Classification Models for the Fashion MNIST Dataset

Abeba Bade
Texas A&M University
abeba.bade@tamu.edu

Hussein Mansour
Texas A&M University
hussein.mansour@tamu.edu

Tobi Oladunjoye
Texas A&M University
oooladu2@tamu.edu

Abstract: This report details the development and implementation of Convolutional Neural Networks (CNNs) and non-neural network models (NNNs) to classify clothing articles using the famous FashionMNIST dataset and determine the optimal approach for classifying this dataset. The experiment explored MLP, SVM, Random Forest, kNN, Gradient Boosting Machines and two CNN machine learning modes to classify the images into distinct clothing categories. The two CNN models, 4Convolutional Layers, 4Fully Connected Layers (4XConv-layers 4XfC-layers(CNN-1) and 5Convolutional Layers, 5Fully Connected Layers(5XConv-layers 5XfC-layers(CNN-2) were evaluated to improve the classification.

CNN-2 was selected as the best model for FashionMNIST dataset classification since it outperformed all the other models in all areas of metrics evaluation, showing an overall accuracy of 91.5% on the testing dataset. Recall, F1-score, and computational efficiency were metrics utilized to evaluate the performance to select this optimal algorithm. Even though its architecture may offer CNN-2 for a better and more detailed feature extraction, it also risked overfitting due to its complexity as compared to CNN-1. Although CNN2 seemed to have a more complex architecture with additional layers, the performance improvement from CNN-1, which had an overall accuracy of around 90.9% on the test, to CNN-2 wasn't substantial.

GBM models also demonstrated competitive performance, achieving accuracy levels around 85% to 89% on the FashionMNIST dataset. While not as high performing as CNNs, other NNNs showed good precision and recall across different classes. The results of the confusion matrices highlighted some deviations in classification, especially for several models in the Shirts category, indicating that accurate classification of this clothing type (Figure 3):

Introduction: This task details the use of CNN and NNN algorithms to classify items of clothing from the FashionMNIST dataset. The data set is made up of 70000 images (each28x28px in grayscale) distributed equally amongst 10 different classes: T-shirt/Top, Trouser, Pullover, Dress, Coat, Sandals, Bags, and Ankle Boots. The diverse collection of classes allows for a comprehensive challenge to machine learning methodologies.

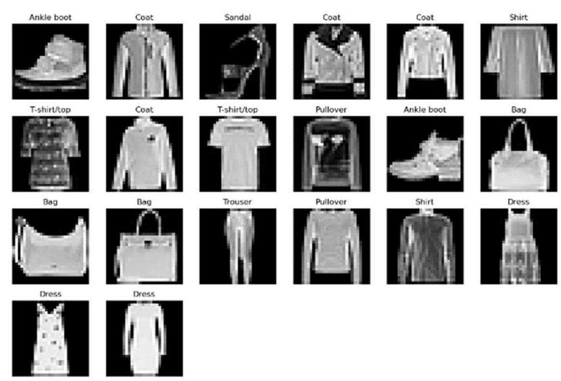

Fig 1: Sample class visualization from FashionMNIST dataset

This dataset has gained significant popularity for its role as a standard benchmark in testing the efficacy of image classification algorithms, especially CNNs [14]. Due to the complex nature of clothing properties leading to resemblance between different categories, optimal classification for this dataset is challenging [2]. Clothing item classification serves as a fundamental task in the realm of computer vision[1].

Many research papers suggested CNN-based architecture as inherently suitable for image classification[1,2,3]. In machine learning, focusing on a single perfect model makes it harder to understand the details of the dataset and how different methods classify clothes. Exploring different models also lets us think about combining them to improve overall performance. This way of approaching this task makes our classification system stronger and helps us learn more about the patterns in the FashionMNIST dataset. Thus, besides two CNN models, five other NNNs algorithms were employed in order to compare and contrast their evaluations on the dataset: MLP, SVM, RF, KNN and GBM.

RFs generally do well in performing image classification tasks for datasets like FashionMNIST. They also create an ensemble of decision trees, making them suitable for handling complex relationships within images. They are less prone to overfitting and can manage high-dimensional image data effectively[8]. GBM and KNN algorithms are also powerful in image classification [9,10]. GBMs tend to improve the classification accuracy iteratively and construct an ensemble of weak learners, often decision trees, to create a strong classifier, making it adept at capturing patterns[9]. Even though KNNs can be used for image classification by measuring similarities between images, it may face challenges with larger datasets due to its computational intensity[10]. MLPs excel in learning complex image features through their multiple layers[6]. They are commonly used due to their ability to capture complex patterns. SVMs work well with moderate-sized datasets and can handle high-dimensional image data efficiently [7]. SVMs are versatile and can adapt to nonlinear image data using kernel functions[7]

The complex structure of the Fashion MNIST dataset is challenging to classify. These difficulties arise due to their high-dimensional nature, distortions in image variability and quality, and similarities between clothing categories. CNNs are particularly well-suited for classifying such datasets due to their ability to effectively capture spatial hierarchies, feature extraction, capabilities and patterns within images. Incorporating Principal Component Analysis (PCA), reducing the dimensionality of the dataset by identifying essential features, with CNN can simplify computations and potentially enhance model generalization.

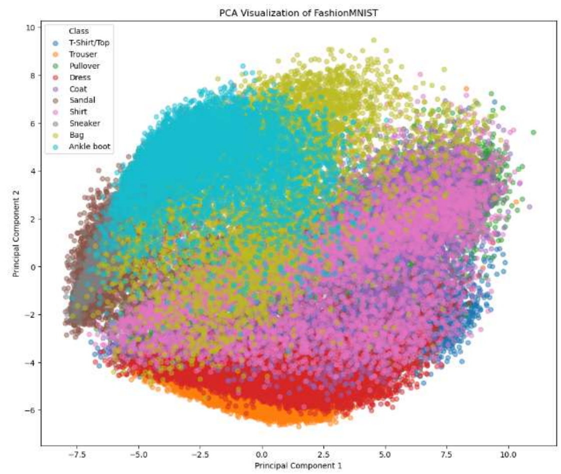

Fig 2 : PCA Visualization for the FashionMNIST dataset

Methods/Experiments - 5xConv-5xFC(CNN-2):

Out of the seven models evaluated, 5xConv - 5xFC(CNN-2) was selected as the best model and implemented for classifying the FashionMNIST dataset. The methodology for classifying the FashionMNIST Dataset while evaluating each model includes initialization transformation, normalization, data loading, splitting the dataset into training and test sets, model training in a training set, hyperparameter tuning, cross-validation, and comprehensive performance evaluation metrics. For CNN-2, the following steps were followed to execute the task of classification.

Feature extraction is done to reduce the spatial dimensions of the feature map followed by introducing ReLU to help capture more complex patterns in the data. Each of the five convolutional layers which are the core of CNN architecture, are equipped with a set of learned filters. These filters are adept at extracting a wide range of features from the input images, from basic edges and textures to more complex patterns. As the data progresses through these layers, the model effectively captures hierarchical features, with each subsequent layer building upon the previous ones to extract increasingly abstract features. This hierarchical feature extraction is key to understanding the complex details within the FashionMNIST images.

Advanced pooling and interpolation were incorporated. Following each convolutional layer, max pooling is employed to reduce the spatial dimensions of the feature maps. This decreases the computational load while also ensuring the most prominent features are retained for further processing. The Rectified Linear Unit (ReLU) activation function introduces non-linearity to the network, enabling the model to learn complex patterns and representations from the image data.

Comprehensive Fully Connected Layers were added for processing abstracted features, which serves as decision-making part of the network. These dense layers interpret and classify features, extracted, and abstracted by the convolutional layers, into the corresponding clothing categories.

To mitigate the risk of overfitting, dropout regularization was implemented, since this technique randomly deactivated certain neurons during training, encouraging the model to learn more robust and generalized features.

Then hyperparameters were tuned by a grid search approach to find the optimal combination of learning rate, batch size, and dropout rate. This involved iterating over predefined values for these parameters and evaluating the model's performance to identify the most effective configuration.The best performing grid_parameters {'dropout_rate': 0.2, 'learning rate': 0.01, 'batch_size': 16} in the validation dataset were used in the final training of the Model.

The final step involved training the model on the training set using the best performing grid_parameters. The training loop, which was encapsulated involves iterating through the FashionMNIST dataset, computing predictions(Forward Pass**),** calculating the cross-entropy loss, which computes the discrepancy between the predicted outputs and the true labels. Then executing backpropagation was executed(Backward Pass**)** to update the model weights and minimize the loss . Data augmentation techniques like random horizontal flipping, normalization, and random erasing are applied to diversify the dataset and improve the model's generalization ability. The optimizer updates the weights of the network based on the computed gradients at the **Optimizer Step.** The classification_report() function is used to display precision, recall, and F1-scores for each class as well as overall accuracy.

The classification output is obtained after training the model, where the network properly classifies each input image into one of the predefined classes of the FashionMNIST dataset. This classification is based on the network and its ability to interpret and synthesize its complex properties and patterns. learned through several layers.

Results/ Conclusion:

The project compares the performance of diverse machine learning models for clothing classification using the FashionMNIST dataset. Out of the seven models evaluated, CNN-2 was selected as an optimal choice for classifying FashionMNIST dataset. This was because CNN-2 achieved the highest overall accuracy(91.5%), F1- score(91.6%), precision(91.8%), and recall(91.5) on the testing dataset as compared to the NNNs which achieved overall accuracy ranging from 84.6% to 89.3% on the testing data. These findings show the varying strengths and weaknesses of each model in classifying FashionMNIST clothing items.

Metrics	Machine Learning Models						
	MLP	**SVM**	**kNN**	**RF**	**GBM**	**CNN-1**	**CNN-2**
Accuracy	0.864	0.880	0.846	0.874	0.893	0.909	0.915
F1-score	0.864	0.879	0.847	0.873	0.892	0.910	0.916
Precision	0.864	0.880	0.849	0.873	0.892	0.912	0.918
Recall	0.864	0.879	0.846	0.874	0.893	0.909	0.915

Table I: Performance Metric Comparison on FashionMNIST Testing Dataset

The Model did well in classifying Ankle boots. Bags and Sneakers. This is due to the nature of the features of the images in those classes. Observations from confusion matrices highlighted certain outliers in classification performance especially in the top wears. T-Shirts were often misclassified as Shirts, as well as Pullovers, Dresses and Coats. This is attributed to the fact that those garments exhibit patterns that are similar to those found in the shirts making it difficult to generalize possible variations of those features due the sensitive nature of CNNs with regards to feature selection.

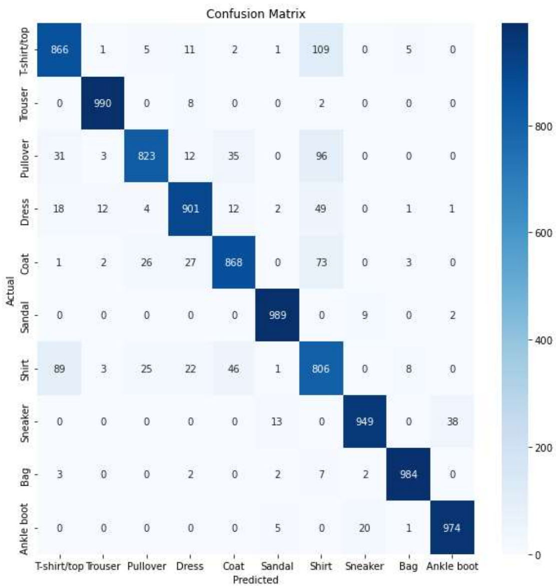

Fig 3: Confusion Matrix for the CNN-2 testing set.

For optimal choice in classifying FashionMNIST clothing items, the CNN-2 model CNNs are highly recommended due to their superior performance, especially in handling image data. They have shown consistent accuracy, good precision, and recall, making them well-suited for this image classification task. However, if computational resources are limited or if CNNs are not feasible due to complexity, Gradient Boosting Machines can serve as a viable alternative, offering decent performance on this dataset.

Classifying items in a clothing company can lead to better and improved efficiency in various aspects of business for the company. Classification models can be used to automatically identify and categorize items in the inventory. It does this by analyzing the visual features and when this analysis is done, the system can efficiently sort and catalog products, which reduces the need for manual intervention in the inventory management process.

References :

[1] M. Kayed, A. Anter and H. Mohamed, "*Classification of Garments from Fashion MNIST Dataset Using CNN LeNet-5 Architecture,*" 2020 International Conference on Innovative Trends in Communication and Computer Engineering (ITCE), Aswan, Egypt, 2020, pp. 238-243, doi: 10.1109/ITCE48509.2020.9047776.

[2] W. Hu, Y. Huang, L. Wei, F. Zhang and H. Li, *"Deep convolutional neural networks for hyper spectral image classification"*, Journal of Sensors, 2015.

[3] Z. H. A. N. G. Jianming, Z. H. A. N. Zhicai, C. H. E. N. G. Keyang and Z. Yongzhao, "*Review on development of deep learning", Journal of Jiangsu university: natural science editions*, vol. 36, no. 2, pp. 191-200, 2015.

[4]AlexNet: Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). *ImageNet classification with deep convolutional neural networks. Communications of the ACM,* 60(6), 84-90.

[5] Bishop, C. M. (2006). *Pattern recognition and machine learning.* Springer.

[6]R. Arora, S. S. Khanam, and S. Khan, "*A review on deep learning techniques for image classification,*" in IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), 2019.

[7] A. A. Hassanien, S. A. Ghaleb, and A. Abraham, "*Support Vector Machines Applications in Image Recognition and Classification,*" in IEEE International Conference on Intelligent Computing and Intelligent Systems (ICIS), 2009.

[8] Y. Zhang, H. Li, and Y. Wang, "*Application of Random Forest Algorithm in Image Classification*," in IEEE International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), 2018.

[9] X. Yuan, M. Zhang, and L. Han, "*Scalable and Efficient Gradient Boosting Machine for Image Classification*," in IEEE Transactions on Multimedia, vol. 22, no. 6, pp. 1560-1573, 2020.

[10] T. T. Hoang and P. T. Thanh, *"Image Classification Using K-Nearest Neighbor Classifier,"* in IEEE International Conference on Advanced Technologies for Communications (ATC), 2016.

[11] A. Krizhevsky, I. Sutskever, and G. Hinton, *"ImageNet Classification with Deep Convolutional Neural Networks,"* in Advances in Neural Information Processing Systems, vol. 25, pp. 1097-1105, 2012.

[12] K. Simonyan and A. Zisserman, "*Very Deep Convolutional Networks for Large-Scale Image Recognition,"* arXiv preprint arXiv:1409.1556, 2015.

[13] C. Szegedy et al., *"Going Deeper with Convolutions,"* in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-9, 2015.

[14] Xiao, Han, Kashif Rasul, and Roland Vollgraf. "*Fashion-MNIST: A Novel Image Dataset for Benchmarking Machine Learning Algorithms.*" arXiv preprint arXiv:1708.07747 (2017).