ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
ФГАОУ ВО НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

Факультет компьютерных наук

Образовательная программа «Прикладная математика и информатика»

Отчет об исследовательском проекте на тему:

Анализ ЭКГ на основе ML

Выполнил студент:

группы #БПМИ228, 2 курса Ковыляев Александр Максимович

Принял руководитель проекта:

Хельвас Александр Валериевич

старший преподаватель, МФТИ.

Москва 2024

Содержание

Термины и определения

Искусственный интеллект (далее - ИИ) — - комплекс технологических решений, позволяющий имитировать когнитивные функции человека (включая самообучение и поиск решений без заранее заданного алгоритма) и получать при выполнении конкретных задач результаты, сопоставимые, как минимум, с результатами интеллектуальной деятельности человека. Комплекс технологических решений включает в себя информационно-коммуникационную инфраструктуру, программное обеспечение (в том числе в котором используются методы машинного обучения), процессы и сервисы по обработке данных и поиску решений;

Открытые данные — Информация, размещаемая ее обладателями в сети «Интернет» в формате, допускающем автоматизированную обработку без предварительных изменений человеком в целях повторного ее использования

f_0 – медиана частоты в спектре;

f_{HR} - частота сердечных сокращений;

Обозначения и сокращения

ASR — automatic speech recognition (автоматическое распознавание речи)

CSV — (CSV от англ. Comma-Separated Values — значения, разделённые запятыми) — текстовый формат, предназначенный для представления табличных данных

Dynamic HTML — набор средств, которые позволяют создавать более интерактивные Web-страницы без увеличения загрузки сервера

HTML — Язык гипертекстовой разметки документов (от англ. Hypertext Markup Language – "язык гипертекстовой разметки")

HTTP — Протокол прикладного уровня для передачи данных, используемый в Web (от англ. HyperText Transfer Protocol - «протокол передачи гипертекста»)

IP-адрес — Уникальный сетевой адрес узла в компьютерной сети, построенной по протоколу IP

JavaScript — Прототипно-ориентированный сценарный язык программирования. Наиболее широкое применение находит в браузерах как язык сценариев для придания интерактивности веб-страницам

JPEG (JPG) — JPEG - один из популярных графических форматов, применяемый для хранения фотоизображений и подобных им изображений. Файлы, содержащие данные JPEG, обычно имеют расширения .jpg, .jfif, .jpe или .jpeg.

LSTM — Long short-term memory - разновидность архитектуры рекуррентных нейронных сетей, обладающая эффектом "краткосрочной памяти"

MS SQL — Microsoft SQL Server — система управления реляционными базами данных (РСУБД), разработанная корпорацией Microsoft

PDF — Portable Document Format (PDF) — межплатформенный формат электронных документов, разработанный фирмой Adobe Systems

PNG — Растровый формат хранения графической информации, использующий сжатие без потерь качества

STFT — Short-Time Fourier Transform – оконное преобразование Фурье

НСИ — Нормативно – справочная информация

НИР — Научно - исследовательская работа

АС — Автоматизированная система

Интернет — Информационно телекоммуникационная сеть Интернет

ПО — Программное обеспечение

АИС — Автоматизированная информационная система.

АРМ — Автоматизированная рабочее место

ЧСС — Частота сердечных сокращений

CPU — Central processing unit

Введение

Внедрение анализа ЭКГ при помощи компьютерных технологий, а именно машинного обучения способствует увеличению точности диагнозов, уменьшению нагрузки на врачей и возможности выявления заболеваний на более ранних стадиях по ещё маловыраженным, незримым для человеческого глаза, характеристикам.

ЭКГ - электрокардиограмма - является одним из показателей сердечной активности. Нормальная ЭКГ состоит из нулевой линии, 6 зубцов (P, Q, R, S, T и иногда небольшой зубец U), а также двух сегментов (PQ и ST). Заболевания сердца вызывают на ЭКГ отклонение от нормы, это называется аритмией сердца. 0.1

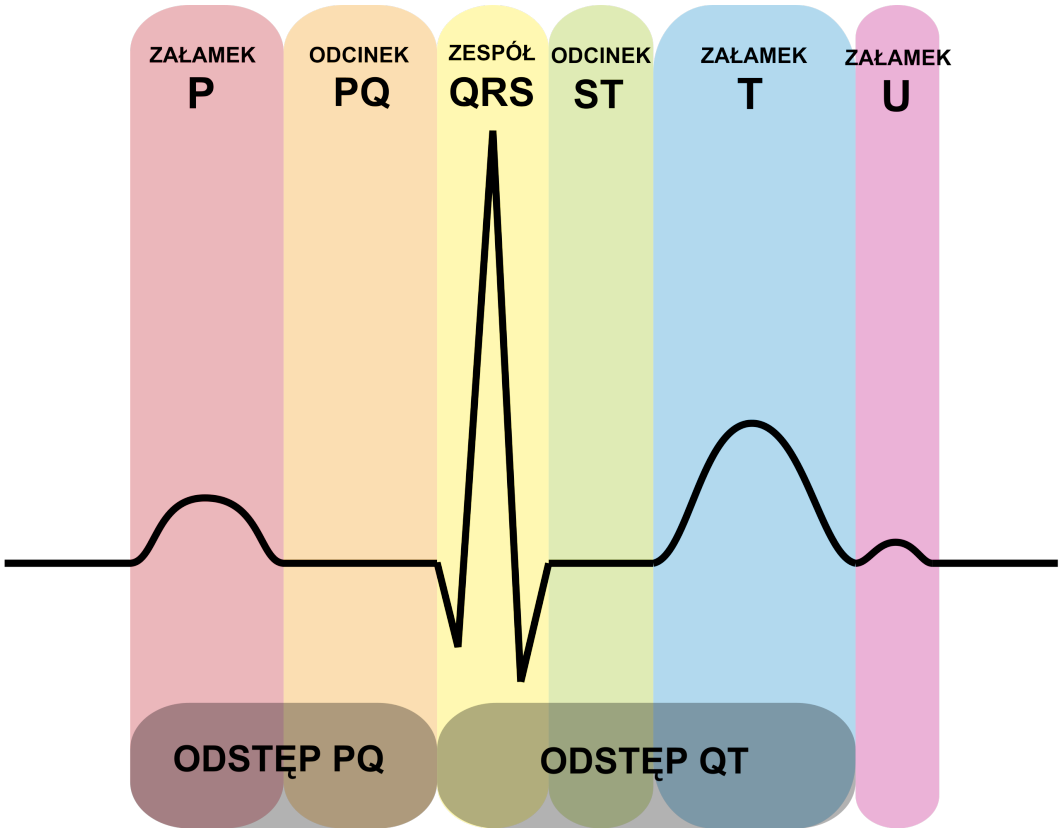

Рисунок 0.1 — P, QRS, T, U

В современном мире технология снятия ЭКГ достигла значительной простоты и распространённости, что вкупе с высокой информативностью этих данных и важностью для здоровья, а следовательно и для жизни человека, ставит перед нами задачу создания методики быстрого и точного диагностирования заболеваний по ЭКГ.

Целью этой работы является исследование возможности детекции аритмии на краткосрочных (10-секундных) ЭКГ в 12-ти отведениях с помощью нейросети.

1 Постановки задач

Начальная задача состояла в скачивании датасета и написании программы, рассчитывающей математическое ожидание частоты сердечных сокращений и её дисперсию. Цель этой задачи была в обучении работе с данными в формате WFDB.

Основной задачей в данной работе является создание и оценка алгоритма диагоностирующего аритмию сердца. Алгоритм представляет собой нейросеть LSTM, которая была обучена с помощью "обучения с учителем"на большом объёме данных ЭКГ здоровых и больных людей. Алгоритм определяет принадлежность записи ЭКГ к классам: норма и аритмия, - путём сложных математических вычислений. Также было необходимо оценить качество этого алгоритма, для чего использовались такие метрики, как accuracy, true positive и true negative.

План моих действий выглядел следующим образом:

— Создать программу для определения ЧСС на 10-секундной записи ЭКГ.

— Развернуть и подключить python- WFDB

— Изучить статьи и видеоматериалы, связанные с ЭКГ, нейросетями, преймущественно LSTM типа, и различными способами оценки качества предсказаний алгоритмов.

— Написать функцию для предобработки данных.

— Написать функции обучения, валидации и оценочных метрик, а также интерфейс, предоставляющий доступ к записям ЭКГ и разметке - диагнозам.

— Написать и обучить модель для бинарной классификации записей.

— Оценить модель на тестовой выборке данных.

2 Массивы данных используемые в проекте

Массив данных 12-lead electrocardiogram database [1] состоит из 45.152 записей ЭКГ более, чем 10.000 пациентов. Версия датасета - 1.0.0.

Каждая запись базы данных является 10-секундной записью ЭКГ в 12-ти отведениях с частотой дискретизации 500 Гц, соответственно запись имеет размер 12 на 5000. Единицей измерения является микровольт, верхний предел значений - 32.767, нижний - -32,768. Показатели выходящие за эти ограничения помечались как nan (not a number - "не число").

Каждая запись представлена 2 файлами:

— текстовый файл с расширением .hea, содержащий информацию, включающую в себя конфигурацию отведений, возраст и пол пациентов, а также SNOMED CT, соответствующий диагнозу, поставленному врачом;

— бинарный файл с расширением .dat, содержащий оцифрованные данные ЭКГ;

Также в датасете присутствуют файлы:

— текстовый файл LICENSE с расширением .txt, содержащий лицензионные права;

— текстовый файл ConditionNames_SNOMED-CT с расширением .csv, сопоставляющий акронимы их расшифровкам и SNOMED CT кодам;

— текстовый файл RECORDS, содержащий пути к папкам с записями;

— текстовый файл SHA256SUMS с расширением .txt;

3 Обзор литературы

В работе [2] были приведены такие методики визуального представления и обработки записей ЭКГ, как обработка записи при помощи корреляционной функции с синусовым фильтром и функции поиска пиков. Схожие способы использовались в данной работе для предобработки данных ЭКГ, а именно обрезания записи в случае неполного попадания удара сердца на запись или если первый (последний) удар находились слишком далеко от начала (конца) записи.

В работе [3] была натренирована модель глубокого обучения, которая получила возможность обнаружения случаев внезапной внебольничной сердечной смерти с точностью, превышающей традиционный не нейросетевой метод, что свидетельствует о перспективе применения нейросетевых технологий для анализа ЭКГ и их большей точности по сравнению с методами, использовавшимися в прошлом. Это также позволит частично снять с врачей нагрузку по анализу ЭКГ.

В работе [4] было проведено исследование применения нейронной сети на основе LSTM для обнаружения аритмии. В статье тестировалось различное количество скрытых слоёв (1-3) и применение адаптивного learning rate. В этом исследования авторам удолось достичь наиболее высокой точности диагностирования ЭКГ при параметрах количества скрытых слоёв - 3 и при использовании алгоритма скользящего окна обновления градиентов AdaDelta [5]. В данной же работе будет исследоваться подход с одним внутренним слоём и без алгоритма AdaDelta.

В работе [6] было представлено несколько типов нейронных сетей и различных способов их изменений и улучшений для различных задач. Также в этой работе была продемонстирована работа и преймущество над RNN и GRU сетями модели на основе LSTM. Эта сеть отлично справилась с определением части речи слов в предложении на английском языке. В курсе говорится, что LSTM предназначена для работы на распределённых во времени данных благодаря встроенным в неё затворам, помогающим ей "вспоминать информацию из прошлого". Основываясь на этом курсе, LSTM была выбрана для анализа ЭКГ в данном иссле-

дованнии. Также на основе этого курса и источника [7] были выбраны метрики для оценки качества предсказаний модели.

4 Описание решения

4.1 Решение промежуточных задач

4.1.1 Вычисление частоты сердечных сокращений

Алгоритм вычисления частоты сердечных сокращений состоит из пунктов:

— Загрузка данных - одной записи ЭКГ, выбор отведения с наиболее чёткой структурой и ярко выраженными QRS комплексами.

— Создание синусового фильтра и применения его к последовательности с помощью корреляционной функции для увеличения амплитуды в пиках.

— Нахождение выделенных пиков. 4.1

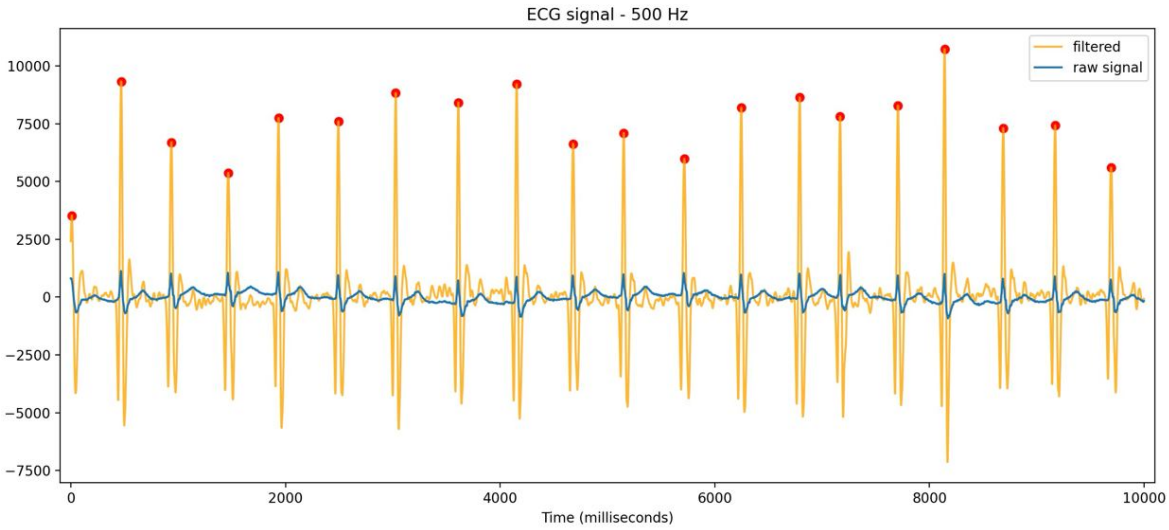

Рисунок 4.1 — Ecg, transformed ecg, peaks

— Вычисление среднего значения расстояния между пиками (зубцами R на ЭКГ) и подсчёт матожидания квадрата отклонения от матожидания частоты.

Было выбрано 11 отведение - V5. Характеристики синусового фильтра - равномерно распределённые 30 точек со значениями синуса на промежутке от -0.5π до 1.5π) Собственная корреляционная функция является обёрткой для встроенной в numpy функции correlate. Обёртка позволяет достигнуть бОльшей линейности в выходных данных. Так,

если подать на вход немного уменьшающуюся последовательность, то из-за особенностей функции correlate на небольшом расстоянии от начала записи образовывался новый пик, мешающий правильной работе последующего алгоритма. Функция-обёртка my_correlate практически не допускает таких образований новых пиков из-за того, что использует для корреляции не начальную последовательность, а немного увеличенную её версию засчёт дополнения в начале и конце крайними значениями. Размер возвращаемой последовательности равен размеру входящей. Нахождение пиков было произведено с помощью функции из scipy.signal - find_peaks. Её параметры были подобраны в результате тестирования для наилучшего соответствия задаче: минимальная высота пиков - 2500, минимальное расстояние между пиками - 120, что соответствует 250 ударам сердца в минуту при 500 Гц. Меньшие значения интервала брать не имеет смысла, поскольку вероятность, что у пациента на записи будет пульс больше, чем 250 ударов в минуту, - незначительна. Матожидание ЧСС и её дисперсия вычислены в соответствии с математическими формулами для данных понятий.

4.2 Предлагаемый метод предсказания диагноза по 12-канальному сигналу ЭКГ

4.2.1 Подготовка данных

— Загрузка всего датасета и итерация по нему.

— Применение функции обрезки и заполнения утрат cut_n_fill.

— Нормализация данных.

— При получении излишне странной записи из предыдущих 2 пунктов, запись удалялась из используемого датасета.

— Расспределение на тренировочную и валидационую часть и сохранение этих массивов данных.

При загрузке использовался пакет WFDB. Функция cut_n_fill заключалась в нахождении пиков на записи. Если крайние из них находятся слишком близко к краю, а именно на расстоянии меньшем оптимального расстояния от края - половина от минимума из 500 и

расстояния между крайним и ближайшим к нему пиком, то в таком случае запись обрезается по этому пику. Следующий шаг функции - обрезка записи таким образом, чтобы крайние пики (возможно, новые крайние пики) находились на оптимальном расстоянии до края записи. Поскольку в дальнейшем нейросети понадобятся для работы записи одинаковой длины, в начале и в конце записи удалённые отрезки заполняются нулями до начальной длины. 4.2 Нормализация данных

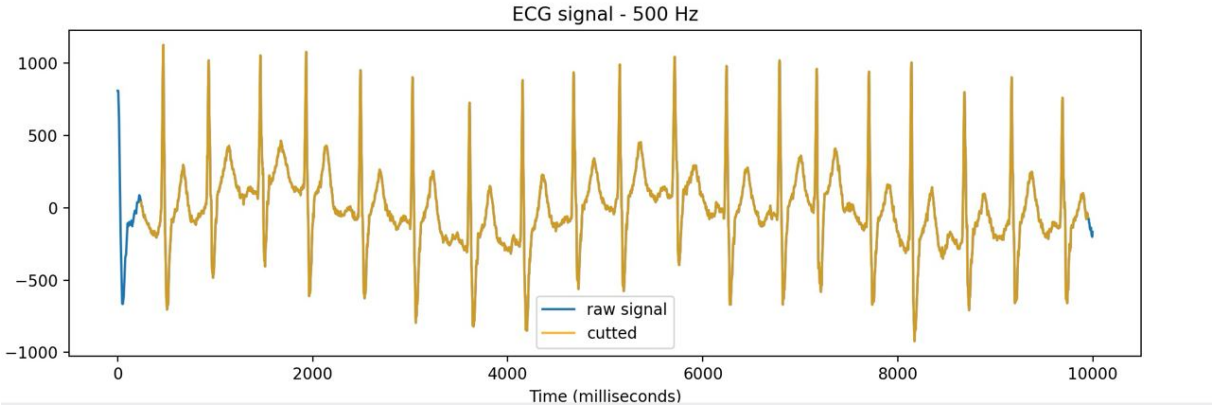

Рисунок 4.2 — Raw ecg, cutted signal before filling with zeroes

выполняется по следующему алгоритму: во-первых, все значения, равные nan, заменяются в записи на предшествующие им, во-вторых, для каждой записи, для каждого отведения в отдельности проводится минимаксная нормализация к диапазону [0, 1].

Было: рисунок 4.3

Стало: рисунок 4.4

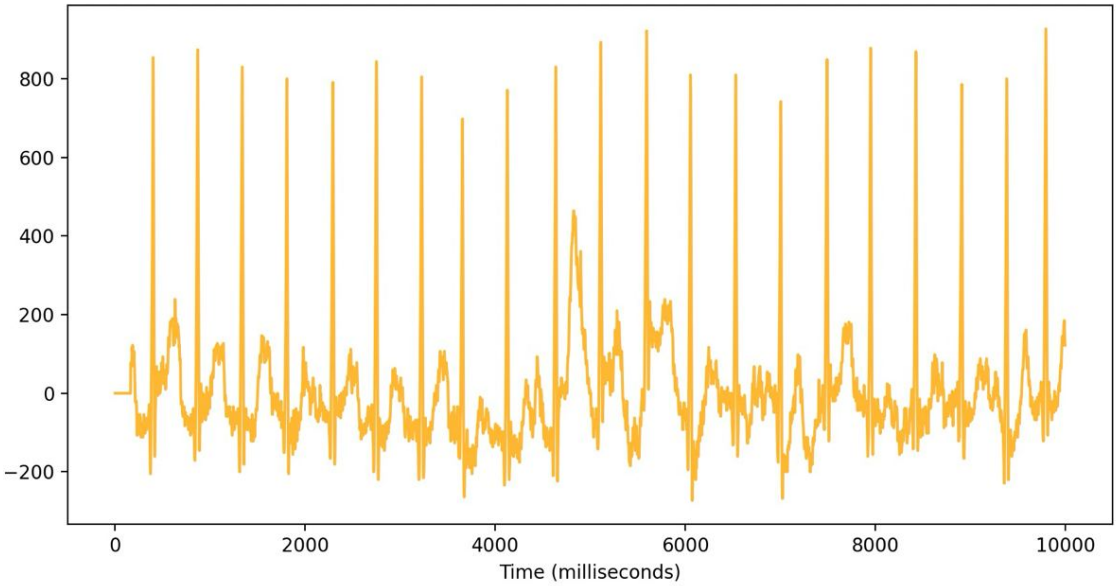

Рисунок 4.3 — Ecg before normalization

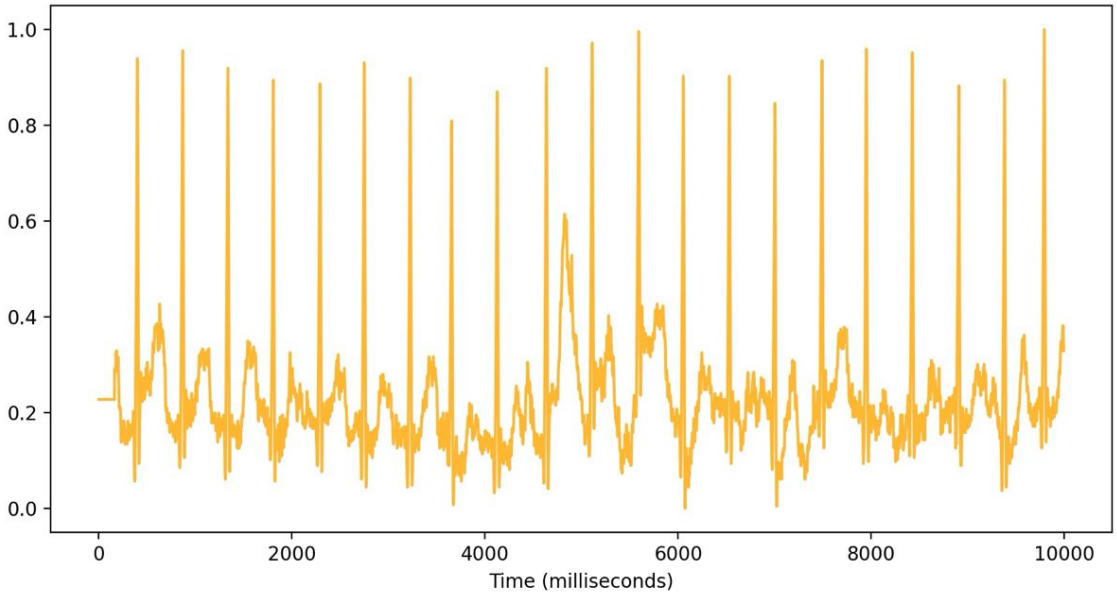

Рисунок 4.4 — Ecg after normalization

Графики идентичны за исключением вертикальной шкалы.

Излишне странной записью считается такая запись, в процессе обработки которой произошли такие события: разница между минимальным и максимальным значением оказалась близка к нулю, или не было обнару-

жено пиков, или произошла ошибка во время обрезания и применения корреляционной функции.

4.2.2 Описание модели

Для диагностирования аритмии была выбрана LSTM модель нейросети из-за её высокой эффективности при работе с распределёнными во времени данными.
Характеристики модели:

— Входная размерность вектора последовательности - 12.

— Выходная размерность предсказания - 1.

— Размерность скрытого слоя - 128.

— Количество скрытых слоёв - 1.

— Остальные параметры стандартные для модели из библиотеки torch.nn.LSTM [8].

— Инициализация параметров модели - Xavier initialization [9].

— После применения lstm присутствует полносвязный слой, превращающий выходные значения [batch size, hid dim] в predictions [batch size, output dim].

4.2.3 Описание сопутствующих компонент

Характеристики:

— Во время тренировки число, инициализирующее генератор случайных чисел, было зафиксировано и равно 1234

— Для ускорения процесса тренировки происходили на GPU.

— Датасет унаследован от torch.utils.data.Dataset [10]

— Датасет содержит данные о длине массива данных и о шаблоне пути к файлам с записями ЭКГ и диагнозами

— Загрузчик создан на основе torch.utils.data.DataLoader [10]

— параметры загрузчика, размер батча, перетасовка, pin-memory (для увеличения скорости переноса данных на GPU) и количество потоков загрузки - 100, "да"("нет для валидационной выборки), "да 2 соответственно.

— Оптимизатор - torch.optim.AdamW [11] с параметрами learning rate и weight_decay - 1e-3 и 0 соответственно.

— Для подсчёта функции потерь использовалась torch.nn.BCEWithLogitsLoss() [12] - более стабильная версия объединения сигмоиды и BCELoss (Binary Cross Entropy).

— Используемые метрики качества предсказаний - самописные accuracy - общая точность, true positive - верно определённая аритмия и true negative - верно определённая норма.

— Во время обучения нейросети вёлся подсчёт времени выполнения программой одной эпохи обучения.

— Процесс обучения происходил как на локальной машине, так и на серверах облачного сервиса Google colaboratory [13] с подключёнными дополнительными вычислительными мощностями.

Необходимость написания этих метрик была обусловлена незнанием, что существуют библиотеки с уже определёнными функциями, а в последствии тем, что для использования встроенных в библиотеки метрик были необходимы операции по переносу данных обратно на CPU и превращения тензора в numpy массив, что привносило некоторое замедление в обучение, а практических выгод, таких как ускорение работы, уменьшение объёма кода и других, обнаружено не было.
Интересный факт: за время выполнения этой работы удалось оптимизировать процесс обучения с 76 минут на одну эпоху до времени меньшего, чем секунда, при выполнении того же объёма действий.

4.3 Результаты

4.3.1 Код

Весь код находится по ссылке: Код

4.3.2 Вычисление ЧСС в Python

ЧСС вычислено.

4.3.3 Диагностирование аритмии на ЭКГ при помощи нейросетевой модели типа LSTM

В процессе обучения, состоящего из множества вариаций параметров, машин, на которых оно происходило, и эпох были подобраны параметры, при которых удалось получить наилучшие значения оценочных метрик.

Среда выполнения - Google colab L4 GPU.

Значения метрик представлены на графике: 4.5

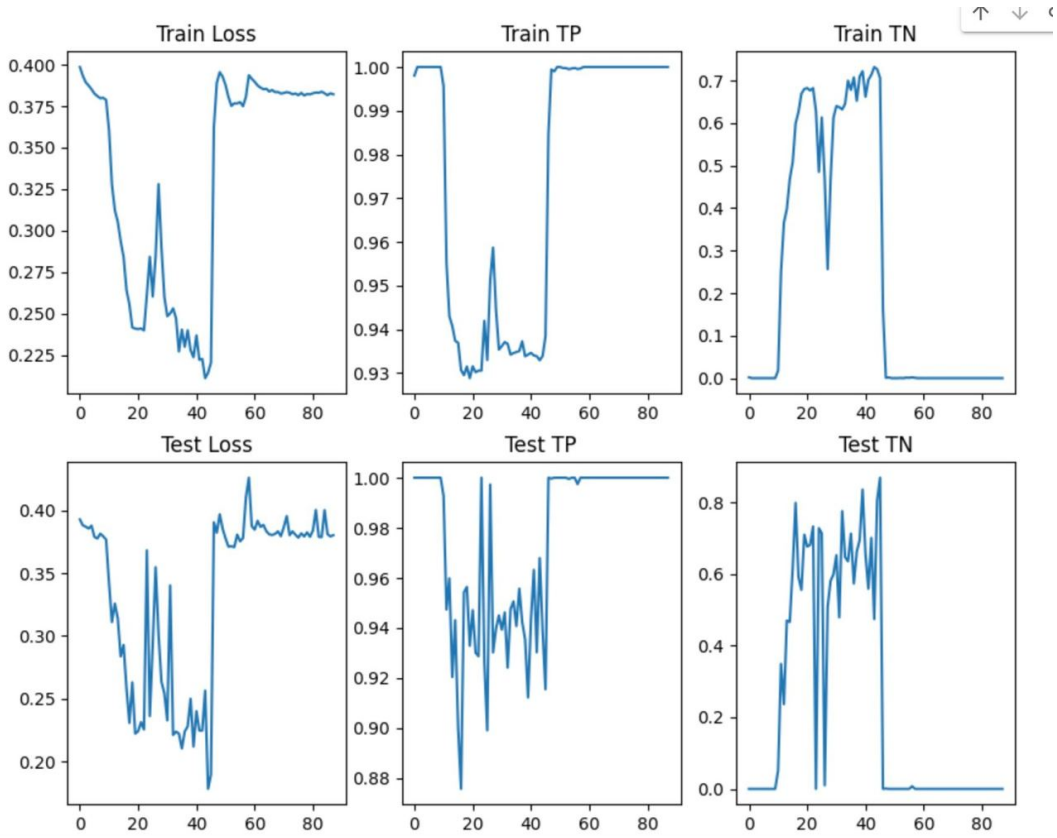

Рисунок 4.5 — Loss, TP, TN

Точные значения метрик на тестовой выборке:

— TP - 91.8% - вероятность верного определения аритмии, при условии её наличия.

— TN - 83.8% - вероятность верного определения нормы, при условии здоровья пациента.

Такой перекос в диагностировании в пользу бОльшего качества определения аритмии связан с данными, на которых проводилось обучение. В исходном датасете находилось сильно больше записей ЭКГ с аритмией, нежели ЭКГ, не отклонявшихся от нормы. Примерное соотношение - 1 к 5.25, где 1 - количество записей ЭКГ с нормой, а 5.25 - с аритмией. Как видно на графике, в какой-то момент обучения нейросеть начинает выдавать константное предсказание, что также свидетельствует о несбалансированности данных.

К сожалению, файлы с состоянием модели, показывающей такие характеристики на метриках качества, были утеряны и воспроизвести их не удалось. Поэтому реально существующий результат таков:

— TP - 89.4% - вероятность верного определения аритмии, при условии её наличия.

— TN - 77.2% - вероятность верного определения нормы, при условии здоровья пациента.

4.4 Выводы

Удалось написать нейросеть на основе LSTM, диагностирующую сердечную аритмию на 500 Гц ЭКГ в 12-ти каналах с достаточно высокой точностью, что говорит о больших перспективах использования нейросетей не только для анализа ЭКГ, но и во всей сфере медицинской диагностики.

Список использованных источников

1. *Zheng, Jianwei.* A large scale 12-lead electrocardiogram database for arrhythmia study / Jianwei Zheng, Hangyuan Guo, Huimin Chu. — 2022. — 08. https://www.physionet.org/content/ecg-arrhythmia/1.0.0/#files-panel.

2. *Kulas, Bartek.* Working with ECG — Heart Rate data, on Python by Bartek Kulas Medium / Bartek Kulas // *Medium.* — 2023. — no. 11. https://www.google.com/url?sa=t&source=web&rct=j&opi=89978449&url=https://bartek-kulas.medium.com/working-with-ecg-heart-rate-data-on-python-7a45fa880d48&ved=2ahUKEwjXtrLo7t6FAxV0JhAIHfggCLYQFnoECBQQAQ&usg=AOvVaw0cFZZR7ySRp8BzHcTTWvcD.

3. *Lauri Holmstrom Harpriya Chugh, Kotoka Nakamura Ziana Bhanji Madison Seifer Audrey Uy-Evanado Kyndaron Reinier David Ouyang Sumeet S. Chugh.* An ECG-based artificial intelligence model for assessment of sudden cardiac death risk / Kotoka Nakamura Ziana Bhanji Madison Seifer Audrey Uy-Evanado Kyndaron Reinier David Ouyang Sumeet S. Chugh Lauri Holmstrom, Harpriya Chugh // *Communications Medicine.* — 2024. — 02. https://www.nature.com/articles/s43856-024-00451-9.

4. *Hilmy Assodiky Iwan SyarifI, Tessy Badriyah.* Arrhythmia Classification Using Long Short-Term Memory with Adaptive Learning Rate / Tessy Badriyah Hilmy Assodiky, Iwan SyarifI // *ResearchGate.* — 2018. — 07. https://www.researchgate.net/publication/326303899_Arrhythmia_Classification_Using_Long_Short-Term_Memory_with_Adaptive_Learning_Rate.

5. *Zeiler, Matthew D.* ADADELTA: An Adaptive Learning Rate Method / Matthew D. Zeiler // *arXiv:1212.5701.* — 2012. — 12. https://doi.org/10.48550/arXiv.1212.5701.

6. *Konyagin, Egor.* DL-EGOR 2022 / Egor Konyagin. — 2022. https://disk.yandex.ru/d/6BkVj6Dy5XC0PQ.

7. Metrics and scoring: quantifying the quality of predictions. https://scikit-learn.org/stable/modules/model_evaluation.html.

8. LSTM. https://pytorch-org.translate.goog/docs/stable/generated/torch.nn.LSTM.html?_x_tr_sl=en&_x_tr_tl=ru&_x_tr_hl=ru&_x_tr_pto=sc.

9. Tutorial 4: Optimization and Initialization. https://uvadlc-notebooks.readthedocs.io/en/latest/tutorial_notebooks/tutorial4/Optimization_and_Initialization.html.

10. torch.utils.data. https://pytorch.org/docs/stable/data.html.

11. Guide 3: Debugging in PyTorch. https://uvadlc-notebooks.readthedocs.io/en/latest/tutorial_notebooks/guide3/Debugging_PyTorch.html.

12. BCEWithLogitsLoss. https://pytorch.org/docs/stable/generated/torch.nn.BCEWithLogitsLoss.html.

13. Google Colaboratory. https://colab.google.