Chapter 3

A NOVEL MACHINE LEARNING METHOD FOR PATTERN RECOGNITION AND CLASSIFICATION

Ali Mert CEYLAN[1]
Mete EMİNAĞAOĞLU[2]

INTRODUCTION

Handwritten digit recognition is known to have practical importance because of well-established datasets and real-world applications such as automated form reading and handwritten zip code processing (Keysers, 2007). History of MNIST database resides in the early requirements and the developments achieved in the AT&T Bell Laboratories. The MNIST database is the historical descendant of the NIST database. However, NIST training and test sets were representatives of different distributions (LeCun et al., 1995). After NIST dataset has been re-partitioned, and images' size been normalized, the MNIST database was created. It consists of 60000 training images and 10000 test images, drawn from the same distribution. All these grayscale digits are normalized and centered in a fixed-size image with 28x28 pixels (Deng, 2012).

Several classification algorithms have been developed to deal with the task specifically or optimized for pattern recognition tasks (Bishop, 2006). The rapid progress is a result of various developments, such as inexpensive computers and new algorithms "that take advantage of allowing larger datasets to be used for training and testing of pattern recognition" (LeCun et al., 1995). However, there exist kernel methods that hash data points and preserve similarity in the origin space (Li et al., 2016). During the new progress in data science, various algorithms are devised that can be grouped as linear classifiers, k-nearest neighbors, ensemble learners, decision trees, non-linear classifiers, support vector machines, artificial neural networks, convolutional networks (Deng, 2012; Bouali & Akaichi, 2016).

Most of the pattern recognition systems rely on two main modules. The first module is the feature extractor, which enables the representation of input patterns by vectors that could be matched and that are relatively invariant to trans-

[1] PhD student, Department of Computer Engineering, Ege University, Bornova, 35100, İzmir, Turkey
[2] Assist. Prof. Dr., Department of Computer Science, Dokuz Eylül University, Buca, 35390, İzmir, Turkey

be trained with the dataset first and then the artificial neural network can begin training on reconstructed images.

If the training times could be reduced and the accuracy performance could be increased, then this novel model might be considered as a promising alternative classifier for industrial implementations and applications that require a very small amount of storage space and limited processors, such as the sensors used in IoT (Internet of Things) and Industry 4.0. Even with their outstanding accuracy and precision performances, convolutional neural networks and other similar deep learning models and architectures are still very difficult to be implemented and used in simple devices with low capacities, such as the ones used in IoT.

REFERENCES

Adelson-Velsky, G. M. & Landis, E. M. (1963). *An algorithm for the organization of information.* London: DTIC Document.

Aha, D. W., Kibler, D., & Albert, M. K. Instance-based learning algorithms. *Machine Learning*, 6(1), 37-66. 1991.

Bishop, C. (2006). *Pattern Recognition and Machine Learning*. New York: Springer-Verlag.

Bouali H. & Akaichi, J. Event Streams Clustering Using Machine Learning Techniques. *J. Syst. Integr,* 6(4), 53-66. 2015.

Breiman, L. Random Forests. *Machine Learning*, 45(1), 5-32. 2001.

Buduma, N. & Locascio, N. (2017). Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithms. USA: O'Reilly Media, Inc.

Cohen, W. W. (1995). Fast Effective Rule Induction. *Twelfth Int. Conf. on Machine Learning,* Tahoe City, USA, (pp. 115-123).

Deng, L. The MNIST database of handwritten digit images for machine learning research. *IEEE Signal Process. Mag.*, 29(6), 141–142. 2012.

Dictionary of Algorithms and Data Structures: AVL tree. (https://xlinux.nist.gov/dads/HTML/avltree.html. Accessed: 17 November 2018).

Digit Recognizer: Learn computer vision fundamentals with the famous MNIST data. (https://www.kaggle.com/ngbolin/mnist-dataset-digit-recognizer/data. Accessed: 15 October 2019).

EMNIST (Extended MNIST): An extended variant of the full NIST dataset. (https://www.kaggle.com/crawford/emnist. Accessed: 19 July 2019).

Fashion MNIST: An MNIST-like dataset of fashion images. (https://www.kaggle.com/zalando-research/fashionmnist. Accessed: 07 July 2019)

Gavrila, D. M. A Bayesian, exemplar-based approach to hierarchical shape matching. *IEEE Trans. Pattern An. Mach. Intell.,* 29(8), 1408–1421. 2007.

Gavrila, D. M. & Philomin, V. (1999). Real-time object detection for smart vehicles. *Seventh IEEE Int. Conf. Comput. Vis.,* 20-27 September 1999, Kerkyra, Greece, (pp. 87–93).

Go, C. et al. Deep neural network with FGL for small dataset classification. *IET Image Process*, 13(3), 779-784. 2019.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. Cambridge: MIT Press.

Han, J., Pei, J. & Kamber, M. (2011). *Data mining: concepts and techniques*. New York: Elsevier.

Haykin, S. (2009). *Neural networks and learning machines, vol. 3*. New Jersey: Pearson Education Inc.

Keysers, D. Comparison and combination of state-of-the-art techniques for handwritten character recognition: topping the MNIST benchmark. *ArXiv Prepr. ArXiv07102231*. 1–13. 2007.

Kohonen, T. (1990). Improved Versions of Learning Vector Quantization. *IEEE IJCNN*, San Diego, USA, (pp. 545–550).

Larose, D. T. & Larose, C. D. (2014). *Discovering knowledge in data: an introduction to data mining.* New York: John Wiley & Sons.

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE.*, 86(11), 2278–2323, 1998.

LeCun, Y. et al. (1995). Comparison of learning algorithms for handwritten digit recognition. *ICANN95 - Int. Conf. Artif. Neural Netw.*, 9-13 October 1995, Maison de la Chimie, Paris, (pp. 53–60).

Li, Y-Q. et al. Kernelised supervised context hashing. *IET Image Process*, 10(12), 986-995. 2016.

Nievergelt, J. & Reingold, E. M. Binary Search Trees of Bounded Balance. *SIAM J. Comput*, 2(1), 33–43. 1973.

Quinlan, R. J. (1993). *C4.5: Programs for Machine Learning.* San Mateo, CA: Morgan Kaufmann Publishers.

Salzberg, S. A Nearest Hyperrectangle Learning Method. *Machine Learning,* 6(3), 251–276. 1991.

Stefanowski, J. (1998). The rough set based rule induction technique for classification problems. *Sixth European Cong. on Intelligent Tech. and Soft Computing*, 7-10 September, Aachen, Germany, (pp. 109-113).

Sumner, M., Frank, E., & Hall, M. (2005). Speeding up Logistic Model Tree Induction. *Ninth European Conf. on Principles and Practice of Knowledge Discovery in Databases,* 3-7 October 2005, Porto, Portugal, (pp. 675-683).

Tsapanos, N., Tefas, A. & Pitas, I. Online shape learning using binary search trees. *Image Vis. Comput,* 28(7), 1146–1154. 2010.

Weiss, M. A. (2013). *Data Structures and Algorithm Analysis in C++* (4th ed.). New York: Pearson Education.

Weka: Weka Machine Learning Group at the University of Waikato. (http://www.ss.waikato.ac.nz/ml/Weka/ Accessed: 07 June 2018).

Wetzstein, B. et al. Preventing KPI violations in business processes based on decision tree learning and proactive runtime adaptation. *J. Syst. Integr,* 3(1), 3–18. 2012.

Witten, I. H., Frank, E., & Hall, M. A. (2011). *Data Mining: Practical Machine Learning Tools and Techniques*. Burlington, MA: Morgan Kaufmann Publishers.

Yang, W. et al. Thinning of convolutional neural network with mixed pruning. *IET Image Process*, 13(5), 491-497. 2019.