International Journal of Applied Science & Engineering Research

Manually Written Digit Identification using Machine Learning Based Logistic Regression Algorithm

Shruti Bhargava Choubey[1], M. Harshitha[2], B. Keerthi[3], Abhishek Choubey[4]

Department of Electronics & Communication Engineering, Sreenidhi Institute of Science and Technology, Hyderabad, India

shrutibhargava@sreenidhi.edu.in, 17311A04K6@sreenidhi.edu.in, 17311A04K7@sreenidhi.edu.in

Date of Submission: 12th July 2020 Revised: 16th October 2020 Accepted: 15th December 2020

***Abstract*—Recognition of handwritten digits is a serious issue in using the pattern identifications. The use of digital recognition includes posting bank check editing, data entry etc.Tones of studies have shown that Neural networks and machine learning models have a good performance. Deep learning and Neural Network algorithms are a part of Machine Learning methods of Data Classification, which can use the previous data to predict the upcoming data which will also help in decision altering in many situations. Digital Recognition is a combination of Deep Learning as well as Neural Network algorithms, which use packages such as sklearn, numpy and pandas to build a model. This project describes how data is taken, classified, processed and predicted appropriately using Machine Learning Models. In this paper we use Logistic Regression to predict the accuracy and we have got an accuracy of 95%.**

INTRODUCTION

Recognition of handwritten digits is an important issue for Recognizing the Characters, and can be applied as a test case for ideas like recognizing patterns and machine learning's high efficiency models. To promote machine learning research as well as pattern recognition, many common details have emerged.

Handwriting recognition is one of the obligatory as well as attractive jobs because everyone in this country has their style of writing. The great difficulty of handwriting digit recognition varies greatly in size, shape, thickness, severity of stroke, rotation and twisting of number images since they are written by a large number of people of unique styles of writing. In day to day life , such as the conversion of manual information into digital, postal code recognition, processing of checks, identifying signatures, this process is required[12,19].

This study aims to identify handwritten numbers using tools from Machine Learning to train classifiers, therefore producing higher recognition performance.

The data set from MNIST is used for this process. This data set contains seventy thousand handwritten numbers from 0 to 9. Every picture in this data set is represented as a list of features of dimensions 28x28. 784 matching pixels with numbers from 0 to 255.This study focuses on feature removal as well separation.

The objective of this study is to build a credible approach to recognition of handwritten digits. Getting complete accuracy is still a challenge and research is being carried by many engineers to minimize the rate of error. Accuracy is an important aspect for such applications. Small errors can also cause big changes in results that are to be obtained.

1.1 Brief History

Using Shallow networks, which have previously produced good results in handwriting recognition. Detailed study on novel ways for separating handwritten numbers, characters, and words has been published in a number of studies. On the MNIST database, deep belief networks (DBN) with three layers and a greedy method are explored and found to be 98.75 percent accurate[11]. Simard et al. established a modular network topology for visual text processing in 2003, laying the foundation for neural network training to become less complicated[15].Pham et al. improved the performance of duplicate neural networks (RNNs) in the acceptance of unrestrained handwriting using a typical drop-out strategy[14].The convolutional neural network aims to revolutionize handwriting analysis and makes it easier to create works of art in this sector. In ICDAR 2003 and Street View Text,Wang et al. proposed a novel approach to end-to-end text recognition utilising CNN with several layers and found satisfactory accuracy in data storage. Shi et al. recently termed the common duplicate neural network after combining the strengths from both the deep CNN and the duplicate neural network (RNN)(CRNN)[2,17].

Researchers determined that CRNN was more accurate than standard strategies for text analysis. Badrinarayanan et al. pushed for the creation of a semantic separation of characters using deep convolutional networks. SegNet is a class design that includes a decoding and coding network, and a smart pixel segmentation layer. The suggested technique aggregated the maximum of the feature map while examining and determining the best output. This strategy is also evaluated and compared to based on current tactics as well as internal knowledge. CNN has proved its incredible capacity to recognise fine handwriting language, Telugu character identification, Urdu handwriting recognition, handwriting recognition in Indic scripts, and Chinese handwritten text recognition.

Gupta et al. have created a hybrid framework for determining the most competent local regions from an image frame. Separate English handwritten numbers, i.e. MNIST pictures, along with three other well-known Indic texts, notably Bengal manuscripts and Devanagari handwriting, were used to test the work[1]. The approach implemented 95.96 percent recognition accuracy by using characteristics collected from a convolutional neural network in their model. Nguyen et al. employed CNN's multidimensional scale to extract the geographical aspects of the handwritten statistical field in their study (HME). For HME imagery, local features and metadata were used to merge them. Activity viewed the CROHME database, which is a greater database. They also drew the conclusion that segmentation may be developed by incorporating a complete integration process and global listening into CNN training.

For speedier text / word space and recognition in historical documents, Ziran et al. designed a R-CNN-based methodology. The studies look at these in-depth research techniques in Gutenberg's Bible pages[20]. Ptucha et al study cleverly overcomes the problem of handwriting recognition by presenting an intellectual algorithmic (ICR) system based on a regular neural network. The study has shown considerable gains when evaluated on French-based dictionary data sets as well as English-based IAM data.

1.2 Objective

The purpose is to test the MNIST database's reliability. Handwritten Digital Recognition (HDR) is a technique for recognizing handwritten numbers. It's already commonly utilized in automated bank check processing, postal address computation, and mobile phone processing, To verify the database's correctness, we employ a regression analysis in this work.

LITERATURE SURVEY

The recognition of digital manuscripts has attracted much research and analysis because there are many places where handwritten digital manuscripts still exist, for example, filtering default characters in the post, processing bank checks, or historical documents.

Unfortunately, such writing texts are very difficult to see even by humans. Therefore, a system that can facilitate the automatic recognition of manuscripts may be extremely desired.

Handwritten digital recognition is a key component of optical character recognition (OCR) and might be considered a sub-problem. It refers to a computer's willingness to handle and decipher handwritten numbers from a variety of sources, including images, text, and other devices. Despite the development of a number of recommended system strategic planning strategies in this field, the absolute correctness of the pattern predictions remains in question. The following are some of the proposals.

In the paper “HandWritten Digit Classification using Machine Learning Models” published by “Vidushi Garg”[4]. To forecast the accuracy of the MNIST dataset, the paper used three models: support vector machine, logistic regression and random forest classifier. He compared the accuracies of the above three models using the confusion matrices, precision, F1score and recall. The accuracies are as follows: SVM-98%, LRM-86%and RFC-96%. This paper was published in November 2019.

In the paper “Handwritten Digit Recognition System based on LRM and SVM Algorithm” published by Hafiz ahmed,Ishraq Alam and Md.Manirul Islam, [5] They found the accuracy of the dataset using the supervised machine learning algorithms such as Logistic Regression model and Support Vector machine of accuracies 92.8% and 97.83% respectively. This paper was published in January 2019.

From the paper “Recognition of Handwritten Digits using Convolutional Neural Networks” published by Md. Anwar Hossain and Md. Mohon ali, They found the accuracy of Mnist dataset using the convolutional neural networks of 7 layers and the accuracy is 99.5%.[6] This paper was published in 2019

As a result, comparing relevant techniques has become complex and appears to be tough to select the best because their success is dependent on data. It is determined by a number of parameters, including high accuracy, short operation times, low memory usage, and the amount of training time required.Identifying handwritten digits is a critical challenge in visual recognition, and it has been used as a test for character segmentation theories and machine learning techniques to several years.

SOFTWARE SPECIFICATIONS

Software used:

Operating System: Windows 7

S/W Tool: Pandas, SKlearn, Matplotlib, Numpy

3.1 NUMPY:

NumPy is the most important Python package for numerical computation. It's a Python library, after all.

It consists of multi array objects, and also some derived objects and infrastructure needs routines for performing fast array functions, such as manipulation of shape, logical, mathematical, discrete Fourier transforms, selecting input and output, basic linear algebra, and fundamental statistical operations., random simulation and many more[10].

The ndarray object, which is at the heart of the Numpy package, gathers arrays of n dimensions of related data types and performs various operations in generated code.

3.2 PANDAS:

Pandas is an open source Python library that uses advanced analytical features that give great data performance and tool manipulation. Panel Data - Econometrics from Multidimensional Data is how Pandas got its name[3].

When developer Wes McKinney needed greater, modular data processing techniques, he started designing pandas in 2008. Python was commonly used for data and prepared to make to the Pandas. There was only a smidgeon of help with data analysis. This issue was fixed by pandas. Regardless of the source, we may execute five standard phases in data processing and analysis with Pandas: upload, prepare, manage, model, and analyses.

Python with Pandas is utilized in a wide range of business and educational settings, such as finance, marketing, mathematics,etc.

Pandas' key characteristics include:

- A quick and efficient data frame with an automatically generated and tailored index.
- Tools for loading recollection data objects from various file formats.
- Data synchronisation and comprehensive data loss monitoring.
- Cutting, indexing, and reset of large data sets based on labels.
- The data structure's columns can be added or removed.
- Gather statistics on integration and transformation.
- Data integration and high-performance integration.
- Operating Time.

3.3 SKlearn

Scikit-learn is largely built in Python, and it makes extensive use of the previous package for significantly high algebra and equivalent member functionality. In particular, cyton is used to write some fundamental techniques that boost performance.[7] The Cython wrapper for LIBSVM uses vector support machines; LIBLINEAR shows specific decomposition and vector support mechanisms with similar packaging. This might not be possible to upgrade these methods with Python in such instances.

Numerous similar Python libraries, such as matplotlib, NumPy of the same vectorization, pandas data frames, scipy, and others, work well with Scikit-learn.

3.4 Matplotlib

Numpy is a numerical addition to Matplotlib, a Python language manipulating package. Uses conventional GUI technologies like tkinter, wxpython, Qt, or GTK to provide AMP-focused site input applications.[13] There is also a cutting-edge "pylab" system built on a machine (similar to OpenGL) that is supposed to be very comparable to MATLAB, albeit its use is not advised. Matplotlib is used by Scipy.

John H. Dunter was the designer of Matplotlib. It has had a thriving reference implementation since then, and it is still available under the BSD licence. Plotibib's lead engineer, Michael Droettboom, was selected shortly before John Hunter's death in August 2012, and was re-joined by Thomas Caswell.Matplotlib 2.0.x Python versions 2.7 to 3.6 are supported. Matplotlib was the first library to support Python 3.

Pyplot is a MATLAB interface for the Matplotlib library. Matplotlib was created to be used as a MATLAB replacement, with the option to use Python and the potential to be open source.

3.5 MNIST

The Modified National Institute of Standards and Technology database (MNIST) is a massive handwritten database that is commonly used to train image analysis systems. In the field of machine learning, the dataset is also commonly utilised for training and testing. samples from NIST's original data sets were "re-mixed" . The designers recognised that the NIST training database was not well-suited because the test data was gathered from Children in the u.s., but the NIST training database was gathered from employees of the American Census Bureau[16]. In addition, NIST black and white photos were anti-aliased and resized to fit a size of 28x28 pixel box, resulting in grayscale values[9].

There are sixty thousand training images and ten thousand test images in the MNIST database. A portion of the training set and a portion of the test set came from the NIST training data, while the rest of the training and test data came from NIST. Some of the tried approaches were retained on the list by the actual developers. They found an error rate of 0.8 percent using a support-vector machine in their original study. MNIST, for example, provides 2.4 lakh training images as well as 40 thousand digit testing sets and characters.

3.6 Machine Learning

Machine learning is concerned with the creation of computer programs that can learn to adapt and learn in response to new input[18].

It's also known as Predictive Analytics or Statistical Learning, and it's a research area in the field of statistics, big data, and software engineering.

DESIGN AND IMPLEMENTATION

4.1Description of the data set

The MNIST dataset, a subset of a larger set MNIST, is a database of 70,000 handwritten digits, divided into 60,000 training examples and 10,000 test samples.Images in the MNIST database exist in the form of identical members of size 28x28 values representing the image and its labels[1]. This is also the case in the case of test images. The gray values of each pixel are coded in this function at intervals of [0,255], using a value of 0 for white and 255 pixels for blacks.

DIGITS	DATA SPLITTING		SUBTOTAL
	TRAINING	TESTING	
0	5923	980	6903
1	6742	1135	7877
2	5958	1032	6990
3	6131	1010	7141
4	5842	982	6824
5	5421	892	6313
6	5918	958	6876
7	6265	1028	7293
8	5851	974	6825
9	5949	1009	6958
TOTAL	60,000	10,000	70,000

Fig 1: Dataset

4.2 Data preprocessing

An important point for managing high performance in the learning process is the construction of a useful training data set. The 70000 different patterns contained on the MNIST website may seem like a kind of set, but the evidence shows that standard learning algorithms run into a major problem with approximately a set of 100 or more samples. Therefore, a specific strategy is needed to increase technical training and diversity.Typical actions consist of geometric modifications such as displacement, rotation, measurement and other distortions.

The proposed changes to this paper is to make digitization of handwritten digits require images at the binary level. The process of binarization assumes that images have two categories of pixels: the front (or white pixels, large in size, i.e., equal to 1) and the back (or black pixels with a minimum intensity, i.e., equal to 0). The goal of the method is to separate all pixels with values above the given threshold white and all other black pixels.

That, given the limit value t and image X. The pixels denoted as x(i, j), the binarized image X_b with elements $x_b(i, j)$ is obtained as follows:

If $x(i, j) > t$ $x_b(i, j) = 1$

Else

$x_b(i, j) = 0$

After that, the main problem in binarization is how to choose the right limit t of a given image. We see that the composition of any object in the image is sensitive to the difference in the boundary value, and is more sensitive to the condition of the handwritten number. Therefore, we consider that a binary handwritten number is better identified by designation if its trace is complete and continuous, this is the situation we use in the end, its most important choice.

IMPLEMENTATION

5.1 Logistic Regression Model

Logistic Regression is a Machine Learning classification algorithm with which the probability of a categorical dependent variable can be determined. In logistic regression, the dependent variable is a binary variable that contains data coded as 1 (yes, success, etc.) or 0 (no, failure, etc.). This model predicts probability of Y as a function of X.

Thereafter,the training setup data is entered by logging into the Logistic Regression model for the separator to be trained. The estimated label corresponds to the real one to determine the accuracy of the trained separator.Once the training is done, the test data is provided with a label predictor and the accuracy of the test data is obtained.Then,generates a confusion matrix that provides a gap between real data and predicted data. Using the confusion matrix, practical steps such as accuracy, memory and f1 score can be calculated. Using Logistic Regression, 95.4% accuracy is obtained from test data

The Test Data Set obtained accuracy of 95.4% using the Logistic Regression on MNIST data set.

5.2 Flow Chart

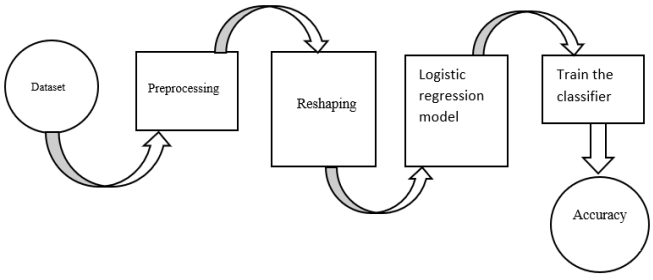


Fig 2: Flow Chart

Steps to build a logistic regression model to find the accuracy of the dataset.

1. Fetch the dataset from MNIST
2. Check the size of the dataset.
3. Check for Null Values(if any).
4. Import matplotlib package for visualization
5. Reshaping the data in order to plot the data.Reshaping is done for plotting.
6. Split the data into testing data and training data
7. In the 70000 images, 60000 images are considered as training data and the remaining are testing data.
8. Import logistic Regression model from Sklearn Create a classifier as "clf".
9. Fit the training data into the classifier
10. Predict the test data using the training data.
11. Now find the accuracy of the predicted data as p.mean().

These are the steps of algorithm

RESULT AND DISCUSSION

6.1 Result

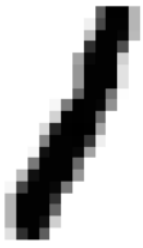
A hand-written digit recognition system is used to visualize artificial neural networks. It is hugely applicable in the automatic processing of postal addresses, bank cheques in mobile phones.

After preprocessing the data, training and evaluating the model, we have achieved an accuracy of 95.4 .

We have to reshape the data before splitting it in order to plot it .Fig shows the image of data in the dataset after reshaping.

```
a[4561] #after reshaping and plotting
```

```
(-0.5, 27.5, 27.5, -0.5)
```



```
b[4561]
```

```
'1'
```

Fig 3: Pixel image

Here this is the pixel image that is stored in a[4561] and b[4561] contains "1" .

```
p.mean()
```

```
0.9548333333333333
```

Fig 4: Accuracy

Fig.4 is the image of accuracy obtained by employing a logistic regression model on the MNIST dataset.

S.No.	Algorithm	Accuracy of Correctly Classification
1	Multilayer Perceptron	90.37
2	Support Vector Machine	87.97
3	Random Forest	85.75
4	Bayes Net	84.35
5	Random Tree	85.6
6	Proposed method	95.4

Table:1 Comparison with existing work [8]

Compared to the existing models mentioned in the above table,our proposed method's accuracy is significant

This algorithm allows models to be easily updated to reflect new data, unlike decision trees or vector support machines.Revision can be made using a stochastic gradient decrease.

In a low-density database with a sufficient number of training examples, the reverse of order of things is not usually excessively balanced.

Logistic Regression shows that it works best when the database has sequentially fragmented features.

Training features are known as independent variants. Logistic Regression requires a balance or lack of quantity between independent variables. This means that if two independent variables have a high interaction, only one of them should be used. Repetition of information can lead to improper training of parameters (instruments) during the reduction of work costs. Multicollinearity can be removed using size reduction techniques.

CONCLUSION AND FUTURE SCOPE

Conclusion

Therefore, digital recognition using machine learning by modeling systems is proven to be very effective up to 95 percent when some algorithms are not very reliable.

Logistic regression model is quick and efficient

This model can be used to process bank checks, posting data entry, etc.

- Extend the model to work in the MNIST database.
- And to find handwritten numbers.

REFERENCES

[1] Gupta, A.; Sarkhel, R.; Das, N.; Kundu, M. Multiobjective optimization for recognition of isolated handwritten Indic scripts. Pattern Recognit. Lett. 2019, 128, 318–325.

[2] Shi, B.; Bai, X.; Yao, C. An End-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE Trans. Pattern Anal. Mach. Intell. 2017, 39, 2298–2304.

[3] https://www.tutorialspoint.com/python_pandas/python_pandas_quick_guide.htm

[4] Vidushi Garg, "HandWritten Digit Classification using Machine Learning Models",International Research Journal of Engineering and Technology (IRJET),Vol. 06 no. 11 ,Nov 2019

[5] Hafiz ahmed,Ishraq Alam and Md.Maniru, Islam " Handwritten Digit Recognition System based on LRM and SVM Algorithm",International Conference on Engineering Research and Education School of Applied sciences & Technology, SUST, Sylhet.

[6] Md. Anwar Hossain and Md. Mohon ali on "Recognition of Handwritten Digits using Convolutional Neural Networks",Global Journal of Computer Science and Technology: D Neural & Artificial Intelligence ,Volume 19 Issue 2 Version 1.0 Year 2019.

[7] https://www.tutorialspoint.com/scikit_learn/index.htm

[8] M Shamim, Mohammad Badrul Alam Miah, Angona Sarker, Masud Rana & Abdullah Al Jobair, Handwritten Digit Recognition using Machine Learning Algorithms, Global Journal of Computer Science and Technology: D Neural & Artificial Intelligence, Volume 18 Issue 1 Version 1.0 Year 2018.

[9] Handwritten Digit Recognition using Convolutional Neural Networks in Python with Keras.

[10] https://en.wikipedia.org/wiki/NumPy

[11] Hinton, G.E.; Osindero, S.; Teh, Y.W. A fast learning algorithm for deep belief nets. Neural Comput. 2006, 18, 1527–1554.

[12] Plamondon, R., & Srihari, S. N. Online and off-line handwriting recognition: a comprehensive survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(1), 63-84,

[13] https://en.wikipedia.org/wiki/Matplotlib

[14] Pham, V.; Bluche, T.; Kermorvant, C.; Louradour, J. Dropout improves recurrent neural networks for handwriting recognition. In Proceedings of the 14th Int. Conf. on Frontiers in Handwriting Recognition, Heraklion, Greece, 1–4 September 2014.

[15] Simard, P.Y.; Steinkraus, D.; Platt, J.C. Best practice for convolutional neural networks applied to visual document analysis. In Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003), Edinburgh, UK, 3–6 August 2003Simard, P.Y.; Steinkraus, D.; Platt, J.C. Best practice for convolutional neural networks applied to visual document analysis. In Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR 2003), Edinburgh, UK, 3–6 August 2003.

[16] https://en.wikipedia.org/wiki/MNIST_datab

[17] Wang, T.; Wu, D.J.; oates, A.; Ng, A.Y. End-to-end text recognition with convolutional neural networks. In Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), Tsukuba, Japan, 11–15 November 2012.

[18] https://www.geeksforgeeks.org/machine-learning/

[19] Andrea Giuliodori, Rosa Lillo and Daniel Peña paper on "HANDWRITTEN DIGIT CLASSIFICATION"

[20] Ziran, Z.; Pic, X.; Innocenti, S.U.; Mugnai, D.; Marinai, S. Text alignment in early printed books combining deep learning and dynamic programming. Pattern Recognit. Lett. 2020, 133, 109–115.

[21] Goel, A., Bhujade, R.K. "A functional review, analysis and comparison of position permutation based image encryption techniques" (2020) International Journal of Emerging Technology and Advanced Engineering, 10 (7), pp. 97-99.

[22] Goel, A., Bhujade, R., "A functional review of image encryption techniques", International Journal of Scientific and Technology Research, 2019, 8(9), pp. 1203–1205

[23] Alanazi, B.S., Rekab, K. "Fully sequential sampling design for estimating software reliability"(2020) International Journal of Emerging Technology and Advanced Engineering, 10 (7), pp. 84-91.

[24] Flesch, B.F., Tedeschi, I., De Figueiredo, R.M., Prade, L.R., Da Silva, M.R."A functional safety methodology based on IEC 61508 for critical reliability FPGA-based designs" (2020) International Journal of Emerging Technology and Advanced Engineering, 10 (7), pp. 12-19.

[25] Laber, J., Thamma, R. "MATLAB simulation for trajectory/path efficiency comparison between robotic manipulators" (2020) International Journal of Emerging Technology and Advanced Engineering, 10 (11), pp. 74-88.

[26] Meshram, S., Kumar, S., Shukla, S. "Enhanced robust and invisible of digital image using discrete cosine transform technique and binary shifting technique" (2020) International Journal of Emerging Technology and Advanced Engineering, 10 (10), pp 113-118.