CNN with Uncertainty Quantification

Turion Space - Take Home Assessment

By: Amogh Palasamudram

Overview

This report proposes and experiments with a novel uncertainty quantification method – Rejection Confidence Variance. Rather than performing a basic experiment with a given uncertainty quantification method, this report compares this new method to standard uncertainty measures such as Dropout Ensemble. All hyperparameters, datasets, etc. have been standardized for these two methods, so that this report can highlight the effectiveness and applications of these uncertainty quantifications methods. In this report, the performance and accuracy are not much of a concern compared to the uncertainty and how accurately it has been quantified. Hence, the models need not be extensively trained and optimized.

Introduction

Uncertainty in a neural network relates to the model's confidence that it accurately predicted its given inputs. This is especially important in applications of high stakes, such as image classification for detecting debris and satellites in Low Earth Orbit (LEO).

There are two types of uncertainties in deep learning models – epistemic uncertainty and aleatoric uncertainty. Epistemic uncertainty refers to uncertainties in the model's prediction due to the imperfect nature of the model's parameters while aleatoric uncertainty refers to uncertainty due to inherent randomness in the data itself. This report focuses on epistemic uncertainty and how it can be quantified.

Epistemic uncertainty can be reduced by improving network architectures or increasing the dataset size. There are various methods to quantify epistemic uncertainty. Most methods either calculate the variance of confidence levels predicted by different models and others use a form of distribution and dropout to determine the uncertainty.

This report proposes a novel method of quantifying the epistemic uncertainty of deep learning models, called the **Rejection Confidence Variance (RCV)**. This new method takes inspiration from a specific neural network, the Neural Layer Bypassing Network (NLBN)[1]. This network was designed and created by Amogh Palasamdram – the author of this report. The NLBN uses rejection layers after each layer of the main network to determine whether the semi-processed inputs must be propagated through the rest of the model for an accurate prediction. This increases the efficiency of the neural networks by passing inputs only through the minimum number of layers required for accurate predictions.

The concept of the NLBN can be utilized for uncertainty quantification by comparing the confidence levels of the rejection layers to the confidence of the completely processed output of the model. The uncertainty of the model's predictions is determined as the variance between the rejection layer confidences and the model output confidences. This will be calculated similar to a loss function rather than variance. This way, if the confidences of the rejection layers are similar to those of the model's outputs, then there is more consistency in the predictions and the model's uncertainty is low. On the other hand, if the rejection layer and model output confidences are very different, the uncertainty is high as the model is not confident that its outputs are correct.

As a result, we can use the features of the NLBN to determine epistemic uncertainty of a model. By using this uncertainty quantification method, machine learning developers can also understand the effectiveness of the model at different layers, providing a more comprehensive and granular dive into the performance of the network. This can significantly benefit machine learning engineers when it comes to quickly testing and designing networks for applications that require high performance, low uncertainty, and speed.

---

[1] https://www.techrxiv.org/articles/preprint/Neural_Layer_Bypassing_Network/16806928

This is the proposed Rejection Confidence Variance uncertainty quantification method from this report, and it will be compared to the dropout ensemble technique which can be used to determine epistemic uncertainty in deep learning models.

Note: The Rejection Confidence Variance is not exactly variance. The calculation is more similar to a loss function or T-Test. It aims to highlight how similar or different the confidence levels are for the rejection layers and output layer. This allows the developers to know how effective certain layers are.

Objectives

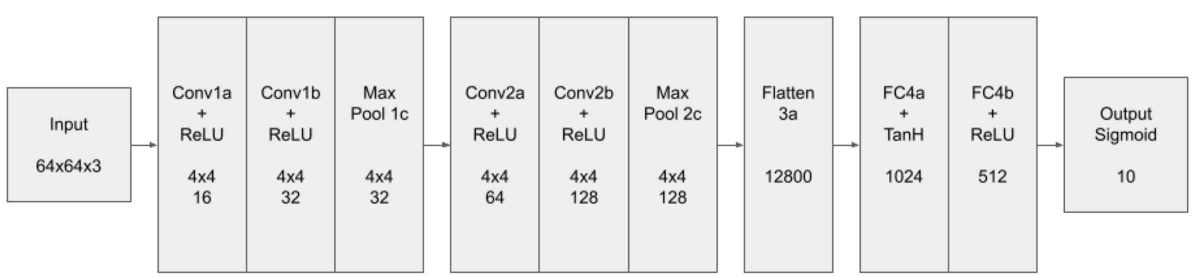
Aim: Understand the effectiveness of Rejection Confidence Variance (the proposed method to quantify the uncertainty of computer vision models).

This report will compare the efficacy of this new method with the concept of dropout as an ensemble technique for uncertainty quantification.

Experiment

Setup

Model - Convolutional Neural Network for Image Classification:



*Dropout after each Conv and FC layer (dropout probability of 0.1)

Effects of Non-Linearity:

1. By having convolutions, max pooling layers, and activation functions like ReLU and Tanh add depth to the model. These layers and activation functions allow for complex patterns to be found. A linear transformation of a linear transformation results in a linear transformation. Hence, if there was no non-linearity, the maximum depth of the network would be one fully connected layer; this cannot be optimized for the data and would result in underfitting.

Loss Function:

1. Cross Entropy Loss
    a. This model attempts to perform image classification so it must utilize a classification loss.
    b. Cross Entropy Loss was chosen because it is meant for classification and encourages the expected label to be predicted with a high confidence while also encouraging the other labels to have as little confidence as possible. Additionally, this loss is considered efficient because it provides smooth gradients for optimization.

Framework:

1. PyTorch

Datasets:

1. EuroSAT
    a. The EuroSAT dataset was chosen because it consisted of real, low-resolution images that could simulate real-world applications that would require models to be accurate, fast, and confident in their predictions.
    b. The dataset consists of 27,000 images and is shuffled before the split into training and testing images. This ensures the data is arbitrary and minimal aleatoric uncertainty.
    c. More datasets that conform to similar metadata attributes as the EuroSAT dataset can be used

Optimizer:

1. Adam: (learning rate = 0.001, weight_decay = 0.001)
    a. Adam is one of the popular optimizers and is commonly coupled with cross entropy loss for image classification. This is because it uses adaptive learning rates to efficiently optimize its steps to the global minima.

Validation of Bounds:

1. The Sigmoid function has been used to provide the outputs for each class. This restricts the output values for each label to fall within range of (-1, 1).

Epistemic Uncertainty Quantifier:

1. Proposed Uncertainty Quantifier using the NLBN (Rejection Confidence Variance - RCV)
2. Dropout Ensemble Technique

The Dropout Ensemble Technique was chosen as the base uncertainty quantification method because it is a very popular, well-known, and well-tested method to understand uncertainty. It is also relatively simple to implement and encourages dropout which suppresses overfitting.

Procedure

Steps:

1. Quantifying uncertainty using the Dropout Ensemble Technique:
    a. Create and train a standard model using dropout
    b. Perform forward propagation multiple times on the standard model with dropout on.
    c. Calculate averages and variance of confidence levels of these forward propagations to determine epistemic uncertainty.
2. Quantifying uncertainty using the Rejection Confidence Variance method:
    a. Create and train NLBN rejection layers
    b. Perform forward propagation multiple times on NLBN.
    c. Calculate variance of confidence level of rejection layers and model outputs to determine epistemic uncertainty.

Note:

1. To standardized variance, the last activation function of the model is a sigmoid function.
2. Some functions and variables in the NLBN implementation have been hardcoded to the given model structure for simplicity.
3. This report focuses on the use of uncertainty quantification over actual performance. Hence, the model's hyperparameters have not been extensively tuned with the goal of performance.

Results

After training the image classification model for 25 epochs on the EuroSAT dataset with 20,000 training images and 7,000 testing images, the following results were obtained.

Dropout Ensemble Technique

By using the Dropout Ensemble Technique, the variance of the model predictions for each label can be calculated. Forward propagation was repeated 10 times to get the variance.

Label	0	1	2	3	4	5	6	7	8	9
Variance	0.021	0.036	0.064	0.065	0.008	0.062	0.040	0.006	0.059	0.003

The variance produced by the Dropout Ensemble Technique suggests that uncertainty of the model is quite low (between 0.3% and 6.5%) and that the model is quite confident with its predictions. This result is expected as the EuroSAT dataset is relatively easy to train on for the given architecture and network size.

Hence, to improve the model, the quantified uncertainty suggests the model should be trained on more data, for more epochs, or with different neural network hyperparameters. This will allow the model to understand the data better and hence be more confident in its predictions.

Rejection Confidence Variance (RCV)

By using the proposed Rejection Confidence Variance, the uncertainty was quantified for each rejection layer and for each label in the dataset.

The greater the R Layer Index, the further the down the rejection layer is with respect to the main model.

As seen in the data, the uncertainty is relatively higher in the initial layers as compared to the final layers. This is expected and is justified by the fact that the final layers can process the inputs more and, hence, be more confident in its predictions. The earlier layers are more likely to suffer from underfitting and will hence not be as accurate and certain in its predictions.

Additionally, the predictions are overall quite certain. The variance is relatively less, suggesting that the model's (NLBN's) rejection layers determine obvious outputs effectively with respect to the main model's actual predictions.

All in all, the Rejection Confidence Variance is low for this network as the model structure is large enough to prevent underfitting on this dataset. To further optimize the performance of the model, one must train it on more data, for more epochs, etc.

Note: The rejection layers do not need to be accurate. They simply need to have high confidence levels when they are predicting labels that they can accurately predict. Hence, it can be seen that the overall loss for these layers is generally high compared to the loss of the actual model.

Conclusion

The Dropout Ensemble Technique and the Rejection Confidence Variance are two different uncertainty quantification methods that can effectively assess the performance of the model and how confident it is with its predictions.

The Rejection Confidence Variance, however, provides a wider scope as to what can be analyzed and explored when it comes to improving the model. The Dropout Ensemble is a more standardized uncertainty quantification method that can be used to compare models and given benchmarks.

This report briefly delves into Dropout Ensemble and elaborates on the new, proposed method of quantifying uncertainty – Rejection Confidence Variance. The two methods were only tested on the EuroSAT dataset with the Cross Entropy Loss and Adam Optimizer. More loss functions and optimizers can be used to explore the holistic uncertainty quantification of the Rejection Confidence Variance.

Overall, both methods serve the purpose of calculating the uncertainty in model predictions.

Strengths and Limitations of RCV

Strengths	**Limitations**
Provides a more comprehensive understanding of the model's performance and uncertainties	Not a standardized and widely accepted method of quantifying uncertainty.
Can make hyperparameter tuning and model development, integration, and deployment quicker.	Results may require further analysis to determine next steps for improving and optimizing the model
Has the potential to scale and be optimized for a specific task, allowing developers to create the best models in fewer iterations.	Relatively difficult to implement (requires the NLBN architecture, functionality, dataset classes, etc.)

Applications

The Rejection Confidence Variance method to quantify uncertainty in deep learning models can be utilized in work environments where accurate and quick development of machine learning models is essential. Moreover, this method can be used to get a more granular understanding of the performance of the trained neural network. It can also be integrated with the NLBN architecture, minimizing the speed-accuracy trade-off the model would have otherwise. Finally, this method can be coupled with other, more standard uncertainty quantification methods such as Drop Ensemble, Model Averaging, and Bayesian Neural Networks.

All in all, the Rejection Confidence Variance as proposed in this report can be utilized to get a deeper understanding of one's neural network and can be used alongside other uncertainty quantification methods to point developers in the optimal direction for solutions with better performance and less uncertainty.