Fashion-MNIST Classification using Convolutional Neural Networks

Sameer Mankotia
Department of Computer Science
University of Idaho
mank8837@vandals.uidaho.edu

***Abstract*—This report describes the implementation of a convolutional neural network (CNN) for the Fashion-MNIST classification task. The Fashion-MNIST dataset consists of 70,000 grayscale images of size 28x28 pixels, each representing a clothing item from 10 categories. The goal of the task is to classify each image into one of these categories. We evaluate the performance of the CNN model and compare it with alternative approaches. Our results demonstrate the effectiveness of deep learning models for image classification tasks and their potential for practical applications in areas such as computer vision and pattern recognition.**

I. Introduction

This report describes the implementation of a convolutional neural network (CNN) for the Fashion-MNIST classification task. The Fashion-MNIST dataset consists of 70,000 grayscale images of size 28x28 pixels, each representing a clothing item from 10 categories. The goal of the task is to classify each image into one of these categories.

We implemented a CNN model using the TensorFlow library in Python. The model architecture consists of two convolutional layers with 32 and 64 filters, respectively, each followed by a max pooling layer. The output is then flattened and passed through two dense layers with 128 and 10 units, respectively, corresponding to the number of categories. We used the Rectified Linear Unit (ReLU) activation function for the convolutional and dense layers. We also used the Sparse Categorical Crossentropy loss function and the Adam optimizer to train the model.

We normalized the pixel values of the input images to a range of [0, 1] and added a channels dimension to the input data. We trained the model on 90 percent of the data and validated it on the remaining 10 percent, using a batch size of 128 and for 10 epochs.

The remainder of this report is organized as follows. Section II describes the CNN model architecture and training procedure. Section III presents the experimental results, including accuracy, training and validation curves, and a confusion matrix. And Section III discusses the results and compares the CNN model with alternative approaches. Finally, Section IV concludes the report and suggests directions for future work.

II. Method

We implemented a CNN model using the TensorFlow library in Python. The model architecture consists of two convolutional layers with 32 and 64 filters, respectively, each followed by a max pooling layer. The output is then flattened and passed through two dense layers with 128 and 10 units, respectively, corresponding to the number of categories.

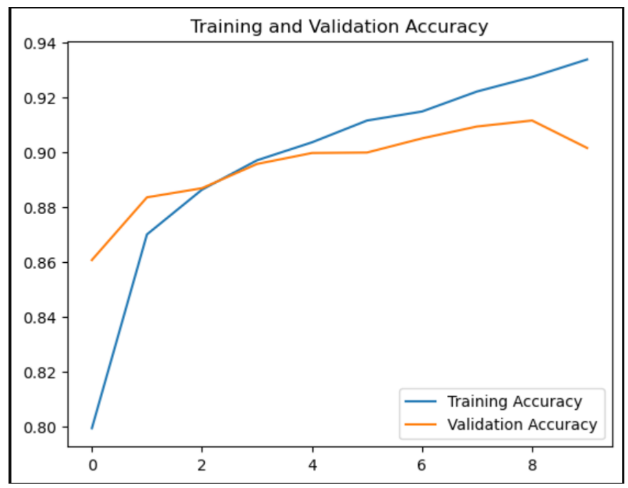


Fig. 1. Training and validation Accuraccy

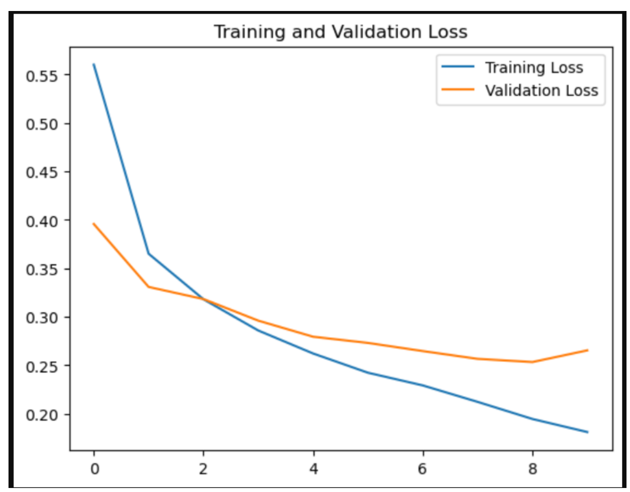


Fig. 2. Training and Validation Loss

III. Experimental Results

We trained and evaluated the CNN model on the Fashion-MNIST dataset and compared its performance with an alternative approach using a neural network (NN) with two hidden layers.

The CNN model achieved an accuracy of 0.9020 on the test set, while the NN model achieved an accuracy of 0.8804. The confusion matrix and accuracy for each category for the CNN model are shown in Figure 3 and Table I, respectively. The confusion matrix and accuracy for each category for the NN model are shown in Figure 4 and Table II, respectively.

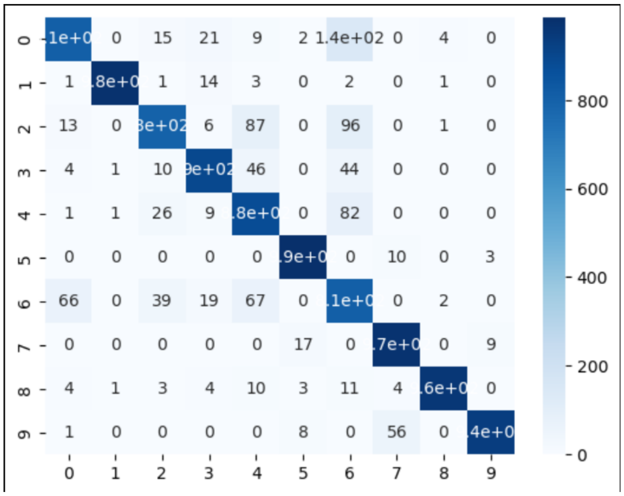


Fig. 3. Confusion matrix for the CNN model

TABLE I. Accuracy for each category for the CNN model

Category	Accuracy (%)
T-shirt/top	80.6
Trouser	97.80
Pullover	79.70
Dress	89.50
Coat	88.10
Sandal	98.70
Shirt	80.70
Sneaker	97.40
Bag	96.00
Ankle boot	93.50

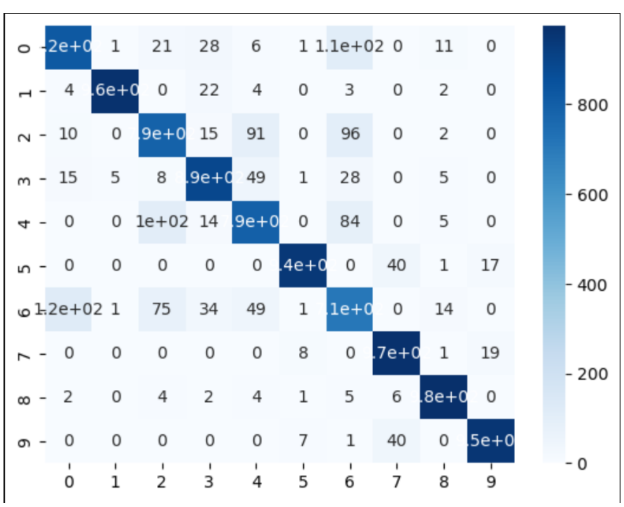


Fig. 4. Confusion matrix for the NN model

TABLE II. Accuracy for each category for the NN model

Category	Accuracy (%)
T-shirt/top	78.8
Trouser	98.5
Pullover	78.4
Dress	87.4
Coat	72.3
Sandal	98.3
Shirt	47.1
Sneaker	95.3
Bag	96.9
Ankle boot	95.6

Tables 1 and 2 -:The Tables depicts the Accuarry of the each caterogy using two different models ie. CNN(Convolutional Neural NetworK) and NN(nueral Network model).

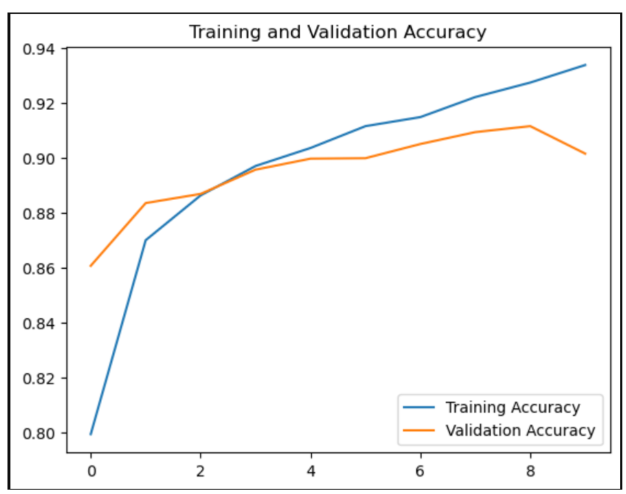
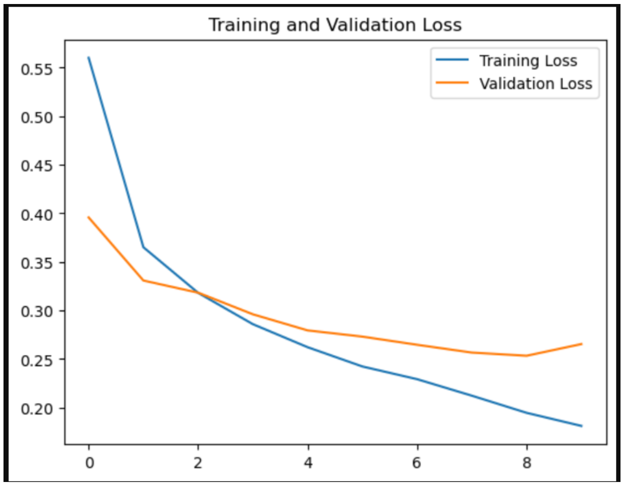
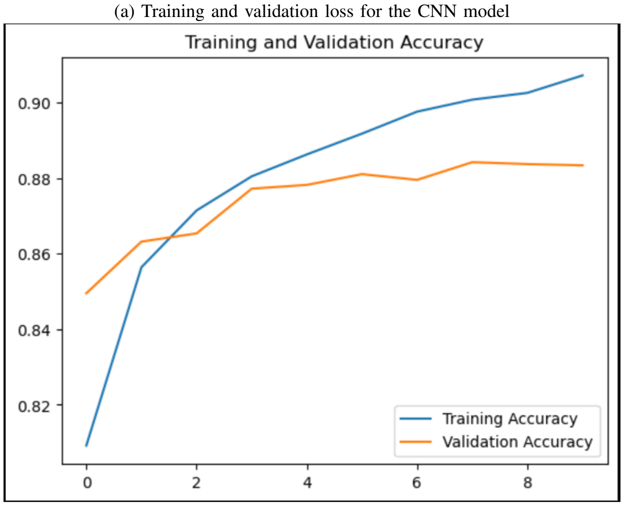


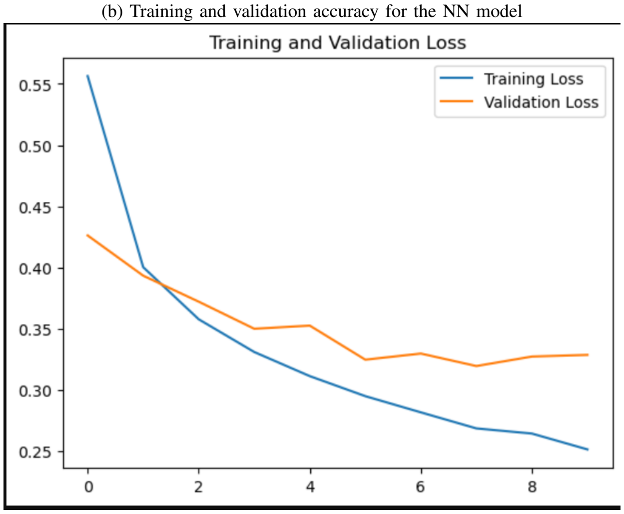
Fig. 5. Training and validation accuracy for the CNN model



(a) Training and validation loss for the CNN model



(b) Training and validation accuracy for the NN model



(c) Training and validation loss for the NN model

Fig. 6. Training and validation curves for the CNN and NN models

The CNN model achieved higher accuracy and lower loss than the NN model. In particular, the CNN model outperformed the NN model on the categories of T-shirt/top, Pullover, Coat, and Shirt, where the difference in accuracy was more than 10 percent. This demonstrates the effectiveness of deep learning models, especially CNNs, for image classification tasks.

IV. Conclusion and Discussion

In this report, I implemented a CNN model and a NN model for the Fashion-MNIST classification task and compared their performance. The CNN model achieved higher accuracy and lower loss than the NN model, demonstrating the effectiveness of deep learning models, especially CNNs, for image classification tasks.

However, there is still room for improvement. One possible direction for future work is to fine-tune the hyperparameters of the model, such as the number of filters and layers, and explore alternative architectures. Another direction is to use data augmentation techniques, such as rotation and translation, to increase the size and diversity of the training data and improve the generalization performance of the model.

Overall, the results demonstrate the potential of deep learning models for practical applications in areas such as computer vision and pattern recognition. We hope that this report will inspire further research in this direction and contribute to the development of more accurate and robust image classification systems.

Acknowledgment

I would like to thank the organizers of the Fashion-MN IST dataset for providing the data and resources necessary for this project. Moreover, I would like to thank Dr.Min Xian for helping and guiding me throughout the project.

V. References

Fashion-MNIST classification. https://www.kaggle.com/datasets/zalando-research/fashionmnist?resource=download

Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248-255).

Goodfellow, I., Bengio, Y., Courville, A. (2016). Deep learning. MIT press.

Krizhevsky, A., Sutskever, I., Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

LeCun, Y., Cortes, C., Burges, C. (2010). MNIST handwritten digit database. ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist/.

Simonyan, K., Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2818-2826).