# Devanagari Character Recognition and Confidence Calibration

**Naiqing Guan**
Department of Computer Science
University of Toronto
naiqing.guan@mail.utoronto.ca

**Varun Dharmendrakumar Pandya**
Department of Computer Science
University of Toronto
varun.pandya@mail.utoronto.ca

## Abstract

While the Devanagari language is used by more than 500 million people in the world, research on recognizing Devanagari characters is relatively sparse. In this paper, we implement two Convolutional Neural Network(CNN) models for Devanagari character recognition and reach 94% accuracy on the Devanagari Handwritten Character Dataset. We also implement and compare two calibration methods on top of the CNN models. Experiments show that while temperature scaling works well, deep ensemble exacerbates calibration error for our models.

## 1 Introduction

Character recognition is an important task in this digital era, where the need for optical character recognition (OCR) and handwriting recognition is becoming paramount to store and process digital records of text. While a lot of research and literature is available for English character recognition, research for other languages is relatively sparse.One such example is the Devanagari language. Considered to be the origin for many languages of the Indian subcontinent, the Devanagari language is currently being used by more than 500 million people worldwide. Despite the popularity of the Devanagari language, when searching on google scholar, we find the amount of papers on Devanagari character recognition are only 0.3% of the papers on English character recognition, which motivates us to work on the Devanagari character recognition problem.

When performing character recognition, apart from the class prediction, users may want a confidence score denoting the model's estimated accuracy on the prediction. This score can help users find places where the classification is error prone. The confidence score is also useful when the system requires high processing speed, where a less-accurate but fast model can scan over the whole text and an accurate but costly model can recheck the less confident predictions. In practice, people often use the softmax output of neural network as the confidence score, but research has shown that it has high calibration error[4]. To make the confidence score representative of the true correctness likelihood, various confidence calibration methods[4, 6, 13] have been proposed.

The goal of this research is twofold. The first is to design and implement CNN models to make accurate Devanagari character recognition, and the second is to test the performance of various confidence calibration methods on this task. In this paper, we implement two CNN models for Devanagari character recognition, reaching 90% and 94% accuracy on the Devanagari Handwritten Character Dataset separately. We also compare two calibration methods on top of the model with higher accuracy. Surprisingly, deep ensemble method[6] increases the calibration error for our models. To the best of our knowledge, our work is the first that employs confidence calibration on top of the Devanagari character recognition problem and reveals a limitation of deep ensemble method for calibration.

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

# 2 Related work

Before a standardized dataset was proposed, a majority of the research articles focused on implementing different methods on different datasets. Sharma et al.[15] used the Quadratic classifier using the histogram of directional chain code feature extraction method on dataset consisting of about 11,000 images. Arora et al. [2] employed a Neural Network classifier using the structural features of a dataset consisting of about 50,000 images. Pal et al. [11, 10, 12] used various classifiers like the quadratic classifier, Support Vector Machine(SVM) and Mirror Image Learning(MIL) on a dataset consisting of about 36,000 images. Kumar [5] used the Multilayer Perceptron(MLP) and SVM on a dataset of about 25,000 images.

In 2015, Pant et al.[1] introduced a dataset consisting of about 92,000 images and trained a CNN to achieve an accuracy of about 98.2%. This paper standardized the dataset for Devanagari Character Recognition task and subsequently, other authors like Ram et al. [14], Sonawane et al. [16], Mhapsekar et al. [7] used this dataset with a CNN model, the AlexNet and the ResNet respectively and reported an accuracy of about 96.9%, 95.4% an 99.35% respectively.

There are various ways to represent and quantify model calibration. Reliability diagrams[9] represent the model calibration by plotting expected sample accuracy as a function of confidence. When the plot resembles an identity function, the model is well calibrated. Expected Calibration Error (ECE) [8] is a scalar summary statistic of calibration, which is achieved by first partitioning the predictions into equally spaced bins, and then take a weighted average of the differences between accuracy and confidence of each bin. Lower ECE score means better calibrated results.

# 3 Data

The Devanagari Handwritten Character Dataset was first introduced by Pant et al. [1] in 2015 for the Devanagari Character Recognition task. The dataset consists of 92,000 images of handwritten Devanagari characters, out of which 85%(78,200) of the images are used as the training set and 15%(13,800) are used as the testing set. The dataset consists of images belonging to one of the 46 classes(36 base forms of consonants and 10 numerals) with each character having about 2000 images, making the dataset well-balanced. All of the images are in grey scale format and 28X28 in dimension, with padding being used to make the images more centred, and thus increasing the dimension to 32X32. Classification task is challenging since a lot of the different characters are written similarly with minute variations in the strokes of the lines.

We perform pre-processing steps on every image in the dataset to make the classification task more accurate. Figure 1 describes the preprocessing steps. The original 32X32 image is passed through the Contrast Limited Adaptive Histogram Equalization(CLAHE) step which smoothens the overamplification of the contrast in the image, so as to reduce the problem of noise amplification [13]. Further, the filter kernel [[-1,-1,-1], [-1,9,-1], [-1,-1,-1]] is applied on the image to sharpen it and define the contours in the image. Finally, the image is converted into a binary format and this image is then used with the different models, which increases their accuracy by about 0.5% to 1%.

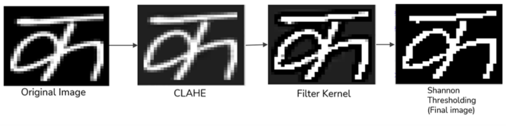

Figure 1: Preprocessing steps.

# 4 Model

We implement two CNN models for character recognition task and illustrate their structures in Figure 2. Model 1, proposed by us, consists of two convolutional layers each followed by a max-pooling layer and a fully connected layer at the end which classifies the image in of the 46 classes. The convolutional layers have a 3*3 kernel size. Model 2, proposed by Pant et al. [1] has a similar structure as that of model 1, but has a larger kernel size (5*5) and more filters.

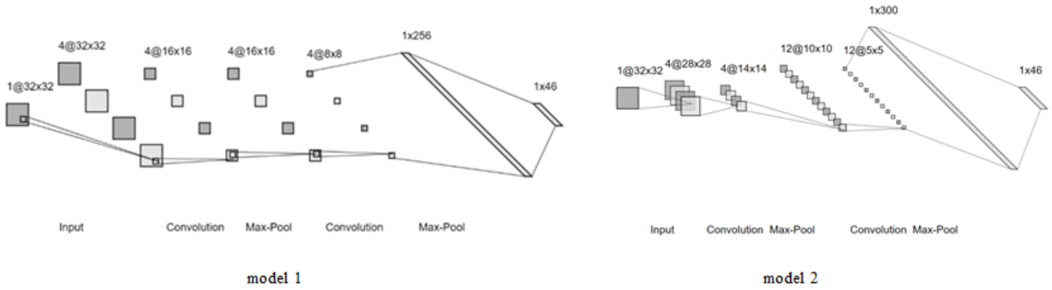

Figure 2: Structure of two CNN models.

The two calibration approaches we investigate in this paper are temperature scaling [4] and deep ensemble [6]. Temperature scaling is an extension of Platt scaling [13], which uses the temperature parameter T to scale the logit vector before it is fed into the softmax layer. To be specific, let $\mathbf{z}$ be the logits vector produced for input $\mathbf{x}$, and $\sigma$ be the softmax function. The confidence prediction after temperature scaling is

$$\hat{q} = \underset{k}{max}\ \sigma(\mathbf{z}/T)^{(k)} \tag{1}$$

$T = 1$ returns the original confidence; $T < 1$ increases the confidence and $T > 1$ decreases the confidence. The prediction class $\hat{y}$ remains unchanged because temperature scaling doesn't change the maximum index of logit vector.

Deep ensemble[6] proposes to use M neural networks trained with random initialized parameters and randomly shuffled data points. After training, suppose the probability estimation of model i is $\hat{\mathbf{p}}_\mathbf{i}$, then the probability estimation of the ensemble is

$$\hat{\mathbf{p}} = \frac{1}{M}\sum_{i=1}^{M}\hat{\mathbf{p}}_\mathbf{i} \tag{2}$$

and the confidence prediction after deep ensemble is

$$\hat{q} = \underset{k}{max}\ \hat{\mathbf{p}}^{(k)} \tag{3}$$

In essence, the method averages the probability estimation of each model and makes final prediction according to the averaged result. Different from temperature scaling, ensemble of models might change the prediction class, and has long been used to improve classification accuracy[3].

## 5 Results

**Comparison of CNN models** We test the performance of model 1 and model 2 on Devanagari character recognition using two metrics: classification accuracy and expected calibration error (ECE). Figure 3 demonstrates the result. Model 2 outperforms model 1 in accuracy, reaching about 94% accuracy after 30 epochs. The calibration error of two models are quite similar. As the number of epochs increases, ECE first drops and then increases for both models.

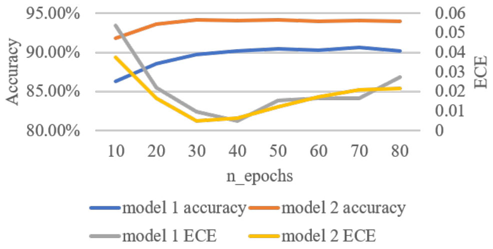

Figure 3: Comparison of two CNN models.

**Comparison of calibration methods** We compare the two calibration methods on model 2 when trained with 10 epochs. Figure 4 shows the reliability diagrams for different calibration methods. For deep ensemble method, we use an ensemble size of 3. Temperature scaling performs well in reducing ECE and has almost no effect on accuracy. The slight improvement in accuracy is due to random initialization of model parameters. Deep ensemble, however, increases the accuracy and ECE at the same time.

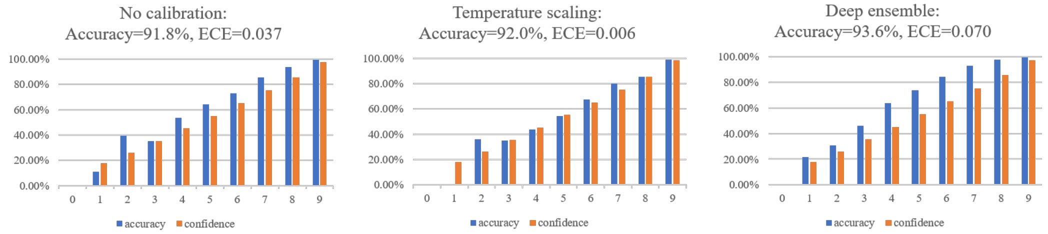

Figure 4: Reliability diagram using different calibration methods.

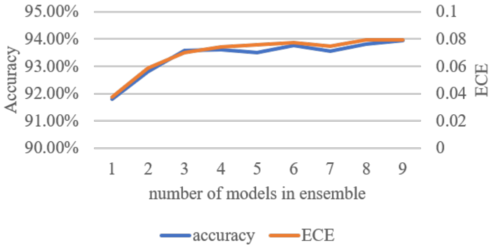

Figure 5: Impact of number of models in ensemble.

**Impact of number of models in ensemble** We test the impact of number of models in ensemble on accuracy and ECE. Figure 5 shows the result for model 2 when trained with 10 epochs. As can be seen from the figure, the trends of accuracy and ECE are fairly similar, which indicates that the calibration performance of deep ensemble is closely related to its ability in improving prediction accuracy.

## 6 Discussion

When comparing the two CNN models, we find model 2 outperforms model 1 in accuracy. This is probably because model 2 has larger kernel size and more filters, making it more effective in capturing local features of the Devanagari characters.

While temperature scaling successfully calibrates the estimated confidence, we find deep ensemble exacerbates miscalibration in our experiments. The most possible reason for that is that while deep neural networks always suffer from making overconfident estimates[4], our model has unconfident estimates (confidence less than accuracy) due to its small size. As we observe in our experiments, the deep ensemble method has a tendency to "push" the estimation to unconfident side, probably because it can improve prediction accuracy and hence reduce the difference between confidence estimation and prediction accuracy. While this effect might calibrate overconfident models, it will exacerbate miscalibration for unconfident or well calibrated models.

There are several limitations of this project. First, we only compared two models performance for Devanagari character recognition, while a systematic investigation using various model structures will be more helpful. Second, although we suspect deep ensemble might help calibrate overconfident models, we lack direct evidence due to the small size of our model. Future works can try more model structures and see their effect on prediction performance and calibration.

## Attributions

This work is conducted by Naiqing Guan and Varun Dharmendrakumar Pandya, both contributing equally to it. Varun mainly works on the devanagari character recognition part. He analyzes and preprocesses the dataset and builds two CNN models for character recognition. Naiqing mainly works on the confidence calibration part. He implements two calibration methods, conducts the experiments and analyzes the results.

## References

[1] Shailesh Acharya, Ashok Kumar Pant, and Prashnna Kumar Gyawali. Deep learning based large scale handwritten devanagari character recognition. In *2015 9th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, pages 1–6. IEEE, 2015.

[2] Sandhya Arora, Debotosh Bhatcharjee, Mita Nasipuri, and Latesh Malik. A two stage classification approach for handwritten devnagari characters. In *International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007)*, volume 2, pages 399–403. IEEE, 2007.

[3] Thomas G. Dietterich. Ensemble methods in machine learning. In *Proceedings of the First International Workshop on Multiple Classifier Systems*, MCS '00, page 1–15, Berlin, Heidelberg, 2000. Springer-Verlag.

[4] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1321–1330. JMLR.org, 2017.

[5] Satish Kumar. Performance comparison of features on devanagari hand-printed dataset. *International journal of recent trends in engineering*, 1(2):33, 2009.

[6] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6405–6416, Red Hook, NY, USA, 2017. Curran Associates Inc.

[7] Mandar Mhapsekar, Prathamesh Mhapsekar, Aniket Mhatre, and Vinaya Sawant. Implementation of residual network (resnet) for devanagari handwritten character recognition. In *Advanced Computing Technologies and Applications*, pages 137–148. Springer, 2020.

[8] Mahdi Pakdaman Naeini, Gregory F Cooper, and Milos Hauskrecht. Obtaining well calibrated probabilities using bayesian binning. In *Proceedings of the... AAAI Conference on Artificial Intelligence. AAAI Conference on Artificial Intelligence*, volume 2015, page 2901. NIH Public Access, 2015.

[9] Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 625–632, 2005.

[10] Umapada Pal, Sukalpa Chanda, Tetsushi Wakabayashi, and Fumitaka Kimura. Accuracy improvement of devnagari character recognition combining svm and mqdf. In *Proc. 11th Int. Conf. Frontiers Handwrit. Recognit*, pages 367–372. Citeseer, 2008.

[11] Umapada Pal, Nabin Sharma, Tetsushi Wakabayashi, and Fumitaka Kimura. Off-line handwritten character recognition of devnagari script. In *Ninth International Conference on Document Analysis and Recognition (ICDAR 2007)*, volume 1, pages 496–500. IEEE, 2007.

[12] Umapada Pal, Tetsushi Wakabayashi, and Fumitaka Kimura. Comparative study of devnagari handwritten character recognition using different feature and classifiers. In *2009 10th International Conference on Document Analysis and Recognition*, pages 1111–1115. IEEE, 2009.

[13] John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.

[14] Shrawan Ram, Shloak Gupta, and Basant Agarwal. Devanagri character recognition model using deep convolution neural network. *Journal of Statistics and Management Systems*, 21(4):593–599, 2018.

[15] Nabin Sharma, Umapada Pal, Fumitaka Kimura, and Srikanta Pal. Recognition of off-line handwritten devnagari characters using quadratic classifier. In *Computer Vision, Graphics and Image Processing*, pages 805–816. Springer, 2006.

[16] Prasad K Sonawane and Sushama Shelke. Handwritten devanagari character classification using deep learning. In *2018 International Conference on Information, Communication, Engineering and Technology (ICICET)*, pages 1–4. IEEE, 2018.