Прежде чем представить наш проект, хотелось бы поблагодарить организаторов и особенно экспертов за возможность поработать над столь интересной практической задачей, подготовленный реальный датасет, мы с большим увлечением думали над задачей и решением.

## 2

Мы – команда начинающих дата-сайентистов. Нас объединяет то, что мы выпускники программы Data Science МГТУ им. Н. Э. Баумана и решили попробовать силы не на учебных, а на реальных задачах.

Мы выбрали задачу 15 потому, что ее фокус не на разработке, а на анализе данных.

## 3

Познакомившись с задачей ближе и осознав ее масштаб, мы решили сконцентрироваться на разделении подходов для предсказания наличия аварий типа M1 в пропущенных интервалах (подзадача №1) и предсказания неисправностей типа M3 в остальных интервалах (подзадача №2) и построении отдельных моделей для двух подзадач.

## 4

В рамках первой задачи наша гипотеза состоит в том, что, незадолго до наступления аварий, на графиках показателей соответствующей машины будут появляться характерные последовательности (паттерны), которые сможет распознать одномерная сверточная нейронная сеть.

Для обучения CNN в задаче 1, мы создали специальный датасет. Создавали его следующим образом: исходный файл X_train мы разбили на сэмплы: по столбцам - на 6 столбцов, каждый из которых соответствует одной машине, и по строкам - на 10-минутные интервалы, следующие подряд друг за другом. Таким образом, в каждый сэмпл попадает 16 столбцов и 61 строка исходного датасета.

## 5

Каждому сэмплу присваивается одна из трех меток класса:

1 - если сэмпл содержит аварию или часть аварии M1

2 - если сэмпл попадает в окно определенной длины перед аварией M1 (мы использовали длину окна в 3 часа)

0 - в противном случае

Информация об авариях бралась из файла messages.xlsx, и поэтому файл y_train.parquet вообще не использовался в данном подходе.

Таким образом, весь исходный X_train оказывается разбит на сетку из прямоугольных сэмплов, и каждому сэмплу присвоена одна метка класса. Сэмплов с меткой 0 на порядки больше, чем сэмплов с метками 1 и 2, т.к. аварий M1 мало и они имеют малую протяженность. Поэтому в датасет для обучения CNN мы включили:

1) все сэмплы с метками 1 и 2

2) некоторое количество сэмплов с меткой 0, равномерно распределенных по всему X_train с тем, чтобы получить сбалансированный датасет.

В итоге размер датасета получился около 10,000 пар "сэмпл + метка".

## 6

При обучении CNN на валидационной выборке мы получали цифры F1 от 0.83 до 0.92, accuracy около 0.95.

## 7

Для получения предсказаний о наличии или отсутствии аварий в выброшенном интервале, на вход обученной CNN подавались аналогичные сэмплы из файла X_test, попадающие в 3-х часовое окно перед авариями M1 - суммарно 18 сэмплов на каждый пропуск и на каждый эксгаустер. Таким образом, в качестве предсказания на каждый пропуск и на каждый эксгаустер мы получали 18 меток принадлежности к одному из трех классов, из которых методом голосования определяется наличие или отсутствие аварии в пропуске. (Метки суммируются и сумма сравнивается с пороговым значением, которое можно варьировать, изменяя чувствительность метода.)

Данный подход обладает тем ограничением, что не предусматривает определения конкретного технического места.

## 8

В рамках второй задачи определения неисправностей типа М3 применялся подход, который как считается в последние годы обладает высоким потенциалом – сверточные нейронные сети, дополненные слоями долгой краткосрочной памяти.

## 9

Для этой задачи датасет также формировался специальным образом. Первоначально для обучения отбирались данные для одного эксгаустера. Далее после установления периодов с авариями типа М1, они исключаются или подбираются обучающие данные, не содержащие М1 для всех таргетов конкретного эксгаустера.

Из данных исключаются интервалы, не содержащие информации о дате устранения неисправности для определенного техместа. Таким образом, в большинстве случаев выборка ограничивалась начальным отрезком времени, включающем участки штатной работы и два участка неисправностей.

## 10

После этого отбиралось разное количество оптимальных параметров с датчиков для прогнозирования, таргет для анализа работы модели брался один.

## 11

Данные преобразовывались в последовательность в виде скользящего окна прогнозирования. Каждому окну массива X сопоставлялась одна метка y, представленная в максимальном количестве в данном окне. Данный подход удобен для последующей реализации прогнозирования в режиме реального времени.

## 12

Данные в виде набора окон подаются в модель нейронной сети для классификации. На выходе можно поучать предсказание как для оконных данных, так и восстанавливать метки для исходного поданного массива для обработки файла X_test.

## 13

На слайде представлена обобщенная архитектура нейросети для данной задачи. В последствии возможно расширение или изменение архитектуры для оценки длительности времени до наступления неисправности.

## 14

В результате разработки модели по подзадаче 1 точность accuracy классификации на обучающей выборке составила примерно 95 %, F-мера гармоническое среднее от 83 до 91% в зависимости от параметров.

Подзадача № 2 находится в разработке, по ней необходимо дальнейшее развитие модели, а также подбор влияющих параметров и релевантных сбалансированных данных.

## Благодарю за внимание

Мы как команда намерены продолжить работу над данной задачей и хотели бы оставаться на связи с эскпертами.