Exploring MNIST for Robust Handwritten Digit Classification

Mr. Pranav Dalvi
M.Sc Computer Science

Dr. Apurva Yadav
H.O.D MSc Computer Science

Dr. Neha Ansari
Lecturer M.Sc Computer Science

Organisation: Kirti M. Doongursee College

Abstract

In machine learning and pattern recognition, classifying handwritten characters is both challenging and crucial. This research investigates the Modified National Institute of Standards and Technology (MNIST) dataset to improve the robustness and accuracy of handwritten digit classification using Support Vector Machines (SVM), Decision Trees (DT) and K-Nearest Neighbours (KNN). The MNIST dataset, comprising grayscale images of handwritten digits, serves as an extensive platform for assessing character recognition models.

This study employs SVM, KNN, and DT to determine their efficacy in classifying handwritten digits. Through meticulous preprocessing steps, we ensure consistency and comparability across different models.

GridSearchCV is utilized for hyperparameter tuning, optimizing the performance of each model. The results highlight the superior accuracy and robustness of the models, with the SVM model achieving a cross-validation accuracy of 96.97% on the MNIST dataset. Additionally, this research underscores the significance of preprocessing techniques in improving classification performance, particularly when dealing with diverse handwriting styles.

Our findings suggest that the MNIST dataset, coupled with advanced machine learning techniques, can significantly contribute to the development of more accurate and reliable handwritten digit recognition systems. This work paves the way for further research into robust classification methods, potentially extending to real-world applications such as automated form processing and digital handwriting analysis.

Introduction

Handwritten digit recognition is a critical component in various applications, ranging from automated postal mail sorting and bank check processing to digital document archiving and handwriting-based user interfaces. The accuracy and robustness of these systems have significant practical implications, influencing the efficiency and reliability of data extraction from handwritten documents.

The Modified National Institute of Standards and Technology (MNIST) dataset has been a foundational benchmark for assessing handwritten digit recognition algorithms. Created by LeCun et al., the MNIST dataset includes 70,000 grayscale images of handwritten digits, with 60,000 used for training and 10,000 for testing, each measuring 28x28 pixels. It has become the standard for evaluating and comparing various machine learning models.

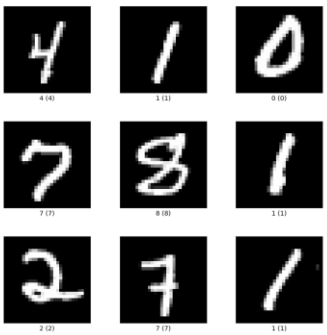


Fig-1. MINIST Dataset example.[8]

This research aims to explore the MNIST dataset for robust handwritten digit classification using Support Vector Machines (SVM), Decision Trees (DT) and K-Nearest Neighbours (KNN). By leveraging the capabilities of these models, we seek to enhance the performance of digit recognition systems. Our study involves a detailed analysis of these models to

determine their effectiveness in classifying handwritten digits from the MNIST dataset.

We employ thorough preprocessing techniques to ensure the input data's consistency and quality. Additionally, we use GridSearchCV for hyperparameter tuning, optimizing each model's performance to achieve the highest possible classification accuracy.

The findings from this research shed light on the strengths and weaknesses of SVM, KNN, and DT in handwritten digit recognition. By analysing the performance of these models on the MNIST dataset, we aim to pinpoint the most effective strategies for creating robust and accurate digit recognition systems. This study ultimately contributes to the progress of handwritten digit recognition technology, with potential applications in various real-world settings.

Literature Review

For decades, handwritten digit recognition has been a pivotal research area in pattern recognition and machine learning. The MNIST dataset, introduced by LeCun et al. in the 1990s, established itself as the standard benchmark for handwritten digit classification, driving the development of various classification algorithms, such as neural networks, support vector machines, and decision trees.`[6][7]`.

With the progress in machine learning, Support Vector Machines (SVMs), K-Nearest Neighbours (KNN), and Decision Trees (DT) have exhibited exceptional performance in image recognition tasks, particularly in handwritten digit classification. Their success on the MNIST dataset is well-documented, with these models achieving high accuracy levels due to their ability to handle various handwriting styles and noise effectively.`[1][8]`.

Previous studies have explored the application of SVMs, KNN, and DTs on MNIST, highlighting the importance of preprocessing steps such as normalization and contrast enhancement to improve model performance`[9][10]`. Additionally, hyperparameter tuning techniques like GridSearchCV have been employed to optimize parameters and achieve better classification accuracy`[11]`.

Despite the advances, challenges remain in achieving robustness against varying handwriting styles, noise, and distortions. This research aims to build upon existing knowledge by conducting a comprehensive evaluation of SVMs, KNN, and DTs on the MNIST dataset, focusing on preprocessing techniques and hyperparameter tuning to enhance robustness and accuracy.

Methodology

To deploy the classification algorithm, we follow a structured approach encompassing data preprocessing, model training, hyperparameter tuning, and evaluation.

1. Model Training:

- Pipeline Creation: Set up a machine learning pipeline with StandardScaler and the classifier (SVM, KNN, DT).
- Hyperparameter Tuning: Use GridSearchCV to find optimal hyperparameters.

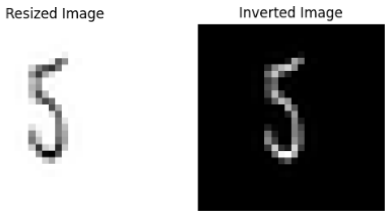
2. Evaluation:

- Cross-Validation: Perform 5-fold cross-validation to evaluate model performance.
- Test Set Evaluation: Assess the final model on a separate test set to determine accuracy.

3. Algorithm Deployment:

- **Load MNIST Dataset:** Use fetch_openml to load the dataset.
- **Train-Test Split:** Divide the dataset into training and testing sets.
- **Fit Model**: Train the model using the best hyperparameters identified by GridSearchCV.
- **Predict and Visualize:** Use the model to predict characters from new images and visualize the results.

During prediction, image processing techniques are applied to reduce noise and enhance the accuracy of digit recognition.



On left we have original image and on right we have processed image.

Results and Discussion

The performance of the SVM, KNN, and DT models was assessed on the MNIST dataset using a range of metrics, including accuracy, precision, recall, and F1-

score. The table below summarizes the optimal parameters and performance metrics for each model:

Model	Best Parameters	Cross-Validation Accuracy	Test Accuracy
SVM	`C=10`, `gamma='scale'`, `kernel='poly'`	96.97%	97.18%
KNN	`n_neighbors=3`, `weights='distance'`, `metric='euclidean'`, `algorithm='auto'`	96.54%	96.62%
DT	`criterion='entropy'`, `max_depth=30`, `min_samples_split=10`, `min_samples_leaf=5`, `splitter='best'`	85.69%	86.17%

Hyperparameter Tuning: Hyperparameter tuning using GridSearchCV proved essential for optimizing model performance. The choice of parameters for each model had a substantial impact on accuracy.

Challenges and Limitations:

In the research, the challenges and limitations encountered in the deployment of handwriting recognition models highlight significant obstacles in achieving uniform accuracy across diverse handwriting styles.

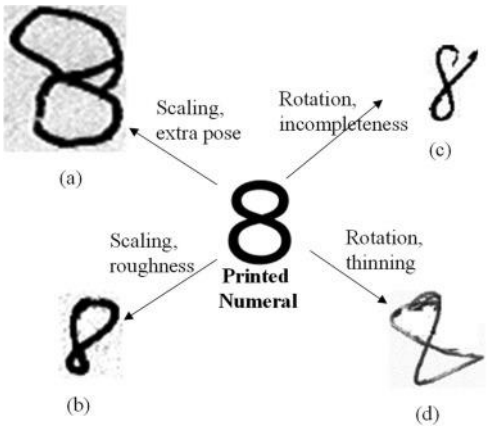


Fig-2. Variations in handwriting.[9]

The variability in individual handwriting presents a significant challenge to the robustness of models like SVM, KNN, and DT, as these variations can significantly affect classification performance. Additionally, the need for extensive preprocessing and hyperparameter tuning is emphasized, as achieving high levels of accuracy necessitates rigorous and often complex adjustments to the models. These processes are vital to align the diverse characteristics of handwritten samples with the trained models, ensuring that the recognition system can effectively interpret a wide range of handwritten inputs. Such challenges underscore the importance of developing more adaptable and sophisticated machine learning techniques to handle the inherent variability in handwritten documents.

Conclusion and Future Enhancement:

This research explored the application of Support Vector Machines (SVM), Decision Trees (DT) and K-Nearest Neighbours (KNN) for handwritten digit recognition using the MNIST dataset. Through rigorous preprocessing and hyperparameter tuning, the models achieved high accuracy, demonstrating their efficacy for this task.

Conclusion:

The conclusion of the research underlines the effective performance of SVM, KNN, and DT models in recognizing handwritten digits when supported by rigorous preprocessing and tuning. These models demonstrated high accuracy, reflecting their robustness in handling the complexities of handwritten character recognition. Furthermore, the critical role of preprocessing is highlighted, indicating that thorough and consistent preprocessing steps are essential for aligning the characteristics of custom images with those of the training data. Such alignment significantly enhances model performance, enabling the models to more accurately interpret and classify diverse handwriting styles. This underscores the necessity of meticulous data preparation to maximize the effectiveness of machine learning models in practical applications.

Future Enhancement:

- Dataset Augmentation: Incorporate more diverse handwriting samples to enhance model robustness against varying styles and noise.
- Advanced Models: While these models performed well, exploring advanced deep learning architectures, such as deeper Convolutional Neural Networks (CNNs), could potentially yield even better results.
- Real-World Application: Extend the research to include real-world handwritten documents and multi-lingual character recognition to evaluate model performance in practical scenarios.
- Automation and Scalability: Develop automated preprocessing pipelines and scalable training frameworks to handle larger

datasets and more complex models efficiently.

References:

1. LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-Based Learning Applied to Document Recognition. _Proceedings of the IEEE_, 86(11), 2278-2324. https://doi.org/10.1109/5.726791

2. Cortes, C., & Vapnik, V. (1995). Support-Vector Networks. _Machine Learning_, 20, 273-297. https://doi.org/10.1007/BF00994018

3. Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Springer.

4. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. _Journal of Machine Learning Research_, 12, 2825-2830. https://jmlr.org/papers/v12/pedregosa11a.html

5. Salakhutdinov, R., & Hinton, G. E. (2007). Learning a Nonlinear Embedding by Preserving Class Neighbourhood Structure. _Artificial Intelligence and Statistics (AISTATS)_.

6. Schölkopf, B., & Smola, A. J. (2002). Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT Press.

7. Platt, J. (1998). Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines. Microsoft Research Technical Report MSR-TR-98-14.

8. Fig-1: https://www.tensorflow.org/datasets/catalog/mnist

9. Fig-2: https://www.sciencedirect.com/science/article/pii/S1319157822002270