Содержание

Об авторах

Рич Кольер (Rich Collier) – архитектор решений в Elastic. Он присоединился к команде Elastic после приобретения Prelert. Рич имеет более чем 20-летний опыт работы в качестве архитектора решений и системного инженера предпродажной подготовки программного обеспечения, оборудования и сервисных решений. Профессиональные интересы Рича включают аналитику больших данных, машинное обучение, обнаружение аномалий, обнаружение угроз, обеспечение безопасности, управление производительностью приложений, веб-приложения и технологии контакт-центров. Рич проживает в Бостоне, штат Массачусетс.

Камилла Монтонен (Camilla Montonen) – старший инженер по машинному обучению в Elastic.

Бахаалдин Азарми (Bahaaldine Azarmi), или коротко Баха, – архитектор решений в Elastic. До этого Баха был соучредителем ReachFive, платформы маркетинговых данных, ориентированной на поведение пользователей и социальную аналитику. Баха также сотрудничал с различными поставщиками программного обеспечения, такими как Talend и Oracle, где он занимал должности архитектора решений и системного архитектора. Баха является автором нескольких книг, в том числе *Learning Kibana 5.0*, *Scalable Big Data Architecture* и *Talend for Big Data*. Живет в Париже и имеет степень магистра компьютерных наук Парижского технологического института.

О рецензентах

Апурва Джоши (Apoorva Joshi) в настоящее время работает специалистом по безопасности данных в Elastic (ранее Elasticsearch), где она занимается применением машинного обучения для обнаружения вредоносных программ в конечных точках системы. До Elastic работала научным сотрудником в FireEye, где исследовала применение машинного обучения в задачах безопасности электронной почты. У нее разностороннее инженерное образование: бакалавр электротехники и магистр информационных технологий (с акцентом на машинное обучение).

Лицзюань Чжун (Lijuan Zhong) – опытный инженер в области технологий Elastic и облачных вычислений. У нее степень магистра информационных технологий и почти 20-летний опыт работы в сфере информационных технологий и телекоммуникаций. Сейчас сотрудничает с Netnordic – основным партнером Elastic в Швеции. Она начала свой путь в Elastic в 2019 году и стала сертифицированным инженером Elastic, также прошла курс машинного обучения Стэнфордского университета. Возглавляет множество образовательных программ и проектов, связанных с Elastic и машинным обучением, и клиенты всегда очень довольны результатом. Она была соорганизатором конференции Elastic Stockholm в 2020 г., приняла участие в конференции сообщества Elastic в 2021 г. и выступила с докладом о машинном обучении с помощью Elastic Stack. В 2021 году была удостоена бронзовой медали за вклад в развитие Elastic.

Предисловие

Elastic Stack, ранее известный как ELK Stack, представляет собой комплексное решение для анализа журналов, которое помогает пользователям эффективно получать, обрабатывать и анализировать данные поиска. Благодаря применению машинного обучения – ключевой особенности решения – Elastic Stack делает этот процесс еще более эффективным. Эта книга содержит всесторонний обзор функций машинного обучения Elastic Stack как для анализа данных временных рядов, так и для классификации, регрессии и обнаружения выбросов.

Знакомство с экосистемой Elastic Stack начинается с интуитивно понятного объяснения концепций машинного обучения. Затем под руководством авторов вы выполните анализ временных рядов для различных типов данных, таких как файлы журналов, сетевые потоки, показатели приложений и финансовые данные. По мере прочтения глав вы научитесь использовать машинное обучение в Elastic Stack для ведения журнала, обеспечения безопасности и отслеживания показателей. Наконец, вы узнаете, как анализ фреймов открывает доступ к совершенно новым сценариям использования данных, в которых вам поможет машинное обучение.

После прочтения этой книги вы приобретете практический опыт совместного использования технологии машинного обучения и Elastic Stack, а также знания, необходимые для включения машинного обучения в вашу платформу распределенного поиска и анализа данных.

Для кого эта книга

Если вы профессионал в области данных и хотите получить представление о технологиях Elasticsearch, не прибегая к помощи специалиста по машинному обучению и не разрабатывая собственные решения, то эта книга про совместное применение машинного обучения и Elastic Stack для вас. Вы также найдете эту книгу полезной, если хотите интегрировать машинное обучение с вашими приложениями для мониторинга, обеспечения безопасности и аналитики. Чтобы извлечь из данной книги максимальную пользу, необходимо знать и уметь применять на практике Elastic Stack.

Какие темы охватывает эта книга

Глава 1 служит введением в тему и справочным пособием по историческим проблемам ручного анализа данных в IT и технологиях безопасности. В этой главе также представлен всесторонний обзор базовых принципов работы

машинного обучения Elastic (Elastic ML), чтобы читатель получил полное представление о том, что происходит «за кулисами».

В *главе 2* объясняется, каким образом применяются возможности машинного обучения в Elastic Stack, а также подробно описывается теория работы алгоритмов Elastic ML. Кроме того, в этой главе дается подробное объяснение логистики операций машинного обучения применительно к Elastic.

В *главе 3* подробно рассматриваются методы автоматического обнаружения аномалий с обучением без учителя, которые лежат в основе анализа временных рядов.

В *главе 4* показано, что сложные модели временных рядов Elastic ML можно использовать не только для обнаружения аномалий. Возможности прогнозирования позволяют пользователям экстраполировать тенденции и поведение в будущее, чтобы помочь в решении таких задач, как планирование мощности.

В *главе 5* рассказано, как детально истолковать результаты обнаружения и прогнозирования аномалий и использовать их в своих целях в визуализации данных, информационных панелях и инфографике.

В *главе 6* рассмотрены различные методы интеграции возможностей упреждающего уведомления Elastic и данных, полученных с помощью машинного обучения, чтобы сделать обнаружение аномалий еще более эффективным.

В *главе 7* показано, как использование Elastic ML для проверки целостности и анализа данных из разрозненных источников в коррелированных представлениях дает аналитику преимущество с точки зрения наследования подходов.

В *главе 8* объясняется, как обнаружение аномалий используется другими приложениями в Elastic Stack для повышения эффективности анализа данных.

В *главе 9* рассмотрен анализ фрейма данных, его отличие от обнаружения аномалий временных рядов и рассказано, какие инструменты доступны пользователю для загрузки, подготовки, преобразования и анализа данных с помощью Elastic ML.

В *главе 10* описано применение анализа фреймов в сочетании с Elastic ML в аналитике данных.

В *главе 11* говорится о возможности классификационного анализа фреймов данных в сочетании с Elastic ML.

В *главе 12* рассмотрено использование регрессионного анализа фреймов данных в сочетании с Elastic ML.

Глава 13 описывает использование обученных моделей машинного обучения для логического вывода – прогнозирования выходных значений во время реальной работы системы.

Приложение включает в себя множество практических советов, которые отчасти выходят за рамки других глав. Эти полезные советы помогут вам максимально эффективно использовать Elastic ML.

Как получить максимальную отдачу от этой книги

Чтобы получить максимальную отдачу от этой книги, вам понадобится компьютер с хорошим подключением к интернету и учетной записью Elastic.

Скачивание исходного кода примеров

Вы можете скачать файлы примеров кода для этой книги с GitHub по адресу https://github.com/PacktPublishing/Machine-Learning-with-Elastic-Stack-Second-Edition. Если выйдет обновление кода, оно появится в репозитории GitHub.

Условные обозначения и соглашения, принятые в книге

В книге используются следующие типографские соглашения.

Курсив – используется для смыслового выделения важных положений, новых терминов, имен команд и утилит, а также слов и предложений на естественном языке.

`Моноширинный шрифт` – применяется для листингов программ, а также в обычном тексте для обозначения имен переменных, функций, типов, объектов, баз данных, переменных среды, операторов, ключевых слов и других программных конструкций и элементов исходного кода.

`Моноширинный полужирный шрифт` – используется для обозначения команд или фрагментов текста, которые пользователь должен ввести дословно без изменений, а также в листингах программ, если необходимо обратить особое внимание на фрагмент кода.

`Моноширинный курсив` – применяется для обозначения в исходном коде или в командах шаблонных меток-заполнителей, которые должны быть заменены соответствующими контексту реальными значениями.


Советы оформлены так.


Примечания оформлены так.


Важные примечания оформлены так.

Часть I

ЗНАКОМСТВО С МАШИННЫМ ОБУЧЕНИЕМ И ELASTIC STACK

В этой части представлено обобщенное описание Elastic ML – не только с точки зрения алгоритмов, но и с точки зрения организации работы программного обеспечения в Elastic Stack.

Эта часть книги состоит из следующих глав:

- главы 1 «Машинное обучение в информационных технологиях»;
- главы 2 «Подготовка и использование Elastic ML».

Глава 1

Машинное обучение в информационных технологиях

Десять лет назад идея использования технологий машинного обучения (ML) в IT-структурах или IT-безопасности казалась чем-то вроде научной фантастики. Однако сегодня это одно из самых популярных модных словечек, используемых поставщиками программного обеспечения. Очевидно, что за минувшее десятилетие произошел серьезный сдвиг как в осознании потребности в технологии ML, так и в возможностях, которые она предоставляет. Эта эволюция важна для понимания того, как появился инструмент Elastic ML и для решения каких проблем он был разработан.

Эта глава посвящена обзору истории и концепций, лежащих в основе работы Elastic ML. В ней также представлены различные виды анализа, которые можно провести, и задачи, которые можно решить с помощью Elastic ML. В частности, мы рассмотрим следующие темы:

- преодоление исторических вызовов в IT;
- что нам делать с потоком данных;
- причины появления автоматического обнаружения аномалий;
- машинное обучение без учителя и с учителем;
- использование машинного обучения без учителя для обнаружения аномалий;
- применение машинного обучения с учителем в аналитике фреймов данных.

Преодоление исторических вызовов в IT

К специалистам по поддержке IT-структур и архитекторам решений предъявляют высокие требования. Их роль не ограничивается внедрением новых передовых проектов и технологий для бизнеса; они также должны поддерживать безопасную и бесперебойную работу развернутых в настоящее время приложений. Сегодняшние приложения значительно сложнее, чем

когда-либо прежде, – они разбиты на множество компонентов, распределены и, возможно, виртуализированы/контейнеризированы. Они могут быть разработаны с использованием Agile-методики или аутсорсинговой командой. Вдобавок они, скорее всего, постоянно меняются. Некоторые команды DevOps заявляют, что обычно они вносят более 100 изменений в день в действующую производственную систему. Пытаться понять состояние и поведение современного приложения уровня предприятия – все равно что механику пытаться отремонтировать автомобиль, пока он едет по шоссе.

Аналитики по безопасности в области IT тоже с трудом справляются с повседневной работой, но, очевидно, у них другой фокус внимания – обеспечение безопасности предприятия и устранение возникающих угроз. Хакеры, вредоносные программы и инсайдеры-мошенники стали настолько распространенными и изощренными, что по мнению большинства специалистов по безопасности сегодня вопрос заключается не в том, будет ли взломана организация, а в том, насколько быстро она узнает об этом (если вообще узнает). Очевидно, что узнать о взломе как можно раньше (до того, как будет нанесен слишком большой ущерб) предпочтительнее, чем услышать об этом впервые от правоохранительных органов или из вечерних новостей.

Но что же нам делать? Возможно, проблема в том, что экспертам по приложениям и аналитикам службы безопасности не хватает данных, которые помогли бы им эффективно выполнять свою работу? На самом деле в большинстве случаев ситуация противоположная. Многие IT-специалисты и организации тонут в данных.

Что нам делать с потоком данных?

IT-отделы десятилетиями вкладывали силы и средства в инструменты мониторинга, и нередко в их распоряжении есть дюжина или более инструментов, активно собирающих и архивирующих данные, объем которых измеряется в терабайтах или даже петабайтах в день. Источники этих данных чрезвычайно вариативны – от элементарной статистики на уровне инфраструктуры и сети до результатов глубокой диагностики и/или файлов журналов системы и приложений.

Ключевые показатели эффективности (key performance indicators, KPI) на уровне бизнеса также можно отслеживать, иногда включая данные об опыте конечного пользователя. Глубина и широта охвата доступных данных сегодня больше, чем когда-либо. Для обнаружения возникающих проблем или угроз, скрытых в этих данных, традиционно использовалось несколько основных подходов к преобразованию «сырых» данных в информационные объекты:

- **фильтрация/поиск:** некоторые инструменты предоставляют пользователю возможность фильтрации или поиска, чтобы сократить данные до более удобоваримого ограниченного набора. Хотя эта возможность чрезвычайно полезна, она чаще всего используется бессистемно, в основном когда возникает подозрение в наличии проблемы. Даже в этом

случае успех обычно зависит от способности пользователя понять, что он ищет, и от его уровня опыта – как от знания предыдущих ситуаций, так и от навыков использования самой технологии поиска;

- **визуализация:** панели мониторинга, диаграммы и виджеты также чрезвычайно полезны для понимания того, что происходит с данными, и выявления тенденций. Однако визуализации пассивны по своей сути и требуют постоянного наблюдения на предмет обнаружения значимых отклонений. Когда количество собираемых и отображаемых на экране показателей превышает возможности человеческого восприятия (или даже площадь экрана для их отображения), полезность визуализации резко снижается;
- **пороговые значения/правила:** чтобы обойти физические ограничения визуализации и внести в наблюдение за данными активный компонент (т. е. реакцию на изменение данных), многие инструменты позволяют пользователю определять правила или действия, которые запускаются при соблюдении определенных условий или возникновении определенных зависимостей между элементами данных. Однако маловероятно, что вы сможете реалистично определить все подходящие рабочие диапазоны или смоделировать все возможные зависимости в современных сложных и распределенных приложениях. Кроме того, количество и скорость изменений в приложении или среде могут быстро сделать любой статический набор правил бесполезным.

Аналитики обнаружили, что погрязли в ложных срабатываниях предупреждающих систем – широко известная проблема мальчика, который кричал о несуществующих волках, – что приводит к возмущению пользователей по поводу инструментов, генерирующих предупреждения, и скептицизму по поводу ценности, которой обладает такое предупреждение.

В конечном счете стало ясно, что нужен другой подход – такой, который не обязательно был бы полным отказом от прошлых методов, но мог бы обеспечить уровень автоматизации и значимого увеличения эмпирической ценности данных. Посмотрим правде в глаза: люди несовершенны – у нас есть скрытые предубеждения и ограничения способности запоминать информацию, и мы легко отвлекаемся и утомляемся. Алгоритмы машинного обучения при правильном использовании могут легко восполнить эти недостатки.

Причины появления автоматического обнаружения аномалий

Машинное обучение, будучи очень обширной темой, охватывающей многие области, от беспилотных автомобилей до компьютерных программ, приносящих выигрыш в играх, стало естественным кандидатом на роль эффективного решения. Если вы понимаете, что большинство задач мониторинга приложений или поиска угроз безопасности представляют собой всего лишь вариации на тему *поиска отличий от обычного хода событий*, тогда дисципли-

на обнаружения аномалий становится естественным местом для применения методов машинного обучения IT-специалистами.

Однако в науке об обнаружении аномалий, безусловно, нет ничего нового. Многие очень умные люди в течение долгих лет исследовали и применяли различные алгоритмы и методы. Но на практике обнаружение аномалий в IT-данных сопровождается некоторыми специфическими ограничениями, которые делают интересные с академической точки зрения алгоритмы непригодными для работы. Речь о следующих ограничениях:

- **своевременность.** Уведомление об отключении, нарушении или другой существенной аномальной ситуации должно стать известно как можно быстрее, чтобы смягчить последствия. Стоимость простоя или риск продолжения нарушения безопасности сводятся к минимуму, если быстро устранить проблему. Алгоритмы, которые не успевают отслеживать сегодняшние IT-данные в реальном времени, имеют ограниченную ценность;
- **масштабируемость.** Как упоминалось ранее, в современных IT-средах объем, скорость и вариативность IT-данных продолжают стремительно расти. Алгоритмы, которые занимаются мониторингом и анализом огромных данных, должны иметь возможность линейного масштабирования сообразно с данными, иначе спустя какое-то время они утратят применимость;
- **эффективность.** Бюджеты IT-подразделений часто подвергают тщательной проверке на предмет нерациональных расходов, и многие организации постоянно пытаются их урезать. Закупка дополнительного парка суперкомпьютеров для запуска неэффективных алгоритмов вряд ли будет утверждена руководством компании. Скорее, в качестве частичного решения придется использовать скромное стандартное оборудование с типичными характеристиками;
- **обобщаемость.** Хотя узкоспециализированная наука о данных часто является лучшим способом решения конкретной информационной проблемы, разнообразие данных в IT сформировало потребность в подходе, который применим в большинстве случаев. Повторное использование одних и тех же методов намного более рентабельно в долгосрочной перспективе;
- **адаптивность.** Постоянно меняющаяся IT-среда быстро сделает жесткий алгоритм бесполезным. Обучение и переподготовка модели ML превращаются в бесконечное занятие и фактически в пустую трату времени, чего мы не можем себе позволить;
- **точность.** Мы уже говорили, что усталость от ложных предупреждений из-за устаревших пороговых и основанных на правилах систем является реальной проблемой. Замена одного генератора ложных тревог на другой никого не впечатлит;
- **простота использования.** Даже если все ранее упомянутые ограничения могут быть удовлетворены, любое решение, для реализации которого потребуется армия специалистов по данным, окажется слишком дорогостоящим и будет немедленно отвергнуто.

Итак, мы подошли к сути задачи – созданию быстрого, масштабируемого, точного и недорогого решения для обнаружения аномалий, которое все будут охотно использовать, потому что оно работает безупречно. Без проблем!

Как бы пугающе это ни звучало на самом деле, основатель и технический директор Prelert Стив Додсон принял этот вызов еще в 2010 году. Хотя Додсон, несомненно, заложил в фундамент компании свои академические знания, технология, которая в конечном итоге превратилась в Elastic ML, зародилась в муках попыток устранения реальных сбоев приложений уровня предприятия. Первая из них – досадный периодический сбой в работе торговой платформы в крупной лондонской финансовой компании. Додсон и несколько инженеров, присоединившихся к этому предприятию, помогли команде банка использовать технологию обнаружения аномалий для автоматического поиска «иголки в стоге сена», что позволило аналитикам сосредоточиться на небольшом наборе соответствующих показателей и записей в журналах событий, которые вызывали подозрения. Выявление первопричины (отказ службы, восстановление которой вызывало каскадный сбой других служб и причиняло ущерб) в конечном итоге обеспечило стабильную работу приложения и избавило банк от необходимости тратить много денег на другое решение – дорогостоящее обновление сетевой инфраструктуры.

Однако со временем стало ясно, что даже этот показательный успех был только началом. Спустя несколько лет и несколько тысяч примеров использования в реальном мире возник союз Prelert и Elastic – сочетание платформы, делающей большие данные легкодоступными, и технологий, которые помогли преодолеть ограничения традиционных методов анализа.

Перенесемся в 2021 год, спустя полные 5 лет после объединения усилий, когда Elastic ML прошел долгий путь в развитии и расширении возможностей платформы ML. Это второе издание книги включает в себя обновления, внесенные в Elastic ML за прошедшие годы, в том числе интеграцию с некоторыми решениями Elastic, касающимися наблюдаемости и безопасности. Во второе издание мы добавили введение в аналитику фреймов данных, которая подробно обсуждается в третьей части книги. Чтобы получить обоснованное и глубокое понимание того, как работает Elastic ML, нам сначала нужно рассмотреть терминологию и идеи, а потом двигаться дальше.

Машинное обучение без учителя и с учителем

Хотя существует множество подтипов машинного обучения, два самых известных (и относящихся к Elastic ML) – это *обучение без учителя* и *обучение с учителем*.

В машинном обучении без учителя нет внешнего руководства или указаний со стороны людей. Другими словами, алгоритмы должны изучать (и моделировать) шаблоны данных исключительно самостоятельно. В целом самая большая проблема здесь состоит в том, чтобы алгоритмы точно выявляли отклонения от нормальных шаблонов входных данных, дабы обеспечить

вывод модели, значимый для пользователя. Если алгоритм не может этого сделать, то он бесполезен и непригоден для использования. Следовательно, алгоритмы должны быть достаточно надежными и способны учитывать все тонкости поведения входных данных.

В машинном обучении с учителем для моделирования желаемого результата используются размеченные входные данные (часто многомерные). Ключевое отличие от машинного обучения без учителя состоит в том, что человек априори решает, какие переменные использовать в качестве входных данных, а также предоставляет «достоверные» примеры ожидаемой целевой переменной – *обучающие данные*. Затем алгоритмы машинного обучения изучают, как входные переменные взаимодействуют и влияют на известную выходную цель. Чтобы точно получить желаемый результат (например, прогноз), алгоритм должен иметь набор «правильных данных», которые не только отражают зависимость выхода от входа, но и достаточно разнообразны, чтобы модель смогла изучить максимально широкий спектр сочетаний переменных на входе.

Таким образом, в обоих случаях требуются качественные входные данные, хорошие алгоритмические подходы и хороший механизм, позволяющий ML как изучать поведение данных, так и применять это обучение для оценки последующих наблюдений за этими данными. Давайте посмотрим, как Elastic ML использует машинное обучение без учителя и с учителем.

Использование машинного обучения без учителя для обнаружения аномалий

Чтобы сформировать у вас более полное представление о том, как работает обнаружение аномалий Elastic ML с использованием машинного обучения без учителя, мы рассмотрим следующие темы:

- строгое определение необычности в контексте технологии;
- наглядный пример обучения без учителя;
- описание того, как технология ML моделирует, устраняет тенденции и оценивает данные.

Что такое необычность?

Обнаружение аномалий – это то, чем мы регулярно занимаемся в повседневной жизни, поэтому имеем интуитивное представление о сути процесса. Люди довольно хорошо работают с визуальной информацией, поэтому неудивительно, что если бы я спросил у 100 человек на улице, что необычного в графике на рис. 1.1, подавляющее большинство (включая далеких от техники людей) указало бы на всплеск линии.

Аналогично мы можем спросить у людей, что необычного на следующей фотографии (рис. 1.2).

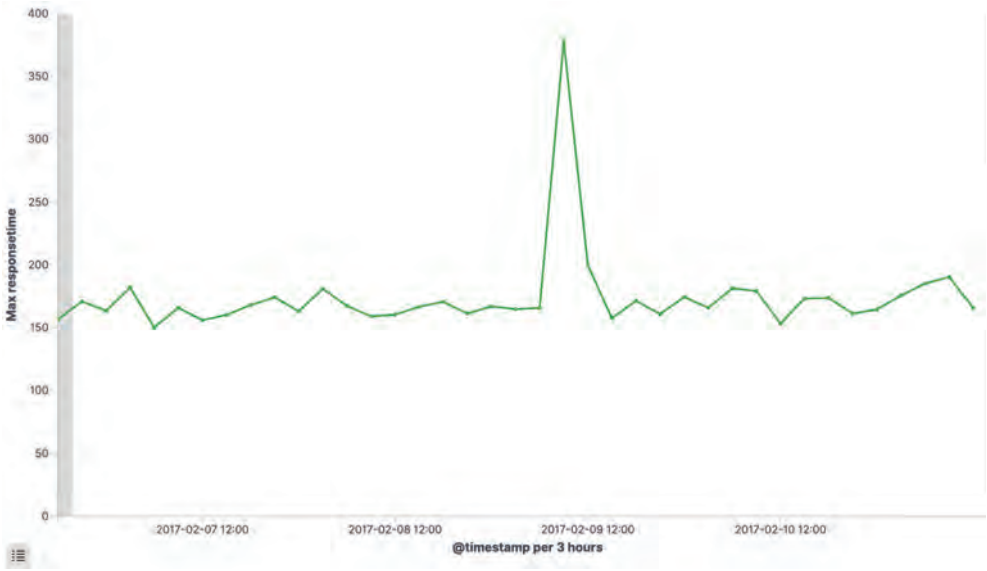

Рис. 1.1 ❖ Этот график содержит визуально заметную аномалию

Рис. 1.2 ❖ На этой фотографии запечатлен тюлень среди пингвинов

Большинство опрошенных наверняка ответят, что тюлень в окружении пингвинов – явление весьма необычное. Но людям бывает сложно сформулировать в явных терминах эвристику, которая лежит в основе таких выводов.

Есть две разные эвристики, которые мы могли бы использовать для определения различных видов аномалий, показанных на этих изображениях:

- сущность необычна, если ее поведение значительно отклоняется от установленного шаблона или диапазона, основанного на исторических данных;