

## **Анализ ЭКГ-сигналов для диагностики сердечных патологий**



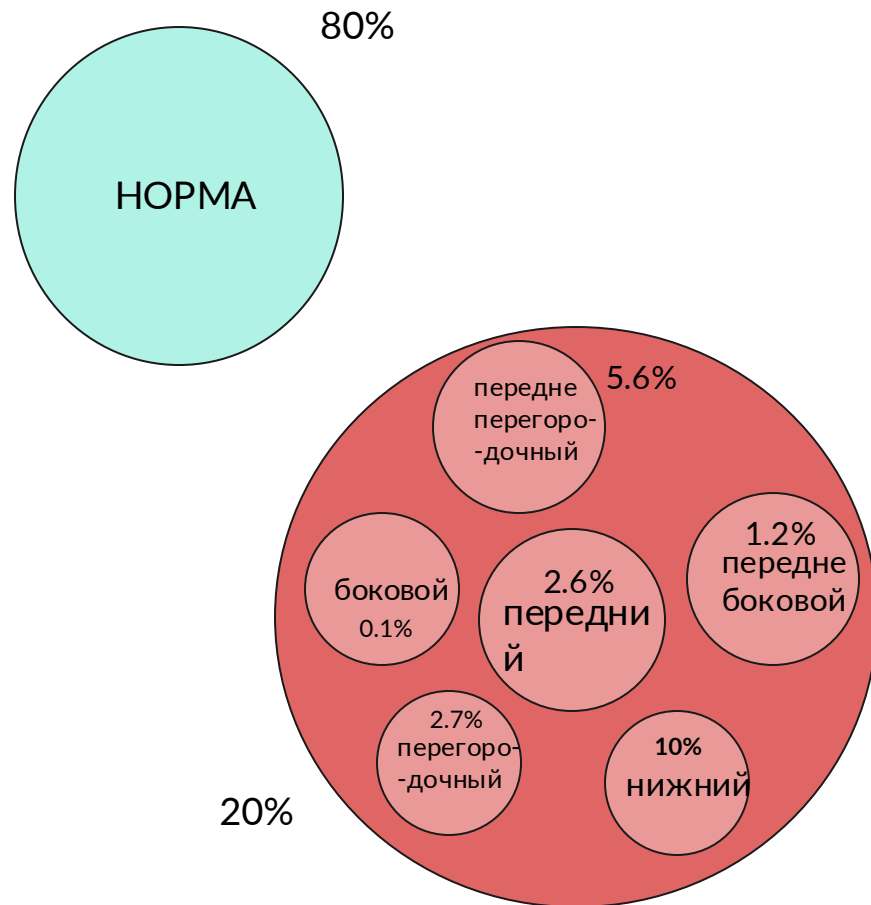
**MINIONS**

**Владивосток 2023**

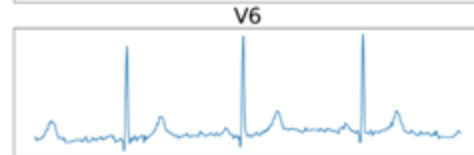
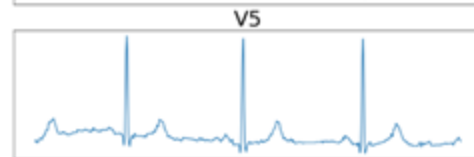
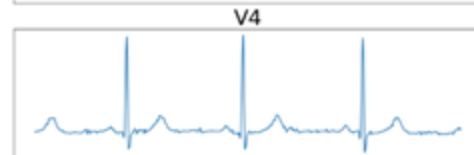
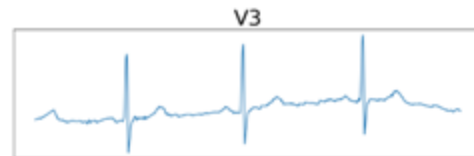
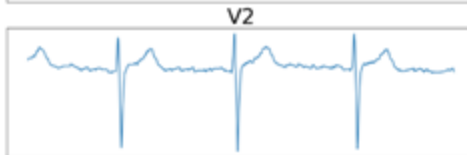
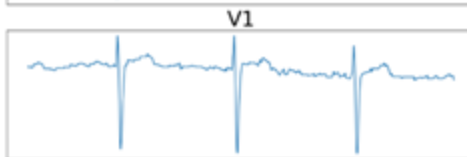
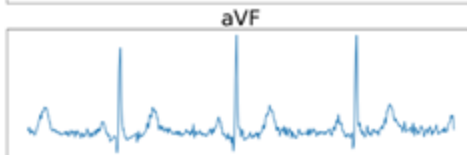
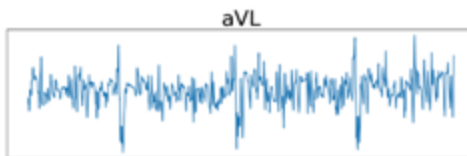
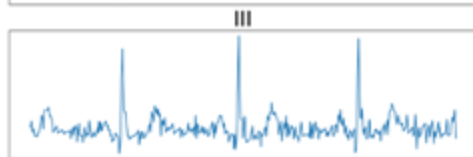
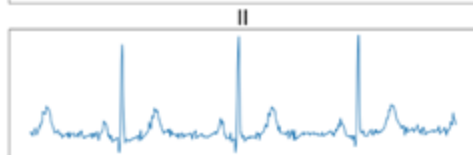
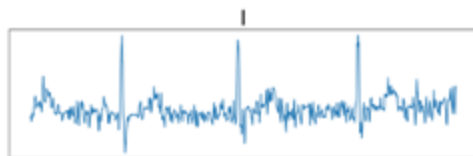
## Описание задачи.

Определение отсутствия или локализации

инфаркта миокарда у пациента, на основе результатов его ЭКГ и некоторых физиологических данных (возраст, рост, вес и пол)



# Как выглядит ЭКГ-сигнал





# Предобработка сигнала

1. `ecg_clean` (библиотека `neurokit2`, функция нормализует сигнал относительно нулевой горизонтальной линии и убирает часть шума)
2. Скользящее среднее ([1, 2, 10, 2, 6] -> [1, 6.5, 7, 9, 6])
3. Скользящая медиана ([1, 2, 10, 2, 6] -> [1, 2, 2, 6, 6])
4. Обрезание сигнала после функции (1) (иногда получались артефакты на концах сигналов после использования этой функции)

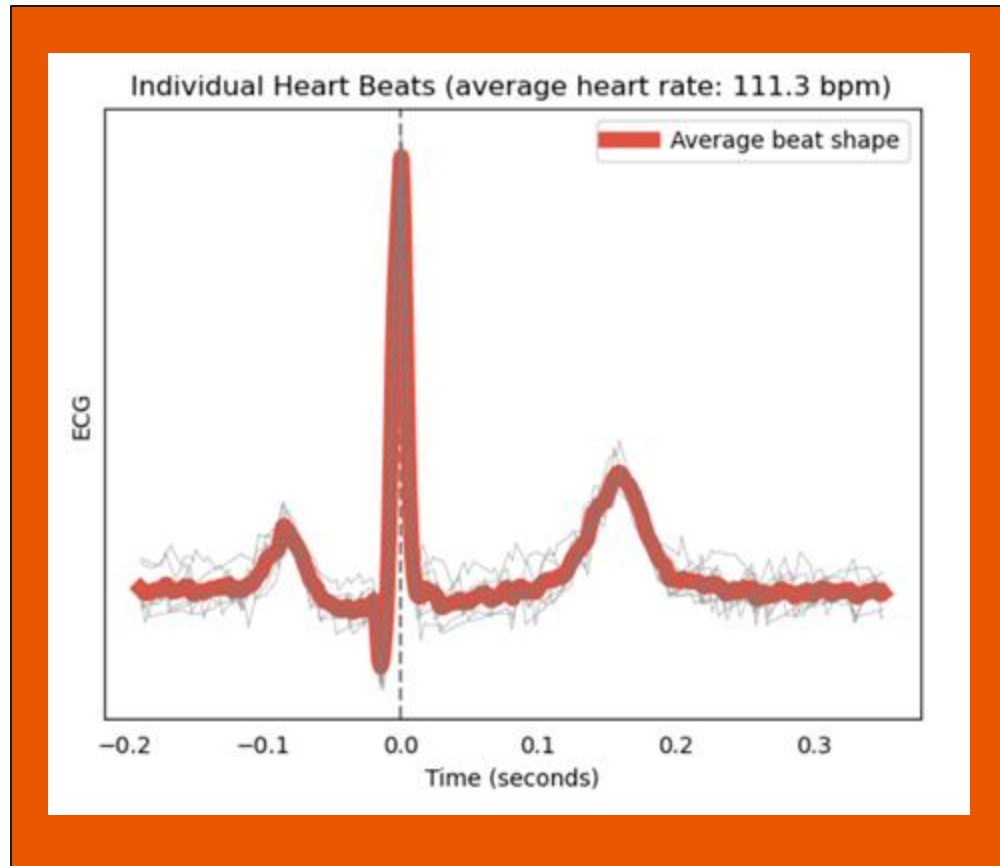
## Наши первые попытки

Сверточная нейронная сеть  
(Conv1D) на сегментах экг сигнала

Р-пик — Т-пик

(Разделяли при помощи функции `ecg_segments` библиотеки `neurokit2`, она нормализовывала все сегменты так, чтобы все **Р-пики** находились в одном месте)

Best F1 - (0.36)



## Обучение на полном сигнале (Conv1D + FullyConnected)

FullSignal_model_норма	0.7383	0.7516	14	Sun Oct 15, 16:27:54	1m 24s
FullSignal_model_перегородочный	0.6327	0.6057	84	Sun Oct 15, 15:55:26	1m 27s
FullSignal_model_передне-боковой	0.8488	0.8889	74	Sun Oct 15, 16:16:20	1m 15s
FullSignal_model_передне-перегородочный	0.8868	0.91	84	Sun Oct 15, 16:04:21	1m 28s
FullSignal_model_передний	0.857	0.974	99	Sun Oct 15, 15:43:59	1m 42s

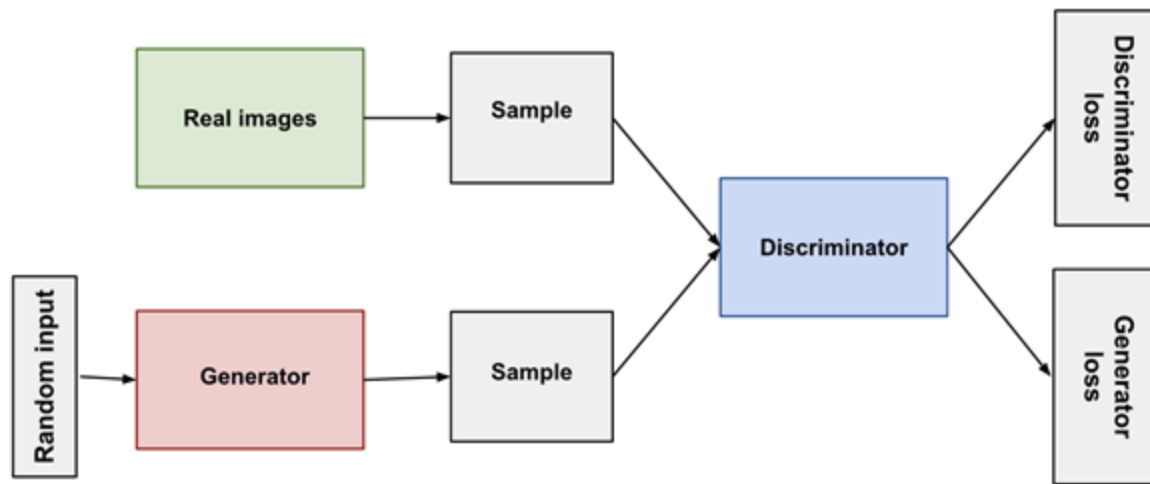
T  
R  
A  
I  
N

FullSignal_model_норма	0.7741	0.8208	14	Sun Oct 15, 16:27:55	1m 23s
FullSignal_model_перегородочный	0.4477	0.5143	84	Sun Oct 15, 15:55:26	1m 27s
FullSignal_model_передне-боковой	0.5673	0.6667	74	Sun Oct 15, 16:16:21	1m 15s
FullSignal_model_передне-перегородочный	0.581	0.601	84	Sun Oct 15, 16:04:21	1m 27s
FullSignal_model_передний	0.6442	0.7048	95	Sun Oct 15, 15:43:55	1m 38s

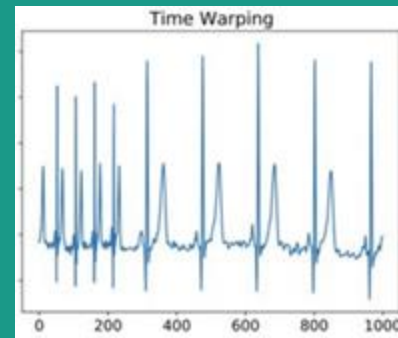
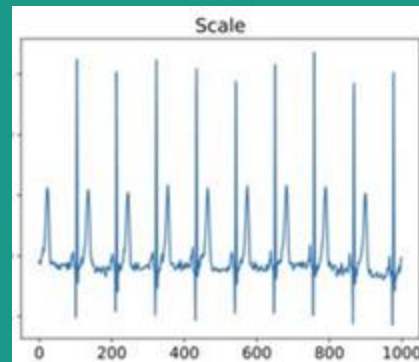
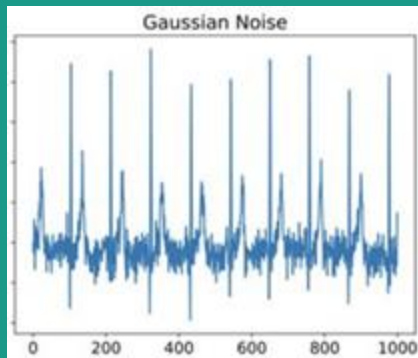
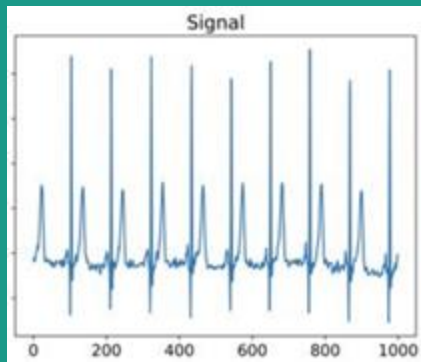
V  
A  
L

## DcGAN (Синтетические данные ЭКГ) (Неудача)

Пытались сделать генерацию ЭКГ-сигнала для каждого отведения в отдельности, но получались некорректные результаты, вследствие дефицита данных

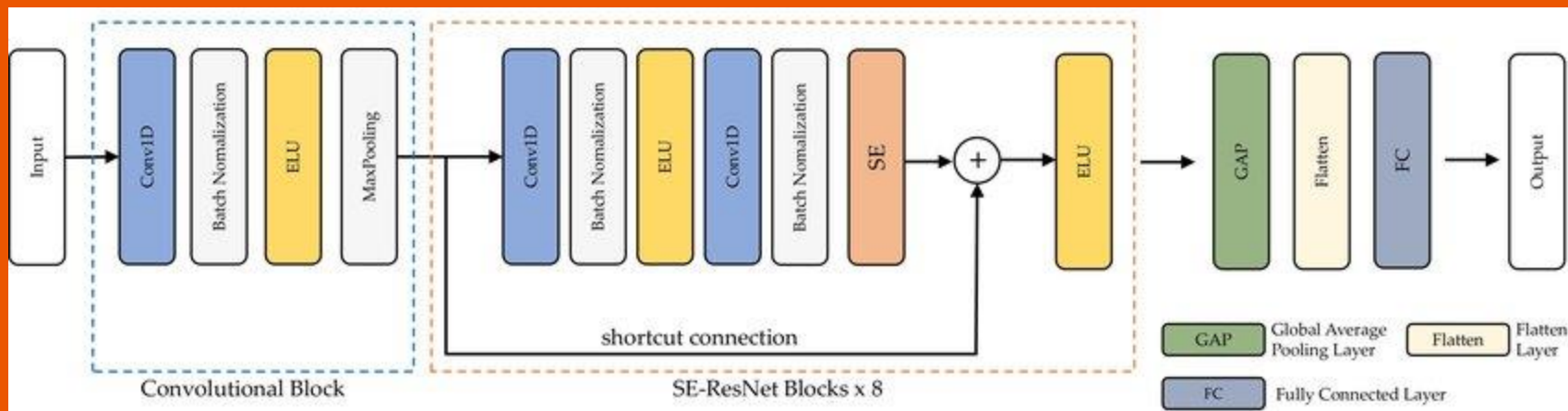


## Аугментация (Успех) (TTA)

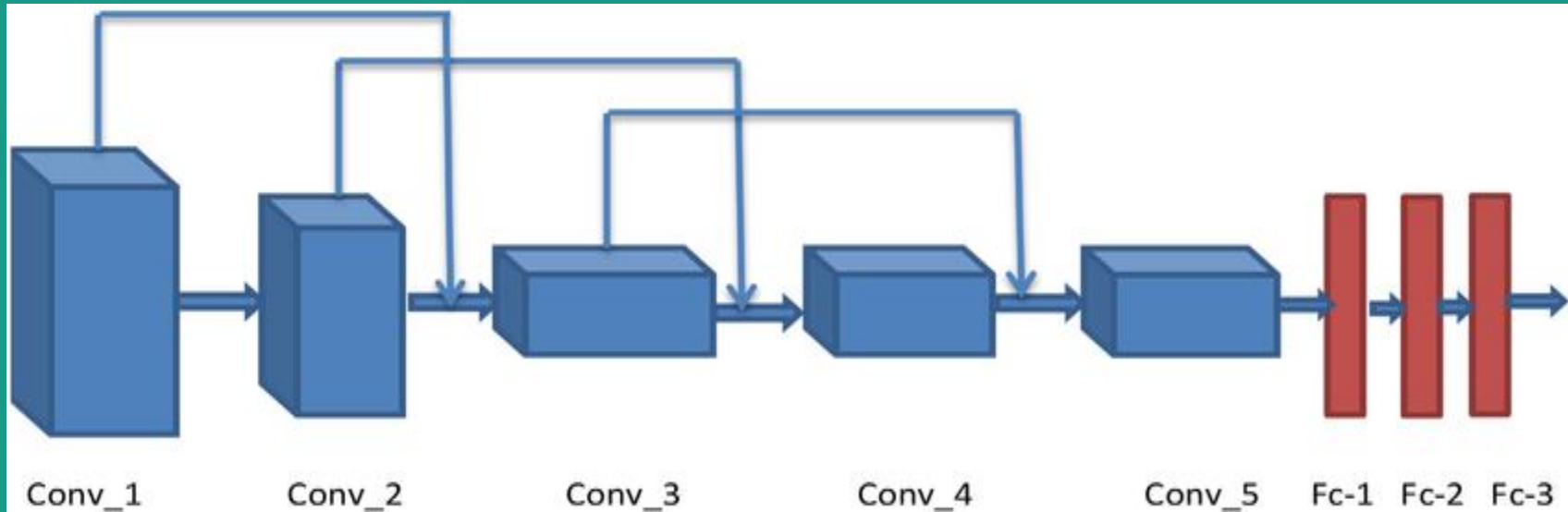




## Новые сверточные сети (SeResnet).

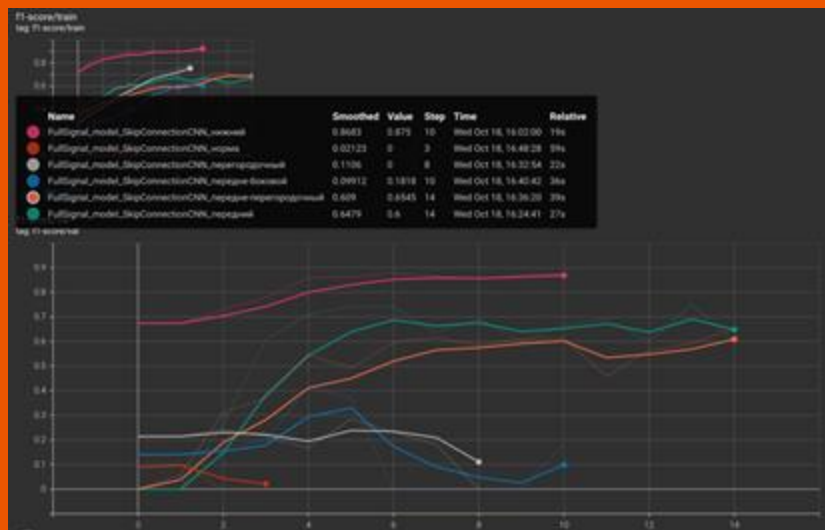


## Новые сверточные сети (SkipConnection)



# Результаты новых сверточных сетей

## ScipConnection



## SeResNet





# Генерация числовых фич на основе сигнала ЭКГ

Генерировали фичи на основе длины интервала R-R (HRV)

Used:

1. neurokit2 ecg\_rate
2. neurokit2 hrv\_time, hrv\_frequency
3. neurokit2 entropy\_sample

(Генерировали только для первого отведения так как это требует много вычислительной мощности для приемлимой скорости работы, а одного отведения достаточно, для того, чтобы делать выводы)

# Объединение результатов всех моделей. CatBoost.

Генерация датасета для CatBoostClassifier:

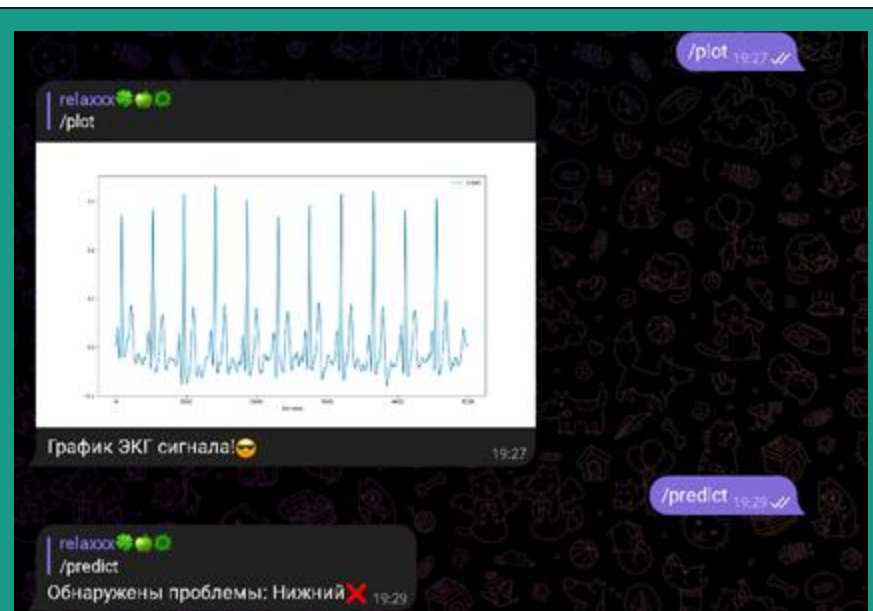
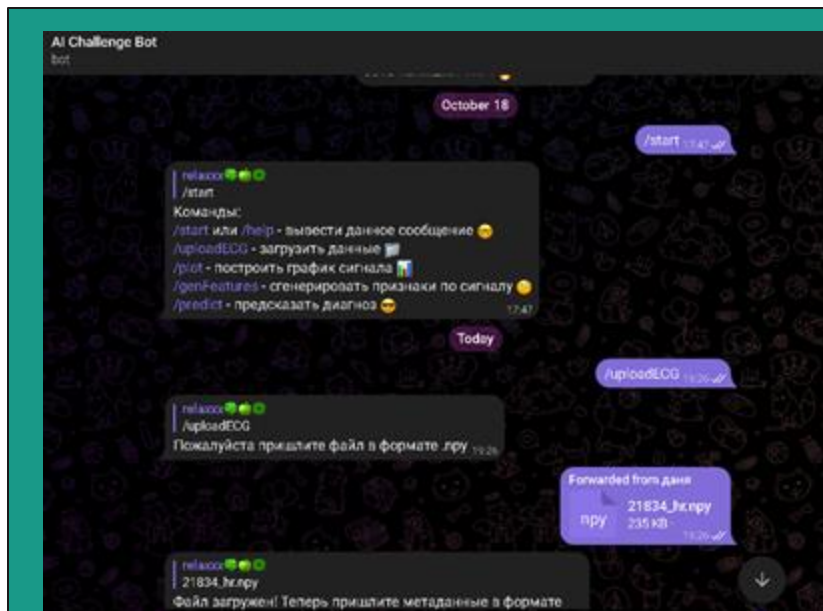
1. Предпоследние слои всех сверточных сетей (брали не все числа со слоя а какую то часть, чтобы избежать переобучения)
2. Числовые фичи ЭКГ-сигнала
3. МетаДанные пациента

(Во время обучения использовали маленькую глубину деревьев, чтобы уменьшить переобучение)



CatBoost

# Telegram-Bot. (@minions\_AI\_Challenge\_Bot)





## Возможности улучшения.

1. Генерация синтетических данных
2. Увеличение вариативности аугментации
3. Расширенный подбор гиперпараметров для сетей и градиентного бустинга
4. Тестирование разных новых архитектур сетей или изменение имеющихся (например: LSTM)
5. Тестирование обучения на неполном мульти-лейбле (например: передний; передне-боковой; боковой)
6. Другие градиентные бустинги (xgboost, lgbm ...)
7. Объединение нескольких отведений в одно, используя какую-нибудь функцию (например сложение)
8. В случае использования мультитейбла - микс



## Итог.

Мы смогли сделать алгоритм по определению локализации инфаркта, используя ЭКГ-сигнал пациента, а также его метаданные.

Итоговое решение включает в себя множество сверточных нейронных сетей, которые были обучены по методу OneVsRest, определение некоторых числовых фич на основе ЭКГ-сигнала и градиентный бустинг, который объединяет все эти методы и выдает результат.

Итогом нашего решения стал инференс, в который можно загрузить датасет и получить DataFrame ответ, как в условии задачи, а также telegram bot, в который можно загрузить один ЭКГ-сигнал и получить диагноз.