ANN vs CNN for Fashion Image Classification

Name: Sivanesh Muruganandham
Student ID: P2911313

Overview of the Project

This paper is aimed at the design of an ANN and a CNN for classifying images of different types of garments from a Fashion-MNIST dataset comprising 70,000 grey-scaled images with a size of 28 × 28 pixels across ten categories such as shirts, trousers, dresses, coats, and sneakers. The main task was to identify which of these two models yielded the best performance in recognising the type of clothes correctly.

Introduction

In this work, a performance comparison on image classification using two variants of ANN, namely ANN and CNN, is made. Fashion-MNIST consists of a dataset of 70,000 grayscale images with each image of size 28 × 28 pixels. Examples every ten categories of different types of clothing are T-shirt, trouser, pullover, dress, coat, sandal, shirt, sneaker, bag, and ankle boot.

The work has been motivated, therefore, by the need to assess the performance of various neural network models in carrying out tasks like that of image classification, with the view to establishing which best recognises patterns within images related to fashion. This will develop a better understanding of how to design more efficient computer vision systems for a wide range of applications, from retail automation and e-commerce to fashion recommendations.

Any company or research team that wants to automate this task of image-based classification of clothing items can be considered a client. Among several motivating factors, this could be because of the effective and accurate visual recognition systems competent in classifying thousands of images of clothes automatically with minimum human effort while minimising the classification errors.
Key questions addressed included the following:

- How do the performances of these architectures of ANN and CNN differ on the same dataset of images?
- The points of main challenge or confusion in the classification of clothing images are:

- Which suggestions can be made for improved model performance in future studies?

Correspondingly, the results of this study will support both practitioners and researchers in gaining an understanding of not only which model performs better but, more importantly, why misclassifications occur and how such models may be improved by further changes in the data and architecture.

Methods

The project was implemented using the PyTorch framework, one of the most used when working with deep learning. Both ANN and CNN models have been designed, trained, and tested on the Fashion-MNIST dataset while following the same preprocessing and evaluation procedure.

Further pre-processing involved normalising the images in the range from -1 to 1 for better stability during training and faster convergence. Afterwards, it was divided into three portions: a training set, a validation set, and a test set, following the 80:20 ratio of the two training and validation subsets while utilising a standard 10,000-image test set for final evaluation.

Two different model architectures have been implemented:

Artificial Neural Network (ANN):
The neural network architecture is design in such a way that contains three fully connected layer such as an: an input layer with 784 nodes and flattened on 28 × 28 images, followed by a hidden layer containing 256 neurons, another hidden layer with 128 neurons, and an output layer with 10 nodes for the ten categories of clothes.

Convolutional Neural Network(CNN):
We propose a CNN architecture that includes two convolutional layers followed by pooling in order to extract spatial hierarchies from image data. The first convolutional layer extracts 32 feature maps, while the second layer extracts 64 feature maps. Each of these convolutional layers is followed by a ReLU activation and further a max-pooling layer to reduce the dimensionality, and hence computation complexity, of data. Further, the code is followed by flattening, then a fully connected layer with 128 neurons, and an output layer with 10 neurons.

All the above models were trained using the Adam optimiser and cross-entropy loss for the 10 epochs, with a batch size of 64. During training, both training and validation losses were recorded to monitor performance and convergence across epochs.

First, the images were normalised and then divided into training, validation, and test sets. Both models were implemented in PyTorch with the Adam optimiser and cross-entropy loss. They were both trained for ten epochs.
Convolutional layers, which are very useful in the detection of edges, shapes, and textures, were used while the ANN had fully connected layers. To evaluate these models, accuracy and confusion matrices were used in determining classes that were frequently misclassified.
Then, both models were evaluated on the same test dataset for testing the accuracy of the result. Confusion matrices were also created to visualize misclassifications and identify which items of clothing are most frequently confused with others.

Results

These results indicate that CNN outperforms ANN both in test accuracy and overall performance.
Model Performance:

- ANN: 86% - relatively performed well but showed poor performance when having to deal with the features of spatial images. ANN achieved 86%, good results, but struggled with the details of the image.
- CNN: 92% - way better in recognizing the categories of clothes, and pattern recognition. Higher Accuracy, and Pattern Recognition.

The results from this model were not that good compared to the performance of the CNN model.

Considering it is simple and no feature extraction of the spatial information has been performed, an accuracy of about 86% is decent. But the performance on categories that are confused visually reflects that it treats every pixel independently without considering any spatial relations.

As the CNN achieved an accuracy of about 92%, this can be explained by its capability in the modelling and learning of spatial dependencies, which it does as an important ingredient in understanding images that contain edges, textures, and patterns. These convolutional layers of the CNN thus managed to recognize unique visual cues distinguishing one type of clothing from another.

So, studying the confusion matrices, some issues are shared by both models. Items like shirts versus coats, and pullovers versus dresses, were most often confused for one another because these happen to be similar in the sense of outward appearance. These categories often have comparable textures and outlines, and it's hard even for the advanced model to discriminate them without extra context.

We note that both models struggled with distinguishing between shirts versus coats, and pullovers versus dresses since they look somewhat similar.
The bar chart comparison clearly demonstrated the superiority of the CNN over the ANN in terms of model accuracy. Besides, the CNN performed higher but also more consistently for different categories of clothing.
The graph clearly depicted that, out of the two models, the CNN model had higher overall accuracy.

Recommendations

Although the CNN model performed a little better, there are some further steps that can be considered in order to enhance accuracy and robustness:
Data Augmentation: Introduce rotation, flip, zoom, or brightness changes to artificially increase the diversity of the dataset. This helps the model generalize better and reduces overfitting. Data augmentation like rotation or flipping normally serves to improve the model's robustness.

Deeper Network Architectures: You can try adding more convolutional layers, residual connections, or kernels of a bigger size to capture more complicated features. Further adaptation with ResNet or VGGNet may be performed for further gains on this dataset. Increases the numbers of CNN layers and utilises the batch normalisation to improve feature learning.

Batch Normalisation: Accordingly, applying batch normalisation between layers could stabilize learning, improve gradient flow, and lead to faster convergence during training.
Optimiser Experimentation: This could also be implemented using any of the other optimisers such as AdamW or RMSprop, including the learning rate schedulers to adjust the learning rate dynamically during training. Try running it using other optimisers such as AdamW; then try other methods of learning rate schedules.

Additional Evaluation Metrics: Include precision, recall, and F1-score that give more details about model performance between classes, especially in cases with an imbalance of classes. Include more measures such as precision and recall to get an in-depth analysis.

Visualise Features: Utilise some technique such as feature maps or Grad-CAM so as to observe which features of the image are driving model decisions. Such interpretability allows for model refinement and generates more transparent classifications.
This would be even better if the following were done through these improvements.
With these improvements, future experiments could achieve even higher accuracy and reliable performance, possibly higher than 94-95%.

Conclusion

The work clearly shows that the CNNs is better than ANN in the task of image classification. Both models were best to learn to classify images of clothes from the Fashion-MNIST dataset, but the CNN was reaching an accuracy of about 92%, against 86% reached by ANN. That is because a CNN is good at processing spatial and structural information in images and is hence better suited for recognizing edges, textures, and shapes of objects.

This work demonstrates that on image classification tasks, CNNs outperform ANNs significantly: CNN achieved about 92%, while ANN achieved about 86%. This is due to the nature of CNNs, which are capable of understanding spatial features within a picture like edges, textures, and patterns.
ANNs treat the images as one-dimensional vectors, which causes loss of critical spatial relationships.

In practical terms, the performance of the CNN model presents a wide range of applications in different walks of life. For instance, in retail automation, the work of the CNN model may be employed to automatically identify and categorize pieces of clothing to update inventory. In e-commerce, they could improve product tagging and search recommendations. Also, CNNs can be adapted for fashion trend analysis and quality control systems within manufacturing.

Though most image recognition systems are performing better with CNNs, their applications can be found in retail automation, e-commerce, and AI-based visual classification.

While these results are promising, challenges still remain in making fine-grained distinctions between similar items. Larger datasets, multiclass visual embeddings, and transfer learning using pre-trained large models such as ResNet or EfficientNet should be researched in future work in order to achieve near-human performance in the classification task.

This, in turn, provides a solid platform to understand the influence of neural network architecture on performance for the selected image classification task. Therefore, CNNs represent the current state-of-the-art model in different modern computer vision tasks because of their spatial pattern grasping abilities. The following recommended enhancements will further strengthen their accuracy, robustness, and applicability in real-world applications.