

## ЗАВИСИМОСТЬ РАСПРЕДЕЛЕНИЯ ВХОДНЫХ И ВЫХОДНЫХ ДАННЫХ В МНОГОУРОВНЕВЫХ ГЕНЕРАТИВНО-СОСТЯЗАТЕЛЬНЫХ СЕТЯХ

Ковбаса Г.А.<sup>1</sup>, аспирант-соискатель кафедры ЭВС

Белорусский государственный университет информатики и радиоэлектроники<sup>1</sup>  
г. Минск, Республика Беларусь

Азаров И.С. – докт. техн. наук

**Аннотация.** Многоуровневые генеративные состязательные сети (multi-level GANs) позволяют поэтапно улучшать сгенерированные данные. Однако выход каждого уровня ГАН формирует промежуточное распределение, зависящее от исходной обучающей выборки. Ошибки или искажения в этих промежуточных распределениях могут усиливаться на последующих этапах, что приводит к деградации качества и разнообразия результатов. В данной статье рассматривается влияние формы и качества исходной выборки на распределения выходов нейросетей в многоуровневых ГАН, а также предлагаются методы компенсации каскадного искажения данных. Показано, что использование U-Net способствует сохранению локальных зависимостей и уменьшению искажения распределений, повышению реалистичности изображения, но в свою очередь сужает область распределения данных и разнообразие выходной выборки.

**Ключевые слова.** Нейронные сети, многоуровневые генеративные состязательные сети, нейросетевой коллапс, распределение данных.

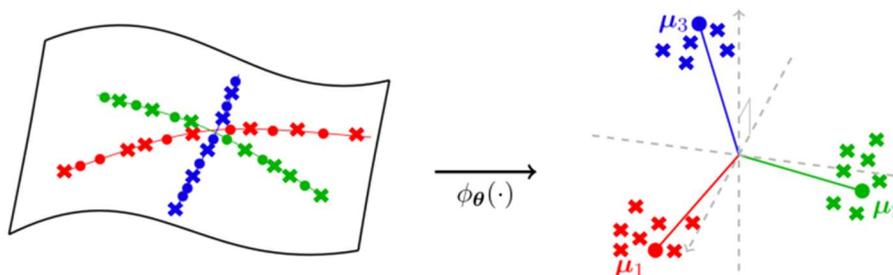
Генеративные состязательные сети (GAN) продемонстрировали высокую эффективность в синтезе изображений, аудио и текста. В многоуровневых архитектурах (каскадные или иерархические GAN) генерация происходит поэтапно: первый уровень создаёт заготовку, последующие уточняют и улучшают результат. Однако каждый уровень формирует собственное приближённое распределение данных, и ошибки могут накапливаться [1, 2].

Развитие генеративных моделей, в особенности GAN, сопровождается рядом системных ограничений, среди которых особое место занимает феномен нейросетевого коллапса — деградации способности модели к генерации новых, разнообразных и правдоподобных данных. Это состояние может быть следствием чрезмерной оптимизации, неадекватного распределения обучающих данных или неудачного взаимодействия между генератором и дискриминатором. Синтетические выборки, порождаемые самими генеративными моделями для дальнейшего обучения, также могут вызывать накопление искажений. Повторное обучение на таких данных ведет к сужению распределения, усилению доминирующих паттернов и исчезновению редких признаков.

Синтетические данные редко содержат уникальные события, аномалии, редкие сочетания признаков. Поэтому сеть теряет способность обнаруживать исключения, что особенно критично в медицине, безопасности, передовых научных исследованиях.

Искажения на каждом уровне могут усиливаться, особенно в условиях узких, несбалансированных или синтетических обучающих выборок. На практике коллапс можно визуализировать через метод главных компонент (рисунок 1) или t-SNE на последних скрытых слоях модели.

Рисунок 1 – Иллюстрация коллапса нейросети [3]



В состоянии коллапса:

1. все векторы одного класса скучены вместе, почти совпадают;
2. расстояния между классами равномерны и симметричны;
3. объекты вне этих кластеров ошибочно интерпретируются как ближайший класс.

Теоретическая основа зависимости распределений данных может выглядеть следующим образом:

$$X \sim P_{data} \text{ — истинное распределение данных,}$$

$$G_1(z) \sim P_1 \text{ — распределение первого генератора,}$$

$$G_2(G_1(z)) \sim P_2 \text{ — второго и так далее генератора.} \quad (1)$$

где  $P_1$  слабо аппроксимирует  $P_{data}$ , тогда  $P_2$  будет содержать ещё более выраженные искажения. Таким образом, если искажения переходят от слоя к слою, то наблюдается эффект *distributional shift compounding* [4].

В StackGAN [1] отмечается влияние качества первого уровня на детальность второго. В StyleGAN [5], несмотря на неглубокую иерархию, уровни латентного пространства контролируют ключевые аспекты распределения стиля. Проблемы усиливаются при обучении на данных, содержащих синтетические артефакты [6].

Методы компенсации и стабилизации распределений:

1. Смешивание с реальными данными. Реальные образцы вводятся на каждом уровне для коррекции распределения.
2. Auxiliary losses. Дополнительные функции потерь контролируют расстояние между промежуточными и реальными распределениями (например, KL-дивергенция).
3. Reprojection. Использование автоэнкодеров или регрессоров, возвращающих сгенерированные данные в пространство признаков исходного распределения.
4. Совместное обучение уровней. End-to-end fine-tuning всей цепочки позволяет компенсировать искажения ранних этапов [7].

В многоуровневых генеративных сетях зависимость выходных параметров от формы и полноты исходной выборки носит критический характер. Без соответствующих механизмов компенсации искажений возможно ухудшение качества генерации на каждом последующем уровне. Контроль распределения на промежуточных этапах необходим для предотвращения деградации модели и устойчивого обучения [8, 9].

В многоуровневых GAN, таких как MSGAN (Multi-Scale GAN) [10] и HIGAN (Hierarchical GAN) [11], проблема усугубляется тем, что каждая генерация зависит от предыдущих уровней. Ошибки и искажения, возникшие на низких уровнях, передаются по всей иерархии, снижая общую выразительность и достоверность изображений.

Современные методы оптимизации, такие как Adam, RMSProp и др., позволяют добиться быстрой сходимости. Однако это ускорение часто приводит к так называемому *overshooting* — быстрому закреплению на одном или нескольких модах распределения. В контексте GAN это приводит к неразнообразному выводу, а в случае синтетической переобученной выборки — забыванию моделью ключевых особенностей распределения [9].

Более устойчивыми оказываются модели, использующие регуляризацию, частичную оптимизацию или стохастическое семплирование на каждом уровне, как в HIGAN. Например, MSGAN внедряет шум и дискриминацию на каждом разрешении, позволяя нивелировать ошибки начальных слоев.

Одним из решений является внедрение U-Net генераторов в многоуровневые GAN. Такие архитектуры обеспечивают сквозные связи между уровнями разрешения, что позволяет сохранять важные признаки и избегать искажения на более высоких уровнях генерации. U-Net-подобные архитектуры решают эту проблему благодаря симметричным skip-соединениям, которые позволяют напрямую передавать многоуровневые признаки (features) с низких уровней сети (например, контуры, структура) на более высокие уровни (детали, текстура). Это даёт следующие преимущества:

1. Сохранение семантической согласованности: даже при деградации промежуточного вывода структура объекта (лицо, рука, объект сцены) восстанавливается на следующем уровне с учётом исходных признаков.
2. Снижение потерь в передаче информации между уровнями: т.к. данные из входа доступны напрямую, сеть не обязана повторно "восстанавливать" всё с нуля.
3. Стабильность распределения на выходе: прямой доступ к информации из исходного пространства уменьшает склонность к схлопыванию разнообразия (mode collapse) [7].

Использование U-Net в иерархических GAN позволило достичь лучшей консистентности изображений при генерации спутниковых изображений и медицинских снимков [12].

Применение U-Net-подобных генераторов в многоуровневых GAN показало свою эффективность, например, в архитектурах, объединяющих semantic-to-image и super-resolution фазы, а также в некоторых вариантах conditional GAN [11].

В отличие от каскадных GAN на основе традиционных генераторов, U-Net-подобные архитектуры позволяют сохранить и передать информацию из исходных данных напрямую на всех уровнях генерации. Благодаря наличию skip-соединений между энкодером и декодером, генератор может восстанавливать как глобальную структуру, так и локальные текстурные детали без необходимости полагаться исключительно на промежуточные представления предыдущих уровней. Это снижает вероятность накопления смещений в распределении и позволяет каждому уровню GAN действовать не только как трансформатор предыдущего вывода, но и как уточняющая модель с прямым доступом к изначальным признакам входа.

Присутствуют также и недостатки у описанного подхода. U-Net в генераторе приводит к условной генерации, приближая  $p_G(x)$  к  $p(x|xin)$ , а не к  $p_{data}(x)$ , что улучшает детальность, но снижает разнообразие, что продемонстрировано на рисунке 2 как более компактное и структурированное распределение GAN (with U-Net).

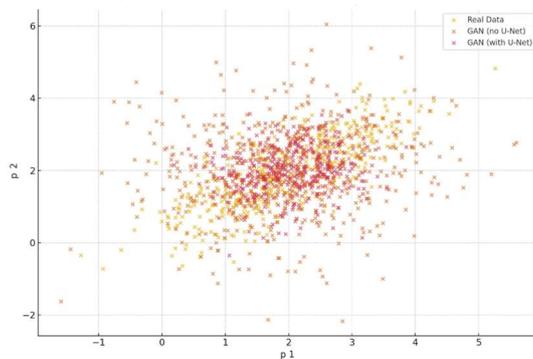


Рисунок 2 – Визуализация распределения реальных и синтетических данных, где GAN (no U-Net) — более рассеянное и шумное распределение, демонстрирующее низкую локальную согласованность

Без U-Net генератор лучше приближает глобальное распределение, но может страдать от артефактов и коллапса мод. Для оценки близости  $p_G(x)$  к  $p_{data}(x)$ , можно использовать метрики:

1. Fréchet Inception Distance (FID) — чувствителен к различиям в высокоуровневых признаках.
2. Kernel Inception Distance (KID) — устойчив к размерам выборки.
3. JS-дивергенция, KL-дивергенция — в случае известной плотности.
4. t-SNE / UMAP визуализации распределений признаков.

Анализ закономерностей распределения данных в Conditional Adversarial Networks [13] показал, что при использовании U-Net в генераторе, распределение выходов ближе к распределению реальных данных по FID, но область распределения сужается: модель начинает «залипать» на шаблонных генерациях и хуже покрывает малораспространённые моды. Исследования указывают, что распределение выходных данных GAN часто имеет меньшую энтропию, чем распределение обучающей выборки [8]. Это свидетельствует о потере разнообразия и предсказуемости результатов. Одной из причин является несоответствие между распределением латентного пространства (обычно нормального или равномерного) и сложной структуры пространства данных [8].

Оптимальное решение — комбинирование U-Net архитектуры с многоуровневым стохастическим шумом и прогрессивным обучением, как в HiGAN, чтобы сохранить как структуру, так и разнообразие. Также эффективно будет использование методов imitation learning или reinforcement learning с применением методов оценки распределения данных, таких как log-likelihood, как метрики оценки реалистичности полученного изображения в процессе прогрессивного обучения [14]. За основу оценки log-likelihood берется приближенный метод AIS, annealed importance sampling, который может с достаточной точностью аппроксимировать оценку для GAN сетей [14]. Используя оценку AIS можно оценивать не только соответствие распределения выходных данных исходной выборке, но и реалистичность изображения в целом, что выгодно отличает ее от метрики FID.

#### Список использованных источников:

1. Zhang H., Xu T., Li H., et al. StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks // Proc. IEEE Int. Conf. Comput. Vis. – 2017. – P. 5907–5915.
2. Wang T.-C., Liu M.-Y., Zhu J.-Y., et al. High-resolution image synthesis and semantic manipulation with conditional GANs // Proc. IEEE Conf. Comput. Vis. Pattern Recognit. – 2018. – P. 8798–8807.
3. Li X., Liu S., Zhou J., Lu X., Fernandez-Granda C., Zhu Z., Qu Q. Principled and Efficient Transfer Learning of Deep Models via Neural Collapse // Transactions on Machine Learning Research. – 2024. – Vol. 5. – Art. no. 537. – DOI: 10.55464/tmlr.v5i.537.
4. Ramesh A., Pavlov M., Goh G., et al. Zero-shot Text-to-Image Generation // Proc. Int. Conf. Mach. Learn. – 2021.
5. Karras T., Laine S., Aila T. A Style-Based Generator Architecture for Generative Adversarial Networks // Proc. IEEE Conf. Comput. Vis. Pattern Recognit. – 2019. – P. 4401–4410.
6. Carlini N., Mishra P., Shumailov I., et al. Poisoning and Distillation: A Synthesis of Neural Network Poisoning Attacks [Электронный ресурс] // arXiv preprint arXiv:2310.00065. – 2023. – Режим доступа: <https://arxiv.org/abs/2310.00065> – Дата доступа: 01.03.2025.
7. Esser P., Sutter E., Ommer B. A Variational U-Net for Conditional Appearance and Shape Generation // Proc. IEEE Conf. Comput. Vis. Pattern Recognit. – 2018. – P. 8857–8866.
8. Goodfellow I., Pouget-Abadie J., Mirza M., et al. Generative adversarial nets // Advances in Neural Information Processing Systems. – 2014. – Vol. 27.
9. Arjovsky M., Bottou L. Towards Principled Methods for Training Generative Adversarial Networks [Электронный ресурс] // arXiv preprint arXiv:1701.04862. – 2017. – Режим доступа: <https://arxiv.org/abs/1701.04862> – Дата доступа: 03.03.2025.
10. Karnewar A., Wang O. MSG-GAN: Multi-scale Gradient GAN for Stable Image Synthesis [Электронный ресурс] // arXiv preprint arXiv:2010.02517. – 2020. – Режим доступа: <https://arxiv.org/abs/2010.02517> – Дата доступа: 03.03.2025.
11. Huang R., Zhang H., Liu Y., Ma J. HiGAN: Hierarchical GAN for Conditional Image Generation [Электронный ресурс] // arXiv preprint arXiv:2204.08599. – 2022. – Режим доступа: <https://arxiv.org/abs/2204.08599> – Дата доступа: 10.03.2025.
12. Oktay O., Schlemper J., Le Folgoc L., Lee M., Heinrich M., Misawa K., Mori K., McDonagh S., Hammerla N., Kainz B., Glocker B., Rueckert D. Attention U-Net: Learning Where to Look for the Pancreas [Электронный ресурс] // arXiv preprint arXiv:1804.03999. – 2018. – Режим доступа: <https://arxiv.org/abs/1804.03999> – Дата доступа: 10.03.2025.
13. Isola P., Zhu J., Zhou T., Efros A. A. Image-to-Image Translation with Conditional Adversarial Networks [Электронный ресурс] // arXiv preprint arXiv:1611.07004. – 2017. – Режим доступа: <https://arxiv.org/abs/1611.07004> – Дата доступа: 20.03.2025.
14. Kovbasa, G. Evaluation Metrics and Multi-level GAN Approach for Medical Images = Метрики оценки и применение многоуровневых GAN для медицинских изображений / G. Kovbasa // Открытые семантические технологии проектирования интеллектуальных систем = Open Semantic Technologies for Intelligent Systems (OSTIS) : сборник научных трудов / Белорусский государственный университет информатики и радиоэлектроники ; редкол.: В. В. Голенков [и др]. – Минск, 2024. – Вып. 8. – С. 311–318.

# DEPENDENCE OF THE DISTRIBUTION OF INPUT AND OUTPUT DATA IN MULTI-LEVEL GENERATIVE ADVERSARIAL NETWORKS

*Kovbasa G.A.<sup>1</sup>*

*Belarusian State University of Informatics and Radioelectronics<sup>1</sup>, Minsk, Republic of Belarus*

*Azarov E.S. – Doctor of Technical Sciences*

**Annotation.** Multi-level Generative Adversarial Networks (GANs) refine data generation across sequential stages, where each level forms an intermediate distribution influenced by the original dataset. Distortions in these distributions may accumulate, reducing the final output's quality and diversity. This article examines how the structure and quality of training data affect these intermediate outputs and suggests techniques to mitigate cascading distortion. The use of U-Net-based architectures is shown to preserve local features and enhance realism by stabilizing distribution shifts. However, this approach can also limit the distribution domain, reducing the variety of generated data.

**Keywords.** Neural networks, multi-level generative adversarial networks, neural network collapse, data distribution.