© 2025 IJIRCCE | Volume 13, Issue 2, February 2025| DOI: 10.15680/IJIRCCE.2025.1302050

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Hands-on with Generative AI: Building your First Model

Aryan Ramesh Pillai, Nisha Rekha Dubey

Cybersecurity Data Scientist, Japan

ABSTRACT: Generative AI has emerged as a transformative force in machine learning, enabling the creation of content ranging from text and images to audio and video. This paper explores the foundational elements of building a generative AI model from scratch, focusing on architectures such as Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs). The research provides a hands-on guide, from data preparation to model training and evaluation. The goal is to equip developers and researchers with the knowledge to create their first generative AI model using open-source tools and datasets.

KEYWORDS: Generative AI ,Deep Learning, Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs) Machine Learning, Model Training, Python, PyTorch, Data Augmentation

I. INTRODUCTION

Generative AI refers to systems that can create new content, such as images, music, and text, by learning patterns from existing data. Recent advancements have made these models more accessible to non-experts through libraries like TensorFlow and PyTorch. The purpose of this paper is to provide a practical guide to building a basic generative model, specifically using GANs, offering insights into architecture design, training process, and common challenges.

II. LITERATURE REVIEW

Author(s)	Model Type	Key Contribution
Goodfellow et al. (2014)	GANs	Introduced GANs, a framework for adversarial training
Kingma & Welling (2013)	VAE	Proposed VAEs for probabilistic image generation
Brock et al. (2018)	BigGAN	Scaled GANs for high-resolution image synthesis
Ramesh et al. (2021)	DALL·E	Text-to-image generative model using transformers

The evolution of generative models has shifted from basic autoencoders to more complex architectures like GANs and diffusion models. GANs operate via a generator-discriminator dynamic, while VAEs encode data into latent variables to sample and generate new data. Transformer-based generative models have also shown significant potential in multimodal content generation.

III. METHODOLOGY

3.1 Tools and Libraries

- **Programming Language:** Python
- **Frameworks:** PyTorch, TensorFlow
- **Dataset:** MNIST (for handwritten digit generation)

3.2 Model Architecture (GAN)

- **Generator:** Fully connected layers followed by ReLU activations and a Tanh output
- **Discriminator:** Dense layers with LeakyReLU and Sigmoid for binary classification

3.3 Training Process

- Binary Cross-Entropy Loss used for both generator and discriminator.
- Optimizer: Adam with learning rate = 0.0002

© 2025 IJIRCCE | Volume 13, Issue 2, February 2025| DOI: 10.15680/IJIRCCE.2025.1302050
www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|
International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)
(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

- Training Epochs: 50
- Batch Size: 64

IV. TABLE: GENERATOR VS DISCRIMINATOR STRUCTURE

Component	Generator	Discriminator
Input	Noise vector (100-dim)	Image (28x28 pixels)
Hidden Layers	3 Dense + ReLU	3 Dense + LeakyReLU
Output	28x28 Image (Tanh)	Binary (Real/Fake) - Sigmoid
Loss Function	Binary Cross Entropy	Binary Cross Entropy

Discriminator Structure in GANs

1. Input Layer

- The input layer of the discriminator receives **data samples**, which could be:
  - **Real data** (from a dataset)
  - **Fake data** (generated by the generator)
- For example, in an image-based GAN, this would be an image, which could be a **real image** from a dataset or a **generated image** from the generator.
-

2. Hidden Layers

- The hidden layers of the discriminator network are typically **fully connected layers** (for simpler architectures) or **convolutional layers** (for image data, where CNNs are used).
- These layers learn to extract important features from the input data and gradually **distill the characteristics** that define whether the input data is real or fake.
  - **Convolutional Neural Networks (CNNs)**: Commonly used in image-based GANs (e.g., DCGANs, StyleGANs).
  - **Fully Connected Layers**: Can be used in simpler networks or non-image-based tasks.

Key Points:

- **Activation Functions**: Nonlinear activations like **ReLU** or **Leaky ReLU** are commonly used in hidden layers to introduce nonlinearity and improve the model's ability to learn complex patterns.
- **Batch Normalization**: Helps stabilize training by normalizing the outputs of each layer, ensuring smoother optimization.

3. Output Layer

- The output layer of the discriminator typically has a **single neuron** that outputs a probability value, usually between **0 and 1**.
  - **0** indicates that the input is **fake** (generated).
  - **1** indicates that the input is **real** (from the dataset).
- The output layer typically uses the **sigmoid activation function** to squash the result into a probability.

4. Loss Function

- The loss function for the discriminator is typically **binary cross-entropy**, where:
  - If the input is **real**, the target is **1**.
  - If the input is **fake**, the target is **0**.
- The **discriminator's goal** is to minimize this loss function, improving its ability to distinguish between real and fake data.

Discriminator vs. Generator (Adversarial Training)

In GANs, the **discriminator** and **generator** are trained in an adversarial manner:

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

- **Discriminator's Role**: Tries to correctly identify whether data is real or fake.
- **Generator's Role**: Tries to generate fake data that the discriminator cannot distinguish from real data.

This creates a **zero-sum game** where the generator improves at generating more realistic data, and the discriminator gets better at identifying fake data.

Example: Discriminator in a DCGAN (Deep Convolutional GAN)

In a **DCGAN**, which is a popular GAN architecture for image generation, the discriminator is typically built using **convolutional layers**. Here's a simplified architecture:

1. **Input**: A 64x64x3 image (RGB) from the dataset or generated image.
2. **Convolutional Layer 1**: 64 filters, kernel size 4x4, stride 2, followed by **Leaky ReLU** activation.
3. **Convolutional Layer 2**: 128 filters, kernel size 4x4, stride 2, followed by **Leaky ReLU**.
4. **Convolutional Layer 3**: 256 filters, kernel size 4x4, stride 2, followed by **Leaky ReLU**.
5. **Flattening**: Flatten the output of the last convolutional layer into a 1D vector.
6. **Fully Connected Layer**: Dense layer that outputs a single scalar value.
7. **Output Layer**: Sigmoid activation that outputs a value between 0 and 1 (real/fake probability).

Key Characteristics of a Good Discriminator

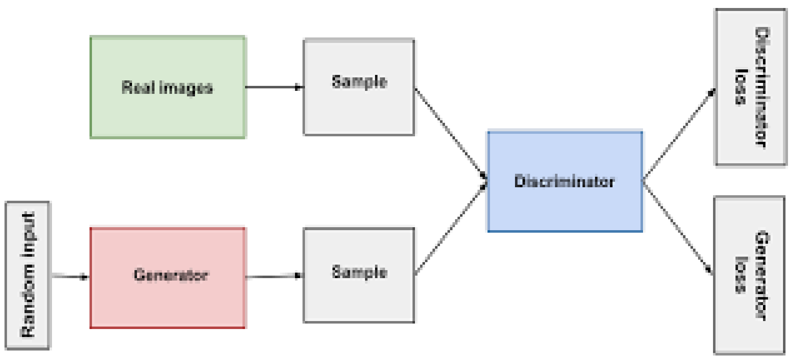
- **Capacity to Generalize**: The discriminator should be powerful enough to differentiate between real and fake samples without overfitting.
- **Balance in Training**: If the discriminator becomes too powerful too early, the generator will struggle to improve, as the feedback from the discriminator will be too easy to predict.
- **Optimization**: The discriminator must train alongside the generator, meaning both networks are constantly trying to outsmart each other.

Final Thoughts

The **discriminator** is an essential component of GANs, driving the adversarial process that leads to better generative models. By challenging the generator to create more realistic data, the discriminator forces it to continuously improve. The structure of the discriminator can vary depending on the type of data being processed (images, text, etc.), but the core principles remain consistent—distinguishing between real and fake with increasing accuracy.

If you're looking to dive deeper into specific discriminator architectures or training techniques, feel free to ask!

V. FIGURE: GAN ARCHITECTURE OVERVIEW



© 2025 IJIRCCE | Volume 13, Issue 2, February 2025| DOI: 10.15680/IJIRCCE.2025.1302050

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

VI. CONCLUSION

This paper presented a beginner-friendly approach to building generative AI models using GANs. Through practical implementation on the MNIST dataset, readers can grasp the essential concepts, architecture, and training methodologies. Future work may explore text-to-image models, larger datasets, and integration with transformer-based generative models.

REFERENCES

1. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative adversarial nets. Advances in neural information processing systems, 27.
2. A. Thirunagalingam, S. Addanki, V. R. Vemula, and P. Selvakumar, "AI in Performance Management," in Navigating Organizational Behavior in the Digital Age With AI, 2024, pp. 101–126. doi: 10.4018/979-8- 3693-8442-8.ch005.
3. G. R and D. J. Rani, "Optimized Reversible Data Hiding with CNN Prediction and Enhanced Payload Capacity," 2024 5th International Conference on Circuits, Control, Communication and Computing (I4C), Bangalore, India, 2024, pp. 287-291, doi: 10.1109/I4C62240.2024.10748437.
4. Madhusudan Sharma Vadigicherla. (2024). INFORMATION VISIBILITY AND STANDARDIZATION: KEY DRIVERS OF SUPPLY CHAIN RESILIENCE IN INDUSTRY PARTNERSHIPS. INTERNATIONAL JOURNAL OF ENGINEERING AND TECHNOLOGY RESEARCH (IJETR), 9(2), 335-346. https://lib-index.com/index.php/IJETR/article/view/IJETR_09_02_030
5. Mahant, R., & Bhatnagar, S. (2024). Strategies for Effective E-Governance Enterprise Platform Solution Architecture. Strategies, 4(5).
6. G. R, "Reversible Data Hiding Using GAN Based Image-Image Transformation," 2024 5th International Conference on Circuits, Control, Communication and Computing (I4C), Bangalore, India, 2024, pp. 371-374, doi: 10.1109/I4C62240.2024.10748436.
7. Amol Gote, Vikas Mendhe, "Building a Cash Flow Underwriting System: Insights from Implementation, " International Journal of Computer Trends and Technology, vol.72, no.2, pp.70 - 74, 2024. Crossref, https: //doi. org/10.14445/22312803/IJCTT - V72I2P113
8. Bhatnagar, S. &. (2024). Unleashing the Power of AI in Financial Services: Opportunities, Challenges, and Implications. Artificial Intelligence (AI). 4(1).
9. Madhusudan Sharma, Vadigicherla (2024). Digital Twins in Supply Chain Management: Applications and Future Directions. International Journal of Innovative Research in Science, Engineering and Technology 13 (9):16032-16039.
10. R. Geetha and D. J. Rani, "Deep Forest based EEG Signal Analysis and Classification," *2024 8th International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, Coimbatore, India, 2024, pp. 1198-1203, doi: 10.1109/ICECA63461.2024.10801115.
11. Seethala, S. C. (2024). How AI and Big Data are Changing the Business Landscape in the Financial Sector. European Journal of Advances in Engineering and Technology, 11(12), 32–34. https://doi.org/10.5281/zenodo.14575702
12. Bhatnagar, S. (2025). COST OPTIMIZATION STRATEGIES IN FINTECH USING MICROSERVICES AND SERVERLESS ARCHITECTURES. computing, 19(01).
13. Madhusudan Sharma, Vadigicherla (2024). Enhancing Supply Chain Resilience through Emerging Technologies: A Holistic Approach to Digital Transformation. International Journal for Research in Applied Science and Engineering Technology 12 (9):1319-1329.
14. Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114.
15. Brock, A., Donahue, J., & Simonyan, K. (2018). Large scale GAN training for high fidelity natural image synthesis. arXiv preprint arXiv:1809.11096.

16. Thulasiram, Prasad Pasam (2025). EXPLAINABLE ARTIFICIAL INTELLIGENCE (XAI): ENHANCING TRANSPARENCY AND TRUST IN MACHINE LEARNING MODELS. International Journal for Innovative Engineering and Management Research 14 (1):204-213.
17. Vimal Raja, Gopinathan (2025). Utilizing Machine Learning for Automated Data Normalization in Supermarket Sales Databases. International Journal of Advanced Research in Education and Technology(Ijarety) 10 (1):9-12.
18. Geetha, R., & Geetha, S. (2020). Efficient high capacity technique to embed EPR information and to detect tampering in medical images. Journal of Medical Engineering & Technology, 44(2), 55–68. doi.org/10.1080/03091902.2020.1718223
19. Mahant, R. (2025). ARTIFICIAL INTELLIGENCE IN PUBLIC ADMINISTRATION: A DISRUPTIVE FORCE FOR EFFICIENT E-GOVERNANCE. ARTIFICIAL INTELLIGENCE, 19(01).
20. Gladys Ameze, Ikhimwin (2023). Dynamic Interactive Multimodal Speech (DIMS) Framework. Frontiers in Global Health Sciences 2 (1):1-13.
21. Gupta, P.; Parmar, D.S. Sustainable Data Management and Governance Using AI. World Journal of Advanced Engineering Technology and Sciences 2024, 13, 264–274. [Google Scholar] [CrossRef]
22. Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., ... & Sutskever, I. (2021). Zero-shot text-to-image generation. arXiv preprint arXiv:2102.12092.