Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»
Факультет «Высшая школа экономики и управления»
Кафедра «Информационные технологии в экономике»

ПРОЕКТ ПРОВЕРЕН
Рецензент, начальник управления реализации проектов Министерства информационных технологий и связи Челябинской области
_____ (И.А. Филатов)
«____» _____ 2019 г.

ДОПУСТИТЬ К ЗАЩИТЕ
Заведующий кафедрой, д.т.н., с.н.с.
_____ (Б.М. Суховилов)
«____» _____ 2019 г.

РАЗРАБОТКА МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ ДЛЯ ПРОГНОЗИРОВАНИЯ СБОЕВ ТЕХНОЛОГИЧЕСКИХ ЛИНИЙ С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ ГРАДИЕНТНОГО БУСТИНГА НА ПРИМЕРЕ КОМПАНИИ BOSCH

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ
ЮУрГУ–38.04.05.2019.126.ПЗ ВКР

Руководитель проекта, д.т.н.
_____ (В.В. Мокеев)
«____» _____ 2019 г.

Автор проекта,
студент группы ЭУ– 244
_____ (С.А. Лайко)
«____» _____ 2019 г.

Нормоконтролер, доцент
_____ (Е.В. Бунова)
«____» _____ 2019 г.

Челябинск 2019

АННОТАЦИЯ

Лайко С.А. – Разработка математических моделей для прогнозирования сбоев технологических линий с использованием методов градиентного бустинга на примере компании BOSCH – Челябинск: ЮУрГУ, ЭУ-244, 2019. – 98 с., 25 ил., 8 табл., библиогр. список – 57 наим.

Выпускная квалификационная работа посвящена разработке математических моделей для прогнозирования сбоев происходящих на технологических линиях с использованием методов градиентного бустинга на примере компании BOSCH.

В работе представлены материалы исследования технологических линий, принцип работы и сбои, возникающие при производстве. Проведен анализ машинного обучения, а также обоснован выбор, почему именно его следует использовать для прогнозирования сбоев на технологических линиях. Представлены методы машинного обучения, а также сделан выбор в сторону метода, использованного при прогнозировании сбоев. В работе присутствует описание выбранных методов, исходные данные предоставленные компанией BOSCH, предварительная обработка данных, а также результаты проведенной работы. Описана дорожная карта коммерциализации проекта, создан сайт по предоставлению услуги прогнозирования внутренних сбоев. Рассчитан медиаплан и ценовая политика коммерциализации проекта.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Машинное обучение является подполем искусственного интеллекта (ИИ). Цель машинного обучения, как правило, состоит в том, чтобы понять структуру данных и приспособить эти данные к моделям, которые могут быть поняты и использованы людьми. Хотя машинное обучение является областью компьютерных наук, оно отличается от традиционных вычислительных подходов. В традиционных вычислениях алгоритмы – это наборы явно запрограммированных инструкций, используемых компьютерами для вычисления или решения проблемы. Вместо этого алгоритмы машинного обучения позволяют компьютерам обучаться вводу данных и использовать статистический анализ для вывода значений, попадающих в определенный диапазон [17].

Машинное обучение – это постоянно развивающаяся область. В связи с этим необходимо учитывать некоторые соображения при работе с методиками или при анализе влияния процессов машинного обучения [49].

Умное производство рекламируется как следующая промышленная революция. Благодаря мониторингу производственных процессов в реальном времени для сохранения конкурентоспособности и повышения производительности, очевидным шагом, становится использование методов обработки больших данных.

Любой пользователь технологии, на сегодняшний день может извлечь выгоду из машинного обучения. Технология распознавания лиц позволяет правоохранительным органам оперативно распознавать и своевременно задерживать преступников. Автомобили с автоматическим управлением, которые полагаются на машинное обучение, могут вскоре стать доступными для потребителей. Также примером может послужить, компания оборонного производства Raytheon. Она внедрила MES (систему управления производством) на своем ракетном заводе в Хантсвилле, штат Алабама. Система позволяет собирать и анализировать данные о заводских цехах и позволяет определить с невероятной точностью, сколько раз необходимо ввернуть винт в критически важный компонент, чтобы этот компо-

нент стал идеальным. Большие данные могут использоваться для прогнозирования частоты отказов оборудования, оптимизации управления запасами и определения приоритетов процессов. В 2012 году Intel сэкономила 3 миллиона долларов на производственных затратах за счет использования прогнозирующих аналитических методов для определения приоритетов проверок своих кремниевых чипов. Интеллектуальное производство – это следующая крупная разработка после хорошо отлаженных процессов бережливого производства и методологии Six Sigma. Следуя этой тенденции, Bosch записал свой набор данных, состоящий из анонимизированных записей измерений и испытаний, проведенных для каждого компонента на конвейере и предоставил возможность предсказать сбои деталей продукта, что позволило Bosch приносить качественные продукты по более низким ценам для конечного пользователя [2].

В выпускной квалификационной работе описывается подход к решению проблемы производительности технологической линии Bosch. Максимизация доходности производства лежит в основе обрабатывающей промышленности. На сборочной линии Bosch данные записываются для продуктов по мере их прохождения на каждом этапе. Методы обработки данных применяются к данному, огромному хранилищу данных, содержащему записи испытаний и измерений, выполненных для каждого компонента на конвейере для прогнозирования внутренних сбоев. Было обнаружено, что можно обучить модель, которая предсказывает, какие части, скорее всего, выйдут из строя [2].

Представляем выводы из набора данных. Исследуем проблемы, с которыми сталкиваются из-за размера набора данных, типа записанных данных и алгоритмов машинного обучения, которые подходят для такого рода задач. В разделе I описаны архитектура и сбои, возникающие при производстве на технологичских линиях, мониторинг производства, в разделе II рассмотрены методы классификации используемы в машинном обучении для прогнозирования сбоев технологических линий, включая алгоритм k ближайших соседей, случайный лес, классификатор экстра-деревьев, градиентный бустинг и легкий градиентный бустинг, а

также метрики качества и процедура – кросс-валидация. В разделе III представлен набор данных и его предварительная обработка, исследование эффективности методов: k ближайших соседей, случайного леса, классификатора экстра-деревьев, градиентного бустинга и легкого градиентного бустинга. В обсуждение проводится подробный анализ прогнозирования сбоев технологических линий на примере данных компании BOSCH и подведены итоги в сравнении всех методов. В разделе IV будет рассмотрена коммерциализация проекта.

Актуальность темы обусловлена необходимостью прогнозирования сбоев технологических линий на примере данных компании Bosch. Благодаря предоставленным данным, может быть построена более разумная система обнаружения сбоев, и выделены части, помеченные с вероятностью выхода из строя. Все это позволит снизить эксплуатационные расходы и увеличить прибыль.

Основной целью работы является – снижение эксплуатационных расходов и увеличение прибыли предприятия на примере компании Bosch посредством прогнозирования сбоев технологических линий.

Чтобы достичь поставленную цель, необходимо решить следующие задачи:

– проанализировать архитектуру производственных линий;

– проанализировать виды сбоев, возникающих при производстве на технологических линиях;

– проанализировать методы классификации прогнозирования сбоев технологических линий;

– объяснить выбор использованных метрик качества и процедуры кросс-валидации;

– проанализировать предоставленный набор данных;

– провести предварительную обработку данных;

– исследовать эффективность классификаторов для прогнозирования сбоев технологических линий;

– спрогнозировать сбои, на примере данных компании Bosch и сравнить результаты;

– разработать коммерциализацию проекта.

Научной новизной является использование метода градиентного бустинга для прогнозирования сбоев технологических линий.

Практическая значимость – использование данного подхода позволяет снизить эксплуатационные расходы и увеличить прибыль производственных предприятий.

Апробации работы:

1. Лайко С.А., WEB-ресурс как способ продвижения предприятия / С.А. Лайко, А.А. Тютёва // Научные исследования: теория, методика и практика: материалы III Междунар. науч.-практ. конф. (Чебоксары, 19 нояб. 2017 г.). В 2 т. Т. 2 / редкол.: О.Н. Широков [и др.] – Чебоксары: ЦНС «Интерактив плюс», 2017. – С. 258-260. – ISBN 978-5-6040208-7-6;

2. Лайко С.А., Оценка эффективности технологических цепочек производственных предприятий / С.А. Лайко, А.А. Тютёва // Образование и наука в современных реалиях: материалы IV Междунар. науч.–практ. конф. (Чебоксары, 26 февр. 2018 г.) / редкол.: О.Н. Широков [и др.] – Чебоксары: ЦНС «Интерактив плюс», 2018. – С. 209-210. – ISBN 978-5-6040732-7-8;

3. Участие в международном конкурсе: SMS Data Challenge 2.0 Data Analytics – Improved Sticker Detection based on Fiber Optical Temperature Measurement. – 2017.

1. СБОИ, ВОЗНИКАЮЩИЕ ПРИ ПРОИЗВОДСТВЕ НА ТЕХНОЛОГИЧЕСКИХ ЛИНИЯХ

1.1 Архитектура производственных линий

Сборочная линия – это производственный процесс, в котором взаимозаменяемые части последовательно добавляются в продукт для создания конечного продукта. В большинстве случаев производственная сборочная линия представляет собой полуавтоматическую систему, по которой перемещается продукт. На каждой станции вдоль линии происходит некоторая часть производственного процесса. Рабочие и машины, используемые для производства изделия, стоят на одной линии, и продукт движется по всему циклу от начала до конца.

Методы сборочной линии были первоначально введены для повышения производительности и эффективности производства. Достижения в методах сборочной линии делаются регулярно, так как обнаруживаются новые и более эффективные способы достижения цели увеличения пропускной способности (количество продуктов, произведенных за определенный период времени). В то время как методы сборочной линии применяются главным образом к производственным процессам, также известно, что бизнес-эксперты применяют эти принципы в других областях бизнеса, от разработки продукта до управления.

Введение сборочной линии в американские производственные цеха в начале двадцатого века коренным образом изменило характер производственных мощностей и предприятий по всей стране. Благодаря сборочной линии, сокращаются сроки производства, увеличиваются затраты на оборудование, а персонал и руководство стараются не отставать от изменений. Сегодня, используя современные методы сборочной линии, производство стало очень усовершенствованным процессом, в котором добавленная стоимость добавляется к деталям вдоль линии. Производство сборочной линии все чаще характеризуется «параллельными процессами» – множеством параллельных операций, которые переходят в финальную стадию сборки. Эти процессы требуют сложных систем связи, планов движения материалов и производственных графиков. Тот факт, что система сборочной линии представляет собой единую большую систему, означает, что сбои в одной точке «линии» приводят к замедлению и последствиям с этой точки вперед.

Обеспечение бесперебойной работы всей системы требует значительной координации между частями системы.

Мощь компьютеров позволила системам слежения стать более совершенными, и это, в свою очередь, позволило снизить затраты, связанные с проведением инвентаризации [14]. Методы изготовления точно в срок – Just-in-time (JIT) были разработаны, чтобы снизить стоимость перевозки деталей и расходных материалов в качестве инвентаря. В рамках системы JIT производственные предприятия имеют запасы всего на один или несколько дней на заводе, полагаясь на поставщиков, которые поставляют детали и материалы "по мере необходимости". Будущие разработки в этой области могут включать поставщиков, осуществляющих деятельность на самом производственном объекте, или расширение электронных связей между производителями и поставщиками для обеспечения более эффективной поставки материалов и деталей.

1.1.1 Вариации в методологиях монтажной линии

Прошедшие годы принесли многочисленные изменения в методологии сборочных линий. Эти новые изменения можно отнести не только к общим улучшениям в технологии и планировании, но и к факторам, которые являются уникальными для каждой компании или отрасли. Ограничения капитала, например, могут оказать большое влияние на план малого бизнеса по внедрению или совершенствованию методов производства сборочных линий, в то время как изменения в международной конкуренции, правилах эксплуатации и доступности материалов могут повлиять на картину сборочных линий целых отраслей. Ниже приводится краткое описание методов сборочной линии, которые в настоящее время пользуются определенной популярностью в мире производства.

– Модульная сборка – это усовершенствованный метод сборочной линии, который предназначен для повышения пропускной способности за счет повышения эффективности параллельных сборочных линий, подающих в ко-

нечную сборочную линию. Применительно к автомобилестроению модульная сборка будет включать сборку отдельных модулей – шасси, салона, кузова – на их собственных сборочных линиях, а затем соединение их вместе на конечной сборочной линии.

– Производство ячеек – этот метод производства развился благодаря возросшей способности машин выполнять несколько задач. Операторы сотовой связи могут выполнять три или четыре задачи, а роботы используются для таких операций, как обработка материалов и сварка. Ячейки продукции могут управляться одним оператором или рабочей группой из нескольких человек. В этих ячейках продукции можно связать старые детали с новыми, тем самым уменьшая объем инвестиций, необходимых для нового оборудования.

– Командное производство – командно-ориентированное производство – это еще одна разработка методов сборочной линии. Там, где работники раньше работали на рабочих местах, состоящих из одного или двух человек, и выполняли повторяющиеся задачи, теперь рабочие группы могут следить за работой на конвейере, проводя окончательные проверки качества. Сторонники считают, что подход к коллективному производству способствует большей вовлеченности работников в производственный процесс и знания системы.

– U-образная сборочная линия – линия, возможно, не самая эффективная форма для организации сборочной линии. На U-образной линии или кривой, рабочие собираются внутри кривой, и связь легче, чем по длине прямой линии. Ассемблеры могут видеть каждый процесс; что будет и как быстро; и один человек может выполнять несколько операций. Кроме того, рабочие станции вдоль «линии» могут одновременно производить несколько конструкций, что делает объект в целом более гибким. Смены также легче выполнять по U-образной линии, а благодаря лучшему общению между работниками упрощается перекрестное обучение. Преимущества U-образной линии послужили широкому расширению их использования.

По мере того, как новые методы сборочной линии вводятся в производственные процессы, бизнес-менеджеры обращают внимание на методы для возможного применения в других областях бизнеса. Одно из таких приложений называется Совместная разработка приложений, или JAD. Это процесс, изначально разработанный для проектирования компьютерной системы. Он объединяет тех, кто работает в сфере бизнеса, и тех, кто работает в области информационных технологий, в единый семинар. Преимущества JAD включают резкое сокращение времени, необходимого для завершения проекта. Процесс JAD делает для разработки компьютерных систем то, что Генри Форд делал для производства автомобилей (метод организации машин, материалов и рабочей силы, чтобы автомобиль мог собираться гораздо быстрее и дешевле, чем когда-либо прежде – сборочная линия).

Аналогичным образом основы теории конвейерных линий успешно применяются к бизнес-процессам. Все эти новые методы организации работы имеют общую цель – повысить пропускную способность за счет сокращения времени, которое отдельные работники и их машины тратят на выполнение определенных задач. Благодаря сокращению времени, необходимого для изготовления изделия, методы сборочной линии позволили производить больше с меньшими затратами. На рисунке 1 показан пример производственной линии.

Рисунок 1 – Блоки управления станками собраны в модули производственной линии, на примере Bosch

1.2 Виды сбоев на производственных линиях

Чтобы поддерживать работу сборочных линий и поддерживать качество продукции, производственные организации в значительной степени полагаются на системное аппаратное и программное обеспечение для минимизации отказов производственной линии [18].

Но даже если производственные линии являются основой бизнеса, они все же очень чувствительны к недостаткам – и проблемы поразительно распространены. Что еще хуже, большинство компаний даже не подозревают об этих слабостях, прежде чем они станут полноценными системными сбоями.

Признаки отказа производственной линии возникают рано и часто слишком легко игнорируются, как потенциальные красные флаги. Некоторые из них включают в себя:

– задержки в получении информации от одного подразделения к другому;

– задержки в получении заказов в производственный график;

– задержки в получении доставленных завершенных заказов.

Будь то необработанная аппаратная или программная проблема, потенциальное нарушение безопасности или задержка в производстве – предупреждающие сигналы есть [27].

Когда системы выходят из строя и производство останавливается, последствия могут быть дорогостоящими и приводить:

– к длительным периодам простоя;
– к порче продукта;
– к потере важных данных;
– к необходимости полной замены оборудования.

Можно задаться вопросом: как возможно остановить проблему, когда даже неизвестно, где и как она начинается [30, 31]?

Некоторые проблемы, возникающие на компьютерных производственных линиях, являются временными, в то время, как другие являются постоянными и требуют полной замены. На систему может повлиять любое количество вещей, в том числе:

– разливы жидкости;
– скачки напряжения;
– вирусные атаки;
– основные глюки – даже простой перезапуск питания может вызвать проблемы.

Прохождение полной и тщательной оценки систем – единственный вариант, и это жизненно важно [27].

Практически в каждой организации имеется множество личных данных, которые легкодоступны любому, у кого есть доступ. Чтобы гарантировать безопасность данных, следует очень хорошо доверять безопасности систем.

Но независимо от того, насколько высока бдительность, системы могут быть уязвимы для вредоносных вирусов, хакерских атак и даже человеческих и системных ошибок, которые предоставляют несанкционированный доступ не тем

людям. Отказ некоторых подразделений не является серьезной проблемой; они могут просто сделать процесс менее эффективным в данный день [39, 42].

Тем не менее, другие сбои подразделения могут нанести вред бизнесу. И в бизнесе репутация является ключевой. Если отказ производственной линии приведет к невозможности предоставить продукт или услугу – есть большая вероятность, что это нанесет ущерб репутации этого бизнеса [32].

Если компания использует функционирующую производственную линию, нужно задать следующие ключевые вопросы:

1) Кто-нибудь делал оценку риска в последнее время?
2) Есть ли подразделение, которое, в случае неудачи, остановило бы всю линию?
3) Есть ли резервный блок?
4) Люди обучены, как управлять устройством, когда оно выходит из строя?
5) Есть ли документированный метод борьбы с ошибками?

Существует много способов перемещения данных из одной системы в другую. Некоторые методы сложны, но есть много простых и не сложных методов.

Большинство современных систем, таких как Xero или Salesforce, имеют API-интерфейсы, которые позволяют оптимизировать и управлять данными из нескольких систем. Эта возможность обеспечивает более плавную интеграцию и меньше ручного, подверженного ошибкам манипулирования данными. Однако, если нет возможности автоматизировать интеграцию между системами, почти всегда будет доступен ручной вариант. Например, большинство систем учета имеют возможность импортировать данные из файла CSV. Это позволяет извлекать ценную информацию из отключенных систем и импортировать данные без необходимости транскрипции вручную.

Каждая производственная компания в какой-то момент испытывает отказ производственной линии. Они разочаровывают, разрушительны и очень дороги.

Должно точно знать, где системы могут быть закрыты. Также необходимо знать, сколько стоит каждый час простоя, с учетом всего, от качества производства до потери рабочего времени. После того, как рассчитана стоимость простоя, можно максимально его ограничить.

1.3 Мониторинг производства

При управлении линией массового производства основной целью является обеспечение хорошего качества производства и своевременной доставки продукции клиентам с требуемым уровнем качества. Вот почему мониторинг производства так важен.

Производственный процесс состоит из длинной цепочки отдельных и, вероятно, сложных действий. Проблемы с качеством произойдут, и, хотя вы сделаете все возможное, чтобы свести их к минимуму, с этим всегда придется столкнуться.

Задача обеспечения высокого уровня качества на производстве становится еще сложнее, когда производство продукции осуществляется субподрядчиком, когда контроль над качеством в производственном процессе ограничен и сложно установить надлежащий мониторинг производства [14].

Давайте рассмотрим 8 самых важных правил, которые необходимо учитывать при создании системы контроля качества при изготовлении производственной линии:

1. Независимость и контроль:

Если выбрать контрактного производителя для запуска своего массового производства, не следует ожидать того же уровня внимания к продукту и производственному мониторингу в целом, как и к самому себе. Но нет никакой гарантии, что контрактный производитель предоставит необходимую информацию, чтобы была возможность преодолеть этот недостаток знаний и опыта работы с продукцией.

Не всегда можно рассчитывать на получение этой важной и полной информации о качестве в процессе производства. И также невозможно рассчитывать на своевременное получение информации, когда она действительно нужна.

Поэтому важно, чтобы было установлено независимое и беспристрастное качество в системе мониторинга производства. Это важно! Должен быть полный контроль над содержанием и временем информации, поступающей с производственной линии.

2. Проверка контрольных точек на протяжении всего производственного процесса:

Каждая отдельная стадия производственного процесса должна быть проверена, прежде чем продукт сможет перейти к следующему этапу. В идеале должны быть использованы испытательные станции (ручные или автоматические) на протяжении всего производственного процесса. Тестирование должно начинаться с проверки поступающего сырья, вплоть до конечной стадии, предшествующей доставке готовой продукции клиентам.

3. Анализ данных тестирования:

Анализ данных испытаний, хранящихся на всех испытательных станциях, расположенных на производственной линии, очень важен. Эти данные предоставляют бесценную информацию. Это позволяет проводить анализ первопричин проблем качества и со временем улучшать качество продукции.

4. Время имеет существенное значение:

Простои производства – это кошмар для производителей. Это может привести к значительным задержкам доставки до клиентов и нанести ущерб «сердцу» бизнеса.

Только правильная система мониторинга производства поможет минимизировать этот риск. Очень важно, чтобы был прямой и быстрый доступ к данным, собранным на испытательных станциях, расположенных на производственной линии.

Также следует быстро реагировать. Запустить анализ основных причин. Определить и устранить проблему, и возобновить полное производство как можно скорее. Это позволяет восстановить контроль над производственной линией, даже если она находится на другом конце света. Система помогает значительно улучшить качество, повысить производительность и свести к минимуму простои.

Работает это так, что данные тестирования автоматически и непрерывно собираются на испытательных станциях, расположенных на производственной линии, анализируются и надежно загружаются на аналитические панели, специально предназначенные для анализа. Предоставляет 24/7 точную информацию о каждой проверенной единице. Возможность просматривать и анализировать данные для одного устройства, проводить быстрый анализ основных причин и улучшать качество продукции.

5. «Pass» или «Fail» не дают оценку:

Обычно испытательные станции, расположенные на производственной линии (как ручные, так и автоматизированные), измеряют несколько технических параметров. Тестирование заканчивается с указанием – «Пройдено» или «Не пройдено». Если результат теста показывает «Проход», то устройство переходит к следующему этапу производства. Если результат теста показывает «Отказ», то прибор отправляется техническому специалисту для дальнейшего анализа.

6. Почему обычно следует обращать внимание только на критерии «Проход» или «Отказ»:

Почему не интересуют другие протестированные параметры для лучшего контроля качества продукции, причина в информационной перегрузке. При запуске линии массового производства невозможно «переварить» всю подробную информацию, собранную на испытательных станциях. Обычно анализ этих данных подробен только тогда, когда обнаруживается проблема качества, и сразу же следует заняться поиском основной причины проблемы. Если получено «Pass», то вся эта подробная информация, как правило, забывается.

Простой «Проход» или «Отказ» дает мало информации или вообще не дает информацию в крайних случаях – когда один или несколько технических параметров устройства находятся в пределах допустимого. Краевые случаи могут привести к выходу блока из строя во время работы, например, в экстремальных условиях (холод, жара, влажность, электрическая перегрузка, удар и т.д.). Для точного и полезного анализа данных о качестве необходимо найти метод, который позволит регулярно просматривать все данные испытаний для устройства и анализировать их осмысленно с другими проверенными устройствами, другими станциями тестирования и с историческими данными испытаний. Это позволит создать надлежащую систему мониторинга производства, которая обеспечит наилучшее качество производства.

7. Видимость всего качества в процессе производства:

Производственный процесс представляет собой цепочку отдельных, но зависимых процессов сборки и тестирования, которые вместе создают конечный продукт.

Техническая проблема, возникшая на одном этапе производственного процесса, может быть выявлена только на более позднем этапе испытаний производственного процесса. Например, неисправная кнопка, собранная на устройстве, может быть обнаружена только во время функционального тестирования, несколькими этапами позже.

Следует ожидать, что результаты испытаний на любом из производственных этапов потенциально могут повлиять на другие этапы процесса. Просмотр и анализ данных, собранных на одной испытательной станции в отдельности, просто недостаточны для надлежащего контроля качества продукции.

Чтобы увидеть полную картину, необходимо собрать и проанализировать сквозные результаты в соответствии с серьезностью и частотой каждой найденной проблемы.

Производство продукции может происходить на другом континенте. Это может происходить в соседней комнате. В любом случае должно быть предупрежде-

ние, даже если не будет пристального внимания за каждым этапом качества производственного процесса, все равно будут известны основные проблемы в момент их возникновения. Механизм автоматического оповещения, который генерирует уведомления о критических проблемах на производственной линии, является обязательным условием для мониторинга качества продукции.

8. Прогнозирование – будущее:

Чтобы быть мудрым следует исправить проблемы с качеством, прежде чем они возникнут. Хороший способ добиться этого – создать механизм прогнозирования, который анализирует тенденции в результатах тестирования и предупреждает о потенциальных проблемах качества.

1.4 Обзор работ

Данная тематика по прогнозированию производственных сбоев технологических линий была реализована исследователями, в следующих работах:

1) Ankita Mangal, Nishant Kumar. 2016. Использование больших данных для повышения производительности производственной линии. Авторы продемонстрировали важность использования больших данных, а также важным моментом является использование их для повышения производительности производственной линии;
2) Pierre Geurts, Damien Ernst, Louis Wehenkel. 2006. Чрезвычайно рандомизированные деревья;
3) Sobhan Sarkar, Atul Patel, Sarthak Madaan, Jhareswar Maiti. 2016. Прогнозирование несчастных случаев на производстве с использованием подхода дерева решений;
4) Mohammed J. Islam*, Q. M. Jonathan Wu, Majid Ahmadi, Maher A. Sid-Ahmed. 2010. Исследование эффективности наивных байесовских классификаторов и K-ближайших соседних классификаторов.

В представленных работах были предложены различные методы, а в частности такие, как:

– k Ближайших Соседей;
– Случайный лес;
– Классификатор экстра-деревьев;
– Градиентный бустинг.

Не стоит забывать о легком градиентном бустинге. Стоит отметить такие методы, как градиентный бустинг и легкий градиентный бустинг, современны и позволяют обучить модель с высокой скоростью, что необходимо для реализации онлайн прогнозирования.

1.5 Постановка задачи

Прогнозирование будущего очаровывало людей с незапамятных времен. Ежедневно над прогнозированием работают несколько миллионов человек: астрологи, метеорологи, политики, специалисты по опросам общественного мнения, фондовые аналитики и врачи, а также специалисты по вычислительной технике и инженеры. Естественно, как все компьютерные ученые, происходит фокусировка на прогнозировании сбоев компьютерных систем, и эта тема вызывает интерес уже более 30 лет. Однако то, что понимается под термином «прогнозирование отказов», варьируется в разных исследовательских сообществах и также изменилось за десятилетия.

Поскольку компьютерные системы становятся все более и более сложными, они также динамически меняются из-за мобильности устройств, изменяющихся сред выполнения, частых обновлений, онлайн-ремонта, добавления и удаления компонентов системы и самой сложности систем (сетей). Классическая теория надежности и традиционные методы редко учитывают фактическое состояние системы и, следовательно, не способны отражать динамику систем времени выполнения и процессов отказа. Такие методы обычно полезны при разработке для долгосрочных или усредненных прогнозов поведения и сравнительного анализа.

Девиз исследования методов онлайн-предсказания неудач может быть хорошо выражен словами греческого поэта К.П. Кавафи, который сказал [Кавафи 1992]: «Обычные смертные знают, что происходит сейчас, боги знают, что ждет их в будущем, потому что они одни полностью просветленны. Мудрецы знают о будущих событиях, которые вот-вот произойдут» [8]. Для обычных смертных предсказание ближайшего будущего является более умным и зачастую более успешным, чем попытка долгосрочных прогнозов. Краткосрочные прогнозы особенно полезны для предотвращения возможных бедствий или для ограничения ущерба, вызванного сбоями компьютерной системы. Учет динамических свойств современных компьютерных систем онлайн-прогнозирования сбоев включает в себя измерения фактических параметров системы во время выполнения, чтобы оценить вероятность возникновения сбоев в ближайшем будущем в виде секунд или минут.

В информатике методы прогнозирования используются в различных областях. Например, прогнозирование ветвления в микропроцессорах пытается выполнить предварительную выборку команд, которые наиболее вероятно будут выполнены, или прогнозирование памяти или кэша пытается предсказать, какие данные могут потребоваться дальше. Ограничивая область сбоев, есть несколько областей, где используется термин прогнозирование. Например, в теории надежности цель прогнозирования надежности состоит в том, чтобы оценить будущую надежность системы на основе ее конструкции или спецификации. Книга [Lyu 1996], и особенно главы [Farr 1996] и [Brocklehurst и Littlewood 1996], дают хороший обзор, а книги [Musa et al. 1987; Blischke and Murthy 2000] охватывают эту тему всесторонне. Denson [1998] дает обзор методов прогнозирования надежности электронных устройств [7, 9]. Тем не менее, тема этого опроса состоит в том, чтобы определить во время выполнения, произойдет ли сбой в ближайшем будущем, на основе оценки текущего состояния системы. Такой тип прогнозирования отказов называется онлайн-прогнозированием отказов [23].

Хотя архитектурные свойства, такие как взаимозависимости, играют решающую роль в некоторых методах прогнозирования. Онлайн прогнозирование отка-

зов связано с краткосрочной оценкой, которая позволяет решить, будет ли сбой, например, на пять минут вперед или нет. Прогнозирование надежности систем, однако, касается долгосрочных прогнозов, основанных, например, на процентах отказов, архитектурных свойствах или количестве исправленных ошибок.

Прогнозирование сбоя в сети часто путают с анализом первопричин. Наблюдая некоторое неправильное поведение в работающей системе, анализ первопричин пытается определить неисправность, вызвавшую его, в то время как прогнозирование ошибок пытается оценить риск того, что неправильное поведение приведет к будущему отказу, рисунок 2. Например, если обнаруживается, что база данных недоступна, анализ первопричин пытается определить причину недоступности: обрыв сетевого подключения или измененная конфигурация и т.д.

С другой стороны, прогнозирование отказов пытается оценить, несет ли эта ситуация риск того, что система не сможет предоставить ожидаемый сервис, что может зависеть от характеристик системы, модели прогнозирования отказов и текущей ситуации: существует ли резервная база данных или какой-либо другой механизм отказоустойчивости? Какова текущая нагрузка на систему? Эти вопросы фокусируются только на прогнозировании сбоев.

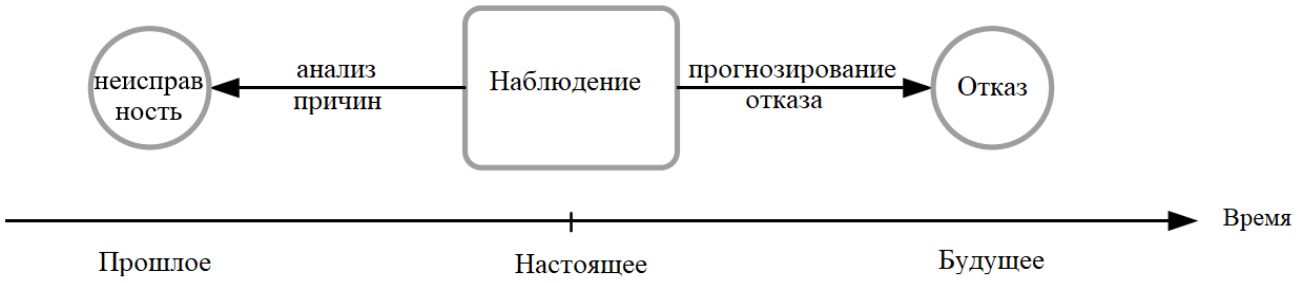

Рисунок 2 – Различие между анализом первопричин и прогнозированием отказов

Производство становится все более прозрачным, универсальным и гибким благодаря плавному взаимодействию физического и виртуального производственного мира. В некоторой степени модули производственной линии производят детали автоматически. Другие производятся людьми, но при поддержке ма-

шин. В любом случае, это может привести к сбоям на технологических линиях и поэтому не маловажно использовать методы машинного обучения, с целью предотвращения этих сбоев.

Выводы по главе 1

Рассмотрено понятие сборочной линии. Сборочная линия – это производственный процесс, в котором взаимозаменяемые части последовательно добавляются в продукт для создания конечного продукта.

Существуют различные вариации сборочных линий:

- модульная сборка;
- производство ячеек;
- командное производство;
- U-образная сборочная линия.

Были рассмотрены признаки отказов производственных линий и их последствия. Рассмотрены наиболее важные правила, которые необходимо учитывать при создании системы контроля качества при изготовлении производственной линии.

Исходя из исследования предметной области, следует сделать вывод, что для решения задачи прогнозирования производственных сбоев на технологических линиях эффективно использовать методы машинного обучения.

2. МЕТОДЫ КЛАССИФИКАЦИИ ПРОГНОЗИРОВАНИЯ СБОЕВ ТЕХНОЛОГИЧЕСКИХ ЛИНИЙ

В машинном обучении задачи обычно делятся на широкие категории. Эти категории основаны на том, как обучение получено или как обратная связь по обучению предоставляется разработанной системе.

Двумя наиболее широко распространенными категориями машинного обучения являются контролируемое обучение, которое обучает алгоритмы на основе примерных входных и выходных данных, помеченных людьми, и неконтролируемое обучение, которое обеспечивает алгоритм без помеченных данных, чтобы позволить ему находить структуру в своих входных данных.

Кто-то может спросить: «Почему машины должны учиться? Почему бы не проектировать машины так, чтобы они работали так, как хотелось бы?» Существует несколько причин, по которым машинное обучение важно. Конечно, достижение обучения на машинах может помочь понять, как животные и люди учатся. Но есть и важные инженерные причины.

Вот некоторые из них:

1. Некоторые задачи не могут быть четко определены, кроме как на примере; то есть мы могли бы указать пары ввода / вывода, но не точную связь между входами и желаемыми выходами. Хотелось бы, чтобы машины могли настраивать свою внутреннюю структуру для получения правильных выходных данных для большого количества выборочных входов и, таким образом, соответствующим образом ограничивать их функцию ввода / вывода для аппроксимации отношения, неявного в примерах.

2. Возможно, что среди больших куч данных скрыты важные взаимосвязи и корреляции. Методы машинного обучения часто могут использоваться для извлечения этих отношений (интеллектуальный анализ данных).

Люди-дизайнеры часто производят машины, которые не работают так, как хотелось бы, в условиях, в которых они используются. Фактически, определенные характеристики рабочей среды могут быть не полностью известны во время разработки. Методы машинного обучения могут быть использованы для улучшения существующих конструкций машин.

– объем доступных знаний об определенных задачах может быть слишком большим для явного кодирования людьми. Машины, которые изучают это знание постепенно, могут захватить его больше, чем люди захотят записать;

– среда меняется со временем. Машины, которые могут адаптироваться к изменяющейся среде, уменьшат необходимость в постоянном перепроектировании;

– новые знания о задачах постоянно открываются людьми.

Изменения словарного запаса. В мире постоянно происходит поток новых событий. Продолжать модернизацию систем искусственного интеллекта для соответствия новым знаниям нецелесообразно, но методы машинного обучения могут быть в состоянии отследить большую часть этого.

2.1 Методы классификации, используемые в машинном обучении

Обучение, как и интеллект, охватывает такой широкий спектр процессов, которые трудно точно определить. Словарное определение включает в себя такие фразы, как «чтобы получить знания, или понимание, или навыки, путем изучения, обучения или опыта» и «изменение поведенческой тенденции с помощью опыта». Зоологи и психологи изучают обучение на животных и людях. В данной работе внимание будет сосредоточено на обучении в машинах. Существует несколько параллелей между обучением животных и машин. Конечно, многие методы машинного обучения основаны на попытках психологов уточнить свои теории обу-

чения животных и человека с помощью вычислительных моделей. Представляется также вероятным, что концепции и методы, изучаемые исследователями в области машинного обучения, могут пролить свет на некоторые аспекты биологического обучения [17, 49].

Что касается машин, можно очень широко сказать, что машина учится всякий раз, когда она меняет свою структуру, программу или данные (на основе своих входных данных или в ответ на внешнюю информацию) таким образом, что улучшается ожидаемая в будущем производительность. Некоторые из этих изменений, такие как добавление записи в базу данных, удобно укладываются в область других дисциплин и не обязательно лучше понимаются, поскольку их называют обучением. Но, например, когда производительность машины для распознавания речи улучшается после прослушивания нескольких образцов речи человека, в этом случае все ощущается вполне оправданным для того, чтобы сказать, что машина научилась. Машинное обучение обычно относится к изменениям в системах, которые выполняют задачи, связанные с искусственным интеллектом (ИИ). Такие задачи включают распознавание, диагностику, планирование, управление роботом, прогнозирование и т.д [17].

«Изменения» могут быть либо улучшением уже работающих систем, либо синтезом новых систем. Эта система воспринимает и моделирует свое окружение и вычисляет соответствующие действия, возможно, предвидя их последствия. Изменения, внесенные в любой из компонентов, могут учитываться как обучение. Различные механизмы обучения могут быть использованы в зависимости от того, какая подсистема изменяется. В данной работе будет рассмотрено изучение нескольких различных методов обучения, рассмотрим каждое поподробнее.

Как область машинное обучение тесно связано с вычислительной статистикой, поэтому наличие базовых знаний в области статистики полезно для понимания и использования алгоритмов машинного обучения [49].

Для тех, кто, возможно, не изучал статистику, может быть полезно сначала определить корреляцию и регрессию, поскольку они обычно используются для

исследования взаимосвязи между количественными переменными. Корреляция – это мера связи между двумя переменными, которые не обозначены как зависимые или независимые. Регрессия на базовом уровне используется для изучения взаимосвязи между одной зависимой и одной независимой переменной. Поскольку статистика регрессии может использоваться для прогнозирования зависимой переменной, когда независимая переменная известна, регрессия обеспечивает возможности прогнозирования [20].

Методы машинного обучения постоянно развиваются. Исходя из исследованных работ, рассмотрим несколько популярных алгоритмов, которые используются в машинном обучении для решения такого рода проблем.

2.1.1 k Ближайших Соседей (k Nearest Neighbor)

kNN расшифровывается как k Nearest Neighbor или k Ближайших Соседей — это один из самых простых алгоритмов классификации, также иногда используемый в задачах регрессии. Благодаря своей простоте, он является хорошим примером, с которого можно начать знакомство с областью Machine Learning [13].

В машинном обучении задача классификации – это задача отнесения объекта к одному из заранее определенных классов на основании его формализованных признаков [20]. Каждый из объектов в этой задаче представляется в виде вектора в N-мерном пространстве, каждое измерение в котором представляет собой описание одного из признаков объекта. Допустим, нужно классифицировать мониторы: измерениями в пространстве параметров будут величина диагонали в дюймах, соотношение сторон, максимальное разрешение, наличие HDMI-интерфейса, стоимость и др. Случай классификации текстов несколько сложнее, для них обычно используется матрица термин-документ.

Для обучения классификатора необходимо иметь набор объектов, для которых заранее определены классы. Это множество называется обучающей выборкой, её разметка производится вручную, с привлечением специалистов в исследуемой области. Например, в задаче Detecting Insults in Social Commentary для заранее со-

бранных тестов комментариев человеком проставлено мнение, является ли этот комментарий оскорблением одного из участников дискуссии, само же задание является примером бинарной классификации. В задаче классификации может быть более двух классов (многоклассовая), каждый из объектов может принадлежать более чем к одному классу (пересекающаяся) [41].

Для классификации каждого из объектов тестовой выборки необходимо последовательно выполнить следующие операции:

– вычислить расстояние до каждого из объектов обучающей выборки

– отобрать k объектов обучающей выборки, расстояние до которых минимально

– класс классифицируемого объекта — это класс, наиболее часто встречающийся среди k ближайших соседей

kNN — один из простейших алгоритмов классификации, поэтому на реальных задачах он зачастую оказывается неэффективным. Помимо точности классификации, проблемой этого классификатора является скорость классификации: если в обучающей выборке N объектов, в тестовом выборе M объектов, а размерность пространства — K, то количество операций для классификации тестовой выборки может быть оценено как O(K*M*N). Тем не менее, алгоритм работы kNN является хорошим примером для начала знакомства с Machine Learning [41].

2.1.2 Случайный лес (Random forest)

Случайный лес — один из самых потрясающих алгоритмов машинного обучения, придуманные Лео Брейманом и Адель Катлер ещё в прошлом веке. Он дошёл до нас в «первозданном виде» (никакие эвристики не смогли его существенно улучшить) и является одним из немногих универсальных алгоритмов. Универсальность заключается, во-первых, в том, что он хорош во многих задачах (по оценкам, 70% из встречающихся на практике, если не учитывать задачи с изображениями), во-вторых, в том, что есть случайные леса для решения задач классификации, регрессии, кластеризации, поиска аномалий, селекции признаков и т.д.

RF (random forest) — это множество решающих деревьев. В задаче регрессии их ответы усредняются, в задаче классификации принимается решение голосованием по большинству. Все деревья строятся независимо по следующей схеме:

1. Выбирается подвыборка обучающей выборки размера samplesize (м.б. с возвращением) – по ней строится дерево (для каждого дерева — своя подвыборка).

2. Для построения каждого расщепления в дереве просматриваем max_features случайных признаков (для каждого нового расщепления — свои случайные признаки).

3. Выбираем наилучшие признак и расщепление по нему (по заранее заданному критерию). Дерево строится, как правило, до исчерпания выборки (пока в листьях не останутся представители только одного класса), но в современных реализациях есть параметры, которые ограничивают высоту дерева, число объектов в листьях и число объектов в подвыборке, при котором проводится расщепление.

Понятно, что такая схема построения соответствует главному принципу ансамблирования (построению алгоритма машинного обучения на базе нескольких, в данном случае решающих деревьев): базовые алгоритмы должны быть хорошими и разнообразными (поэтому каждое дерево строится на своей обучающей выборке и при выборе расщеплений есть элемент случайности).

Метод RF хорош ещё тем, что при построении леса параллельно может вычисляться т.н. oob-оценка качества алгоритма (которая очень точная и получается не в ущерб разделения на обучение/тест), oob-ответы алгоритмы (ответы, которые выдавал бы алгоритм на обучающей выборке, если бы «обучался не на ней»), оцениваются важности признаков. Также не стоит забывать про полный перебор значений параметров (если объектов в задаче не очень много) [18].

2.1.3 Классификатор экстра-деревьев (Extra-Trees Classifier)

Метод Extra-Trees Classifier, был предложен с главной целью дальнейшего построения дерева рандомизации в контексте числовых признаков ввода, где выбор

оптимальной точки пересечения отвечает для значительной части дисперсии индуцированного дерева.

Что касается случайных лесов, метод исключает идею использования загрузочных копий учебного образца, и вместо того, чтобы пытаться найти оптимальную точку пересечения для каждой из случайно выбранных признаков K на каждом узле, он выбирает точку пересечения наугад.

Эта идея довольно продуктивна в контексте многих проблем, характеризующихся большим числом числовых признаков, изменяющихся более или менее непрерывно: она часто приводит к повышенной точности благодаря ее сглаживанию и в то же время значительно снижает вычислительное бремя, связанное с определением оптимальных срезы в стандартных деревьях и в случайных лесах.

С статистической точки зрения, отбрасывание идеи начальной загрузки приводит к преимуществу с точки зрения смещения, тогда как рандомизация с режущей средой часто является отличным эффектом уменьшения дисперсии. Этот метод позволил получить самые современные результаты в нескольких многомерных сложных задачах.

С функциональной точки зрения, метод Extra-Tree создает кусочно-полилинейные аппроксимации, а не кусочно-постоянные из случайных лесов [20, 21].

2.1.4 Градиентный бустинг (XGBoost)

XGBoost – это контролируемый алгоритм обучения, который реализует процесс, называемый boosting, чтобы дать точные модели. Boosting относится к методу обучения ансамблю для построения многих моделей последовательно, причем каждая новая модель пытается исправить недостатки предыдущей модели. В повышении дерева каждая новая модель, добавленная в ансамбль, является деревом решений. XGBoost обеспечивает параллельное наращивание дерева (также известное как GBDT, GBM), которое быстро и точно решает многие проблемы с

наукой о данных. Для многих проблем XGBoost – одна из лучших рамок ускорителя градиента (GBM) сегодня [10, 21].

Возможности XGBoost – особенности модели и системные функции;

Реализация модели поддерживает особенности реализации scikit-learn и R с новыми дополнениями, такими как регуляризация. Поддерживаются три основные формы повышения градиента:

1. Алгоритм Gradient Boosting также называется градиентной машиной повышения, включая скорость обучения;

2. Stochastic Gradient Boosting с суб-выборкой в строке, столбце и столбце на каждый уровень разделения;

3. Регулярное усиление градиента с регуляцией L1 и L2.

Библиотека предоставляет систему для использования в различных вычислительных средах, не в последнюю очередь:

1. Параллелизация построения дерева с использованием всех ваших ядер процессора во время обучения;

2. Распределенные вычисления для обучения очень крупных моделей с использованием кластера машин;

3. Вне корпоративного вычисления для очень больших наборов данных, которые не вписываются в память;

4. Кэш Оптимизация структуры данных и алгоритма для наилучшего использования аппаратного обеспечения.

Реализация алгоритма была разработана для эффективности вычислительных ресурсов времени и памяти. Цель проекта заключалась в том, чтобы наилучшим образом использовать имеющиеся ресурсы для обучения модели. Некоторые ключевые функции реализации алгоритма включают:

1. Редкая реализация Aware с автоматической обработкой отсутствующих значений данных;

2. Блочная структура для поддержки распараллеливания конструкции дерева;

3. Продолжение обучения, чтобы вы могли еще больше повысить уже установленную модель для новых данных.

2.1.5 Легкий градиентный бустинг (LightGBM)

Что такое легкий градиентный бустинг – LigthGBM. LightGBM – это быстрая, распределенная, высокопроизводительная структура повышения градиента, основанная на алгоритме дерева решений, используемая для ранжирования, классификации и многих других задач машинного обучения [21].

Поскольку он основан на алгоритмах дерева решений, он разделяет лист дерева с наилучшим соответствием, тогда как другие алгоритмы повышения делят дерево по глубине или уровню, а не по листу. Таким образом, при выращивании на одном и том же листе в LightGBM, листовой алгоритм может уменьшить больше потерь, чем алгоритм по уровням, и, следовательно, приводит к гораздо лучшей точности, что редко может быть достигнуто любым из существующих алгоритмов бустинга. Кроме того, это на удивление очень быстро, отсюда и слово «свет».

Ниже продемонстрировано схематичное представление создателей LightGBM, чтобы четко объяснить разницу, рисунки 3,4.

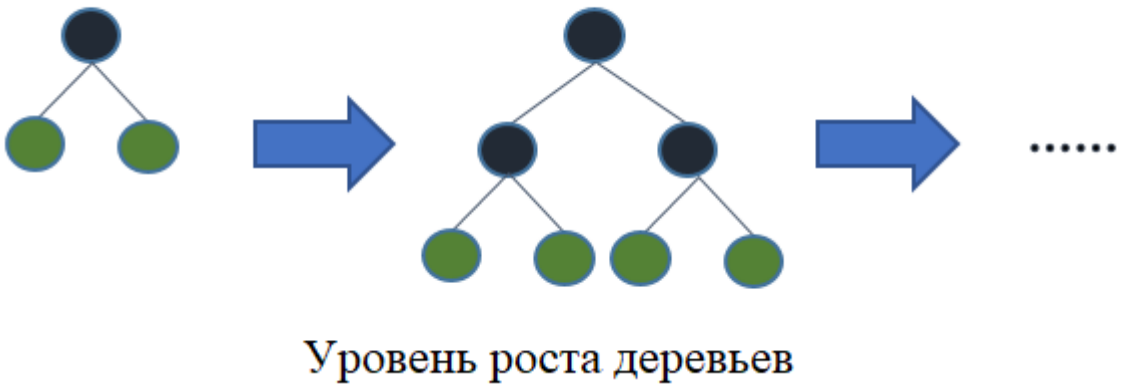

Рисунок 3 – Выращивание деревьев по уровням в XGBOOST

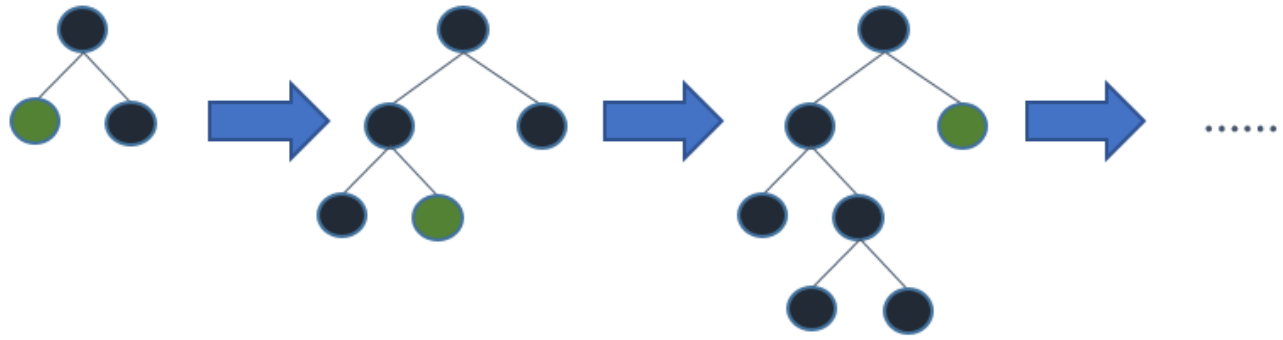

Рисунок 4 – Листовой рост деревьев в LightGBM

Листовое разбиение приводит к увеличению сложности и может привести к переоснащению, и его можно преодолеть, указав другой параметр max-depth, который определяет глубину, на которую будет происходить разбиение. Давайте посмотрим на некоторые преимущества Light GBM [21].

Преимущества Light GBM:

Более быстрая скорость обучения и более высокая эффективность: LightGBM использует алгоритм, основанный на гистограмме, то есть он объединяет непрерывные значения признаков в дискретные ячейки, которые ускоряют процедуру обучения.

Меньшее использование памяти: заменяет непрерывные значения дискретными ячейками, что приводит к меньшему использованию памяти.

Лучшая точность, чем у любого другого алгоритма бустинга: он создает гораздо более сложные деревья, следуя подходу разбиения листа, а не подходу уровня, который является основным фактором достижения более высокой точности. Однако иногда это может привести к переоснащению, которого можно избежать, установив параметр max_depth.

Совместимость с большими наборами данных – он способен одинаково хорошо работать с большими наборами данных со значительным сокращением времени обучения по сравнению с XGBoost. Также, присутствует возможность параллельного обучения.

2.2 Метрики качества

2.2.1 Метрика – ROC-AUC

Многие предикторы отказов включают в себя настраиваемый порог принятия решения, при котором выдается предупреждение о сбое или нет. Если пороговое значение низкое, предупреждение о сбое возникает очень легко, что увеличивает вероятность обнаружения истинного сбоя (что приводит к высокому отзыву). Однако низкий порог также приводит ко многим ложным тревогам, что приводит к низкой точности. Если порог очень высок, ситуация наоборот: точность хороша, а отзыв низок. Кривые точности / отзыва используются для визуализации этого компромисса путем построения графика зависимости точности от отзыва для различных пороговых уровней. Графики иногда также называют положительными прогностическими значениями / графиками чувствительности. Пример показан на рисунке 5.

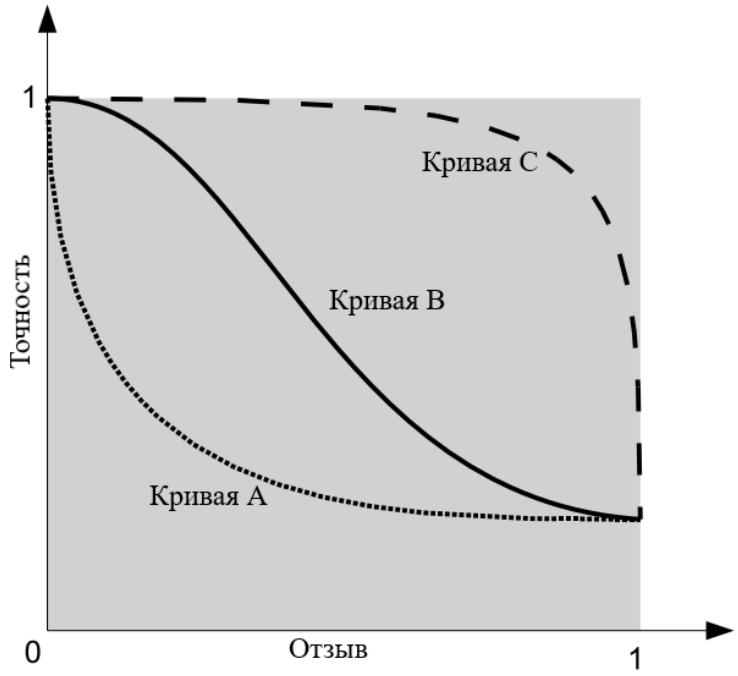

Рисунок 5 – Выборочные кривые точности / отзыва, отображающие компромисс между ними

Кривая A показывает предиктор, который работает довольно плохо: нет точки, в которой точность и отзыв одновременно имеют высокие значения. Метод про-

гнозирования отказов, изображенный кривой B, работает немного лучше. Кривая C отражает алгоритм, прогнозы которого в основном верны.

В машинном обучении измерение производительности является важной задачей. Поэтому, когда дело доходит до проблемы классификации, можно рассчитывать на кривую AUC-ROC. Когда нужно проверить или визуализировать производительность задачи классификации нескольких классов, то используем кривую AUC (область под кривой), ROC (рабочие характеристики приемника). Это один из наиболее важных показателей оценки для проверки производительности любой модели классификации. Он также записывается как ROC/AUC (область под рабочими характеристиками приемника).

Кривая AUC-ROC – это измерение производительности для задачи классификации при различных настройках порогов. ROC – это кривая вероятности, а AUC – степень или мера отделимости. Он рассказывает, насколько модель способна различать классы. Чем выше AUC, тем лучше модель при прогнозировании 0 с 0 и 1 с 1 с. По аналогии, чем выше AUC, тем лучше модель для различения пациентов с болезнью и без болезни.

Кривая ROC строится с TPR (True Positive Rate) относительно FPR (False Positive Rate), где TPR находится на оси y, а FPR на оси x.

Подобно кривым точности / возврата, кривая рабочих характеристик приемника (ROC). Рисунок 6 отображает истинную положительную скорость в сравнении с ложной положительной скоростью (чувствительность / отзыв в сравнении с «1-специфичность» соответственно) и, следовательно, позволяет оценить способность модели различать отказы и безотказность. Чем ближе кривая подходит к верхнему левому углу пространства ROC, тем точнее модель. Поскольку кривые ROC достигаются для всех пороговых значений, точность методов прогнозирования может быть легко оценена путем сравнения их кривых ROC: Площадь под кривой (AUC) определяется как область между кривой ROC и осью X.

AUC рассчитывается по формуле 1.

$$AUC = Z\ 1\ 0\ tpr\ (fpr)dfpr \quad \in [0,1], \tag{1}$$

где tpr – истинная положительная скорость;

fpr – ложная положительная скорость.

По сути, AUC – это вероятность того, что точка данных в ситуации, склонной к сбоям, получит более высокий балл, чем точка данных в ситуации, не склонной к сбоям. Поскольку AUC превращает кривую ROC в единое число, измеряя площадь под кривой ROC, она суммирует внутреннюю способность алгоритма прогнозирования различать сбои и не сбои. Случайный предиктор получает AUC 0,5 (инверсия не всегда верна, см., например, [Flach 2004]), в то время как идеальный предиктор дает AUC единицы.

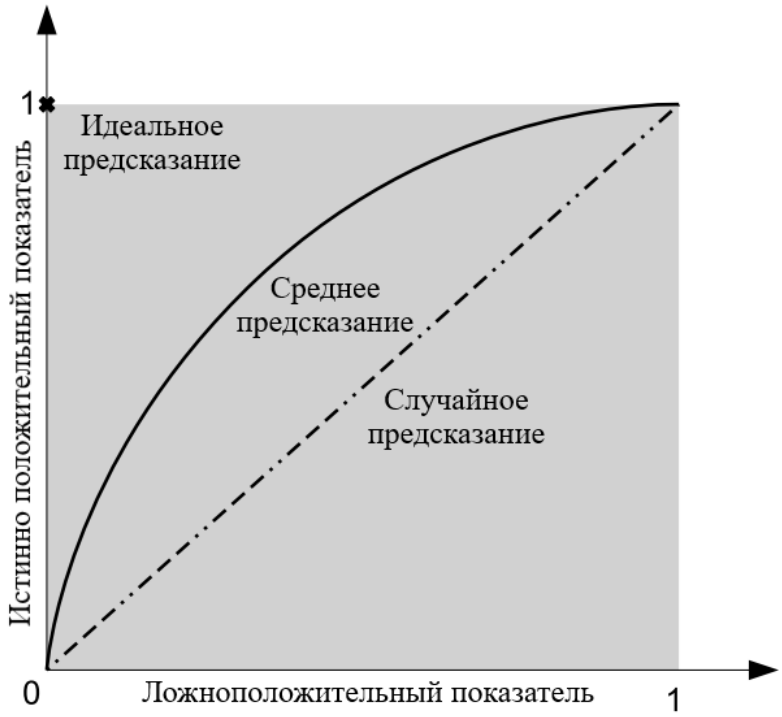

Рисунок 6 – Образцы ROC-графиков

2.2.2 Коэффициент корреляции Мэтьюса (MCC)

Идеальный предиктор ошибок показывает истинную положительную частоту – 1 и ложную положительную частоту – 0. Многие существующие предикторы облегчают настройку компромисса между истинно положительной ставкой и ложноположительной ставкой, как показано сплошной линией. Диагональ показывает случайный предиктор: в каждой точке вероятность ложного или истинного положительного прогноза равна. Поскольку набор данных сильно разбалансиро-

ван по отношению к переменной отклика (171-на правильная выборка для каждого сбоя), простая потеря 0 – 1 будет бессмысленной. Поэтому следует использовать коэффициент корреляции Мэтьюса (MCC), в качестве меры качества двоичных классификаторов.

MCC – это в основном коэффициент корреляции между предсказанным и наблюдаемым классами. Коэффициент +1 представляет собой идеальный прогноз, коэффициент 0 представляет собой не лучше, чем случайный прогноз, в то время как коэффициент -1 означает полное несоответствие между прогнозируемыми и наблюдаемыми значениями. MCC проводится по формуле 2:

$$MCC = \frac{(TP \cdot TN) - (FP \cdot FN)}{\sqrt{(TP + FP) \cdot (TP + FN) \cdot (TN + FP) \cdot (TN + FN)}}, \quad (2)$$

где TP – истинные положительные значения;

TN – истинные отрицательные значения;

FP – ложные положительные значения;

FN – ложные отрицательные значения.

2.3 Процедура – Кросс-валидация

Чтобы исследовать качество алгоритмов прогнозирования отказов и сравнивать их потенциал, необходимо указать метрики (показатели качества). Целью прогнозирования отказов является точное прогнозирование отказов: покрытие максимально возможного количества отказов и одновременное генерирование как можно меньшего количества ложных срабатываний. Идеальное прогнозирование сбоев позволит достичь однозначного соответствия между прогнозируемыми и истинными сбоями. В этом разделе будут представлены несколько установленных метрик для достоверности соответствия прогнозирования. Были предложены некоторые другие метрики, например, статистика каппа [Altman 1991, стр. 404], но они редко используются сообществом. Более подробное обсуждение и анализ показателей оценки для прогнозирования сбоев в Интернете можно найти в [Salfner 2008, глава 8.2].

Таблица 1 определяет четыре случая: Прогноз сбоя является истинно положительным, если сбой происходит в течение периода прогнозирования и выдается предупреждение о сбое. Если сбой не происходит и выдается предупреждение, прогноз является ложноположительным. Если алгоритм не может предсказать истинную ошибку, это ложный минус. Если не происходит никакого истинного сбоя и не выдается предупреждение о сбое, прогноз является истинно отрицательным.

Таблица 1 – Таблица сопряженности

	Отказ – правда	Отказ – не правда	Сумма
Прогноз: отказ (предупреждение о сбое)	Истинно положительный (правильное предупреждение)	Ложноположительный (ложное предупреждение)	Положительный
Прогноз: нет ошибок (нет предупреждения о сбое)	Ложноотрицательный (отсутствует предупреждение)	Истинно отрицательный (правильно, без предупреждения)	Отрицательный
Сумма	Отказы	Без сбоев	Всего

Любой прогноз сбоя относится к одному из четырех случаев: если алгоритм прогнозирования принимает решение в пользу предстоящего сбоя, который называется положительным, это приводит к выдаче предупреждения о сбое. Это решение может быть правильным или неправильным. Если на самом деле неудача неизбежна, прогноз является действительно позитивным. Если нет, ложный положительный результат. Аналогично, если прогноз решает, что система работает хорошо (негативный прогноз), этот прогноз может быть правильным (истинно отрицательным) или неправильным (ложно отрицательным).

Для оценки показателей, обсуждаемых в этом разделе, необходим набор справочных данных, для которого он известен, когда произошли сбои. В машинном обучении это называется «помеченным набором данных». Поскольку метрики

оценки определяются с использованием статистических оценок, набор данных должен быть максимально большим. Однако сбои – это, как правило, редкие события, которые обычно устанавливают естественное ограничение на количество сбоев в наборе данных.

Если метод онлайн-прогнозирования предполагает оценку параметров на основе данных, набор данных должен быть разделен на три части:

1. Набор обучающих данных: данные, по которым выполняется оптимизация параметров;

2. Набор данных проверки. В случае, если алгоритм оптимизации параметров может привести к локальным, а не глобальным оптимумам, или для контроля так называемого компромисса с биасвариацией, данные проверки используются для выбора наилучшего параметра;

3. Набор тестовых данных: оценка эффективности прогнозирования отказов выполняется на данных, которые не использовались для определения параметров метода прогнозирования. Такая оценка также называется оценкой вне выборки.

Чтобы определить количество прогнозов TP, FP, FN и TN, необходимых для заполнения таблицы сопряженности и последующего вычисления таких метрик, как точность и повторный вызов, алгоритм прогнозирования применяется к тестовым данным, и результаты прогнозирования сравниваются с истинными возникновениями сбоев. Четыре случая, которые могут возникнуть, изображены на рисунке 7.

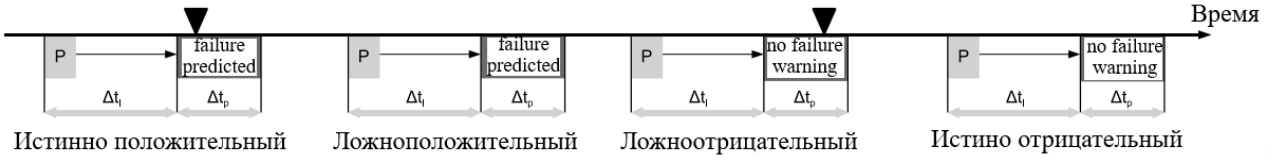

Рисунок 7 – Временная шкала, показывающая истинные ошибки набора тестовых данных (▼) и все четыре типа прогнозов: TP, FP, FN, TN.

Как видно из рисунка, период прогнозирования Δtp используется для определения того, считается ли сбой прогнозируемым или нет. Следовательно, выбор

Δtp влияет на таблицу непредвиденных обстоятельств и должен выбираться в соответствии с требованиями для последующих этапов упреждающего управления отказами.

Отказ засчитывается как предсказанный, если он происходит в течение периода прогнозирования длительностью Δtp, который начинается во время выполнения заказа Δtl после начала прогнозирования P.

Для определения таких кривых, как кривые точности / повторного вызова или графики ROC, следует сохранять рейтинг предикторов, а не двоичное решение на основе порогов, что позволяет генерировать кривую для всех возможных пороговых значений с использованием алгоритма, такого как описанный в [Fawcett 2004].

Оценка показателей из конечного набора тестовых данных дает только приблизительную оценку эффективности прогнозирования и, следовательно, должна сопровождаться доверительными интервалами. Доверительные интервалы обычно оцениваются путем выполнения процедуры оценки несколько раз. Поскольку для этого требуется огромное количество данных, применяются такие методы, как перекрестная проверка, складной нож или начальная загрузка. Более подробное обсуждение таких методов можно найти в [Salfner 2008, глава 8.4].

Самый простой метод, который можно использовать для оценки производительности алгоритма машинного обучения – это использование различных наборов данных для обучения и тестирования.

Можно взять оригинальный набор данных и разделить его на две части. Обучить алгоритм в первой части, затем сделать прогнозы во второй части и сравнить прогнозы с ожидаемыми результатами. Размер разделения может зависеть от размера и специфики набора данных, хотя для обучения обычно используется 67% данных, а для тестирования – 33%.

Этот метод оценки алгоритма довольно быстр. Он идеально подходит для больших наборов данных (миллионы записей), где имеются убедительные доказательства того, что оба разделения данных представляют основную проблему. Из-

за скорости полезно использовать этот подход, когда алгоритм, который исследуется, обучается медленно.

Недостатком этого метода является то, что он может иметь высокую дисперсию. Это означает, что различия в обучающем и тестовом наборе данных могут привести к значительным различиям в оценке точности модели.

Можно разделить набор данных на train и test, используя функцию train_test_split () из библиотеки scikit-learn. Например, можно разделить набор данных на 67% и 33% для train и test следующим образом:

X_train, X_test, y_train, y_test = train_test_split (X, Y, test_size=0.33, random_state=7)

Кросс-валидация, перекрестная проверка – это подход, который можно использовать для оценки производительности алгоритма машинного обучения с меньшей дисперсией, чем в случае разделения набора из одного train.

Он работает, разбивая набор данных на k-части (например, k = 3; 5 или 10). Каждое разделение данных называется фолдом. Алгоритм обучен на k-1 фолд с одним сдержанным и проверен на согнутом фолде. Это повторяется, так что каждому фолду набора данных дается возможность быть сдержанным тестовым набором.

После выполнения перекрестной проверки, получаем k различных показателей производительности, которые можно суммировать, используя среднее значение и стандартное отклонение.

Результатом является более надежная оценка производительности алгоритма на новых данных с учетом тестовых данных. Это более точный прогноз, потому что алгоритм обучается и оценивается несколько раз на разных данных.

Выбор k должен позволить размеру каждого тестового раздела быть достаточно большим, чтобы быть разумной выборкой проблемы, в то же время допуская достаточное количество повторений оценки алгоритма «train-test», чтобы обеспечить справедливую оценку производительности алгоритмов на невидимых дан-

ных. Для наборов данных небольшого размера в тысячах или десятках тысяч наблюдений, значения k 3, 5 и 10 являются общими.

Можно использовать поддержку перекрестной проверки в k-кратном порядке, предоставляемую в scikit-learn. Сначала должны создать объект KFold, указав количество фолдов и размер набора данных. Затем можно использовать эту схему с конкретным набором данных. Функция cross_val_score () из scikit-learn позволяет оценивать модель, используя схему перекрестной проверки, и возвращает список баллов для каждой модели, обученной в каждом фолде. Пример кода:

```
kfold = KFold (n_splits=10, random_state=7)
results = cross_val_score (model, X, Y, cv=kfold)
```

Выполнение этого примера суммирует производительность конфигурации модели по умолчанию в наборе данных, включая точность классификации как среднего, так и стандартного отклонения.

Если есть много классов для задачи прогнозирующего моделирования классификационного типа, или классы несбалансированы (для одного класса существует гораздо больше экземпляров, чем для другого), то при выполнении перекрестной проверки, может быть хорошей идеей создавать стратифицированные фолды.

Это приведет к принудительному распределению классов в каждом сгибе, как и во всем наборе обучающих данных, при выполнении перекрестной оценки. Библиотека scikit-learn предоставляет эту возможность в классе StratifiedKFold.

Когда и какой метод использовать:

1) Как правило, перекрестная проверка в k-fold, является золотым стандартом для оценки производительности алгоритма машинного обучения для невидимых данных с k, равным 3, 5 или 10;

2) Использовать стратифицированную перекрестную проверку для обеспечения распределения классов следует тогда, когда существует большое количество классов или дисбаланс в экземплярах для каждого класса;

3) Использование разделения train/test полезно для скорости при использовании медленного алгоритма и дает оценки производительности с меньшим смещением при использовании больших наборов данных.

Выводы по главе 2

Методы машинного обучения постоянно совершенствуются. Рассмотрены несколько популярных алгоритмов, которые используются в машинном обучении:

– kNN расшифровывается как k Nearest Neighbor или k Ближайших Соседей;
– Случайный лес – RF (random forest);
– Extra-Trees (стоящий для extremely randomized trees);
– XGBoost;
– Light GBM.

Были рассмотрены метрики качества, такие как:

– Кривая ROCAUC – это измерение производительности для задачи классификации при различных настройках порогов.
– MCC – это в основном коэффициент корреляции между предсказанным и наблюдаемым классами.

Использован самый распространенный метод для оценки производительности алгоритма машинного обучения – это использование различных наборов данных train/test, а также деление на k фолды. А также, рассмотрена кросс-валидация, которая будет использована в дальнейшей работе. Это перекрестная проверка – подход, который можно использовать для оценки производительности алгоритма машинного обучения с меньшей дисперсией, чем в случае разделения набора из одного train.

3. ИССЛЕДОВАНИЕ ЭФФЕКТИВНОСТИ КЛАССФИКАТОРОВ ДЛЯ ПРОГНОЗИРОВАНИЯ СБОЕВ ТЕХНОЛОГИЧЕСКИХ ЛИНИЙ НА ПРИМЕРЕ ДАННЫХ КОМПАНИИ BOSCH

3.1 Набор данных

Задача – Уменьшить производственные сбои.

Компания Bosch предоставила огромный набор данных, представляющий 1 измерение деталей, когда они перемещаются по производственным линиям. Каждая часть имеет уникальный идентификатор. Цель состоит в том, чтобы предсказать, будет ли конкретная деталь (идентифицируемая уникальным идентификатором) проходить контроль качества или нет (представленные как «Ответ» = 1).

Для лучшего хранения и более легкого понимания этих огромных данных, данные были разделены на разные файлы в зависимости от типа признака, которую они содержат, а именно: числовых, категориальных и признаков даты. Названия объектов называются в соответствии с соглашением, в котором указывается производственная линия, станция на линии и номер элемента. Например, L2_S46_F4242 изображает признак, измеренный в строке 2 на станции 46 и номер признака 4242. Общий обзор размеров наборов данных показан в таблице 2. Общий обзор системы продемонстрирован на рисунке 8.

Таблица 2 – Общий обзор размеров наборов данных

	Размер файла (сжатый), MB	Размер файла (распакованный), GB	Ряд	Колонка
Категориальный (train/test)	20	2.50	118 378	2 141
Числовой (train/test)	270	2	118 378	1 157
Дата (train/test)	59	2.7	118 378	970

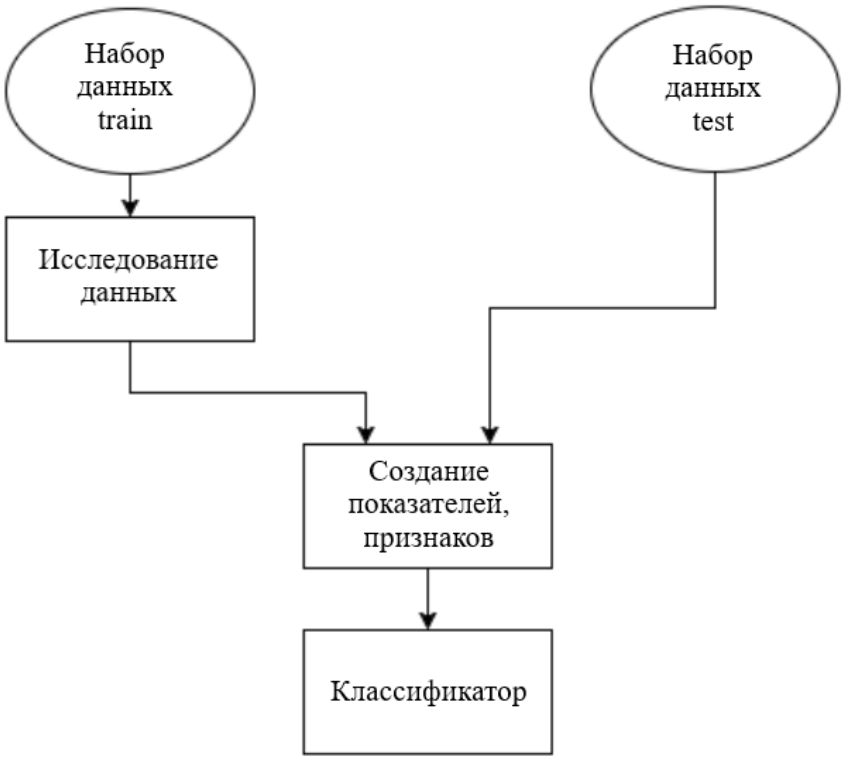

Рисунок 8 – Общий обзор системы

Признаки даты предоставляют метку времени для каждого измерения. Каждый столбец даты заканчивается номером, который соответствует предыдущему номеру признака. Например, значение L0_S0_D1 – это время, когда был взят L0_S0_F0.

Помимо того, что он является одним из крупнейших наборов данных (с точки зрения количества признаков), основополагающая истина для этого решения крайне несбалансирована. Вместе эти два атрибута, как ожидается, сделают эту проблему сложной.

Файлы с данными предоставленные компанией BOSCH:

1. train_numeric.csv – числовые признаки обучающего набора (этот файл содержит переменную Response);
2. test_numeric.csv – набор тестовых числовых признаков (предстоит предсказать «ответ» для этих идентификаторов);
3. train_categorical.csv – обучающий набор категориальных признаков;
4. test_categorical.csv – тестовый набор категориальных признаков;

5. train_date.csv – особенности даты тренировочного набора;
6. test_date.csv – признак проверки набора даты;
7. sample_submission.csv – файл с примерами представления в правильном формате.

На рисунке 9 представлена таблица построенная на основе данных из файла «train_categorical.csv».

Id	L0_S1_F25	L0_S1_F27	L0_S1_F29	L0_S1_F31	L0_S2_F33	L0_S2_F35	L0_S2_F37	L0_S2_F39	L0_S2_F41	L0_S2_F43	...	L3_S49_F4225	L3_S49_F4227
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
6	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
7	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
9	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
11	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN

5 rows × 2140 columns

Рисунок 9 – Таблица представленная данными из файла train_categorical.csv

После того как изучили набор данных стоит сделать выбор в пользу того алгоритма, который будет наилучшим выбором для решения данной задачи. Несомненно, для начала следует разобраться в предварительной обработке данных.

3.2 Предварительная обработка данных

Bosch предоставил огромный набор данных (14,3 Гб), содержащий три типа данных объектов: числовые, категориальные, отметки даты и метки, обозначающие деталь как хорошую или плохую. Учебные данные имеют 1 184 687 выборок (наблюдений), а изученная модель будет использоваться для прогнозирования на тестовом наборе данных, содержащем 1 183 748 выборок. Есть 968 числовых признаков, 2 140 категориальных признаков и 1 156 признаков даты. Следовательно, одна из самых больших проблем этого набора данных состоит в том, чтобы преобразовать эти признаки во что-то значимое, чтобы их можно было использовать для создания прогнозной модели.

Обучающий набор данных сам по себе довольно большой, с очень большим количеством признаков, поэтому очень важно тщательно изучить и понять набор данных. Поскольку никакой предварительной информации о признаках не извест-

но, нужно попытаться найти соответствующие связи между признаками после изучения интересной информации в наборе данных. Получив представление о наборе данных, был выбран лучший набор признаков, которые объясняют набор данных больше всего, и на том же уровне следует поработать над построением классификатора.

1. Категориальные особенности:

Категориальные данные имеют 2 140 признаков, но при дальнейшей оценке обнаружено, что около 500 являются многозначными, 1 490 однозначными и 150 пустыми. Пустые категориальные признаки могут быть отброшены, поскольку они не содержат информации. Однозначные и многозначные категориальные признаки могут быть преобразованы в числовые с использованием метода One-hot кодирования, где каждый класс представлен целым числом. One-hot кодирование преобразует одну переменную с n наблюдениями и d различными значениями в d двоичных переменных с n наблюдениями каждая. Каждое наблюдение указывает на наличие (1) или отсутствие (0) двоичной переменной dth. Поскольку количество категориальных признаков очень велико, традиционным алгоритмам машинного обучения сложно включить его One-hot кодированием, поскольку пространство признаков разбивается на тысячи, что становится сопоставимым с общим числом выборок в наборе данных.

2. Числовые особенности:

Числовые названия элементов содержат информацию о станциях, производственной линии и комбинации номеров испытаний. Значение для этого признака – соответствующее измерение. Например, признак с именем L3 S50 F4243 для компонента указывает, что деталь прошла производственную линию 3, станцию 50, и значение признака соответствует номеру испытания 4243. Таким образом, каждый продукт, выходящий из производственной линии, может быть отделен в соответствии с производственным потоком. Чтобы избежать путаницы, нужно ссылаться на эти значения признаков как на измерения. По наблюдению, существует 51 станция, распределенная между 4 производственными линиями.

Попытками были найти связь между различными признаками, станциями, на которых они записаны, производственной линией, на которой работает станция, и переменной отклика, соответствующей этим наблюдениям, чтобы понять, какие признаки, станции и линии играют более значительную роль, чем другие. На рисунке 6, видна частота различных функций, соответствующих станциям. Некоторые станции, а именно станции: 24, 25, 29, 30, имеют очень большое количество соответствующих им признаков, в то время, как они являются станциями, которые имеют очень небольшое количество соответствующих им признаков. Эти результаты указывают на некоторую относительную важность в определении станций, которые, возможно, сыграют, более важную роль в определении переменной отклика.

Подсчитав общее количество ненулевых измерений в каждой станции (рисунок 10), видно, что у станций 24 и 25 наибольшее количество измерений (> 200), у станции 32 есть только одно измерение, а у остальных станций есть около 20 измерений.

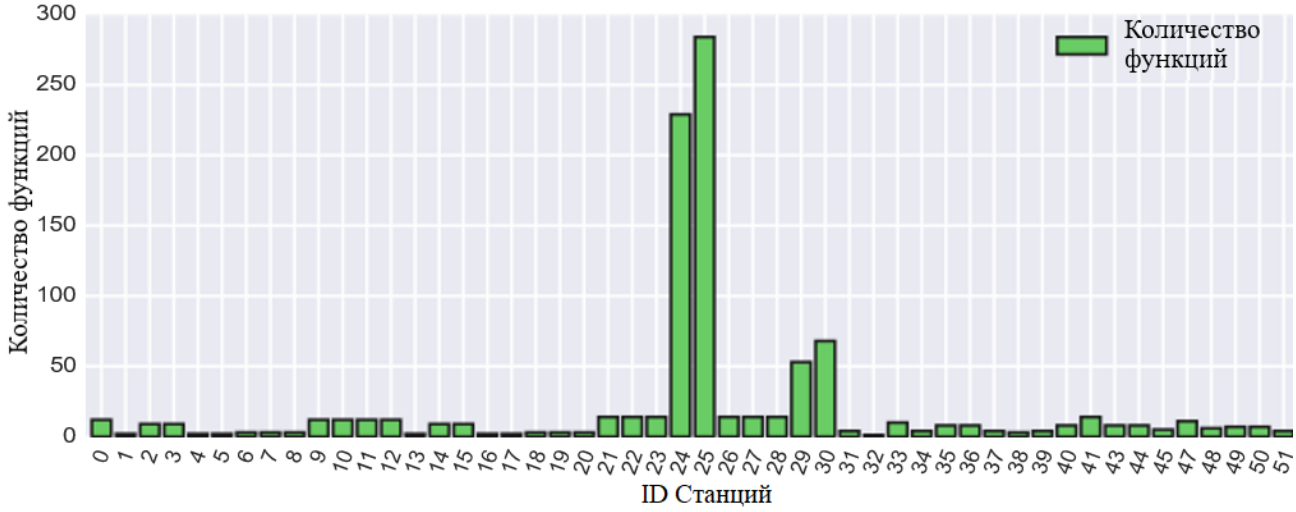

Рисунок 10 – Количество ненулевых измерений на каждой станции

Чтобы понять, как части движутся через станции, подсчет количества частей на станцию показан на рисунке 11. Видно, что каждая станция имеет разное количество проходящих через нее частей, что может означать существование различных классов продуктов, каждая из которых проходит определенный производственный путь. Кроме того, станция 32 имеет очень мало деталей, проходящих

через нее, что означает, что она не обрабатывает много деталей. Это указывает на то, что станция 32 является своего рода станцией повторной обработки или постобработки.

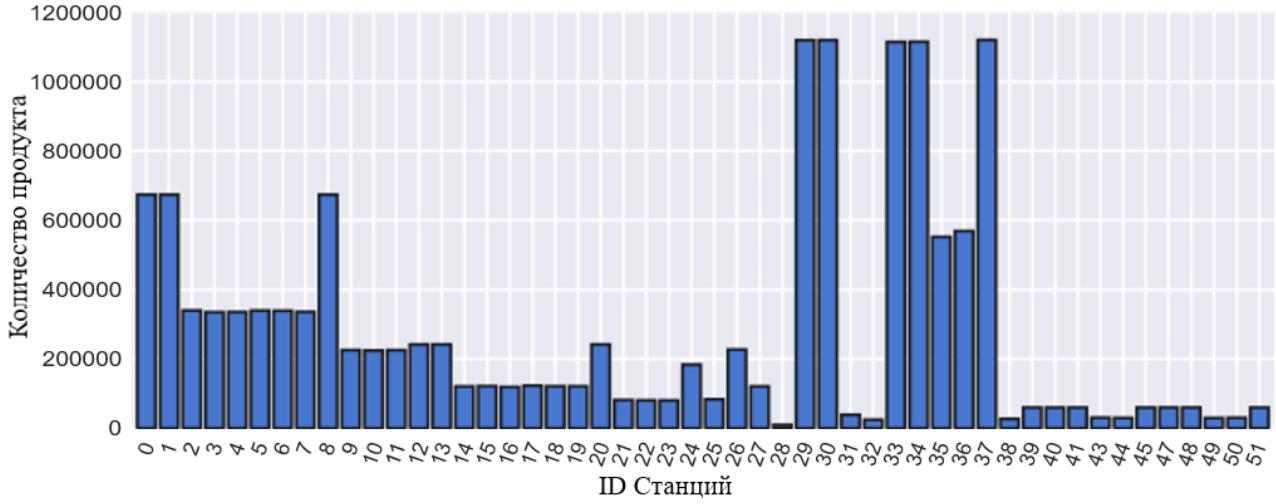

Рисунок 11 – Количество частей продукта, проходящих через каждую станцию

Чтобы выяснить, соотносятся ли определенные производственные линии или станции с более высокими показателями ошибок, следует рассчитать долю дефектных деталей на каждой станции и производственной линии. На рисунке 12 показана процентная ошибка между станциями, и было обнаружено, что станция 32 имеет самую высокую частоту ошибок. Однако станция 32 не обрабатывает много продуктов, поэтому ее влияние на выход продукции минимально. Всего через станцию 32 проходит 24 543 выборки с частотой ошибок 4,7% по сравнению со средней частотой ошибок около 0,6% на других станциях. Он имеет только одну особенность, L3 S32 F3850, что может означать, что станция 32 является частью производственной линии 3, и после повторной обработки все детали проходят тот же тест, обозначенный F3850. Станция 31 связана с самой низкой частотой отказов на уровне 0,27%.

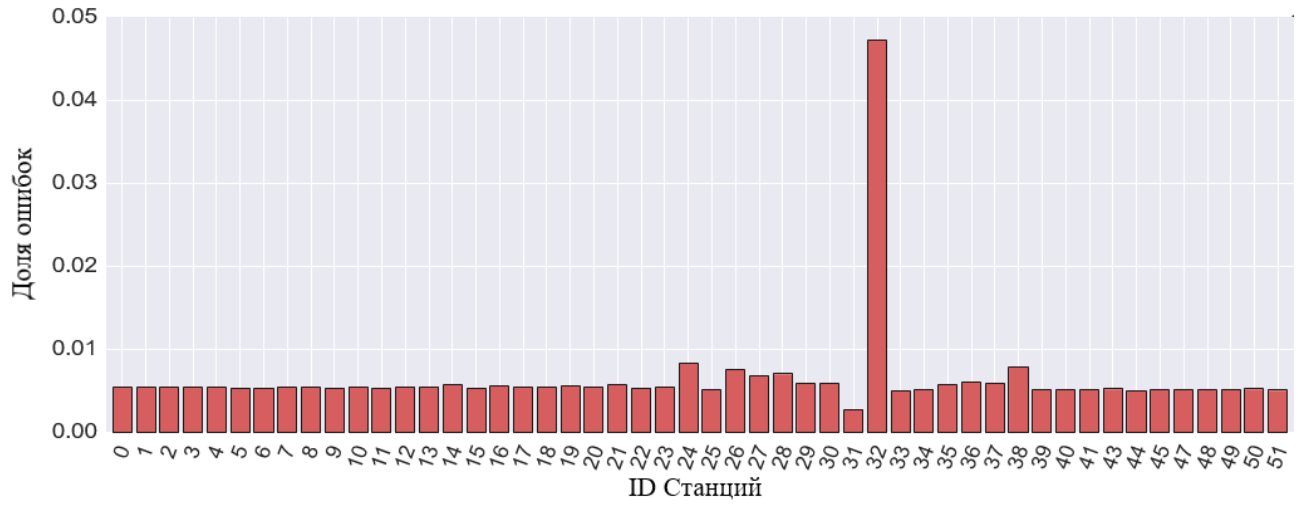

Рисунок 12 – Доля бракованной продукции на каждой станции

На рисунке 13 показаны коэффициенты ошибок по сравнению с различными производственными линиями, и ни на одной линии не было обнаружено необычного уровня ошибок.

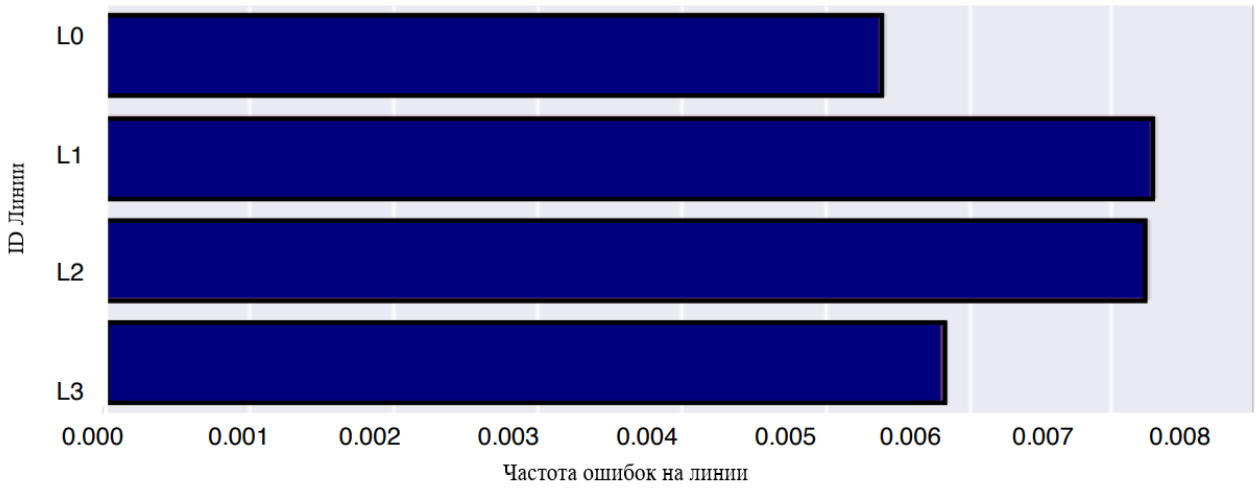

Рисунок 13 – Дробная ошибка в каждой производственной линии

Чтобы исследовать, как детали перемещаются по производственной линии, образцы были агрегированы по линии и номеру станции, и показан производственный поток, как видно на рисунке 14. Всего было найдено 7148 уникальных путей.

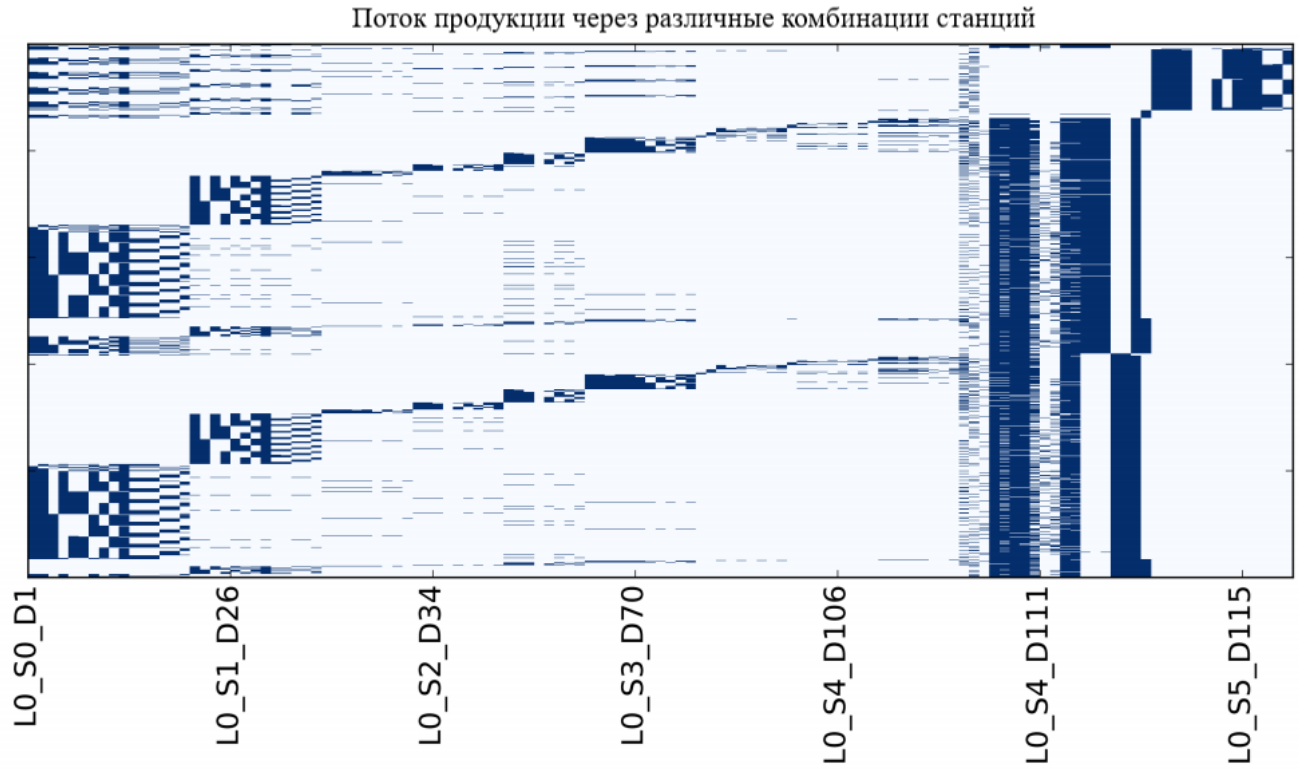

Рисунок 14 – Количество выполненных ненулевых измерений в сравнении с комбинацией номер-дата производственной линии-станции

Однако это означает, что в наборе данных есть несколько категорий продуктов, и для достижения наилучшего прогноза модели должны соответствовать каждой из этих категорий продуктов. Используя данные пути потока, эти части могут быть сгруппированы в семейства продуктов путем группировки аналогичных частотных путей частей, однако, это выходит за рамки данной работы.

Подобные продукты проводят аналогичное время в производственной линии. Используя эту интуицию, был разработан признак «Разница во времени», показывающий общее время, затраченное на каждую деталь в производственной линии. Для простоты нужно продолжить подгонку общей модели ко всем деталям с вышеупомянутым предостережением.

3. Особенности даты:

Названия объектов даты помечены по производственной линии, идентификатору станции и идентификатору даты. Например, L3 S50 D4242 будет означать, что продукт прошел производственную линию 3, станцию 50, и значение призна-

ка соответствует идентификатору даты 4242. Всего имеется 1157 объектов даты с большим количеством пропущенных значений. Одинаковые станции часто имеют одинаковые значения даты. На рисунке 15 представлен график зависимости количества записей от значения признаков даты. В данных можно наблюдать четкую периодическую закономерность, L2 со значениями признаков даты, лежащими между 0-1718, с гранулярностью 0,01.

Чтобы понять периоды времени, соответствующие этим числам, автокорреляция между признаками вычисляется как признак временного интервала между ними, рисунок 16. Наблюдается, что самые большие пики лежат при значении даты 16,75 тиков, и между ними существует около 7 локальных максимумов, которые должны соответствовать дням недели. Таким образом, 1 неделя составляет 16,75 единиц значения даты, и данные записываются с точностью до 6 минут. Поскольку набор данных соответствует измерениям, выполненным за 1718,48 единиц времени, то есть 102,6 недели, это объясняет изменчивость набора данных, поскольку заводские условия могут изменяться со временем.

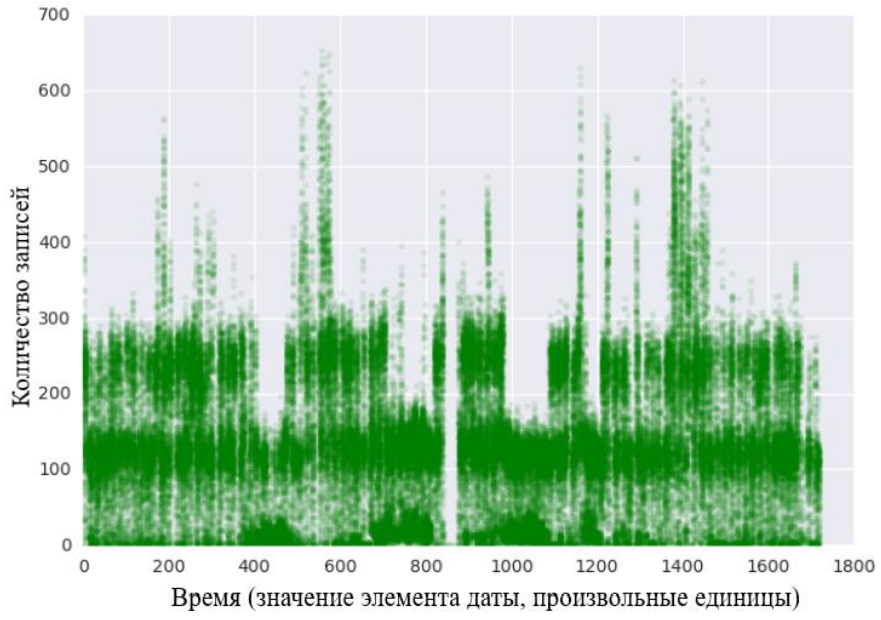

Рисунок 15 – Количество записей, сделанных по значению даты. Периодичность может наблюдаться. Гранулярность составляет 0,01, что соответствует 6 минутам в реальном времени

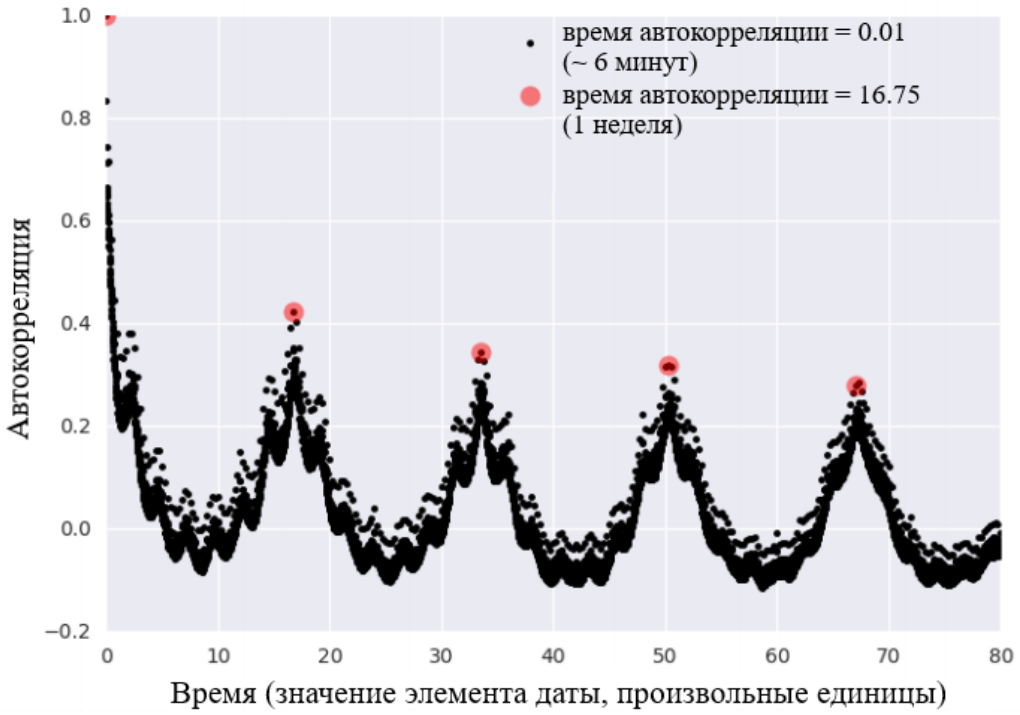

Рисунок 16 – Автокорреляция для числа наблюдений, записанных в день, как признаки временного интервала между ними

Выявлена периодичность: более высокие максимумы (каждые 16,75 тиков) соответствуют неделям, а меньшие локальные максимумы (каждые 2,39 тиков) соответствуют одному дню.

3.3 Исследование эффективности методов машинного обучения

3.3.1 Исследование эффективности метода k Ближайших Соседей

Очевидно, что лучшим K является тот, который соответствует наименьшей частоте ошибок теста, поэтому предположительно, нужно провести повторные измерения ошибки теста для различных значений K. Случайно используем тестовый набор в качестве тренировочного набора! Это означает, что происходит недооценка истинной частоты ошибок, поскольку модель была вынуждена соответствовать тестовому набору наилучшим образом. Модель тогда не в состоянии обобщить более новые наблюдения, процесс, известный как переоснащение. Следовательно, касание тестового набора исключено и должно быть сделано только в самом конце конвейера.

Не стоит забывать, что использование тестового набора для настройки гиперпараметра может привести к переоснащению.

Альтернативный и более разумный подход включает оценку частоты ошибок теста, выделяя подмножество обучающего набора из процесса подбора. Это подмножество, называемое набором проверки, может использоваться для выбора соответствующего уровня гибкости алгоритма. Существуют разные подходы к валидации, которые используются на практике, и будет исследован один из наиболее популярных, называемый k-кратной перекрестной валидацией [12].

Как видно на рисунке 17, перекрестная проверка в k-кратном порядке (k совершенно не связана с K) включает случайное деление обучающего набора на k групп или складок приблизительно одинакового размера. Первый сгиб рассматривается как проверочный набор, а метод подходит для оставшихся k-1 сгибов. Коэффициент ошибочной классификации затем рассчитывается на основе наблюдений в фиксированной таблице. Эта процедура повторяется k раз; каждый раз отдельная группа наблюдений рассматривается как набор проверки. Этот процесс приводит к k оценкам ошибки теста, которые затем усредняются.

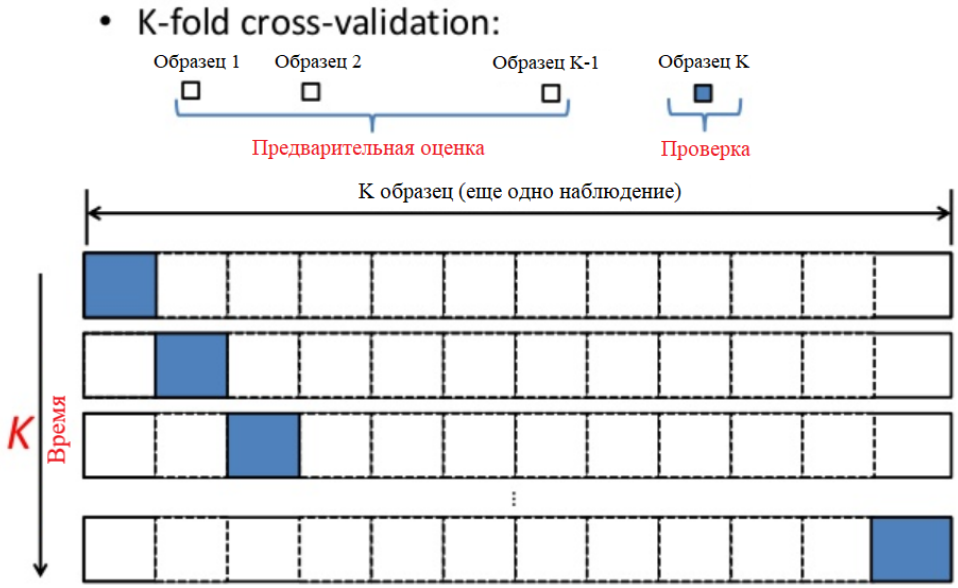

Рисунок 17 – Как работает кросс-валидация

Перекрестная проверка может использоваться для оценки ошибки теста, связанной с методом обучения, для оценки его эффективности или для выбора соответствующего уровня гибкости.

Если это немного ошеломляет, не стоит беспокоиться об этом. Для того, чтобы было более понятно, следует выполнить 3-кратную перекрестную проверку набора данных, используя сгенерированный список нечетных K в диапазоне от 1 до 50.

Пригодится scikit-learn с помощью метода cross_val_score (). Нужно указать, выполнить 3 крат с параметром cv = 3 и что метрика оценки должна быть точной, поскольку находится в классификации.

3-кратная перекрестная проверка говорит, что K = 7 приводит к самой низкой ошибке проверки. Результаты показаны на рисунках 16-17 в сравнении с другими методами, описанными ниже.

3.3.2 Исследование эффективности метода Случайных деревьев

Параметры настройки:

Параметры в случайных деревьях должны либо увеличить предсказательную силу модели, либо упростить обучение модели. Ниже приведены параметры, о которых поговорим более подробно. В первую очередь есть 3 функции, которые можно настроить, чтобы улучшить предсказательную силу модели:

1) max_features:

Это максимальное количество функций, которые разрешено использовать в отдельном дереве. В Python доступно несколько опций для назначения максимальных возможностей. Вот несколько из них:

– Auto/None: просто примет все функции, которые имеют смысл в каждом дереве. Здесь просто не накладываем никаких ограничений на отдельное дерево.

– sqrt: эта опция получит квадратный корень от общего количества объектов в отдельном прогоне. Например, если общее количество переменных

равно 100, можно взять только 10 из них в отдельном дереве. «Log2» – еще один аналогичный тип опции для max_features.

– 0.2: Эта опция позволяет случайному лесу принимать 20% переменных в отдельном прогоне. Можно присваивать и оценивать в формате «0.x», где нужно, чтобы рассматривалось x% функций.

Как max_features влияет на производительность и скорость:

Увеличение max_features, как правило, повышает производительность модели, поскольку на каждом узле теперь у нас есть большее количество вариантов, которые необходимо рассмотреть. Однако это не обязательно так, поскольку это уменьшает разнообразие отдельных деревьев, которые являются USP случайного леса. Но, конечно, уменьшается скорость алгоритма, увеличивая max_features. Следовательно, нужно найти правильный баланс и выбрать оптимальный max_features.

2) n_estimators:

Это количество деревьев, которые нужно построить перед тем, как принять максимальное голосование или усреднить прогнозы. Большее количество деревьев повышает производительность, но замедляет код. Следует выбрать настолько высокое значение, сколько сможет обработать процессор, потому что это делает прогнозы сильнее и стабильнее.

3) min_sample_leaf:

Если уже построено дерево решений, можно оценить важность минимального размера листа выборки. Лист является конечным узлом дерева решений. Меньший лист делает модель более склонной к улавливанию шума в данных train. Как правило, предпочтительным минимальным размером листа, будет более 50. Однако нужно попробовать несколько размеров листа, чтобы найти наиболее оптимальный для определенного случая использования.

2. Особенности, которые облегчат обучение модели:

Есть несколько атрибутов, которые напрямую влияют на скорость обучения модели. Ниже приведены основные параметры, которые можно настроить для скорости модели:

1) n_jobs:

Этот параметр сообщает движку, сколько процессоров ему разрешено использовать. Значение «-1» означает, что ограничений нет, а значение «1» означает, что он может использовать только один процессор. Вот простой эксперимент, который можно выполнить с Python для проверки этой метрики – «% timeit» - это функция awsum, которая запускает функцию несколько раз и обеспечивает самое быстрое время выполнения цикла. Это очень удобно при масштабировании определенной функции от прототипа до окончательного набора данных.

2) random_state:

Этот параметр позволяет легко воспроизвести решение. Определенное значение random_state всегда будет давать одинаковые результаты, если дано с теми же параметрами и данными обучения. Ансамбль с несколькими моделями различных случайных состояний, и все оптимальные параметры иногда работают лучше, чем отдельные случайные состояния.

3) oob_score:

Это метод перекрестной проверки случайных лесов. Это очень похоже на использование одного метода проверки, однако это намного быстрее. Этот метод просто помечает каждое наблюдение, используемое в разных деревьях. И затем он определяет максимальный балл за каждое наблюдение, основываясь только на деревьях, которые не использовали это конкретное наблюдение для обучения.

Вот один пример использования всех этих параметров в одной функции:

model = RandomForestRegressor(n_estimator = 100, oob_score = TRUE, n_jobs = -1,random_state =50, max_features = "auto", min_samples_leaf = 50)

model.fit(X,y)

3.3.3 Исследование эффективности метода Классификатор эктра-деревьев

В этом классе реализована метаоценка, которая подходит к ряду рандомизированных деревьев решений (также называемых дополнительными деревьями) в различных подвыборках набора данных и использует усреднение для повышения точности прогнозирования и контроля соответствия.

1) max_depth: целое число или нет, необязательно (по умолчанию = нет). Максимальная глубина дерева. Если None, то узлы расширяются до тех пор, пока все листья не станут чистыми или пока все листья не будут содержать меньше чем min_samples_split samples.

2) bootstrap: логическое, необязательное (по умолчанию = False). Используются ли образцы начальной загрузки при построении деревьев. Если False, весь набор данных используется для построения каждого дерева.

3) n_jobs: int или None, необязательный (по умолчанию = None). Количество заданий, выполняемых параллельно для обоих fit и predict. None означает 1, если не в joblib.parallel_backend контексте. -1означает использование всех процессоров.

3.3.4 Исследование эффективности метода Градиентного бустинга

Опираясь на идеи, полученные в первоначальных исследованиях, можно истолковать, что существует меньшее подмножество признаков, которые имеют большее значение, чем остальные. Поскольку набор возможных признаков очень велик, следует попробовать разные методы уменьшения размерности. Разработка признаков является важным шагом в рабочем процессе из-за огромного объема данных и интересных выводов об их относительной важности при решении проблемы с продуктом. После преобразования всех категориальных признаков в числовой тип размерность пространства признаков уменьшается до 968 исходных

числовых признаков, 1 инженерного объекта, 1 производного числового объекта и 1 156 объектов даты, что в сумме составляет 2 126 объектов. Все это делает возможным использование обычных алгоритмов машинного обучения.

Опираясь на выводы по этой конкретной проблеме, был использован Extreme Gradient Boosting (XGBoost), чтобы извлечь 15 лучших полезных функций, которые доминировали в результатах любой конкретной модели. XGBoost – это продвинутый алгоритм, основанный на повышении градиента. Его использование обусловлено тем, что XGBoost (более быстрое внедрение традиционных деревьев с градиентным расширением) потому что это нелинейный алгоритм, который очень хорошо работает с числовыми характеристиками и требует меньшего количества технических характеристик и настройки гиперпараметров, что упрощает реализацию в данном случае.

Gradient Boosted Trees, сильный классификационный алгоритм, по сути является ансамблем множества слабых деревьев. Деревья небольшой глубины создаются на выборке строк и объектов на каждом шаге, и эти деревья используются для составления прогноза. Используя текущий прогноз, вычисляется градиент признака стоимости потерь в логарифме относительно цели, а затем создается следующий раунд деревьев для изучения градиента. Таким образом, он пытается минимизировать ошибку, полученную до n-го шага. Этот метод склонен к переоснащению, поскольку он постоянно включает подгонку модели по градиенту. Чтобы предотвратить это, следует оптимизировать количество деревьев до тех пор, пока ошибка из выборки не начнет снова увеличиваться. Некоторые из гиперпараметров, которые будут присутствовать в обсуждении для XGBoost:

– скорость обучения: скорость обучения сокращает вклад каждого дерева на скорость обучения.

– n оценщиков: количество ступеней повышения, которые нужно выполнить. Этот параметр должен контролироваться на нейтральном тестовом наборе, чтобы предотвратить переоснащение.

– максимальная глубина: максимальная глубина отдельных оценок регрессии. Максимальная глубина ограничивает количество узлов в дереве. Наилучшее значение зависит от взаимодействия входных переменных.

– минимальный вес экземпляра: необходимая минимальная сумма веса экземпляра. Если в результате шага разбиения дерева получится листовой узел с суммой веса экземпляра, меньшей, чем минимальный дочерний вес, то процесс построения прекратит дальнейшее разбиение. В режиме линейной регрессии это просто соответствует минимальному количеству экземпляров, которое должно быть в каждом узле. Чем больше это число, тем более консервативным будет алгоритм.

Второй этап обучения состоит в том, чтобы выбрать наиболее важные признаки и затем использовать алгоритм XGBoost (XGB) для прогнозирования вероятностей класса, к которому применяется порог для получения окончательных прогнозов класса.

3.3.5 Исследование эффективности метода Легкого градиентного бустинга

Параметры настройки:

LightGBM использует алгоритм листового роста деревьев, в то время как многие другие популярные инструменты используют рост деревьев по глубине. По сравнению с глубинным ростом листовой алгоритм может сходиться гораздо быстрее. Тем не менее, листовой рост может быть чрезмерным, если не используется с соответствующими параметрами.

Чтобы получить хорошие результаты, используя листовое дерево, нужно использовать некоторые важные параметры:

1) num_leaves. Это основной параметр для управления сложностью модели дерева. Теоретически, можно установить num_leaves = 2 ^ (max_depth), чтобы получить то же количество листьев, что и дерево по глубине. Однако это простое преобразование не очень хорошо на практике. Причина в том, что листовое дерево обычно намного глубже, чем глубинное, для фиксированного числа листьев. Не-

ограниченная глубина может вызвать переоснащение. Таким образом, пытаясь настроить num_leaves, нужно позволить ему быть меньше 2 ^ (max_depth). Например, когда max_depth = 7, дерево по глубине может получить хорошую точность, но установка num_leaves в 127 может привести к перенастройке, а установка в 70 или 80 может получить лучшую точность, чем по глубине.

2) min_data_in_leaf. Это очень важный параметр для предотвращения чрезмерной подгонки в дереве листьев. Его оптимальное значение зависит от количества обучающих выборок и num_leaves. Установка его в большое значение может избежать слишком глубокого роста дерева, но может привести к недостаточной подгонке. На практике для больших наборов данных достаточно задать сотни или тысячи.

3) Максимальная глубина. Также можно использовать max_depth для явного ограничения глубины дерева.

Для более быстрой скорости:

- упаковка, установив bagging_fractionибagging_freq;
- функция подвыборки, установив feature_fraction;
- маленький max_bin;
- save_binary для ускорения загрузки данных в будущем обучении;
- параллельное обучение.

Для лучшей точности:

- большой max_bin (может быть медленнее);
- маленький learning_rate с большим num_iterations;
- большой num_leaves (может привести к переоснащению);
- большие тренировочные данные.

Переоснащение:

- маленький max_bin;
- маленький num_leaves;
- упаковку по набору bagging_fractionибagging_freq;
- функция подвыборки по набору feature_fraction;

– большие тренировочные данные;
– max_depth стараться не растить глубокое дерево.

При применении LightGBM вместо XGBoost наблюдалось лишь незначительное увеличение точности и улучшенных показателей, но во время выполнения процедуры обучения была существенная разница. LightGBM почти в 7 раз быстрее, чем XGBoost, и это гораздо лучший подход при работе с большими наборами данных. Это оказывается огромным преимуществом, когда следует работать с большими наборами данных с ограниченным временем. Этот метод подразумевает его использование в будущем, как интеграция прогнозирования сбоев во время производства.

3.4 Обсуждение

Одним из преимуществ XGBoost является то, что он довольно хорошо обрабатывает NaN. Другой способ состоял бы в том, чтобы заменить NaN медианой или модой столбца, или попробовать подход ближайших соседей.

Большинство отсутствующих значений в этом наборе данных структурно отсутствуют – в этом случае вменение может быть не лучшим подходом. С большим количеством недостающих данных в этой проблеме, вероятно, стоит поэкспериментировать со способами ее решения.

Ограничение по ОЗУ, и первые N строк, вероятно, будут плохой идеей для данных даты, так как они упорядочены. Порог для управляемого количества признаков продемонстрирован на рисунке 18.

```
[  14   23   41   50  385 1019 1029 1034 1042 1056 1156 1161 1166 1171 1172
 1183 1203 1221 1294 1327 1350 1363 1403 1404 1482 1501 1507 1512 1535 1549
 1550 1843 1846 1849 1858 1879 1885 1887 1888 1891 1911 1940 1948 1951 1959
 1974 1975 1982 1985 1988 1993 1994 1995 1999 2006 2007 2010 2028 2040 2046
 2075 2093]
```

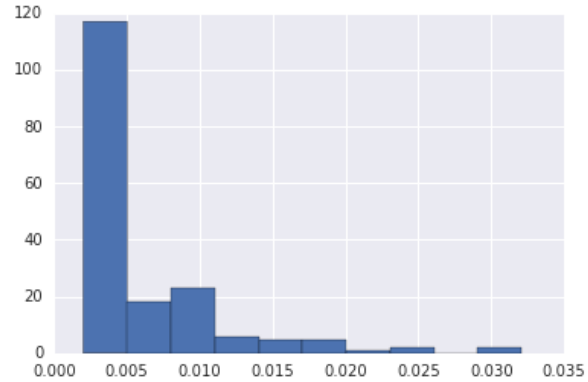

Рисунок 18 – Порог для управляемого количества признаков

Приведенный выше сюжет, отражает относительную важность различных признаков. 15 основных признаков, которые были обнаружены на основе анализа, следующие: «L0_S9_F170», «L1_S24_F1346», «L1_S24_F1366», «L1_S24_F1406», «L1_S24_F1578», «L2_S26_F3036», «L3_S29_F3321», «L3_S29_F3348», «L3_S29_F3427», «L3_S30_F3514», «L3_S31_F3846», «L3_S33_F3850», «L3_S33_F3857», «L3_S33_F3863», «L3_S37_F3950».

В 15 лучших признаках есть 4 признака, соответствующий станции 24, 3 признака для станции 33. Также стоит отметить, что признаки, соответствующие линии 1 и линии 3, являются наиболее важными. Эти наблюдения в значительной степени соответствуют более раннему исследованию данных.

Меры важности признаков основаны на числе раз, когда переменная выбирается для расщепления деревьев компонентов в XGBoost, взвешивается по квадрату улучшения модели в результате каждого расщепления и усредняется по всем деревьям. Другими словами, важность признаков в моделях древовидных ансамблей определяется тем, как часто элемент появляется в деревьях моделей.

Из оставшихся 2 126 компонентов выбор параметров выполняется с использованием градиентного усиления на случайном подмножестве 100 000 выборок обучающих данных, чтобы предотвратить смещение выборки. Параметры, использу-

емые для этой предварительной подготовки: скорость обучения = 0,1, максимальная глубина = 3, минимальный вес = 1, n оценщиков = 100, nthread = -1.

Это помогает в выборе топ 200/2 126 параметров, а затем окончательная модель обучается с использованием всего 1 миллиона данных обучения строк с этими 200 параметрами. Этот двухэтапный процесс помогает сократить время выполнения и объем памяти, необходимый для выбора параметров, а также для окончательного обучения модели.

Используя эти 200 лучших параметров, весь обучающий набор данных (1 миллион образцов) используется для обучения новой модели XGBoost [13], [14]. Все данные обучения делятся случайным образом на три подмножества данных по 33% каждый. Три отдельные модели XGB с одинаковыми гиперпараметрами обучаются на 67% данных и оцениваются на оставшихся 33% данных, чтобы уменьшить потребление памяти и увеличить скорость обучения. Прогнозы для каждого сгиба суммируются и затем используются для получения объективной оценки ошибки обучения для всего набора тренировок. Этот метод тройной перекрестной проверки используется для точной настройки гиперпараметров модели XGB при оптимизации области под кривой (AUC). Конечные индивидуальные площади под кривыми для трех моделей: 0,719, 0,718, 0,718. После суммирования всех их прогнозов общий AUC для всех данных тренировки становится 0,718 0,001.

Параметры, используемые для окончательного обучения XGB: скорость обучения = 0,01, максимальная глубина = 5, минимальный вес = 5, n оценщиков = 100, nthread = -1. С наилучшими настроенными гиперпараметрами в руках весь обучающий набор данных используется для обучения окончательной модели. Аналогичный подход 3-кратной перекрестной проверки был опробован с другими алгоритмами классификации, из которых модель XGB показала наилучшие результаты. В таблице 3 продемонстрировано осреднение результатов по итогам нескольких проверок.

Таблица 3 – Сравнение различных алгоритмов в обучающем наборе данных Bosch с 3-кратной перекрестной проверкой

Используемая модель классификации	3-fold Cross Validation Training AUC
k Nearest Neighbor	0.614 ± 0.004
Random Forest Classifier	0.685 ± 0.003
Extra Trees Classifier	0.709 ± 0.003
Light Gradient Boosting Machine (LightGBM)	0.716 ± 0.001
eXtreme Gradient Boosting (XGB)	0.718 ± 0.001

MCC рассчитывается для порога оценки, который делит признаки на два класса. Сначала баллы прогнозирования получают для обучающих данных, используя метод трехкратной перекрестной проверки, как описано выше. Затем, чтобы выбрать наилучшее значение отсечения для коэффициента Мэтьюса, нужно перебрать все возможные значения отсечения, рассчитываемые и наносимые на график коэффициента Мэтьюса. Из рисунка 19, можно увидеть, что для обучающих данных наилучший достигнутый коэффициент Мэтьюса, был 0,227 при пороге вероятности 0,11. Это значение MCC соответствует AUC 0,718.

Это означает, что признаки, имеющие оценку прогнозирования выше 0,11, могут быть помечены для последующей обработки, поскольку они, скорее всего, не будут работать. В обучающем наборе данных только 3 235 выборок подпадают под эту категорию и должны быть переоценены, в отличие от 2 признаков, помеченных AUC. Это приводит к экономии времени и ресурсов, а также к увеличению прибыли из-за снижения качества продукции, увеличения утилизации и повышения производительности.

Рисунок 19 – Определение вероятностного порога для максимизации коэффициента корреляции Мэтьюса (обучающий набор данных)

Наиболее классическая "корреляционная" мера между номинальной и интервальной ("числовой") переменной, равна Eta, также называемой коэффициентом корреляции. Eta можно рассматривать как симметричную меру ассоциации, такую как корреляция, потому что Eta из ANOVA (с номинальной независимой, числовой как зависимая) равна трассировке Pillai многомерной регрессии (с числовым как независимым, набором фиктивных переменных, соответствующих номинальной как зависимой). Результаты параметра ETA показаны на рисунке 20.

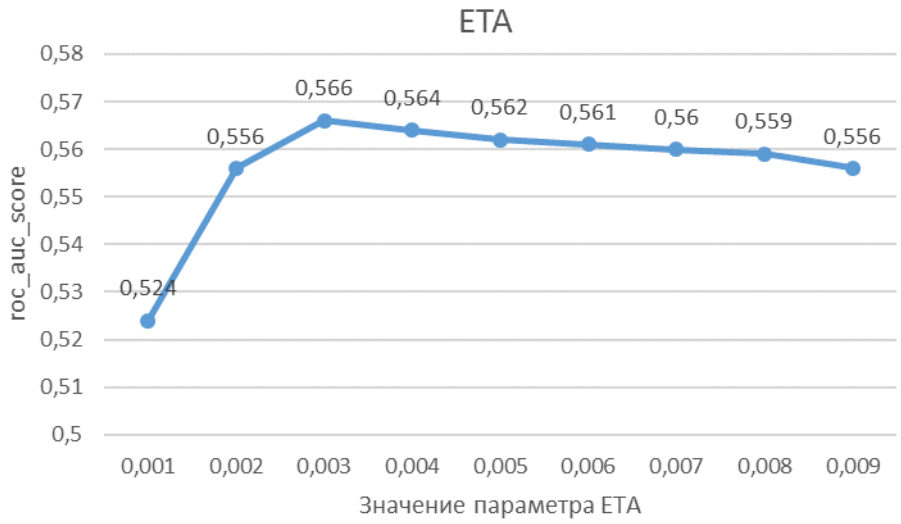

Рисунок 20 – Результат accuracy при заданных параметрах ETA

Исходя из графика понятно, что лучший результат достигается при значении параметра «ETA» равного 0,003.

Таким же образом вычисляем лучшее значение параметра «MaxDepth». Результаты представлены на рисунке 21.

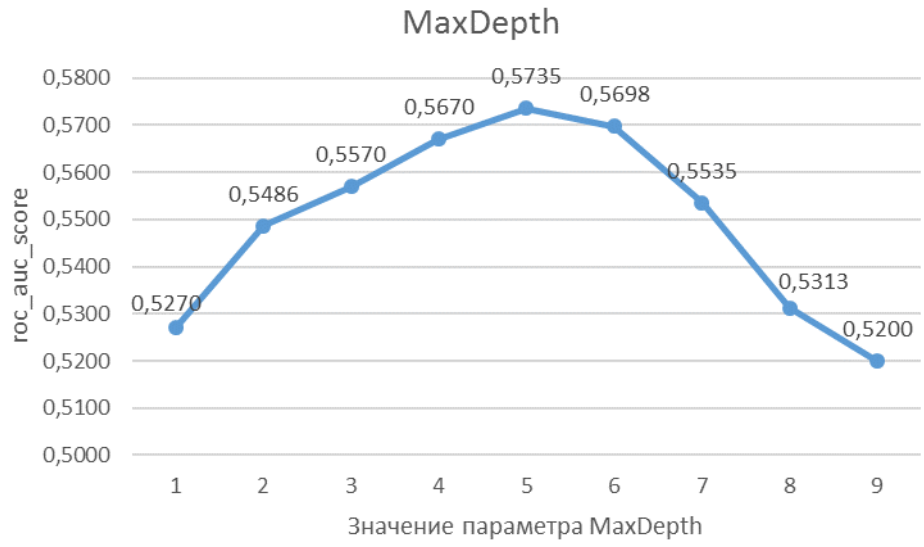

Рисунок 21 – Результат accuracy при заданных значениях параметра MaxDepth

Остальные параметры, не отраженные в графиках, либо были установлены в значении по умолчанию в Scikit-learn, или же их изменение не оказывало влияния на результативность модели.

Вероятности прогноза для всего обучения сортируются по убыванию, а затем делятся на децили. Для модели случайного выбора, чтобы покрыть половину неисправных продуктов, необходимо проверить 50% всей совокупности продуктов. Однако, с моделью, можно увидеть, что, если нацелиться на две верхние вероятности сбоев дециля из модели, то получится собрать около 50% целевой совокупности сбоев. Это намного лучше, чем случайная проверка на наличие дефектов.

Была оптимизирована модель для лучшего AUC, которая является мерой ранжирования того, как модель способна различать два класса меток. Тем не менее, даже два десятка параметров могут означать проверку более 200 000 деталей на наличие дефектов, что является финансовым и временным недостатком. Следовательно, лучшим критерием оценки является именно ROC/AUC. Однако ниже приведено сравнение результатов методов, как по ROC/AUC, так и по MCC.

Этот набор данных является редким, с большим количеством пропущенных значений, более миллиона выборок и редким событием с <1% положительных выборок. Это делает в вычислительном отношении дорогостоящим применение традиционных методов машинного обучения. Количество признаков очень велико, и их пространство еще больше взрывается после One-hot кодирования 2 140 категориальных признаков. Таким образом, набор данных очень склонен к переобучению из-за «проклятия размерности». Наличие большинства категориальных признаков способствует использованию модели онлайн-обучения, которая использована здесь в качестве метода сокращения возможностей. Был получен тренинг AUC 0,718 0,001, что является хорошей базовой моделью, но имеет еще возможности для улучшения.

Было обнаружено, что вдоль производственной линии и станций имеется 7 158 уникальных путей потока, что указывает на наличие нескольких категорий продуктов. Для достижения более точных прогнозов продукты должны быть сгруппированы по этим категориям, и для каждой категории должны быть созданы отдельные модели. Также по наблюдениям стало известно, что существует еженедельная периодичность количества наблюдений, записанных в день. Это понима-

ние в будущем дает возможность улучшить модель путем включения сложных признаков, основанных на дате, с учетом изменчивости, наблюдаемой из-за изменений в заводских условиях во времени. На рисунках 22, 23 показаны результаты полного набора данных. ROCAUC и MCC увеличиваются с увеличением размера обучающих данных. Методы градиентного бустинга и легкого градиентного бустинга продемонстрировали наилучший результат, чем методы простых классификаторов (k Ближайших соседей, случайные деревья и классификатор экстра-деревьев).

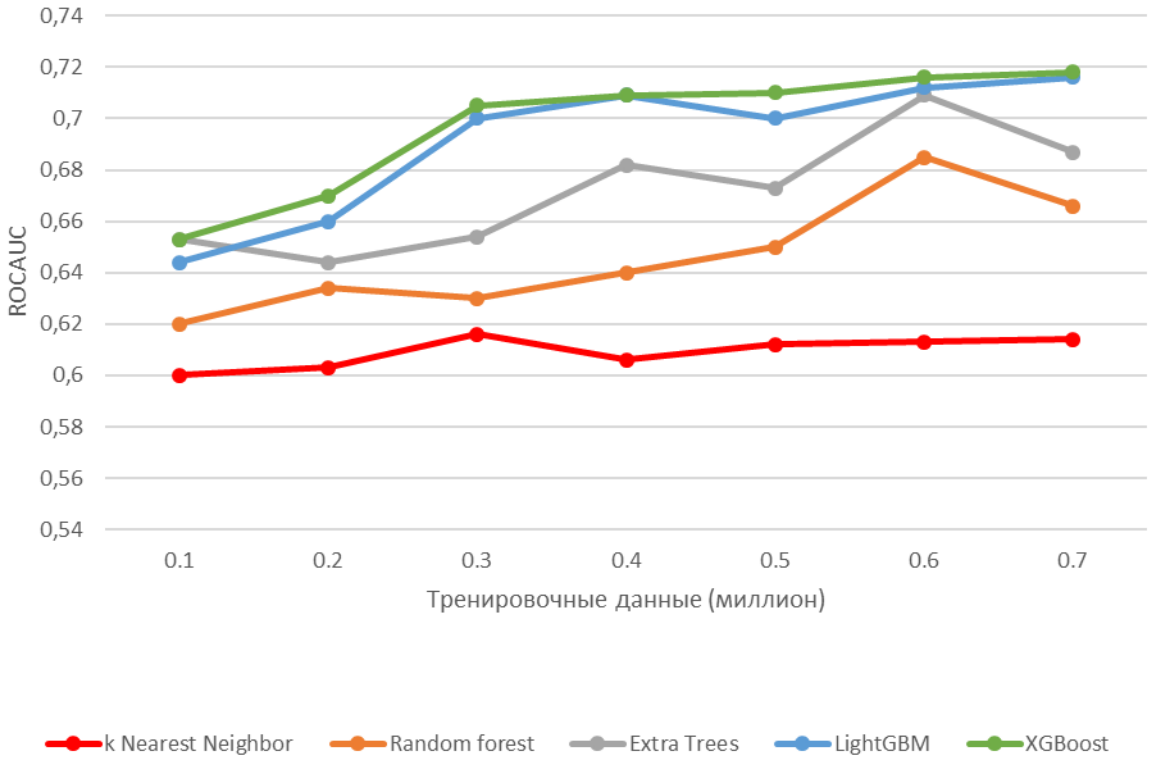

Рисунок 22 – Сравнение методов по метрике ROC/AUC

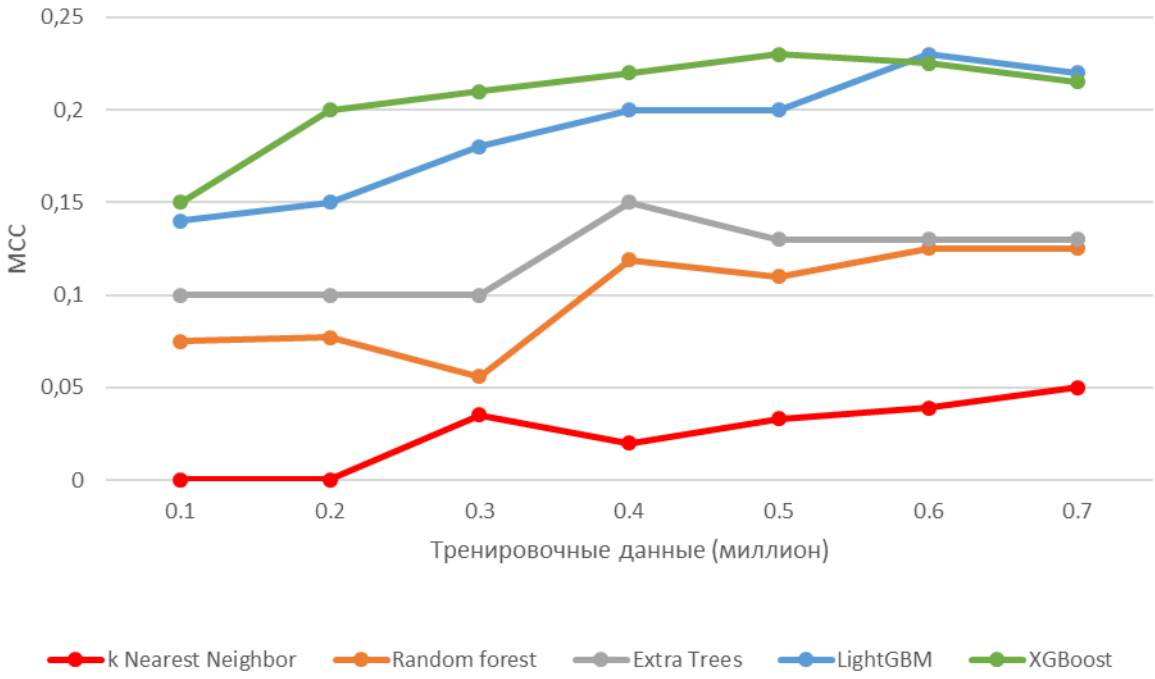

Рисунок 23 – Сравнение методов по метрике MCC

Выводы по главе 3

Была поставлена цель – предсказать, будет ли конкретная деталь (идентифицируемая уникальным идентификатором) проходить контроль качества или нет (представленный как «Ответ» = 1). Данные предоставлены компанией BOSCH.

Чтобы приступить к прогнозированию, для начала следует тщательно изучить и проанализировать набор данных, который очень большой и с очень большим количеством признаков. Поскольку никакой предварительной информации о признаках не было известно, была попытка найти соответствующие связи между признаками после изучения информации в наборе данных. Получив представление о наборе данных, были выбраны лучшие наборы признаков, которые объясняют набор данных лучше всего. Следующим шагом были исследованы параметры методов:

– k ближайших соседей;
– случайный лес;
– классификатор экстра-деревьев;
– градиентный бустинга;

– легкого градиентного бустинга.

В табличном виде продемонстрировано сравнение всех алгоритмов в обучающем наборе данных Bosch с 3-кратной перекрестной проверкой. На рисунках показаны результаты полного набора данных по ROCAUC и MCC. Методы градиентного бустинга и легкого градиентного бустинга продемонстрировали наилучший результат, чем методы простых классификаторов (k Ближайших соседей, случайные деревья и классификатор экстра-деревьев).

4. КОММЕРЦИАЛИЗАЦИЯ ПРОЕКТА

4.1 Дорожная карта коммерциализации проекта

Дорожная карта – это развернутый пошаговый план развития проекта, сформированный с учетом особенностей рынка и существующих технологий. Грамотно составленная «дорожная карта» помогает спрогнозировать пути развития проекта и выбрать наиболее эффективную стратегию. Таким образом, «дорожная карта» является мощным инструментом стратегического развития, планирования и принятия управленческих решений.

Составление «дорожной карты» — важный этап в создании инновационного продукта. Для формирования «дорожной карты» требуется провести тщательный анализ рынка, изучить технологии, оценить продукт, учесть особенности отрасли.

Дорожная карта – это наглядное представление пошагового сценария развития определенного объекта. Процесс формирования дорожной карты:

- коллективная ревизия имеющегося потенциала развития;
- обнаружение возможностей роста;
- обнаружение рисков;
- выявление потребности в ресурсном обеспечении.

Дорожное картирование связывает между собой видение, стратегию и план развития объекта и выстраивает основные шаги этого процесса во времени по принципу «прошлое – настоящее – будущее». Оно опирается на сбор экспертной информации о продукте, технологии, отрасли и т. д., позволяющей прогнозировать варианты их будущего состояния.

4.1.1 Планирование стратегии: основные цели и источники доходов проекта

Любое производство, начиная от выпуска памперсов и заканчивая строительством космических кораблей, нуждается в определенном наборе услуг. Таким образом предприятие осуществляет свою деятельность в самой динамичной сфере – сфере услуг.

Основная цель проекта заключается в создании веб-представительства компании, деятельность которого будет направлена на предоставление услуг по предотвращению и сокращению технологических сбоев производственной цепи на предприятии.

Наличие веб-представительства даст компании следующие преимущества и решение таких задач, как:

– создание веб-представительства для получения высоких доходов;
– организация получения стабильной прибыли;
– формирование и продвижение имиджа компании;
– увеличение спроса на предоставляемую услугу;
– улучшение системы связей с общественностью;
– обеспечение потребителей, партнеров, рекламных агентов полной и актуальной информацией о товаре и фирме;
– обеспечение информационной поддержки потребителей посредством обратной связи;
– расширение каналов сбыта предприятия.

Однако основной сферой деятельности планируется разработка, реализация и сервис по предоставлению услуги, а также в дальнейшем разработка и доработка существующего программного решения на базе новых информационных технологий. Данная отрасль является достаточно молодой для российского рынка и поэтому большинство компаний испытывают нехватку в профессиональных, качественных услугах. Потребность в данной услуге весьма велика. Все это позволит предоставить необходимые решения для плодотворного функционирования различного рода компаний.

Доходность предприятия подразумевает распространение (продажу) предоставляемых услуг, а также размещение интернет рекламы, схожей тематики (услуги по оптимизации производственной цепи, дополнительное ПО и тд.), на разработанном интернет ресурсе. Планируется что реклама будет осуществляться

по модели CPM – цена, устанавливаемая за тысячу показов. Доходность предприятия отображена в таблице 4.

Таблица 4 – Доходность веб-представительства компании

Период	Вид услуги	Объем реализации в месяц, шт.	Стоимость, руб.	Прибыль от реализации, руб.
1-3 месяц	Предоставление услуги	5-15 шт.	15 000	75 000 – 225 000
	Реклама	1 шт.	350 CPM (показов – 2 000)	700
4-6 месяц	Предоставление услуги	20-35 шт.	25 000	500 000 – 875 000
	Реклама	1-3 шт.	700 CPM (показов – 10 000)	7 000

4.1.2 Оценка потенциальных возможностей Интернета для бизнеса

В целевой аудитории веб-сайта можно выделить следующие группы:

1) Предоставляемая услуга рассчитана на уменьшение количества сбоев на конвейере – в основе направлено на решение проблем конвейера предприятия BOSCH;

2) Целевая аудитория: являются предприятия с похожими сбоями на конвейерах;

3) География: направлено на зарубежную компанию BOSCH, в дальнейшем привлечение потребителей в Российской Федерации и иных странах, и государствах;

4) Средний доход ЦА: любое крупное предприятие, которое в полной мере может позволить себе приобретение данной услуги;

5) Сфера деятельности: предоставление крупным корпорация (фирмам) услуги по предотвращению и сокращению технологических сбоев, которые имеют средства на ее приобретение;

6) Интересы: хотят частично или в полной мере устранить возникающие сбои на конвейерах.

В настоящее время количество Интернет-ресурсов, реализующих те или иные услуги или программные средства достаточно велико. Однако Web-сайтов, предлагающих целенаправленную услугу по решению данной проблемы не так много, особенно те, кто предлагает эксклюзивные решения, чем и будет являться данная услуга. В связи со всем вышесказанным следует сделать вывод, что конкуренцию могут составлять только Программисты (Freelance) или же участники конкурсных разработок.

Потенциальными партнерами следует выделить организации, которые могут предоставить рекламу или размещение данной услуги на своем Web-ресурсе.

Для того чтобы определить внешние и внутренние факторы, влияющие на возможности работы в Интернете и формирование интернет-стратегии, следует провести SWOT-анализ, который продемонстрирован в таблице 5.

Таблица 5 – SWOT-анализ

Сильные стороны	Слабые стороны
Низкая себестоимость проекта, так как он уже узкоспециализирован; Использование инновационных технологий; Сопровождение проекта, ведение и поддержка на всех этапах реализации.	Слабый маркетинг; Узкая направленность продукта; Отдельная оплата доп. услуг (личный менеджер, обучение персонала, тех. поддержка и др.)

Окончание таблицы 5

Возможности	Угрозы
Новые технологии; Дополнительные услуги; Сотрудничество с другими (схожими) организациями; Увеличение рекламы;	Конкуренция в отдельно взятых конкурсах, программисты freelance; Зависимость предприятия; Лицензионный барьер, новые законодательные акты;

4.2 Создание сайта

Как придумать доменное имя для своего Интернет-ресурса:

Выбор домена для своего интернет-магазина аналогичен со схемой придумывания названия обычного магазина. Основными методами для этого являются:

- «мозговой» штурм;
- оформление заказа на нейминг одной или нескольких бирж фриланса.

Для успешной работы интернет-представительства компании, доменное имя должно соответствовать некоторым критериям:

- быть созвучным тематике бизнеса, либо продаваемых товаров;
- легкость написания, произнесения, и запоминания;
- быть свободным, т.е. незарегистрированным кем-нибудь другим.

В настоящее время регистраторы доменных имен предлагают на выбор более 740 различных доменных зон, из которых фактически для бизнеса подойдет не более десятка. Топ-4 самых популярных зон – это .ru, .com, .net, org. В таблице 6 представлены варианты доменных имен, выделены их достоинства и недостатки.

Таблица 6 – достоинства и недостатки доменных имен

Домен	Достоинства	Недостатки
Prediction.com/.ru	Незарегистрированный домен; Созвучно с тематикой; Частично отражает суть услуги и сайта; Просто запомнить; Популярная зона.	Частично понятна суть предлагаемой услуги и наполнения сайта;
Failure.com/.ru	Незарегистрированный домен; Созвучно с тематикой; Частично отражает суть услуги и сайта; Просто запомнить; Популярная зона.	Частично понятна суть предлагаемой услуги и наполнения сайта;
Failure-prediction.com/.ru	Незарегистрированный домен; Созвучно с тематикой; Понятна суть предлагаемой услуги и сайта в целом; Популярная зона.	Сложность в запоминании и написания ссылки сайта.
Poif.com/.ru	Незарегистрированный домен; Краткое запоминающееся название; Популярная зона.	Непонятен смысл сайта и предлагаемой услуги; В полной мере не отражает сущность наполнения сайта, предлагаемой услуги.

Острым вопросом является выбор типа сайта, и для решения данной проблемы наиболее оптимальным решением является Landing page. Это «легкий» сайт, созданный для привлечения целевой аудитории к товарам, услугам или акциям. Обычно на целевую страницу попадают благодаря переходу с контекстной рекламы или информации поисковиков. На подобных одностраничных сайтах расположена необходимая для посетителя информация в такой форме, чтобы он максимально сфокусировался на ней. Более того, правильный лендинг направлен на стимулирование желания совершить полезное действие: регистрация на сайте, оформление заказа, звонок в офис компании, подписка на рассылку. Благодаря такой направленности landing page обеспечивает повышение конверсии до 30% и более. Как правило, landing page имеют привлекательный и в меру лаконичный дизайн. Все делается для того, чтобы на странице отсутствовали факторы, отвлекающие от ее содержания.

Преимущества успешного лендинга:

– ориентируясь на конкретную целевую аудиторию при правильной раскрутке и рекламе, конверсия landing page будет намного больше, чем у обычных сайтов;

– благодаря простоте создания страницы она может быть готова к работе и запущена за несколько часов, а изменение информации на ней происходит в считанные минуты;

– посадочные страницы обычно быстро загружаются, даже на устройствах со слабым интернетом, посетителю не надо долго ждать;

– landing page – это весьма действенный и результативный инструмент, ведь если даже посетитель сайта ничего не приобретет или не закажет, велика вероятность, что он оставит свои данные. Таким образом, сформируется база потенциальных клиентов, которым в дальнейшем можно напоминать о себе по средствам e-mail рассылки;

– при помощи landing page можно успешно повысить эффект от контекстной рекламы;

– лендинг пейдж позволяет оценить и проанализировать объемы и целесообразность интернет-продаж;

– помогает увеличить продажи при некачественном основном сайте;

– низкая стоимость разработки.

Для того чтобы сайт работал, следует продумать его информационное наполнение:

1) Тип и формат представления информации: текст и картинки обозначающие программное обеспечение;

2) Структурирование информации:

– логотип и заголовок;

– демонстрация услуги;

– преимущества данного решения, в дальнейшем возможные акции;

– описание оффера;

– коммуникация.

3) Форма подачи информации:

Призыв используется в описание заголовка услуги. Призывает пользователя к действию нажать на кнопку заказа данной услуги или получения консультации по данному решению.

Аргументация используется в описание преимуществ программного решения, а также в описание оффера. Из названия следует, что пост-аргументация приводит доказательства определенной точки зрения. Его отличительная черта – вопрос «почему?», с которого начинается заголовок (например, почему предприятия должны радоваться появлению данного программного продукта: четыре причины).

4) Наполнение, расширение и актуализация информации. Landing page будет разбит на блоки:

Главный экран – его функция – произвести нужное впечатление на человека, информировать о том, куда он попал, мотивировать остаться и проскроллить страницу вниз.

Рассказ о проекте – подробное описание продукта или услуги:

- как устроен;
- как работает;
- на кого ориентирован;
- сколько стоит.

Рассказ о проекте – невозможно проигнорировать. Прежде чем объяснять выгоды или призывать совершить действие, необходимо убедиться, что человек понял, что именно вы предлагаете.

Понятные выгоды – этот раздел нужен, чтобы объяснить, чем вы отличаетесь от конкурентов. На большинстве рынков конкуренция высокая, поэтому необходимы доводы, почему человек должен выбрать вас.

Блоки доверия – эта группа блоков помогает сформировать кредит доверия. Отзывы, истории успеха, гарантии и сертификаты, партнеры и даже телефон и адрес офиса помогут развиртуализировать проект, показать, что он реальный и ему можно доверять.

Целевое действие – Бизнесу нужны клиенты, поэтому на лендинге должны быть блоки, которые будут генерировать лиды: формы заказа, подписки, обратной связи или телефон.

5) Дизайн сайта (обложка), наполнение:

Главный экран сайта – первое впечатление от компании. Есть всего несколько секунд, чтобы убедить пользователя остаться на странице.

Набор инструментов для этого небольшой: заголовок, подзаголовок, кнопка или форма, логотип, фон или изображение на фоне, меню, стрелка вниз.

Заголовок и подзаголовок – сделайте оффер – вдохновляющую фразу, которая передает суть проекта. Как правило, заголовок более эмоциональный, подзаголовок раскрывает смысл.

Форма или кнопка – для тех, кто сразу заинтересовался или зашел повторно, можно сразу на обложке добавить целевое действие.

Фон обложки – хорошая фотография, атмосферное видео, просто цвет, градиент или иллюстрация. Стоит обратить внимание на сочетание фона с текстом: фотография может быть удачной сама по себе, но если она неоднородная, пестрая, то она будет плохо работать с текстом. Видео нужно снимать, во-первых, плавно, во-вторых, лучше брать увеличенный фокус, чтобы все объекты были крупноваты.

Логотип – компании или продукта можно расположить как на самой обложке, так и в меню.

Стрелка вниз – Не обязательный элемент обложки. Стоит ориентироваться на аудиторию – если она консервативная, то стоит добавить. Новое поколение привыкло к скроллу, но кто-то может застопориться.

Меню – Также не обязательный элемент, но если он нужен для навигации, то выделить основные смысловые секции на странице, к которым нужен быстрый доступ.

На рисунке 24 продемонстрирована главная страница Web-ресурса. Предлагаемая услуга отображена на рисунке 25.

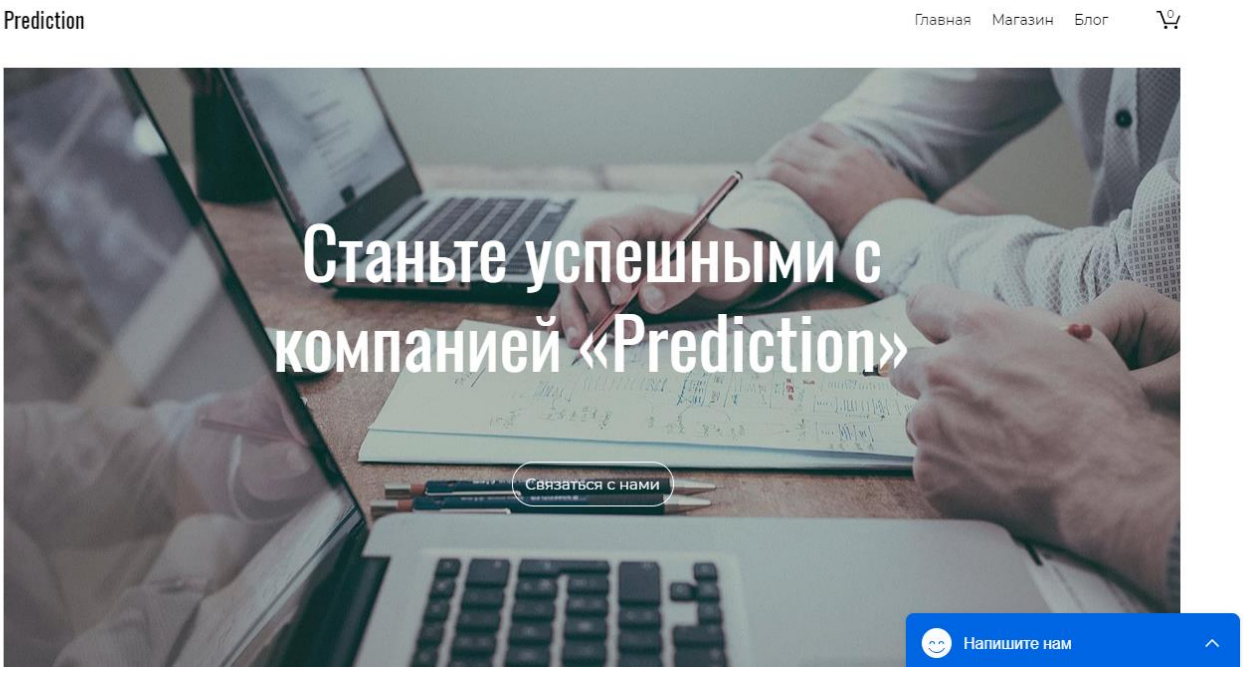

Рисунок 24 – Главная страница Landing page

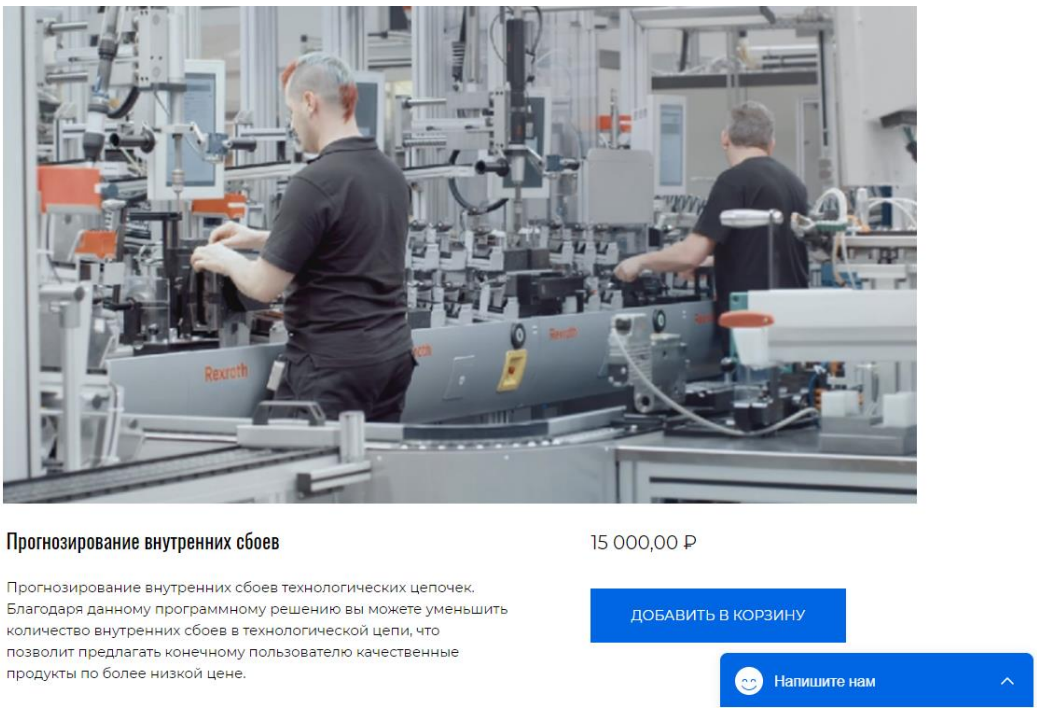

Рисунок 25 – Предлагаемая услуга

Для любого сайта важны инструменты работы с аудиторией. Ниже приведен примерный список, который следует реализовать:

1) Анализ поведения пользователей на сайте – владельцы ресурса могут следить за посещаемостью сервера, за наиболее популярными маршрутами по сайту, точками входа и выхода посетителей, временем, проведенным на каждой из страниц и т.д. Данная информация используется и для определения эффективности рекламных направлений, и для оптимизации структуры и навигации сайта. Получать подобные данные можно с помощью анализатора логов сайта или продвинутых счетчиков.

2) Консультации – с помощью интернет-технологий можно эффективно осуществлять информационную поддержку своих клиентов. Специалисты компании с помощью on-line конференций, чата или по e-mail могут отвечать на вопросы, давать консультации. В случае с конференцией это будет не столь оперативно (хотя и конференции могут проводиться в реальном режиме времени), но наглядно и информативно. Конференции имеют удобную древовидную структуру, а от-

сутствие необходимости отвечать сразу позволяет более тщательно подготовить ответ.

3) Чат – в дальнейшем планируется разработка чата. Он дает максимальную оперативность, ту же, что и телефонная линия, но при этом не надо платить за международные переговоры, а специалист службы поддержки может одновременно отвечать сразу на несколько вопросов. Самым же распространенным способом поддержки пользователей остаются консультации посредством электронной почты.

4) Патчи, драйвера и обновления программ – продавцы программного обеспечения, помимо консультаций и инструкций, посредством Интернета могут распространять как непосредственно свою продукцию, так и патчи и обновления к ней. А производители высокотехнологического оборудования могут выкладывать на сайте для скачивания последние версии драйверов устройств.

В таблице 7 отображена полная информация о возможностях сайта.

Таблица 7 – Возможности сайта

Наименование	Prediction
Информационное наполнение	Информация четко структурирована (в дальнейшем планируется добавлять новости и обновления в блог), используются различные форматы представления информации, тех. поддержка, адекватная и структурированная информация сайта, имеется расстановка информационных акцентов.
Функциональность	Представление товара и формирование заказа осуществляется по нажатию кнопки и переходу на вспомогательный экраны, а также окно формирования заказа, присутствует связь при помощи e-mail или теле-

	фона (в дальнейшем предусмотрена разработка чат-бота).

Окончание таблицы 7

Наименование	Prediction
Usability	Сайт эргономичен и удобен в использовании (присутствует простая и эффективная навигация, имеется карта местоположения, привычный вид полей и кнопок.
Дизайн	Дизайн дополняет и усиливает заложенную в сайт информацию и функционал, простота использования сайта благодаря легкому дизайну, возможность изменения дизайн-решений (гибкость), уникальность и запоминаемость.
Техническая реализация	Сайт написан при помощи движка Word-Press.
Маркетинг	На сайте присутствуют адреса, ссылки на сайт, средства сбора информации о посетителях сайта, посещаемость и поведенческая линия на сайте, работа с аудиторией сайта.

4.3 Медиапланирование и ценовая политика сайта

Медиапланирование – это планирование каналов и способов рекламы для составления медиаплана на основе прогнозов и полученных результатов.

Медиаплан для первого рекламного мероприятия по продвижению, созданного Web-ресурса, должен выполнять следующие условия:

1) Бюджет – 10 000-30 000 $ (по нынешнему курсу рубль-$ = 64,60);
2) Время рекламной компании – 4 недели;
3) Задача рекламной компании – привлечение посетителей (раскрутка нового ресурса).

Для рекламы, конечно же, будут использоваться самые распространенные и популярные рекламные площадки такие как Вконтакте и Яндекс. В таблице 8 продемонстрирована полная информация по выбору той или иной площадки.

Для начала, чтобы раскрутить бренд следует максимизировать сиюминутную прибыль. Иначе говоря – извлечь как можно больше денег из каждой продажи предоставляемой услуги, даже если это сокращает количество потенциальных покупателей. В итоге, будет меньше клиентов, но и количество проблем по их обслуживанию также сократится. И, кроме того, каждый из клиентов принесет больший доход.

Расчет будет производиться по общим издержкам – сумма постоянных и переменных издержек.

"Снятие сливок". Предлагая новую революционную услугу, стоит изначально установить на нее высокую цену, так как тот, кто чувствителен к нововведениям – нечувствителен к цене. Затем снижаем цену и "снимаем сливки" со следующего слоя покупателей и так далее. В конечном счете, цена падает под воздействием того, что товар укрепляет свои позиции на рынке и конкуренты снижают цены. Так же стоит задуматься о скидках на повторное приобретение услуги если в таковой нуждаются.

Таблица 8 – Медиаплан

Рекламные каналы	Дополнительная информация	Посыл	Формат	Общая стоимость рекламы, руб.	Число публикаций	CPT, руб.	Частота	Бюджет
Контекстная реклама в Яндексе	Только горячие запросы	Инновационная разработка	Спец. размещение	120 000	1	1 500	1	120 000
РСЯ	Публикация рекламных постов	Инновационная разработка	Графические баннеры	450 000	3	40	4	1 350 000
Группа Вконтакте	Публикация рекламных постов	Инновационная разработка	Промо-пост	10 000	2	833	1,2	20 000
Таргетированная реклама Вконтакте	Публикация рекламных постов	Инновационная разработка	Лид-форма	100 000	3	1 500	1,2	300 000
Mail.Ru Group	Рекламный баннер	Инновационная разработка	Графические баннеры	55 000	2	900	1	110 000
Итого:				735 000	11			1 900 000

Выводы по главе 4

Составление «дорожной карты» — важный этап в создании инновационного продукта. Благодаря составлению которой, было спланировано веб-представительство будущей компании, деятельность которого направлена на предоставление услуг по предотвращению и сокращению технологических сбоев производственной линии на предприятии.

Была составлена таблица примерной доходности веб-ресурса. Рассмотрена целевая аудитория и проведен SWOT-анализ для определения внешних и внутренних факторов, влияющих на возможности работы в Интернете и формировании интернет стратегии.

Подходящим вариантом типизации сайта был выбран Landing page с доменным именем Prediction.com/.ru. После чего, продумано информационное наполнение, продемонстрирован первоначальный предполагаемый вид и раскрыты возможности сайта.

В дальнейшем планируется разработать инструменты работы с аудиторией, такие как:

- анализ поведения пользователей на сайте;
- консультации;
- чат;
- патчи, драйвера и обновления программ.

Разработан медиаплан и ценовая политика.

ЗАКЛЮЧЕНИЕ

1. В ходе исследования предметной области были изучена интеллектуальные производственные линии их архитектура, мониторинг производства и виды сбоев, возникающих при производстве. Которая предоставила возможность компании BOSCH реализовать сбор больших данных на своем производстве. Это позволило собрать и оценить данные для проверки качества в производстве при управлении технологической линией. Также в работе отмечена важность использования именно методов машинного обучения для прогнозирования сбоев, возникающих при производстве на технологической линии.

2. В выпускной квалификационной работе были изучены методы классификации прогнозирования сбоев технологических линий. Рассмотрены популярные алгоритмы, которые используются в машинном обучении для решения данных проблем, такие как k ближайших соседей (k Nearest Neighbor), случайный лес (Random forest), классификатор экстра-деревьев (Extra-trees classifier), градиентный бустинг (XGBoost) и легкий градиентный бустинг (LightGBM). Проанализированы используемые метрики качества и процедура кросс-валидации. Но для того чтобы понять в пользу какого алгоритма сделать выбор, который будет наилучшим для решения данной задачи, стоило разобраться в предоставленных данных и задачах, которые нужно решить.

3. Была раскрыта тема проекта и ее цель, которая подразумевает прогнозирование того, какие части не пройдут контроль качества (представленные в виде: «Ответ» = 1). Показано описание набора данных предоставленных компанией Bosch, продемонстрированных в виде табличных данных на рисунке. В связи с тем, что данных огромное количество необходимо было первостепенно произвести отчистку (предварительную обработку) данных. Все данные по итогу были разбиты на категориальные, числовые и особенности даты. Были проведены исследования эффективности методов машинного обучения: k ближайших соседей (k Nearest Neighbor), случайный лес (Random forest), классификатор экстра-

деревьев (Extra-trees classifier), градиентный бустинг (XGBoost) и легкий градиентный бустинг (LightGBM). Исходя из исследований эффективности методов и набора данных, был сделан выбор в пользу метода градиентного бустинга XGBoost с описанием преимуществ, в сравнении с другими методами. Следующим шагом был выбор признаков и оптимизация параметров модели таких, как: ETA и MaxDepth. Также не маловажную роль в измерении производительности задачи классификации при различных настройках порогов является кривая AUC-ROC. При заданных параметрах был получен тренинг AUC 0,718 0,001, что является хорошей базовой моделью, но имеет еще возможности для улучшения.

4. Разработана коммерциализация проекта по этапам. Изначально была разработана дорожная карта коммерциализации данного проекта, которая подразумевает наглядное представление пошагового сценария развития, в которую входит – планирование стратегии, исходя из задач, для решения поставленной цели, описаны источники доходов по видам предоставляемых услуг и их стоимость в рублях. Проведена оценка потенциальных возможностей Интернета для бизнеса, в которой были рассмотрены: целевая аудитория, конкурентная среда и потенциальные партнеры. Также продемонстрирована таблица SWOT-анализа. По потенциальным возможностям Интернета было выбрано создать сайт по предоставлению услуги прогнозирования сбоев технологических линий другим компаниям. Для решения данной задачи первостепенным было принято решение выбора доменного имени для сайта, а также был выбран тип и информационное наполнение сайта. Следующим шагом был выбор инструментов для работы с аудиторией сайта. В табличном виде представлен мониторинг сайта. Описано продвижение и ценовая политика сайта. Разработан медиаплан, также продемонстрированный в табличном виде.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Abraham, A. and Grosan, C. 2005. Genetic programming approach for fault modeling of electronic hardware. In IEEE Proceedings Congress on Evolutionary Computation (CEC’05). Vol. 2. Edinburgh, UK, 1563-1569.

2. Aitchison, J. and Dunsmore, I. R. 1975. Statistical Prediction Analysis. Cambridge University Press.

3. Ankita Mangal, Elizabeth A. Holm, "Applied Machine Learning to Predict Stress Hotspots I: Face Centered Cubic Materials", International Journal of Plasticity, 2018.

4. Ankita Mangal, Elizabeth A. Holm, "Applied Machine Learning to Predict Stress Hotspots II: Hexagonal close packed materials", International Journal of Plasticity, 2018.

5. B. W. Matthews, “Comparison of the predicted and observed secondary structure of T4 phage lysozyme,” Biochimica et Biophysica Acta (BBA)-Protein Struc- ture, vol. 405, no. 2, pp. 442-451, 1975.

6. Basseville, M. and Nikiforov, I. 1993. Detection of abrupt changes: theory and application. Prentice Hall.

7. Blischke, W. R. and Murthy, D. N. P. 2000. Reliability: Modeling, Prediction, and Optimization. Probability and Statistics. John Wiley and Sons.

8. Cavafy, C. P. 1992. But the wise perceive things about to happen. In Collected Poems, G. Savidis, Ed. Princeton University Press.

9. Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In Proceedings of the 22Nd ACM SIGKDD International Conference on Knowl- edge Discovery and Data Mining, KDD ’16, pages 785-794, New York, NY, USA. ACM.

10. Cleveland, W. et al. 1979. Robust locally weighted regression and smoothing scatterplots. Journal of the American Statistical Association 74, 368, 829-836.

11. Coleman, D. and Thompson, C. 2005. Model Based Automation and Management for the Adaptive Enterprise. In Proceedings of the 12th Annual Workshop of HP OpenView University Association. 171-184.

12. Cover, T. Hart, P, “Nearest Neighbor Pattern Classification”, IEEE Transaction on Information Theory, vol. 13, pp. 21-27, 1967.

13. Croci, F., M. Perona, and A. Pozzetti. "Work Force Management in Automated Assembly Systems." International Journal of Production Economics. 1 March 2000.

14. Crowell, J., Shereshevsky, M., and Cukic, B. 2002. Using fractal analysis to model software aging. Tech. rep., West Virginia University, Lane Department of CSEE, Morgantown, WV. May.

15. Denson, W. 1998. The history of reliability prediction. IEEE Transactions on Reliability 47, 3 (Sep.), 321-328.

16. Discenzo, F., Unsworth, P., Loparo, K., and Marcy, H. 1999. Self-diagnosing intelligent motors: a key enabler for nextgeneration manufacturing systems. In IEE Colloquium on Intelligent and Self-Validating Sensors.

17. E. Alpaydin, “An Introduction to Machine Learning” The MIT press, Cambridge, Massachusetts, London, England, 2004.

18. Elbaum, S., Kanduri, S., and Amschler, A. 2003. Anomalies as precursors of field failures. In IEEE Proceedings of the 14th International Symposium on Software Reliability Engineering (ISSRE 2003). 108-118.

19. Felix Reinhart, Sebastian von Enzberg, Arno Kühn, Roman Dumitrescu, Machine Learning for Cyber Physical Systems, vol. 11, pp. 25, 2017.

20. Flach, P. A. 2004. The many faces of roc analysis in machine learning. Tutorial at International Conference on Machine Learning (ICML’04). http://www.cs.bris.ac.uk/ flach/ICML04tutorial/.

21. Freund, Y. and Schapire, R. E. (1996). Experiments with a new boosting algorithm. In Saitta, L., editor, Proceedings of the Thirteenth International Conference on Machine Learning (ICML 1996), pages 148–156. Morgan Kaufmann.

22. Fu, S. and Xu, C.-Z. 2007. Quantifying temporal and spatial fault event correlation for proactive failure management. In IEEE Proceedings of Symposium on Reliable and Distributed Systems (SRDS 07).

23. Hastie, T., Tibshirani, R., and Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition. Springer Series in Statistics. Springer.

24. Ho, D. W. C., Zhang, P. A., and Xu, J. 2001. Fuzzy wavelet networks for function learning. IEEE Transactions on Fuzzy Systems 9, 1, 200-211.

25. Hoffmann, G. A. 2004. Adaptive transfer functions in radial basis function (rbf) networks. In Proceedings of 4th International Conference on Computational Science (ICCS 2004), M. Bubak, G. D. van Albada, P. M. A. Sloot, et al., Eds. LNCS, vol. 3037. Springer-Verlag, 682-686.

26. Hoffmann, G. A. 2006. Failure Prediction in Complex Computer Systems: A Probabilistic Ap- proach. Shaker Verlag.

27. Hoffmann, G. A., Trivedi, K. S., and Malek, M. 2006. A best practice guide to resource forecasting for the apache webserver. In IEEE Proceedings of the 12th International Symposium Pacific Rim Dependable Computing (PRDC'06). University of California, Riverside, USA.

28. IEC: International Technical Comission, Ed. 2002. Dependability and Quality of Service, 2 ed. IEC, Chapter 191.

29. J. H. Friedman, "Greedy function approximation: a gradient boosting machine," Annals of statistics, pp. 1189-1232, 2001.

30. Kiciman, E. and Fox, A. 2005. Detecting application-level failures in component-based internet services. IEEE Transactions on Neural Networks 16, 5 (Sep.), 1027-1041.

31. Korbicz, J., Ko´scielny, J. M., Kowalczuk, Z., and Cholewa, W., Eds. 2004. Fault Diagnosis: Models, Artificial Intelligence, Applications. Springer Verlag.

32. Lal, R. and Choi, G. 1998. Error and failure analysis of a unix server. In IEEE Proceedings of third International High-Assurance Systems Engineering Symposium (HASE). IEEE Computer Society Washington, DC, USA, 232-239.

33. Laprie, J.-C. and Kanoun, K. 1996. Software reliability and system reliability. In Handbook of software reliability engineering, M. R. Lyu, Ed. McGraw-Hill, Chapter 2, 27-69.

34. Lunze, J. 2003. Automatisierungstechnik, 1 ed. Oldenbourg.

35. Maloney, David. "New Roots." Modern Materials Handling. October 2003.

36. Manning, C. D. and Schu¨tze, H. 1999. Foundations of Statistical Natural Language Processing. The MIT Press, Cambridge, Massachusetts.

37. Melliar-Smith, P. M. and Randell, B. 1977. Software reliability: The role of programmed exception handling. SIGPLAN Not. 12, 3, 95-100.

38. Meng, H., Di Hou, Y., and Chen, Y. 2007. A rough wavelet network model with genetic algorithm and its application to aging forecasting of application server. In IEEE Procedings of International Conference on Machine Learning and Cybernetics. Vol. 5.

39. Murray, J., Hughes, G., and Kreutz-Delgado, K. 2003. Hard drive failure prediction using non-parametric statistical methods. Proceedings of ICANN/ICONIP.

40. Musa, J. D., Iannino, A., and Okumoto, K. 1987. Software Reliability: Measurement, Predic- tion, Application. McGraw-Hill.

41. N. Roussopoulos, S. Kelley, F. Vincent, “Nearest Neighbor Queries”, Proceedings of the 1995 ACM SIGMOD international conference on Management of data, pp. 71-79, 1995.

42. Neville, S. W. 1998. Approaches for early fault detection in large scale engineering plants. Ph.D. thesis, University of Victoria.

43. Nieble, Benjamin, and Andris Freivalds. Methods, Standards, and Work Design. July 2002.

44. Pettitt, A. 1977. Testing the normality of several independent samples using the anderson-darling statistic. Applied Statistics 26, 2, 156-161.

45. Q.M. Jonathan Wu,” Class Notes- Machine Learning and Computer Vision”, University of Windsor, Windsor, ON, Canada, 2007.

46. R. Duda, P. Hart, and D. Stork, “Pattern Classification”, 2nd ed., Wiley Interscience.

47. Siewiorek, D. P. and Swarz, R. S. 1998. Reliable Computer Systems, third ed. A. K. Peters, Ltd., Wellesley, MA.

48. Smith, T. and Waterman, M. 1981. Identification of common molecular subsequences. Journal of Molecular Biology 147, 195-197.

49. T.M. Mitchell, “Machine Learning”, McGraw-Hill, 1997

50. Tang, D. and Iyer, R. 1993. Dependability measurement and modeling of a multicomputer system. IEEE Transactions on Computers 42, 1 (Jan.), 62-75.

51. Umble, Michael, Van Gray, and Elisabeth Umble. "Improving Production Line Performance." IIE Solutions. November 2000.

52. van Rijsbergen, C. J. 1979. Information Retrieval, second ed. Butterworth, London.

53. Vesely, W., Goldberg, F. F., Roberts, N. H., and Haasl, D. F. 1981. Fault tree handbook. Tech. Rep. NUREG-0492, U.S. Nuclear Regulatory Commission, Washington, DC.

54. Ward, A., Glynn, P., and Richardson, K. 1998. Internet service performance failure detection. SIGMETRICS Performance Evaluation Review 26, 3, 38-43.

55. Whitfield, Kermit. "Assembly: How Standard Can You Get?" Automotive Design & Production. March 2004.

56. Wong, K. C. P., Ryan, H., and Tindle, J. 1996. Early warning fault detection using artificial intelligent methods. In Proceedings of the Universities Power Engineering Conference.

57. Xiaoming Fan, Xuan Zhu, Kuei Chi Kuo, Cong Lu, Jason Wu, "Big data analytics to improve photomask manufacturing productivity", Industrial Engineering and Engineering Management (IEEM) 2017 IEEE International Conference on, pp. 2341-2345, 2017.