Fashion MNIST Classification through CNN and ResNet-inspired Architectures

Technical report

Author: MERIDJA Nabil

Master's degree in Artificial intelligence
Computer Vision & Deep learning
Spring 2024

Table of content:

1. Motivation and rationale

This project is about classification applied to the dataset Fashion MNIST. We will compare two types of models for classification, a CNN and a ResNet-inspired architecture: Mini ResNet. This classification solves a significant problem, because it permits to improve the clothing items classification, which is essential for e-commerce softwares and inventory management. By comparing the performances between these two models, our project aims to identify the best model to have more accurate classifications for real world applications.

2. State of the art

At this point, CNN networks have been largely used to classify the Fashion MNIST dataset. We can have a glance at the performance report for different CNN architectures [1].

Reference	Method	Activation Function	Optimizer	Batch size	Epoch	Layers	Dropout	Learning rate	Training accuracy	Testing accuracy
Bhatnagar et al. (2017)	CNN with Batch normalization and Residual skip	NA	NA	NA	NA	2 Convolution 2 fully connected	NA	NA	NA	92.54%
Bhatnagar et al. (2017)	CNN	SVM	NA	NA	NA	2 Convolution 2 fully connected	NA	NA	NA	90.72%
Bhatnagar et al. (2017)	CNN	softmax	NA	NA	NA	2 Convolution 2 fully connected	NA	NA	NA	91.86%
Bhatnagar et al. (2017)	CNN with Batchnor-malization	NA	NA	NA	NA	2 Convolution 2 fully connected	NA	NA	NA	92.22%
Manessi & Rozza, 2018	KerasNet	aff({id,ReLU, tanh})	RMSprop	NA	NA	4 convolution 2 fully connected	NA	0.0001	NA	94.41%
Manessi & Rozza, 2018	KerasNet	id	RMSprop	NA	NA	4 convolution 2 fully connected	NA	0.0001	NA	90.51%
Manessi & Rozza, 2018	KerasNet	ReLU	RMSprop	NA	NA	4 convolution 2 fully connected	NA	0.0001	NA	90.79%
Manessi & Rozza, 2018	KerasNet	Tanh	RMSprop	NA	NA	4 convolution 2 fully connected	NA	0.0001	NA	93.43%
Manessi & Rozza, 2018	KerasNet	LReLU	RMSprop	NA	NA	4 convolution 2 fully connected	NA	0.0001	NA	91.13%
Manessi & Rozza, 2018	KerasNet	conv({id,ReLU})	RMSprop	NA	NA	4 convolution 2 fully connected	NA	0.0001	NA	92.39%
Manessi &	KerasNet	conv({id,tanh})	RMSprop	NA	NA	4 convolution	NA	0.0001	NA	93.64%

figure 1: Comparing CNN Architectures for Fashion-MNIST

However, fewer examples of ResNet used on Fashion MNIST can be found, this article [2] used Mini ResNet and achieved the following results

Metric	Test Set	Validation Set	Training Set
Loss	0.1954	0.2226	0.1541
Accuracy	92.95%	92.28%	94.40%
Precision	93.08%	92.44%	94.83%
Recall	92.95%	92.28%	94.77%
F1 Score	92.98%	92.33%	94.78%

figure 2: Mini ResNet classification results

As this article is short on information, we will try to reproduce the results and tune the parameters to improve performance.

3. Objectives

The general objective is to have a performant model for classification on fashion items. To do so, we will follow an article's work [2] to try to reproduce its results and pick the best model for a real world application.

4. Methodology

- **Dataset**

I used for this project the Fashion MNIST dataset. It consists of 70 000 grayscale images, labeled in 10 classes representing different fashion items. It contains 60 000 images for training and 10 000 for testing. Each image measures 28x28 pixels and has been normalized to improve the models' convergence.

- **Librairies**

I also used the following libraries:

- PyTorch: Deep learning library to create and train neural networks
- Matplotlib and seaborn: Data visualization and performance comparisons of models using histograms and confusion matrix
- Scikit-learn: Classification report and metrics computation

- **Algorithms**

The first model used is the Convolutional Neural Network. CNNs are distinguished from other neural networks by their superior performance with image. They have three main types of layers, convolutional layer, pooling layer and fully-connected layer.

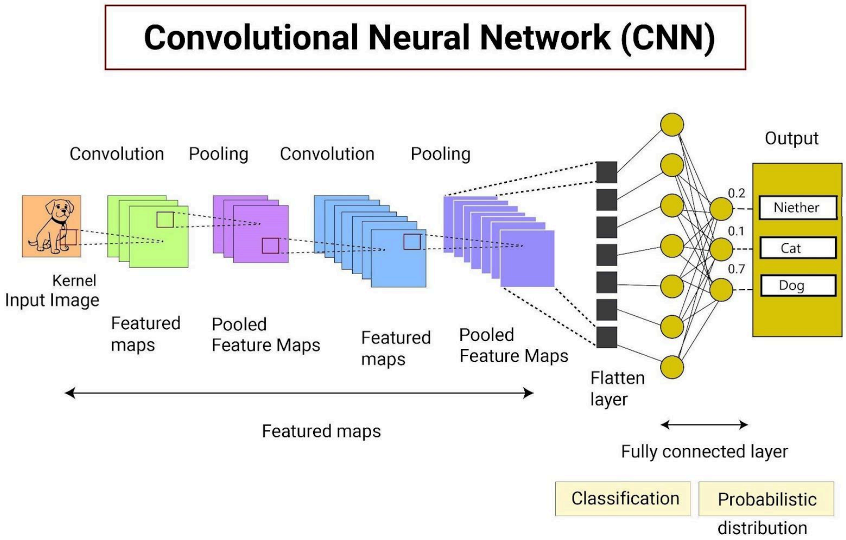


figure 3: Architecture of a CNN

The other model used is Mini ResNet, an architecture derived from the ResNet. ResNet is a type of CNN designed to tackle the issue of vanishing gradients. Its architecture enables the network to learn multiple layers of features without getting stuck in local minima, a common issue with deep networks. Here, our Mini ResNet is a simplified version that uses fewer residual blocks, making it lighter and faster while retaining the benefits of residual connections. This approach allows us to efficiently classify images from the FashionMNIST dataset while maintaining good performance.

- **Metrics**

I used different tools to measure the performance of the models

- Confusion matrix: number of correct predictions for each pair
- Accuracy: proportion of correct predictions made by the model across the entire dataset
- Precision: proportion of true positive predictions among all positive predictions made by the model
- Recall: proportion of true positive predictions among all actual positive instances
- F1 Score: harmonic mean of precision and recall

- **Loss function**

To train the models, I used the cross entropy loss function, it is particularly well-suited to multi-class classification.

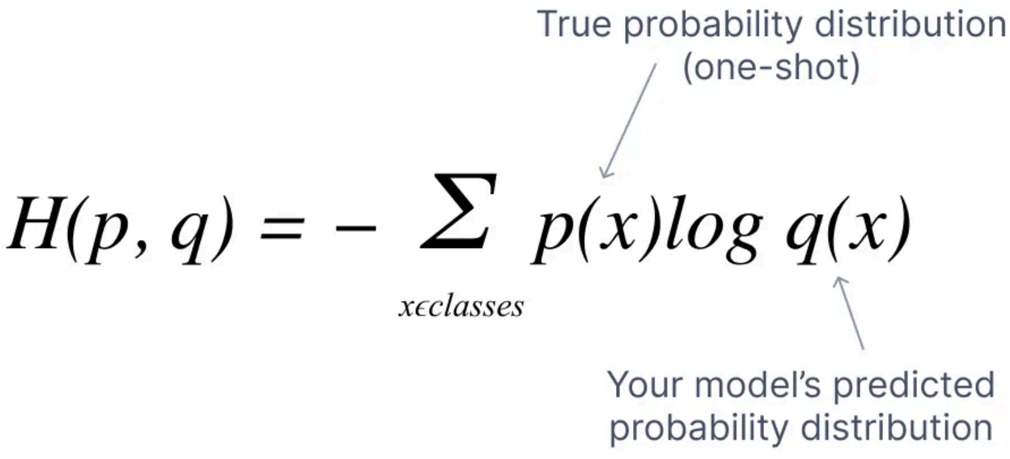


figure 4: Cross entropy

- **Learning rate**

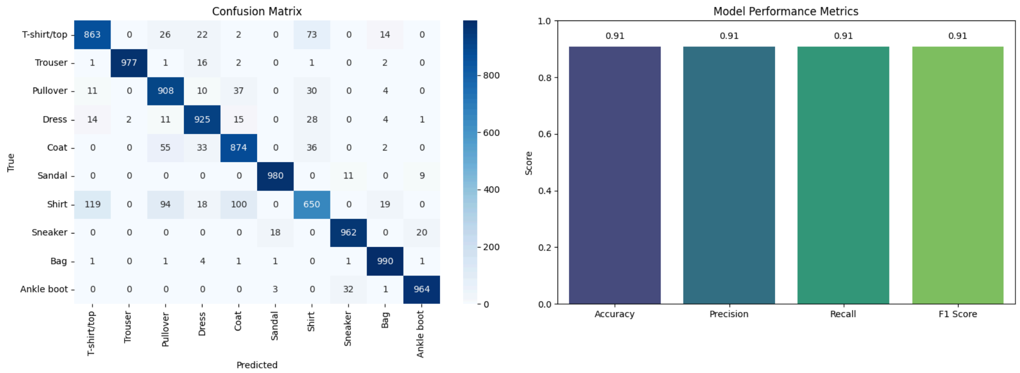
I set a learning rate of 0.001 which is commonly used and offers a good compromise, ensuring stable convergence without skipping the global minimum.

- **Epoch**

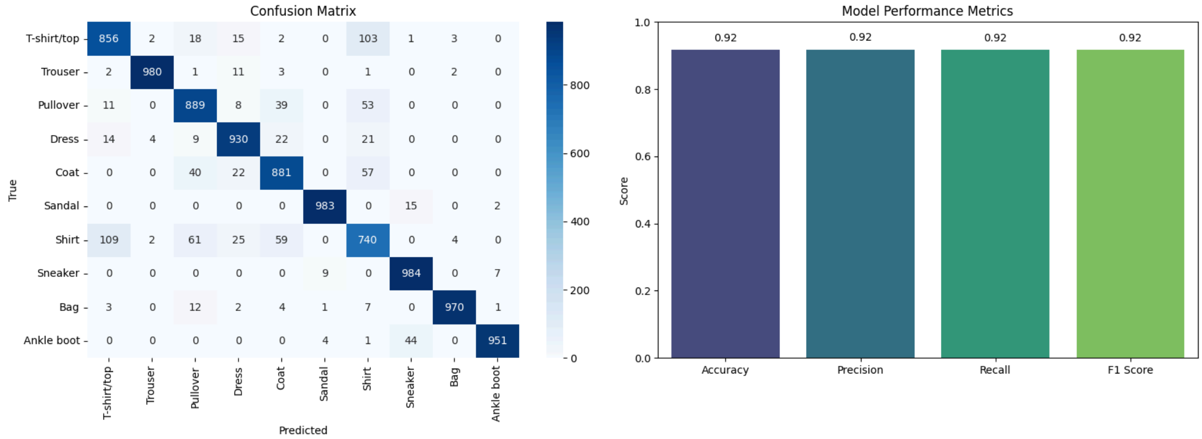
I used 10 epochs for the training, to allow the model to learn enough, keep the training time on my machine reasonable and limit overfitting.

5. Results

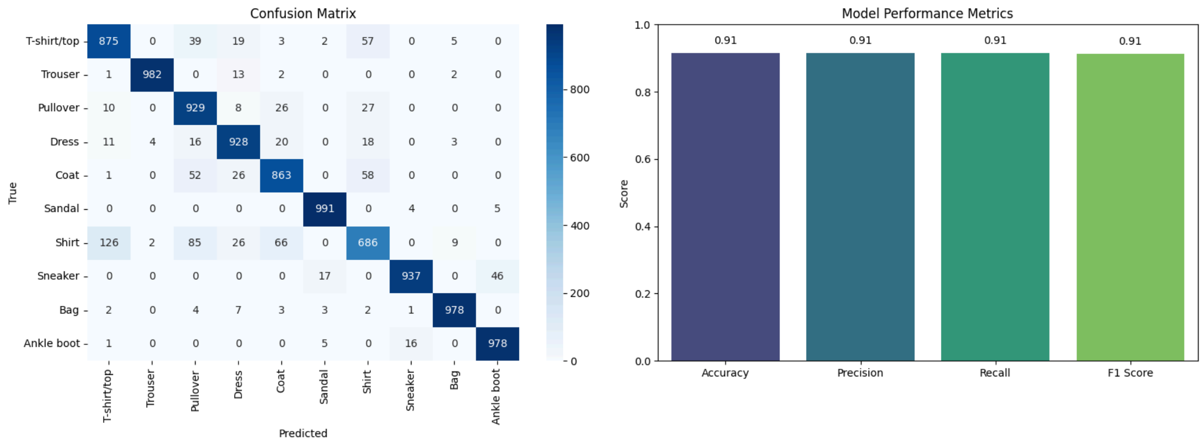
Now let's compare the results from the different models:
CNN test 1: Two convolutional layers with 32 and 64 filters, followed by a dense layer of 64 neurons. (Loss = 0.1945)



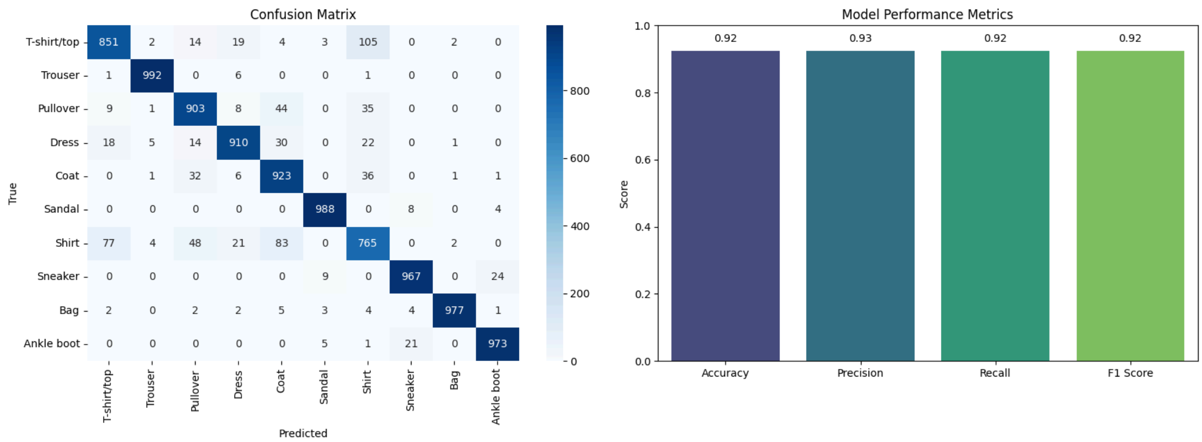
CNN test 2: Two convolutional layers with 64 and 128 filters, followed by a dense layer of 64 neurons. (Loss = 0.1383)



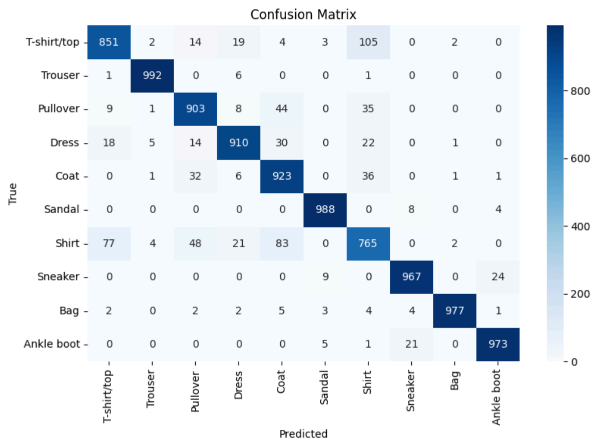
CNN test 3: Two convolutional layers with 64 and 128 filters, followed by a dense layer of 32 neurons. (Loss = 0.2102)

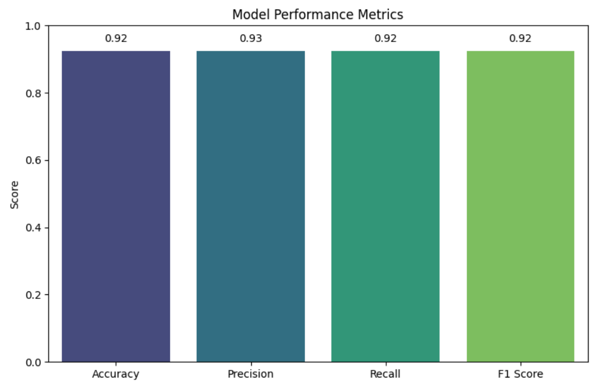


Mini ResNet test 1: 3 residual blocks with 64 filters. (Loss = 0.0856)



Mini ResNet test 2: 4 residual blocks with 64 filters. (Loss = 0.0867)





We can also plot some images with the prediction and the ground truth (here with *Mini ResNet test 2* model). We see that every prediction is correct.

6. Conclusion

To conclude, we succeeded in reproducing the article's [2] results. For a real world application in fashion item classification, I would pick the second Mini ResNet (*4 residual blocks with 64 filters*) model because it has shown the best metrics. For possible future improvements, we could use augmentation techniques on the dataset or try different model architectures with more complex models.

7. Bibliography

[1] *CNN Model for Image Classification on MNIST and Fashion-MNIST Dataset, Shivam S. Kadam, Amol C. Adamuthe*, and Ashwini B. Patil*

https://bhu.ac.in/research_pub/jsr/Volumes/JSR_64_02_2020/51.pdf

[2] *Fashion MNIST Classification : Enhancing Fashion MNIST Classification through Optimized CNN and ResNet-inspired Architectures, Sai Nihith*

https://medium.com/@sainihith0130/fashion-mnist-classification-enhancing-fashion-mnist-classification-through-optimized-cnn-and-84a74eb2d3a1