

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«НОВОСИБИРСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ» (НОВОСИБИРСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ, НГУ)

Механико-математический факультет  
Кафедра «Международный математический центр»  
Направление подготовки «Математика и компьютерные науки»,  
бакалавриат

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
БАКАЛАВРА**

Кочанов Максим Александрович

**Применение алгоритмов сегментации опухолей  
головного мозга для предсказания состояния пациентов**

**«К защите допущен»**

Директор математического  
центра

д.ф.-м.н., профессор РАН

Марчук И.В. /\_\_\_\_\_

«\_\_\_\_» \_\_\_\_\_ 20\_\_\_\_ г.

**«Научный руководитель»**

Заведующий лабораторией,  
старший преподаватель

Тучинов Б.Н. /\_\_\_\_\_

«\_\_\_\_» \_\_\_\_\_ 20\_\_\_\_ г.

Дата защиты: «\_\_\_\_» \_\_\_\_\_ 20\_\_\_\_ г.

**Новосибирск  
2021**

## Реферат

*Название работы: Применение алгоритмов сегментации опухолей головного мозга для предсказания состояния пациентов*

*Количество страниц: 27*

*Количество рисунков: 22*

*Количество таблиц: 2*

*Количество использованных источников: 15*

*Ключевые слова: сегментация изображений, предсказание выживаемости, нейронные сети, случайный лес, опухоли головного мозга*

### КРАТКОЕ ОПИСАНИЕ

В дипломной работе рассматриваются 2 задачи: сегментация опухолей головного мозга высокой и низкой степени злокачественности; предсказание состояния пациентов с помощью результатов, полученных в ходе решения задачи сегментации. Данные предоставляются международным конкурсом Brain Tumor Segmentation Challenge 2020.

# Содержание

<b>1 ВВЕДЕНИЕ</b>	<b>4</b>
<b>2 ТЕОРЕТИЧЕСКАЯ ЧАСТЬ</b>	<b>5</b>
2.1 Постановка задач	5
2.2 Используемая терминология	5
2.3 Аналитический обзор	6
2.3.1 U-Net	7
2.3.2 DeepLabV3	8
2.3.3 Ternaus-Net	9
2.3.4 Pix2Pix	9
<b>3 АНАЛИЗ И ПРЕДВАРИТЕЛЬНАЯ ОБРАБОТКА ДАН- НЫХ</b>	<b>11</b>
3.1 Задача сегментации опухолей головного мозга	11
3.2 Задача прогнозирования состояния пациентов	13
<b>4 ЭКСПЕРИМЕНТАЛЬНАЯ ЧАСТЬ</b>	<b>15</b>
4.1 Задача сегментации опухолей головного мозга	15
4.2 Задача прогнозирования состояния пациентов	21
<b>5 ЗАКЛЮЧЕНИЕ</b>	<b>23</b>
<b>Приложение 1. Результаты сегментации опухолей головного мозга</b>	<b>26</b>

## 1. ВВЕДЕНИЕ

По данным Министерства здравоохранения РФ показатель смертности от онкологических заболеваний в России в настоящее время составляет более чем 200 человек на 100 тысяч населения. При этом у 65-80% умерших от онкологии обнаруживаются опухоли в головном мозге. Вопрос диагностики и контроля динамики нейроонкологических заболеваний у пациентов с первичными и вторичными опухолями головного мозга крайне актуален. Раннее обнаружение новообразований значительно улучшает результативность целевой терапии (оперативной, лучевой, химиотерапевтической и иммунологической). Наиболее распространенным методом первичного выявления, неинвазивного контроля и источником рекомендаций по дальнейшему лечению является МРТ. Головной мозг является сложной структурой, разные области которой обладают различной функциональной значимостью. В результате возникает проблема, связанная с оценкой влияния стадии развития опухоли и/или хирургического повреждения структур головного мозга на здоровье и качество жизни пациента, в том числе на клиническую выраженность неврологического дефицита. На данный момент уровень специалистов, производящих обработку данных, должен быть крайне высоким, чтобы исключить фактор «человеческой ошибки». Одним из возможных подходов к решению вышеперечисленных проблем является разработка новых методов диагностики на основе систем искусственного интеллекта. Но они требуют привлечения высококвалифицированных специалистов различных направлений: лучевых диагностов, неврологов, нейроонкологов, математиков, информатиков, программистов. Необходимо создание больших баз данных со специальным образом размеченными снимками.

## **2. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ**

### **2.1. Постановка задач**

Целью настоящей работы было создание прототипа системы обнаружения опухолей головного мозга и прогнозирование состояния пациента при персонализированной диагностике.

Для достижения цели поставлены следующие задачи:

1. Описать основные методы и определения, используемые в работе
2. Провести аналитический обзор современной литературы
3. Проанализировать медицинские данные и провести их предварительную обработку
4. Изучить существующие подходы и алгоритмы для сегментации опухолей головного мозга и выбрать наиболее эффективные
5. Провести исследования и эксперименты по задаче сегментации опухолей головного мозга
6. Извлечь признаки из МРТ-снимков для задачи классификации
7. Провести эксперименты и исследования по задаче классификации
8. Сделать выводы

### **2.2. Используемая терминология**

В этом пункте мы кратко опишем основные определения и методы, используемые далее.

Сегментация изображений - классификация пикселей изображения на определенные классы.

Сегментационная маска или маска - результат сегментации.

Сверточный слой или свертка - основной блок сверточной нейронной сети. Свертка содержит в себе заранее заданное количество фильтров. Каждый фильтр - это набор ядер, каждое из которых соответствует одному каналу

изображения. Размер ядер также определяется заранее. Ядро фильтра обрабатывает канал изображения, суммируя результаты поэлементного произведения участка канала на себя. Результаты обработки всех каналов ядрами одного фильтра суммируются для формирования общего канала, и к нему добавляется линейное смещение. Так происходит для каждого фильтра. Весовые коэффициенты ядер и коэффициент линейного смещения изначально задаются случайным образом и настраиваются в процессе обучения нейронной сети. Если пишут «свертка  $N \times N$ », то под  $N \times N$  подразумевают размер ядер фильтров свертки.

Операция max-pooling  $N \times N$  или операция объединения - это операция выбора максимального пикселя из  $N$  соседних по вертикали и  $N$  соседних по горизонтали. Max-pooling снижает размерность изображения.

Слой Drop-out с определенной вероятностью зануляет случайные выходы нейронов. Этот подход хорошо зарекомендовал себя в борьбе с переобучением нейронных сетей.

$k$ -кратная кросс-валидация - метод тренировки алгоритма машинного обучения.  $k$ -кратная кросс-валидация работает следующим образом: тренировочная выборка делится на  $k$  частей, алгоритм обучается на  $k - 1$  частях, а на оставшейся тестируется. Так повторяется  $k$  раз (каждый раз тестовая часть - новая). Затем значения метрики, показанные на тестовых выборках, складываются и делятся на  $k$ .

Алгоритм  $K$ -средних - алгоритм кластеризации данных, основанный на последовательном нахождении оптимального положения центроидов этих данных. Количество центроидов определяется заранее и подается на вход алгоритму.

Алгоритм «Случайный лес» - алгоритм классификации (регрессии), заключающийся в использовании ансамбля решающих деревьев.

## 2.3. Аналитический обзор

Для сегментации опухолей головного мозга по МРТ-снимкам мы возьмем несколько существующих архитектур нейронных сетей: U-Net, TernausNet, DeepLabV3, Pix2Pix и сравним их результаты между собой с помощью мет-

рики «коэффициент Дайса»:

$$dice = \frac{2|X \cap Y|}{|X| + |Y|}, \quad (2.1)$$

где  $|X|$  и  $|Y|$  - мощности двух наборов.

В дальнейшем мы будем приводить *dice* только на валидационных данных.

### 2.3.1. U-Net

U-Net [1] считается одной из стандартных и одной из самых популярных архитектур сверточных нейронных сетей для задач сегментации изображений. Она состоит из «сжимающего» и «разжимающего» путей. «Сжимаю-

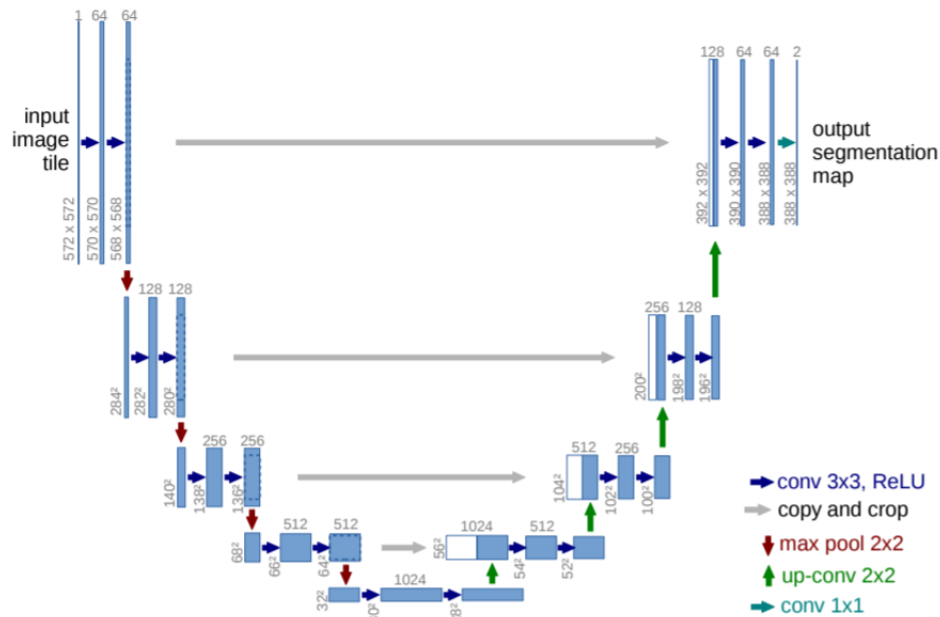


Рис. 2.1 – Архитектура U-Net

щий» путь, представленный в левой части рис. 1, предусматривает типичную архитектуру сверточной нейронной сети. Он состоит из нескольких последовательных блоков. Каждый блок содержит две свертки  $3 \times 3$ , за каждой из которых следует функция активации:

$$ReLU(x) = \begin{cases} 0 & x < 0 \\ x & x \geq 0 \end{cases} \quad (2.2)$$

и операция  $2 \times 2$  max-pooling с шагом 2 для снижения разрешения изображения. После каждого блока «сжимающего» пути количество каналов изображения удваивается. «Разжимающий» путь также состоит из нескольких последовательных блоков. Вместо операции  $2 \times 2$  max-pooling блок этого пути содержит операцию обратной свертки, которая увеличивает разрешение изображения и снижает количество его каналов. Также каждый блок «разжимающего» пути содержит в себе операцию конкатенации с каналами из соответствующего блока «сжимающего» пути. На последнем слое нейронной сети применяется свертка  $1 \times 1$  для того, чтобы отобразить каждый компонент предыдущего слоя на желаемое количество каналов.

### 2.3.2. DeepLabV3

Особенностью DeepLabV3 [2] является применение расширенных сверток (atrous convolutions). В отличие от классической свертки ядро расширенной свертки разбивается на фрагменты, расстояние между которыми задается параметром *rate*.

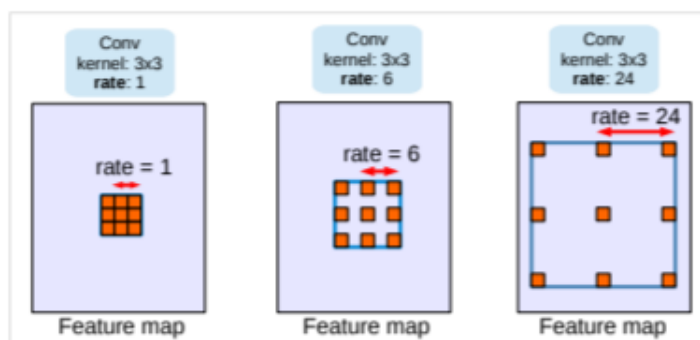


Figure 1. Atrous convolution with kernel size  $3 \times 3$  and different rates. Standard convolution corresponds to atrous convolution with  $rate = 1$ . Employing large value of atrous rate enlarges the model's field-of-view, enabling object encoding at multiple scales.

Рис. 2.2 – Примеры ядра расширенной свертки

С помощью сверток такого вида был создан специальный слой нейронной сети - Atrous Spatial Pyramid Pooling (ASPP), который является конкатенацией классической свертки  $1 \times 1$  и трех расширенных сверток  $3 \times 3$  со следующими  $rate = (6, 12, 18)$ .



DeepLabV3 имеет следующую архитектуру: признаки извлекаются из магистральной сети (VGG, DenseNet, ResNet). В последних нескольких блоках магистрали используется расширенная свертка. Далее применяется слой ASPP. Выходные данные из слоя ASPP проходят через свертку  $1 \times 1$ , чтобы получить фактический размер изображения, который будет окончательной сегментированной маской для изображения.

### 2.3.3. Ternaus-Net

Ternaus-Net [8] - это нейронная сеть, архитектура которой построена на базе U-Net. Ternaus-Net получается из U-Net с помощью замены сжимающего пути U-Net на независимую нейронную сеть. В работе для такой замены используется ResNet18, предобученная на датасете ImageNet.

### 2.3.4. Pix2Pix

Для сегментации изображений мы можем рассматривать не только нейронные сети, специализирующиеся на этом. Взглянем на задачу сегментации немного шире: результат сегментации - изображение, значит если нейронная сеть будет не сегментировать опухоль головного мозга, а генерировать изображение опухоли и сопоставлять ее МРТ-снимку, то в случае успеха, результат генерации этого изображения будет совпадать с результатом сегментации. Такая задача называется Image-to-Image-Translation, а нейронные сети, решающие ее, - условно- состязательными генеративными [3].

Условно-состязательные генеративные сети отображают входное изображение  $x$  в изображение  $y$ . Архитектура таких нейронных сетей включает в себя наличие двух независимых нейронных сетей: генератор и дискриминатор. Генератор  $G$  обучен производить выходные данные, которые нельзя отличить от «реальных» изображений с помощью обученного дискриминатора  $D$ , который обучен как можно лучше выявлять «подделки» генератора. Схема этой процедуры обучения представлена на следующем рисунке.



Рис. 2.3

Выбранная нами условно-сопязательная генеративная сеть называется Pix2Pix. В качестве генератора Pix2Pix использует нейросеть, созданную на базе U-Net, а в качестве дискриминатора используется дискриминатор генеративно-сопязательной сети PatchGan, который является сверточной нейронной сетью, классифицирующей изображения.

### 3. АНАЛИЗ И ПРЕДВАРИТЕЛЬНАЯ ОБРАБОТКА ДАННЫХ

#### 3.1. Задача сегментации опухолей головного мозга

Как правило, результатом МРТ - исследования головного мозга пациента выступают несколько файлов формата DICOM с «непрерывными» (по слайсам) МРТ-снимками головного мозга. Каждый такой файл отвечает определенному режиму МРТ-последовательностей: T1, T2, T1CE, FLAIR. Для ознакомления с результатами обследования и просмотра файлов формата DICOM врачи и пациенты используют специальные приложения (например, 3D-Slicer).

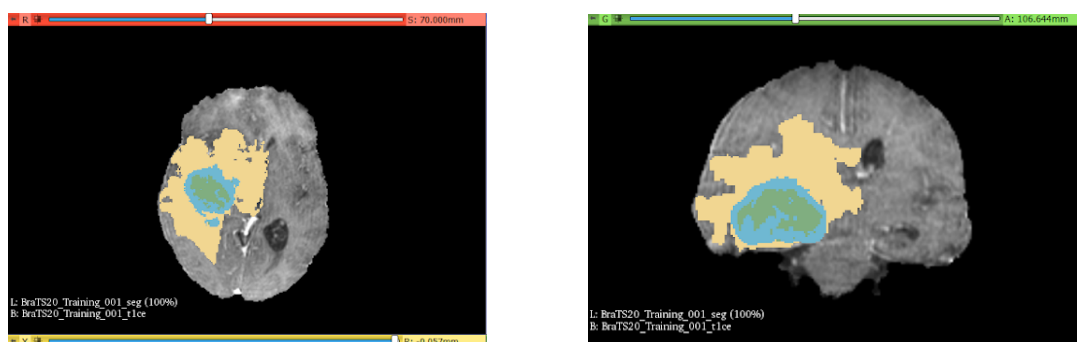


Рис. 3.4 – Вид сверху и вид спереди

Клинически полученные данные МРТ - обследований и соответствующие им сегментационные маски предоставляются Brain Tumor Segmentation Challenge 2020 в виде файлов формата DICOM.

Субрегионами, рассматриваемыми для оценки в сегментационных масках, являются:

- Перитуморальный отек
- Ядро опухоли
- Вся опухоль

Перитуморальный отек описывается областями, которые демонстрируют повышенную интенсивность в T1Gd по сравнению с T1, а также по сравнению со «здоровым» белым веществом в T1Gd. Ядро опухоли описывает

основную часть опухоли, которую обычно удаляют. Ядро опухоли влечет за собой усиливающую опухоль, а также некротические (заполненные жидкостью) и неусиливающие (твердые) части опухоли. Появление некротического и неусиливающего ядра опухоли обычно является гипоинтенсивным в T1Gd по сравнению с T1. Вся опухоль описывает полную степень заболевания, так как влечет за собой ядро опухоли и перитуморальный отек, который обычно отображается с помощью сверхинтенсивного сигнала в FLAIR.

К сожалению, файлы в таком формате не пригодны для обучения алгоритмов: из «непрерывного» представления файла в формате DICOM мы извлекли отдельные 2D - изображения (слайсы), представленные в привычном формате для хранения изображений (.png). Для этого мы применили библиотеку nibabel для Python.

Извлеченные 2D - изображения являются одноканальными. Единственный канал сегментационной маски представлен пикселями, значения которых равны либо 0, либо 1, либо 2, либо 4. По одному значению на каждый сегментируемый класс: пустота, усиливающая опухоль, ядро опухоли и вся опухоль. Для того, чтобы подготовить маски для мультиклассовой сегментации, из одного имеющегося канала мы сделали три канала, пиксели которых принимают значения из  $\{0, 1\}$ . Каждый канал изображения отвечает за свой сегментируемый класс: нахождение на  $i$ -м месте пикселя со значением 1 (0) в  $j$ -м канале означает присутствие (отсутствие)  $j$ -го сегментируемого класса в маске в  $i$ -м пикселе. Если же маска пуста (т.е. маска МРТ-снимка без видимых патологий), то все каналы маски заполнены пикселями со значениями, равными нулю.

Выбрав наиболее информативную модальность (в нашем случае - это T1CE) и переведя медицинские файлы в изображения формата (.png), мы преобразовали значения пикселей из отрезка  $[-32768, 32767]$  в отрезок  $[0, 255]$  с помощью функции:  $y(x) = 255/\lambda$ , где  $\lambda$  - максимальное значение среди значений пикселей изображения.

Закончив обработку данных и сформировав пары вида (МРТ-снимок, маска), мы получили готовый датасет для обучения алгоритмов сегментации.

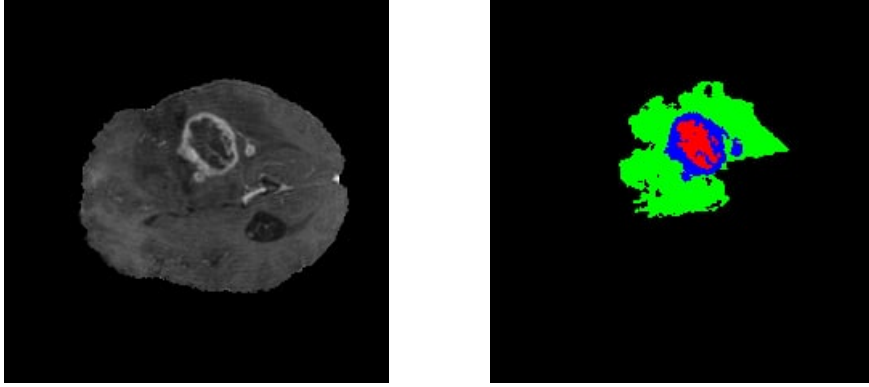


Рис. 3.5 – МРТ - снимок и соответствующая ему сегментационная маска

Набор тренировочных данных состоит из изображений МРТ-снимков 369 пациентов (57195 изображений). Набор валидационных (тестовых) данных состоит из изображений МРТ-снимков 81 пациентов (12555 изображений).

### 3.2. Задача прогнозирования состояния пациентов

С помощью библиотеки `pyradiomics` из DICOM-файлов, предоставляемых Brain Tumor Segmentation Challenge, мы извлекли 321 численных признаков. Также Brain Tumor Segmentation Challenge предоставляет данные о возрасте пациентов и количестве прожитых ими дней.

Чтобы разделить данные о количестве прожитых пациентами дней на классы, мы решили использовать алгоритм К-средних. Прежде всего необходимо было понять оптимальное количество классов для данного алгоритма. Для выбора оптимального числа классов мы воспользовались следующим методом: мы выбрали то количество классов, начиная с которого функционал

$$J(C) = \sum_{k=1}^K \sum_{i \in C_k} ||x_i - \mu_k||^2 \rightarrow \min_C, \quad (3.1)$$

падает "уже не так быстро". ( $C$  - множество классов мощности  $K$ ,  $\mu_k$  - центр тяжести кластера  $C_k$ )

Или более формально:

$$D(k) = \frac{|J(C_k) - J(C_{k+1})|}{|J(C_{k-1}) - J(C_k)|} \rightarrow \min_k \quad (3.2)$$

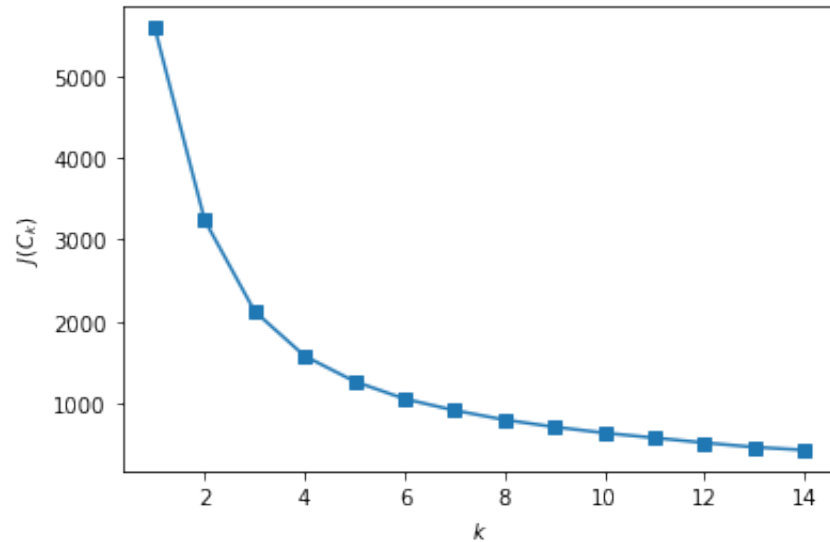


Рис. 3.6

Мы выбрали 3 класса выживаемости:

- 0 - 387 дней
- 388 - 946 дней
- 947 и более дней

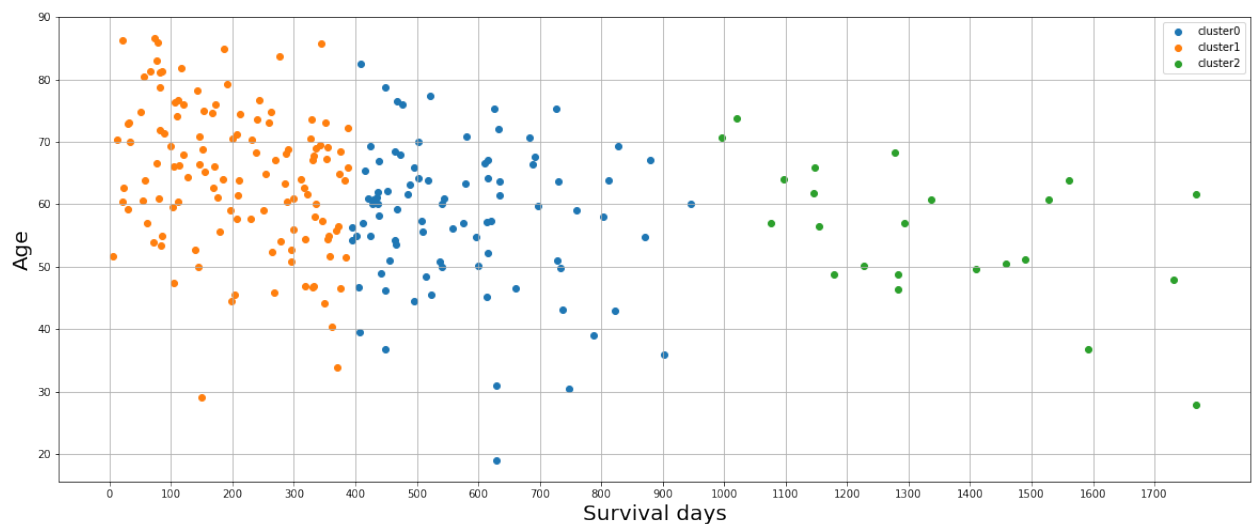


Рис. 3.7 – Зависимость дней выживаемости от возраста пациента

Таким образом мы получили тренировочную выборку для задачи предсказания выживаемости пациентов, состоящую из 321 численных признаков и возраста, и целевой признак - класс выживаемости.

## 4. ЭКСПЕРИМЕНТАЛЬНАЯ ЧАСТЬ

### 4.1. Задача сегментации опухолей головного мозга

В ходе первого эксперимента установлена функция ошибки, с которой каждая архитектура достигает максимального *dice*. Функции ошибок выбирались из следующего набора:

(a)  $0.3 * loss_{bce} + 0.7 * loss_{dice}$

(b)  $0.5 * loss_{bce} + 0.5 * loss_{dice}$

(c)  $0.3 * loss_{bce} + 0.2 * loss_{dice} + 0.5 * loss_{focal}$

Где:

- $loss_{dice} = 1 - dice$

- $loss_{bce} = \frac{1}{N} * (-y \log(p) - (1 - y) \log(1 - p))$ , где

- $N$  - количество примеров, подающихся на вход нейронной сети

- $y$  - истинное значение пикселя

- $p$  - предсказанное нейронной сетью значение пикселя

- $loss_{focal} = (1 - e^{-loss_{bce}})^2 * loss_{bce}$

Для каждой из выбранных нами архитектур (кроме Pix2Pix, причина будет объяснена далее) была обучена модель с каждой функцией ошибок из этого множества. Таким образом в ходе этого эксперимента было обучено 9 моделей. Для проведения экспериментов использовались 2 компьютера. Первый компьютер: Intel(R) Core(TM) i5-9500, Nvidia GeForce RTX 2060 Super, 16 ГБ ОЗУ, Windows 10 Home. Второй компьютер: Intel(R) Core(TM) i5-9500, Nvidia GeForce RTX 2080, 8 ГБ ОЗУ, Windows 10 Home. Все модели, которые тренировались в рамках этого эксперимента, быстро переобучались. Чтобы бороться с эффектом переобучения, мы сохраняли веса моделей при достижении лучшего *dice*. Когда модель начинала переобучаться, *dice* начинал уменьшаться, и мы, соответственно, не сохраняли веса модели. Так же для борьбы с переобучением в U-Net было добавлено 2 слоя Drop-out.

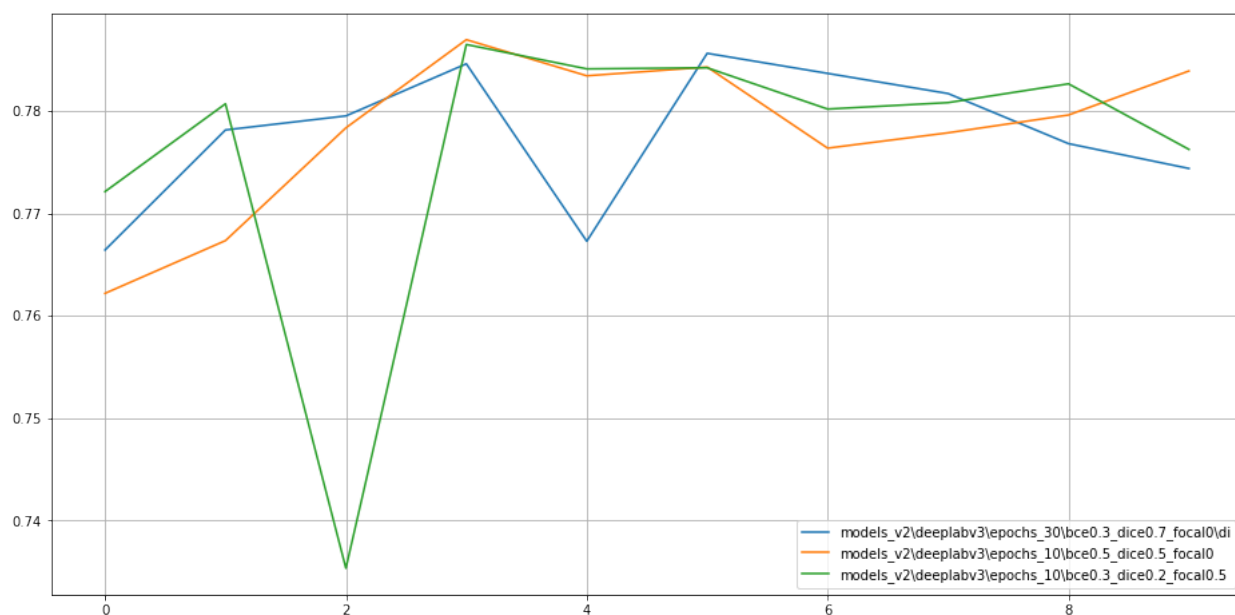


Рис. 4.8 – *dice* в зависимости от количества эпох (DeepLabV3)

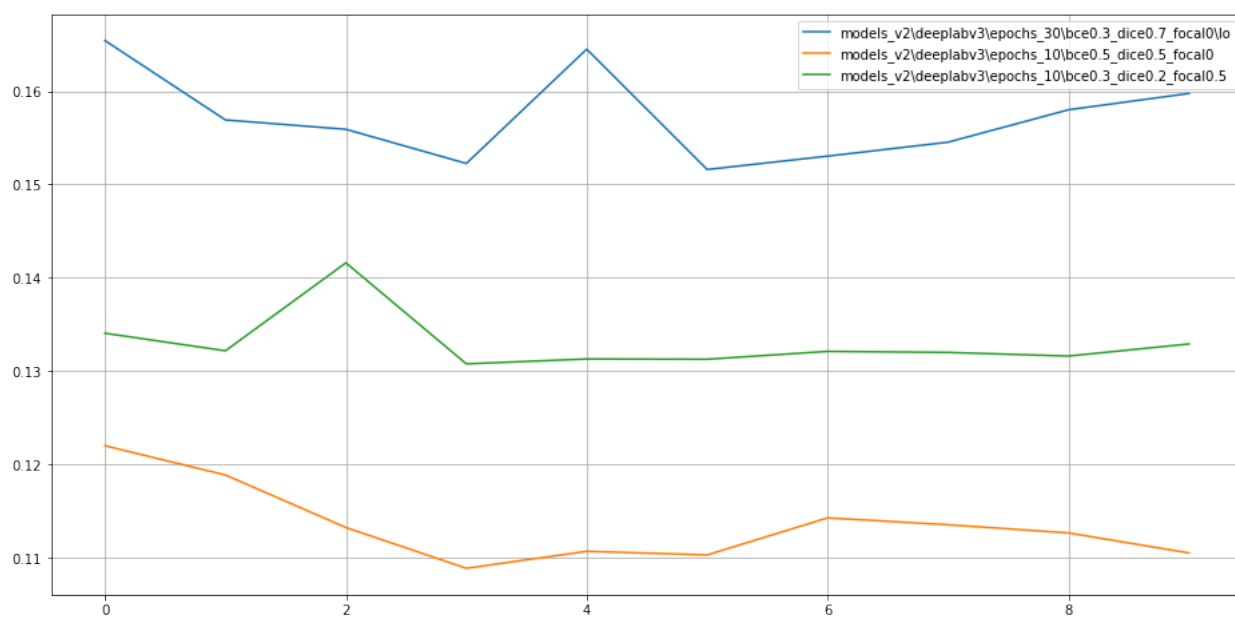


Рис. 4.9 – Значения функций ошибки в зависимости от количества эпох (DeepLabV3)



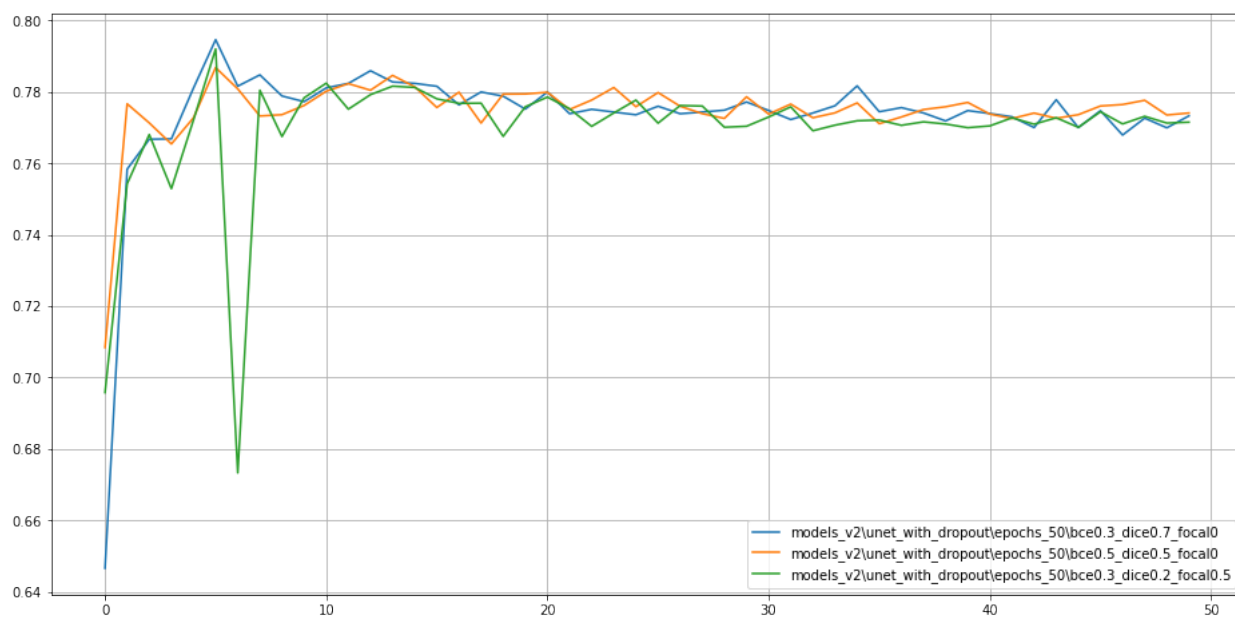


Рис. 4.10 – *dice* в зависимости от количества эпох (U-Net)

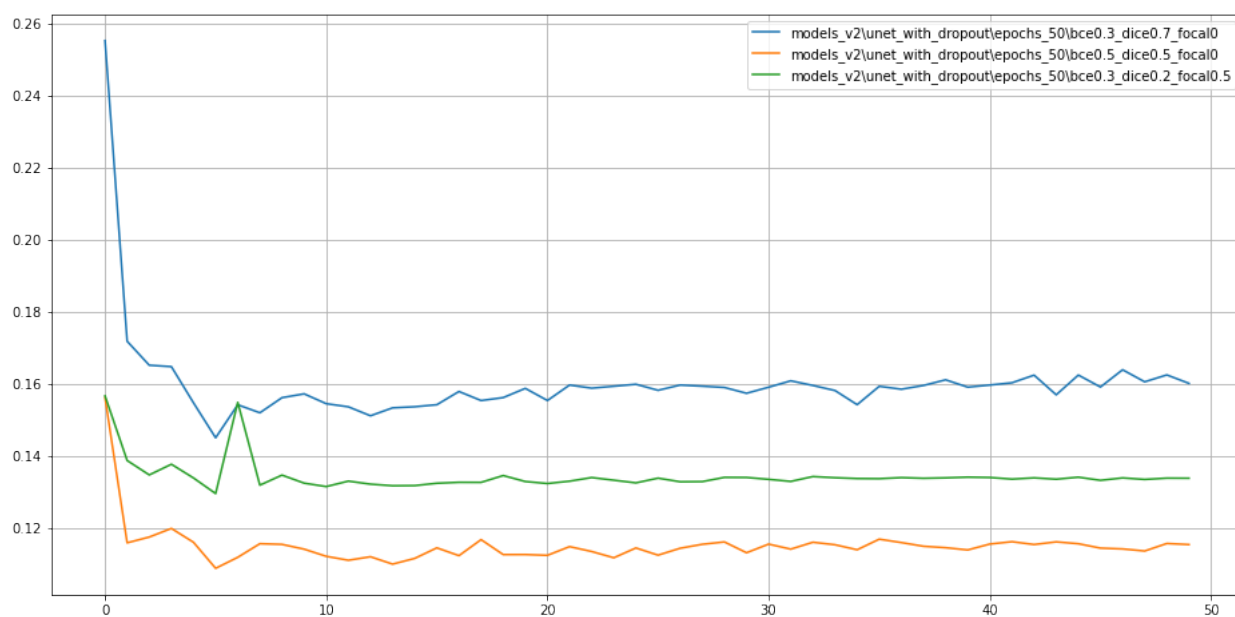


Рис. 4.11 – Значения функций ошибки в зависимости от количества эпох (U-Net)

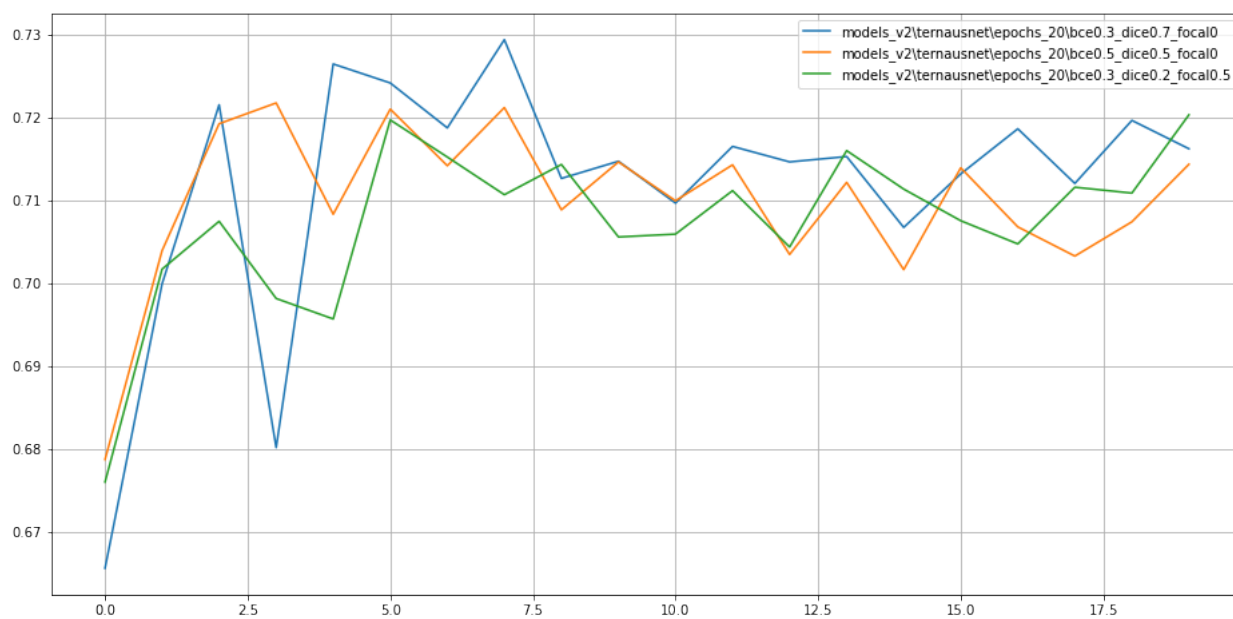


Рис. 4.12 – *dice* в зависимости от количества эпох (Ternaus-Net)

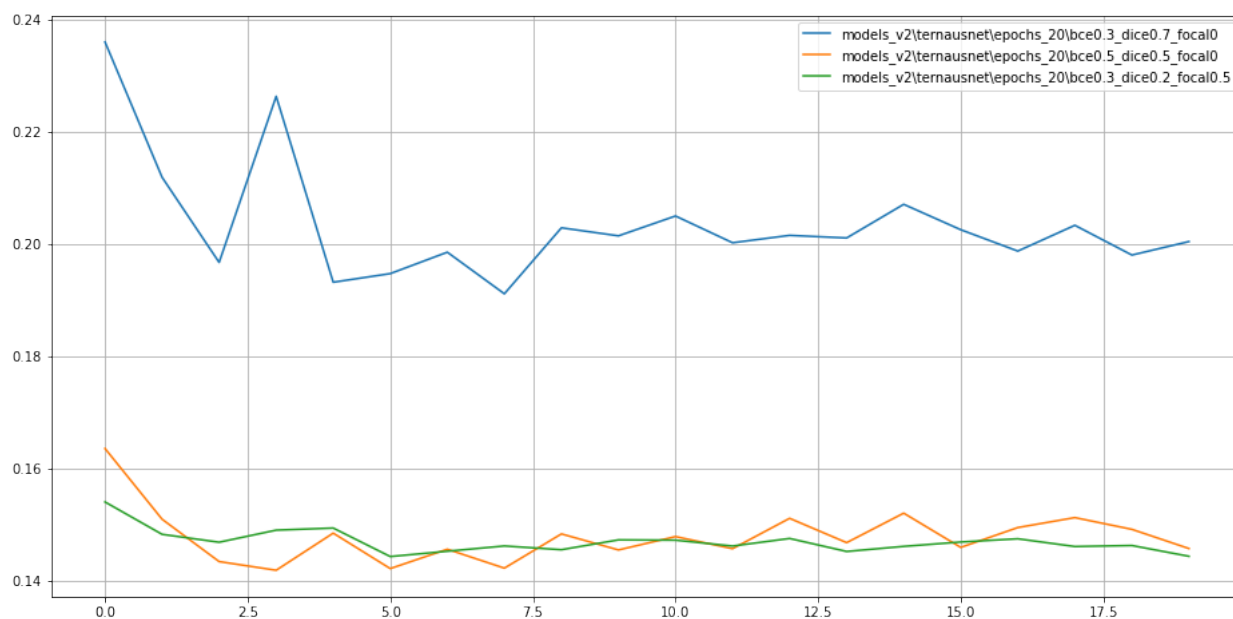


Рис. 4.13 – Значения функций ошибки в зависимости от количества эпох (Ternaus-Net)

Таблица 4.1: лучший *dice* моделей в зависимости от функции ошибки

Функция ошибки	<i>DeepLabV3</i>	<i>U – Net</i>	<i>Ternaus – Net</i>
(a)	0.786	<b>0.795</b>	<b>0.729</b>
(b)	<b>0.7869</b>	0.787	0.722
(c)	0.7865	0.729	0.720

Таким образом, для задачи сегментации по результатам эксперимента мы выбрали U-Net с функцией ошибки -  $0.3 * loss_{bce} + 0.7 * loss_{dice}$  и в дальнейшем работали с этой моделью.

Мы также приводим некоторые результаты сегментации опухолей головного мозга используемыми архитектурами:

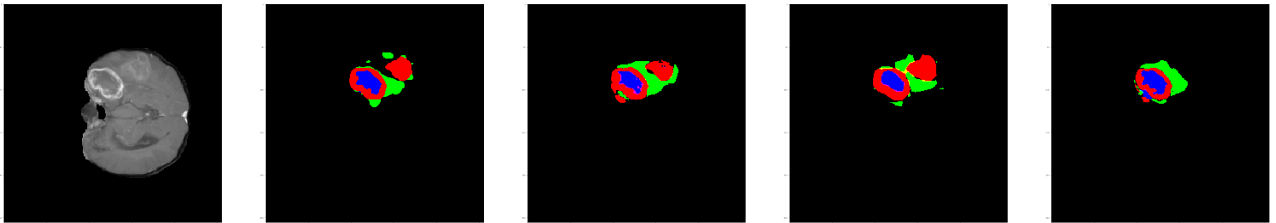


Рис. 4.14 – Слева направо: МРТ, оригинальная маска, U-Net, DeepLabV3, Ternaus-Net

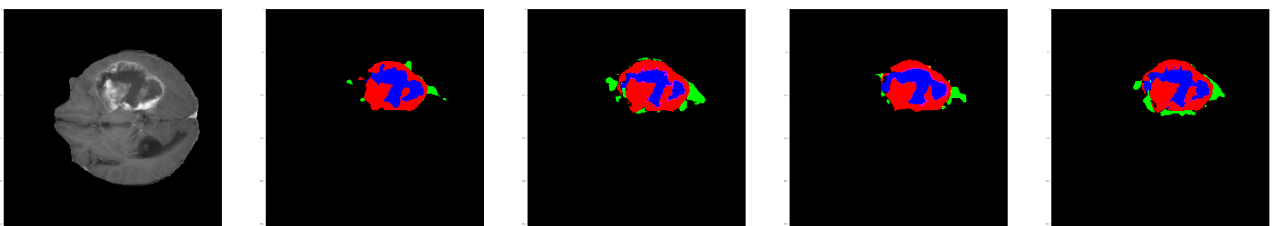


Рис. 4.15 – Слева направо: МРТ, оригинальная маска, U-Net, DeepLabV3, Ternaus-Net

Pix2Pix, как было сказано ранее, не принимала участия в проведенных экспериментах, поскольку, являясь условно-сопоставительной генеративной нейронной сетью, имеет функции ошибки, несравнимые с функциями ошибок «классических» архитектур для сегментации. Тем не менее, эта архитектура была испытана нами. Некоторые результаты представлены на следующих двух рисунках.

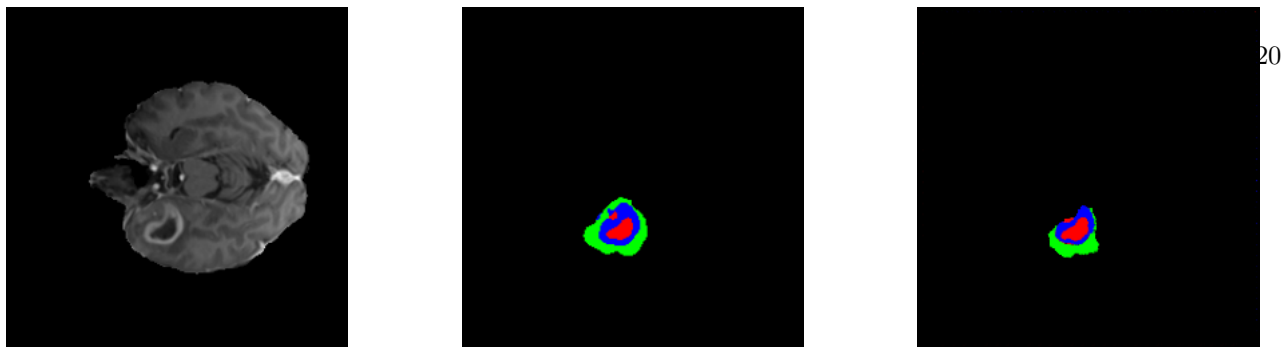


Рис. 4.16 – Слева направо: MRT, оригинальная маска, предсказанная маска

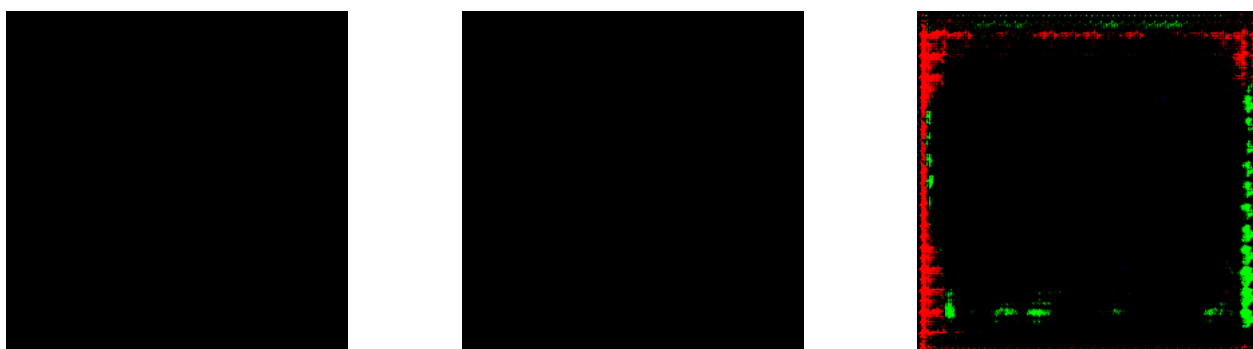


Рис. 4.17 – Слева направо: MRT, оригинальная маска, предсказанная маска

Проблема, из-за которой мы отказались от дальнейшего использования этой архитектуры, продемонстрирована на рисунке выше. При поступлении на вход пустых изображений Pix2Pix генерировала не пустые изображения, а изображения с шумом, в связи с чем *dice* был относительно низким.

В задачах компьютерного зрения для тренировочной выборки часто применяются трансформации изображений с целью достижения лучших результатов. В ходе второго эксперимента был определен оптимальный набор трансформаций. Мы выбирали из трех наборов:

- Аффинные трансформации: вертикальный поворот, горизонтальный поворот, случайное вращение
- Слабые трансформации: аффинные трансформации, изменение яркости, изменение контраста, изменение гаммы
- Сильные трансформации: слабые трансформации, сдвиг, масштабирование, искажение сетки, оптическое искажение

В ходе эксперимента были обучены 3 модели U-Net - по одной для каждого набора трансформаций.

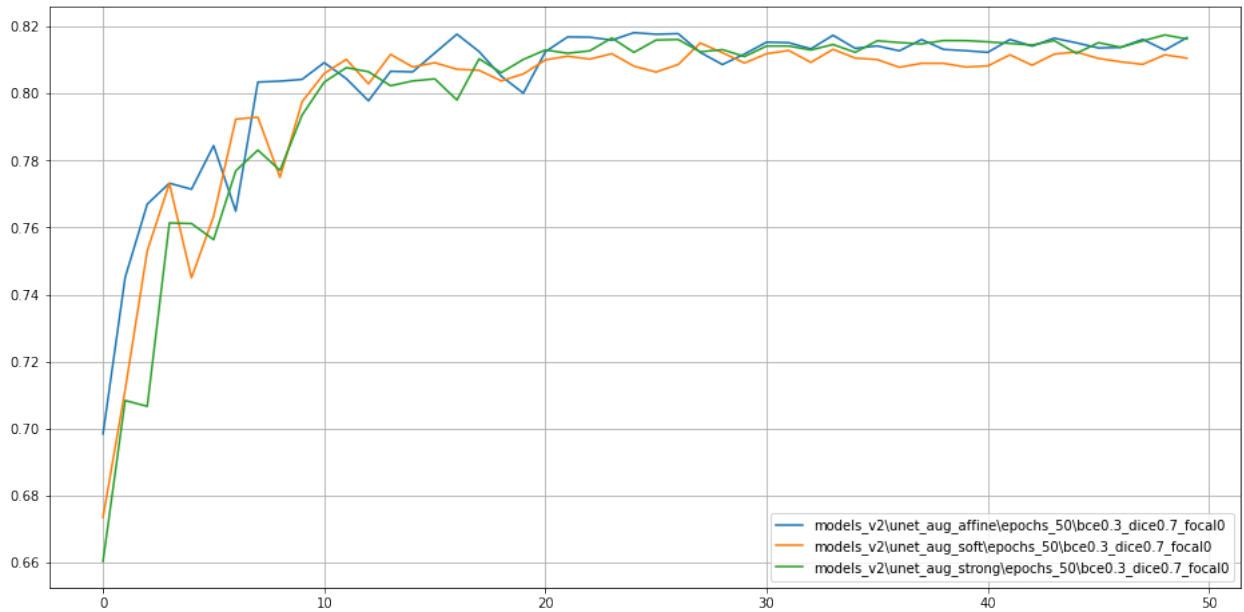


Рис. 4.18 – *dice* в зависимости от количества эпох

Таблица 4.2: лучший *dice* в зависимости от трансформаций тренировочных данных

Трансформации	$U - Net$
Афинные	<b>0.818</b>
Слабые	0.815
Сильные	0.816

Таким образом, набор афинных трансформаций - оптимальный выбор. В дальнейшем мы работали с ним.

## 4.2. Задача прогнозирования состояния пациентов

Для предсказания классов выживаемости был использован алгоритм классификации «Случайный лес».

Результаты классификации мы оценивали с помощью метрики

$$F1 = \frac{2 * precision * recall}{precision + recall}, \quad (4.1)$$

где

$$precision = \frac{true \ positive}{true \ positive + false \ positive}, \quad (4.2)$$

$$recall = \frac{true \ positive}{true \ positive + false \ negative} \quad (4.3)$$

Для поиска оптимальных признаков была реализована кросс-валидация с последовательным добавлением признаков [4,5]. Признак добавлялся в набор оптимальных, если  $F1$  после его добавления возрастала. В результате оптимальный набор признаков выглядел следующим образом:

- 8 признаков, описывающих форму
- 1 статистика первого порядка
- 3 признака, зависящих от интенсивности серого цвета

После проведения экспериментов  $F1$  случайного леса составила **0.574**.

Для улучшения качества предсказания была реализована классическая пятикратная кросс-валидация. Подбираемыми параметрами случайного леса были:

- Количество деревьев решения
- Функция ошибки
- Максимальная глубина дерева решения
- Минимальное количество примеров, необходимых для разделения внутреннего узла
- Минимальное количество примеров, необходимых для нахождения в конечном узле
- Количество признаков, которые следует учитывать при поиске лучшего разделения

После проведения экспериментов и подбора параметров случайного леса,  $F1$  составила **0.586**.

## 5. ЗАКЛЮЧЕНИЕ

В работе были решены 2 задачи: сегментация опухолей головного мозга и предсказание временного интервала жизни пациента. В первой задаче мы достигли коэффициента Дайса, равного 0.818. Во второй задаче мы достигли метрику F1, равную 0.586. Благодаря этому можно говорить об успешной разработке прототипа системы прогнозирования состояния пациента при персонализированной диагностике.

Работа была представлена на международной научной студенческой конференции 2021 в секции «Информационно-управляющие системы и интернет вещей».

Дальнейшие планы заключаются в применении 3D моделей нейронных сетей для задачи сегментации опухолей головного мозга и использовании данных медицинских анализов пациентов. Таким образом мы планируем увеличить как качество сегментации опухолей головного мозга, так и качество предсказания временного интервала жизни пациентов.

## Список литературы

1. Olaf Ronneberger, Philipp Fischer, Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. URL: <https://arxiv.org/pdf/1505.04597>
2. Liang-Chieh Chen, George Papandreou, Florian Schroff, Hartwig Adam. Rethinking Atrous Convolution for Semantic Image Segmentation. URL: <https://arxiv.org/pdf/1706.05587>
3. Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, Alexei A. Efros. Image-to-Image Translation with Conditional Adversarial Networks. URL: <https://arxiv.org/pdf/1611.07004>
4. Vikas Kumar Anand, Sanjeev Grampurohit, Pranav Aurangabadkar, Avinash Kori, Mahendra Khened, Raghavendra S Bhat, Ganapathy Krishnamurthi. Brain Tumor Segmentation and Survival Prediction using Automatic Hard mining in 3D CNN Architecture. URL: <https://arxiv.org/pdf/2101.01546>
5. Alain Jungo, Richard McKinley, Raphael Meier, Urspeter Knecht, Luis Vera, Julian Perez-Beteta, David Molina-Garcia, Victor M. Perez-Garcia, Roland Wiest, Mauricio Reyes. Towards Uncertainty-Assisted Brain Tumor Segmentation and Survival Prediction. Lecture Notes in Computer Science, 2018.
6. Debesh Jha, Michael A. Riegler, Dag Johansen, Pål Halvorsen, Håvard D. Johansen. DoubleU-Net: A Deep Convolutional Neural Network for Medical Image Segmentation. URL: <https://arxiv.org/abs/2006.04868>
7. Fabian Isensee, Paul F. Jaeger, Peter M. Full, Philipp Vollmuth, Klaus H. Maier-Hein. nnU-Net for Brain Tumor Segmentation. URL: <https://arxiv.org/abs/2011.00848>
8. Vladimir Iglovikov, Alexey Shvets. TerausNet: U-Net with VGG11 Encoder Pre-Trained on ImageNet for Image Segmentation. URL: <https://arxiv.org/abs/1801.05746>
9. Fabian Isensee, Klaus H. Maier-Hein. An attempt at beating the 3D U-Net. URL: <https://arxiv.org/abs/1908.02182>
10. Andriy Myronenko. 3D MRI brain tumor segmentation using autoencoder regularization. URL: <https://arxiv.org/abs/1810.11654>
11. Mateusz Buda, Ashirbani Saha, Maciej A. Mazurowski. Association



- of genomic subtypes of lower-grade gliomas with shape features automatically extracted by a deep learning algorithm. URL: <https://arxiv.org/abs/1906.03720>
12. Xiaomeng Li, Hao Chen, Xiaojuan Qi, Qi Dou, Chi-Wing Fu, Pheng Ann Heng. H-DenseUNet: Hybrid Densely Connected UNet for Liver and Tumor Segmentation from CT Volumes. URL: <https://arxiv.org/abs/1709.07330>
  13. Dominik Müller, Frank Kramer. MIScnn: A Framework for Medical Image Segmentation with Convolutional Neural Networks and Deep Learning. URL: <https://arxiv.org/abs/1910.09308>
  14. Sahar Yousefi, Hessam Sokooti, Mohamed S. Elmahdy, Irene M. Lips, Mohammad T. Manzuri Shalmani, Roel T. Zinkstok, Frank J. W. M. Dankers, Marius Staring . Esophageal Tumor Segmentation in CT Images using Dilated Dense Attention Unet (DDAUnet). URL: <https://arxiv.org/abs/2012.03242>
  15. Xiaoqing Guo, Chen Yang, Pak Lun Lam, Peter Y. M. Woo, Yixuan Yuan. Domain Knowledge Based Brain Tumor Segmentation and Overall Survival Prediction. URL: <https://arxiv.org/abs/1912.07224>

## ПРИЛОЖЕНИЕ 1

## Результаты сегментации опухолей головного мозга

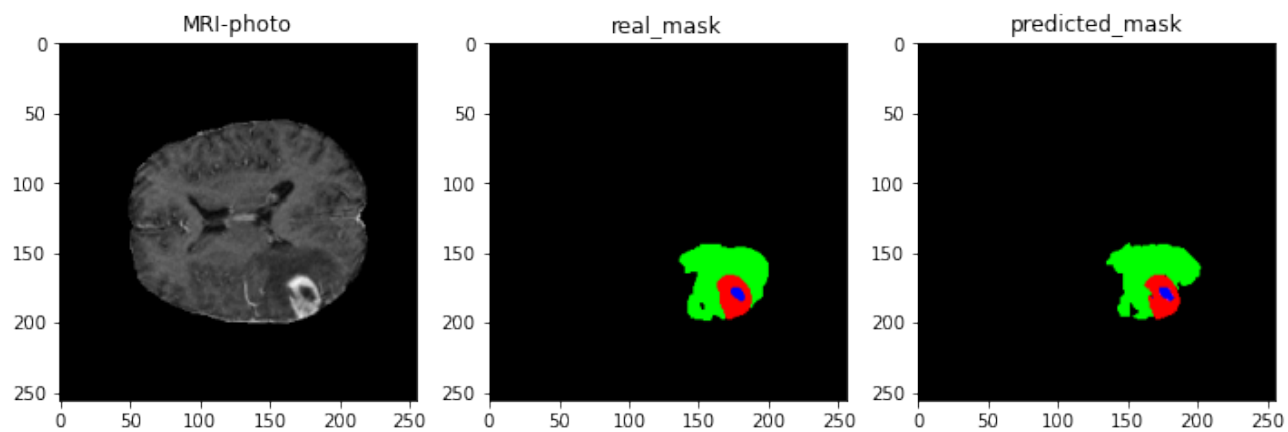


Рис. А.19

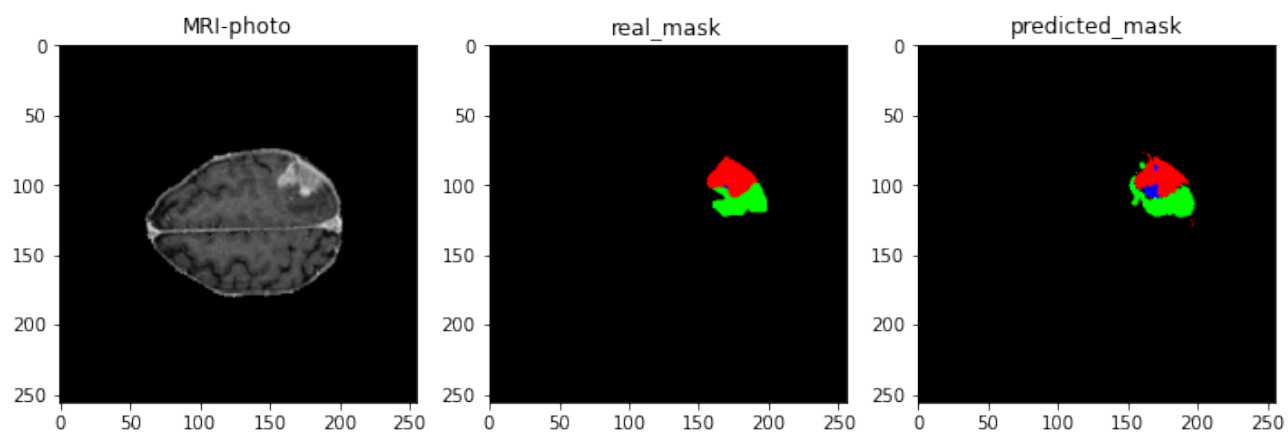


Рис. А.20

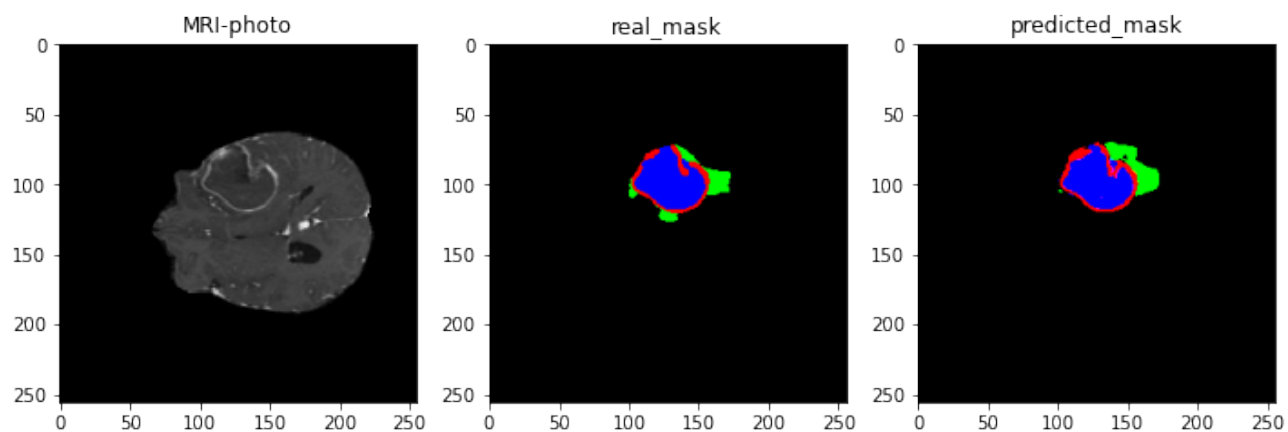


Рис. А.21

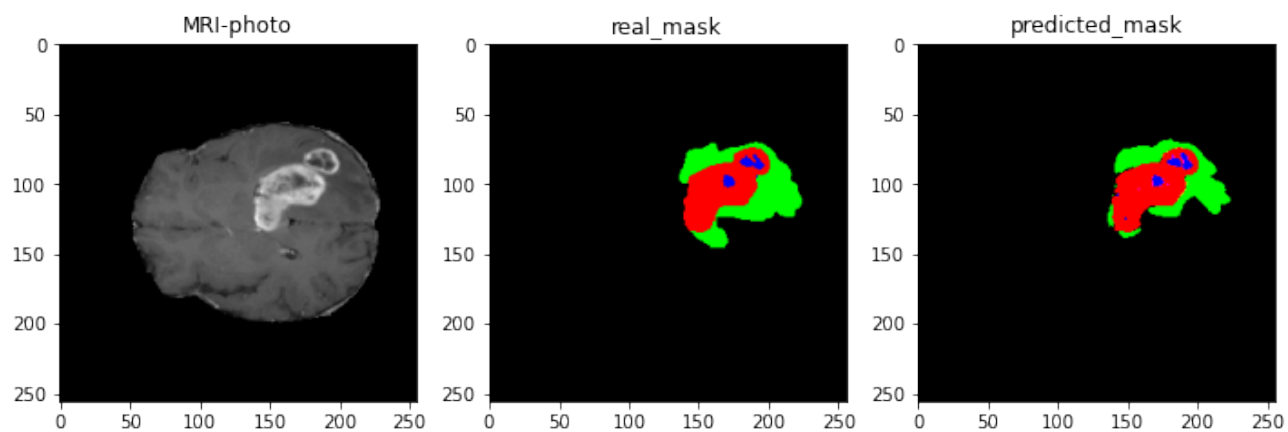


Рис. А.22