Classification of Fashion MNIST Dataset Using Dense Neural Network and Convolutional Neural Network

Karan Saraswat
M.S. in Industrial Engineering
Texas A&M University
College Station, U.S.
ksaraswat@tamu.edu

Geetesh Challur
M.S. in Computer Science
Texas A&M University
College Station, U.S.
geeteshch@tamu.edu

Venkat Pitta
M.S. in Computer Science
Texas A&M University
College Station, U.S.
pitta.venkat@tamu.edu

***Abstract*— In this article, both Convolutional Neural Network (CNN) and Dense Neural Network (DNN) models are developed and assessed for their effectiveness in image recognition task on Fashion MNIST dataset for classifying articles of clothing based on the performance of these models. The algorithmic frameworks are applied to the Fashion MNIST dataset, and a comprehensive evaluation of their performance is conducted. The outcomes of this study reveal that the DNN model achieves an accuracy rate of 89% on the MNIST dataset, while the CNN model demonstrates a commendable accuracy of 90%. The utilization of these models on the Fashion MNIST dataset showcases their capabilities in handling complex image recognition tasks.**

Keywords—Convolutional Neural Network, Accuracy, Dense Neural Network, PCA, t-SNE, Cross Validation, Clothing Industries

I. INTRODUCTION

Image classification is a fundamental aspect of computer vision that involves the automated categorization of images into predefined classes or labels. Leveraging advanced machine learning techniques, particularly Neural Networks, image classification algorithms learn to discern and extract distinctive features within images. A Dense neural network, or dense neural network, consists of layers where each neuron is connected to every neuron in the adjacent layers, enabling information flow throughout the network. Notable for their hierarchical architecture, CNN employ convolutional layers to automatically extract features from input data, capturing spatial hierarchies and enabling robust pattern recognition. This model consist of convolutional, pooling, and Dense layers, allowing it to efficiently learn and generalize complex visual patterns. Both DNN and CNN have revolutionized image-related tasks, exhibiting exceptional performance in image classification, object detection, and facial recognition.

This study offers a concise exploration of image classification models. Two neural network models are employed: (a) DNN, and (b) CNN specifically designed to tackle the issue of model overfitting. The subsequent sections of the paper include a summary of related work in Section 2, an overview of methods and the proposed algorithm in Section 3, classification experiments and discussions in Section 4, and concluding remarks with future directions in Section 5.

II. LITERATURE REVIEW

In the past few years, significant advancements have been made in constructing image detection and recognition classifiers across diverse datasets, employing a range of machine learning algorithms. Notably, deep learning has demonstrated substantial enhancements in accuracy across various datasets. Specifically, the convolutional neural networks revealed special effects on computer vision and image classification. Several noteworthy endeavors in this realm are outlined below:

Shobhit et al. [1] introduced an innovative approach to classifying images of fashion articles. The research focused on training convolutional neural network (CNN) architectures within a deep learning model to effectively categorize the Fashion-MNIST dataset. To enhance the learning rate, three distinct CNN methods were implemented, featuring residual skip networks and batch normalization. The findings of the study revealed a notable 2% increase in accuracy rates for existing systems, showcasing the effectiveness of the proposed model.

Wang Di [2] emphasizes the importance of deploying computationally efficient neural networks in real-world applications. Comparing Fully Connected Neural Network, CNN, MobileNetV1, and MobileNetV2 on Fashion-MNIST, MobileNetV2 emerges as the most effective, achieving a remarkable 93% accuracy. The findings underscore the practical significance of MobileNet's simplicity and heightened accuracy.

Yehya Abouelnaga et al. [3] developed an ensemble of classifiers utilizing the K-Nearest Neighbors (KNN) algorithm. Integrated KNN with Convolutional Neural Networks (CNN) to address Overfitting concerns, implementing Principal Component Analysis (PCA) for dimensionality reduction. The amalgamation of these two classifiers resulted in a notable enhancement, yielding an improved accuracy by approximately 0.7%.

Edmira Xhaferra et al. [4] investigates the efficacy of Convolutional Neural Networks (CNNs) in fashion industry applications, targeting challenges such as model overfitting and intricate garment classification. The study employs CNN-C1 and CNN-C2 architectures for image classification in the FASHION MNIST database. Results demonstrate that CNN-C2 outperforms CNN-C1, achieving an enhanced accuracy of 93.11%.

Rahul Chauhan et al. [5] investigates Deep Learning algorithms, focusing on Convolutional Neural Networks (CNNs) that mirror the human cerebral cortex. His study evaluates CNN models' performance on MNIST and CIFAR-10 datasets, achieving an impressive 99.6% accuracy on MNIST and optimizing CIFAR-10 results through real-time data augmentation and dropout on a CPU unit.

III. METHOD

A. Data Preparation

In this study, we tackled the challenge of fashion image classification within the extensive Fashion MNIST dataset, which consists of 70,000 images. Employing two different Neural Networks, we aimed to improve efficiency in categorizing clothing types. The dataset contains images, each represented in grey scale with dimensions of 28x28.

The complexity of classification lies in the nuanced fashion divisions, where labels reflect the diverse types of clothing depicted in the images. Our study aims to enhance the accuracy and effectiveness of fashion image classification, ensuring a more refined understanding of the dataset's 10 distinct classes, as illustrated in Fig. 1.

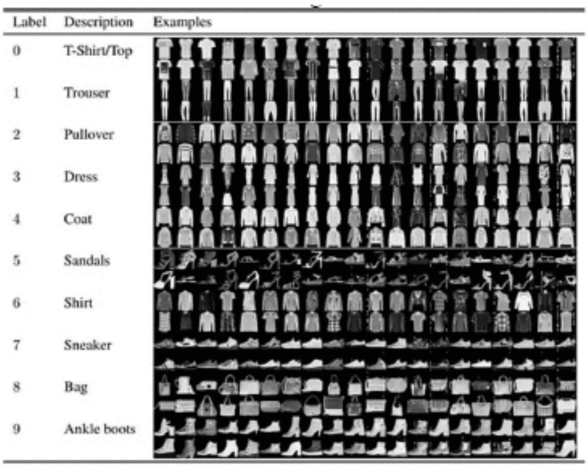

Fig 1. Fashion MNIST dataset overview[4]

i. Data cleansing and transformation

Dataset doesn't have any missing values or corrupted images which makes it easier to use for our analysis. Data normalization is crucial for image classification as it ensures consistent feature scales, preventing dominance by certain features. It is done by scaling pixel values to a range between 0 and 1.

ii. Data Splitting

The dataset is sourced from torch vison's fashion clothing image database, includes 60,000 training images and 10,000 test images. Training data is further split into validation sets keeping equal number of data points in each class.

B. Exploratory Data Analysis

i. Descriptive statistics

Ensuring uniform data length across all classes is crucial to prevent analysis errors stemming from class imbalance. This is verified by visualizing class distribution with a count plot showing an equally distributed data in each class for split training and validation datasets given below:

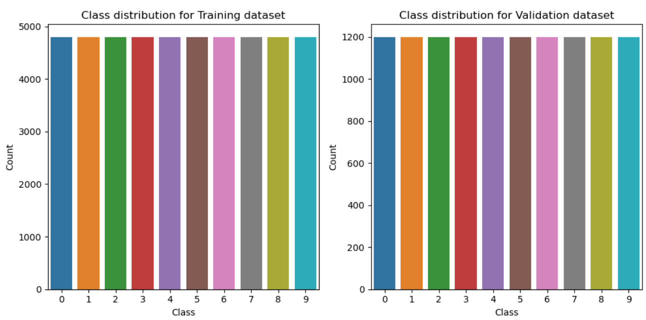

Fig 2. Data distribution among ten classes for training and validation

ii. Data Visualization

In this study, Principal Component Analysis (PCA) is employed to reduce dataset dimensions while preserving key information. By assessing correlations between dimensions, it retains the most meaningful variables, discarding those with lower explained variance. The following visualization illustrates images before PCA and post-PCA application, showcasing the preservation of key information in the reduced dimensionality.

Fig 3. Fashion accessories picture before (top) and after PCA (bottom)

In the graph below, 87 components cover about 90% variance, while 187 components explain around 95%. As components increase, variance reaches 100%, aligning with the original 784 dimensions.

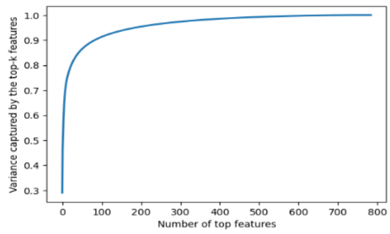

Fig 4. Cumulative variance captured with respect to number of features

Following are plots of first two and first three components of PCA data, while each color a label:

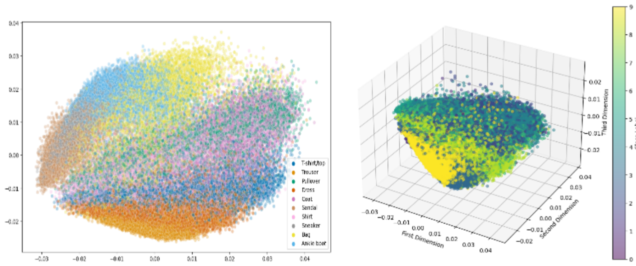

Fig 5. PCA components plots: 2 components (left) and 3 components(right)

From the above plots, different categories are separated apart to some extent, but it's unclear how distinct the various categories are from one another. Even a plot with three components does a poor job of separating classes. Thus, t-Distributed Stochastic Neighboring Entities is applied on first five thousand rows to reduce the number of dimensions. The following plot shows significant improvement in classification wherein each class is properly clustered and separated from one another.

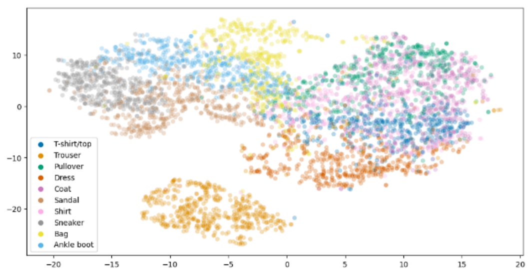

Fig 6. t-SNE first two components plot

C. Model Selection

i. Algorithm selection

Drawing insights from the visualizations in Zalando's fashion-mnist repository, it becomes evident that the data is characterized by non-linearity and complex pattern relationships [6]. In addressing this complexity, neural networks stand out from k-nearest neighbors and Decision Tree algorithms by autonomously learning hierarchical feature representations. This inherent capability contributes to their superior performance on the dataset, enabling effective adaptation to intricate image patterns unlike k-nearest neighbors and decision tree algorithms.

Hence, in this study, the initial choice of a Dense Neural Network (DNN) is motivated by its capability to capture intricate patterns within the data. As a second choice, a Convolutional Neural Network (CNN) is used, because of its specialization in image-related tasks and its efficient handling of spatial hierarchies of features through shared weight parameters.

ii. Model building

Two neural network architectures are developed - a Dense Neural Network (DNN) and a Convolutional Neural Network (CNN). The models are implemented in Pytorch.

Dense Neural Network: The 28x28 image is flattened into a 786-pixel vector. Then, Principal Component Analysis is applied to reduce the dimension from 786 to a 187-pixel vector, which is fed to the dense neural network. The DNN model consists of fully connected layers arranged in a feedforward architecture. The number of nodes in each layer, the optimizers, and the batch size are tuned as a hyperparameter. ReLU activation is used for all hidden layers. The output layer uses softmax activation to output classification probabilities.

Convolutional Neural Network: The CNN architecture comprises convolutional, pooling and dense layers tailored for image data. Input 28x28 images are passed through 3 convolutional blocks with kernels of size 3x3 and appropriate padding to retain spatial dimensions. Along with conv blocks, 2x2 max pooling reduces dimensions while preserving key activations. Leaky ReLU activation is used for convolutional layers and ReLU for dense layers. Batch normalization is performed on each layer. Final classification layer is a softmax activated dense layer for 10-way probabilistic prediction. An Adam optimizer with cross-entropy loss is used for end-to-end model training.

Both models are trained to optimize cross-entropy loss. L2 regularization is used to prevent overfitting.

Binary cross entropy loss is given by (1) as follows:

$$H(X) = -\sum P(X = i) * \log P(X = i) \quad (1)$$

Where H(X) is cross entropy function and P(X) is a target distribution.

The training set and test set are loaded from the dataset separately. The training set is divided into 80% training and 20% validation. The models are evaluated on validation set performance after each training epoch. After selecting the best model through hyperparameter tuning, a final evaluation is done on the held-out test set. Hyperparameter is not performed on the convolutional neural network due to the time and resource constraints.

IV. EXPERIMENTS AND RESULTS

The objective of the experiments was to train and tune a DNN model and a CNN model for fashion image classification and evaluate its performance thoroughly.

A. Experiment 1: Dense Neural Network

i. Model Tuning

The DNN comprised fully connected layers with input size 784, corresponding to the flattened 28x28 images. Hyperparameter tuning was conducted on key parameters - number of layers, nodes per layer, activation function, optimizer, and batch size. The tuning was automated using a training loop that evaluated all combinations systematically. Models were trained for 30 epochs with cross-entropy loss optimization with a learning rate of 0.001. Performance was tracked using accuracy. Precision, and recall and F1 were calculated and analyzed for the best model.

ReLU activation was selected in hidden layers for faster convergence compared to sigmoid/tanh functions. Batch normalization was avoided as it slowed down training. RMSprop and Adam were chosen because they provide faster loss reduction compared to SGD. A batch size of 32 and 64 is chosen for balanced computation efficiency with ability to estimate loss and gradients effectively from each batch.

The hyperparameters search revealed that a 3-layer model with [187, 64, 10] nodes, Adam optimizer and batch size of 32 achieved the best validation accuracy of 89.74%. This model was saved for final evaluation.

I. DNN HYPERPARAMETER TUNING

Model	Layers	Nodes/Layer	Optimizer	Batch Size	Validation Accuracy
DNN 1	**2**	**[187, 64, 10]**	**Adam**	**32**	**89.74%**
DNN 2	2	[187, 64, 10]	Adam	64	89.58%
DNN 3	2	[187, 64, 10]	RMSprop	32	87.85%
DNN 4	2	[187, 64, 10]	RMSprop	64	88.44%
DNN 5	3	[187, 64, 32, 10]	Adam	32	89.17%
DNN 6	3	[187, 64, 32, 10]	Adam	64	89.58%
DNN 7	3	[187, 64, 32, 10]	RMSprop	32	88.01%
DNN 8	3	[187, 64, 32, 10]	RMSprop	64	88.73%

ii. Model Evaluation

The tuned DNN model was tested on the unseen test set of 10,000 images. It attained a competitive accuracy of 89%, demonstrating effective generalization. Precision, recall and F1 scores for each class were computed to analyze performance variation across classes. A confusion matrix was plotted to uncover specific misclassifications.

The model performed very well in categories like T-shirt, Pullover, Dress and Shirt with F1 scores above 0.9, aided by large training samples. Performance was weaker for more niche categories like Sandal and Sneaker due to greater visual variability and smaller training set. Overall results confirm DNN's competence at extracting discriminative fashion item features.

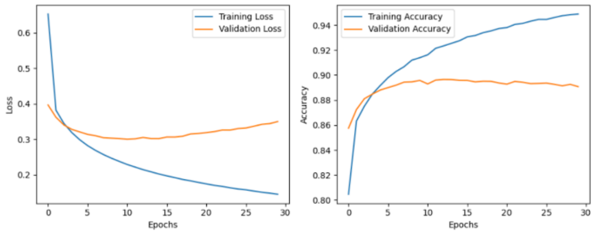

Fig 7. Loss (left) and accuracy (right) plot for DNN

```
Classification report for best FNN on test dataset :
              precision    recall  f1-score   support

           0       0.82      0.83      0.83      1000
           1       0.99      0.97      0.98      1000
           2       0.78      0.82      0.80      1000
           3       0.88      0.89      0.89      1000
           4       0.83      0.80      0.82      1000
           5       0.97      0.96      0.96      1000
           6       0.71      0.69      0.70      1000
           7       0.93      0.96      0.95      1000
           8       0.97      0.97      0.97      1000
           9       0.97      0.96      0.96      1000

    accuracy                           0.89     10000
   macro avg       0.89      0.89      0.89     10000
weighted avg       0.89      0.89      0.89     10000


Confusion Matrix on Test dataset:
[[833   0  20  33   3   2 100   0   9   0]
 [  4 970   1  19   3   0   2   0   1   0]
 [ 20   0 822   9  71   1  74   0   3   0]
 [ 18   9  12 892  33   2  29   0   5   0]
 [  2   1 101  27 800   0  69   0   0   0]
 [  1   0   0   1   0 956   0  28   2  12]
 [126   1  96  28  49   0 690   1   9   0]
 [  0   0   0   0   0  15   0 964   2  19]
 [  5   0   4   5   2   3  10   4 967   0]
 [  1   0   0   0   0   5   1  37   0 956]]
```

Fig 8. Classification report (top) and confusion matrix (bottom) for DNN

B. *Experiment 2: Convolutional Neural Network*

i. *Hyperparameter Selection*

The convolutional neural network is implemented after the dense neural network, given its specialization in image-related tasks and efficient handling of spatial hierarchies of features through shared weight parameters. The CNN consists of a conv block followed by a dense block. The conv block extracts the important features from the data, while the dense block is used to classify labels based on the extracted features.

Hyperparameter tuning was not performed due to time and resource constraints. Kernels of size 3x3 and stride 1 are used as they are the popular choice. The Leaky ReLU activation was selected in the convolutional layers instead of ReLU to prevent dead neurons and enable more robust learning. The batch size was increased to 128 from the DNN's 64 to allow for larger batch-normalized parameter updates to speed up training. Batch normalization enabled using a higher learning rate of 0.001 for faster convergence.

ii. *Model Evaluation*

CNN was trained for 10 epochs with a batch size of 128. Training and validation accuracy reached 94.4% and 90.77%, respectively, after the final epoch. The small gap indicates minimal overfitting.

```
Epoch [10/10], Training Loss: 0.1467, Validation Loss: 0.2960, Training Accuracy: 0.9440, Validation Accuracy: 0.90
77
```

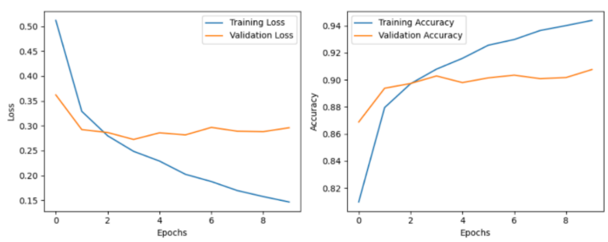

Fig 9. Loss (left) and accuracy (right) plot for CNN

The best-performing model based on validation accuracy was evaluated on the unseen test set. It achieved an accuracy of 90%, comparable to the DNN performance. The per-class precision, recall, and F1 scores reflect reasonably balanced performance across categories.

```
Classification report for best CNN on test dataset :
              precision    recall  f1-score   support

           0       0.87      0.82      0.85      1000
           1       0.98      0.98      0.98      1000
           2       0.85      0.85      0.85      1000
           3       0.86      0.94      0.90      1000
           4       0.87      0.82      0.84      1000
           5       0.95      0.97      0.96      1000
           6       0.73      0.73      0.73      1000
           7       0.96      0.91      0.93      1000
           8       0.97      0.97      0.97      1000
           9       0.93      0.97      0.95      1000

    accuracy                           0.90     10000
   macro avg       0.90      0.90      0.90     10000
weighted avg       0.90      0.90      0.90     10000


Confusion Matrix on Test dataset:
[[823   5  10  37   2   3 111   1   8   0]
 [  0 979   1  17   0   0   1   0   2   0]
 [ 16   2 851  17  48   1  62   1   2   0]
 [ 12   5   6 944  11   0  21   0   1   0]
 [  1   4  63  45 819   1  65   0   1   1]
 [  0   0   0   1   0 972   0  12   0  15]
 [ 89   5  66  33  64   0 732   0  11   0]
 [  0   0   0   0   0  34   0 909   0  57]
 [  0   1   6   6   1   3   7   4 971   1]
 [  0   0   0   0   0  10   0  19   1 970]]
```

Fig 10. Loss (left) and accuracy (right) plot for CNN

The confusion matrix revealed commonly confused categories like Shirt/T-shirt, Sandal/Sneaker, etc. This is expected to be due to subtle visual differences between similar fashion classes. Overall, CNN demonstrates reliable generalization ability.

V. CONCLUSION

The study covers various aspects of deep learning, DNN and CNN and performs image classification on Fashion MNIST dataset. The outcome of this study reveals noteworthy insights into the performance of Dense Neural Network (DNN) and Convolutional Neural Network (CNN) algorithms. CNN exhibited superior accuracy, achieving an impressive 90%, compared to DNN's commendable 89%. This outcome underscores the effectiveness of CNNs in discerning intricate patterns within fashion-related images.

From a business perspective, the high accuracy achieved by the CNN model holds promising implications for a clothing company. The superior performance in image classification using CNNs suggests that such models can play a crucial role in automating the categorization of clothing items. This automation, driven by accurate and efficient algorithms, could streamline inventory management, enhance product recommendation systems, and ultimately contribute to an improved customer experience. The deployment of robust image classification models, as demonstrated in our study, has the potential to revolutionize various aspects of the clothing industry, offering efficiency gains and valuable insights for business decision-making.

REFERENCES

[1] S. Bhatnagar, D. Ghosal, and M. H. Kolekar, "Classification of fashion article images using convolutional neural networks," in 2017 Fourth International Conference on Image Information Processing (ICIIP), 2017: IEEE, pp. 1-6.

[2] Wang Di, "A comparative research on clothing images classification based on neural network models" in 2020 IEEE 2nd International Conference on Civil Aviation Safety and Information Technology, 2020, pp. 1-5.

[3] Yehya Abouelnaga , Ola S. Ali , Hager Rady , Mohamed Moustafa, " CIFAR-10: KNN-based ensemble of classifiers", IEEE, March 2017.

[4] Edmira Xhaferra, Elda Cina, and Luçiana Toti, "Classification of Standard FASHION MNIST Dataset Using Deep Learning Based CNN Algorithms" in International Symposium on Multidisciplinary Studies and Innovative Technologies, IEEE, October 2022.

[5] Rahul Chauhan, Kamal Kumar Ghanshala, and R.C. Joshi, "Convolutional Neural Network (CNN) for Image Detection and Recognition" in 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), IEEE, 2018.

[6] Github repository : https://github.com/zalandoresearch/fashion-mnist

[7] Blog Post, "Classification of Fashion MNIST Dataset Using DNN and CNN", https://medium.com/@ksaraswat_97923/fashion-mnist-image-classification-deep-learning-using-dnn-and-cnn-675f786d1b0d

Link to Blog Post: https://medium.com/@ksaraswat_97923/fashion-mnist-image-classification-deep-learning-using-dnn-and-cnn-675f786d1b0d
Link to code: https://github.com/geeteshCh/FashionMNIST_Classification