

## **СИСТЕМНЫЙ АНАЛИЗ, УПРАВЛЕНИЕ И ОБРАБОТКА ИНФОРМАЦИИ**

УДК 004.032.26

DOI: 10.24412/2071-6168-2023-8-88-89

### **МОДЕЛИРОВАНИЕ И УПРАВЛЕНИЕ ДВУНАПРАВЛЕННОЙ СОРЕВНОВАТЕЛЬНОЙ НЕЙРОСЕТЬЮ И СОРЕВНОВАТЕЛЬНОЙ НЕЙРОСЕТЬЮ С ВСПОМОГАТЕЛЬНЫМ КЛАССИФИКАТОРОМ В ЗАДАЧЕ УВЕЛИЧЕНИЯ РАЗРЕШЕНИЯ И ОБРАБОТКИ ИНФОРМАЦИИ**

Ю.Н. Кожубаев, В.А. Прокофьев

*В работе рассматриваются две различные модели GAN (Generative adversarial network – генеративная состязательная сеть) нейросетей в задачах увеличения качества изображений. Для оценки их эффективности предлагается использовать такие измеряемые характеристики, как точность генератора и дискриминатора GAN моделей, бинарная и межгрупповая ошибки и время работы. Для измерения избранных параметров был проведен ряд опытов с использованием двух разных наборов данных: черно-белых фотографий элементов одежды и черно-белых фотографий примеров рукописного текста, оба набора были разбиты на 10 групп по типу предмета или текста, содержащегося на фотографии. Генеративные модели позволяют представить структуру, лежащую в основе данных. Хотя целью является точное генерирование данных с репрезентативной статистикой, сила генеративных моделей частично заключается в представлении данных в виде гипотетической сжатой версии лежащей в их основе структуры. Генеративные состязательные сети – это уникальная генеративная система обучения, в которой используются две отдельные модели с противоположными, конкурирующими или состязательными целями. Автоэнкодеры на основе нейронных сетей могут быть использованы для обучения фильтра удаления шума на основе набора данных, из которого необходимо удалить шум.*

*Ключевые слова: состязательные нейронные сети, глубокое обучение, улучшение разрешения, обработка изображений.*

Первоначально нейронные сети были разработаны для решения важных дискриминационных задач, таких как классификация и регрессия. Нейронные сети в основном использовались для работы с неструктурированными данными, где каждый объект представлен вектором независимых признаков [1-3]. Позднее нейронные сети были применены в задачах обработки изображений благодаря предложенному сверточному слою.

С тех пор сверточные нейронные сети успешно применяются для решения задач компьютерного зрения, таких как классификация, обнаружение и сегментация изображений. В указанных задачах нейронная сеть обучается отображать входные объекты в какое-либо представление, полезное на практике: метка изображения для классификации, семантическая карта для сегментации или ограничивающая рамка для обнаружения. Сверточные нейронные сети также успешно применялись для решения та-

ких задач восстановления изображений, как увеличение разрешения, очистка «шума» или восстановление, где задача заключается в создании изображения, а не в предсказании характеристик и атрибутов данного изображения [4-6].

В отличие от моделей, которые генерируют данные из параметрического распределения, таких как направленные графовые модели и вариационные автоэнкодеры или неориентированные графовые модели (ограниченные машины Больцмана и глубокие машины Больцмана [10]), GAN не полагаются на оценку максимального правдоподобия для обучения. Скорее, обучение GAN, состоящей из двух конкурирующих сетей - дискриминатора и генератора - требует только обратного распространения обучающего сигнала, исходящего из выученной объективной функции, соответствующей потерям дискриминатора, обученного недобросовестным образом [7-9].

**Классическая архитектура GAN нейросетей.** Генеративные состязательные сети моделируют распределение данных как преобразование фиксированного скрытого распределения  $p_Z(z)$  для  $z \in \Omega_Z$  [3]. Это преобразование, называемое генератором, выражается в виде детерминированной прямой сети  $G : \Omega_Z \rightarrow \Omega_X$  с  $p_G(x|z) = \delta(x - G(z))$  и  $p_G(x) = E_z \sim p_Z[p_G(x|z)]$ . Цель состоит в том, чтобы обучить генератор так, чтобы  $p_G(x) \approx p_X(x)$ .

Генеративная состязательная сеть (GAN) состоит из двух нейронных сетей, обученных в оппозиции друг к другу. Генератор  $G$  принимает случайный вектор шума  $z$  в качестве входных данных и генерирует изображение  $X_{fake} = G(z)$ . Дискриминатор  $D$ , с другой стороны, принимает либо обучающее изображение, либо синтезированное изображение от генератора и производит распределение вероятности  $P(S|X)=D(X)$  относительно всех возможных источников изображения [4].

Данный процесс можно описать следующей формулой:

$$L = E[\log P(S = real|X_{real})] + E[\log P(S = fake|X_{fake})] \quad (1)$$

Генератор обучается минимизировать второй член в Уравнении 1. Базовая схема GAN может быть дополнена с помощью побочной информации. Одна из стратегий заключается в том, чтобы снабдить генератор и дискриминатору метками классов, чтобы получить классово условные выборки. Синтез с учетом классов может значительно улучшить качество генерируемых образцов. Более богатая побочная информация, такая как подписи к изображениям и локализация ограничительных рамок, может еще больше повысить качество выборки [10-12].

Вместо того чтобы подавать побочную информацию на дискриминатор, можно поручить дискриминатору восстановить добавочную информацию. Это делается путем модификации дискриминатора, чтобы он содержал вспомогательную сеть декодера  $1$ , которая выдает метку класса для обучающих данных или подмножество латентных переменных, на основе которых генерируются образцы.

**Принцип обучения.** Основной принцип обучения GAN заключается в уменьшении различий между истинным распределением данных  $p(x)$  и поддельным распределением данных  $p(G(z; \theta))$ . Поскольку есть много способов оценки расстояния между двумя распределениями, таких, как расхождение Дженсена-Шеннона, расхождение Куллбэка-Лейблера и интегральная метрика вероятности, существует множество видов потерь GAN, включая потери насыщенного GAN, потери ненасыщенного GAN, потери Васерштейна GAN, потери наименьшего квадрата GAN, потери петлевого GAN, потери f-дивергенции GAN и потери релятивистского GAN. Эмпирически, производительность GAN-потерь зависит от приложения, а также от архитектуры сети. На данный момент не существует однозначного мнения о том, какой из них лучше [13-15]

**Алгоритм обучения.** Приводим общую формулировку задачи обучения GAN, которая включает в себя несколько популярных формулировок. Для шага обновления дискриминатора задача обучения имеет вид:

$$E_{x \sim D}[f_D(Dx; \varphi)] + E_{z \sim Z}[f_G(D(G(z, \theta); \varphi))], \quad (2)$$

где  $f_D$  и  $f_G$  - выходные слои, преобразующие результаты, вычисленные дискриминатором D, в оценки классификации для реального и поддельного изображений, соответственно.

Формула для шага обновления генератора имеет следующий вид:

$$E_{z \sim z}[g_G(D(G(z; \theta); \varphi))] \quad (3)$$

где  $g_G$  - выходной слой, который преобразует результат, вычисленный дискриминатором, в оценку для поддельного изображения. В табл. 1 приводится сравнение функций  $f_D$ ,  $f_G$  и  $g_G$  для нескольких популярных методов расчета GAN-потерь [16-18].

Таблица 1

Функции расчета потерь

Метод	$f_D(x)$	$f_G(x)$	$g_G(x)$
Насыщенный	$\log \sigma(x)$	$\log (1 - \sigma(x))$	$\log (1 - \sigma(x))$
Ненасыщенный	$\log \sigma(x)$	$\log (1 - \sigma(x))$	$-\log \sigma(x)$
Васерштейна	$x$	$-x$	$-x$
Наименьшего квадрата	$-(x - 1)^2$	$-x^2$	$(x - 1)^2$
Петлевой	$(0, x-1)$	$(0, -x-1)$	$-x$

Для обучения GAN обычно используются два варианта схем стохастического градиентного спуска/восхождения: схема одновременного обновления и схема попеременного обновления. Пусть  $V_D$  и  $V_G$ , - целевые функции в (1) и (2), соответственно. При одновременном обновлении каждая итерация обучения содержит шаг обновления дискриминатора и шаг обновления генератора, заданные следующим образом

$$\varphi^{t+1} = \varphi^t + a_D \frac{\partial V_D(\theta^t, \varphi^t)}{\partial \varphi}; \quad (4)$$

$$\theta^{t+1} = \theta^t - a_G \frac{\partial V_G(\theta^t, \varphi^t)}{\partial \theta}; \quad (5)$$

где  $a_D$  и  $a_G$  - скорости обучения для генератора и дискриминатора, соответственно. При попеременном обновлении каждая итерация обучения состоит из одного шага обновления дискриминатора, за которым следует шаг обновления генератора, заданный следующим образом

$$\varphi^{t+1} = \varphi^t + a_D \frac{\partial V_D(\theta^t, \varphi^t)}{\partial \varphi}; \quad (6)$$

$$\theta^{t+1} = \theta^t - a_G \frac{\partial V_G(\theta^t, \varphi^{t+1})}{\partial \theta}; \quad (7)$$

Обратим внимание, что в схеме попеременного обновления генератора (6) использует новые обновленные параметры дискриминатора  $\theta^{t+1}$ , а в схеме одновременного обновления - нет.

Эти две схемы имеют свои плюсы и минусы. Схема одновременного обновления может быть вычислена более эффективно, так как большая часть вычислений на двух этапах может быть разделена. С другой стороны, метод чередования обучения имеет тенденцию быть более устойчивым, поскольку обновление генератора основано на последнем дискриминаторе. В современных исследованиях, касающихся GAN, метод чередования обучения преобладает. Иногда дискриминатор обновляется несколько раз перед выполнением вычислений [19-21].

Среди различных алгоритмов градиентного спуска/восхождения для обучения GAN пользуется популярностью алгоритм ADAM, основанный на адаптивных оценках моментов первого и второго порядка. ADAM имеет несколько параметров, определяемых пользователем. Обычно первый момент устанавливается на 0, а второй - на 0,999. Скорость обучения для обновления дискриминатора часто устанавливается в 2-4 раза больше, чем скорость обучения для обновления генератора (обычно устанавливается на 0,0001), что называется шкалой двухкратного обновления (TTUR).

**Исследуемые модели. Соревновательная нейросеть с вспомогательным классификатором (AC-GAN).** Из различных исследований [22-24] известно, что принуждение модели к выполнению дополнительных задач улучшает производительность

при выполнении исходной задачи. Кроме того, вспомогательный декодер может использовать предварительно обученные дискриминаторы (например, классификаторы изображений) для дальнейшего улучшения обрабатываемых изображений. Исходя из этих факторов, данная модель объединяет обе стратегии использования дополнительной информации. Другими словами, модель, которая будет рассмотрена ниже, является условно-классовой, но с дополнительным декодером, который выполняет задачу восстановления меток классов.

Рассматриваемый вариант архитектуры GAN-сети был назван вспомогательным классификатором GAN (или AC-GAN). В AC-GAN каждый сгенерированный образец имеет соответствующую метку класса  $c \sim p_c$ , в дополнение к шуму  $z$ .  $G$  использует их для генерации изображений  $X_{fake} = G(c, z)$ . Дискриминатор дает как распределение вероятности по источникам, так и распределение вероятности по меткам классов,  $P(S | X)$ ,  $P(C | X) = D(X)$ . Целевая функция состоит из двух частей: лог-вероятность правильного источника,  $L_S$ , и лог-вероятность правильного класса,  $L_C$ .

$$L_S = E[\log P(X_{real})] + E[\log P(S = fake | X_{fake})]; \quad (8)$$

$$L_C = E[\log P(X_{real})] + E[\log P(C = c | X_{fake})] \quad (9)$$

$D$  обучается максимизировать  $L_S + L_C$ , в то время как  $G$  обучается максимизировать  $L_C - L_S$ . AC-GAN обучаются представлению для  $z$ , которое не зависит от метки класса.

**Двухнаправленная соревновательная нейросеть (BiGAN).** Общая модель изображена на рис. 1. В данной модели, в дополнение к генератору  $G$  из стандартной структуры GAN, присутствует кодировщик  $E$ , который сопоставляет данные  $x$  со скрытыми представлениями  $z$ . Дискриминатор в BiGAN проводит различие не только в области данных ( $x$  против  $G(z)$ ), но и одновременно в пространстве данных и скрытом пространстве (координированный набор  $(x, E(x))$  против  $(G(z), z)$ ). Здесь скрытый компонент может быть выходом кодировщика  $E(x)$  или входом генератора  $z$  [25-27].

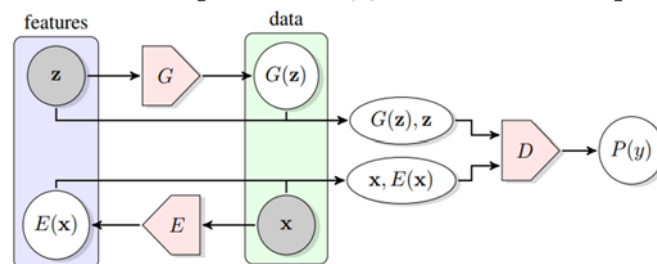


Рис. 1. Структурная схема BiGAN

Исходя из этого следует, что в модели типа BiGAN кодировщик  $E$  должен научиться инвертировать генератор  $G$ . Эти два модуля не связаны друг с другом напрямую: кодировщик не получает данных с выходов генератора ( $E(G(z))$  не вычисляется), и наоборот [28-30]

Поскольку кодировщик BiGAN учится предсказывать признаки  $z$  по данным  $x$ , после обучения он может служить полезным представлением признаков для соответствующих семантических задач, подобно тому, как полностью контролируемые визуальные модели, обученные предсказывать семантические "метки" для изображений, служат мощными представлениями признаков для соответствующих визуальных задач. В этом контексте скрытое представление  $z$  можно рассматривать как "метку" для  $x$ , которая ставится без необходимости наблюдения.

BiGAN являются надежным и очень общим подходом к обучению признаков без учителя, не делая никаких предположений о структуре или типе данных, к которым они применяются.

**Методология.** Оценка качества моделей увеличения качества изображений является сложной и комплексной задачей как из-за большого числа вероятностных переменных, влияющих на качество и скорость процесса обработки изображений (разреше-

ние, содержаний изображений, обширность зашумленности), так и из-за отсутствия однозначного объективного критерия оценки качества финального результата. Однако в описанных ниже разделах будет проведено несколько опытов, целью которых ставится оценка объективно измеримых критериев работы, таких как скорость работы и сходимость ошибки, выбранных типов GAN моделей при максимально равных начальных условиях. Целью проводимых опытов ставится получение количественных показателей, на основе которых можно проводить выбор модели GAN сети под конкретные задачи [31-33].

**Настройка параметров.** В силу различия структур GAN моделей, одни обладают как общими, так и индивидуальными для каждой модели параметрами.

**AC-GAN.** В AC-GAN генератор  $G$  представлен последовательностью слоев "деконволюции", которые преобразуют шум  $z$  и класс  $c$  в изображение. Дискриминатор  $D$  представлен глубокой сверточной нейронной сетью с функцией активации Leaky ReLU. Кроме того, стоит отметить, что сокращение изменчивости, вызванной всеми 1000 классами ImageNet, приводит к значительному улучшению качества обучения.

**BiGAN.** В каждом эксперименте каждый модуль -  $D$ ,  $G$  и  $E$  - представлен глубокой параметрической (многослойной) сетью. Дискриминатор BiGAN  $D(x, z)$  получает данные  $x$  в качестве начального входа и на каждом последующем линейном слое скрытое представление  $z$  изменяется с помощью выученного линейного преобразования для размерности скрытого слоя, и затем добавляется к входу нелинейной функции.

**Настройка общих параметров.** Во всех последующих опытах каждая из нейросетевых моделей будет обучаться на наборе изображений высокого разрешения ImageNet.

Рассчитаем зависимость ошибок генератора ( $G$ ) и дискриминатора ( $D$ ) от времени работы программы для разных структур GAN при одинаковых размерах обрабатываемых изображений и числе итераций.

Основной принцип работы GAN сети – состязание генератора ( $G$ ) и дискриминатора ( $D$ ). Функция  $G$  стремится максимизировать ошибку функции  $D$ , тем самым обучаясь создавать правдоподобные изображения на основе входных данных [34-36].

Опытным путем найдем и сравним время работы каждого алгоритма при одинаковых входных данных, одинаковом разрешении финального изображения и количестве итераций. Для сравнения эффективности работы структур нейросетей проведем ряд опытов. Целью их проведения является измерение следующих параметров: ошибка, бинарная точность и категориальная точность генератора и дискриминатора обеих моделей. Опыты проведем на двух различных базах данных, первая состоит из черно-белых фотографий элементов одежды (всего 10 типов элементов), вторая - черно-белых примеров рукописных чисел от 0 до 9.

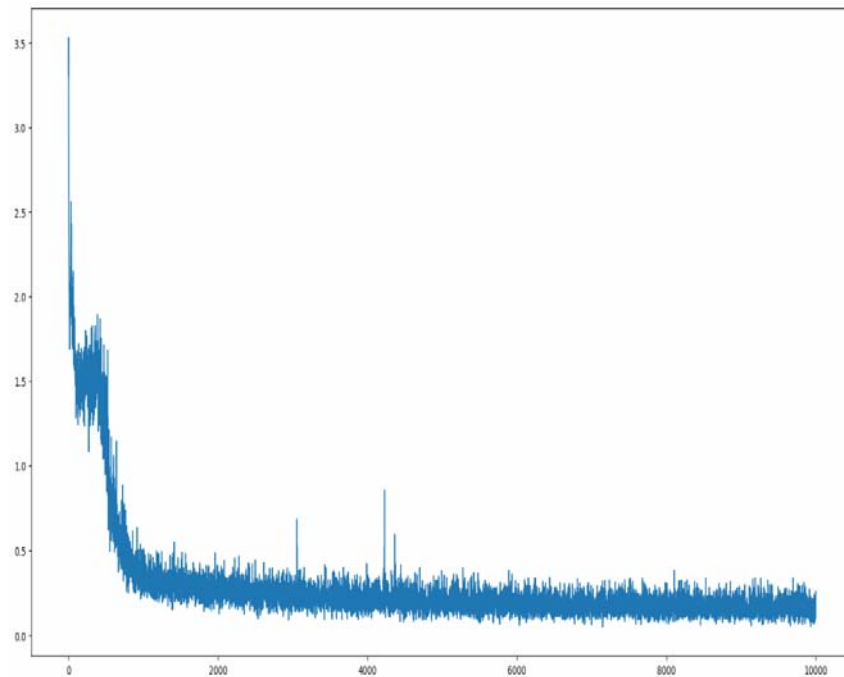
**Результаты и обсуждение. AC-GAN.** Возможно увеличить разрешение синтезированных изображений, выполнив билинейную интерполяцию. Хотя это позволяет создавать изображения более высокого разрешения, полученные изображения будут представлять собой размытые версии фотографий с низким разрешением, которые трудно отличить друг от друга. Поэтому цель модели синтеза изображений заключается не только в создании изображений с высоким разрешением, но также в получении более различных изображений с высоким разрешением, которые не могут быть подменены изображениями с более низким разрешением.

В ходе опытов с использованием набора фотографий одежды ошибка дискриминатора уменьшилась вдвое за первые 500 эпох обучения сети, после чего установилось на значениях 1-3% за 3000 эпох. При максимальном значении 35%.

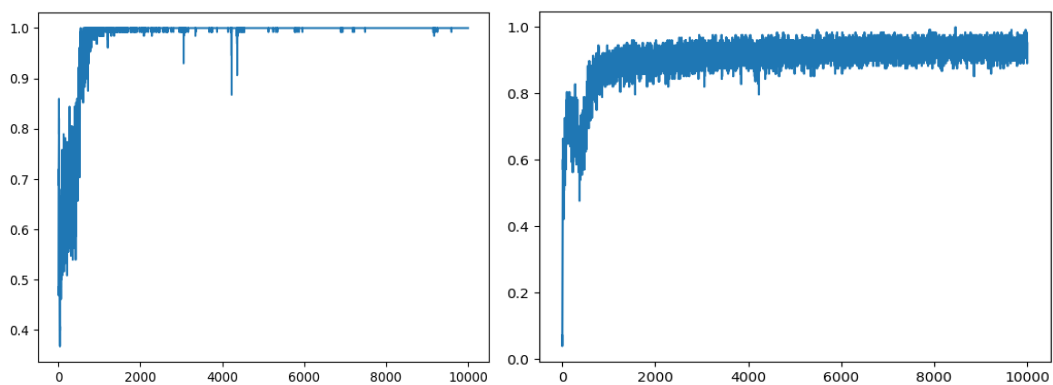
Бинарная точность дискриминатора приближается к 1 за 3000 эпох, после чего остается таковой, за исключением небольших единичных отклонений в некоторых эпохах.

Точность "распознавания" категорий для дискриминатора при длительном обучении колеблется от 1 до 0,87, стабилизируя в этих значениях за 3000 эпох.

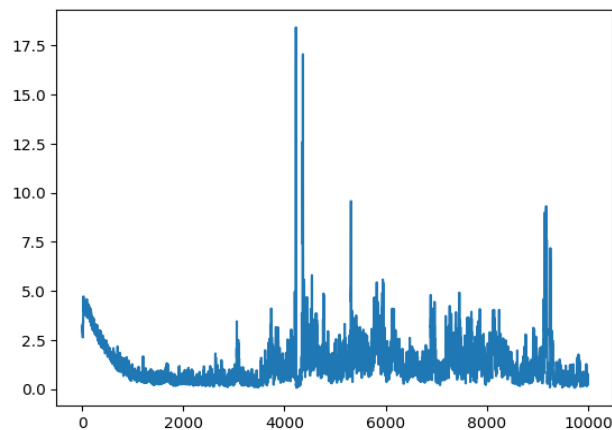
Также были произведены аналогичные измерения для генератора. Его ошибка показала себя менее стабильной, в сравнении с ошибкой дискриминатора. По итогу 10000 эпох она установилась на значениях 0-1,5 при максимальном за все время 17,5.



**Рис. 2. Ошибка дискриминатора AC-GAN в работе с изображениями элементов одежды**

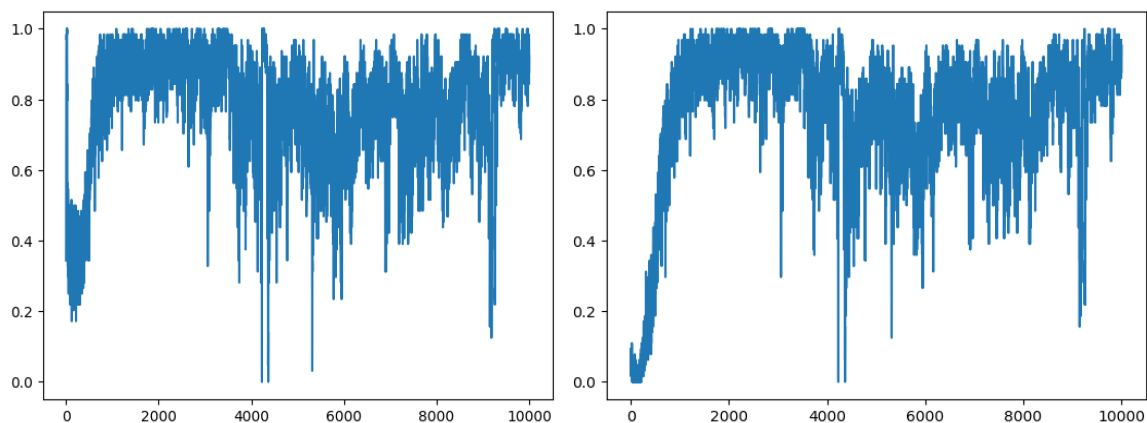


**Рис. 3. Бинарная точность и точность распознавания дискриминатора AC-GAN в работе с изображениями элементов одежды**



**Рис. 4. Ошибка генератора AC-GAN в работе с изображениями элементов одежды**

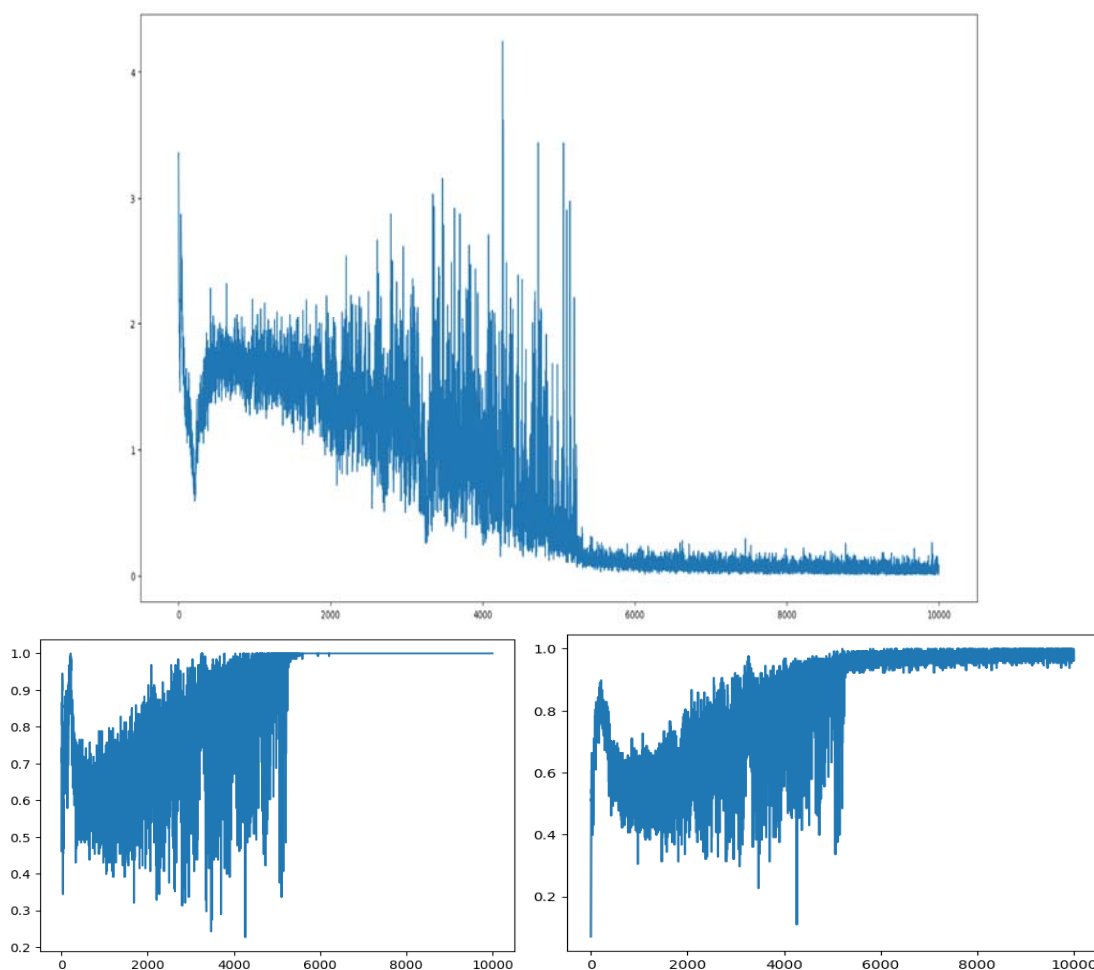
Бинарная точность и точность распознавания категорий имеют схожий процесс изменения, останавливаясь на значении 1-0,8.



**Рис. 5. Бинарная точность и точность распознавания генератора AC-GAN в работе с изображениями элементов одежды**

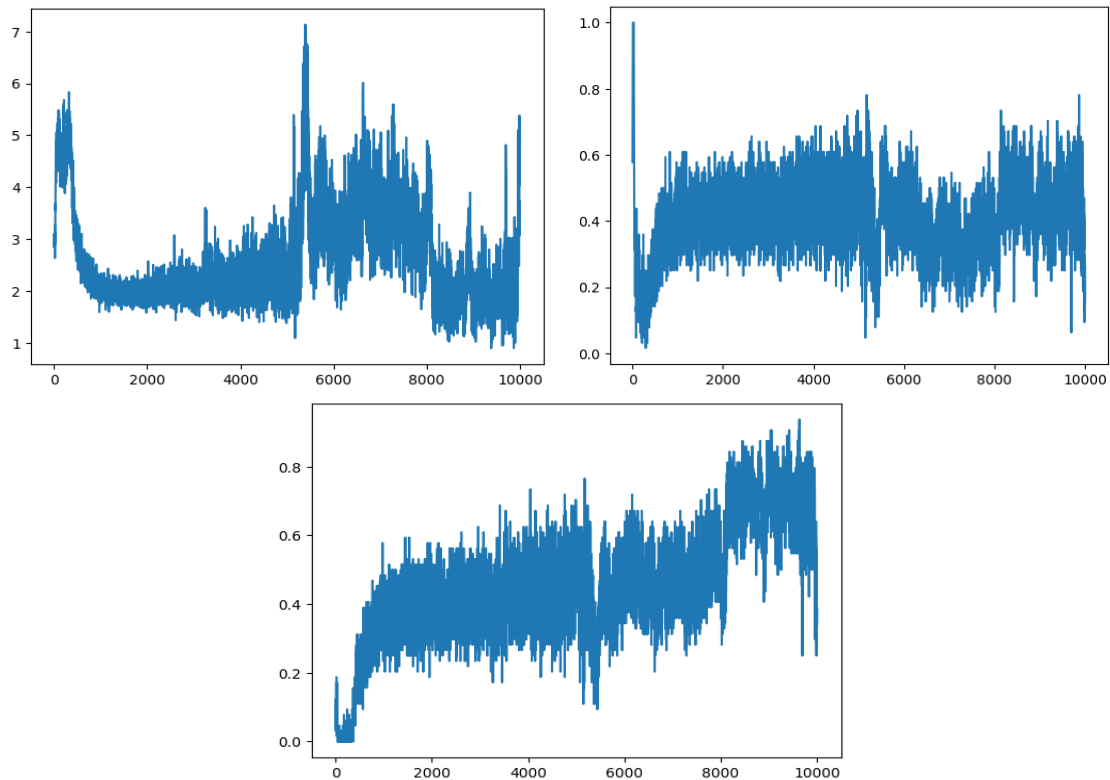
Проанализируем поведение параметров сети при использовании базы изображений рукописных чисел.

Ошибки дискриминатора показывают сильный разброс в значениях в первых 5000 эпох обучения, однако резко стабилизируются во второй половине.



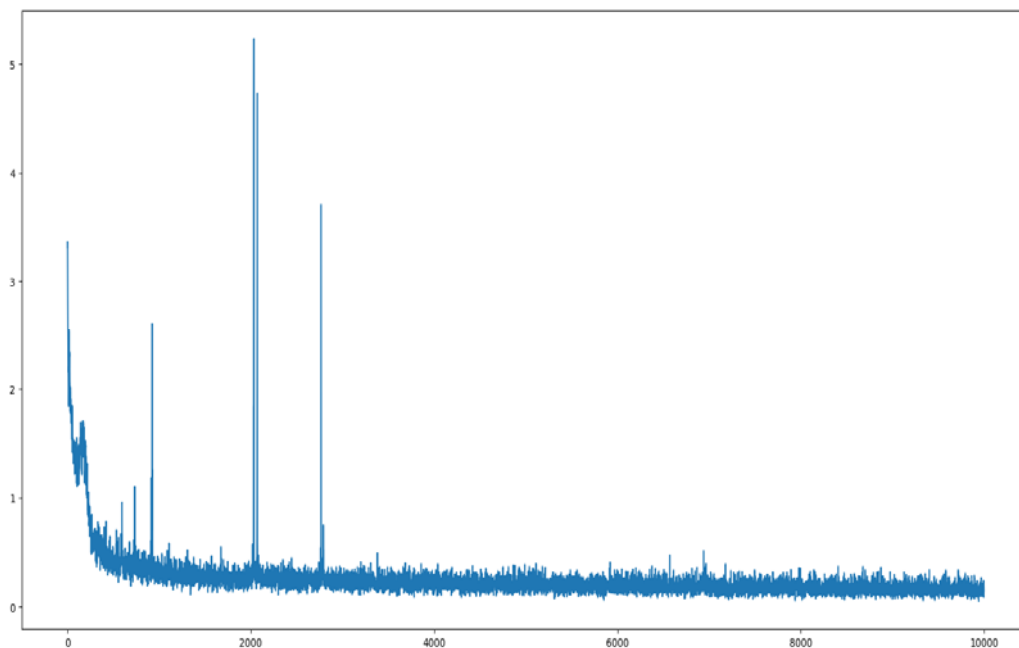
**Рис. 6. Ошибка и точности дискриминатора AC-GAN в работе с изображениями рукописных чисел**

Генератор в задаче увеличения разрешения рукописного текста справляется хуже. По завершению 10000 эпох обучения точность не успевает стабилизироваться.



*Рис. 7. Ошибка и точности генератора AC-GAN в работе с изображениями рукописных чисел*

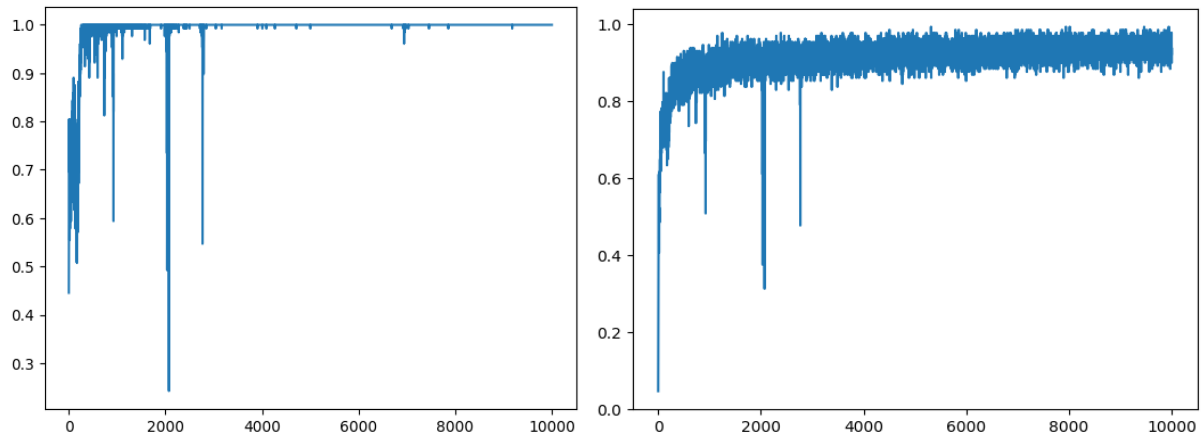
**BiGAN.** Результаты работы с набором данных mnist.fashion следующие: точность BiGAN модели увеличилась в два раза за 200 эпох. По итогу 10000 эпох она стабилизировалась в промежутке значений 2-3%.



*Рис. 8. Ошибка дискриминатора BiGAN в работе с изображениями элементов одежды*



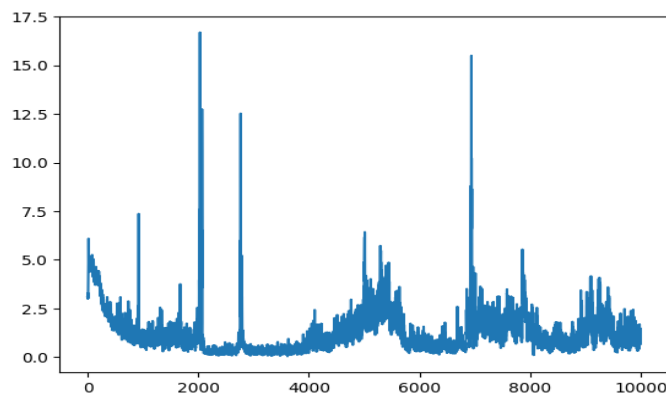
Бинарная точность стабилизировалась в 1 за 3000 эпох, после чего не отклонялась более чем на 0,05.



**Рис. 9. Бинарная точность и точность распознавания дискриминатора BiGAN в работе с изображениями элементов одежды**

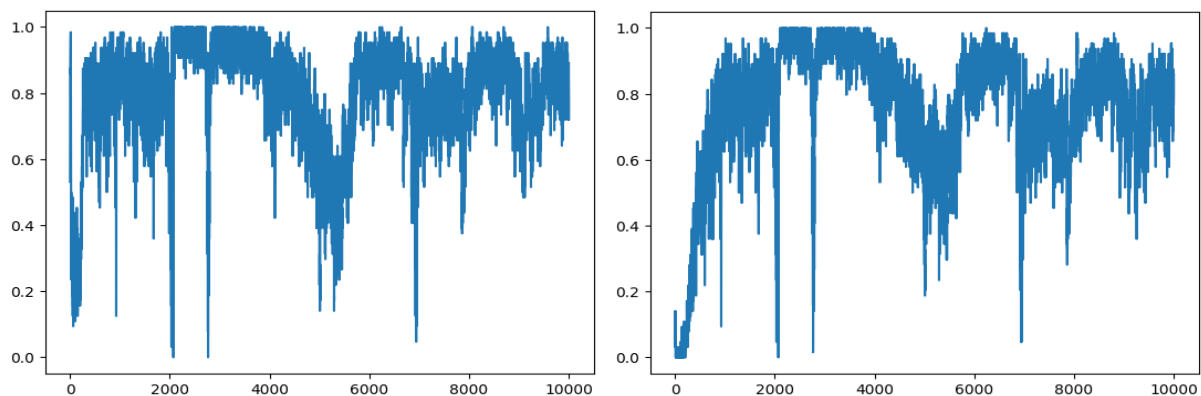
Точность “по категориям” за аналогичное число эпох стабилизировалось на значениях 1-0,85.

Ошибка генератора имеет несколько пиковых значений в 17% и 16%, по истечении обучения стабилизируется в области 5-20%.



**Рис. 10. Ошибка генератора BiGAN в работе с изображениями элементов одежды**

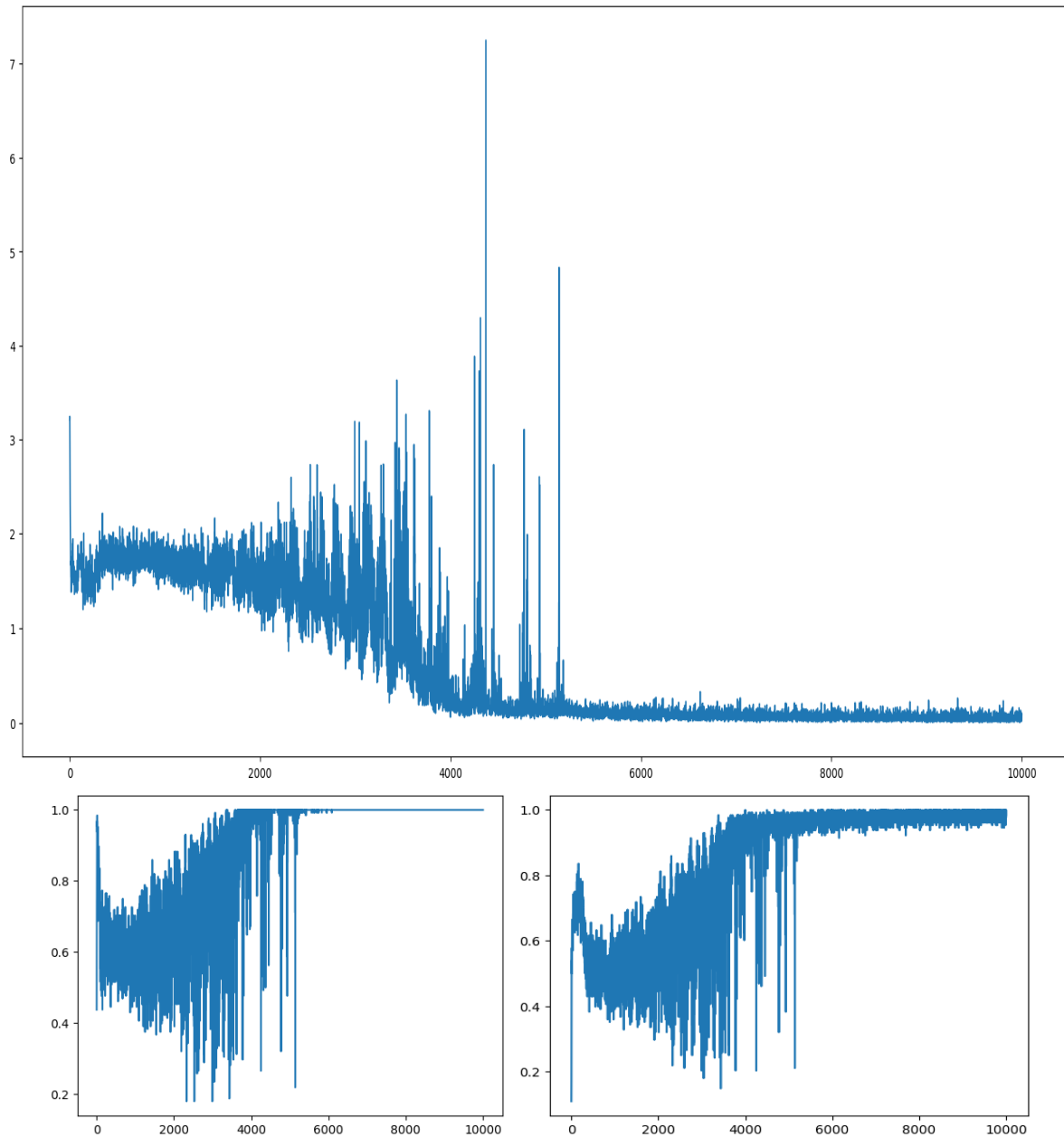
Также, как и с ACGAN, точности генератора имеют схожую траекторию изменения, выходя на значения 0,9-0,7 в обоих случаях.



**Рис. 11. Бинарная точность и точность распознавания генератора BiGAN в работе с изображениями элементов одежды**

Проанализируем поведение параметров сети при использовании базы изображений рукописных чисел.

Ошибки дискриминатора показывают сильный разброс в значениях в первых 5000 эпох обучения, однако также резко стабилизируются во второй половине.

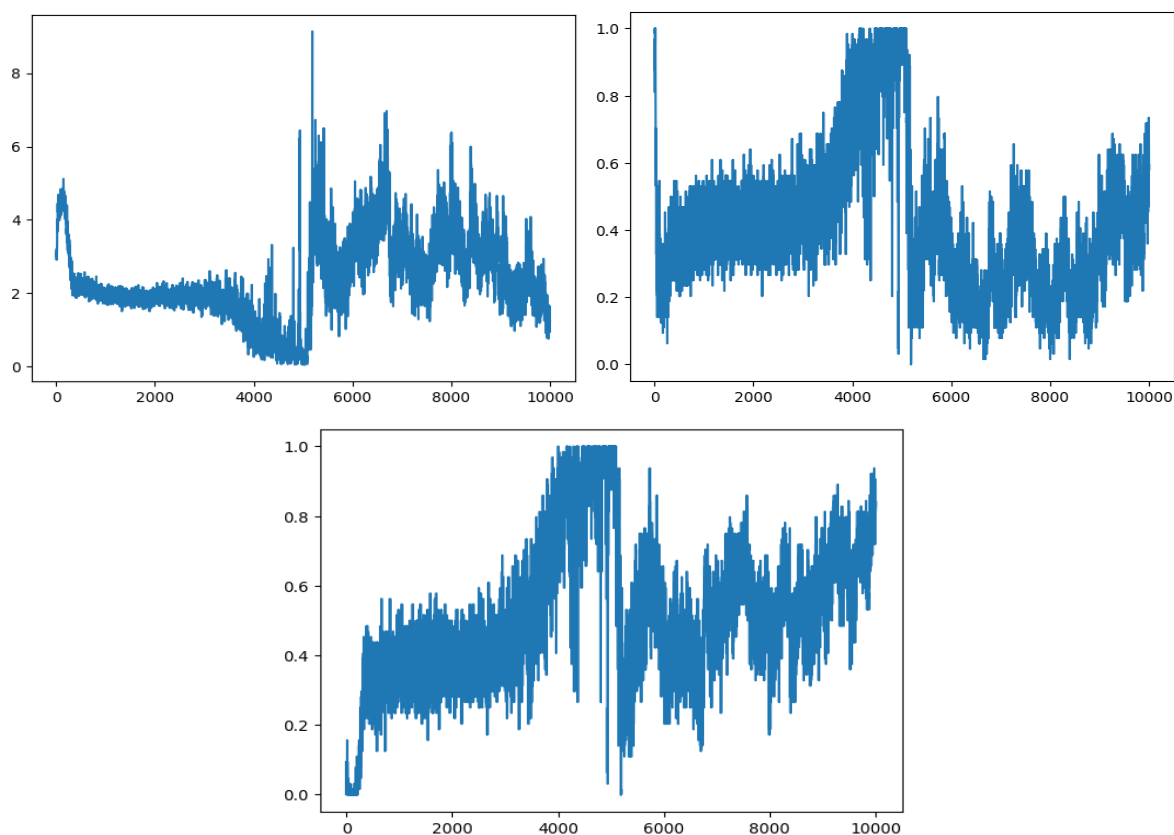


**Рис. 12. Ошибка и точности дискриминатора BiGAN в работе с изображениями рукописных чисел**

BiGAN хуже справляется с улучшением, ошибки генератора не стабилизируются на удовлетворительных значениях.

В ходе работы в качестве объективно измеряемых критериев качества работы GAN-моделей были выбраны ошибка генератора и дискриминатора моделей, а также бинарная и многоклассовая точность генератора и дискриминатора. Для сбора информации и оценки различных моделей нейросетей типа GAN был проведен ряд опытов с различными наборами данных. На основе полученных результатов можно сделать следующие выводы: при работе с фотографиями элементов одежды дискриминатор и генератор ACGAN и BiGAN показали схожие результаты, при чуть большем времени

обучения ACGAN (в среднем 5ч., при 4,5ч. у BIGAN). Однако скорость стабилизации ошибки генератора у модели BIGAN выше, чем у ACGAN, что может быть критично при работе в условиях ограниченных вычислительных и временных ресурсов.



**Рис. 13. Ошибка и точности генератора BiGAN в работе с изображениями рукописных чисел**

**Вывод.** В опыте обработки экземпляров рукописного текста обе нейросетевые модели показали худшие результаты, что обусловлено более строгими различиями между подгруппами фотографий внутри базы данных.

### Список литературы

1. Прохоров А.С. Применение нейронных сетей для обеспечения безопасности человека в жилых и промышленных помещениях. *Международный научный журнал Символ Науки*, 2021, 1. С. 25-29.
2. Zhukovskiy Y.L., Kovalchuk M.S., Batueva D.E., Senchilo N.D. Development of an algorithm for regulating the load schedule of educational institutions based on the forecast of electric consumption within the framework of application of the demand response. *Sustainability*, 2021. 13 (24). 13801.
3. Mlađan J., Mark C. Generative Artificial Intelligence: Trends and Prospects. *IEEE Computer Society*, 2022. P. 107-112.
4. Ian J. Goodfellow; Jean Pouget-Abadie; Mehdi Mirza; Bing Xu. Generative Adversarial Nets. *Advances in Neural Information Processing Systems*, 2014, 3(11).
5. Shakir Mohamed; Balaji Lakshminarayanan. Learning in Implicit Generative Models. 27.02.2017, arXiv:1610.03483v4.
6. Zemenkova M.Y., Chizhevskaya E.L., Zemenkov Y.D. Intelligent monitoring of the condition of hydrocarbon pipeline transport facilities using neural network technologies. *Journal of Mining Institute*, 2022. 258. P. 933-944. DOI: 10.31897/PMI.2022.105.

7. Masatosi U., Issei S., Masahiro S., Kotaro N., Yutaka M. Generative Adversarial Nets from a Density Ratio Estimation Perspective. 2016, arXiv:1610.02920v2.
8. Augustus O., Christopher O., Jonathon S. Conditional Image Synthesis with Auxiliary Classifier GANs. 2017, arXiv:1610.09585v4
9. Jeff D., Trevor D., Philipp K. Adversarial feature learning. 2017, arXiv:1605.09782v7.
10. Zakharov L.A., Martyushev D.A., Ponomareva I.N. Predicting dynamic formation pressure using artificial intelligence methods. *Journal of Mining Institute*, 2022. 253. P. 23-32. DOI: 10.31897/PMI.2022.11.
11. Like H., Mikhail B. Evaluation of neural architectures trained with square loss vs cross-entropy in classification tasks. 2021, arXiv:2006.07322v5.
12. Ming-Yu L., Onel T. Coupled Generative Adversarial Networks. 2016. arXiv:1606.07536v2.
13. Yakovleva T.A., Romashev A.O., Mashevsky G.N. Digital technologies for optimizing the dosing of flotation reagents during flotation of non-ferrous metal ores. *Mining Informational and Analytical Bulletin*, 2022. 6(2), P. 175-188. DOI: 10.25018/0236\_1493\_2022\_62\_0\_175.
14. Shamil I.R., Grigoriev A., Beloglazov I.I., Savchenkov S.A., Gudmestad O.T. Research risk factors in monitoring well drilling — A case study using machine learning methods / *Symmetry*, № 7, V 13, 2021. P. 1 – 19.
15. Sultanbekov R.R., Beloglazov I.I., Islamov S.R., Ong M.C. Exploring of the Incompatibility of Marine Residual Fuel: A Case Study Using Machine Learning Methods. *Energies*, № 8422, V 14, 2021. P. 1 - 16.
16. Ackley, David H., Hinton, Geoffrey E., Sejnowski, Terrence J. A Learning Algorithm for Boltzmann Machines. *Cognitive Science* 9 (1), 1985. P. 147—169.
17. Nurgalieva K.S., Abdullah K.A., Seyed M.A., Slavko N., John William G.G. Application of Neural Network and Time-Domain Feature Extraction Techniques for Determining Volumetric Percentages and the Type of Two Phase Flow Regimes Independent of Scale Layer Thickness *Applied Sciences*. 2022. №12. P. 1-13.
18. Ushakov E., Aleksandrova T., Romashev A. Neural network modeling methods in the analysis of the processing plant's indicators, 2021. DOI:10.1007/978-3-030-57453-6\_4.
19. Filippov E.V., Zakharov L.A., Martyushev D.A., Ponomareva I.N. Reproduction of reservoir pressure by machine learning methods and study of its influence on the cracks formation process in hydraulic fracturing. *Journal of Mining Institute*, 2022. 258. P. 924-932. DOI: 10.31897/PMI.2022.103.
20. Brilliant L.S., Zavialov A.S., Danko M.U., Andronov K.A., Shpurov I.V., Bratkova V.G., Davydov A.V. Integration of machine learning methods and geological and hydrodynamic modeling in field development design. *Neftyanoe Khozyaystvo - Oil Industry*, 2022(3), 48-53. DOI: 10.24887/0028-2448-2022-3-48-53.
21. Romashev A.O., Nikolaeva N.V., Gatiatullin B.L. Adaptive approach formation using machine vision technology to determine the parameters of enrichment products deposition. *Journal of Mining Institute*, 2022. 256. P. 677-685. DOI: 10.31897/PMI.2022.77.
22. Amari S.i. Backpropagation and stochastic gradient descent method. *Neurocomputing* 1993, 5. P. 185–196.
23. Filippov E.V., Zaharov L.A., Martyushev D.A., Ponomareva I.N. Reproduction of reservoir pressure by machine learning methods and study of its influence on the cracks formation process in hydraulic fracturing. *Journal of Mining Institute*, 2022. 258. P. 924–932.
24. Islamov S., Grigoriev A., Beloglazov I., Savchenkov S., Gudmestad O.T. Research Risk Factors in Monitoring Well Drilling—A. Case Study Using Machine Learning Methods. *Symmetry* 2021, 13. DOI: 10.3390/sym13071293.
25. Montiel J., Halford M., Mastelini S.M., Bolmier G., Sourty R., Vaysse R., Zouitine A., Gomes H.M., Read J., Abdessalem T. et al. River: machine learning for stream- ing data in python. *The Journal of Machine Learning Research*, 2021, 22. P. 4945–4952.

26. Vasilev I., Slater D., Spacagna G., Roelants P., Zocca V. Python Deep Learning: Exploring deep learning techniques and neural network architectures with Pytorch, Keras, and TensorFlow; Packt Publishing Ltd, 2019.
27. Kim D., Kim S.H., Kim T., Kang B.B., Lee M., Park W., Ku S., Kim D., Kwon J., Lee H. et al. Review of machine learning methods in soft robotics. Plos one 2021. 16. e0246102.
28. Sun W., Akashi N., Kuniyoshi Y., Nakajima K. Physics-informed recurrent neural networks for soft pneumatic actuators // IEEE Robotics and Automation Letters 2022. 7. P. 6862–6869.
29. Sultanbekov R., Beloglazov I., Islamov S., Ong M.C. Exploring of the Incompatibility of Marine Residual Fuel: A Case Study Using Machine Learning Methods. Energies 2021, 14. DOI: 10.3390/en14248422.
30. Brilliant L., Zavialov A., Danko M., Andronov K., Shpurov I., Bratkova V., Davydov A. Integration of machine learning methods and geological and hydrodynamic modeling in field development design (Russian). Oil Industry Journal, 2022, P. 48–53.
31. Romashev A.O., Nikolaeva N.V., Gatiatullin B.L. Adaptive approach formation using machine vision technology to determine the parameters of enrichment products deposition. Journal of Mining Institute 2022. 256. P. 677–685.
32. Kashyap P., Kashyap P. Industrial applications of machine learning. Machine Learning for Decision Makers: Cognitive Computing Fundamentals for Better Decision Making, 2017. P. 189–233.
33. Educational Institutions Based on the Forecast of Electric Consumption within the Framework of Application of the Demand Response. Sustainability 2021, 13. DOI: 10.3390/su132413801.
34. Takishima Y., Yoshida K., Khosla A., Kawakami M., Furukawa H. Fully 3D-printed hydrogel actuator for jellyfish soft robots. ECS Journal of Solid-State Science and Technology, 2021. 10. 037002.
35. Zakharov L.A., Martyushev D.A., Ponomareva I.N. Predicting dynamic formation pressure using artificial intelligence methods. Journal of Mining Institute, 2022, 253. P. 23–32.
36. Elango N., Faudzi A.A.M. A review article: investigations on soft materials for soft robot manipulations. The International Journal of Advanced Manufacturing Technology 2015, 80. 1027–1037. 393.

*Кожубаев Юрий Нургалеевич, канд. техн. наук, доцент, [y.n.kozhubaev@gmail.com](mailto:y.n.kozhubaev@gmail.com), Россия, Санкт-Петербург, Санкт-Петербургский горный университет,*

*Прокофьев Виктор Александрович, преподаватель, [um-urii@rambler.ru](mailto:um-urii@rambler.ru), Россия, Санкт-Петербург, Санкт-Петербургский политехнический университет Петра Великого*

**MODELING AND CONTROL OF A BIDIRECTIONAL COMPETITIVE NEURAL NETWORK AND A COMPETITIVE NEURAL NETWORK WITH AUXILIARY CLASSIFIER IN THE PROBLEM OF INCREASING RESOLUTION AND INFORMATION PROCESSING**

*Yu.N. Kozhubaev, V.A. Prokofiev*

*The paper considers two different models of GAN (Generative adversarial network - generative adversarial network) neural networks in the tasks of increasing image quality. To evaluate their effectiveness, it is proposed to use such measurable characteristics as the accuracy of the generator and discriminator of GAN models, binary and intergroup errors, and operating time. To measure selected parameters, a series of experiments were conducted*

*using two different data sets: black and white photographs of clothing items and black and white photographs of handwritten text examples, both sets were divided into 10 groups according to the type of object or text contained in the photograph. Generative models allow you to represent the underlying structure of the data. While the goal is to accurately generate data with representative statistics, part of the power of generative models lies in presenting the data as a hypothetical compressed version of the underlying structure. Generative Adversarial Networks is a unique generative learning system that uses two separate models with opposing, competing or adversarial goals. Neural network-based autoencoders can be used to train a denoising filter based on the dataset from which the denoising is to be removed.*

*Key words: competitive neural networks, deep learning, resolution enhancement, image processing.*

*Kozhubaev Yury Nurgalievich, candidate of technical sciences, docent, [y.n.kozhubaev@gmail.com](mailto:y.n.kozhubaev@gmail.com), Russia, St. Petersburg, St. Petersburg Mining University,*

*Prokofiev Viktor Alexandrovich, lecturer, [um-urii@rambler.ru](mailto:um-urii@rambler.ru), Russia, St. Petersburg, Peter the Great St. Petersburg Polytechnic University*

УДК 621

DOI: 10.24412/2071-6168-2023-8-101-102

## **АНАЛИТИЧЕСКОЕ СРАВНЕНИЕ ПРОЦЕССОВ ПРИ ИСПОЛЬЗОВАНИИ РАЗНОГО ТЕХНОЛОГИЧЕСКОГО ОБОРУДОВАНИЯ**

О.Г. Ветков

*Гидравлические и кривошипные прессы являются наиболее распространенными типами штамповочного оборудования, используемыми в промышленных масштабах. Оба типа прессов предназначены для осуществления деформирования материала с использованием силы, и оба обладают своими характеристиками и преимуществами. В статье рассматриваются основные принципы работы прессов, их преимущества и недостатки, а также технологические инновации, вносимые в эти машины для повышения эффективности производства. Гидравлические и кривошипные прессы, несмотря на свою популярность, имеют ряд недостатков, которые могут ограничивать их эффективность и производительность, поэтому в данной работе ставится акцент на сравнительном анализе преимуществ и недостатков этих видов оборудования. Также описываются основные области применения гидравлических и кривошипных прессов. Полученные результаты могут быть полезны для инженеров и специалистов, занимающихся выбором и эксплуатацией прессов. Делаются выводы о том, какое оборудование следует использовать при производстве определенных видов изделий.*

*Ключевые слова: прессовое оборудование, штамповка, технологическое оборудование, промышленность, преимущества и недостатки, изделия, детали.*

Производственная промышленность характеризуется большим числом промышленного оборудования для создания необходимой продукции. В машиностроении каждый вид оборудования предназначен для получения определённых видов изделий, которые выполняются из разных материалов, однако наиболее распространенным материалом в данной отрасли является металл [1]. Изделия из металла могут выполняться разными заготовительными методами (см. рисунок), в числе которых можно выделить сварку, литье, порошковую металлургию, обработку давлением [2]. Последняя в данном списке на данный момент является одной из наиболее применяемых, и при эксплуатации технологии применяется большое разнообразие оборудования. Наиболее популярным являются прессы, а именно гидравлические и кривошипные [3-10], о которых и пойдет речь в данной статье.