Hybrid CNN-SVM Classifier for Handwritten Digit Recognition on the MNIST Dataset

Jaewon Tudor

Abstract

Convolutional Neural Networks (CNN) excel in feature extractions, but they rely on softmax layers for classification, which limit their accuracy when working with complex scenarios and data, for example handwritten digit recognition which is an important problem in computer vision particularly when working with digitalized data processing. This project implements a new approach on creating a highly accurate and fast hybrid model for digit recognition by fusing a Convolutional Neural Network for feature extraction along with a Support Vector Machine (also known as SVM) for classification, inspired by the research paper by Abien Fred Agarap titled 'An Architecture Combining CNN and SVM for Image Classification (Agarap 2017). This hybrid architecture, along with custom optimizations such as data augmentation and dropout regularization added, improved the classification accuracy to 99.45 percent on my implementation as opposed to 72-90 on Abien's research proving that this project indeed has a lot of potential in improving image classification and image recognition software.

1. Introduction

Handwritten digit recognition is a fundamental problem in computer vision, powering vast amount of application that range from financial document digitalization to automated data entry systems and many more. The MNIST (Modified National Institute of Standards and Technology) dataset is one of the most popular digit recognition datasets in the field of machine learning and AI, it consists of a large amount of grayscale images of handwritten digits by many individuals which differ in style, this dataset has been used widely as a way to evaluate classification algorithms and model performance. Some of the traditional methods for solving this problem include Support Vector Machines (SVMs) and CNNs, where each has its strengths and limitations.

CNNs generally excel in extracting spatial features from images making them the the perfect choice for visual data procssing on platforms like Jupyter Notebook. However CNNs use a softmax classifier which can underperform in separating more complex patterns from data, for example when the feature space is highly-dimensional. Support Vector Machines (SVMs) however, are known for their excellent performance in classification tasks when the feature space is high dimensional, especially when combined with RBFs (Radial Basis Function). SVMs do struggle with raw image data since they aren't able to handle spatial data directly.

Several research studies have validated the use and effectiveness of CNN and CNN-SVM based models for handwritten digit recognition. The research paper titled 'Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN) ' highlights the CNN's ability to effectively extract the features from an image while emphasizing the importance of proper hyperparameter tunning such as kernel size and the number of layers for optimal performance. The paper 'Improved Recognition of Handwritten Digits Using Convolutional Neural Network (CNN) ' compares the performance of different optimizers such as the Adam optimizer and Stochastic Gradient Descent with Momentum (SGDM), which achieves a benchmark accuracy of 99.50 precent on the MNIST dataset with a 7-layer CNN architecture. These findings aided in the design and optimization of the hybrid architecture for the CNN-SVM model.

This report is focusing on a custom hybrid approach at solving both of these model's limitations inspired by the work done by Abien Fred Agarap (Agarap 2017) where an SVM is able to achieve high prediction and classification performance by leveraging features extracted through the use of CNNs. Since both CNNs have active limitations in feature extraction or classification, using this hybrid approach in combining the two will mitigate these limitations and will improve the final model classification performance by a large margin.

The MNIST dataset was used in training and evaluation, which consists of 60.000 images used for training and a testing data split that consists of 10.000 images, each of these is normalized to grayscale with a total size of 28x28px across all images to ensure no errors happen during training which might affect the model's performance. The classification performance of the model was measured in F1-score, recall as well as confusion matrices, accuracy and precision

scores. Data Augmentation and and regularization methods such as dropout layers were added to enhance the model's generalization. The testing and validation results show an accuracy score of 99.45 percent for the hybrid model which surpasses the performance on standalone CNN models. that rely on softmax layers for classification.

2. Methods

2.1. Model Overview

This project uses a hybrid model approach by combining a CNN (Convolutional Neural Network) and an SVM (Support Vector Machine) for classification with features extracted from the MNIST dataset using the CNN for its advanced feature extraction capabilities. CNNs are also meaningful and effective at extracting spatial data and features from images, and SVMs excel in separating data in high-dimensional feature spaces, preferably non-linear relationships by using a kernel known as RBF or Radial Basis Function.

2.2. Dataset

The MNIST dataset is a widely used evaluation and benchmark dataset for models where handwritten digit recognition was utilized. The dataset consists of 60,000 training images and 10,000 test images, each of size 28×28 pixels in grayscale. The images were normalized to values in the range [0, 1] for uniformity and making the training process error-free while maintaining model accuracy.

2.3. Model Architecture

The hybrid model consists of the following components:

- **Convolutional Neural Network (CNN) Feature Extractor:** The CNN architecture includes two convolutional layers with ReLU activation functions, after which max-pooling layers are added to reduce spatial dimensions. Dropout layers are also incorporated to prevent overfitting of the data in the model. A fully connected dense layer outputs feature vectors for classification, which is later used by the SVM.

- **Support Vector Machine (SVM) Classifier:** The extracted feature vectors from the CNN are then passed to an SVM with an RBF kernel. Hyperparameters, such as the regularization strength C and kernel coefficient γ, were optimized using grid search.

2.4. Mathematical Formulation

The CNN extracts features $f(x)$ from input images $x \in \mathbb{R}^{28\times28}$. These features are input into the SVM classifier, which minimizes the following objective function:

$$\min_{w,b} \frac{1}{2}||w||^2 + C\sum_{i=1}^{m} \max(0, 1 - y_i(w^T f(x_i) + b)), \quad (1)$$

where:

- w is the weight vector,
- b is the bias,
- C is the regularization parameter,
- y_i is the true label, and
- x_i are the input features.

2.5. Training Process

The model was trained using the following steps:

- **CNN Training:** The CNN was trained on the MNIST dataset for 20 epochs with a batch size of 32 using the Adam optimizer ($\alpha = 0.001$) and sparse categorical cross-entropy as the loss function.
- **Feature Extraction:** After training, the CNN dense layer outputs were extracted as feature vectors, then fed into the Support Vector Machine.
- **SVM Training:** The feature vectors extracted through the CNN were used to train the SVM classifier.

2.6. Hyperparameters

The performance of the SVM classifier depends heavily on the choice of hyperparameters. Table 1 lists the key hyperparameters and their selected values.

Table 1. Hyperparameter settings for the SVM classifier.

Parameter Name	Value
Regularization Strength (C)	1.0
Kernel	RBF
Kernel Coefficient (γ)	0.1

2.7. Hardware and Software

The training was done on a personal deep learning system with an NVIDIA RTX 6000 Ada with 48GB VRAM, 92 GB of system RAM, and an AMD EPYC 4564P 16-core 32-thread processor. The following libraries and frameworks were utilized in an NVIDIA AI Workbench custom environment based on the NGC Rapids Base Anaconda Environment:

- **TensorFlow/Keras:** For CNN implementation and training.

- **Scikit-learn:** For SVM implementation and hyperparameter tuning.
- **Matplotlib and Seaborn:** For visualization of results.

3. Experiments

3.1. Experimental Settings

The goal of the experiment was to evaluate the effectiveness of using a hybrid CNN-SVM model for handwritten digit recognition on the MNIST dataset:

- **Tasks:** The goal was to classify handwritten digits (0–9) by using a Convolutional Neural Network (CNN) for feature extraction and an SVM for classification. The hybrid model's classification performance was compared to a baseline CNN with a softmax classifier.
- **Dataset:** The MNIST dataset, described in Section 2.2, contains 60,000 training images and 10,000 test images, each normalized to values in the range [0, 1] for uniformity and to prevent any interference with the model training.
- **Comparison:** The baseline CNN used a softmax classifier for classification, while the hybrid model used SVM with RBF kernel on CNN-extracted feature vectors.
- **Hyperparameter Selection:** The SVM hyperparameters, regularization strength C and kernel coefficient γ, were tuned using grid search over $C \in \{0.1, 1, 10\}$ and $\gamma \in \{0.01, 0.1, 1\}$.
- **Hardware and Software:** Training and evaluation were conducted on an NVIDIA RTX 6000 Ada GPU with 48 GB VRAM, using TensorFlow/Keras for CNN training, Scikit-learn for SVM implementation, and Matplotlib/Seaborn for visualizations, the results were later generated on matplotlib.

3.2. Evaluation Criteria

The model was evaluated using the following criteria:

- **Accuracy:** Measures the proportion of correctly classified digits out of the total samples.
- **Precision:** The ratio of true positives to all predicted positives, reflecting prediction quality on the MNIST dataset.
- **Recall:** The ratio of true positives to actual positives, indicating the ability to identify all instances
- **F1-Score:** The harmonic mean of precision and recall, providing a balanced evaluation metric, this is also widely used in computer vision models as well.
- **Confusion Matrix:** A visualization to analyze classification accuracy across all digit classes for the MNIST dataset

These metrics are the industry standard for classification models, providing good insights into the model's performance before deployment in real-life situations.

3.3. Results

Table 2 summarizes the performance.

Table 2. Performance of CNN-SVM hybrid model

Model	Accuracy	Precision	Recall	F1-Score
CNN-SVM	**99.45%**	**99.45%**	**99.45%**	**99.45%**

Figure 1 illustrates the model's performance through combined plots showing training/validation loss, accuracy, confusion matrix, and metrics.

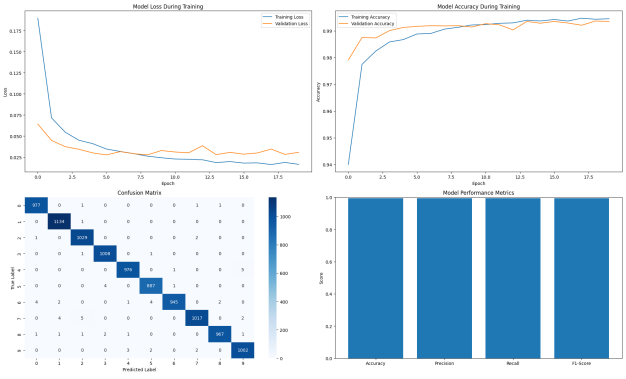

Figure 1. Performance metrics of the CNN-SVM hybrid model, including training/validation loss, accuracy, confusion matrix, and evaluation metrics.

3.4. Discussion

The results demonstrate that the hybrid CNN-SVM model is achieving high accuracy and balanced metrics. Key factors contributing to this improvement include:

- **Feature Extraction:** The CNN's structure enabled effective extraction of spatial features, which served as robust inputs for the SVM classifier.
- **SVM's Classification:** The RBF kernel provided flexibility in modeling non-linear decision boundaries, improving precision and recall.

- **Regularization and Augmentation:** The inclusion of dropout and data augmentation during CNN training enhanced generalization, resulting in better performance on the test set.

4. Conclusion

This report showed a hybrid CNN-SVM model for handwritten digit recognition using the MNIST dataset. By combining the feature extraction capabilities of Convolutional Neural Networks (CNNs) with the classification properties of Support Vector Machines (SVMs), the proposed model achieved a high accuracy of 99.45percent. This surpasses the limitations of standalone CNNs, particularly their reliance on softmax classifiers, which can underperform in separating complex patterns in high-dimensional feature spaces, this usually applies for larger datasets.

The MNIST dataset served as the benchmark for the model, demonstrating the hybrid model's ability to perform well across diverse handwritten digits. Techniques such as dropout regularization and data augmentation also improved the model's performance and prevented overfitting, validating the effectiveness of the hybrid approach in possible real world problems.

The content of this report highlight the potential of hybrid architectures for image classification tasks, particularly when applied to high-dimensional data. The model's strong performance in accuracy, precision, recall, and F1-score underscores its suitability for real-world applications, such as automated document processing and optical character recognition (OCR) systems. Overall, the hybrid CNN-SVM model demonstrates a significant advancement in solving handwritten digit recognition problems.

5. References

- Agarap, A. F. (2018). "An Architecture Combining Convolutional Neural Network (CNN) and Support Vector Machine (SVM) for Image Classification". arXiv preprint arXiv:1712.03541.

- PubMed Central (2020). "Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN)". Available at: https://pubmed.ncbi.nlm.nih.gov/32545702/.

- Authors from Sensors Journal (2020). "Improved Recognition of Handwritten Digits Using Convolutional Neural Network (CNN): Comparison of Adam and SGDM Optimizers". Sensors, 20(12), p. 3344.

- Raj, D. (2023). "Convolutional Neural Networks (CNN) — Architecture Explained". Medium. Available at: https://medium.com/@draj0718/convolutional-neural-networks-cnn-architectures-explained-716fb197b243.