Expected Calibration Error (ECE)

The difference between prediction confidence and empirical accuracy is measured by the Expected Calibration Error (ECE). It is used to assess if the predicted probabilities of the model are closer to actual probabilities; if so, the network model is well calibrated.

The ECE is calculated by calculating the difference between the confidence and accuracy for a set of bins, and it is equated as,

$$\text{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{N} |conf_m - acc_m|,$$

where

$$acc_m = \frac{1}{|B_m|} \sum_{i \in B_m} 1(\hat{y}(x_i) = y_i) \text{ and } conf_m = \frac{1}{|B_m|} \sum_{i \in B_m} \hat{p}(x_i)$$

Here, B_m is the mth bin for all $m = 1, ..., M$, N is the number of data, $\hat{y}(x_i)$ is the estimation for the data x_i and $\hat{p}(x_i)$ is the predicted probability for the data x_i where $i = 1, ..., N$.

For every bin m, if $acc_m \approx conf_m$, then the model can be considered to be well calibrated and the ECE value would be as less as possible, aiming to reach the value zero or ideally should ECE become zero. Reliability curves can display the curve with $conf_m$ on the x-axis and $(acc_m - conf_m)$ on the y-axis. An under-confident model's curve lies below the line $acc_m - conf_m = 0$ [4].

3 OBSERVATIONS ON PROPOSED SOLUTION

3.1 Proposed Solution

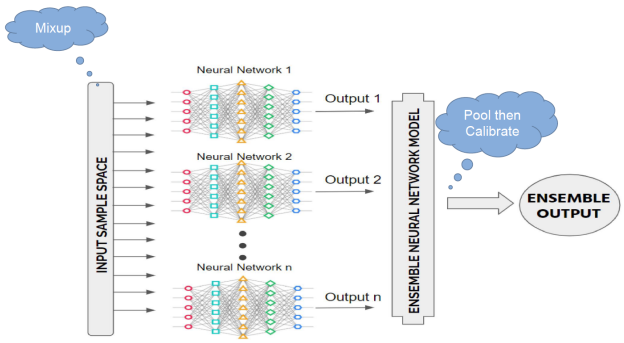

Figure 6: Idea behind the proposed solution with Mixup and Temperature scaling (used in pool-then-calibrate methodology) (Primary Image source: [2])

The proposed solution can be visualized as shown in the Figure 6. As the data supplied is less, the input sample space undergoes mixup data augmentation, and the output is calibrated by temperature scaling performed in the sequence described over the "pool-then-calibrate" technique. The performance of the proposed solution can be measured by comparing the outputs of the model in the solution with the outputs of a simple deep ensemble model for the same data input. Comparisons can also be done to identify the effects and advantages of Mixup augmentation and Temperature scaling.

3.2 Linear Averaging (Deep Ensembles)

Deep ensembles are intended to give more precise and calibrated estimates. To train neural network models with $K = 30$ separate network models, three distinct datasets were employed. Forecasts are also aggregated by averaging probabilistic estimations from all 30 models. The generated reliability curves are depicted in Figure 7. The curves for the network trained on the CIFAR 10 dataset indicate that both individual and averaged predictions are under-confident, particularly aggregated forecasts, which might be useful when each individual network produces over-confident findings. Individual networks are overconfident in the CIFAR 100 dataset but near-calibrated in the Imagewoof dataset. The aggregated predictions

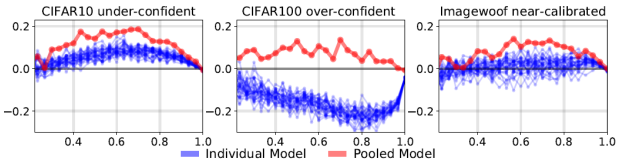

Figure 7: Reliability curve to identify the confidence of the deep ensembles model for different datasets (Iamge source: [4])

that are under-confident can be used to calibrate a deep ensembles model. As a result, deep ensembles are a better tool for calibration.

3.3 Deep Ensembles against BNN Methods

When predictions are aggregated, certain Bayesian Neural Network models exhibit characteristics comparable to deep ensembles, according to the research. The under-confidence of aggregated predictions persists in BNN models as well. In Table 1, the resultant ECE scores for ensembled networks for the SWAG and MC-Dropout BNN models are greater for both the CIFAR 10 and CIFAR 100 training datasets. These models are computationally more expensive than the non-BNN deep ensembles.

Dataset	Method	Single models	Ensemble
CIFAR 10	SWAG	3.17±.27	4.36
	MC-Dropout	6.55±.10	7.59
CIFAR 100	SWAG	3.34±.14	5.49
	MC-Dropout	4.92±0.19	9.05

Table 1: ECE scores of BNN methods for CIFAR 10 and CIFAR100 datasets: under-confident models [4]

The BNN ensemble models are less calibrated than their individual models. Deep ensembles is much easier to implement and calibrating under-confident predictions is still feasible.

3.4 Effects of Mixup Augmentation and Temperature Scaling

The CIFAR 10 Mixup augmented dataset is used for training deep ensembles with varying regularization $\alpha = \{0.2, 0.5, 0.8, 1.0\}$. The Figure 8 depicts several key metrics for unscaled (no post-processing) and temperature scaled ensemble models. The unscaled model exhibits an increase in entropy as the distance between the validation and training images grows, indicating a reduction in overconfidence. Despite the increasing distance between validation and training images, the model trained on no mixup dataset delivers comparably overconfident results. The increase in distance is denoted in quantiles on the x-axis.

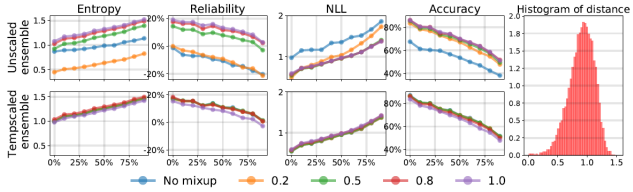

Figure 8: Comparing the effects of mixup and temperature scaling on the basis of different matrics (Iamge source: [4])

With the exception of the mixup augmented model with α=0.2, all of the other models exhibit better reliability, a decrease in negative log-likelihood scores, and only a slight decrease in the accuracy as the distance rises. When the models with varied α values are temperature scaled, their entropies agree, as do their reliability, NLL, and accuracy. This is because the training dataset was augmented using the mixup approach.

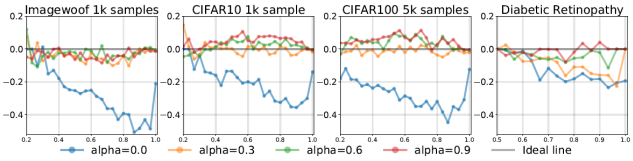

Figure 9: Effects of mixup and temperature scaling on training different datasets (Iamge source: [4])

The same combination of mixup augmented data and temperature scaling is used in conjunction with deep ensembles to train with different datasets. The Figure 9 shows the resulting reliability curves for models with different α values. The predictions for the ensemble models with no mixup (α=0) are overconfident, while other models with α=0.3, 0.6, and 0.9 show better calibration of predicted estimations for all datasets. As a result, using mixup augmentation and temperature scaling in combination with deep ensembles result in better-calibrated forecasts.

4 RESOLVING THE CALIBRATION ISSUE

Temperature scaling as a post-processing technique aids in the generation of better-calibrated models. According to the study, the order in which aggregation and scaling are applied has a major impact on model calibration. Various approaches were used, their results were compared, and eventually which methodology was best suited is summarized in the following subsections.

4.1 Calibration Methodologies

There are 4 methodologies introduced which explain the sequence in which the aggregation and scaling are done. They are as follows:

A. Averaging the estimations from all the models hoping to obtain a better-calibrated model. No temperature scaling is to be applied.
B. Each individual network will be calibrated before aggregating the estimations. Each individual model is temperature scaled separately, then the estimates are pooled.
C. Aggregating and calibrating the estimations simultaneously for each individual model. Each model is scaled with a common temperature parameter and simultaneously they are pooled.
D. Aggregating the estimates of each individual model before pooled estimates are calibrated. First, the estimates from the models are aggregated by pooling them, then calibrated by the application of temperature scaling (Pool-then-calibrate).

The authors recommend the pool-then-calibrate strategy over other methods. This can be substantiated by comparisons and examination of its impact on various data regimes.

4.2 Comparison of the Methodologies

For different datasets, all four approaches were implemented and investigated. The findings are shown in Figure 10, which shows the values obtained for three performance metrics: ECE, NLL, and Brier. The ECE values, after using strategies C and D, as seen in the first row, are almost the same, even for different pooling methods such as linear pooling, median pooling, and trimmed pooling. In terms of all metrics utilized to measure the performance, strategies A and B yield fairly unconvincing outcomes. So, C and D are deemed preferable strategies to calibrate ensemble models since their results show lower ECE, NLL, and Brier scores throughout.

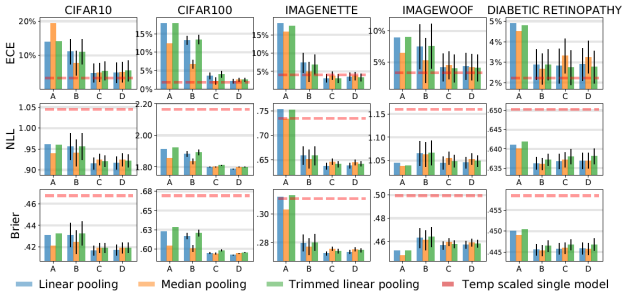

Figure 10: Comparing different calibration methodologies on the basis of diffrent scoring metrics (Iamge source: [4])

The red lines represent the results of single models trained using mixup augmentation at α=1. These red lines serve as the baselines for all of the techniques. As shown in the graphs, linear pooling as the pooling method in all strategies has an effect on models trained with different training datasets. Because linear pooling produces lower ECE, NLL, and Brier scores, it was chosen as the pooling strategy in the suggested solution. Also, among strategies C and D, strategy D, "pool-then-calibrate", appears to be the simplest to execute, and because C and D perform practically identically, D was chosen to be used in the suggested solution.

4.3 "Why pool-then-calibrate?" and Benefits of Mixup Augmentation

As previously stated, pool-then-calibrate is the preferred method. It is used in conjunction with the mixup-augmentation for different α values and the results are given in terms of performance metrics in the Figure 11. The deep ensembles of K=30 networks is trained on N=1000 CIFAR 10 training data with varying amounts of mixup-augmentation, with N_{val}=50 validation samples included.

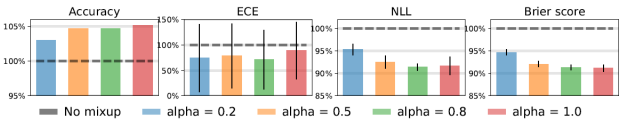

Figure 11: Graphs to compare the scores: Before and after applying mixup augmentation (Iamge source: [4])

The dotted black line represents the baseline performance of a model trained without mixup-augmentation on the same CIFAR 10 mixup-augmented dataset. The accuracy graph demonstrates that using mixup augmented data with varying alpha values increases model accuracy by nearly 5%. Furthermore, ECE appears to have been decreased by 25% for models trained on mixup data. NLL appears to be lowered by around 8-9 percent. Brier score is also dropped by 8-9 percent.

These characteristics imply that the use of mixup augmentation, in conjunction with the pool-then-calibrate technique, yields considerable success in calibrating the deep ensembles model.

4.4 Effects of Pool-then-Calibrate

In terms of low-data regime, the pool-then-calibrate technique on the deep ensembles model trained using the mixup-augmented dataset shown to be advantageous. It is appropriate to use the same approach in a high-data regime and compare it with other known methods. The Table 2 displays the performance metrics of the model based on the proposed solution, as well as additional models such as scaled individual models, unscaled individual models, and ensembled scaled and unscaled models.

Method	Accuracy	ECE	NLL	Brier
(1) Individual models (unscaled)	70.8±.36	9.8±.31	1.17±.01	0.411
Ensemble of models in (1)	78.4	5.9	0.782	0.308
(2) Individual models (scaled)	70.8±.36	2.1±.4	1.07±.01	0.396
Ensemble of models in (2)	78.4	13.2	0.859	0.331
Pool-then-calibrate	78.4	3.4	0.770	0.303

Table 2: Matrics to measure the performance of different models trained with large CIFAR 100 50k data: Pool-then-calibrate is a better method [4]

The results in the table are from experiments performed on the CIFAR 100 complete dataset. The accuracies of unscaled individual model ensembles, scaled individual model ensembles, and deep ensembles with pool-then-calibrate are identical, thus applying this idea will not help in judging the success of the suggested pool-then-calibrate technique. As a result, ECE, NLL, and Brier scores better reflect the performance of each method. The accuracy of the ensemble of individual unscaled and scaled models is good, but the ECE value of the unscaled ensembles is poorer due to high under-confidence. When compared to other methods, the suggested pool-then-calibrate method has the lowest ECE.

Even the suggested method's NLL (negative log-likelihood) score is the lowest among the scores of other methods. Finally, the Brier score agrees with the suggested method's performance in the high-data setting by having the lowest value of all the remaining methods. As a result, the suggested pool-then-calibrate strategy has a considerable influence in high data regimes.

5 ADVANTAGES AND DRAWBACKS IN THE SOLUTION

5.1 Advantages of the Proposed Solution

The proposed solution comprises mixup data augmentation and temperature scaling in combination with deep ensembles. The solution has some advantages:

(1) All of the methods used in the solution are straightforward, making them simple to implement.
(2) The suggested model's probabilistic predictions are likely to be well-calibrated, as the model's overconfidence is potentially rectifiable.
(3) The methods used in combination with deep ensembles are simplistic, and they all perform well in the low-data realm, where the calibration problem is inevitable.

5.2 Drawbacks and Possible Fixes

The model in the solution is simple to implement, but it has its own set of limitations. Some of the remedies that can be considered to potentially correct these limitations are as follows:

- Deep ensembles are simple to implement in general, but they are computationally expensive, which might have an impact on runtime performance when the input data is enormous. Deep sub-ensembles are less expensive, and their influence may be explored by using them instead of deep ensembles, although at the risk of a minor increase in error [6].
- Image data augmentation with Mixup may result in overlapping of objects from distinct pairs of images, making determining object boundaries challenging. This problem is likely to degrade performance. In this instance, background mixup data augmentation can be employed as an alternative. Background mixup augmentation combines training and background images to increase generalization ability and, as a result, overall performance [5].

6 SUMMARY

The study focuses on the calibration challenges of deep learning algorithms trained in a low-data regime. The proposed method addresses these challenges while also assisting deep ensembles in calibrating their probabilistic predictions. The researchers investigated the interplay of three of the most basic strategies for implementing deep learning in the low-data domain. Deep ensembles, mixup data augmentation, and temperature scaling are the employed methods in the solution.

The research also demonstrated how deep ensembles alone cannot fix calibration issues since they provide overconfident forecasts.

As a result, it was demonstrated that employing mixup data augmentation reduces overconfidence, after which the probabilistic estimates can be pushed through a calibration process using temperature scaling to obtain a better-calibrated model. After comparing its results and performance to other known methods, this combination of methods was found to be effective. The impacts of the methods used were further investigated by comparing the performance of the models with and without them. The influence of the hyperparameters utilized in the methods in terms of the model's performance was investigated by comparing the performance by varying the values of these parameters. These parameters may be chosen by analyzing the model's performance as measured by metrics such as Expected Calibration Error (ECE), Negative log-likelihood (NLL), and Brier scores. ECE was primarily used in the research work to evaluate the model's calibration abilities.

Various approaches can be used to calibrate the estimations. The researchers determined which among the four approaches was the best by analyzing the performance metrics of the results acquired after calibrating the estimations using all those methodologies. The authors chose the pool-then-calibrate strategy because of its simplicity and efficacy in both low-data and high-data regimes. This method combines the pooled predictions from individual neural networks before post-processing the results with the resilient temperature scaling method.

The proposed solution has a few advantages, including ease of implementation and better-calibrated models. In terms of run-time efficiency, there are a few constraints. Although deep ensembles are a simple method to implement, they are computationally expensive. If the primary goal of an experimenter is to make the model computationally as less expensive as possible with tolerance to a minor increase in error, deep sub-ensembles [6] might be proposed in place of deep ensembles. Mixup data augmentation may result in overlapped images, making it difficult for the model to recognize object boundaries, and thus lowering the performance. In such a case, a recently discovered augmentation approach called Background Image Augmentation [5] may be useful. This method combines the item from the training image with the background from another image, resulting in objects that are easily recognized by the models and improved performance.

Finally, AI-powered applications like these are susceptible to approval based on dependability and trustworthiness. Simple implementation may not imply superior results. Other difficult approaches can be utilized to develop a far more efficient model if the goal is to get accurate, dependable, robust, and trustworthy outcomes where processing power is not a concern.

REFERENCES

[1] AWS. 2022. *Temperature scaling.* Retrieved July 31, 2022 from https://docs.aws.amazon.com/prescriptive-guidance/latest/ml-quantifying-uncertainty/temp-scaling.html

[2] Neha Gianchandani, Aayush Jaiswal, Dilbag Singh, Vijay Kumar, and Manjit Kaur. 2020. Rapid COVID-19 diagnosis using ensemble deep transfer learning models from chest radiographic images. *J Ambient Intell Human Comput* (2020). Retrieved July 31, 2022 from https://doi.org/10.1007/s12652-020-02669-6

[3] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. 2009. *The CIFAR-10 dataset.* Retrieved July 31, 2022 from https://www.cs.toronto.edu/~kriz/cifar.html

[4] Rahul Rahaman and Alexandre H. Thiery. 2021. Uncertainty Quantification and Deep Ensembles. In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (Eds.), Vol. 34. Curran Associates, Inc., 20063–20075. https://proceedings.neurips.cc/paper/2021/file/a70dc40477bc2adceef4d2c90f47eb82-Paper.pdf

[5] Koya Tango, Takehiko Ohkawa, Ryosuke Furuta, and Yoichi Sato. 2022. Background Mixup Data Augmentation for Hand and Object-in-Contact Detection. (2022). arXiv:2202.13941v2 https://arxiv.org/abs/2202.13941

[6] Matias Valdenegro-Toro. 2019. Deep Sub-Ensembles for Fast Uncertainty Estimation in Image Classification. *CoRR* abs/1910.08168 (2019). arXiv:1910.08168 http://arxiv.org/abs/1910.08168

[7] Cameron Wolfe. 2022. *Confidence Calibration for Deep Networks Why and How?* Retrieved July 31, 2022 from https://towardsdatascience.com/confidence-calibration-for-deep-networks-why-and-how-e2cd4fe4a086