Expert Systems with Applications 41 (2014) 8180–8188

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Combining additive input noise annealing and pattern transformations for improved handwritten character recognition

J.M. Alonso-Weber [*], M.P. Sesmero, A. Sanchis

Department of Informatics, Polytechnic School of Engineering, Universidad Carlos III de Madrid, Avenida de la Universidad 30, Leganés 28911, Madrid, Spain

ARTICLE INFO

Article history:
Available online 19 July 2014

Keywords:
Artificial Neural Networks
Back Propagation
MNIST
Handwritten text recognition

ABSTRACT

Two problems that burden the learning process of Artificial Neural Networks with Back Propagation are the need of building a full and representative learning data set, and the avoidance of stalling in local minima. Both problems seem to be closely related when working with the handwritten digits contained in the MNIST dataset. Using a modest sized ANN, the proposed combination of input data transformations enables the achievement of a test error as low as 0.43%, which is up to standard compared to other more complex neural architectures like Convolutional or Deep Neural Networks.

1. Introduction

Handwritten text recognition is a demanding problem for which ANNs are well suited learning models, as is shown by ongoing research. There are multiple technological applications that require a more or less robust ability to perform handwritten text recognition, for example the validation of signatures in banks, the entry of text in mobile devices or for large-scale digitizing and archival of manuscripts (Plamondon & Srihari, 2000).

The complexity of the text recognition varies greatly depending on the type of the context or the required application. The recognition of handwritten text always demands a more sophisticated approach that in the case of machine written texts. There is a greater difficulty in recognizing writer independent unconstrained continuous handwritten text than for texts written with printed isolated letters.

Some applications such as the identification of signatures in banks use a different approach as they rely on touch-pads that allow obtaining the data describing the sequence followed during the writing process, in what is called on-line recognition. In this case there is information available in real time on the strokes that make up the text, i.e., the direction, angle, speed and pressure.

The biggest challenge in text recognition arises when addressing the off-line variant, which operates on digitized raster images and lacks information about the strokes, sequencing and timing of the writing process.

The digitized image will be contained in a plane, where each point represents a pixel with a variable intensity between the white paper and the dark ink strokes. Such collection of pixels represents a high dimensional and complex set of data with regard to the recognition of characters. This is the reason for which the off-line recognition techniques have to resort to a series of previous processes that can transform the images into a more usable set of data. These processes involve several tasks as binarization, contrast and brightness adjustments, skeletonization, and noise removal. A crucial and complex task prior to the recognition is to segment each of the paragraphs, lines and characters contained in the image (Alonso-Weber, Galván, & Sanchis, 2003; Alonso-Weber & Sanchis, 2011; Lacerda & Mello, 2013), which may require some feedback from the recognition stage (Fernández-Caballero, López, & Castillo, 2012).

Mainstream research deals with the problem of extracting and using explicit information from the high dimensional image data. The purpose of this Feature Extraction is two-fold. At first hand, the interest is to extract structured and relevant information from the high volume of data available in an image, and to dismiss redundant or useless information. The second purpose is to reduce the dimensionality of the data, which can help to find better solutions and decreases the computational costs associated to the learning and test processes.

A different approach deals with implicit feature extraction, where the recognition is performed directly on the original, unprocessed image data. A machine learning method is applied and expected to extract the relevant information from the raw data. Here, several ANN models have shown to be particularly efficient, achieving rather low error rates such as a 0.4% and 0.39% for Convolutional Neural Networks (Ranzato, Poultney, Chopra, & LeCun,

[*] Corresponding author.
E-mail addresses: jmaw@ia.uc3m.es (J.M. Alonso-Weber), msesmero@inf.uc3m.es (M.P. Sesmero), masm@inf.uc3m.es (A. Sanchis).

http://dx.doi.org/10.1016/j.eswa.2014.07.016
0957-4174/© 2014 Elsevier Ltd. All rights reserved.

2006; Simard, Steinkraus, & Platt, 2003), 0.35% for Deep Neural Networks (Ciresan, Meier, Gambardella, & Schmidhuber, 2010), or even 0.27% and 0.23% for Committees of Convolutional Neural Networks (Ciresan, Meier, Gambardella, & Schmidhuber, 2011; Ciresan, Meier, & Schmidhuber, 2012).

Our proposal follows the raw recognition approach, and is based on a modestly sized ANN with two hidden layers containing 300 × 200 cells. This is considerably smaller than most of the ANN architectures shown in the literature. The training is performed with the standard Back Propagation Algorithm. We show that good results can be achieved boosting the training process with a combination of several input pattern transformations. These transformations comprise a line-up of affine transformations including a trapezoidal deformation, a dimensionality reduction through image downsizing, and an annealed input noise addition schema. Previous work of the authors involved handwritten recognition through skeletal structures and had a limited accuracy result due to the usual problems of premature stalling of the Back Propagation algorithm (Alonso-Weber & Sanchis, 2011).

The performance of our proposal is validated against the MNIST dataset (LeCun, Bottou, Bengio, & Haffner, 1998; LeCun & Cortes, 1998), that has been used thoroughly over the time to develop increasingly refined recognition algorithms.

The next section contains a review of the work related to our proposal, which is presented in section three, whereas the experimental setup and evaluation are shown in section four.

2. Related work

As it has been already mentioned, there are several ANN models that have shown to be particularly efficient performing the handwritten text recognition starting from the raw images. Due to the complexity of the task, ANNs with multiple layers are used, because it is expected that the first layers are able to detect the most basic features, while the subsequent layers construct higher level feature detectors based on the detected features from the precedent layers.

Several proposals have been made, including Convolutional Neural Networks (CNN) and Deep Neural Networks (DNN), which are now described. We also depict here several issues that are related with the proposal of this paper, such as pattern deformations, input noise addition and dimensionality reduction.

2.1. Convolutional and Deep Neural Networks

Convolutional Neural Networks (LeCun et al., 1998) are biologically inspired models that can be traced back (Fernandes, Cavalcanti, & Ren, 2013) to the work of (Hubel & Wiesel, 1962) and Fukushima's Neocognitron (Fukushima, 1980). In a CNN each cell in the first layer has a set of inputs (receptive fields) coming from a limited region of the input space (retinotopic region). These cells are arranged in a tile structure covering the whole input space. The function of these cells is to perform a simple filtering i.e., a feature detection based on the high correlation between neighboring input pixels, which is called convolutional mapping. The cells in the subsequent layers replicate the same structure, with their inputs covering a small region of the preceding layer. The stacking of these retinotopic mappings builds progressively more complex feature detectors that cover a greater input region.

Layers with a different functionality can be alternated: a convolutional layer can be cascaded with a so called *max-pooling* layer that performs a subsampling of the detected features. For the final output a conventional fully-connected layer can be used.

A specific constraint joint to the max-pooling layers allows these detectors to develop invariance to the position of the features. This is accomplished using neighboring cells that share the same weights, i.e., they perform the same feature detection with a slight displacement in the input space.

This constraint and the sparse connectivity of the model have associated a reduced parameter space, where the search for an optimal weight configuration can be performed in a more efficient way than in a conventional ANN with a fully connected architecture.

Some applications of CNNs on the MNIST dataset have reported very low error rates: such as a 0.4% for (Simard et al., 2003) and 0.39% for (Ranzato et al., 2006).

Training ANNs with Back Propagation has a serious inconvenience when several hidden layers are used. The errors fade (Hochreiter & Munchen, 1998) when they are propagated back through multiple layers, and this hinders a proper learning. Several authors proposed alternative algorithms to overcome this inconvenience. Hinton (2007) and Bengio and Lamblin (2007) proposed unsupervised methods for training independently each layer of a Deep Belief Neural Network, with a subsequent fine tuning with a supervised learning algorithm. Based on these concepts (Ciresan et al., 2010) proposed a Deep Neural Network with error rates as low as 0.35%.

There are other posterior works that show even lower error rates as 0.27% (Ciresan et al., 2011) and 0.23% (Ciresan et al., 2012), based on combining several CNNs into additive ensembles or committees. These ensembles can achieve an improvement in the accuracy in an additional 0.1% (at these extreme performance ratios) with respect to the underlying neural model.

In practice, the multilayer models are extremely profuse in the number of layers, cells and weights, even in spite of the sparse connections and the weight sharing of the CNNs. This leads to one of their major inconveniences, which is the need of a very high computational power in order to achieve reasonable training times. In the case of Ciresan et al. (2010) the DNN with the best performance has 7500 cells distributed in five hidden layers that are fully connected, and about 12.000.000 connections. The use of Graphical Processing Units (GPU) allows a considerable boosting in the processing times (Ciresan et al., 2010) with a reported speedup of 40x. These GPUs have allowed exploring the possibilities of the DNNs, but require more elaborate programs for the simulations.

2.2. Pattern deformations

Although the use of DNNs and CNNs may help to find good solutions, another problem that they do not avoid is the need of using a very large set of training samples. Even datasets like the MNIST that contains 60,000 training instances fall short for a satisfactory training.

A usual strategy is extending the training set performing some kind of changes on the images. These changes are mostly affine transformations (Yaeger, Lyon, & Webb, 1996) like displacements, rotations and linear deformations. Simard et al. (2003) proposes a distortion process based on elastic deformations which tries to reproduce the natural uncontrolled oscillations of the writer's hand. Two parameters allow adjusting these distortions.

The deformation helps the learning process to develop a recognition ability that is more invariant to changes in the input data of the test, and reduces the error rate.

2.3. Input noise addition, weight decay and regularization

Another problem that weights down the performance of Back Propagation is the fact that convergence usually stalls in local minima, often far away from a reasonable working point.

Several perturbation techniques have been used in the past in order to improve the convergence and the generalization abilities