Plagiarism Checker X - Report

Originality Assessment

8%

Overall Similarity

Date: Mar 23, 2024
Matches: 266 / 2093 words
Sources: 16

Remarks: Moderate similarity detected, consider enhancing the document if necessary.

Verify Report:
Scan this QR Code

v 8.0.1 - WML 4 FILE - MICRO PROJECT REPORT PDF

HANDWRITTEN DIGIT RECOGNITION WITH CONVOLUTIONAL NEURAL NETWORKS

A Micro Project Report Submitted by GOSANGI DEEPTHI Reg.no:992200416 99 B.Tech - CSE, AIML Kalasalingam Academy of Research and Education (Deemed to be University) Anand Nagar, Krishnankoil- 626 126 FEBRUARY - 2024

SCHOOL OF COMPUTING i DEPARTMENTOF COMPUTERSCIENCEAND ENGINEERING BONAFIDE CERTIFICATE Bonafide record of the work done by GOSANGI DEEPTHI - 99220041699 in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in AIML of 15 the Computer Science and Engineering, during the Academic Year Even Semester (2023-24) Mr.M.Jafar Sathick Ali Mr.N.R.Sathish Kumar Project Guide Faculty Incharge [Designation] Associate Professor [DepartmentName] CSE KalasalingamAcademy of KalasalingamAcademy 12 of Research and Education Research and Education Krishnan kovil- 626126 Krishnan kovil- 626126 Mr.N.R.Sathish Kumar Evaluator Associate Professor CSE KalasalingamAcademy of Research and Education Krishnan kovil- 626126

ABSTRACT This project focuses on developing a 11 convolutional neural network (CNN) for handwritten digit recognition using the MNIST dataset. Utilizing PyTorch, 7 a deep learning framework, the goal is to create an accurate model capable of classifying hand-written digits effectively. The CNN architecture incorporates convolutional layers, batch normalization, 10 and fully connected layers for feature extraction and classification. Training is conducted using the Adam optimizer and Cross EntropyLoss function, optimizing for accuracy over multiple epochs. Training progress is monitored, and performance metrics are visualized using Matplotlib. The model's effectiveness is evaluated on a separate test dataset, and predictions are made on sample images to assess real-world performance. This project showcases the practical application of deep learning in solving handwritten digit recognition tasks with potential applications in various domains such as document processing and educational assessment. 3

CONTENTS

CHAPTER-I 1.INTRODUCTION: Handwritten digit recognition is a foundational problem in machine learning, with applications ranging from document processing to educational assessment. This project focuses on developing a Convolutional Neural Network (CNN) solution for accurate digit recognition, using the MNIST dataset as a benchmark. Leveraging PyTorch, a powerful deep learning framework, the project aims to showcase the effectiveness of CNNs in addressing real-world challenges. By exploring advancements in deep learning and leveraging PyTorch's capabilities, the project aims to construct a robust model capable of achieving high accuracy on the MNIST dataset. Furthermore, the project seeks to highlight the broader implications of CNN technology, including its potential applications in various domains. Through rigorous experimentation and evaluation, the project aims to contribute to advancements in pattern recognition and machine learning. 5

CHAPTER-II 2. OBJECTIVE: Develop Convolutional Neural Networks (CNNs) to recognize handwritten digits, utilizing the MNIST dataset as a foundational benchmark. This involves designing CNN architectures tailored to digit recognition tasks, preprocessing data for model input, and implementing PyTorch-based CNN models for training. Train CNN models with a primary objective of achieving precise classification of hand-written digits, emphasizing not only high precision but also robustness and generalization across diverse handwriting styles and variations. This entails optimizing model hyperparameters, selecting appropriate loss functions, and fine-tuning training procedures to maximize performance. Thoroughly evaluate the performance of trained CNN models on the

MNIST test dataset, employing rigorous [8] metrics such as accuracy, precision, recall, and F1-score to assess their effectiveness in digit recognition tasks. This involves conducting extensive testing and validation, analyzing model predictions, and comparing results with ground truth labels to quantify performance. □ Conduct a comprehensive comparative analysis of various CNN architectures, exploring different network depths, layer configurations, and optimization techniques to identify the most efficient and accurate models for handwritten digit recognition. This includes experimenting with different convolutional layers, pooling strategies, activation functions, and regularization methods to optimize model performance and generalization. 6

CHAPTER-III 3.1 SOFTWARE TOOLS: A [3] software tool is a computer program that is used to create, debug, maintain, or otherwise support other programs and applications. Software tools are also commonly referred to as software programming tools. Examples of software tools include: Code editors: Jupyter, Google Colab Packages: □ Python □ PyTorch □ Keras □ NumPy □ OpenCV □ Matplotlib 3.2 USAGE OF TOOLS: □ Jupyter is an open-source interactive development environment (IDE) for Python, R, Julia, and other programming languages. It allows users [2] to create and share documents that contain live code, equations, visualizations, and narrative text. Jupyter is typically used for data science, scientific computing, and machine learning. Jupyter is made up of two main components: the Jupyter Notebook and JupyterLab. Jupyter Notebook: The Jupyter Notebook is a web-based application that allows users to create and share documents that contain live code, equations, visualizations, and narrative text. The Jupyter Notebook is the most popular way to use Jupyter. 7

•Data analysis: Jupyter [5] can be used to clean, analyze, and visualize data. •Machine learning: Jupyter can be used to train and evaluate machine learning models. •Scientific computing: Jupyter can be used to perform scientific computations, such as numerical simulations and data modeling. •Education: Jupyter can be used to create interactive educational materials, such as tutorials and textbooks. Python: Python is a high-level, interpreted programming language known for its simplicity and readability. It offers extensive libraries and frameworks for various tasks, including data analysis, machine learning, and web development. PyTorch: PyTorch is an open-source deep learning framework developed by Facebook's AI Research lab. It provides a flexible and dynamic computational graph, making it suitable for research prototyping and production deployment. Keras: Keras is a high-level neural networks API written in Python, which can run on top of TensorFlow, Theano, or Microsoft Cognitive Toolkit (CNTK). It provides a user-friendly interface for building and training deep learning models. NumPy: NumPy is a fundamental library for numerical computing in Python. It provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays efficiently. OpenCV: OpenCV (Open Source Computer Vision Library) is an opensource computer vision and machine learning software library. It provides a wide range of functions for image processing, including image manipulation, feature detection, and object recognition. Matplotlib: Matplotlib is a plotting library for Python that provides a MATLAB-like interface for creating static, interactive, and animated visualizations. 7 It is widely used for data visualization and analysis in various fields, including scientific computing and machine learning. 8

CHAPTER-IV 4.LITERATURE SURVEY S . n o Author Paper Title Year Abstract 1 S M Shamim Handwritten Digit Recognition using Machine Learning Algorithms March 2018 This paper explores machine learning algorithms such as Multilayer Perceptron and Support Vector Machine in WEKA software to achieve reliable recognition of handwritten digits, critical for applications like mail sorting and data entry. 2 Tsehay Admassu Assegie 9 Handwritten digits recognition with decision tree classification: a machine learning approach May 2019 This paper applies decision tree classification to recognize handwritten digits using the Kaggle digits dataset, evaluating the model's accuracy for digits from 0 to 9. 3 Birjit Gope 1 Handwritten Digits Identification Using Mnist Database

Via Machine Learning Models October 2020 This paper explores diverse machine learning algorithms, including SVM, MLP, and Decision Trees, to achieve accurate offline recognition of hand-written digits, crucial for applications like postal codes and check processing. 4 Preetha S Machine Learning for Handwriting Recognition May 2022 This paper discusses the application of machine learning in recognizing hand-written characters through pattern recognition, exploring methods to extract hidden information from image data to predict outputs for unknown data. 5 Mayank Sharma [1] Handwritten Digit Recognition Using Machine Learning March 2023 In a digit-centric world, automation and digitalization demand accurate digit recognition. This project achieves 98.40% accuracy using [7] Convolutional Neural Networks (CNNs), streamlining tasks and enhancing efficiency in various applications. 9

CHAPTER-V 5. TIMELINE OF WORK PROPOSAL: Week 1 : 1.Researching different methods [1] for Handwritten Digit Recognition, identifying potential technologies and equipment to use, and determining the specific goals and objectives of the project. 2. Conduct a literature review to gather information on existing Handwritten Digit Recognition systems, identify potential technologies and equipment, and determine the specific goals and objectives of the project. Week 2 : 1.Clearly define the problem and aims to solve the problem using various techniques. 2.Outline specific goals and aims to achieve solution for them. 3.Creating block diagram for the given problem statement. Week 3 : 1.Finding the methods and techniques to achieve the problem statement. 2.Making ppt and knowing required components related to problem statement. 3.Knowing uses of component which are used in making the prototypes. Week 4 : 1.Completed the software part and making ppt and report. 2.Submission of ppt and report to mentor 10

CHAPTER-VI 6.1 ALGORITHMS USED: □ Convolutional Neural Networks (CNNs) are deep learning models designed for processing structured grid-like data, such as images. They consist of multiple layers, including convolutional [6] layers, pooling layers, and fully connected layers, which extract hierarchical features from input data. □ CNNs leverage convolutional operations to [4] learn spatial hierarchies of features by applying filters across input images. These filters, also known as kernels, slide over input images, detecting patterns such as edges, textures, and shapes, which are then progressively aggregated and processed in subsequent layers. □ Pooling layers in CNNs downsample feature maps, reducing spatial dimensions while retaining essential information. Common pooling operations include max pooling and average pooling, which help to increase the model's translation and rotation invariance while reducing computational complexity. □ Fully connected layers, often found at the end of CNN architectures, perform classification tasks by combining high-level features learned from previous layers. These [4] layers connect every neuron in one layer to every neuron in the next layer, enabling the network to make predictions based on learned representations. □ CNNs are widely used in various computer vision tasks, including [6] image classification, object detection, and image segmentation, due to their ability to automatically learn hierarchical representations from raw data. Their hierarchical structure enables them to capture both low-level features (e.g., edges) and high-level semantic information (e.g., object classes) from input images. □ Training CNNs typically involves forward and backward passes through the network, where input images are fed into the network, and gradients are computed using backpropagation to adjust model parameters (e.g., weights and biases) iteratively. This process aims to minimize a predefined loss function, optimizing the network's performance on a given task.

6.2 STEP BY STEP PROCESS: Data Preparation: Load MNIST dataset, normalize pixel values, reshape images, and convert labels to tensors. Model Architecture: Define a CNN with convolutional, pooling, and fully connected layers. Specify activation functions and regularization techniques. Training Setup: Choose hyperparameters (learning rate, batch size, epochs), optimizer (e.g., Adam), and loss function (e.g., cross-entropy).Split dataset [5] into training and validation sets. Model Training: Iterate over training dataset for each epoch.Perform forward pass, compute loss, backpropagate, and update model parameters. Evaluation: Validate model on validation dataset.Compute evaluation metrics 8 (accuracy, precision, recall) and visualize training progress. Inference: Use trained model to predict labels for new images.Preprocess input images, pass through model, and obtain predicted labels. Output Visualization: Visualize predictions alongside original images.Display actual vs. predicted labels and optionally plot confusion matrix. 12

CHAPTER-VII 7.1 IMPLEMENTATION SCREENSHOTS: 13

14

7.2 OUTPUT: 15

CHAPTER-VIII 8.1 RESULTS: Model Performance: The basic CNN model achieved an average training accuracy of approximately 99.1% over 10 epochs, with a corresponding loss of 0.0224. The accuracy and loss were visualized over epochs, indicating a consistent increase in accuracy and decrease in loss over training iterations. Sample Prediction: For a randomly selected test image (index 267), the model correctly predicted the digit label. Actual Label: 4, Predicted Label: 4 8.2 REFERENCES: • [1] 16 Machine Learning for Handwriting Recognition Preetha Sa *, Afrid I Mb , Karthik Hebbar Pc , Nishchay S Kd a,b,c,d Department of ISE,B.M.S. College of Engineering/ VTU, India, {preetha.ise,1bm16is006, 1bm16is044, 1bm16is060} aEmail: preetha.ise@bmsce.ac.in • [2] 1 Handwritten Digit Recognition using Machine Learning Algorithms By S M Shamim, Mohammad Badrul Alam Miah, Angona Sarker, Masud Rana & Abdullah Al Jobair • [3] Handwritten digits recognition with decision tree classification: a machine learning approach Tsehay Admassu Assegie, Pramod Sekharan Nair Department of Computing Technology, College of Engineering and Technology, Aksum University, Ethiopia • [4] 1 Handwritten Digits Identification Using Mnist Database Via Machine Learning Models To cite this article: Birjit Gope et al 2021 IOP Conf. Ser.: Mater. Sci. Eng. 1022 012108 • [5] Handwritten Digit Recognition Using Machine Learning,Mayank Sharma, Pradhyuman Singh Sindal & M. Baskar ,Conference paper,25 March 2023. 16

17 CHAPTER-IX 9. COURSE CERTIFICATE

Sources

1	https://www.researchgate.net/publication/348591122_Handwritten_Digits_Identification_Using_Mnist_Database_Via_Machine_Learning_Models_Handwritten_Digits_Identification_Using_Mnist_Database_Via_Machine_Learning_Models INTERNET 3%
2	https://www.geeksforgeeks.org/jupyter-notebook/ INTERNET 2%
3	https://www.geeksforgeeks.org/what-is-software-tool/ INTERNET 1%
4	https://medium.com/@koushikkushal95/understanding-convolutional-neural-networks-cnns-in-depth-d18e299bb438 INTERNET 1%
5	https://www.geeksforgeeks.org/train-a-deep-learning-model-with-pytorch/ INTERNET 1%
6	https://www.nature.com/articles/s41598-024-51258-6 INTERNET 1%
7	https://www.geeksforgeeks.org/deep-learning-with-pytorch-an-introduction/ INTERNET 1%
8	https://medium.com/@zhonghong9998/evaluating-model-performance-a-comprehensive-guide-6f5e7c11409f INTERNET 1%
9	https://www.researchgate.net/publication/336888176_Handwritten_digits_recognition_with_decision_tree_classification_a_machine_learning_approach INTERNET 1%
10	https://link.springer.com/chapter/10.1007/978-3-031-28387-1_22 INTERNET <1%
11	https://www.geeksforgeeks.org/handwritten-digit-recognition-using-neural-network/ INTERNET <1%
12	https://kalasalingam.ac.in/course/b-tech-computer-science-and-engineering/ INTERNET <1%
13	https://sist.sathyabama.ac.in/sist_naac/documents/1.3.4/b.e-ece-batchno-187.pdf INTERNET <1%

14	https://medium.com/@cemanahuacstrategies/pytorch-a-powerful-deep-learning-framework-fc1d1e3b0f65 INTERNET <1%
15	https://manipal.edu/mit/program-list/btech/BTech-Computer-Science-and-Engineering-Artificial-Intelligence-and-Machine-Learning.html INTERNET <1%
16	https://www.semanticscholar.org/paper/Machine-Learning-for-Handwriting-Recognition-Preetha-Afridi/9581e4817ffefcbb38844e62a5c4ebd9e8184678 INTERNET <1%

EXCLUDE CUSTOM MATCHES	ON
EXCLUDE QUOTES	OFF
EXCLUDE BIBLIOGRAPHY	OFF