Выявление аномалий в работе механизмов методами машинного обучения

© А.Г. Дьяконов[1] © А.М. Головина[2]

[1] Московский государственный университет имени М. В. Ломоносова,
[2] Московский государственный технический университет имени Н. Э. Баумана,
Москва, Россия

djakonov@mail.ru nastya_gm@mail.ru

Аннотация. Описано исследование по выявлению поломок механизмов методами машинного обучения. Задача сведена к классической задаче машинного обучения без учителя: детектированию аномалий. Сделан обзор современных подходов к решению этой задачи, все они были апробированы на реальных данных. В результате построен алгоритм, который детектирует поломки сложных механизмов в режиме online. В силу своей специфики он также детектирует любое аномальное поведение: нештатный запуск, работу в опасном режиме, неверную эксплуатацию и т. п.

Ключевые слова: большие данные, анализ данных, выбросы, аномалии, поломки.

Anomaly Detection in Mechanisms Using Machine Learning

© A.G. D'yakonov[1] © A.M. Golovina[2]

[1] Lomonosov Moscow State University,
[2] Bauman Moscow State Technical University,
Moscow, Russia

djakonov@mail.ru nastya_gm@mail.ru

Abstract. The research on breakdown detection in mechanisms using machine learning methods is described. The problem is reduced to anomaly detection. The review of modern approaches to anomaly detection is made, all of them have been approved on real data. As a result, the algorithm for online breakdown detection in complicated mechanisms is constructed. By its nature the algorithm also detects any abnormal behavior: emergency start, work in the dangerous mode, incorrect operation, etc.

Keywords: big data, data mining, outliers, anomaly, breakdown.

Введение

В настоящее время стремительно развиваются приложения методов машинного обучения (machine learning) и анализа данных (data mining), что вызвано, с одной стороны, появлением универсальных и практически полезных моделей алгоритмов, например, бустинга (в его современной реализации [7]) и свёрточных нейронных сетей [17], с другой стороны, определённой тенденцией в бизнесе и индустрии улучшать доходность и качество услуг с помощью современных IT-технологий. Отметим, что такая тенденция появилась в последнее десятилетие прежде всего за счёт миниатюризации и удешевления устройств хранения и обработки данных, датификации процессов компаний ([18], постоянного логирования, быстрого перевода в удобный для обработки формат). Такая тенденция привела к появлению специального термина – «Большие данные» (Big Data), как технологии оперирования с современными огромными массивами информации [9].

Если в интернет-компаниях и банках подобные процессы начались раньше (в силу специфики их деятельности, наличия логов, транзакций и т. п.), то в производстве и тяжелой промышленности применение Big Data только начинается.

Прежде всего, здесь возникают следующие группы задач:

- прогноз потребления энергоресурсов и материалов, необходимых для производства, оптимизация закупки и доставки материалов;
- оптимизация процесса производства (построение моделей: как используемые материалы влияют на качество производимого продукта);
- прогнозирование цен на продукцию, прогнозирование спроса, планирование сортамента и оптимизация доставки продукции клиентам;

Труды XIX Международной конференции «Аналитика и управление данными в областях с интенсивным использованием данных» (DAMDID/ RCDL'2017), Москва, Россия, 10–13 октября 2017 года

- диагностика оборудования, обнаружение и прогнозирование неисправностей.

Ниже рассказано о построении алгоритма обнаружения поломок и определения их типа. По договорённости с заказчиком не конкретизируется тип оборудования, на котором проводилась апробация алгоритма, но описана математическая составляющая использованных подходов. Также сделан обзор современных методов обнаружения аномалий (anomaly detection) и результаты их тестирования на данных реальной задачи.

1 Прикладная задача

Массив исследуемых данных состоит из показаний датчиков, установленных на оборудовании. На каждой установке – около 50 датчиков, замеры производятся каждую секунду (такая частота избыточна, поэтому показания были агрегированы до минутных), данные предоставлены за последние 3 года (т. е. около $5 \cdot 10^9$ показаний для одной установки).

Пример показаний датчиков:

- температура (°C);
- давление (кгс/м2);
- уровень шума (дБ);
- скорость вращения (об/мин);
- уровень вибрации (Гц).

На Рис. 1 и 2 показаны примеры сигналов.

Кроме того, есть текстовые логи, в которых описаны, какие работы проводились с оборудованием, а также факты возникновения внештатных ситуаций. Логи вносятся в систему логирования специалистами, поэтому являются текстами с использованием специальных терминов и сокращений. Даты и время внесения в систему добавляются автоматически.

Пример текстового лога

```
10.07.2016 10:03 Нач. бурение 200м 2к скважина станд. р38
10.07.2016 10:48 Кон. бурения Разрыв т15
```

Из логов, в частности, извлекается информация об обнаружении поломок:

- тип поломки;
- где обнаружена (id прибора и секция);
- когда обнаружена;
- когда начались ремонтные работы;
- когда закончены ремонтные работы.

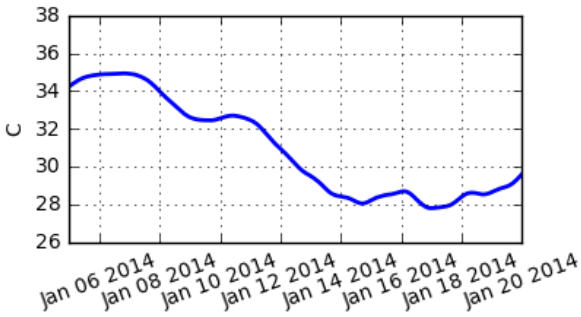

Рисунок 1 Пример сигнала датчика (температура)

За 3 года происходило в среднем 10 поломок на одно оборудование, ремонт длился от 1 дня до 1 месяца.

На основании данных требуется построить алгоритм, который детектирует поломки в режиме online, сообщая тип поломки и её локализацию.

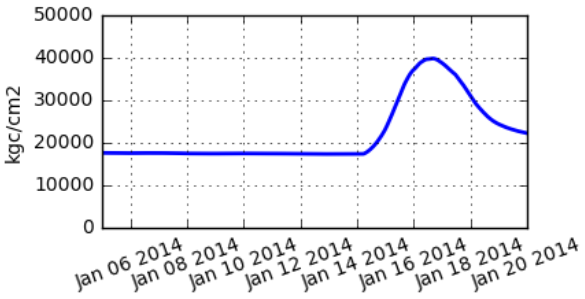

Рисунок 2 Пример сигнала датчика (давление)

2 Обработка текстовых логов

Задача обработки текстовой информации не является центральной при решении описанной выше проблемы детектирования поломок, тем не менее, с помощью обработки получены дополнительные признаки, поэтому опишем её решение.

Для анализа текстовой информации сначала был реализован классификатор текстов, который для каждого лога указывал тип действия, которое ему соответствует: запуск оборудование, выключение, начало ремонтных действий и т. п. Общая схема работы с текстами в рамках данной задачи следующая:

- удаление некоторых спецсимволов;
- переход к буквенным n-граммам;
- модель «мешок слов» (bag of words);
- tf-idf-нормировка;
- решение задач классификации;
- выделение числовых признаков из текстов;

Ниже подробно описаны все этапы.

2.1 Предварительная обработка текстов

Учитывая, что в тексте могут быть сокращения и ошибки (опечатки), изначально текст приводился в один регистр и преобразовывался в буквенные *n*-граммы (большинство спецсимволов удалялось). Лемматизация (приведение к нормальной словарной форме) и удаление стоп-слов не выполнялись [23].

Пример преобразования в 3-граммы

```
Нач. бурение → «НАЧ», «АЧ.», «Ч.Б», «.БУ», «БУР», «УРЕ», «РЕН», «ЕНИ», «НИЕ»
```

Также не были использованы различные методы исправления опечаток. Как показали эксперименты, эту проблему позволяет решить переход к *n*-граммам, поскольку, например, слова «бурение», «буриние» и сокращение «бур.» совпадают по первой триграмме «бур». Кроме того, какие-то отдельные опечатки и сокращения могут соответствовать конкретному человеку, заполнявшему журнал (что позволяет его идентифицировать).

2.2 Перевод текстов в векторную форму

Для перевода текстов в векторную форму

использовался стандартный подход: мешок слов (bag of words, [23]), т. е. была составлена разреженная матрица размера $m \times n$, где m – число текстов, n – число слов во всех текстах, ij-й элемент равен числу, выражающему, сколько раз в i-м тексте встретилось j-е слово. Таким образом, порядок слов в документе не учитывался, а фиксировались лишь числа вхождений слов. Как показали эксперименты, учёт порядка слов переходом к словарным n-граммам не улучшал качество решаемой задачи.

Над построенной матрицей было произведено tf-idf-преобразование [23]. Напомним, что tf-преобразование (term frequency) заключается в вычислении величин

$$\mathrm{tf}(h_{ij}) = \frac{h_{ij}}{\sum_{t=1}^{n} h_{it}}$$

(отношение числа вхождений определённого слова в предложение к числу слов в данном предложении). Смысл подобного преобразования – в инвариантности к повторам текста (скажем, дважды повторенное предложение имеет тот же смысл, что и однократно повторенное).

Idf-преобразование (inverse document frequency) учитывает, что чем чаще встречается слово, тем меньший смысл оно несёт:

$$\mathrm{idf}(h_{ij}) = \frac{m}{\log\left|\{t \mid h_{tj} > 0\}\right|}.$$

Tf-idf-преобразование заключается в замене каждого элемента матрицы на

$$\mathrm{tf}(h_{ij}) \cdot \mathrm{idf}(h_{ij})^{d}.$$

Обычно используется значение степени d=1, но в рассматриваемой задаче почти все слова являются профессиональными терминами, и их частое вхождение не всегда означает бесполезность для решения задачи классификации текста, а степень d как раз контролирует «учёт популярности слов». Было установлено, что оптимальное значение d=0.35.

2.3 Задача классификации текстов

Для «сырых» (необработанных) логов была сделана экспертная разметка: для каждой записи указан тип действия, о котором идёт речь в данной записи. Для разметки использованы логи из первого года, за который есть статистика. Были взяты 1000 случайных записей за год, эксперты выделили 15 классов действий.

Далее был построен классификатор на 15 классов, работа которого в дальнейшем также была оценена экспертами. В качестве тестовой выборки использовались записи 2 и 3 года. Точность определения класса действия – 97% – была признана достаточной.

К сожалению, в этой задаче достаточно трудно сделать экспертную разметку. В отличие от асессорской разметки при поиске, которую может производить практически любой человек, поскольку поисковая выдача как раз и должна быть оптимизирована под нужды среднестатистического пользователя, разметка логов оборудования понятна только людям, знакомым со специальной терминологией.

2.4 Выделение числовых признаков из текстов

Как было показано в примере лога (см. выше), кроме описания действий в нём может содержаться какая-то числовая информация, например, «бурение на глубину 200 метров». Ясно, что число 200 здесь надо уметь вычленять, чтобы потом сравнивать с аналогичными числами в других записях. В задаче классификации числа не учитывались, все они заменялись на специальное слово «number», которое указывало просто наличие какого-то числа в тексте.

Для решения описанной задачи использован следующий подход. Были отобраны типы действий, которые могут содержать числовую информацию. Для каждой числовой информации сформированы правила: где она может встречаться в записи, что ей предшествует и/или что следует после неё. На основе этих правил производился поиск соответствующих чисел. По экспертной оценке точность такого подхода составила 95%.

3 Математическая постановка задачи

После обработки логов изначально заданные многомерные временные ряды показаний датчиков были дополнены категориальными рядами действий, совершаемых с оборудованием (категориальными, поскольку значения ряда в каждый момент времени – тип действия, т. е. категориальная переменная), а также рядами значений признаков, выделенных из текстов (принимают ненулевые значения, только когда совершается соответствующее действие и в логах есть указанное числовое значение, соответствующее этому действию).

Каждый механизм был описан 64-мерным временным рядом. Как отмечено выше, известна информация об обнаружении поломок и их устранении (моменты времени). Необходимо разработать алгоритм автоматического обнаружения поломок: обучить его на статистике первых двух лет и проверить на третьем годе.

Специфика задачи состоит в том, что момент обнаружения поломки не всегда соответствует её возникновению, т. е. поломка может быть обнаружена не вовремя. По оценке экспертов, разность между этими временами может достигать нескольких недель. Кроме того, некоторые поломки могут не оказывать влияния на показания приборов, например, небольшая течь из какого-нибудь шланга (не так сильно изменяются давление и температура, кроме того, изменения происходят плавно).

Описанная задача решалась как задача машинного обучения без учителя – детектирования аномалий (anomaly detection, [5]). Далее представлен обзор современного состояния в области обнаружения аномалий. Главная причина сведения рассматриваемой проблемы именно к этой задаче: основную часть времени оборудование работает в штатном режиме. Поломки выводят оборудование из этого режима: повышается температура отдельных узлов, понижается давление и т. п. Вероятно, статистики достаточно мало, чтобы в значительной степени покрыть все виды поломок, но достаточно,

чтобы описать нормальную работу. Если детектирование аномальной работы будет соответствовать поломкам, то можно использовать такой детектор с требуемой целью.

4 Методы обнаружения аномалий (обзор)

Строго говоря, есть две похожие задачи обнаружения аномалий (Anomaly Detection): детектирование выбросов (Outlier Detection) и «новизны» (Novelty Detection). Как и выброс, «новый объект» – это объект, который отличается по своим свойствам от объектов (обучающей) выборки, но, в отличие от выброса, его в самой выборке пока нет (он появится через некоторое время, задача как раз и заключается в том, чтобы обнаружить его при появлении). Объясним это на примере решаемой задачи. Если по статистике показаний датчиков ищутся моменты времени, когда эти показания сильно отличались от показаний в остальные моменты, то это обнаружение выбросов. Если же статистика используется как пример нормальных показаний, и каждое новое показание проверяется на нормальность (похожесть на старые), то это обнаружение новизны.

Задачи обнаружения аномалий возникают при решении большого числа прикладных проблем, вот далеко не полный их перечень:

- обнаружение подозрительных банковских операций (Credit-card Fraud);
- обнаружение вторжений (Intrusion Detection);
- обнаружение нестандартных игроков на бирже (инсайдеров);
- обнаружение неполадок в механизмах по показаниям датчиков;
- медицинская диагностика (Medical Diagnosis);
- сейсмология.

Далее опишем современные методы обнаружения аномалий.

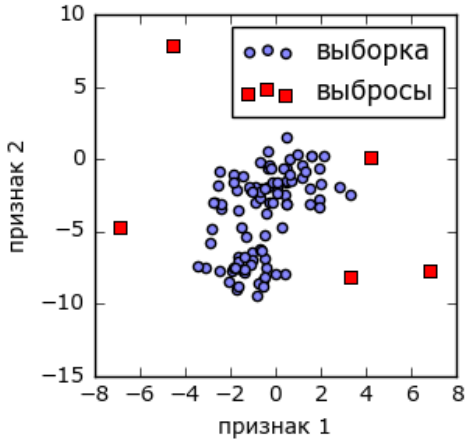

Рисунок 3 Пример выбросов в задаче с двумя признаками

4.1 Статистические тесты

Как правило, их применяют для отдельных признаков и отлавливают экстремальные значения (Extreme-Value Analysis). Для этого используют, например, Z-value [14]:

$$Z_i = \frac{x_i - \mu}{\sigma}$$

или Kurtosis [14]:

$$\frac{1}{n}\sum_{i=1}^{n} Z_i^4.$$

Многие методы визуализации, например, ящик с усами (box plot, [10]), имеют встроенные средства для детектирования и показа таких экстремальных значений.

Важно понимать, что экстремальное значение и аномалия – разные понятия. Например, в небольшой выборке

```
[1, 39, 2, 1, 101, 2, 1, 100, 1, 3, 101, 1, 3, 100, 101, 100, 100]
```

значение 39 можно считать аномалией, хотя оно не является максимальным или минимальным. Также стоит отметить, что аномалия характеризуется, как правило, не только экстремальными значениями отдельных признаков, см. Рис. 3.

4.2 Модельные тесты

Идея очень простая: строим модель, которая описывает данные; точки, которые сильно отклоняются от модели (на которых модель сильно ошибается), и есть аномалии, см. Рис. 4. При выборе модели можно учесть природу задачи, функционал качества и т. п.

Например, в исследуемой задаче можно прогнозировать значения временных рядов с помощью LSTM-нейронной сети [11]. Если реальные значения сильно отличаются от предсказываемых, то это свидетельствует об аномальном поведении. Такой подход хорошо показал себя на недавнем хакатоне лаборатории Касперского [24] по распознаванию аномалий в технологических процессах завода [9].

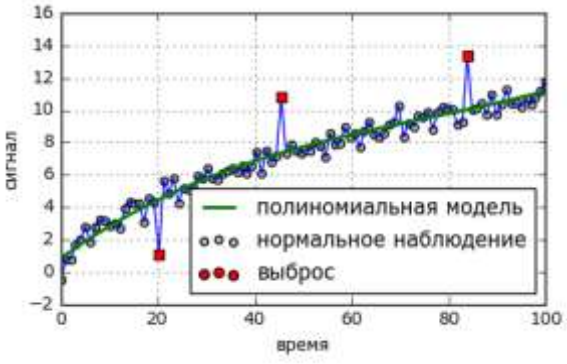

Рисунок 4 Пример применения модельного подхода

Как правило, в задачах обнаружения поломок исходная информация представлена в виде сигналов. Поэтому используется аппарат обработки цифровых сигналов, по крайней мере, на первом этапе решения задачи (для уменьшения размерности и чистки данных). Например, при анализе вибраций [13]

используют дискретное преобразование Фурье (DFT), вейвлеты [21], спектрограммы. В задачах без известной разметки поломок применяют скрытые марковские модели (HMM), а также их различные обобщения [6]. Основная проблема таких алгоритмов – большие временные затраты. Многие подходы практически бесполезны при работе с большими данными из-за использования трудоёмких методов оптимизации: EM-алгоритма [8], MCMC (Markov Chain Monte Carlo) [12] и т. д.

4.3. Итерационные методы

Можно последовательно удалять группы «особо подозрительных объектов». Например, в *n*-мерном признаковом пространстве можно удалять выпуклую оболочку точек-объектов, считая её представителей выбросами, см. Рис. 5. Как правило, методы этой группы достаточно трудоёмки.

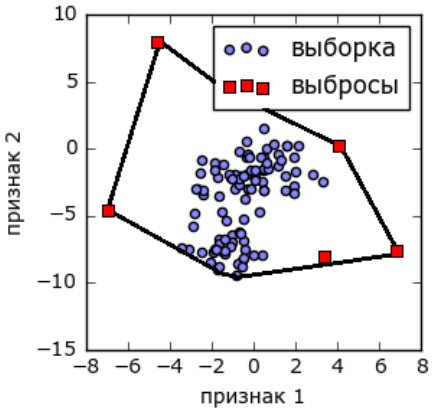

Рисунок 5 Выпуклая оболочка множества точек

4.4. Метрические методы

Это одни из самых популярных методов (судя по числу публикаций, [1]), в них постулируется существование некоторой метрики в пространстве объектов, которая и помогает найти аномалии. Интуитивно понятно, что у выброса мало соседей, а у типичной точки много. Поэтому хорошей мерой аномальности может служить, например, «расстояние до *k*-го соседа» [4]. Здесь используются специфические метрики, например, расстояние Махаланобиса.

4.5. Методы подмены задачи

В этих методах задача обнаружения аномалии заменяется другой задачей, для которой есть удобные и быстрые методы решения. Например, можно сделать кластеризацию, тогда маленькие кластеры, скорее всего, состоят из аномалий, см. Рис. 6.

В исследуемой задаче есть разметка: известны времена обнаружения неисправностей, поэтому описания работы оборудования в эти моменты можно считать классом 1 (размеченные аномалии), а описания работы оборудования после ремонтов и плановых проверок – классом 0 (нормальная работа). Таким образом, решение задачи сводится к решению задачи классификации (classification).

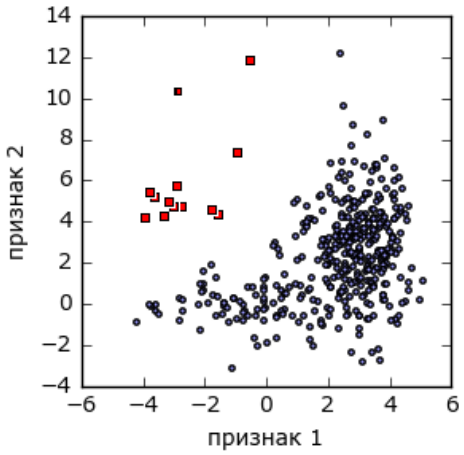

Рисунок 6 Пример конфигурации точек с малым кластером

4.6. Методы машинного обучения

Задачу обнаружения аномалий рассматривают также как отдельную задачу обучения без учителя (unsupervised learning). Такой метод решения может быть отнесён к модельному подходу 4.2, но в этот подраздел вынесены самые популярные алгоритмы (есть реализации в библиотеке scikit-learn языка Python [22]):

- **метод опорных векторов для одного класса** (OneClassSVM, [20]);
- **изолирующий лес** (IsolationForest, [16]);
- **эллипсоидальная аппроксимация данных** (EllipticEnvelope, [19]).

Первый метод – это обычный метод опорных векторов (SVM, [2]), который отделяет выборку от начала координат. Изолирующий лес (Isolation Forest) – это одна из вариаций идеи случайного леса (Random Forest, [3]):

- лес состоит из деревьев;
- каждое дерево строится до исчерпания выборки;
- для построения ветвления в дереве выбираются случайный признак и случайное расщепление,
- для каждого объекта мера его нормальности – среднее арифметическое глубин листьев, в которые он попал (изолировался, см. Рис. 7).

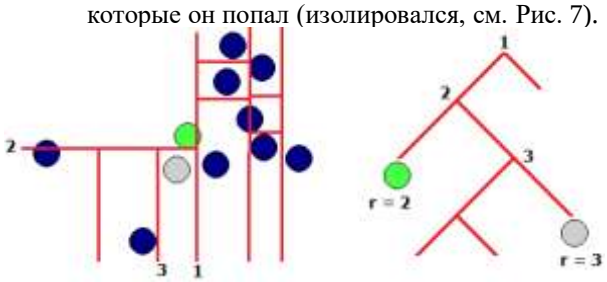

Рисунок 7 Вычисление оценки аномальности в изолирующем лесе

Логика алгоритма простая: при описанном «случайном» способе построения деревьев выбросы

будут попадать в листья на ранних этапах (на небольшой глубине дерева), т. е. выбросы проще «изолировать» (напомним, что дерево строится до тех пор, пока каждый объект не окажется в отдельном листе).

В эллипсоидальной аппроксимации данных, как следует из названия, облако точек моделируется как внутренность эллипсоида. Метод хорошо работает только на одномодальных данных, а особенно хорошо – на нормально распределённых. Степень новизны здесь фактически определяется по расстоянию Махалонобиса.

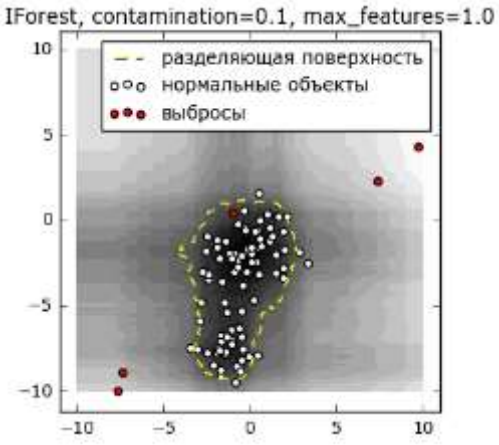

Рисунок 8 Оценка аномальности, полученная с помощью изолирующего леса библиотеки scikit-learn (чем светлее фон, тем аномальнее)

4.7 Ансамбли алгоритмов

Как и во многих других областях машинного обучения, при поиске аномалий часто используют несколько алгоритмов, как правило, разной природы. Каждый из них даёт оценку аномальности, и эти оценки потом «усредняют». Не всегда используют обычное среднее арифметическое, например, иногда хорошее качество показывает максимум (если какой-то алгоритм уверен в аномальности объекта, то, скорее всего, так оно и есть).

Поскольку ключевым моментом в реальных задачах обнаружения аномалий является выбор признаков, которые характеризуют те или иные отклонения от нормы, алгоритмы из ансамбля строят, пытаясь угадать хорошие пространства. Здесь популярны:

- Feature Bagging – для каждого алгоритма берут случайное признаковое подпространство [15];
- Rotated Bagging – в выбранном случайном признаковом подпространстве совершают случайный поворот [1].

5 Исследование методов обнаружения аномалий на реальных данных

Опишем результаты применения различных методов обнаружения аномалий для решения реальной прикладной задачи детектирования поломок. Каждый объект – признаковое описание оборудования в рассматриваемый момент времени. В качестве признаков использовались 64 исходных значения (показания датчиков, а также информация из текстовых логов). Кроме того, были построены признаки, описывающие поведение в прошлом (1 минуту назад, 5 минут, 1 час, 1 сутки назад), и разности между текущими показаниями и показаниями в прошлом. Для некоторых подходов (например, модельного) построения признакового пространства не требуется.

Методы тестировались на последнем году, за который есть статистика. В таблицах указана полнота (какой процент поломок найден) и точность (сколько из детектируемых аномалий действительно являются поломками). Порог детектирования (если оценка анормальности выше него, то алгоритм сигнализирует поломку) подбирался так, чтобы среднее число детектирований совпадало с ожидаемым числом поломок. В таблицах подходы пронумерованы согласно обзору раздела 4.

В целом результаты, представленные в табл. 1, можно считать неудовлетворительными, поскольку заказчик рассчитывал на точность 90% при такой же полноте, но при анализе ошибок были выявлены следующие особенности предложенного подхода. В большинстве случаев детектируется именно аномальное поведение в работе оборудования, т. е. отличающееся от штатного. Поэтому большинство сигналов об анормальности относилось к

- поломкам оборудования;
- неправильной эксплуатации (нарушению правил);
- смене режимов (в том числе, включению и выключению).

В результате получился алгоритм, который детектирует все эти ситуации, что вполне устраивало заказчика. Если пересчитать качество в терминах точности и полноты обнаружения перечисленных ситуаций, то получим Табл. 2.

Таблица 1 Точность и полнота распознавания поломок различными подходами

подход	точность	полнота
4.1	72%	30%
4.2	55%	80%
4.3	68%	32%
4.4	70%	52%
4.5	81%	45%
4.6	80%	80%
4.7	85%	70%

Таблица 2 Точность и полнота распознавания аномалий различными подходами

подход	точность	полнота
4.1	77%	40%
4.2	80%	92%
4.3	70%	45%
4.4	78%	62%
4.5	80%	60%
4.6	97%	87%
4.7	95%	90%

Отметим, что для достижения высокого качества достаточно использовать методы машинного обучения, описанные в разделе 4.6. Использование ансамблей не сильно улучшает качество, но существенно усложняет алгоритмы. Как показано в Табл. 3, самый лучший метод здесь – изолирующий лес.

Таблица 3 Точность и полнота распознавания аномалий методов машинного обучения

метод	точность	полнота
OneClassSVM	88%	85%
IsolationForest	97%	87%
EllipticEnvelope	72%	70%

При правильном детектировании поломки точность определения типа поломки – 87% (для этого решалась отдельная задача классификации), что также оказалось приемлемо, поскольку некоторые типы поломок сложно различать на основе показаний датчиков, например, «низкое давление воды» и «засор подводящих шлангов».

Таблица 4 Среднее время работы алгоритмов разных подходов

подход	время	
	обучения	детектирования
4.1	5 сек	< 1сек
4.2	12 мин	1 сек
4.3	–	9 мин
4.4	9 мин	6 мин
4.5	8 мин	< 1 сек
4.6	10 мин	< 1 сек

В Табл. 4 показано среднее время работы алгоритмов разных групп. Для большинства алгоритмов можно выделить отдельно этап обучения (анализ исходной информации) и детектирования (принятие решения о поломке на основе только что поступивших данных). Все алгоритмы были реализованы на языке Python 3.x.

6 Благодарности

Авторы выражают благодарности компании ООО «Алгомост» за поставленную задачу и консультации со специалистами.

7 Заключение

Разработан алгоритм выявления аномалий в работе оборудования. Кроме поломок, он сигнализирует также о любой некорректной работе и смене режимов работы. Качество оказалось достаточно высоким и полностью удовлетворило заказчика: 97% точности и 87% полноты.

Дальнейшие планы по усовершенствованию алгоритма:

- решение задачи прогнозирования поломок (для заказчика актуально составление расписания проверок и капитального ремонта с учётом износа оборудования);
- решение задачи размещения датчиков (от некоторых датчиков можно отказаться, не снижая качества детектирования поломок);
- использование видеоинформации и изображений (для некоторого оборудования есть кадры съёмки рабочего процесса, которые также регулярно производятся и сохраняются);
- улучшение качества определения типа поломки (пока при решении этой задачи не было сделано такого же масштабного перебора различных подходов, как для детектирования самого факта поломки).

Литература

[1] Aggarwal, C.C.: Outlier Analysis. Springer-Verlag, New York (2013). doi: 10.1007/978-1-4614-6396-2

[2] Boser, B.E., Guyon, I.M., Vapnik, V.N.: A Training Algorithm for Optimal Margin Classifier. Proc. of the Fifth Annual Workshop on Computational Learning Theory – COLT'92, p. 144 (1992). doi: 10.1145/130385.130401

[3] Breiman, L.: Random Forests. Machine Learning, 45 (1), pp. 5-32 (2001). doi:10.1023/A:1010933404324

[4] Breunig, M.M., Kriegel, H.-P., Ng, R.T., Sander, J.: LOF: Identifying Density-based Local Outliers. Proc. of the 2000 ACM SIGMOD Int. Conference on Management of Data, pp. 93-104 (2000)

[5] Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. ACM Computing Surveys, 41 (3), pp. 1-58 (2009). doi: 10.1145/ 1541880.1541882

[6] Chao, Y.: Unsupervised Machine Condition Monitoring using Segmental Hidden Markov Models. IJCAI'15 Proc. of the 24th Int Conf. on Artificial Intelligence. AAAI Press, pp. 4009-4016 (2015)

[7] Chen, T., Guestrin, C.: XGBoost: A Scalable Tree Boosting System. Proc. of the 22Nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, San Francisco, CA, USA (2016)

[8] Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum Likelihood from Incomplete Data via the EM Algorithm. J. of the Royal Statistical Society. Series B. 39, pp. 1-38 (1977)

[9] Filonov, P., Lavrentyev, A., Vorontsov, A.: Multivariate Industrial Time Series with Cyber-Attack Simulation: Fault Detection Using an LSTM-based Predictive Data Model. NIPS Time Series Workshop (2016). https://arxiv.org/abs/1612.06676

[10] Frigge, M., Hoaglin, D.C., Iglewicz, B.: Some Implementations of the Box Plot. The American Statistician, 43 (1), pp. 50-54 (1989)

[11] Hochreiter S.: Long Short-term Memory. Neural Computation, 9 (8), pp. 1735-1780 (1997). doi: 10.1162/neco.1997.9.8.1735

[12] Johnson, M.J., Willsky, A.S.: Bayesian Nonparametric Hidden Semi-Markov Models. J. of Machine Learning Research, 14 (1), pp. 673-701 (2013)

[13] Klein, R.: A Method for Anomaly Detection for Non-stationary Vibration Signatures. Annual Conf. of the Prognostics and Health Management Society (2013). https://www.phmsociety.org/sites/phmsociety.org/files/phm_submission/2013/phmc_13_ 038.pdf

[14] Kreyszig, E. Advanced Engineering Mathematics. John Wiley & Sons Inc, 4th edition, 880 p. (1979)

[15] Lazarevic, A., Kumar, V.: Feature Bagging for Outlier Detection. Proc. of the ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining, pp. 157-166 (2005). doi: 10.1145/1081870.1081891

[16] Liu, F.T., Tony, T.K.M., Zhou, Z.H.: Isolation Forest. Proc. of the 2008 Eighth IEEE Int. Conf. on Data Mining, pp. 413-422 (2008)

[17] Matusugu, M., Mori, K., Mitari, Y., Kaneda, Y.: Subject Independent Facial Expression Recognition with Robust Face Detection using a Convolutional Neural Network. Neural Networks, 16 (5), pp. 555-559 (2003). doi: 10.1016/S0893-6080(03)00115-1

[18] Mayer-Schönberger, V., Cukier, K.: Big Data: A Revolution that will Transform How We Live, Work, and Think. John Murray, London (2013)

[19] Rousseeuw, P.J., Van Driessen, K.: A Fast Algorithm for the Minimum Covariance Determinant Estimator. Technometrics, 41 (3), pp. 212-223 (1999)

[20] Schölkopf B., et al.: Estimating the Support of a High-dimensional Distribution. Neural Computation, 13 (7), pp. 1443-1471 (2001)

[21] Sheriff, M.Z., Nounou, M.N.: Improved Fault Detection and Process Safety Using Multiscale Shewhart Charts. J. Chem. Eng. Process Technol., 8 (2), pp. 1-16 (2017). doi: 10.4172/2157-7048.1000328

[22] Библиотека алгоритмов машинного обучения для Python, http://scikit-learn.org/stable/

[23] Маннинг, К., Рагхаван, П., Шютце, Х.: Введение в информационный поиск. М.: Изд-во Вильямс (2011)

[24] Хакатон по анализу данных от лаборатории Касперского. https://events.kaspersky.com/hackathon/