Deep Dive into FashionMNIST Image Classification: Unveiling the Dominance of CNNs over XGBoost and Random Forests

Ching-Yao Lin
Dept. Data Science
Texas A&M University
College Station, Texas
chris1997@tamu.edu

Harika Juvvanapudi
Dept. Data Science
Texas A&M University
College Station, Texas
hjuvvana@tamu.edu

Arunkumar Tamilselvan
Dept. Electrical and Computer Engineering
Texas A&M University
College Station, Texas
arunkumartselvan@tamu.edu

***Abstract*—This paper presents a comparative study of three machine learning models—Random Forest, XGBoost, and FashionCNN—applied to the FashionMNIST dataset. We found that the first two models, while having relatively high accuracy, are struggling with modeling nonlinearity in the dataset. The FashionCNN outperforms the others with an accuracy of 92.16%. Results highlight trade-offs between traditional machine learning and deep learning, offering insights for practical applications.**

Keywords—FashionMNIST, CNN, Random Forest, XGBoost

I. INTRODUCTION

In image classification, the FashionMNIST dataset serves as a widely adopted benchmark, featuring 60,000 training and 10,000 testing images across 10 distinct fashion categories. Early research explored traditional models like k-Nearest Neighbors and Support Vector Machines, followed by advancements in ensembled tree-based models such as Random Forest and XGBoost, achieving superior performance. In recent years, the rise of deep learning, exemplified by Convolutional Neural Networks (CNN), has revolutionized image classification, outperforming previous models in feature extraction and processing.

This paper focuses on assessing three model types—Random Forest, XGBoost, and Deep Learning (DL), with specific attention to two DL models: (1) FashionMLP and (2) FashionCNN. Through meaningful comparisons, the study aims to provide insights into their relative performance. Additionally, the relevance of PCA and t-SNE in enhancing interpretability is explored, contributing to ongoing discussions on optimizing image classification for the FashionMNIST dataset.

The relevant code used for this work as well as a blog post is provided at the end of this paper.

Literature Review

The application of deep neural networks in image processing spans various domains, significantly contributing to efficient solutions, including both traditional machine learning (ML) and deep learning methods [1] [2] [3]. In our literature review, we explored papers that encompassed these realms.

In [1], J. Luo critically assessed various approaches, guiding model selection by evaluating the strengths and weaknesses of different architectures, particularly in the context of challenges posed by the FashionMNIST dataset. In [2], Kim, Bashir, and Cavallo delved into applying multiple CNNs for fashion image classification, emphasizing ensemble models to enhance accuracy. These insights are invaluable for improving model performance, a crucial consideration in our experimental framework.

In [3], Gadri and Neuhold explored predictive modeling using a hybrid approach, combining traditional ML techniques with cutting-edge DL methods. This comprehensive exploration is pertinent to our experiment's goal of optimizing fashion image classification through hybrid models, highlighting the nuances involved in merging ML and DL approaches.

II. METHOD

A. Exploratory Data Analysis (EDA)

Before any analyses or modeling efforts, it is imperative to understand the distribution of the data.

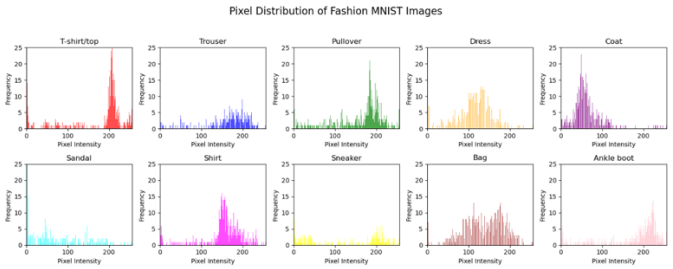

Fig 1 Pixel Distribution of FMNIST Images for Each Class

First, we discovered that the distribution of classes is perfectly balanced. Each fashion category has exactly 6,000 training samples, totaling 60,000 over 10 classes. This means there is no need to worry about imbalanced classes.

Second, given that the pixels serve as the fundamental features in this particular task, our attention is directed toward analyzing the pixel distributions. Figure 1 illustrates the pixel distributions associated with each distinct fashion class. Notably, various fashion items exhibit discernibly different frequencies in their pixel usage across the dataset. Overall, we can conclude that the pixel-level variations in the images indeed provide distinctive signals for classification.

Beyond routine descriptive statistics, we utilized PCA and t-SNE to gain a deeper understanding of the dataset's structure. PCA illustrates the variance captured by each principal component, revealing the contribution of features to the overall data structure. However, PCA assumes a linear relationship between features and hence is not well suited to capturing non-linear relationships. In Figure 2, it is apparent that PCA struggles with separating data into different classes and we can reasonably hypothesize that the FashionMNIST data is inherently nonlinear. Besides, the explained variance ratio is around 34%, which is relatively low.

On the other hand, t-SNE is a nonlinear dimensionality technique that is particularly used for visualizing high-dimensional data while preserving the local structure of the data, as shown in Figure 3. Each point on the scatter plot represents an image in the reduced space, with colors indicating their respective class labels. This visualization provides insights into the nonlinear separability and clustering of different fashion items in the dataset.

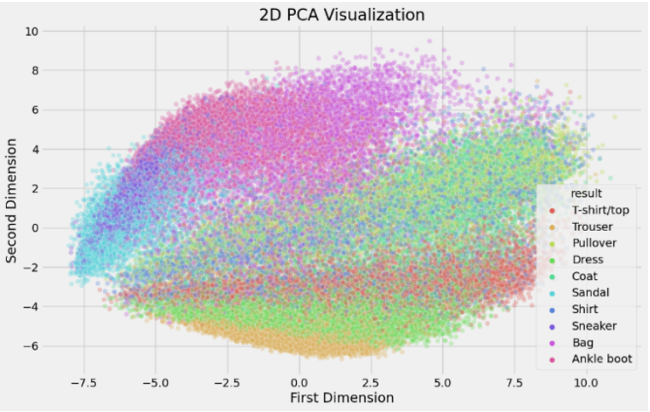

Fig 2 PCA Visualization of High-dimensional Image

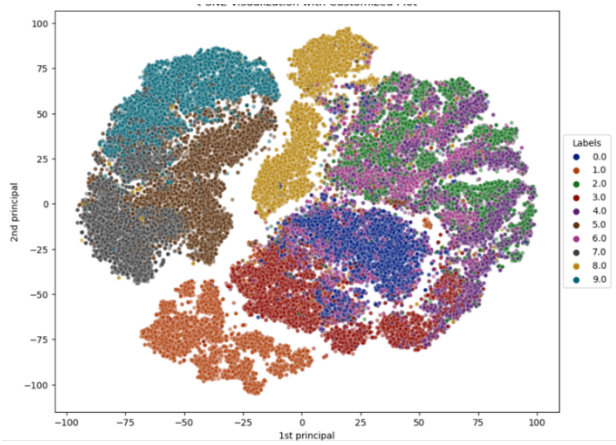

Fig 3 T-SNE Visualization of High-dimensional Images

B. *Random Forest*

After some investigation into the data, the focus now shifts towards model development. Random Forest was chosen due to its ability to capture non-linear relationships, robustness to irrelevant features, regularization to prevent overfitting, effectiveness in high-dimensional data, and support for multiclass classification.

We trained the model on the training data and performed cross-validation, presenting the accuracy, average precision, recall, and F1-score to measure the model's performance. A simple grid search was employed to fine-tune the entire model.

C. *XGBoost*

XGBoost was chosen for similar reasons as Random Forest, due to its ensemble nature that effectively mitigates overfitting and handles the inherent complexity of high-dimensional image data. These characteristics make it well-suited for classifying fashion items from image data. Furthermore, simple preprocessing steps, including

flattening the images and scaling pixel values to zero mean and unit variance, were taken to optimize the input for efficient learning by the algorithm.

An XGBoost model was built on the training set and Bayesian optimization was employed for hyperparameter tuning due to its efficiency in optimizing complex, black-box functions. Model evaluation was done using accuracy, precision, recall, and F1-score, along with a visualization of its confusion matrix.

D. Deep learning: FashionMLP

Deep learning models has shown promising potential in previous studies; hence we developed two deep learning models, FashionMLP and FashionCNN. The models were implemented using the PyTorch framework, with FashionMLP serving as the baseline architecture for comparison against the more advanced FashionCNN model.

FashionMLP was designed similar to a multi-layer perceptron (MLP) architecture to process the gray-scale images in FashionMNIST. It was comprised of three linear layers arranged in a sequential manner. The initial layer had 28 x 28 input features, followed by two hidden layers with 400 and 150 neurons, respectively, and finally outputting 10 classes.

Note that the FashionMLP model does not contain any activation functions, which means it is a purely linear model. Therefore, it is not well-suited for this nonlinear classification task.

The primary aim of FashionMLP was to establish a foundational architecture and serve as a baseline for image classification tasks on FashionMNIST. The model's design prioritized simplicity while providing a reference point for evaluating the performance of more complex models.

E. Deep learning: FashionCNN

FashionCNN, our advanced image classification model tailored for FashionMNIST, leverages CNN architecture enriched with regularization layers to grasp intricate patterns within the dataset. This model marks a significant advancement over the basic FashionMLP. Crafted meticulously, FashionCNN consists of three primary layers, each dedicated to extracting crucial features from the images, ultimately enhancing its ability to discern and classify items accurately.

- CNN Layer 1:
  - **Conv2d**: Convolved input images with 32 filters with 3 x 3 kernels and padding of 1.
  - **BatchNorm2d**: Normalized and stabilized the gradients during training.
  - **ReLU**: Introduced nonlinearity to capture complex patterns.
  - **MaxPool2d**: Downsampled the spatial dimensions by a factor of 2 to extract dominant features.
- CNN Layer 2: Possessed the same architecture as CNN Layer 1.
- Output Layer:
  - **Linear**: Flattened the output from the previous layer and converted it to a dimensional space of 600 neurons.
  - **ReLU**: Same as before.
  - **Dropout**: The dropout rate was set to 0.2 to prevent overfitting.
  - **Linear**: Contained 120 neurons.
  - **Linear**: Contained 10 neurons.
  - **Softmax**: Generated class probabilities, aiding in making informed classification decisions by assigning likelihoods to each fashion category based on the learned representations.

FashionCNN amalgamates convolutional layers for feature extraction with fully connected layers for representation refinement, culminating in a sophisticated network designed to outperform other baseline models in image classification.

III. EXPERIMENTS

We evaluated our models with both cross-validation and the separate test set of size 10,000. Then, we fine-tuned our best model, FashionCNN, and provided our best result on the testing data. For cross-validation, in order to ensure fair comparisons across different models, we adopted stratified 5-fold cross-validation to evaluate all our models.

A global random seed of 42 was set for reproducibility. Evaluation metrics include accuracy (Acc), precision (P), recall (R) and F1-score (F1). For precision, recall, and F1-score, their macro-averages and micro-averages are the same (and therefore presented) since the classes are perfectly balanced, each having 6,000 training and 1,000 testing samples.

A. *Results on Cross Validation*

Cross-validation ensures reliable assessment of the model's generalization performance by systematically training and evaluating on diverse subsets of the dataset. FashionCNN demonstrates superior performance across all metrics in a 5-fold cross-validation, outperforming XGBoost and Random Forest, while FashionMLP performs the least effectively. Based on these results, FashionCNN is selected as the best model for further fine-tuning.

Both Random Forest and XGBoost undergo fine-tuning, adjusting parameters like the number of estimators, maximum tree depth, and minimum samples for splitting or leaf nodes to enhance predictive power. Although PCA was considered for feature preprocessing to reduce dimensionality, it did not yield a significant accuracy increase, leading to the decision to retain models without PCA in the feature preprocessing pipeline.

TABLE I. CROSS VALIDATION RESULTS

Models	**Acc**	**P**	**R**	**F1**
Random Forest	0.8818	0.8809	0.8818	0.8803
XGBoost	0.9045	0.9039	0.9045	0.9038
FashionMLP	0.8619	0.8623	0.8619	0.8605
FashionCNN	**0.9605**	**0.9606**	**0.9605**	**0.9604**

Note that FashionMLP merely serves as a baseline and is not exactly of our interest. This is because it is incapable of modeling nonlinear relationships, while such relationships indeed exist according to our EDA.

However, capturing only linear patterns, it achieved an accuracy of 86.19% on the test set. This indicates that the nonlinear relationships that Random Forest and XGBoost attempted to model were not effective, as they are only slightly better than FashionMLP, especially given the staggering amount of time to train these complex models.

B. *Hyperparameter Tuning for FashionCNN*

The goal of this experiment for FashionCNN is to systematically search and optimize the model's hyperparameters, such as learning rate and batch size, to enhance its performance on image classification. This process aims to find the most effective configuration that maximizes the model's accuracy, ensuring optimal learning and representation of features for improved classification results.

Fig 4 Confusion Matrix for FashionCNN

Table II represents the results of fine-tuning FashionCNN with cross-validation across various configurations of learning rates and batch sizes. The evaluation metrics are averaged over 5 folds. Note that, for all settings, the number of epochs is set to 500 and the number of early stopping epochs to 50. The model usually converges within 100 epochs.

TABLE II. HYPERPARAMETER TUNING FOR FASHIONCNN

Learning rate	**Batch size**	**Acc**	**P**	**R**	**F1**
0.001	100	0.8878	0.8883	0.8878	0.8870
0.0005	10005	0.9484	0.9485	0.9484	0.9482
0.00001	100	0.9589	0.9588	0.9589	0.9588
0.001	150	0.9017	0.9025	0.9017	0.9012
0.0005	150	0.9625	0.9626	0.9625	0.9624
0.00001	150	0.9608	0.9608	0.9608	0.9607
0.001	200	0.9134	0.9139	0.9134	0.9130
0.0005	200	**0.9665**	**0.9666**	**0.9665**	**0.9665**
0.00001	200	0.9593	0.9593	0.9593	0.9592

According to our experiment, it is evident that the FashionCNN model performs the best with the learning rate set to 0.0005 and batch size set to 200. We were able to reach an accuracy of 96.65%.

C. *Results on Test Set*

Finally, we evaluated all models on the official test set to gain insights into the generalizability of our models and comparability with other existing work. The results are displayed in Table III. Note that all the models were fine-tuned and FashionMLP was excluded as it is not of interest as a candidate model.

TABLE III. TEST RESULTS

Models	**Acc**	**P**	**R**	**F1**
Random Forest	0.8761	0.8750	0.8761	0.8746
XGBoost	0.8980	0.8973	0.8980	0.8973
FashionCNN	**0.9216**	**0.9210**	**0.9216**	**0.9211**

As we can see, the FashionCNN model continues to stand out as the best across all metrics on test performance.

D. *Model Interpretability*

The goal for model interpretability is to enhance our understanding of how models make predictions. This involves uncovering the factors and features that contribute to a model's decision-making process, providing insights into their internal mechanisms.

Figure 5 depicts the top 10 most important features in XGBoost. Pixels with the index of 337, followed by 346 and 471, tend to possess the most discriminative signals for this classification task.

Figure 6 provides the feature maps of two training samples, shedding light into the internal decision process of the convolution layers in FashionCNN. Feature maps are representations of learned patterns and features extracted from input images as they pass through different layers of the network. Each layer corresponds to specific filters that detect increasingly complex features.

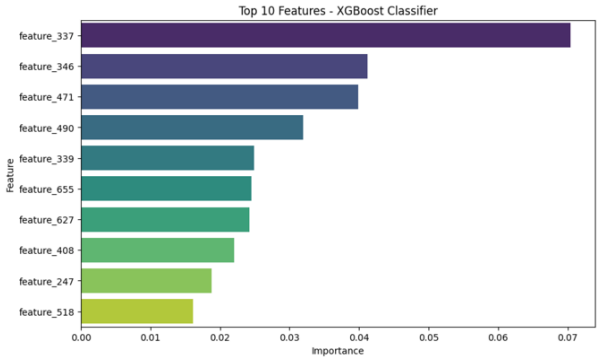

Fig 5 Top 10 Features Based on XGBoost Feature Importance

IV. Conclusions

A. *Analytical Results*

According to PCA and t-SNE, we discovered that the data has nonlinearity. Therefore, our study compared three nonlinear models—XGBoost, Random Forests, and FashionCNN—for FashionMNIST classification. FashionCNN excelled in capturing intricate patterns in images and achieved the best classification results, an accuracy of 92.16%. Random Forests and XGBoost, on the other hand, were ineffective in modeling nonlinear relationships within the data.

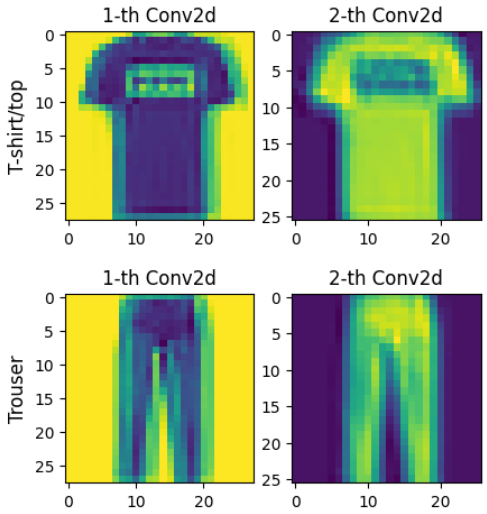

Fig 6 Examples of FashionCNN Feature Maps

B. *Business Implications for a Clothing Company*

This work presents interesting practical implications for businesses. Clothing companies can create image-based search functionalities on their websites or applications. Customers can upload or search for specific clothing items based on images, enhancing their ability to find desired products and thus boosting sales.

Additionally, image classification models can be beneficial for marketing analytical purposes. They can automatically tag, categorize, and analyze large amounts of online visual marketing content. The results can later be useful for downstream tasks, such as recommending visual content for marketing campaigns or monitoring competitor's marketing strategies.

V. References

[1] J. Luo, " Comparison of Different Models for Clothing Images Classification Studies," in AIAM2020: Proceedings of the 2nd International Conference on Artificial Intelligence and Advanced Manufacture, October 2020.

[2] J. Kim, M. Z. Bashir, and F. Cavallo, " Image Classification Using Multiple Convolutional Neural Networks on the Fashion-MNIST Dataset," in Sensors, Dec. 2022.

[3] Gadri, S., Neuhold, E. (2020). Building Best Predictive Models Using ML and DL Approaches to Categorize Fashion Clothes. In: Rutkowski, L., Scherer, R., Korytkowski, M., Pedrycz, W., Tadeusiewicz, R., Zurada, J.M. (eds) Artificial Intelligence and Soft Computing. ICAISC 2020.

VI. Published Post

Code base for this study can be found here.
A separate blogpost is dedicated to this work as found here.