**Implementation of GAN in Fashion MNIST**

Jean P. Melendez Villanueva

University of Maryland

Data 655 Deep Learning and Neural Networks (9040)

Professor Jeremy Bolton

**Implementation of GAN in Fashion MNIST**

**Abstract**

This report provides an in-depth examination of the implementation of a Generative Adversarial Network (GAN) using the Fashion MNIST dataset. The analysis includes an exploration of the dataset characteristics, the rationale for using GANs, and a step-by-step breakdown of model architecture and training methodology. We further conduct a comprehensive evaluation of model performance through both visual and quantitative metrics, including Fréchet Inception Distance (FID) scores and loss function convergence. The findings demonstrate the model’s efficacy in generating high-fidelity synthetic images while highlighting challenges such as mode collapse and hyperparameter sensitivity. This study serves as a foundational guide for researchers and practitioners looking to optimize GAN performance for image generation tasks.

* 1. **Introduction**

**Objective**

The primary objective of this study is to design, implement, and evaluate a GAN that is capable of synthesizing realistic clothing images using the Fashion MNIST dataset. By systematically adjusting architectural configurations and hyperparameters, this study aims to determine the impact of different model settings on GAN performance. The report also aims to answer several key research questions:

* How well can a GAN learn the underlying distribution of the Fashion MNIST dataset?
* What is the effect of various architectural modifications on the generated image quality?
* How can we quantitatively assess the quality of generated images beyond visual inspection?
* What are the primary challenges in training GANs, and how can they be mitigated?

By addressing these questions, I aim to provide actionable insights into the training and optimization of GANs for fashion image synthesis.

**Problem Domain**

GANs have revolutionized deep learning applications in computer vision by enabling machines to generate high-quality synthetic images. The Fashion MNIST dataset, a widely used benchmark for image classification and generative modeling, contains grayscale images of ten different clothing categories, including shirts, dresses, sneakers, and handbags.

Fashion retailers and designers can leverage GANs for tasks such as:

* **Synthetic Data Generation**: Augmenting datasets for training robust machine learning models.
* **Product Design Prototyping**: Automatically generating new fashion designs based on trends.
* **Anomaly Detection**: Identifying counterfeit or defective clothing items in manufacturing.

Despite their potential, GANs are notoriously difficult to train due to issues such as mode collapse, unstable training dynamics, and sensitivity to hyperparameters. This study aims to explore these challenges by experimenting with various GAN configurations and analyzing their effectiveness in generating high-quality images.

**Method Rationale**

GANs, first introduced by Goodfellow et al. (2014), consist of two competing neural networks:

* **Generator (G):** Learns to create realistic synthetic images from random noise.
* **Discriminator (D):** Learns to distinguish real Fashion MNIST images from fake ones.

The two networks engage in an adversarial game, where the generator improves its ability to fool the discriminator, while the discriminator improves its classification accuracy. The iterative process enhances the generator’s capability to produce authentic-looking fashion items.

Unlike traditional image generation methods, GANs provide:

1. **Unsupervised Learning Capability**: No need for labeled data.
2. **High-Resolution Image Synthesis**: Ability to generate detailed textures and patterns.
3. **Adaptive Learning**: The model dynamically improves based on feedback from the discriminator.

This study employs a Deep Convolutional GAN (DCGAN) architecture due to its proven effectiveness in stabilizing GAN training for image generation. Furthermore, loss function selection and hyperparameter tuning are explored to optimize performance.

* 1. **Analysis**

**Data Overview**

The Fashion MNIST dataset comprises 70,000 grayscale images, each measuring 28x28 pixels and categorized into 10 distinct classes: T-shirt/top, trouser, pullover, dress, coat, sandal, shirt, sneaker, bag, and ankle boot. Each image is labeled with a class, making the dataset a structured benchmark for image-based deep learning tasks. Designed as a drop-in replacement for the traditional MNIST dataset of handwritten digits, Fashion MNIST presents a more challenging classification problem due to its complex visual patterns and textures. This makes it an ideal dataset for evaluating deep learning models in fashion-related image recognition tasks.

**Exploratory Analysis**

**1. Sample Image Inspection**

To understand the dataset structure, random samples were visualized at different epochs during GAN training. Below are images generated at various training stages:

|  |  |
| --- | --- |
| Epoch | Figure Reference |
| 10 | Figure 1: Generated Images at Epoch |
| 20 | Figure 2: Generated Images at Epoch |
| 30 | Figure 3: Generated Images at Epoch |
| 40 | Figure 4: Generated Images at Epoch |
| 50 | Figure 5: Generated Images at Epoch |
| 60 | Figure 6: Generated Images at Epoch |
| 70 | Figure 7: Generated Images at Epoch |
| 80 | Figure 8: Generated Images at Epoch |
| 90 | Figure 9: Generated Images at Epoch |
| 100 | Figure 10: Generated Images at Epoch |

During the early training phase (epochs 10-30), the generated images were highly noisy with unclear structures, indicating the generator's struggle to produce realistic clothing items. As training progressed (epochs 40-60), distinct clothing shapes started to form, though artifacts and blurring remained. By epochs 70-100, the images closely resembled Fashion MNIST clothing categories, demonstrating the generator's ability to create realistic fashion items. Further evaluation using Figure 11 showed diverse generated samples, confirming that the GAN successfully learned multiple clothing categories and produced varied images rather than repeating the same patterns.

**Preprocessing**

The dataset was preprocessed to ensure compatibility with the deep learning framework and to optimize data representation for GAN training. First, pixel values were normalized from their original range of [0, 255] to [0,1], which stabilized training by preventing large gradients and improving learning efficiency. Next, the images were reshaped into a three-dimensional tensor of (28, 28, 1) to maintain compatibility with deep learning frameworks, ensuring proper input handling for convolutional layers. Additionally, the dataset was divided into mini-batches of size 128 to enhance computational efficiency, reduce memory consumption, and allow for smoother convergence by updating model weights frequently.

To further optimize training, optional data augmentation techniques such as cropping, rotation, and flipping were considered to enhance diversity in generated images and reduce mode collapse, though they were not explicitly used in this study. A key GAN-specific preprocessing step was the injection of noise, where the generator model received a latent noise vector sampled from a standard normal distribution. Experiments were conducted with different latent vector sizes ranging from 64 to 256 to assess their impact on the quality and diversity of generated images. These preprocessing steps collectively ensured stable training, accelerated convergence, and improved the model’s ability to generate sharp and realistic images.

**Algorithm Intuition**

The GAN model consists of two primary components: the generator and the discriminator. The generator takes random noise as input and generates synthetic images, attempting to mimic the real Fashion MNIST images. Meanwhile, the discriminator is tasked with distinguishing between real and generated images, providing feedback to the generator to improve its output. The two networks are trained in an adversarial manner, where the generator aims to minimize the discriminator’s ability to classify images correctly, and the discriminator seeks to enhance its classification accuracy. This continuous competition drives the GAN towards generating more realistic images over successive epochs. Different loss functions and optimization techniques were tested to evaluate their impact on convergence and output quality.

**Model Fitting**

**Training Loss and Performance Analysis**

GAN training requires balancing generator and discriminator loss to prevent issues like mode collapse or instability. As shown in Figure 12, the initial training phase (epochs 0-30) sees the discriminator outperforming the generator, which struggles to produce realistic images due to its random initialization. By the mid-phase (epochs 40-70), losses stabilize, indicating balanced learning. In the late phase (epochs 80-100), the generator improves further, generating higher-quality Fashion MNIST samples.

**Fréchet Inception Distance (FID)**

FID quantifies the similarity between generated and real images, with lower scores indicating better performance. As seen in Figure 13, the FID score steadily decreases throughout training, confirming improved image quality. Minor fluctuations suggest challenges in certain clothing categories, which could be improved through hyperparameter tuning or architectural refinement.

**Generator Weight Distribution**

Examining generator weight distribution provides insights into training stability. Figure 14 shows well-distributed weights, indicating a stable learning process. However, extreme weight values could signal instability, requiring careful tuning of learning rates and optimizer settings. Overall, loss analysis, FID scores, and weight distributions confirm stable training, though further refinements could enhance performance.

1. **Results**

**Output**

The objective of this study was to train a GAN on the Fashion MNIST dataset and evaluate its ability to generate realistic clothing images while optimizing architectural and hyperparameter settings. The generated outputs were assessed both qualitatively (visual inspection of generated images over training epochs) and quantitatively (using FID and training loss curves).

The results demonstrate that GAN training was successful in generating visually recognizable clothing items. Over the course of training, the generator improved its ability to create diverse and structured fashion images, showing significant enhancement from early to late training epochs. As seen in Figures 1-10, at early training stages (epochs 10-30), the images lacked clear structures and were dominated by noise. However, as training progressed, mid-stage images (epochs 40-70) began showing recognizable clothing shapes. By the later epochs (80-100), the generator was able to produce sharp, well-defined images resembling the original Fashion MNIST dataset.

Furthermore, the FID score analysis (Figure 13) indicates a gradual improvement in the quality of generated images, confirming that the GAN learned a realistic representation of the dataset. The training loss curves (Figure 12) show that the generator and discriminator reached a balanced training state after approximately 60 epochs, further supporting the model’s stability and effectiveness.

Thus, based on both qualitative and quantitative assessments, the GAN successfully met the study’s objective by producing realistic, structured, and diverse synthetic fashion images while maintaining stable training dynamics.

**Model Properties**

The final trained model consists of a Deep Convolutional GAN (DCGAN) architecture optimized for 28×28 grayscale image generation. The generator network transforms a random noise vector (100D) into structured clothing images, while the discriminator network distinguishes real from generated images.

**Key Model Characteristics:**

1. **Generator:**
   * Input: **Random noise vector (100D)**
   * Layers:
     + **Dense (7x7x256 neurons)**
     + **Transposed Convolution (128 filters, ReLU, stride=2)**
     + **Transposed Convolution (64 filters, ReLU, stride=2)**
     + **Final Conv2D (1 filter, tanh activation)**
   * Output: **28×28 grayscale image**
2. **Discriminator:**
   * Input: **28×28 grayscale image**
   * Layers:
     + **Conv2D (64 filters, LeakyReLU, stride=2)**
     + **Conv2D (128 filters, LeakyReLU, stride=2)**
     + **Fully Connected Layer → Sigmoid Output**
   * Output: **Probability score (Real or Fake)**
3. **Training Parameters:**
   * **Optimizer:** Adam (β1=0.5\beta\_1 = 0.5β1​=0.5)
   * **Batch Size:** 128
   * **Loss Function:** Binary Cross-Entropy (BCE)
   * **Training Duration:** 100 epochs

A summary of model parameters and weights distribution was analyzed using Figure 14 (Histogram of Generator Weights, Keras), confirming that the weight values were well-distributed, ensuring stable learning.

Overall, the DCGAN architecture used in this study successfully captured complex patterns in the Fashion MNIST dataset, resulting in a well-trained generator capable of producing realistic images.

**Evaluation**

The GAN model was evaluated using both qualitative visual analysis and quantitative metrics to assess its ability to generate realistic fashion images. Early training phases (epochs 10-30) produced highly noisy and unstructured images, reflecting the generator's initial learning stage. By the mid-training phase (epochs 40-70), clearer clothing outlines emerged, indicating that the model had started capturing meaningful features. In the later training epochs (80-100), the images became sharper and more defined, closely resembling real Fashion MNIST samples.

Training loss analysis further validated the model's progression, with loss curves stabilizing around epoch 60, indicating a balanced state between the generator and discriminator. The steadily decreasing generator loss confirmed its improved ability to produce realistic images, while the stable discriminator loss prevented mode collapse. The continuously improving FID score demonstrated the increasing realism and diversity of generated images. Additionally, the well-distributed generator weights indicated stable learning, reinforcing the effectiveness of hyperparameter tuning in maintaining robust training.

1. **Conclusion**

**Summary of Findings**

The primary objective of this study was to train a GAN on the Fashion MNIST dataset to generate realistic and diverse synthetic clothing images. The model successfully learned to produce well-structured and recognizable clothing items, improving significantly over training epochs. Both visual inspection and quantitative evaluation (FID scores, loss analysis) confirmed that the GAN effectively captured the dataset’s patterns and generated high-quality outputs. These findings demonstrate the potential of GANs in synthetic data generation for fashion applications, including dataset augmentation, creative design, and automated fashion prototyping.

**Limitations**

Despite its success, the model has several limitations that impact performance. The low-resolution (28×28) nature of the Fashion MNIST dataset limited the level of detail the generator could produce, making it difficult to replicate intricate textures and complex clothing designs. Additionally, GAN training stability remains a challenge, as the model is prone to mode collapse, where it generates repetitive outputs instead of diverse samples. The computational cost of training was also significant, requiring careful hyperparameter tuning and extensive processing power to achieve stable convergence.

**Improvement Areas**

Several improvements can be made to enhance the performance of the GAN. Increasing the image resolution by using datasets with higher-dimensional images would allow for more detailed and realistic outputs. Implementing advanced GAN architectures, such as StyleGAN or progressive growing GANs, could improve image diversity and training stability. Additionally, using adaptive learning rate schedules, regularization techniques, and transfer learning from pre-trained models could further optimize the model’s training efficiency. Future research could explore conditional GANs (cGANs) to enable controlled fashion image generation based on specific clothing categories or design attributes.

**References**

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative adversarial networks. Advances in Neural Information Processing Systems (NeurIPS). https://arxiv.org/abs/1406.2661

Radford, A., Metz, L., & Chintala, S. (2016). Unsupervised representation learning with deep convolutional generative adversarial networks. International Conference on Learning Representations (ICLR). https://arxiv.org/abs/1511.06434

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., & Hochreiter, S. (2017). GANs trained by a two time-scale update rule converge to a local Nash equilibrium. Advances in Neural Information Processing Systems (NeurIPS). https://arxiv.org/abs/1706.08500

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). https://arxiv.org/abs/1512.00567

Odena, A., Olah, C., & Shlens, J. (2017). Conditional image synthesis with auxiliary classifier GANs. International Conference on Machine Learning (ICML). https://arxiv.org/abs/1610.09585

TensorFlow. (2024). Deep Convolutional Generative Adversarial Networks (DCGAN) tutorial. TensorFlow Tutorials. https://www.tensorflow.org/tutorials/generative/dcgan

d2l.ai. (2024). Generative adversarial networks (GANs). Dive into Deep Learning. https://d2l.ai/chapter\_generative-adversarial-networks/gan.html

O'Reilly Media. (2024). Generative adversarial networks for beginners. https://www.oreilly.com/content/generative-adversarial-networks-for-beginners/

Fashion MNIST Dataset. (2024). TensorFlow Datasets Catalog. https://www.tensorflow.org/datasets/catalog/fashion\_mnist

Chollet, F. (2017). Deep Learning with Python. Manning Publications.

**Appendix**

**Appendix A –** Refer to the accompanying Jupyter Notebook file for code implementation

**Appendix B** – Visualizations

A collage of different clothing

AI-generated content may be incorrect.

Figure 1 – Epoch\_10

A collage of different clothing

AI-generated content may be incorrect.

Figure 2 – Epoch\_20

A collage of different clothing

AI-generated content may be incorrect.

Figure 3 – Epoch\_30

A collage of different images of clothes

AI-generated content may be incorrect.

Figure 4 – Epoch\_40

A collage of different clothing

AI-generated content may be incorrect.

Figure 5 – Epoch\_50

A collage of different clothing

AI-generated content may be incorrect.

Figure 6 – Epoch\_60

A collage of different clothing

AI-generated content may be incorrect.

Figure 7 – Epoch\_70

A collage of different clothing

AI-generated content may be incorrect.

Figure 8 – Epoch\_80

A collage of different types of clothing

AI-generated content may be incorrect.

Figure 9 - Epoch\_90

A collage of different clothing

AI-generated content may be incorrect.

Figure 10 - Epoch\_100

A graph of different colored rectangular shapes

AI-generated content may be incorrect.

Figure 11 - Diversity of Generated Fashion MNIST Images

A graph with numbers and lines

AI-generated content may be incorrect.

Figure 12 – GAN Training Loss Over Epochs

A graph of a number of lines

AI-generated content may be incorrect.

Figure 13 – FID Score Distribution

A graph of a weight

AI-generated content may be incorrect.

Figure 14 – Histogram of Generator Weights (Keras)