ISBN 978-81-933894-1-6
5th International Conference on Future Computational Technologies (ICFCT'2017)
Kyoto (Japan) April 18-19, 2017

Hybrid Architecture for Handwritten Numeral Recognition

Srihari Akash Chinam[1], N. V. Subba Reddy[1], Prema K. V[1] and Arushi Gupta[1]
[1]Manipal Institute of Technology, Manipal University, Manipal, India - 576104

***Abstract*:** *This paper presents a unique recognition system based on two models – a Random Forest and a Multi-Layer Perceptron. This hybrid architecture is used to classify handwritten digits, taken from the MNIST dataset. The system has two outcomes – prediction or no prediction. If both models provide the same output, system provides a prediction, else it does not. The Random Forest and Multi-Layer Perceptron are used specifically as they have shown high accuracies individually for the dataset, that is, 97.06% and 97.87% respectively. This would allow for lower rates of output mismatches. The paper also demonstrates that excluding certain pixels (features) of the image which have low variance helps increase accuracy and improve speed of computation. The proposed architecture has helped us to keep false predictions under 1% when used for the test set provided. The hybrid architecture performs better than the individual architectures alone.*

Keywords: *Random Forest, Multi-Layer Perceptron, Handwritten Digits, Hybrid architecture*

1. Introduction

In the field of classification, there exist several models which achieve high accuracies, sometimes close to 100%, when used on some testing samples. But what these models lack is the quality of uncertainty that a human being possesses. For every sample, there is an output being given by these models, without considering for once, that they might be wrong. In some cases, false predictions can be trivial but in others, they might be fatal.

This paper presents a Hybrid Classifier (HC) provided with the ability to admit uncertainty by giving no prediction in cases of confusion. It demonstrates this on the Mixed National Institute of Standards and Technology (MNIST) database [3] of handwritten numerals. The HC is composed of two models with high accuracies on the MNIST dataset – a Random Forest Classifier (RFC) [6] and a Multi-Layer Perceptron (MLP) [5]. High accuracy models were chosen so that common true predictions would be more, leading to lesser cases with no prediction.

2. Related Work

Many papers are written comparing different models, which in this case are an MLP and an RFC. For example, in the paper "Performance evaluation of neural network, support vector machine and random forest for prediction of donor splice sites in rice" [1], PK Meher et. al., compare the results of three machine learning models to select the best one for their scenario. On the contrary, in 'Preconditioning an Artificial Neural Network Using Naïve Bayes' [2] by NA Zaidi et. al., it is seen that a Neural Network is pre-trained with a Naïve Bayesian model which optimizes the mean-square-error leading to a faster convergence. Another observation made is that the pre-trained Neural Network becomes a low bias classifier after optimization of mean-square-error.

3. Data Preprocessing

The images containing handwritten numerals are taken from the Mixed National Institute of Standards and Technology (MNIST) database [3]. The dataset contains 60,000 samples for training data and 10,000 samples for testing data, with each sample having dimensions 28x28. To make the images more desirable for analysis, the following preprocessing techniques are applied:

1) *Otsu's method:* The images obtained from the MNIST database are in grayscale with each pixel value ranging from 0 to 255. This range of values provided various levels of the colour grey, leading to a blur forming in the images. After the application of Otsu's binarization [4], every pixel was represented by either a 0 or 1 (binary values). Figure 1(a) and 1(b) shows the effect of Otsu's binarization on the images.

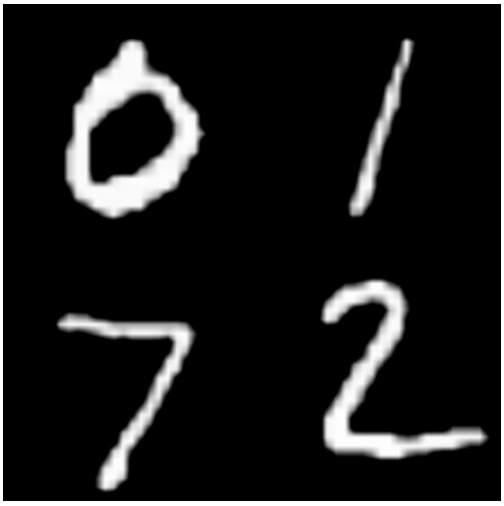

Fig. 2(a): Numerals before binarization

Fig. 1(b): Numerals after binarization

2) *Removal of low variance features:* Each image from the MNIST dataset has dimensions 28x28, resulting in 784 pixels where every pixel is being considered as a feature. However, in these 784 features, not all of them contribute to improve the analysis. There are some pixels which always have the value 0 (low variance [7]), which may reduce the quality of analysis (as they may be assumed as similarity). By eliminating these pixels as features, the total number of features were reduced to 641.

4. Classification

Before beginning the process of classification, the preprocessing techniques used on the training data had to be applied on the testing data as well. After applying Otsu's binarization, the pixels removed in the training data were removed from the testing data, assuming that the same pixels would have low variance. Upon completion of preprocessing, the RFC is trained first.

RFCs [6] are an ensemble of decision trees [8] which predict the class of an item as the mode [9] of all predictions from their respective ensemble. The RFC chosen consists of 500 estimators, that is, 500 decision trees which are trained using the 60,000 training samples first. Each of these trees are generated using Gini index [10] to assess the purity of nodes. The training data provided to each decision tree is different using bagging (bootstrap aggregating) [11], which helps to avoid overfitting [12] of data.

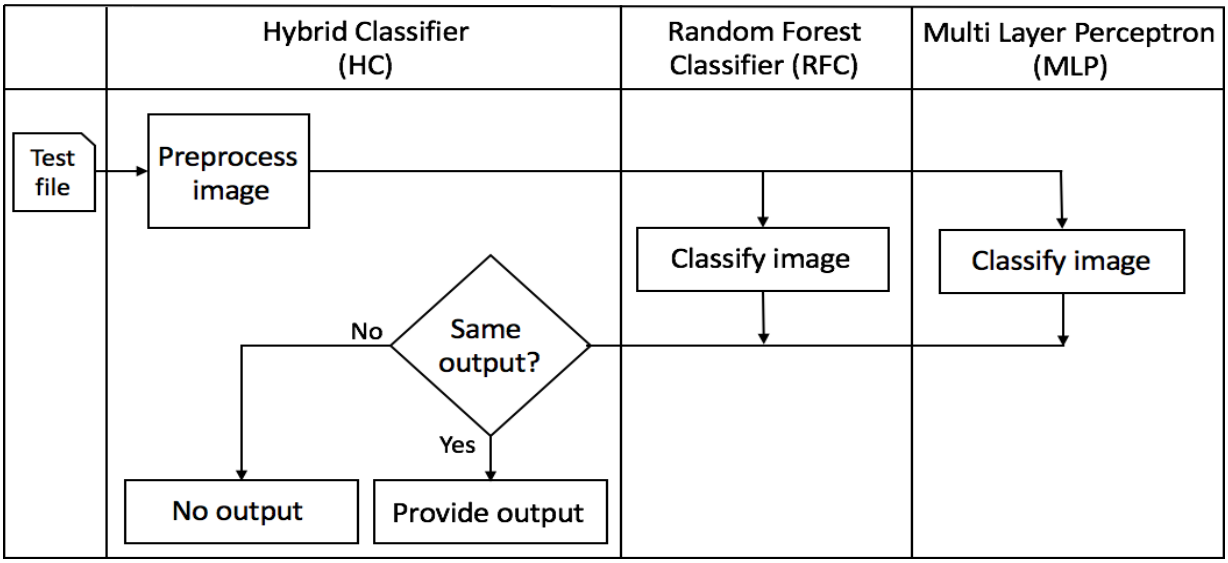

Fig. 2: Decision making process of HC

Upon completion of the training of RFC, the MLP [5] is trained. The MLP chosen consists of one hidden layer consisting of 1500 neurons, which use the logistic function (sigmoid function) [13] as an activation function. The Limited-memory Broyden-Fletcher-Goldfarb-Shannon (L-BFGS) [14] [15] algorithm is used as a solver for weight optimization. After 500 iterations of training with a learning rate of $1x10^{-7}$, the MLP was used for classification.

Initially, the testing data is passed to the RFC and MLP individually and the result is observed. After completion of individual testing, the testing data is passed to the HC, which sends the images one by one to the RFC and MLP. For every image, the HC compares the outputs from both classifiers. If the outputs match, it makes a decision to provide the prediction, else it gives no output. This process is illustrated in Figure 2.

5. Results

Upon training, the two individual models were first tested with the training data itself. It was found that these models recognised the training data with a 100% accuracy [16]. When supplied with the testing data, the RFC had an accuracy of 97±0.08% (results shown in table 1 are for an accuracy of 97.06%) with average classification time as 0.353 milliseconds, while the MLP had an accuracy of 97.87% with average classification time as 0.108 milliseconds. The results of RFC and MLP are tabulated as confusion matrices, shown in Table 1 and Table 2 respectively.

Table I: Confusion Matrix for RFC

Actual\Predicted	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
0	969	1	0	0	0	2	4	1	2	1
1	0	1124	2	4	0	1	2	0	1	1
2	7	0	996	6	1	1	4	10	6	1
3	1	0	9	976	0	7	0	7	6	4
4	1	0	2	0	955	0	5	1	3	15
5	3	0	0	15	0	861	5	1	5	2
6	8	3	0	0	2	3	940	0	2	0
7	1	6	22	0	0	0	0	988	0	11
8	6	0	4	5	5	3	2	3	936	10
9	5	5	1	9	11	5	2	4	6	961
Accuracy – 97.06% Average classification time per image – 0.353 milliseconds										

Table II: Confusion Matrix for MLP

Actual\Predicted	0	1	2	3	4	5	6	7	8	9
0	969	0	2	1	0	2	2	1	2	1
1	0	1126	3	0	0	1	2	1	2	0
2	5	1	1009	2	2	0	3	6	3	1
3	0	0	4	993	0	1	0	2	4	6
4	3	0	0	1	957	0	5	2	3	11
5	2	2	0	12	2	866	3	0	2	3
6	7	2	2	1	4	5	935	0	2	0
7	1	5	7	4	0	1	0	1000	4	6
8	4	1	2	4	6	3	0	3	947	4
9	1	3	0	4	7	3	1	3	2	985

Accuracy – 97.87%
Average classification time per image – 0.108 milliseconds

The HC was then tested with the 10000 testing samples, where it made predictions only for 9695 images (96.95% of test set). Out of these 9695 images, 9601 images were predicted correctly. For the entire test set, this is an accuracy of 96.01%. However, if only the predicted images are considered, that is, the 9695 images, it is observed that the percentage of true predictions is 99.03%, keeping false predictions under 1%. The results for the HC are tabulated as a confusion matrix shown in Table 3.

Table III: Confusion Matrix for HC

Actual\Predicted	0	1	2	3	4	5	6	7	8	9
0	965	0	0	0	0	1	0	1	0	0
1	0	1120	2	0	0	1	1	0	1	0
2	2	0	985	1	0	0	1	5	1	0
3	0	0	2	963	0	0	0	2	2	1
4	1	0	1	0	944	0	3	1	0	5
5	2	0	0	6	0	847	3	1	1	0
6	3	1	0	0	1	1	931	0	0	0
7	0	2	5	0	0	0	0	977	0	3
8	2	0	1	2	3	2	0	1	920	3
9	1	3	0	4	4	0	1	2	1	949

Number of predicted images – 9695
Accuracy for test set – 96.01%
Accuracy for predicted images (9695 images) – 99.03%
Average classification time per image – 2.035 milliseconds

6. Challenges

The challenge faced during the implementation of HC was arbitrary results from RFC. With every generation of an RFC model, the accuracy varied between 96.92% and 97.08%. Upon combination with MLP, a range of accuracies obtained for HC are between 98.89% to 99.03% (results presented in Table 3 are for the best case). This is accounted due to bagging (bootstrap aggregating) [11], as the process assigns a random and unique training set to every decision tree in the RFC each time. After extensive testing, the above best case results were obtained.

7. Conclusion and Future Applications

The motive of the paper is to bring a human touch to the current intelligent systems, which was achieved by providing them with a sense of uncertainty. After successful implementation of this system on the MNIST database, it seems reasonable to apply the strategy on real world data to help systems using AI to behave more like humans and prevent expensive consequences to unnecessary wrong decisions. As the world moves towards automation of most systems, it is important that these systems are made more reliable.

8. References

[1] Meher, Prabina Kumar, Tanmaya Kumar Sahu, and A. R. Rao. "Performance evaluation of neural network, support vector machine and random forest for prediction of donor splice sites in rice." Indian Journal of Genetics and Plant Breeding (The) 76.2 (2016): 173-180.
https://doi.org/10.5958/0975-6906.2016.00027.4

[2] Zaidi, Nayyar A., Francois Petitjean and Geoffrey I. Webb. "Preconditioning an Artificial Neural Network using Naïve Bayes." Pacific-Asia Conference on Knowledge Discovery and Data Mining. Springer International Publishing (2016).

[3] LeCun, Yann, Corinna Cortes, and Christopher JC Burges. "MNIST handwritten digit database." AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist 2 (2010).

[4] Otsu, Nobuyuki. "A threshold selection method from gray-level histograms." Automatica 11.285-296 (1975): 23-27.

[5] Longstaff, Ian D. and John F.Cross. "A pattern recognition approach to understanding the multi- layer perception." Pattern Recognition Letters 5.5 (1987): 315-319.
https://doi.org/10.1016/0167-8655(87)90072-9

[6] Ho, Tin Kam. "Random decision forests." Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on. Vol. 1. IEEE, 1995.

[7] Loève, M. "Probability theory. 1977." (1977).

[8] Safavian, S. Rasoul, and David Landgrebe. "A survey of decision tree classifier methodology." IEEE transactions on systems, man, and cybernetics 21.3 (1991): 660-674.
https://doi.org/10.1109/21.97458

[9] Runnenburg, J. Th. "Mean, median, mode." Statistica Neerlandica 32.2 (1978): 73-79.
https://doi.org/10.1111/j.1467-9574.1978.tb01386.x

[10] Lerman, Robert I., and Shlomo Yitzhaki. "A note on the calculation and interpretation of the Gini index." Economics Letters 15.3-4 (1984): 363-368.

[11] Breiman, Leo. "Bagging predictors." Machine learning 24.2 (1996): 123-140.
https://doi.org/10.1007/BF00058655

[12] Hawkins, Douglas M. "The problem of overfitting." Journal of chemical information and computer sciences 44.1 (2004): 1-12.
https://doi.org/10.1021/ci0342472

[13] Jordan, Michael I. "Why the logistic function? A tutorial discussion on probabilities and neural networks." (1995).

[14] Malouf, Robert. "A comparison of algorithms for maximum entropy parameter estimation." proceedings of the 6th conference on Natural language learning-Volume 20. Association for Computational Linguistics, 2002.
https://doi.org/10.3115/1118853.1118871

[15] Zhu, Ciyou, et al. "Algorithm 778: L-BFGS-B: Fortran subroutines for large-scale bound-constrained optimization." ACM Transactions on Mathematical Software (TOMS) 23.4 (1997): 550-560.
https://doi.org/10.1145/279232.279236

[16] Sokolova, Marina, and Guy Lapalme. "A systematic analysis of performance measures for classification tasks." Information Processing & Management 45.4 (2009): 427-437.
https://doi.org/10.1016/j.ipm.2009.03.002