Handwritten Text Recognition Using EMNIST and Hybrid Techniques

Prajwal Moharana, Pranav Kallem, Tejas Chandramouli, Jayden Lim
University of North Carolina at Chapel Hill
Department of Computer Science

ABSTRACT

This project explores handwritten character recognition using the EMNIST dataset by comparing classical and neural network-based machine learning techniques. Key methods include Multi-Layer Perceptrons (MLP), Convolutional Neural Networks (CNN), and Quadratic Discriminant Analysis (QDA). Dimensionality reduction techniques, such as Principal Component Analysis (PCA), are employed. An external dataset is used as validation data to evaluate model generalizability. The following results highlight trade-offs between accuracy and computational efficiency across methodologies.

I. Introduction

Handwritten character recognition is a pivotal task in machine learning, enabling advancements in document digitization, automated form processing, and accessibility tools. This study leverages the EMNIST dataset [1], a comprehensive extension of the well-known MNIST dataset [2], to evaluate and compare the efficacy of various machine learning approaches, including both deep learning and classical techniques. By exploring Multi-Layer Perceptrons (MLP), Convolutional Neural Networks (CNN), and traditional classifiers like Logistic Regression, Random Forest, and Quadratic Discriminant Analysis (QDA), this research aims to identify the strengths and limitations inherent to each methodology.

The challenges in handwritten text recognition stem from the variability in individual writing styles, noise introduced during data acquisition, and the high dimensionality of image data. The EMNIST dataset addresses these challenges by providing a large and diverse collection of handwritten characters, making it an ideal benchmark for evaluating different machine learning models. This study assesses the performance of each model on the EMNIST dataset and investigates their generalizability by validating the results on an external dataset.

II. Dataset and Preprocessing

The EMNIST dataset consists of 28×28 grayscale images categorized into 26 classes, each representing a distinct handwritten character. These characters encompass both uppercase and lowercase letters merged into single categories, providing a comprehensive challenge for classification algorithms. The dataset is partitioned into training and testing sets, containing 124,800 and 20,800 images, respectively.

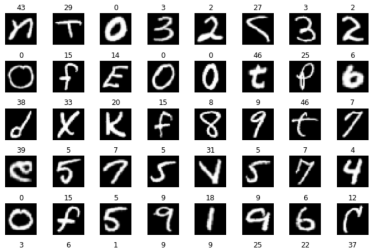

Fig. 1: EMNIST Sample Picture

A. Data Preprocessing

Effective preprocessing is necessary for enhancing model performance and ensuring compatibility with various machine learning architectures. The preprocessing pipeline includes several key steps. First, normalization is applied by scaling pixel values to the range $[0, 1]$, standardizing the input data to facilitate faster convergence during training and prevent numerical instability. Next, images are reshaped depending on the model type. For Multi-Layer Perceptrons (MLPs), images are flattened into vectors of size 784 (28×28), enabling the network to process each pixel as an individual feature. In contrast, Convolutional Neural Networks (CNNs) retain the original spatial structure of the images, which allows for the extraction of hierarchical features through convolutional layers. Finally, Principal Component Analysis (PCA) is employed to reduce feature dimensionality while retaining 95% of the variance. This dimensionality reduction decreases computational overhead and mitigates the curse of dimensionality, making it particularly beneficial for classical machine learning methods.

III. Methodologies

This section highlights the various machine learning techniques employed to tackle the handwritten character recognition task using the EMNIST dataset. The methodologies encompass both neural network-based approaches and classical machine learning classifiers, each with distinct advantages and computational requirements.

A. Multi-Layer Perceptron (MLP)

The Multi-Layer Perceptron (MLP) is a foundational neural network architecture characterized by its feedforward structure with multiple hidden layers. In this study, the MLP was

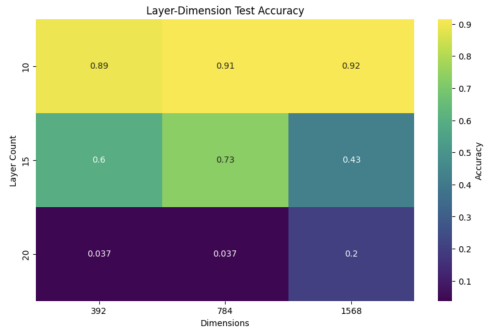

Fig. 2: Effect of layers and dimensions on MLP accuracy.

utilized with varying configurations to determine the optimal architecture for classification accuracy.

1) Architecture and Configuration: Different configurations of hidden layers and dimensions were explored to understand their impact on model performance. The number of hidden layers varied, and for each configuration, the number of neurons per layer was adjusted. The best-performing MLP architecture comprised 10 hidden layers with hidden dimensions calculated as $28 \times 28 \times 2$, resulting in 1568 neurons per layer.

2) Training Procedure: Each MLP model was trained for 15 epochs to observe convergence trends without excessive computational costs. During training, the loss function was monitored by plotting the loss values over epochs, providing insights into the model's learning dynamics and stability. The Cross-Entropy Loss function was employed, coupled with the Adam optimizer [3]. These are known for their adaptive learning rate capabilities and effectiveness in resolving issues like gradient vanishing.

3) Performance and Analysis: The optimal MLP model achieved a test accuracy of approximately 92%, demonstrating accurate performance on the EMNIST dataset. The experiments indicated that increasing the number of hidden units contributed positively to test accuracy, suggesting that a higher capacity model can capture more intricate patterns in the data. Additionally, larger models exhibited reduced complexity during training, likely due to the enhanced representational power of increased hidden units.

However, scaling up the model size introduced challenges related to training stability and convergence speed. Larger networks faced difficulties such as longer training times and potential overfitting. These issues were addressed through several training strategies. Proper weight initialization was crucial to ensure effective gradient flow and to prevent vanishing or exploding gradients. Additionally, batch normalization layers were incorporated to stabilize the learning process by normalizing layer inputs, which accelerated training and improved generalization. Finally, the use of ReLU (Rectified Linear Unit) activation functions mitigated the gradient vanishing problem and introduced the non-linearity essential for learning complex patterns.

```
def train_model(model, num_epochs, train_loader,
    optim, input_dim=28*28, cnn=False):
    model.train()
    loss_arr = []
    for i in range(num_epochs):
        total_loss = 0

        if i == 10:
            optim.param_groups[0]["lr"] = 0.0001
        for _, (data, labels) in tqdm(enumerate(
            train_loader)):
            data = data.to(device)
            labels = labels.to(device)
            labels = labels - 1
            if not cnn:
                data = data.view(-1, input_dim)
            logits, loss = model(data, labels)
            optim.zero_grad()
            loss.backward()
            optim.step()
            total_loss += loss
        loss_arr.append(total_loss.item() / len(
            train_loader))
        print(f"Epoch {i + 1}: {total_loss.item() /
            len(train_loader)}")
    return loss_arr
```

Listing 1: Neural Network Training Code

B. Convolutional Neural Network (CNN)

Convolutional Neural Networks (CNNs) are specifically designed to process data with grid-like topology, such as images. Their ability to capture spatial hierarchies through convolutional layers makes them highly effective for image-based tasks.

1) Architecture and Configuration: The CNN architecture employed in this study consisted of multiple convolutional layers interleaved with activation and pooling layers, followed by fully connected layers. The key parameters tweaked included the number of hidden channels in the convolutional layers and the number of training epochs. Increasing the number of channels allows the network to learn more complex and abstract features from the data.

2) Training Procedure: Similar to the MLP, each CNN model was trained for a varying number of epochs to evaluate the impact on performance. The training process involved monitoring the loss over epochs to ensure proper convergence and to detect potential overfitting. The optimizer used was Adam, facilitating efficient training through adaptive learning rates.

3) Performance and Analysis: All CNN models demonstrated high performance, achieving test accuracies in the range of 94-95%. The experiments revealed a positive correlation between the number of hidden channels and the number of training epochs with the overall test accuracy. Specifically, models with more convolutional channels and trained over more epochs tended to capture more nuanced features, thereby enhancing classification performance.

However, increasing the network size and complexity necessitates careful monitoring to prevent overfitting. Larger CNNs have a higher capacity to memorize training data, which can degrade generalization performance on unseen data. Techniques such as dropout, weight regularization, and early

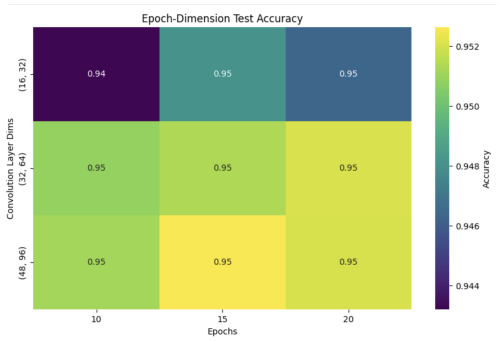

Fig. 3: Effect of convolutional dimensions and epochs on CNN accuracy.

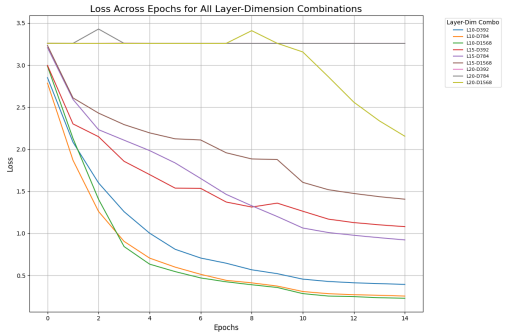

Fig. 4: Loss Across Epochs A.2

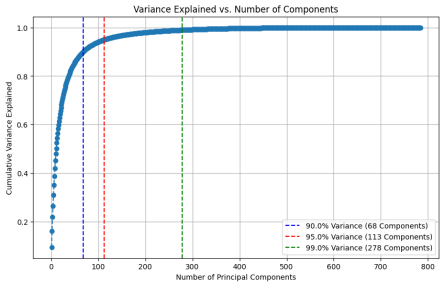

Fig. 5: PCA Dimensionality Reduction: Variance vs Components

stopping were considered to mitigate overfitting, although they were not extensively detailed in this study.

The superior performance of CNNs compared to MLPs underscores the importance of leveraging spatial information inherent in image data. By effectively capturing hierarchical patterns through convolutional operations, CNNs provide a more nuanced understanding of the input, leading to higher accuracy in classification tasks.

C. Classical Techniques

In addition to neural network-based approaches, classical machine learning techniques were applied to the EMNIST dataset to establish a comparative baseline. These methods include Logistic Regression, Random Forest, and Quadratic Discriminant Analysis (QDA).

1) Data Transformation for Classical Models: Classical machine learning algorithms typically operate on tabular data formats, requiring conversion of the PyTorch datasets into pandas dataframes. This transformation provided a structured data format that worked with scikit-learn's interface.

2) Model Implementations and Training: Logistic Regression, a linear model used for classification tasks, was implemented to predict the probability of each class based on input features. Random Forest, an ensemble learning method, constructs multiple decision trees during training and outputs the mode of the classes for classification tasks. Quadratic Discriminant Analysis (QDA), a generative classifier, assumes that each class follows a Gaussian distribution with its own covariance matrix, allowing for more flexible decision boundaries compared to linear models.

3) Performance and Analysis: The classical methods demonstrated respectable performance on the EMNIST dataset but did not surpass the accuracy achieved by deep learning models. Specifically, Logistic Regression and Random Forest both attained an accuracy of 87% on the EMNIST test set. QDA, on the other hand, initially underperformed, achieving only 40% accuracy. This suboptimal performance can be attributed to QDA's assumption of Gaussian distributions for each class, which may not hold true in high-dimensional spaces like those presented by image data.

4) Dimensionality Reduction with PCA: To enhance the performance of classical classifiers, Principal Component Analysis (PCA) [4] was employed to reduce the feature dimensionality. PCA effectively compressed the 784-dimensional input data into 113 principal components, retaining 95% of the variance and achieving an 85% reduction in dimensionality. This reduction not only decreased computational complexity but also mitigated the curse of dimensionality, potentially improving classifier performance by eliminating noise and redundant features.

After applying PCA, the performance of the classical models improved significantly. Logistic Regression maintained a similar accuracy of 87%, indicating that PCA preserved the most relevant features for linear classification. Random Forest also remained consistent at 87%, demonstrating its ability to handle dimensionality reduction effectively. Quadratic Discriminant Analysis (QDA) exhibited a substantial improvement, achieving 83% accuracy post-PCA. This enhancement suggests that PCA mitigated some of the challenges posed by QDA in high-dimensional spaces, allowing it to perform more effectively.

Despite these improvements, classical methods still lagged behind neural network-based approaches in terms of overall accuracy. This discrepancy underscores the advantage of deep learning models in capturing complex, non-linear patterns inherent in image data.

IV. Results and Analysis

The performance of the various models was evaluated on both the EMNIST test set and an external dataset to assess

generalizability. The results are summarized in Table I.

Model	EMNIST Accuracy (%)	External Dataset Accuracy (%)
MLP	92	46
CNN	95	72
Logistic Regression	87	13
Random Forest	87	14
QDA (PCA)	83	15

TABLE I: Comparison of Model Performances

A. Performance on EMNIST Dataset

CNNs outperformed all other models, achieving the highest accuracy of 95%. MLPs followed closely with an accuracy of 92%, demonstrating their effectiveness despite being less specialized for image data. Classical classifiers, enhanced with PCA, exhibited lower accuracies, with Logistic Regression and Random Forest both attaining 87%, and QDA achieving 83%. CNNs outperform MLPs by effectively leveraging spatial information and capturing hierarchical patterns, resulting in higher classification accuracy.

B. Generalizability to External Dataset

To evaluate the ability of the trained models, an external dataset from Kaggle (https://www.kaggle.com/datasets/dhruvildave/english-handwritten-characters-dataset) was utilized. This dataset consists of handwritten English characters with different dimensions (900 $\times$ 1200 pixels), requiring preprocessing steps to align with the training data. The preprocessing involved resizing the images to 40 $\times$ 40 pixels to reduce computational load. Subsequently, the images were cropped to 28 $\times$ 28 pixels to match the EMNIST dataset dimensions. Finally, since the training data was transposed, a similar transformation was applied to the external dataset to maintain consistency.

In scenarios where models made random predictions, an expected accuracy of approximately 3.84% (1 out of 26 classes) was calculated. However, the models demonstrated significantly higher accuracies on the external dataset. The MLP achieved an accuracy of 45.6%, indicating a moderate level of generalization. In contrast, the CNN attained a substantially higher accuracy of 71.9%, showcasing superior adaptability to unseen data.

While these accuracies are lower than those observed on the EMNIST test set, they still reflect a meaningful level of predictive capability. The CNN's superior performance over the MLP can be attributed to its convolutional layers, which effectively capture and generalize spatial patterns in the data. In contrast, MLPs, which lack specialized mechanisms for spatial feature extraction, are less adept at generalizing to new, slightly altered data distributions.

The diminished performance on the external dataset is likely due to the off-centered nature of the training data, which may not have aligned well with the distribution of the new data. This discrepancy highlights the importance of training data diversity and the potential benefits of data augmentation techniques to enhance model robustness.

Classical methods, even when enhanced with PCA, struggled with the external dataset, achieving accuracies ranging from 13% to 15%. This underperformance underscores the rigidity of classical classifiers, which lack the inherent adaptability of deep learning models to changes in data distributions and feature representations.

V. Conclusion

This study comprehensively evaluated the performance of both classical and neural network-based machine learning techniques for handwritten character recognition using the EMNIST dataset. The findings reveal several key insights. Convolutional Neural Networks (CNNs) achieved the highest accuracy of 95% on the EMNIST test set, demonstrating their exceptional capability in capturing and generalizing spatial patterns inherent in image data. Multi-Layer Perceptrons (MLPs) also performed admirably, attaining a 92% accuracy. While slightly less effective than CNNs, MLPs showcased the importance of network depth and hidden unit capacity in enhancing performance.

Classical classifiers, even when augmented with PCA for dimensionality reduction, lagged behind deep learning models in accuracy. Logistic Regression and Random Forest achieved 87% accuracy, while Quadratic Discriminant Analysis (QDA) reached 83% post-PCA. These results highlight the challenges classical methods face in high-dimensional and complex feature spaces typical of image data. Validation on an external dataset confirmed the robustness of CNNs, which maintained a substantial accuracy of 71.9%, compared to 45.6% for MLPs. In contrast, classical methods struggled, underscoring the adaptability of deep learning models to varying data distributions.

The study also emphasized the importance of architectural choices, such as the number of hidden layers and units in MLPs, and the number of convolutional channels in CNNs. Additionally, training techniques like weight initialization, batch normalization, and appropriate activation functions were critical in ensuring model stability and performance.

In conclusion, while classical machine learning techniques offer computational efficiency and simplicity, deep learning models, particularly CNNs, provide superior accuracy and adaptability for handwritten character recognition tasks. The integration of neural network feature extraction with classical classifiers presents a promising avenue for future research, potentially combining the strengths of both methodologies to achieve even higher performance levels.

Future work may explore advanced deep learning architectures, such as Residual Networks (ResNets) [5] or Transformer-based models, to further enhance recognition accuracy. Additionally, incorporating data augmentation and regularization techniques could improve model generalizability and robustness against diverse handwriting styles and data distributions.

REFERENCES

[1] G. Cohen, S. Afshar, J. Tapson, and A. van Schaik, "Emnist: Extending mnist to handwritten letters," *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 2921–2926, 2017.

[2] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[3] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[4] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, p. 20150202, 2016.

[5] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.