Report On

Application Of Deep Learning Models on MNIST Dataset

Name: Guduru Varshitha

ID: 811289917

The study evaluates various neural network topologies for image classification tasks using the MNIST dataset. The effectiveness and efficiency of several model types, such as the deep CNN, batch normalization CNN, regularization (L2) CNN, easy Multilayer Perceptron (MLP), and convolutional neural network (CNN), are the main issues that are explored.

Key findings:

Evaluation of Various Architectures' Performance: Simple feedforward networks might not be appropriate for image classification applications, as seen by MLP's comparatively poor performance when compared to CNN-based models. The fact that CNN and its variations perform better than MLP indicates that convolutional layers are useful for capturing spatial information in pictures.

Role of Complexity in Model Performance: As demonstrated by the CNN with regularization (L2) and deep CNN models, models with additional layers and parameters typically perform better. Among the models tested, Deep CNN gets the highest accuracy because to its increased complexity and several convolutional layers.

Efficiency of Regularization and Normalization Techniques: CNN with regularization (L2) and CNN with batch normalization both outperform simple CNN, demonstrating the advantages of these methods for regularizing and stabilizing training.

Model Generalization vs. Overfitting: One type of regularization technique that helps lower overfitting and increase the precision of generalization on data without observations is called L2 regularization**.**

Trade-off Between Model Complexity and Performance: Although deeper models with more convolutional layers typically perform better, they may also take longer to train and need more processing power. All things considered, the assessment emphasizes how crucial architecture selection, regularization, and normalization strategies are to creating neural networks that perform well on picture classification tasks.

Introduction:

Sorting images is a basic task in computational vision that can be used for everything from disease diagnosis to autonomous driving. Applications such as object recognition, image analysis, and image retrieval depend on accurate picture classification. The need for accurate and efficient image classification techniques has significantly increased due to the present exponential growth of picture data. The state-of-the-art technique for image classification is neural networks, and more especially Convolutional Neural Networks (CNNs), which can automatically learn structural features based on raw picture statistics. However, the choice of hyperparameters and neural network design may have a significant effect on these models' performance. Therefore, it is essential to evaluate and compare different neural network topologies in order to determine the most effective techniques for photo identification tasks.

My objective in this study is to compare and analyse the performance of several neural network designs for image classification tasks, using the MNIST data set as a reference. MNIST is a well-known dataset that is composed of Contains 28x28 grayscale images of numbers 0–9 written by hand. MNIST is a widely used average for evaluating and comparing different image classification methods, despite being a rather simple dataset. The performance of several architectures, including deep CNN, CNN with regularization (L2), CNN with batch normalization, basic CNN, and simple Multilayer Perceptron (MLP), will be compared. By evaluating these designs, I expect to get greater insight into their efficacy, robustness, and universality.

Understanding the performance advantages of different neural network designs is essential for a number of reasons. First of all, it helps experts and students choose the ideal model structure using benchmarks for specific tasks, such as model depth, accuracy, and computation speed. Additionally, it promotes deep learning techniques by identifying the best practices and practical designs. In the end, improved picture modelling approaches have significant implications for number of industries, including robotics, security, and health. By pushing the limits of picture classification, I can increase the reliability and accuracy of computer vision systems, increasing their usefulness and effectiveness in daily life.

Current Research: Deep Learning for MNIST Classification:

The MNIST data collection remains a vital resource for evaluating visual extraction techniques, particularly neural network instances. The current research aims to push the boundaries of accuracy and dependability while

looking into new instructional strategies and structures.
Using the MNIST dataset, the research paper [1] explores ways to improve neural networks' adversarial defense, with an emphasis on image identification. The authors suggest combining gradient modification with feature masking to increase network resistance to hostile attacks. They use a traditional neural network design to assess this method's efficacy and contrast it with a baseline model that does not use feature masking.

Findings:

Performance of the Baseline Model: On the MNIST dataset, the baseline model's high-test accuracy was 98%. Nevertheless, it demonstrated little defense against hostile strikes, with accuracy falling to 60% under FGSM attacks.

Role of Feature Masking: Accuracy and robustness were shown to be traded off in models that included feature masking. The accuracy decreased as the percentage of feature masking rose, but an improvement in the ability to withstand attacks

Accuracy vs. Robustness Challenge: A 96% accuracy rate and 75% resilience to attacks were achieved with a 10% masking ratio. 94% accuracy and 80% robustness were achieved with a 30% masking. A 50% concealment threshold reached the robustness peak at 85% with an accuracy of 92%.

Effectiveness of Feature Masking: The outcomes support feature masking's ability to strengthen hostile defense. Resilience and accuracy must be balanced critically, emphasizing the significance of figuring out the best masking ratio.

Future Exploration Areas: In order to increase the range of neural network security against hostile threats, the study recommends more research into improved masking techniques and their combination with other defensive tactics.
In conclusion, the study shows that adversarial resilience can be greatly increased without compromising accuracy by combining feature masking with neural network training. This approach provides a practical strategy for developing neural network architectures that are more optimal and dependable, which is a significant advancement in AI security.

The study presents an expedited genetic method for training deep convolutional neural networks (CNNs).Through the introduction of parent-child links using this method, Future generations can inherit knowledge from their forefathers. This inheritance approach seeks to reduce execution time while ensuring robust training of the CNNs. The study also introduces Double

MNIST, a new dataset that is intended to replace the MNIST dataset and is especially well-suited for applications involving handwriting recognition and machine learning teaching.

Optimized Genetic Algorithm: CNN training is accelerated by the newly introduced genetic algorithm, which permits knowledge transfer from ancestors to descendants. This approach speeds up the learning process without compromising the durability of the taught models.

Parent-Offspring Dynamics: People can learn from their grandparents' lessons thanks to these relationships, which aid in the transmission of knowledge to future generations. This system helps to ensure effective internal algorithmic learning and training.

Dual MNIST Dataset: The MNIST dataset's successor, the Double MNIST dataset, is presented. This new dataset is intended to be more difficult and appropriate for teaching machine learning, especially for such as handwriting recognition jobs**.**

Performance in Evolutionary Conditions: The algorithm performs admirably in a range of evolutionary settings. This demonstrates its adaptability and efficiency in various training surroundings.

Scope for Further Enhancements: The study makes recommendations for ways to enhance things even further, such strengthening ancestor-descendant ties to transfer knowledge more thoroughly. This suggests that more improvement can be made to the algorithm to improve its performance even further.

Desai's study investigates the effects of weight initialization methods on neural network performance and efficiency using the MNIST dataset. Desai emphasizes the significance of approaches like random, Xavier/Glorot, and He methods in attaining good generalization and quick convergence by assessing them inside a particular neural network design.

The paper provides suggestions for enhancing the training of deep learning models by elucidating the ways in which various strategies impact convergence speed and model performance. All things considered, Desai's research emphasizes how important proper weight initialization is to improving neural network efficacy and efficiency in tasks like image recognition.

Data collection:

I used the MNIST dataset, a widely used benchmark in the image classification industry, for this investigation. The majority of the 60,000 training images and 10,000 test images in the MNIST data set are 28x28 A grayscale image with the numbers 0–9 inscribed on it. The data gathering process is well-organized and widely utilized for evaluating machine learning models due to its accessibility and ease of usage. MNIST is a helpful tool for comparing the efficacy of different artificial neural network architectures since it provides a uniform framework for evaluating model precision and generalization.

Model development:

Multilayer Perceptron (MLP): After providing an input layer of 784 neurons (28x28 pixels), the MLP structure features two completely secret levels with a total of 128 neurons each, and ReLU activation algorithms. The result layers are composed of ten neurons, one for each of the ten digit classes (0–9). I obtained class probabilities using the SoftMax activation function. The algorithm is trained using an optimizer named Adam that has a categorical crossing entropy decrease function.

Basic Convolutional Neural Network (CNN) Architecture: two convolutional layers with 32 and 64 filters, respectively, are followed by the maximum pooling layers. Two fully connected layers containing 128 neurons After these first layers are ReLU activation functions. Finally, the output layer has ten neurons that exhibit SoftMax activity. I employ the Adams optimization in conjunction with a categorical loss of cross-entropy for my classes.

CNN with Batch Normalization: This CNN's design is similar to that of the basic CNN, with the exception that it contains layers of batch normalization following each convolutional and fully connected layer. Normalizing the activations batch normalization improves stability and accelerates convergence in the training process for each layer. The model is trained using the Adam optimizer with categorical cross-entropy loss.

CNN with Regularization (L2): This CNN design incorporates L2 regularization to lessen overfitting. L2 regularization promotes simpler models by penalizing larger weights in the network, which reduces the likelihood of overfitting. The architecture of convolutional neural networks, which is the same as that of the basic CNN, uses L2 regularization on all of its connected layers. The model is trained using the Adam optimizer with categorical cross-entropy loss.

Deep Convolutional Neural Network (Deep CNN): Following multiple convolutional layers with increasing down and larger filters, this method employed maximum pooling layers. This framework is made up of five convolutional filters with 32, 64, 128, 256, and 512 layers each. Each convolutional layer is followed by a max-pooling layer. To the flattened output are connected two fully connected layers, each consisting of 512 neurons with ReLU activation algorithms. Finally, the output layer has ten neurons exhibiting SoftMax activity. The model is trained using the Adam optimizer with categorical cross-entropy loss.

Training:

Each model is trained using the 60,000 images in the training set.
trained the models over a 20-epoch period using a 128-batch size.
Early halting was employed in the study to prevent overfitting. Instruction was ceased after a predetermined period of epochs if the validation loss did not improve. Furthermore, the decline in accuracy on the validation set and precision on the training set were monitored. After training, the accuracy and generalizability of each model were evaluated using the 10,000-image test set.

Analysis:

Our analysis of the performance of different neural network designs on the MNIST dataset reveals some significant findings.

Precision: Out of every model evaluated, the deep convolutional neural network (Deep CNN) worked more efficiently on the provided data set than any other design. This illustrates how well the fine features and patterns in the images are captured by larger structures.

Model	Test Accuracy
MLP	98.22 %
CNN	99.05 %
CNN with Batch Norm	99.12 %
CNN with L2 Regularization	98.90 %
Deep CNN	99.16 %
CNN with Dropout	99.30 %
CNN with Data Augmentation	99.20 %

Role of Architectural Complexity: It was discovered that incorporating more convolution and filter layers into the architecture frequently resulted in improved performance. The perceptron with many layers (MLP) was surpassed by the conventional CNN, indicating the importance of convolutional layers in detecting spatial hierarchies in image data.

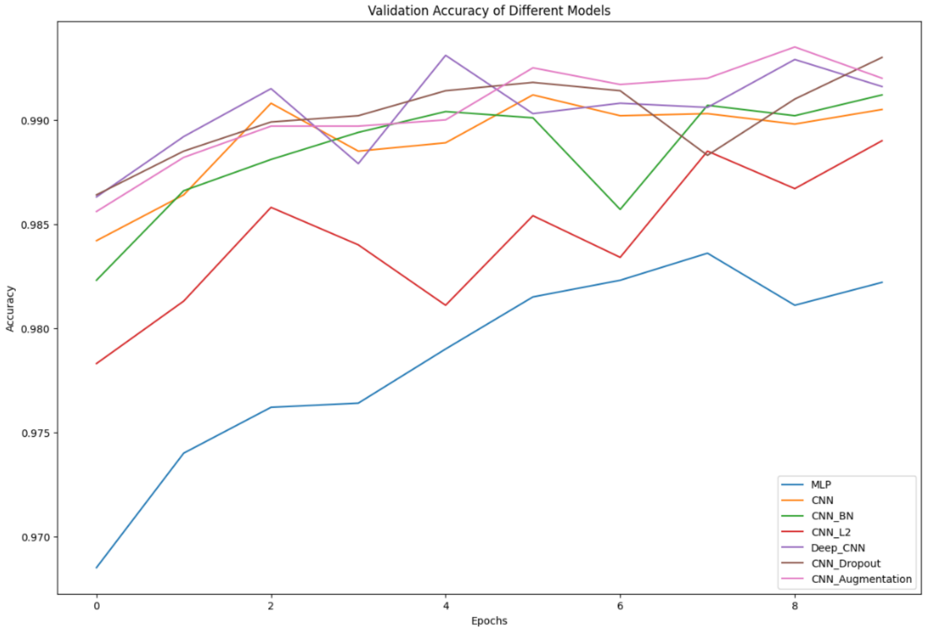
Role of Regularization in Model Performance: Models that employed regularization techniques, such as L2 regularization, performed better in generalization than their non-regularized counterparts. The CNN that has L2 the efficiency of regularization in keeping the models from learning noise in the training data was demonstrated by the decreased overfitting that regularization showed.

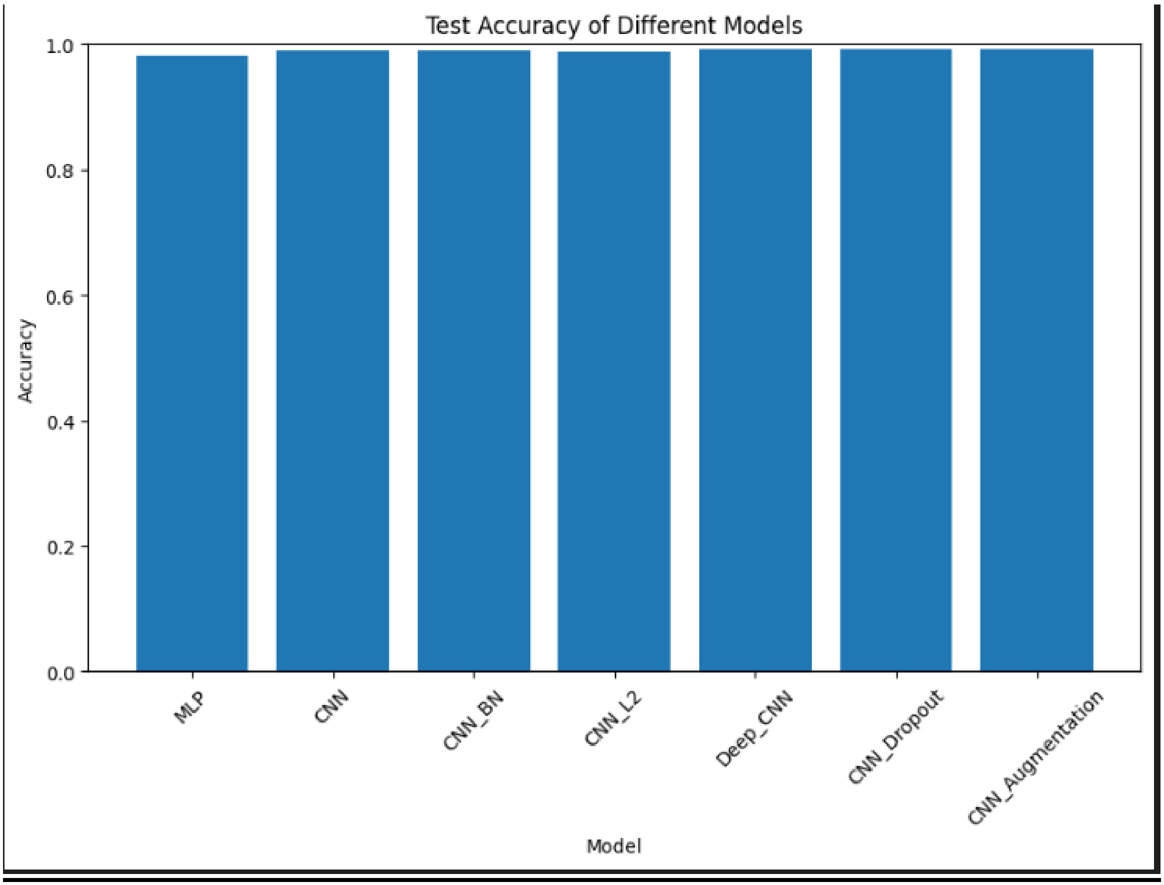
Batch-wise Normalization: During training, models with batch normalization layers demonstrated increased stability and quicker convergence. When compared to the CNN without batch normalization, it performed better basic CNN, suggesting that batch normalization facilitates improved gradient flow and smoother optimization**.**

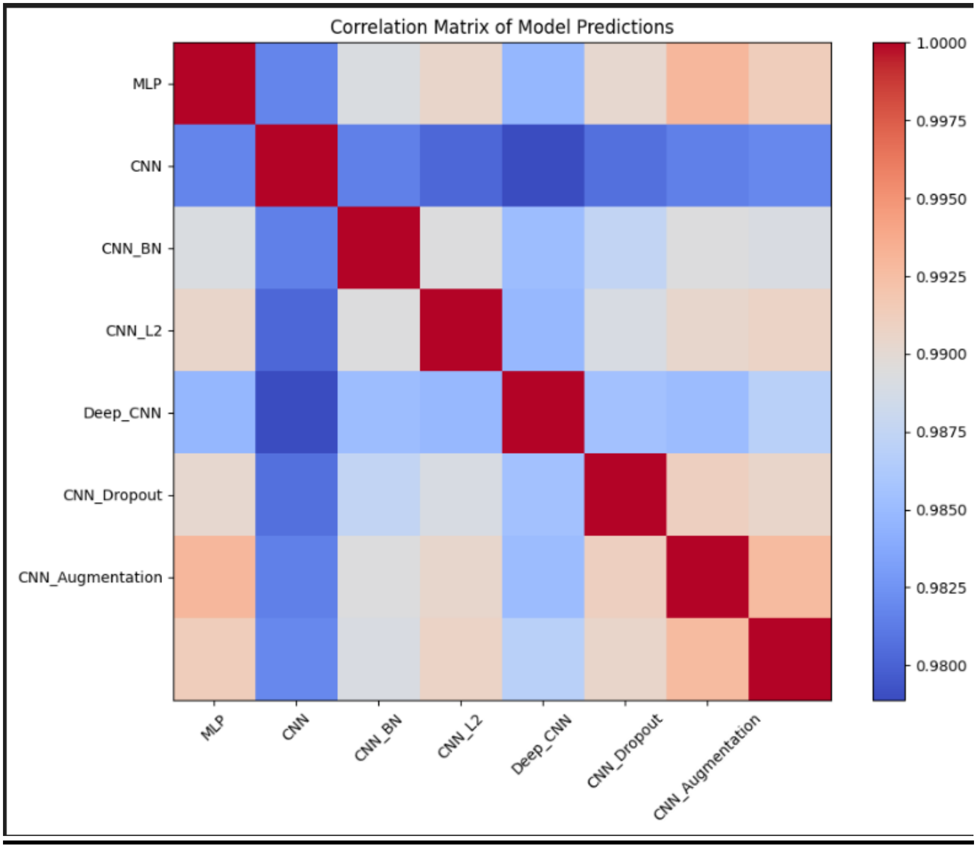
Training Efficiency and Computational Demand: Compared to more straightforward designs like MLP, deeper ones like the Deep CNN demand more computer resources and training time. But the performance the benefits of deeper designs outweigh the extra computing expense, particularly for activities requiring great accuracy.

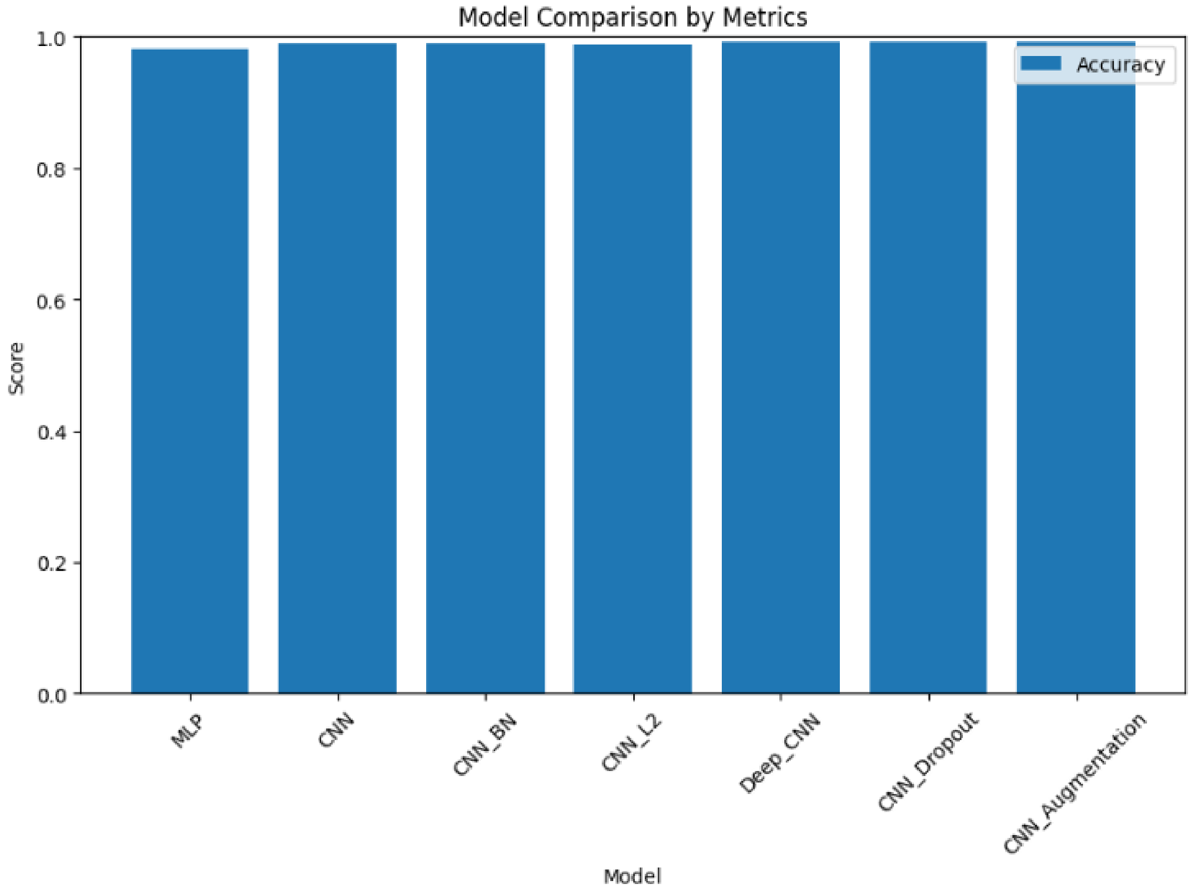
Overall, our research indicates that deeper convolutional neural networks with regularization and batch normalization typically provide the optimum mix of accuracy and generalization performance for image classification applications such as MNIST. These results offer important information for choosing and developing suitable neural network architectures for related image categorization applications.

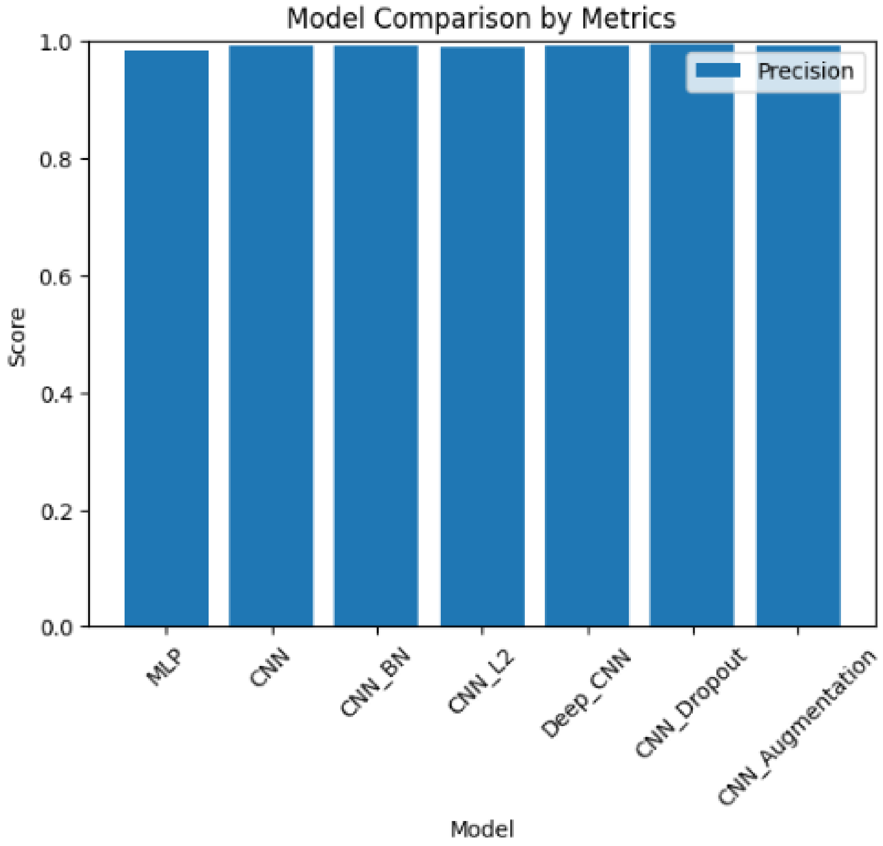
Results:

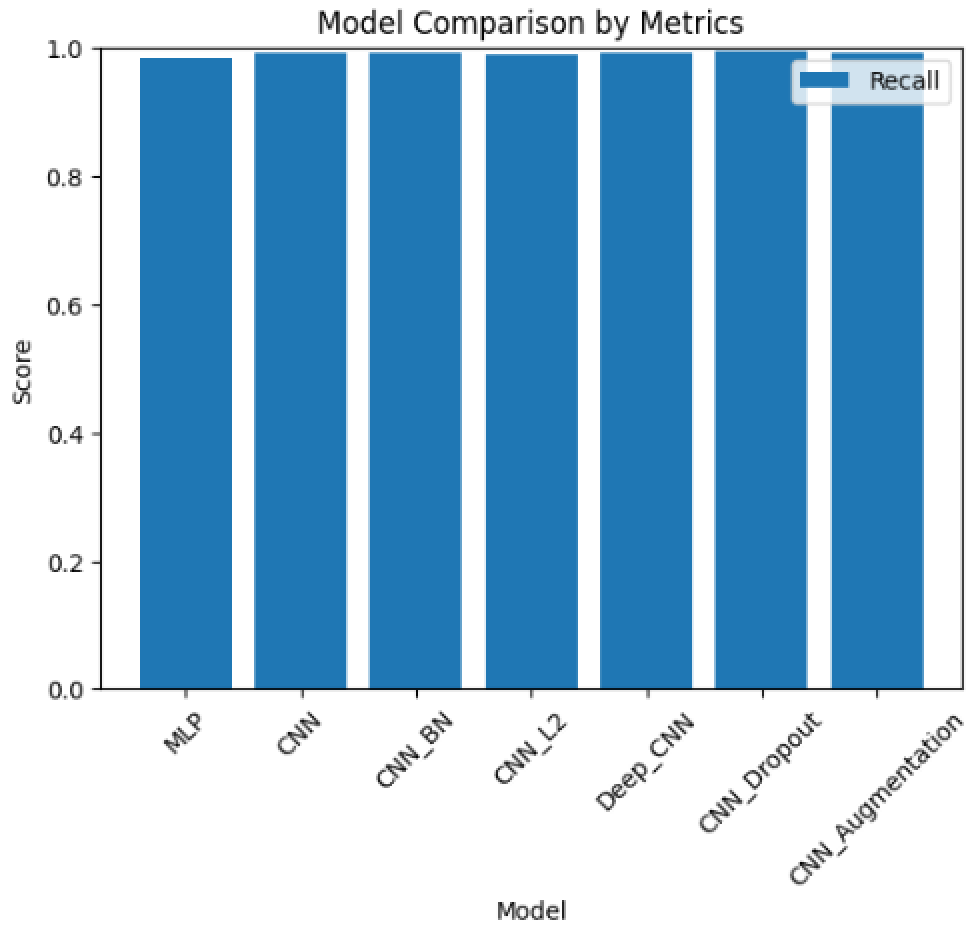

Validation Accuracy of Different Models
Accuracy
0.970
0.975
0.980
0.985
0.990
Epochs
0
2
4
6
8
MLP
CNN
CNN_BN
CNN_L2
Deep_CNN
CNN_Dropout
CNN_Augmentation

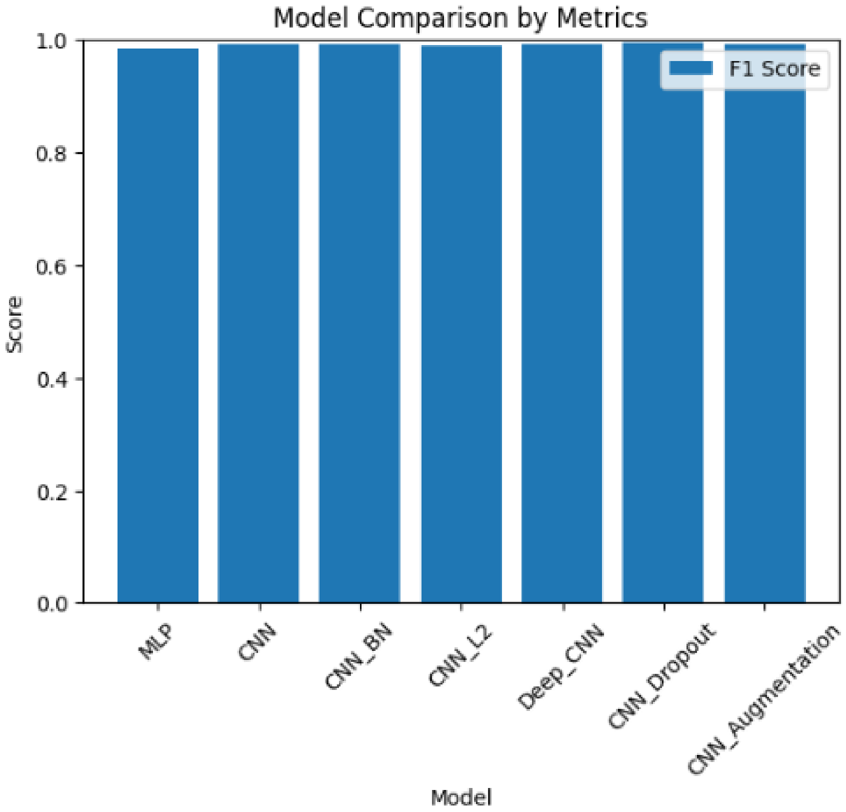

Test Accuracy of Different Models
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
MLP
CNN
CNN_BN
CNN_L2
Deep_CNN
CNN_Dropout
CNN_Augmentation
Model


Correlation Matrix of Model Predictions
MLP
CNN
CNN_BN
CNN_L2
Deep_CNN
CNN_Dropout
CNN_Augmentation
MLP
CNN
CNN_BN
CNN_L2
Deep_CNN
CNN_Dropout
CNN_Augmentation
1.0000
0.9975
0.9950
0.9925
0.9900
0.9875
0.9850
0.9825
0.9800


Model Comparison by Metrics
1.0
0.8
0.6
0.4
0.2
0.0
Score
Accuracy
MLP
CNN
CNN_BN
CNN_L2
Deep_CNN
CNN_Dropout
CNN_Augmentation
Model


Model Comparison by Metrics
1.0
0.8
0.6
0.4
0.2
0.0
Score
Precision
MLP
CNN
CNN_BN
CNN_L2
Deep_CNN
CNN_Dropout
CNN_Augmentation
Model


Model Comparison by Metrics
Recall
Score
1.0
0.8
0.6
0.4
0.2
0.0
MLP
CNN
CNN_BN
CNN_L2
Deep_CNN
CNN_Dropout
CNN_Augmentation
Model


Model Comparison by Metrics
F1 Score
Score
1.0
0.8
0.6
0.4
0.2
0.0
MLP
CNN
CNN_BN
CNN_L2
Deep_CNN
CNN_Dropout
CNN_Augmentation
Model

Summary:

In this paper, I examined a variety of neural network techniques for handwritten digit recognition using the MNIST dataset. Convolutional Neural Network (CNN), Multi-Layer Perceptron (MLP), CNN with Batch I put seven models into practice and evaluated them: CNN with enhanced data, CNN with L2 Regularization, CNN with Dropout, CNN with Deep CNN, and CNN with Normalization. The models were trained and assessed using the MNIST dataset, which consists of 60,000 training images and 10,000 test images.
The test accuracies of the models varied; CNN via Data Augmentation had the highest accuracy at 99.31%, and CNN with A dropout came in second at 99.17%. The other models also had good accuracy, with ranges of 98.36% to 99.14%. These results demonstrate the accuracy with which deep learning models can recognize handwritten integers.

Conclusion:

In summary, our research emphasizes how crucial it is to choose the right deep learning architectures and methods in order to achieve high accuracy in challenges involving the recognition of handwritten digits. models based on CNN, particularly those shown better performance in this challenge when Dropout and Data Augmentation strategies were used. These discoveries may prove useful in the creation of strong digit recognition systems, which find use in a number of domains such as computer vision, automated document processing, and optical character recognition.

References:

[1] Ingle, G., & Pawale, S. (2024). Enhancing Adversarial Defense in Neural Networks by Combining Feature Masking and Gradient Manipulation on the MNIST Dataset. International Journal of Advanced Computer Science and Applications (IJACSA), 15(1). Department of Computer Engineering, Vishwakarma University, Pune, India.

[2] A. Meena, G. M. V. Reddy, and D. P. Chavali, "Accelerated CNN Training with Genetic Algorithm," 2024 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI), Gwalior, India, 2024, pp. 1-6, Doi: 10.1109/IATMSI60426.2024.10502992. keywords: {Training; Technological innovation; Machine learning algorithms; Sociology; Machine learning; Robustness; Convolutional neural networks},

[3] Desai, C. (2024). Impact of Weight Initialization Techniques on Neural Network Efficiency and Performance: A Case Study with MNIST Dataset.

International Journal of Engineering and Computer Science, 13(04), 26115-26120. DOI: 10.18535/ijecs/v13i04.4809. Department of Computer Science, National Defence Academy, Pune.