

Санкт-Петербургский государственный университет

Математическое обеспечение и администрирование информационных  
систем

Системного программирования

Кутуков Сергей Евгеньевич

Поиск аномалий на рентгенограммах  
грудной клетки

Бакалаврская работа

Научный руководитель:  
к.ф.-м.н ст. преп. С. И. Салищев

Рецензент:  
к.т.н. доцент А. Л. Ронжин

Санкт-Петербург  
2018

Saint-Petersburg state university

Software and Administration of Information Systems  
Software System Programming

Kutukov Sergey Evgenevich

# Anomaly detection on Chest X-Ray

Bachelor's Thesis

Scientific supervisor:  
Senior lecturer Sergey Salishchev

Reviewer:  
Assoc. prof. Alexander Ronzhin

Saint-Petersburg  
2018

# Оглавление

<b>Введение</b>	<b>4</b>
<b>Постановка задачи</b>	<b>6</b>
<b>1. Обзор</b>	<b>7</b>
1.1. Нейросеть ChestXNet . . . . .	7
1.2. Генеративные модели . . . . .	8
1.2.1. Автоэнкодеры . . . . .	9
1.2.2. Генеративно-состязательные нейросети . . . . .	10
1.3. Алгоритмы классификации . . . . .	11
1.3.1. Случайный лес . . . . .	12
1.3.2. Машина опорных векторов . . . . .	12
1.4. Метод оценки точности . . . . .	12
1.4.1. Метрика AUC-ROC . . . . .	13
1.4.2. Метрика AUC-PR . . . . .	14
<b>2. Подготовка данных для машинного обучения</b>	<b>15</b>
<b>3. Разработанные модели машинного обучения</b>	<b>16</b>
3.1. Автоэнкодер . . . . .	16
3.2. Генеративно-состязательная нейросеть . . . . .	16
3.3. Классификация . . . . .	19
<b>4. Эксперименты</b>	<b>20</b>
4.1. Автоэнкодер . . . . .	20
4.2. Генеративно-состязательная нейросеть . . . . .	20
4.3. Выводы . . . . .	21
<b>Заключение</b>	<b>24</b>
<b>Список литературы</b>	<b>25</b>

# Введение

При медицинском исследовании легких с помощью рентгенограмм грудной клетки можно выделить примерно 14 видов патологий таких как рак, пневмоторакс, пневмония, фиброз, эмфизема и туберкулез.

Главным заболеванием, из-за которого люди проходят регулярные флюорографические исследования, является туберкулез. Он является весьма заразным и опасным заболеванием, поэтому проведение рентгенологического исследования грудной клетки, является обязательной и ежегодной процедурой для каждого человека в России.

Из-за массовости этой процедуры многие миллионы рентгенограмм должны быть проанализированы, что является огромной нагрузкой для радиологов. Более того, чтобы снизить риски человеческого фактора, рентгенограмма проверяется минимум двумя специалистами или одним и тем же специалистом, но два раза с интервалом минимум один день. Анализ рентгенограмм — это сложная работы над однородными изображениями, и даже опытным радиологам сложно качественно обработать более чем 100 рентгенограмм в день. Также по данным СПб НИИФ причиной 21% поздних стадий туберкулеза является несвоевременное обнаружение на флюорографии. Поэтому существует заинтересованность в разработке компьютерных алгоритмов для поиска аномалий на рентгенограмме, которые впоследствии станут вспомогательным инструментом в помощь радиологам.

Одним из методов машинного обучения являются нейронные сети. Нейронные сети являются универсальными системами для интерполяции функций. Нейронная сеть с оптимальной архитектурой способна аппроксимировать функцию выделения патологий и переложить задачу формализации признаков патологий с человека на нейронную сеть, что упрощает классификацию снимков грудной клетки.

Новизна данной работы состоит в применении автоэнкодеров и генеративно-сопоставительных нейросетей для классификации медицинских данных. В работе проверялось две гипотезы. Первая состояла в том, что в скрытом пространстве автоэнкодера можно разделить здо-

ровые и нездоровые снимки, а вторая — в том, что если обучить генеративную нейронную сеть на здоровых снимках, то патологии можно искать как разницу между входным и выходным снимком.

Работа состоит из обзора, трех глав и заключения.

В обзорной части (1) описываются какие подходы уже применялись для обнаружения патологий в легких, подробно обзревается нейронная сеть ChexNet, описываются основные концепции генеративных методов машинного обучения, в том числе подробно описывается модели автоэнкодера и генеративно-состязательной нейронной сети. Описываются алгоритмы случайный лес и машина опорных векторов, использованные для классификации скрытого представления снимков на здоровые и нездоровые, и метрики AUC-ROC и AUC-PR, использованные для оценки качества алгоритмов.

В первой главе (2) описываются использованные данные: удалось отобрать 27320 здоровых снимков и 13863 снимков с аномалиями из датасета NIH Chest X-ray.

Во второй главе (3) подробно описывается архитектура автоэнкодера и генеративно-состязательной нейронной сети: применяли последовательно сверточные слои, слои нормализации и активационные слои ReLU и рассчитывали ошибки моделей как среднюю квадратическую ошибку. Также для классификации применялся случайный лес с 1000 решающими деревьями и машина опорных векторов с радиальной базисной функцией в качестве ядра.

В третьей главе (4) описывались эксперименты: выборка разделялась в пропорциях 85% и 15% для обучения и тестирования соответственно, применялся метод оптимизации Адама со скоростью обучения  $1e-3$ . Приводятся результаты и делается вывод, что первая гипотеза жизнеспособна и получена наилучшая точность в 0.65, а подтвердить вторую не удалось.

# Постановка задачи

Целью данной работы является применение методов машинного обучения в области поиска патологий на рентгенограммах легких и проведение соответствующих исследований. Для достижения этой цели были поставлены следующие задачи.

1. Провести подготовку данных для машинного обучения: отобрать снимки, подходящие для применяемых методов, и разработать методы предварительной обработки снимков, облегчающие процесс обучения.
2. Разработать модели машинного обучения: построить наиболее оптимальную архитектуру моделей и реализовать их.
3. Провести эксперименты с реализованными моделями машинного обучения: обучить модели, выбрать метрику для оценки эффективности моделей, протестировать модели и сравнить результаты работы моделей, используя выбранную метрику.

# 1. Обзор

Несколько лет уже проводятся соревнования по поиску аномалий на компьютерной томографии легких, такие как Luna16 [1] и Data Science Bowl 2017 [2].

Применение компьютерной томографии более точный метод, чем флюорографическое исследование, но гораздо более сложный, что осложняет его применение также массово.

После публикации датасета NIH Chest X Ray появились работы по применению методов машинного обучения для решения задачи по поиску патологий именно на флюорографиях легких. Работу по поиску аномалий вели несколько групп [3] [4]. Одной из самых успешных работ является нейросеть ChestXNet[5] от Stanford ML Group.

Также для улучшения результатов поиска аномалий на снимках легких в условиях нехватки данных для машинного обучения успешно применялась аугментация с помощью генеративно-сопоставительных нейросетей [6].

## 1.1. Нейросеть ChestXNet

Нейронная сеть ChestXNet принимает на вход снимок, состоит из 121 сверточного слоя, схожими по структуре с нейронной сетью VGG16 [7] и по итогам своей работы выдает вероятность патологии на снимке.

В рамках исследования проводилось несколько серий обучения. В одной серии сравнивался результат с экспертной оценкой нескольких врачей, а во втором сравнивали со связанными работами.

В рамках первого обучения патологии были разделены на 4 группы, датасет был разделен в пропорциях 93% на обучение, 6% на валидацию, и оставшиеся 1% на тестирование, что соответствует 28744, 1672 и 389 пациентам или 98637, 6351 и 420 снимкам. Для подсчета точности модели применялась F1-мера, и была получена точность 0.435.

В рамках второго обучения патологии разделили на 14 групп, а выборку в пропорциях 70%, 10% и 20%, на обучение, валидацию и тестирование соответственно. Для оценки подсчитывалось AUC-ROC и была

получена средняя точность для разных видов патологий 0.8 .

Для наглядной интерпретации извлекались карты особых точек, которые выводились последним сверточным слоем. Далее извлекали особые точки наиболее характерные для классификации патологии, масштабировали их до размера исходного изображения и накладывали их на него.

## 1.2. Генеративные модели

Классический способ классификации с помощью машинного обучения заключается в разметке датасета и применении нейросети, способной возвращать вероятность принадлежности к тому или иному классу. В рамках области обработки медицинских данных такой подход требует большого труда медицинских экспертов на этапе подготовки данных и интерпретация результатов работы классификатора требует дополнительной работы.

Альтернативный подход заключается в применении генеративных нейронных сетей с целью обучить их генерировать снимки без патологий. Для выделения патологий входной снимок обрабатывают нейросетью, на выходе получают снимок без патологии, а патологии ищут как разность исходного и выходного снимка.

Для применения этого подхода требуется отличная от первого подготовка данных. В отличие от первого подхода требуются не снимки с патологиями, на которых отмечены области с патологиями явно, а наоборот здоровые снимки без аномалий, что снижает нагрузку на врачей при разметке данных. Также применение генеративных нейронных сетей упрощает интерпретацию результата работы нейросети. Результатом работы нейронной сети является снимок размерности входного изображения, который и является наглядной интерпретацией её работы.

### 1.2.1. Автоэнкодеры

Автоэнкодер [8] — это комбинация двух нейросетей (рис. 1): кодирующей и декодирующей. Одной на вход подается изображение, и она учится отображать из пространства исходного изображения на скрытое пространство. Второй на вход подается сущность из скрытого пространства, и она учится отображать её в пространство исходного изображения. Так как размерность скрытого пространства меньше, чем пространство исходного изображения, нейросеть отбирает только важные признаки. Таким образом, обученная только на снимках без патологий, нейросеть не будет способна генерировать снимки с аномалиями.

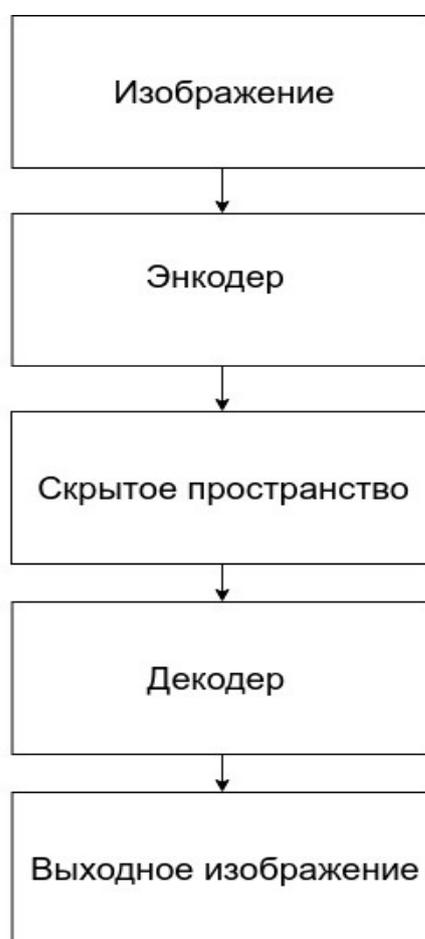


Рис. 1: Схема автоэнкодера.

### 1.2.2. Генеративно-состязательные нейросети

Генеративно-состязательная нейросеть (GAN) [9] - это нейронная сеть (рис. 2), включающая в себя генерирующую и дискриминирующую нейросеть. При обучении такой системы генерирующая и дискриминирующая нейросеть вступают в состязательную игру. В рамках этой игры дискриминирующая нейросеть анализирует настоящие и сгенерированные снимки и стремится научиться отличать настоящие снимки от сгенерированных, а генерирующая сеть анализирует настоящие снимки и стремится научиться генерировать снимки, которые дискриминирующая сеть не сможет отличить от настоящих.

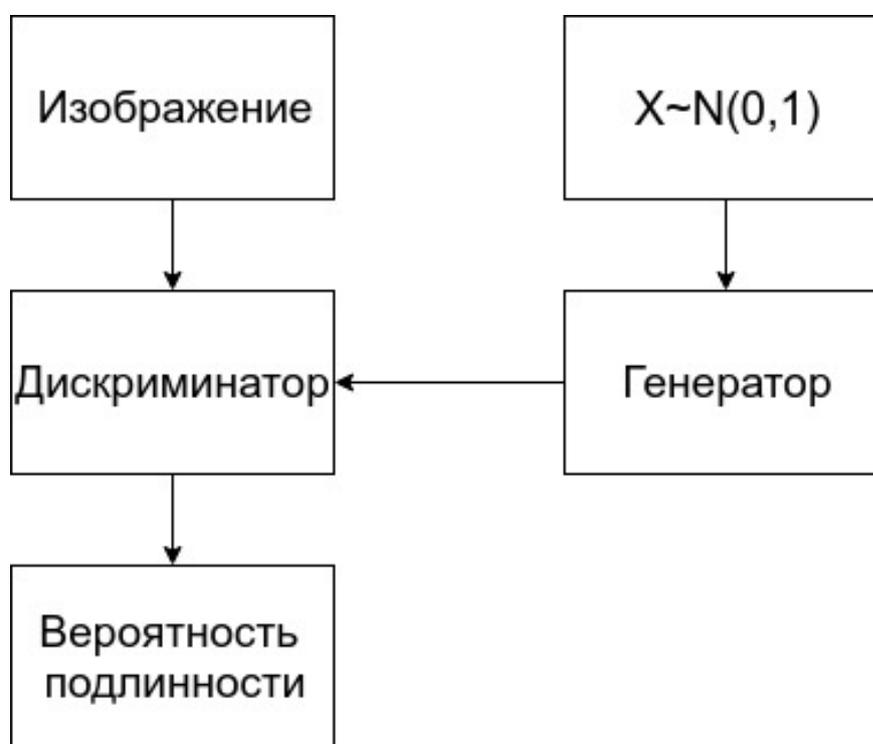


Рис. 2: Схема генеративно-состязательной нейронной сети.

Можно комбинировать генеративно-состязательную сеть с вариационным автоэнкодером (рис. 3). В таком случае нейросеть-генератор представляет сверточный вариационный автоэнкодер, который отличается от обычного тем, что на слое, предшествующему скрытому пространству пытается предсказать не конкретное значение, а среднее значение и логарифм вариации. Также добавляется промежуточный отборочный слой, который с помощью среднего значения и логарифма вари-

ации, корректирует значение, полученное из стандартного нормального распределения.

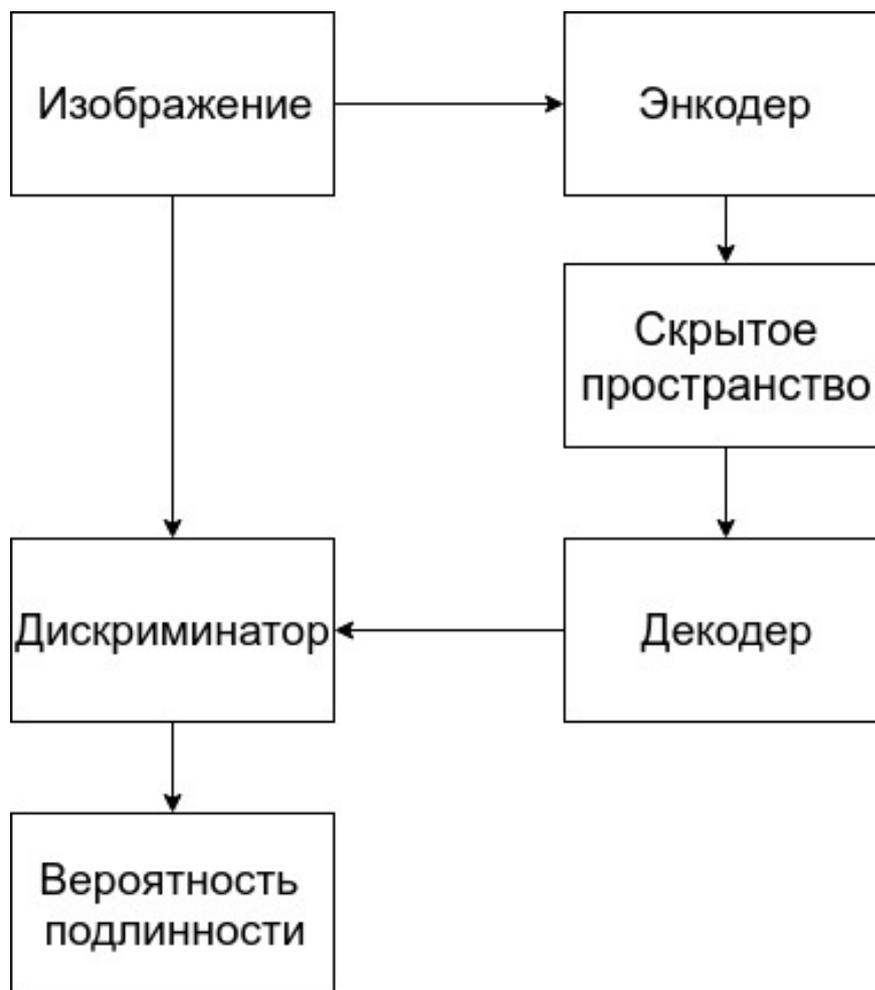


Рис. 3: Схема комбинации генеративно-состязательной нейронной сети и автоэнкодера.

### 1.3. Алгоритмы классификации

Промежуточное скрытое представление изображения можно воспринимать как вектор признаков, который автоэнкодер выделил в процессе обработки. Полученные вектора можно классифицировать различными методами машинного обучения.

### 1.3.1. Случайный лес

Первый примененный алгоритм классификации, это случайный лес[10]. Он состоит в использовании ансамбля решающих деревьев, выдающих вероятность принадлежности снимка к классу. Окончательное решение принимается по результатам голосованием, в котором участвуют все деревья.

### 1.3.2. Машина опорных векторов

Второй примененный алгоритм машинного обучения — это машина опорных векторов (SVM) [11]. Он состоит в построении гиперплоскости между двумя классами и при необходимости замены стандартного линейного ядра на нелинейное. Решение принимается в зависимости от того по какую сторону от гиперплоскости находится объект.

Формально имеется набор векторов вида  $X = \{x_1, \dots, x_m\}$ , содержащие некоторые признаки, и вектор  $C = \{c_1, \dots, c_n\}$ , содержащий метки принадлежности к одному из двух классов каждого из векторов  $X$ . Требуется построить гиперплоскость вида  $W * X + b = 0$ , где  $W$  — перпендикуляр к гиперплоскости,  $b$  — смещение гиперплоскости.

В случае линейно неразделимой выборки для поиска оптимальной гиперплоскости требуется минимизировать следующий функционал:

$$\begin{cases} \frac{1}{2} \|W\|^2 + C \sum_1^n \xi_i \rightarrow \min \\ 1 - \xi_i \leq c_i(wX_i - b), 1 \leq i \leq n \\ 0 \leq \xi_i, 1 \leq i \leq n \end{cases} \quad ,$$

где  $C$  характеризует величину ширину разделяющей полосы.

## 1.4. Метод оценки точности

Для оценки работы использовалась тестовая часть датасета и снимки с патологиями. В качестве метрики для оценки качества работы модели применялись площади под ROC-кривой (AUR-ROC) и PR-кривой

(AUR-PR). Для расчета AUR-ROC и AUR-PR необходимо вычислить:

1. TP — количество снимков с патологией, для которых модель выдает оценку, что патология есть.
2. FP — количество снимков без патологий, для которых модель выдает оценку, что патология есть.
3. TN — количество снимков без патологий, для которых модель выдает оценку, что патологии нет.
4. FN — количество снимков с патологией, для которых модель выдает оценку, что патологии нет.

#### 1.4.1. Метрика AUC-ROC

Для расчета метрики AUC-ROC требуется вычислить TPR, что соответствует способности модели распознавать снимки с аномалией, и FPR, что соответствует способности модели не принимать здоровый снимок за нездоровый.

$$TPR = \frac{TP}{TP + FN},$$

$$FPR = \frac{FP}{TN + FP},$$

Обычно модели возвращают не точную принадлежность к классу, а вероятность. Изменяя порог вероятности  $w_0$ , согласно которому модель относит снимок к нездоровым, можно балансировать TPR и FPR. Если порог равен  $-\infty$ , то все снимки модель определила как нездоровые, что соответствует  $TPR = 1, FPR = 1$ . Если порог равен  $+\infty$ , то все снимки модель определила как здоровые, что соответствует  $TPR = 0, FPR = 0$ .

ROC-кривая показывает зависимость между TPR и FPR от  $w_0$  и проходит из точки (0,0) в точку (1,1). AUR-ROC есть площадь под ROC-кривой. Значению AUR-ROC 0.5 соответствует случайное угадывание, а 1 - идеальная классификация.

### 1.4.2. Метрика AUC-PR

Для расчета метрики AUC-ROC требуется вычислить Precision, что соответствует точности модели, и Recall, что соответствует полноте модели.

$$\text{Precision} = \frac{TP}{TP + FP},$$

$$\text{Recall} = \frac{TP}{TP + FN},$$

Аналогично ROC-кривой, можно варьировать  $w_0$  и балансировать Precision и Recall, чтобы построить PR-кривую и находить AUC-PR (площадь под PR-кривой). Идеальной классификации соответствует AUC-PR равное 1.

## 2. Подготовка данных для машинного обучения

Для обучения использовались датасет NIH Chest X-ray [3].

В датасете NIH Chest X-ray на части снимков присутствуют медицинские приборы на легких и вспомогательные медицинские надписи. Для генеративных методов машинного обучения такие артефакты могут создавать шум при обучении. Медицинские приборы могут иметь совершенно разную форму, поэтому вручную были отобраны снимки без артефактов. Всего удалось отобрать 27320 из 63 337 здоровых снимков и 13863 из 48783 снимков с патологией.

Для ускорения обучения и уменьшения требуемой памяти для хранения модели изображения были уменьшены до размера  $256 \times 256$ .

## 3. Разработанные модели машинного обучения

В рамках данной работы разработаны две модели.

1. Модель сверточного автоэнкодера [8].
2. Комбинированная модель, включающая в себя вариационный автоэнкодер [12] и генеративно-сопоставительную нейросеть [9].

Обе модели были реализованы на языке программирования Python [13] с помощью фреймворка PyTorch [14].

### 3.1. Автоэнкодер

Архитектура кодирующей нейронной сети (рис. 4), состоит из шести расположенных последовательно друг за другом групп слоев, включающих в себя сверточный слой с ядром размера 4, слой нормализации для ускорения обучения и активационный слой Leaky ReLU и завершается сигмоидной. Размерность скрытого пространства равна 2048.

Архитектура декодирующей нейронной сети (рис. 5), устроена схожим образом и состоит из четырех групп слоев, состоящих из обратного сверточного слоя с ядром размера 4, слоя нормализации и активационного слоя ReLU и завершается гиперболическим тангенсом.

Ошибка работы автоэнкодера рассчитывается как средняя квадратическая ошибка между исходным изображением и обработанным нейросетью изображением.

Для интерпретации результатов работы автоэнкодера достаточно просто обработать им некоторый снимок грудной клетки.

### 3.2. Генеративно-сопоставительная нейросеть

Нейросеть-генератор представляет из себя вариационный автоэнкодер, основанный на вышеописанном автоэнкодере.

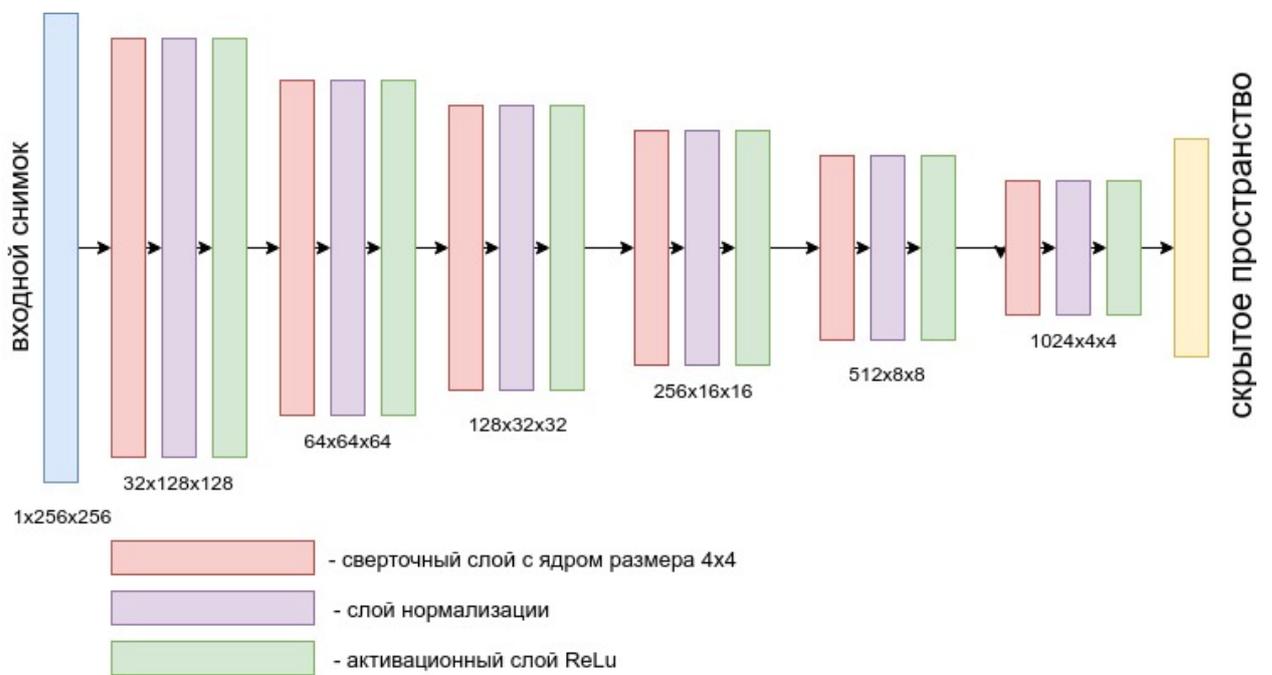


Рис. 4: Архитектура энкодера.

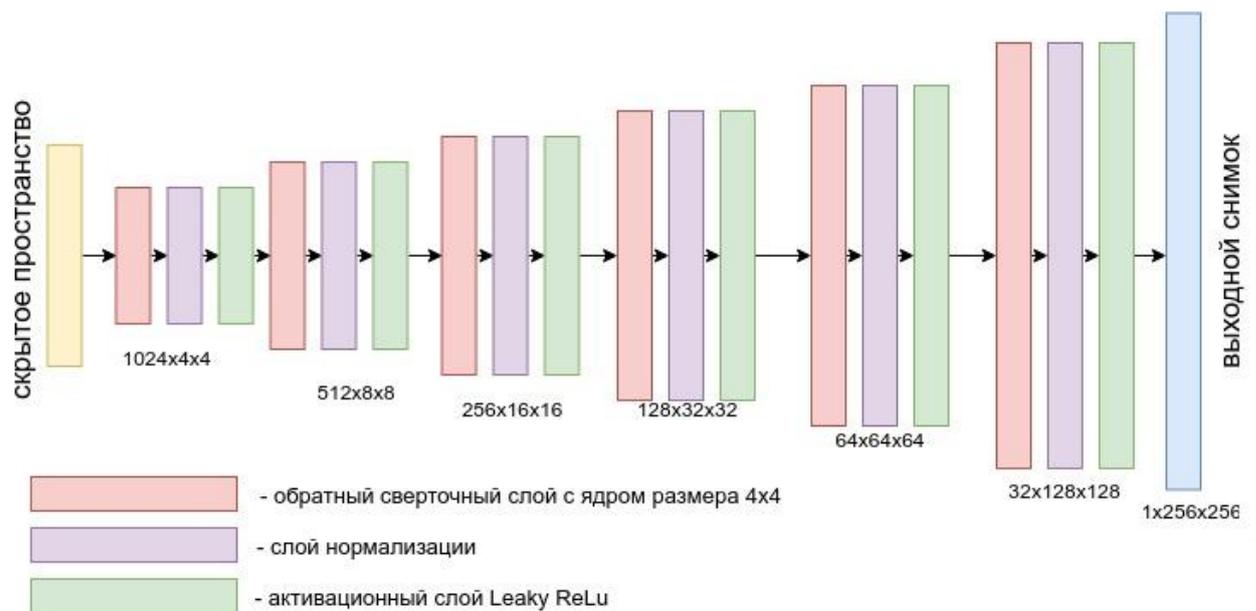


Рис. 5: Архитектура декодера.

Архитектура нейросети-дискриминатора (рис. 6) практически совпадает с кодирующей нейросетью у автоэнкодера с точностью до того, что заключительные слои возвращают не сущность из скрытого пространства, а вероятность, что снимок без патологии.

Ошибка вариационного автоэнкодера считается как среднее квадратичное отклонение между исходным изображением и выходным изоб-

ражением вариационного автоэнкодера.

Ошибка генератора - как перекрестная энтропия[15] между единичным вектором и значениями, которые дал дискриминатор сгенерированным изображениям, что соответствует тому насколько достоверно генератор формирует изображения.

Ошибка дискриминатора - как сумма перекрестной энтропии между нулевым вектором и результатом работы дискриминатора по отношению к сгенерированным изображениям, что соответствует способности дискриминатора отличать сгенерированные изображения от настоящих, и перекрестной энтропии между единичным вектором и результатом работы дискриминатора по отношению к настоящим изображениям, что соответствует способности дискриминатора отличать настоящие изображения от сгенерированных.

Результатом работы генеративно-состязательной нейросети является как вероятность отсутствия на снимке патологий, так и результат обработки входного снимка вариационным автоэнкодером.

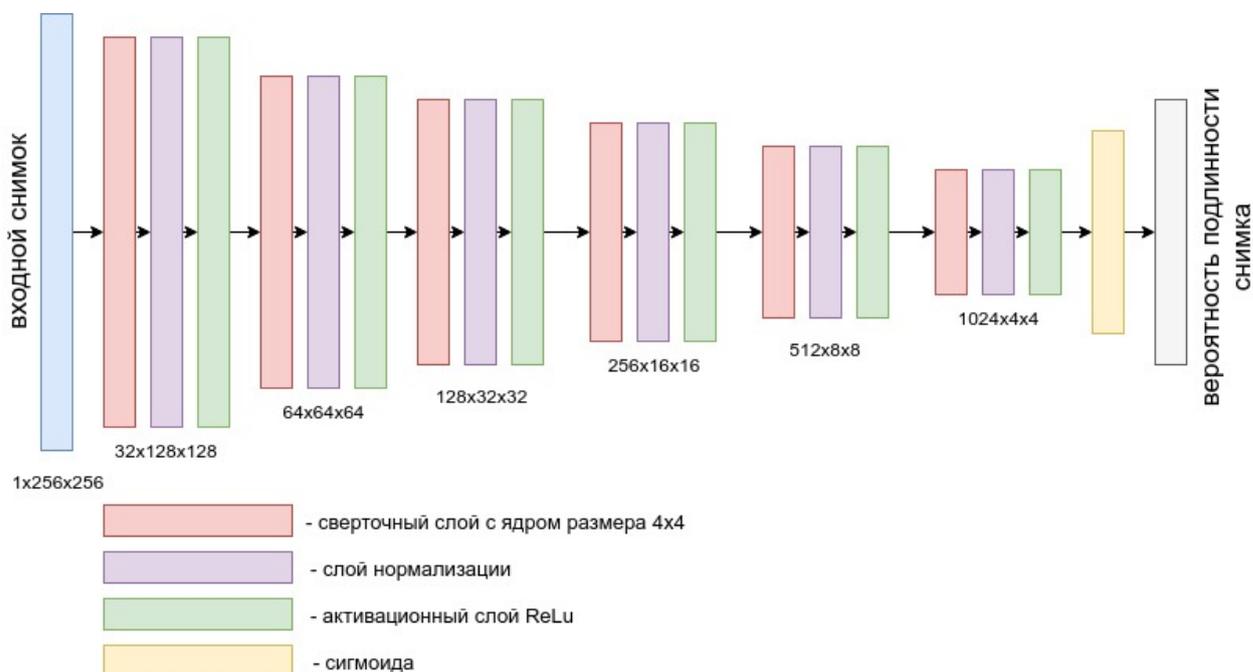


Рис. 6: Архитектура дискриминатора.

### 3.3. Классификация

Для классификации снимков использовалось их скрытое представление в автоэнкодере или генеративной сети и применялись алгоритмы случайный лес и машина опорных векторов. Оба алгоритма написаны с помощью языка Python[13] и библиотеки scikit-learn[16].

При работе с машиной опорных векторов применялась радиальная базисная функция как ядро с параметрами  $\gamma = 0.01$  и  $C = 10$ . Случайный лес состоял из 1000 деревьев.

Для классификации снимков после обработки генеративно-состязательной нейронной сети также подсчитывалась суммарная абсолютная разница пикселей между входным изображением и обработанным, выбирался порог, при превышении которого модель оценивала снимок как нездоровый, и рассчитывались AUR-ROC и AUR-PR.

## 4. Эксперименты

Для обучения автоэнкодера и генеративно-состязательной нейронной сети снимки без патологий были разделены случайным образом в пропорциях 85 % и 15% на обучение и тестирование, что соответствует 23 220 и 4098 снимкам. При обучении не использовались снимки с патологиями, и все 13863 снимков применялись для тестирования.

Для классификации снимки обрабатывались нейронной сетью, и сохранялось скрытое представление снимков. Снимки с аномалиями были разделены в соотношении 70% и 30% на обучения и тестирования, что соответствует 9765 и 4098 снимкам. Количество снимков с аномалиями и без аномалий для тестирования одинаково, так как для расчета метрик AUR-ROC и AUC-PR требуется сбалансировать классы.

### 4.1. Автоэнкодер

При обучении применялся метод оптимизации Адама [17] со скоростью обучения  $1e-3$  и было проведено 200 итераций обучения. Обучение проводилось на NVIDIA 1050 TI и заняло 3 дня.

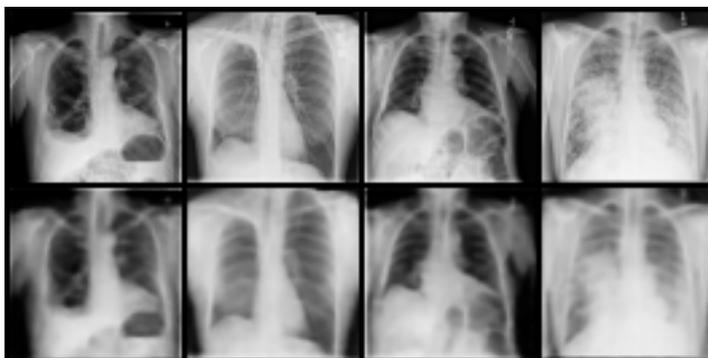


Рис. 7: Результаты работы автоэнкодера. Сверху исходные изображения. Снизу обработанные.

### 4.2. Генеративно-состязательная нейросеть

При обучении как и в предыдущем модели применялась оптимизация Адама [17] с о скоростью сходимости  $1e-3$ . Было проведено 200

Таблица 1: Точность классификации в скрытом пространстве.

Модели	AUC-ROC		AUC-PR	
	SVM	Случайный лес	SVM	Случайный лес
Автоэнкодер	0.58	0.61	0.59	0.65
GAN	0.51	0.52	0.49	0.52

Таблица 2: Точность классификации как суммарная разность пикселей.

Модель	AUR-ROC	AUR-PR
GAN	0.51	0.52

итераций обучения. Обучение проводилось на NVIDIA 1050 TI и заняло 2 дня.

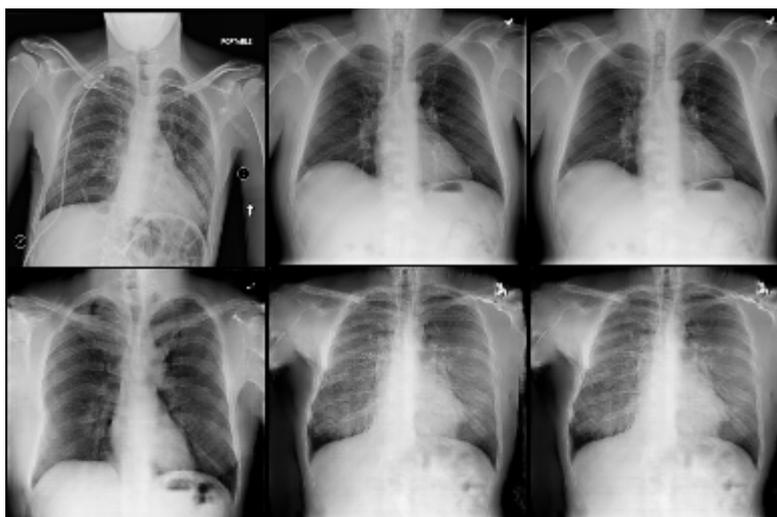


Рис. 8: Результаты работы генеративно-состязательной нейронной сети. Сверху исходные изображения. Снизу обработанные.

### 4.3. Выводы

Результаты работы моделей приведены в таблице 1 и 2. Применение генеративных нейронных сетей, таких как автоэнкодеры и генеративно-состязательные нейронные сети (GAN), не дало таких же хороших результатов, как у традиционных для классификации, глубоких, сверточ-

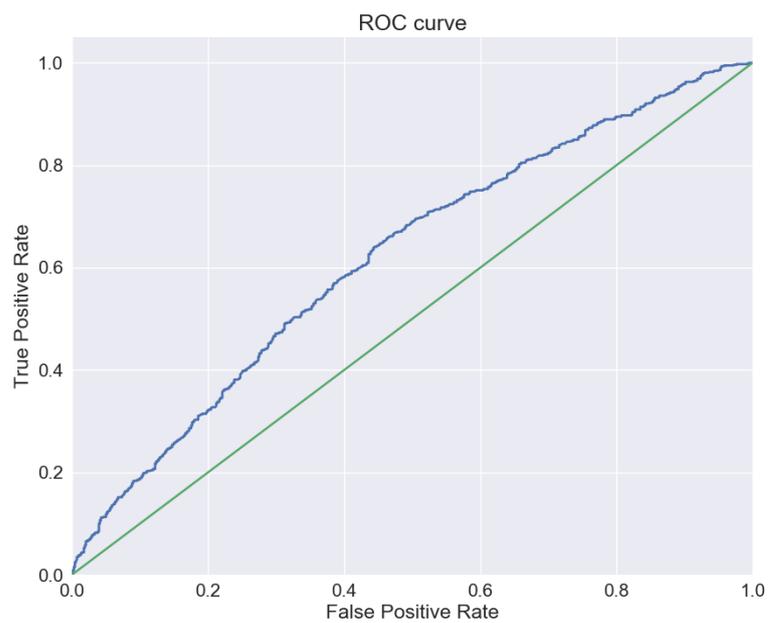


Рис. 9: ROC-кривая результатов работы случайного леса у автоэнкодера.

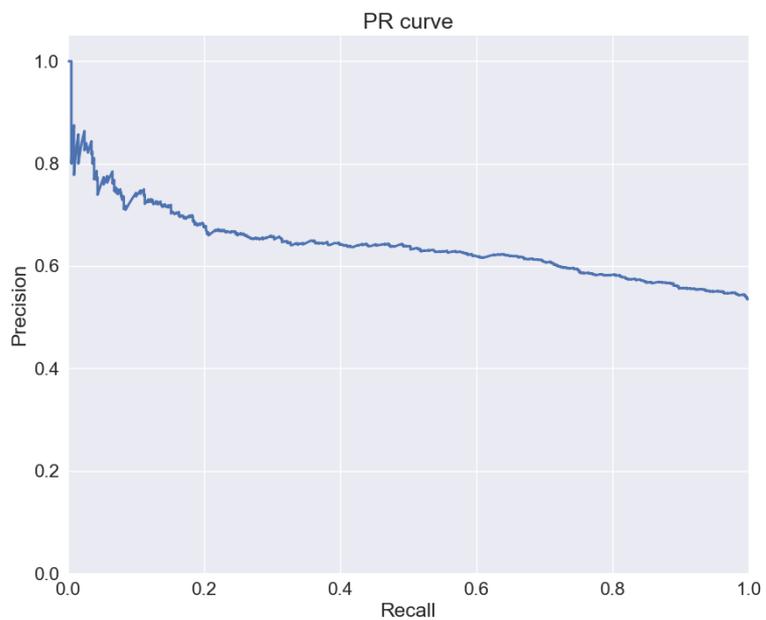


Рис. 10: PR-кривая результатов работы случайного леса у автоэнкодера.

Таблица 3: Сравнение с нейронной сетью ChestXNet.

Модель	AUR-ROC
Автоэнкодер	0.61
GAN	0.52
ChestXNet	0.84

ных нейронных сетей, таких как ChestXNet. Сравнение приведено в таблице 3. Отрицательно на результаты могло повлиять то, что в обучении не участвовали снимки с аномалиями, и итоге в скрытом пространстве они смешались с здоровыми снимками. Можно сделать вывод, что либо метод не подходит для классификации медицинских снимков, либо применение датасета NIH Chest X-ray для обучения генеративных моделей вызвало сторонние эффекты, отрицательно сказавшиеся на результатах.

## Заключение

В рамках данной выпускной квалификационной работы были достигнуты следующие результаты.

1. Проведена подготовка данных для машинного обучения: из датасета NIH Chest X Ray были отобраны снимки без патологий подходящие для обучения.
2. Разработана модель автоэнкодера и комбинированная модель, включающая в себя генеративно-сопоставительную нейросеть и вариационный автоэнкодер. Реализованы модели на языке программирования python с помощью фреймворка Pytorch.
3. Проведены эксперименты с реализованными моделями: проведено обучение на обработанных данных, в качестве метрики для оценки эффективности моделей выбраны метрики AUR-ROC и AUC-PR, модели протестированы на тестовой части данных и получена лучшая точность классификации 0.65.

## Список литературы

- [1] Luna16. — 2016. — Access mode: <https://luna16.grand-challenge.org/>.
- [2] Data science bowl 2017. — 2017. — Access mode: <https://www.kaggle.com/c/data-science-bowl-2017>.
- [3] ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases / Xiaosong Wang, Yifan Peng, Le Lu et al. // CoRR. — 2017. — Vol. abs/1705.02315. — 1705.02315.
- [4] Learning to diagnose from scratch by exploiting dependencies among labels / Li Yao, Eric Poblenz, Dmitry Dagunts et al. // CoRR. — 2017. — Vol. abs/1710.10501. — 1710.10501.
- [5] CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning / Pranav Rajpurkar, Jeremy Irvin, Kaylie Zhu et al. // CoRR. — 2017. — Vol. abs/1711.05225. — 1711.05225.
- [6] Generalization of Deep Neural Networks for Chest Pathology Classification in X-Rays Using Generative Adversarial Networks / Hojjat Salehinejad, Shahrokh Valaee, Tim Dowdell et al. // CoRR. — 2017. — Vol. abs/1712.01636. — 1712.01636.
- [7] Simonyan Karen, Zisserman Andrew. Very Deep Convolutional Networks for Large-Scale Image Recognition // CoRR. — 2014. — Vol. abs/1409.1556. — 1409.1556.
- [8] Autoencoder for words / Cheng-Yuan Liou, Wei-Chen Cheng, Jiun-Wei Liou, Daw-Ran Liou // Neurocomputing. — 2014. — Vol. 139. — P. 84 – 96. — Access mode: <http://www.sciencedirect.com/science/article/pii/S0925231214003658>.
- [9] Radford Alec, Metz Luke, Chintala Soumith. Unsupervised Representation Learning with Deep Convolutional Generative

- Adversarial Networks // CoRR. — 2015. — Vol. abs/1511.06434. — 1511.06434.
- [10] Breiman Leo. Random Forests // Machine Learning. — 2001. — Oct. — Vol. 45, no. 1. — P. 5–32. — Access mode: <https://doi.org/10.1023/A:1010933404324>.
- [11] Steinwart Ingo, Christmann Andreas. Support Vector Machines. — 1st edition. — Springer Publishing Company, Incorporated, 2008. — ISBN: 0387772413.
- [12] Doersch Carl. Tutorial on Variational Autoencoders // CoRR. — 2016. — Vol. abs/1606.05908. — 1606.05908.
- [13] Официальный сайт Python. — Access mode: <https://www.python.org/>.
- [14] Официальный репозиторий PyTorch. — Access mode: <https://github.com/pytorch/pytorch>.
- [15] A Tutorial on the Cross-Entropy Method / Pieter-Tjerk de Boer, Dirk P. Kroese, Shie Mannor, Reuven Y. Rubinstein // Annals of Operations Research. — 2005. — Feb. — Vol. 134, no. 1. — P. 19–67. — Access mode: <https://doi.org/10.1007/s10479-005-5724-z>.
- [16] Scikit-learn: Machine Learning in Python / F. Pedregosa, G. Varoquaux, A. Gramfort et al. // Journal of Machine Learning Research. — 2011. — Vol. 12. — P. 2825–2830.
- [17] Kingma Diederik P., Ba Jimmy. Adam: A Method for Stochastic Optimization // CoRR. — 2014. — Vol. abs/1412.6980. — 1412.6980.