Received: 15 April 2016 | Revised: 17 May 2016 | Accepted: 2 Jun 2016
DOI: 10.37917/ijeee.12.1.10 | Vol. 12| Issue 1 | June 2016

Open Access

Iraqi Journal for Electrical and Electronic Engineering
Original Article

Classification Algorithms for Determining Handwritten Digit

Hayder Naser Khraibet AL-Behadili
Computer Science Department,
Shatt Al-Arab University College, Basrah, Iraq
haider_872004 @yahoo.com

Abstract: *Data-intensive science is a critical science paradigm that interferes with all other sciences. Data mining (DM) is a powerful and useful technology with wide potential users focusing on important meaningful patterns and discovers a new knowledge from a collected dataset. Any predictive task in DM uses some attribute to classify an unknown class. Classification algorithms are a class of prominent mathematical techniques in DM. Constructing a model is the core aspect of such algorithms. However, their performance highly depends on the algorithm behavior upon manipulating data. Focusing on binarazaition as an approach for preprocessing, this paper analysis and evaluates different classification algorithms when construct a model based on accuracy in the classification task. The Mixed National Institute of Standards and Technology (MNIST) handwritten digits dataset provided by Yann LeCun has been used in evaluation. The paper focuses on machine learning approaches for handwritten digits detection. Machine learning establishes classification methods, such as K-Nearest Neighbor(KNN), Decision Tree (DT), and Neural Networks (NN). Results showed that the knowledge-based method, i.e. NN algorithm, is more accurate in determining the digits as it reduces the error rate. The implication of this evaluation is providing essential insights for computer scientists and practitioners for choosing the suitable DM technique that fit with their data.*

Index Terms -Neural Network, K-Nearest Neighbor, Decision Tree, Predictive Modeling

I. Introduction

Machine learning plays an important role in extracting knowledge from huge amount of data by constructing a model from explored dataset. It is the only approach in DM that allows a specific algorithm to learn from data. The functionality of such algorithm relies on building a model based on the inputs in training phase to be used in predicting new rules for next decisions [1]. This implicit learning approach has exchanged successfully traditional way of following the programmed instructions. One of the biggest aims in DM is knowledge discovery [2]. This can be done by analyzing data from different views and perspectives to summarize it into a new knowledge. In DM software, new patterns and correlations can be extracted from dozens of fields in large relational databases.

Data is reorganized in terms of systematic datasets that reflect the perspective database [3]. This paper focuses on exploring handwritten digits datasets to predict unknown handwritten digits. The aim in the present paper is to identify the best algorithm in solving the problem of similarity problem between two digits by answering the following research questions:

How do we determine the right requirements for analyzing data (Data Pre-Processing)?

How do we discover the best DM classifier in handwritten digits?

How do we evaluate the result of DM under different algorithms?

This paper is organized as follows. Section 1 gives an introduction. Section 2 highlights the problem statement and motivation. Methodology to tackle the problem has detailed in Section 3 while the experimental design and results illustrated in the same section. Sections 4 explain the conclusions and future work.

II. MOTIVATION AND PROBLEM STATEMENT

Handwriting determination and recognition has been a main subject of research for more than four decade. This paper analyzes the behavior of some classification techniques in large handwriting dataset to predict the digits number [4]. Machine-learning techniques, particularly when applied to NN, have played an increasingly important role in the design of pattern recognition systems. Several methods have been developed in hand digit detection and classified into categories: knowledge-based methods, feature-based methods, template-based methods, and appearance-based methods [5]. Data pre-processing such as binarazaition is an important phase in manipulating and mining data.

However, there is a lack of experimental evaluation that analyses the behavior of such methods together. In addition, studies that deal with this aspect have ignored the part of binarazaition preprocessing.

III. METHODOLOGY

The dataset used is MNIST which is very popular in field of handwritten digits. This dataset consist of a training set of 60,000 records and a test set of 10,000 records. Each record consists of 784 features which present image in grey scale. The digits have been size-normalized and centered in a fixed-size image [6].

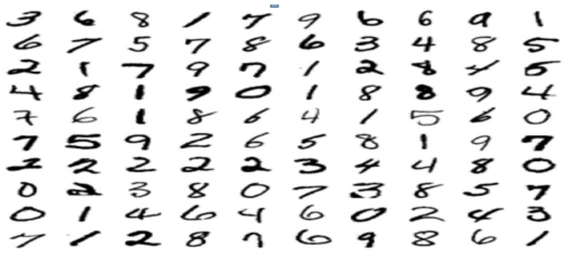


Fig. 1 Digits picked from MNIST

The handwritten digits were centered in a 28x28 pixel image by calculates the center of mass of the pixels, and translating the image to position this point at the center of the 28x28 field. The primary contribution of this work is finding similarities between MNIST digit shapes. The figure below is an example of two handwritten digits. In terms of pixel-to-pixel comparisons, these two digits have many differences, but to a human, the shapes are considered to be corresponding; hence, we need to find a new methodology that uses some feature to predict the digits correctly.

Fig. 2 Two digits randomly picked from the MNIST dataset

Figure 3 illustrates the main steps of mining process. These are data preprocessing (binarization), DM techniques (processing), comparison, and evaluation.

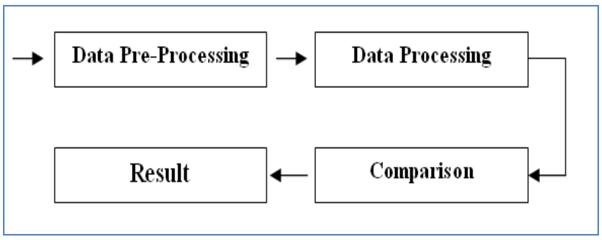


Fig.3 DM process

A. *Data Preprocessing*

In this step, binarization is applied to data preprocessing. Binarization used to modify the digit data from gray scales to a black and white digit data [7]. Every digit in the original dataset is an image with 28 × 28 pixels corresponding to 784 attributes, each of which has a value within a range of 0–255. The binarization process converts each of the 784 values into a 0 or 1. The binarization algorithm leaves 0 as is while converting any non-zero value into 1. Therefore, we essentially change a gray-scale image into a black-and-white one. Now, we have the two datasets, the gray scale and the black-and-white.

B. *Data Processing*

Data should be processed to extract knowledge. Classification technique is useful to predict group membership for data instances. The classification process used to learn from handwritten digits dataset to predict the digit. The algorithms processes are used to train dataset attributes by using the MNIST training dataset. The algorithms and techniques seek to find out the relationship between the features to make the predictive task of the digits are possible. Thus, the model to obtain the error rates is evaluated by using the MNIST test dataset as shown in Figure 4. The classification techniques that have been evaluated in this paper are K-NN, DT, and NN,

K-NN algorithm

K-NN algorithm is a non-parametric method used for classification and regression. It is one of lazy learner methods.

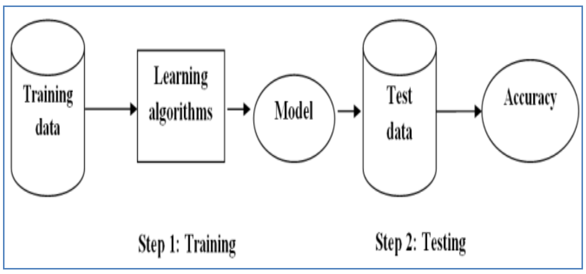
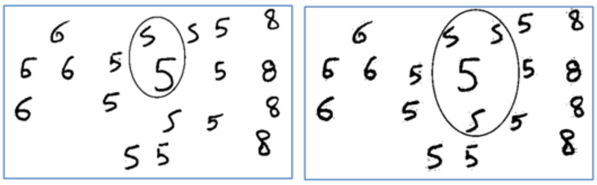


Fig. 4 MNIST test dataset

It waits until it receives test data and once successfully tested, it then determines which of its training data is most similar to the test data [8]. This algorithm works by discovering K points from the training handwritten digit that are nearest to the data that need to be predictive. It calculates the majority vote of the closest neighbors of each K points (see Figure 5).



Nearest Neighbor 3 Nearest Neighbors

Fig 5 The K-NN process

The main idea for KNN is to assign specific class data that own the most representatives within the closest neighbors of the class. The algorithm computes the distance (similarity) between each test example z=(x',y') and all the training example (x,y)∈*D* to determine its closest neighbors, *Dz*.

The KNN classification algorithm:

1. Let K number of closest neighbors & *D* the set of training example.
2. For each test z=(x',y') do
3. Calculate d(x',x), the distance between z and every example , (x,y)∈D.
4. Select *Dz* ⊆ *D the set of K nearest training example to z.*

5.

$$y' = \underset{v}{\operatorname{argmax}} \sum_{(xi,yi)\in Dz} Wi \times I(v = yi) \quad (1)$$

6. End For

Where y' is Majority Voting, v is class, i is pointer, yi is the class attribute of closest neighbor, Wi is weight of each neighbor, xi is closest neighbor.

Let 5 be the digit to be class labeled and name it as (v), (yi) the class attribute for point of the nearest neighbors. I is a pointer procedure which return 1 value when argument is true and 0 when it false. The methods used to decrease the impact of K is to measure the effectiveness only for closest neighbor xi calculated based on its similarity: Wi=1/d(x ',xi)2.

The chart below compares between the accuracies rate of the different seven K neighbors in MNIST data set in both Original & Binary Data set. The various neighbors successively are 1, 3, 5, 10, 15, 17, and 23. Figure 6 depicts the way that this algorithm classifies objects.

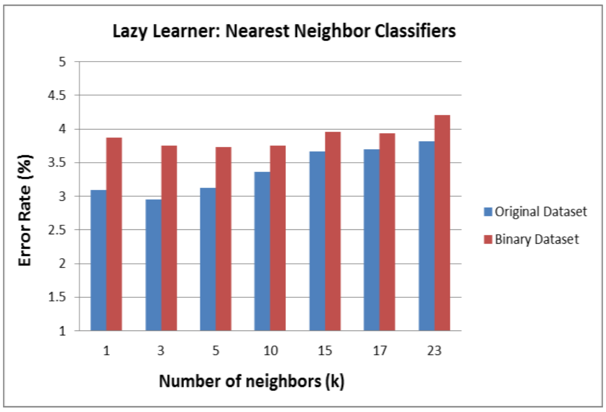


Fig. 6 Comparing the accuracy of various K.

DT Classifier

In this step, the DT classifier is used, which has widely used in classification. The aim is to have a smart tree model that can classify the class of unknown instance by discovering list of rules from the dataset of handwritten digits. DT classifier tools have been used in different fields, such as digit and text predicative, extract information, machine learning, and recognition. This research is focused on Hunt's algorithm to create DTs that are the basis of many existing DT induction algorithms, including the C4.5. The J48 (Weka DM tool) [9] is an open source for implementing the C4.5 algorithm used in this stage. The algorithm works as follows.

(i) Selecting the attribute for the root node and creating the branch for each possible attribute. The method develops by selecting the best spilt of the dataset by measuring the impurity for each child nodes.
The three equations below (Entropy, Gini and classification error) are famous in compute the impurity measure. The C4.5 algorithm used Entropy.

$$Entropy(t) = -\sum_{i=0}^{c=1} p(i|t) log2\, p(i|t) \qquad (2)$$

$$Gini(t) = 1 - \sum_{i=0}^{c=1} [p(i|t)]^{2} \qquad (3)$$

$$classification\, error(t) = 1 - \max_{i} [p(i|t)] \qquad (4)$$

Where t is a sample of training examples, C is number of classes, i is fraction of records belong to class, p(i|t) is relative frequency of class i ,and 0 log2 0=0 in entropy calculation

(ii) Splitting instances into subsets for each branch extending from the node. The information gain is depends on the measuring of the entropy after split the dataset for each attribute and compute the degree of impurity in the population of attributes. Constructing the DT is all about finding the attribute that returns the highest information gain and we choose as the decision node.

$$\mathrm{Gain}(T, X) = Entropy\ (T) - Entropy(T, X) \qquad (5)$$

Where Entropy (T) is entropy (parent) for all class in the dataset, Entropy (T,X) is [average entropy(children)] for specific attribute (X).

(iii) A branch with the value 0 entropy is a leaf node. A branch with the value grater than 0 entropy needs further splitting.
(iv) Iterate for each node using only records that reach the node.
(v) When all instances have the class, then it will stop.

Figure 7 below compares the accuracies of two datasets using only the DT classifier (J48).

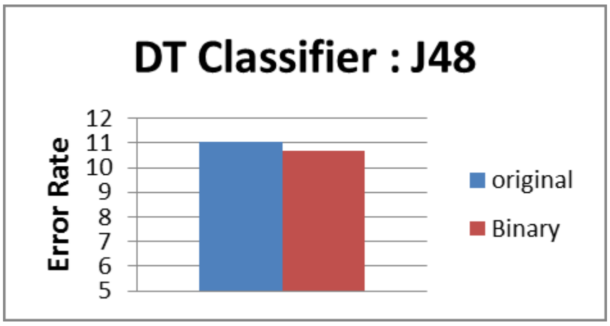


Fig. 7 DT Classifier Results for the Prediction of Handwriting

Neural Network Classifier

Another classifier we used is a fully connected ANN in both One Hidden-Layer and Two Hidden-Layer . The handwritten digits are 28x28 pixel image. The data set consist of 784 attributes numeric type (0 to 255) in the original data set, and nominal type (0-1) after binarized. These attributes represent the input of NN while the output nodes represent the class values (0-9) digits. Theoretical research shows that any function can be approximated by a one hidden-layer NN (two layers of weights). Other researchers have observed that two hidden-layers NN architecture sometimes give better performance in practical situations. Figure 8 shows architecture of an ANN.

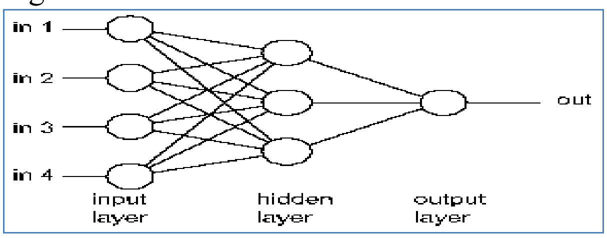


Fig. 8 ANN architecture

The processing of basic ANN consists of receives inputs in the input layer, processing the input and finally delivers output in the output layer. To see the effective architecture, one hidden-layer and two hidden-layers NN were trained with the classical back propagation algorithm gradient. Back propagation is frequently used to learn the model for ANN [10]. It calculates the weights and gives a multilayer network with set of weights and interconnections. This classifier works based on weights and the transfer function together. The function used is Sigmoid, and the output varies continuously but not linearly as input changes.

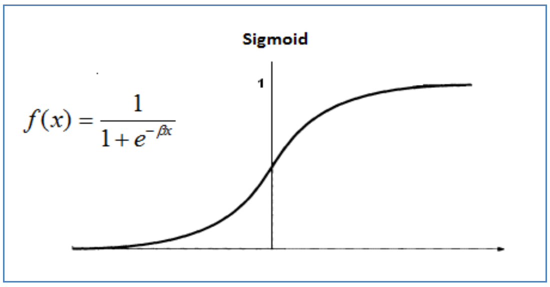


Fig.9 Sigmoid function

Result shows that multiplayer NN is the best in error rates among other DM classifiers especially when we train binary data set. Error by using one and two hidden layer was better when we increase the hidden units. Figures 10 and 11 display the error rate by using ANN classifier.

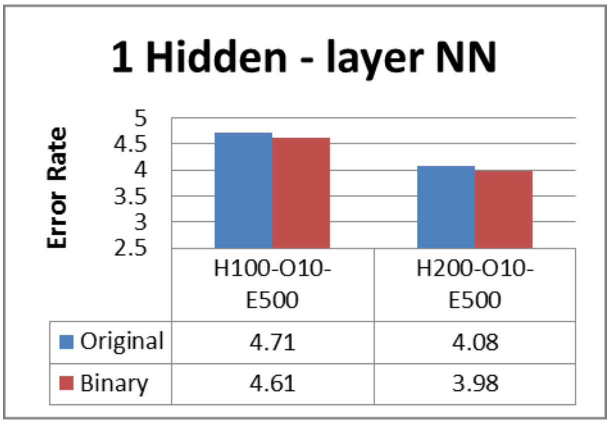


Fig.10 The Error rate using 1 hidden layer ANN Classifier

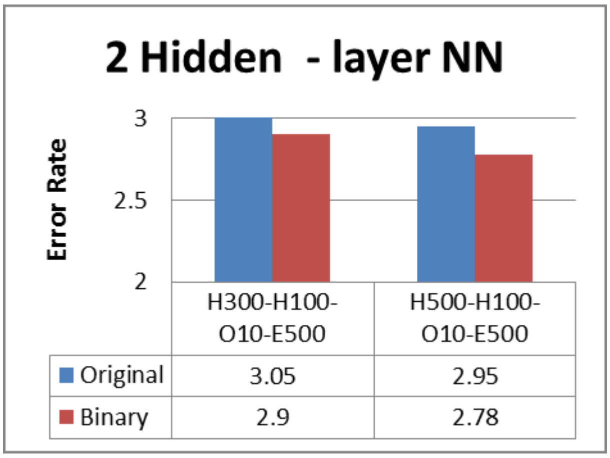


Fig.11 The Error rate using 2 hidden layer ANN Classifier

Where H is Hidden Unit, O is output neurons, E is epoch's number.

IV. CONCLUSION AND FUTURE WORK

As the data-intensive science has become separated paradigm among other sciences, this paper has evaluated the performance of K-nearest neighbor, decision tree and neural network in terms of knowledge extraction from data. New insight has been discovered to be harnessed as guidance for computer scientists and practitioners in handwritten digit recognition. Results showed that both ANN and K-NN classifiers able to provide a high accuracy in classification task for data mining. Future work will focus on exploring metaheuristic algorithms such as ant colony algorithm to discover the best rules as a new approach for data mining in the task of predictive.

REFERENCES

[1] A. Smola and S. V. N. Vishwanathan, "Introduction to machine learning," Methods Mol. Biol., vol. 1107, pp. 105–128, 2014.

[2] Mooney R.J., Bunescu R., "Mining knowledge from text using information extraction," *ACM SIGKDD explorations newsletter*., vol. 7, no. 1, pp. 3–10, 2005.

[3] I. Gorton, P. Greenfield, A. Szalay, and R. Williams, "Data-Intensive Computing in the 21st Century," *IEEE Computer*, vol.41, no.4, pp. 30-32, 2008.

[4] K. P. Feder and A. Majnemer, "Handwriting development, competency, and intervention," *Developmental Medicine and Child Neurology*, vol. 49, no. 4. pp. 312–317, 2007.

[5] C. Davatzikos and J. L. Prince, "Convexity analysis of active contour problems", *Image Visual Computing J.*, vol. 17, no. 1, pp. 27-36, 1999.

[6] E. Kussul and T. Baidyk, "Improved method of handwritten digit recognition tested on MNIST database", *Proc. 15th Int. Conf. Vision Interface*, vol.22, pp. 192-197, 2002

[7] C.L. Liu, K. Nakashima, H. Sako and H. Fujisawa, "Handwritten Digit Recognition: Benchmarking of State-of-the-Art Techniques," *Pattern Recognition*, vol. 36, no. 10, pp. 2271-2285, Oct. 2003.

[8] Y. Lee, "Handwritten digit recognition using k-nearest neighbor, radial-basis functions, and backpropagation neural

networks", *Neural Computation*, vol. 3, no. 3, pp. 440-449, 1991

[9] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA Data Mining Software: An Update," ACM SIGKDD Explor., vol. 11, no. 1, pp. 10–18, 2009.

[10] L. Bottou, C. Cortes, J. S. Denker, H. Drucker, I. Guyon, L. D. Jackel, Y. A. LeCun, U. A. Mü,ller, E. Sä,ckinger, P. Y. Simard, and V. N. Vapnik, "Comparison of classifier methods: A case study in hand written digit recognition", *Proc. 12th Int. Conf. Pattern Recognition and Neural Networks, Jerusalem*, pp. 77-87, 1994.