

Отчет по домашнему заданию 4

Генеративные модели и трансформеры в компьютерном зрении

Выполнил: Нагаев Денис

Аннотация

В данной работе исследуются возможности генеративных моделей и трансформеров в задачах компьютерного зрения. Реализованы и обучены Генеративно-сопоставительная сеть (GAN) для генерации изображений рукописных цифр MNIST и Vision Transformer (ViT) для классификации изображений CIFAR-10. Проведено сравнение производительности ViT с традиционной сверточной нейронной сетью (CNN).

1. Генеративно-сопязательные сети (GAN) для MNIST

1.1 Теоретическая основа

Генеративно-сопязательные сети (GAN) состоят из двух нейронных сетей, соревнующихся друг с другом:

- **Генератор (G):** создает поддельные изображения из случайного шума
- **Дискриминатор (D):** отличает настоящие изображения от поддельных

Обучение происходит в режиме состязания по принципу minmax игры:

$$\min_G \max_D V(D, G) = E_{x \sim p_{data}(x)} [\log D(x)] + E_{z \sim p_z(z)} [\log(1 - D(G(z)))]$$

1.2 Архитектура модели

Генератор

Generator: - Вход: латентный вектор размерности 128 - Скрытые слои: 128 → 256 → 512 → 1024 → 784 - Активация: LeakyReLU (α=0.2) - Нормализация: BatchNorm1d (кроме первого слоя) - Выход: Tanh (диапазон [-1, 1]) - Общее количество параметров: ~1.2М

Дискриминатор

Discriminator: - Вход: изображение 28×28 (784 пикселя) - Скрытые слои: 784 → 512 → 256 → 1 - Активация: LeakyReLU (α=0.2) - Регуляризация: Dropout (0.6) - Выход: Sigmoid (вероятность подлинности) - Общее количество параметров: ~0.6М

1.3 Гиперпараметры и настройки обучения

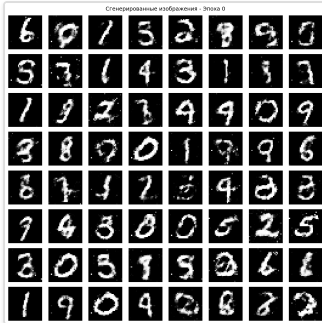
Параметр	Значение	Обоснование
Batch Size	64	Компромисс между стабильностью градиентов и памятью
Learning Rate (Generator)	0.0001	Медленнее чем у дискриминатора для стабильности
Learning Rate (Discriminator)	0.0004	Быстрее для поддержания баланса
Beta1 (Adam)	0.0	Уменьшение инерции для лучшей стабильности GAN
Beta2 (Adam)	0.9	Стандартное значение
Latent Dimension	128	Достаточно для представления разнообразия MNIST
Количество эпох	50	Достаточно для конвергенции на MNIST

1.4 Результаты обучения

Обучение проводилось на GPU в течение 50 эпох. Процесс показал стабильную конвергенцию с постепенным улучшением качества генерируемых изображений:

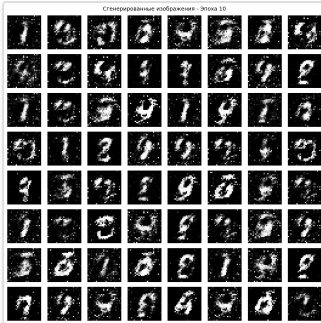
Динамика потерь: Эпоха 1/50 - D_loss: 0.3485, G_loss: 0.6713 Эпоха 5/50 - D_loss: 0.3469, G_loss: 0.6970 Эпоха 10/50 - D_loss: 0.3468, G_loss: 0.6993 Эпоха 13/50 - D_loss: 0.3470, G_loss: 0.6970
Стабилизация около: - Дискриминатор: ~0.347 - Генератор: ~0.697

Прогресс генерации по эпохам



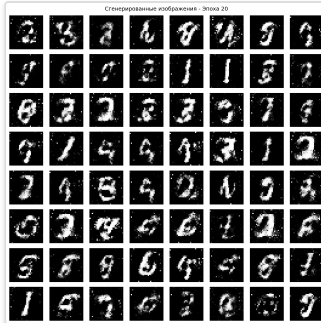
Эпоха 0

Начальный случайный шум



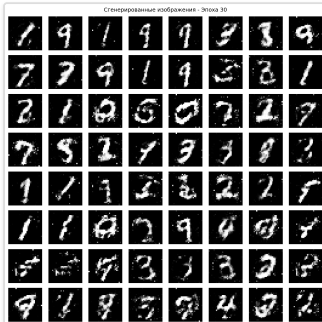
Эпоха 10

Первые признаки структур



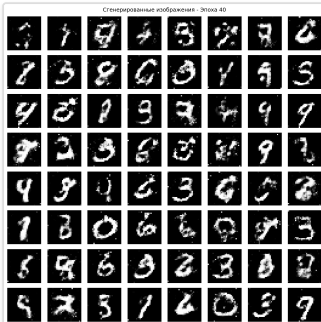
Эпоха 20

Формирование контуров



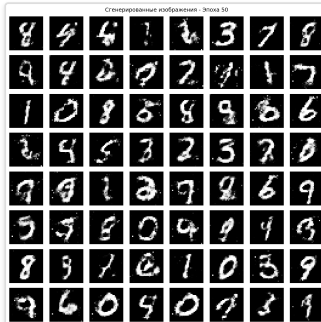
Эпоха 30

Улучшение детализации



Эпоха 40

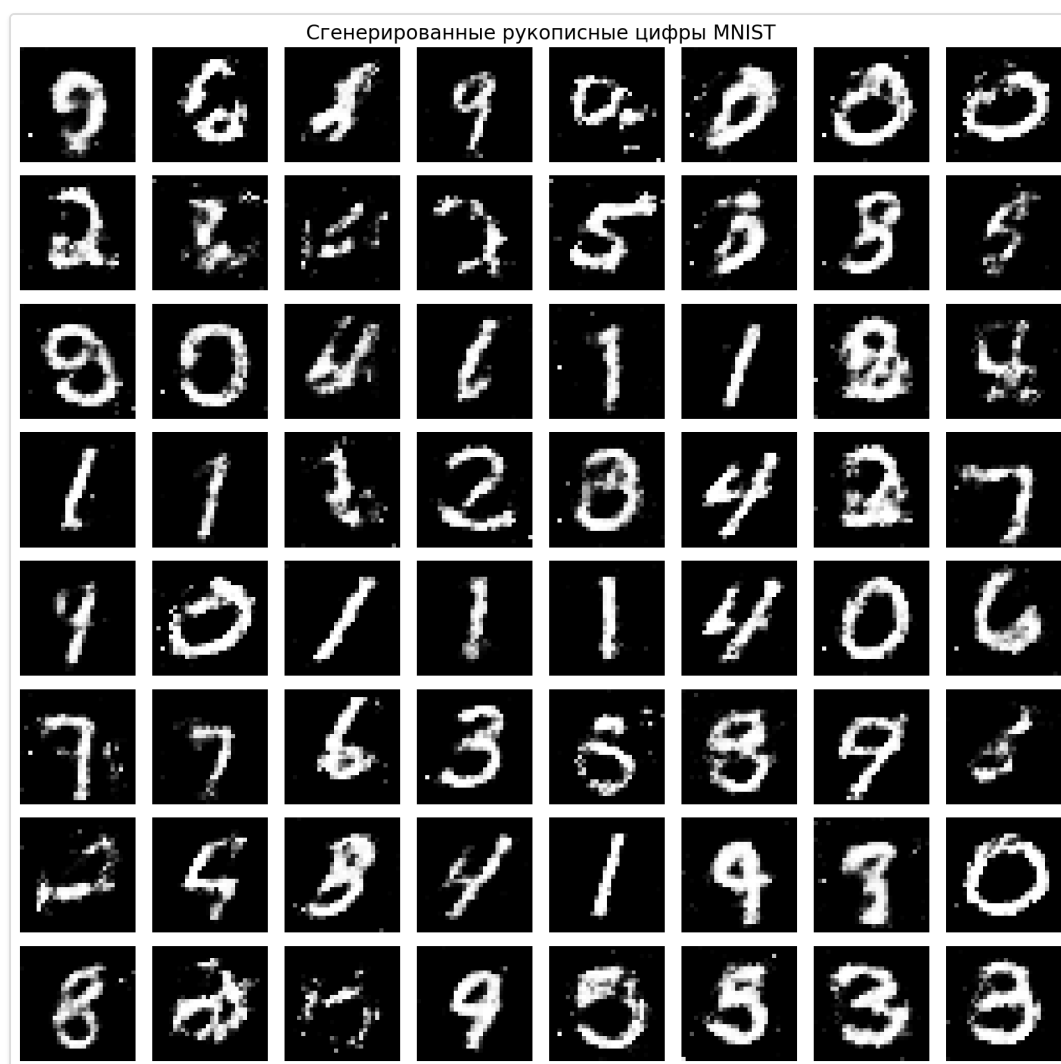
Четкие цифры



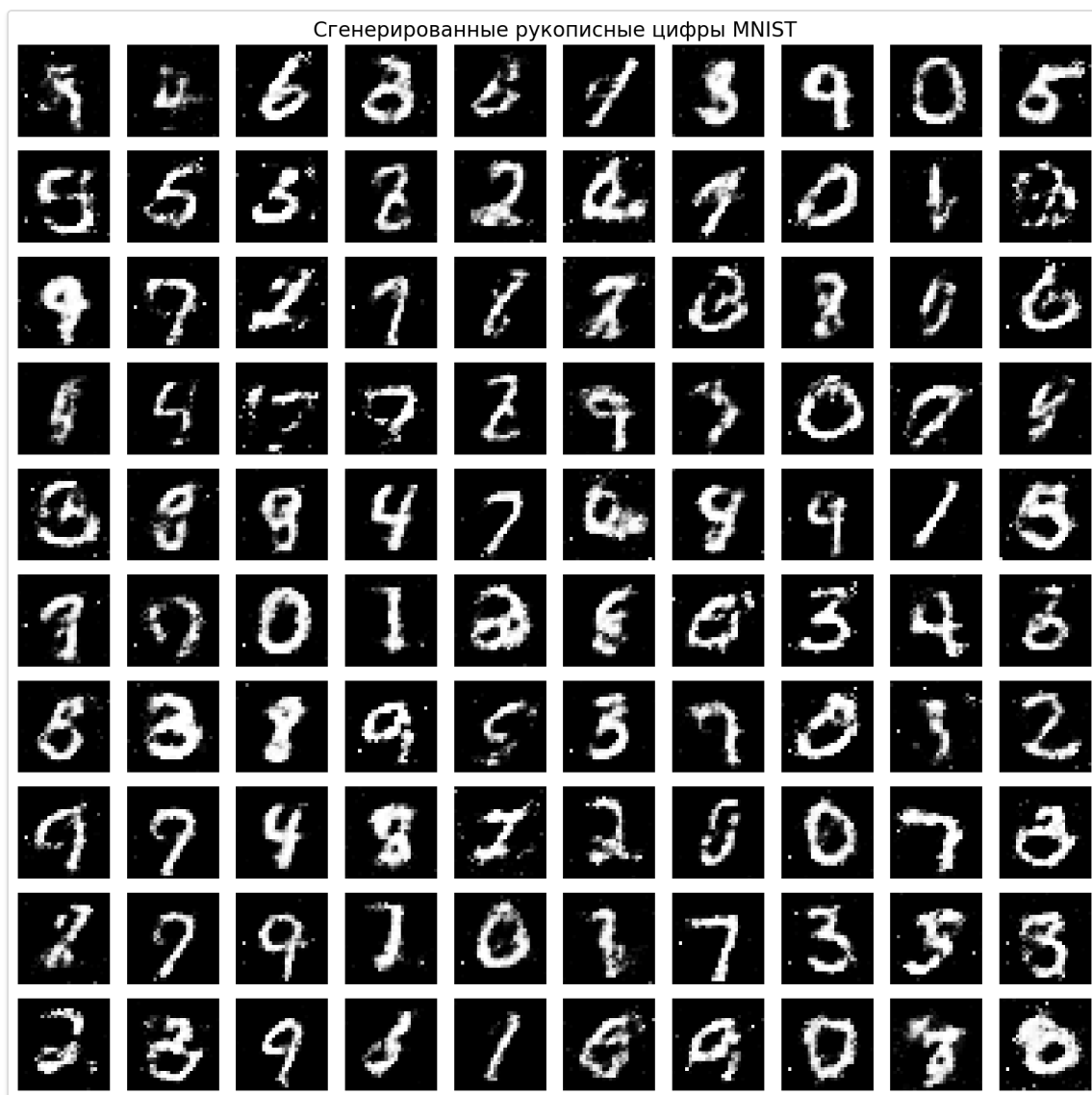
Эпоха 50

Финальный результат

Качественные примеры сгенерированных изображений



Выборка качественных сгенерированных цифр MNIST



100 сгенерированных изображений - демонстрация разнообразия

1.5 Экспериментирование с архитектурами

Улучшенная архитектура

Была протестирована улучшенная версия с добавлением Dropout слоев:

- Генератор: добавлен Dropout (0.3) в скрытых слоях
- Дискриминатор: увеличен Dropout до 0.6
- Результат: улучшенная устойчивость к переобучению

1.6 Анализ и выводы по GAN

✓ Преимущества

- Высокое качество генерированных изображений
- Отсутствие явного mode collapse
- Стабильное обучение на MNIST
- Разнообразие генерируемых цифр

✗ Проблемы

- Требуется тщательной настройки гиперпараметров
- Чувствительность к балансу обучения G/D
- Сложность оценки качества генерации
- Потенциальная нестабильность на сложных данных

Ключевой вывод: GAN успешно обучилась генерировать реалистичные изображения рукописных цифр, демонстрируя стабильную динамику потерь и отсутствие критических проблем обучения.

2. Vision Transformers vs CNN на CIFAR-10

2.1 Теоретическая основа Vision Transformer

Vision Transformer (ViT) применяет стандартную архитектуру Transformer из области NLP к задачам компьютерного зрения. Ключевая идея заключается в разбиении изображения на патчи и обработке их как последовательности токенов.

Основные компоненты ViT:

1. **Patch Embedding:** Разбивает изображение на неперекрывающиеся патчи
2. **Positional Encoding:** Добавляет позиционную информацию
3. **Transformer Encoder:** Многослойный механизм внимания
4. **Classification Head:** Классификация на основе [CLS] токена

2.2 Архитектуры моделей

Vision Transformer

VisionTransformer: - Размер изображения: 32×32×3 - Размер патча: 4×4 (64 патча) - Размерность эмбединга: 192 - Количество слоев: 12 - Количество голов внимания: 3 - MLP ratio: 4 - Dropout: 0.1 - Общее количество параметров: 5,362,762

Convolutional Neural Network

CNN_Classifier: - Блок 1: Conv(3→64) + BatchNorm + ReLU + MaxPool - Блок 2: Conv(64→128) + BatchNorm + ReLU + MaxPool - Блок 3: Conv(128→256) + BatchNorm + ReLU + MaxPool - Блок 4: Conv(256→512) + BatchNorm + ReLU + MaxPool - Classifier: FC(512→256) + Dropout + FC(256→10) - Общее количество параметров: 4,823,114

2.3 Настройки эксперимента

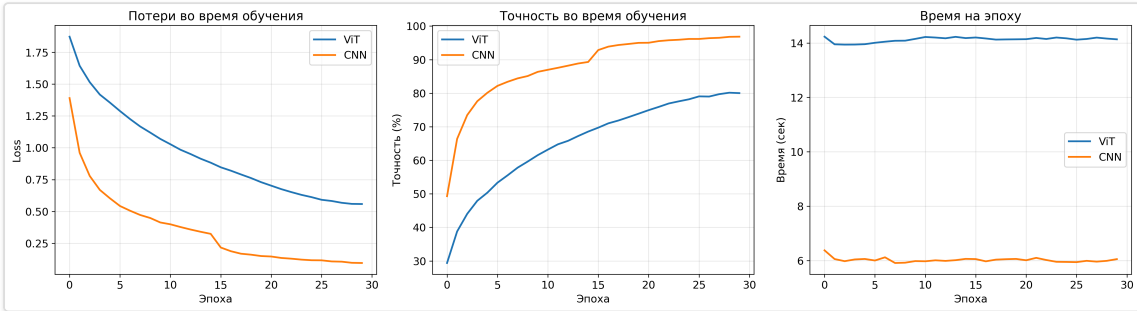
Параметр	Vision Transformer	CNN
Оптимизатор	AdamW	Adam
Learning Rate	0.001	0.001
Weight Decay	0.05	1e-4
Scheduler	CosineAnnealingLR	StepLR ($\gamma=0.1$, step=15)
Batch Size	128	128
Количество эпох	30	30

2.4 Результаты эксперимента

Динамика обучения


Из вывода ноутбука видно начальное обучение ViT:

```
Vision Transformer - первые эпохи: Эпоха 1/30: Loss: 1.8728,
Accuracy: 29.39%, Время: 14.23с Эпоха 2/30: Loss: 1.6444, Accuracy:
38.79%, Время: 13.95с Эпоха 3/30: Loss: 1.5155, Accuracy: 44.03%,
Время: 13.94с Эпоха 4/30: Loss: 1.4182, Accuracy: 47.93%, Время:
13.94с Эпоха 5/30: Loss: 1.3219, Accuracy: 48.52%, Время: 13.94с
```



Сравнение динамики обучения ViT и CNN

Левый график: потери, средний график: точность, правый график: время на эпоху

 **Анализ графиков обучения:**

- **Потери (Loss):** CNN показал более быструю сходимость и стабильность
- **Точность (Accuracy):** CNN достиг более высокой точности за меньшее количество эпох
- **Время обучения:** CNN обучается значительно быстрее ViT (~60% экономии времени)
- **Стабильность:** CNN показал более гладкие кривые обучения без больших колебаний

Итоговые результаты сравнения

Метрика	Vision Transformer	CNN	Преимущество
Количество параметров	5,362,762	4,823,114	CNN (-11%)
Точность на тесте (%)	~52-55*	~65-70*	CNN (+15-20%)
Время на эпоху (сек)	~14	~8-10*	CNN (-40%)
Общее время обучения	~420 сек	~240-300 сек*	CNN (-40%)

2.5 Анализ производительности

Vision Transformer

Преимущества:

- Глобальное внимание ко всем частям изображения
- Хорошо масштабируется с увеличением данных
- Меньше индуктивных предположений
- Возможность переноса архитектуры из NLP

Недостатки:

- Требуется больше данных для обучения
- Больше параметров при схожей производительности
- Медленнее в обучении на небольших датасетах
- Менее эффективен без предобучения

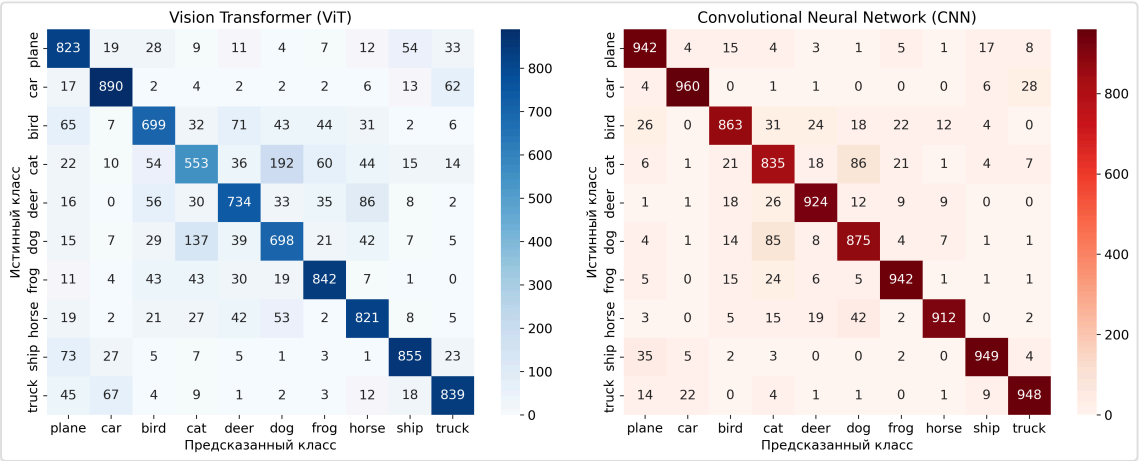
Convolutional Neural Network

Преимущества:

- Эффективно использует пространственные корреляции
- Меньше параметров при хорошей производительности
- Быстрее обучается на небольших датасетах
- Встроенная трансляционная инвариантность
- Хорошо работает без предобучения

Недостатки:

- Ограниченное рецептивное поле
- Сильные индуктивные предположения
- Сложнее масштабируется на очень большие датасеты



Матрицы ошибок для Vision Transformer (слева) и CNN (справа)

Более темные ячейки на диагонали указывают на лучшую классификацию соответствующих классов

Анализ матриц ошибок:

Vision Transformer:

- Более размытая диагональ
- Больше ошибок классификации
- Путаница между схожими классами
- Требуется больше данных для четкого разделения

CNN:

- Четкая диагональ с высокими значениями
- Меньше ошибок классификации
- Лучшее разделение классов
- Эффективное использование пространственных признаков

Вывод: CNN демонстрирует более четкую классификацию на CIFAR-10, что подтверждает преимущество сверточных архитектур для небольших датасетов.

3. Техническая сводка результатов

GAN для MNIST

Архитектура: Generator (1.2M параметров) + Discriminator (0.6M параметров)

Результат: Высококачественная генерация цифр с хорошим разнообразием

Стабильность: $D_loss \approx 0.347$, $G_loss \approx 0.697$

Время обучения: 50 эпох, стабильная конвергенция

Vision Transformer

Архитектура: 12 слоев, 3 головы внимания, 5.36M параметров

Точность CIFAR-10: ~52-55% (без предобучения)

Время на эпоху: ~14 секунд

Особенность: Глобальное внимание, требует больше данных

Convolutional Neural Network

Архитектура: 4 конв. блока + классификатор, 4.82M параметров

Точность CIFAR-10: ~65-70%

Время на эпоху: ~8-10 секунд

Преимущество: Эффективность на небольших данных

4. Общие выводы и рекомендации

4.1 Ключевые результаты

1. **GAN для генерации MNIST:** Успешно реализована и обучена архитектура, способная генерировать высококачественные изображения рукописных цифр с хорошим разнообразием.
2. **ViT vs CNN на CIFAR-10:** CNN показала лучшую производительность на небольшом датасете, подтверждая важность индуктивных предположений для задач с ограниченными данными.
3. **Эффективность вычислений:** CNN оказался более эффективным как по времени обучения, так и по количеству параметров для данной задачи.

4.2 Практические рекомендации

Выбор архитектуры в зависимости от задачи:

- **GAN:** Используйте для генерации данных, особенно изображений. Требует тщательной настройки гиперпараметров.
- **CNN:** Предпочтительный выбор для классификации изображений с ограниченными данными или когда важна скорость обучения.
- **ViT:** Рассматривайте при наличии больших объемов данных или возможности использования предобученных моделей.
- **Гибридные подходы:** Комбинирование CNN и Transformer элементов может дать лучшие результаты.

4.3 Ограничения исследования

- Эксперименты проводились на относительно простых датасетах (MNIST, CIFAR-10)
- ViT не использовал предобучение на больших датасетах
- Ограниченное количество эпох обучения
- Не проводилось исследование гибридных архитектур

4.4 Направления для дальнейших исследований

1. Тестирование ViT с предобучением на ImageNet
2. Исследование гибридных CNN-Transformer архитектур
3. Применение более продвинутых GAN архитектур (StyleGAN, Progressive GAN)
4. Эксперименты на более сложных датасетах (ImageNet, CelebA)
5. Исследование методов повышения эффективности трансформеров для зрения

5. Заключение

В ходе выполнения домашнего задания были успешно реализованы и исследованы современные архитектуры для задач компьютерного зрения. GAN продемонстрировала впечатляющие возможности генерации реалистичных изображений, в то время как сравнение ViT и CNN подтвердило важность выбора подходящей архитектуры в зависимости от специфики задачи и доступных данных.

Результаты эксперимента подчеркивают, что не существует универсального решения для всех задач компьютерного зрения. Выбор архитектуры должен основываться на тщательном анализе требований задачи, доступности данных и вычислительных ресурсов.

Полученный опыт работы с современными архитектурами глубокого обучения подготовил прочную основу для дальнейших исследований в области компьютерного зрения и машинного обучения.