### **Analysis Report: CNN vs RNN for Fashion MNIST Classification**

#### **1. Introduction**

The objective of this analysis is to evaluate and compare the performance of two deep learning architectures—a Convolutional Neural Network (CNN) and a Recurrent Neural Network (RNN)—on the Fashion MNIST dataset. The models were trained, validated, and tested, and their performance was analyzed using accuracy metrics, probability distributions, and confusion matrices.

#### **2. CNN Model Analysis**

The CNN model achieved a training accuracy of **89.83%** with a validation accuracy of **89.42%**, demonstrating strong generalization. The loss values were **0.2814 for training** and **0.2955 for validation**, indicating effective learning with minimal overfitting.

##### **Training vs Validation Accuracy**

The accuracy plot showed rapid convergence after the first epoch, with both training and validation accuracy stabilizing. However, a slight divergence appeared towards the final epochs, particularly at epoch 7, suggesting mild overfitting.

##### **Prediction Probability Distribution**

Visualizing the probability distributions of predictions revealed that, in most cases, the predicted and actual labels were close in probability. However, an interesting case was observed where only the true label was highlighted, suggesting an instance where the model was highly uncertain about its prediction.

##### **Confusion Matrix Analysis**

The confusion matrix showed strong classification performance, with the highest values appearing along the diagonal, meaning most predictions matched the true labels. However, notable misclassifications were observed between **class 6 and class 0**, where **149 instances of class 6 were incorrectly predicted as class 0**, and **100 instances of class 0 were misclassified as class 6**. This suggests that these two classes share similar visual features, making it difficult for the model to distinguish them effectively.

#### **3. RNN Model Analysis**

The RNN model, utilizing an LSTM layer, achieved a training accuracy of **87.90%** and a validation accuracy of **86.71%**. The loss values were **0.3329 for training** and **0.3593 for validation**, indicating a slightly weaker performance compared to the CNN model.

##### **Training vs Validation Accuracy**

The accuracy plot showed initial convergence after the first epoch, but a consistent divergence emerged from epoch 3 onward, indicating that the model was overfitting significantly earlier than the CNN model. This suggests that RNNs may not be well-suited for this image classification task, as they struggle with feature extraction compared to CNNs.

**Prediction results:**  
accuracy: 0.8595 - loss: 0.3797 Test Accuracy (RNN): 0.8578000068664551

##### **Performance Comparison with CNN**

While the RNN model demonstrated reasonable accuracy, it underperformed compared to the CNN model in every metric. The divergence in training vs validation accuracy was more pronounced, and the overall accuracy was lower. The primary reason for this is that RNNs, particularly LSTMs, are designed for **sequential data** rather than spatial data like images. In this case, the CNN architecture is more effective in extracting meaningful features from image pixels.

#### **4. Conclusion and Recommendations**

Based on the results, the CNN model outperformed the RNN model in accuracy, generalization, and stability. The key observations include:

* **CNNs are more suitable for image classification**, achieving higher accuracy and lower loss.
* **Overfitting was present in both models**, but it was more significant in the RNN model.
* **Certain classes, particularly 6 and 0, were frequently misclassified**, indicating possible similarities in their visual structures.

To further improve performance, the following strategies can be considered:

* **Data Augmentation**: Introducing transformations such as rotation, flipping, and contrast adjustments to improve generalization.
* **Regularization Techniques**: Adding dropout layers or L2 regularization to mitigate overfitting.
* **More Advanced Architectures**: Exploring models like ResNet or VGG for enhanced feature extraction.
* **Hyperparameter Tuning**: Adjusting learning rates, batch sizes, and optimizers to refine performance.

In conclusion, while both models demonstrated reasonable classification performance, CNNs remain the preferred approach for image classification tasks due to their superior ability to extract spatial features effectively.