Предисловие

Применение технологии машинного обучения (МО, Machine Learning) в области информационной безопасности, или кибербезопасности, крайне востребовано для специалистов. В частности, инструменты машинного обучения используются для выявления угроз сетевой безопасности и, соответственно, угроз конфиденциальным данным, которые в этих сетях хранятся и передаются.

Необходимо обнаруживать и противостоять сетевым атакам, анализировать и устранять уязвимости, заниматься наполнением базы знаний о киберугрозах, заниматься киберразведкой и т. д. Однако огромный объем данных (логов) и многочисленные задачи не позволяют провести анализ в реальном времени. Решить подобные задачи позволяют технологии МО и искусственного интеллекта, которые хорошо подходят для изучения сетевого трафика, помогают идентифицировать «нормальный» трафик (включая действия пользователей) и отделить его от подозрительного и потенциально опасного.

С недавних пор МО считается одним из ключевых инструментов для обеспечения кибербезопасности. Наиболее перспективной и актуальной в настоящее время технологией является автоматизированное МО, представляющее собой комплекс инструментальных и методических средств, позволяющих значительно сократить долю человеческого участия в создании систем искусственного интеллекта, в том числе средствами автоматической валидации результатов моделирования.

Машинное обучение базируется на трех ключевых элементах.

1. Сбор экспериментального набора данных (Data set, датасет). Это может быть интернет-трафик, сетевые потоки, логи, почтовые сообщения, активность пользователя и многое другое. Чем больше и разнообразнее обучающие данные, тем точнее будет результат предсказания. От качества датасета зависит эффективность МО.

2. Отбор атрибутов (признаков), характеризующих обрабатываемые данные. В зависимости от решаемой задачи атрибутов могут быть сотни. Это могут быть метаданные, ассоциированные с анализируемым файлом: имя, дата создания, размер, наличие сетевых соединений, обращения к реестру и т. д.

3. Выбор существующих или разработка новых алгоритмов и моделей МО. Правильный выбор алгоритма или модели, осуществляющего поиск по определенным признакам искомого в датасете, является компромиссом между скоростью работы алгоритма и его сложностью.

В соответствии с представленной концепцией содержание книги можно условно разбить на три части.

В первой части, включающей первые три главы, вводятся основные понятия интеллектуального анализа данных и машинного обучения. Здесь анализируются методы обнаружения и классификации компьютерных атак и сетевых аномалий. Рассматриваются поведенческие методы, методы, основанные на знаниях, методы вычислительного интеллекта и др. Учитывая направленность дальнейшего изложения, в первой главе анализируются классические методы МО, включая алгоритмы классификации с помощью нейронных сетей и сетей глубокого обучения. Анализируются сети глубокого убеждения (Deep Neural Network, DNN), сверточные нейросети (Convolutional Neural Network, CNN), генеративно-состязательные нейронные сети (Generative Adversarial Network, GAN), рекурсивные нейронные сети (Recurrent Neural Network, RNN), нейронные сети, использующие архитектуру LSTM (Long Short-Term Memory).

Для прогнозирования аномальных событий в компьютерных сетях анализируются методы поиска паттернов в последовательности событий на основе выделения закономерностей и секвенциального анализа. На основе решения задач классификации анализируется структура систем и инструментов обнаружения сетевых атак.

Важное место занимают вопросы классификации и кластеризации МО и метрики оценки эффективности результатов обработки. На основе математической постановки задачи классификации рассматривается широкий ассортимент существующих методов и алгоритмов: линейный классификаторm логистическая регрессия, байесовский классификатор, наивный байесовский классификатор, метод k ближайших соседей (KNN), алгоритмы на основе деревьев решений (CART, C4.5, CHAID, лес решений, случайный лес), ансамблевые алгоритмы. Рассматриваются методы композиции обучающихся алгоритмов бустинг, бэггинг и стекинг.

Важное место в главе 2 занимают вопросы обработки потоковых данных, включая алгоритмы Adaptive Random Forest и алгоритмы обнаружения смены дрейфа концепта.

Рассматриваются наиболее распространенные методы кластеризации, включая иерархические, неиерархические и сетевые методы.

Вторая часть, включающая главу 4, посвящен вопросами изучения, структуры обучающих и тестирующих данных, отбору числа

и состава информативных признаков сетевых атак с использованием программных средств. На примере наборов данных NSL-KDD, CSIC 2010 HTTP, UNSW-NB15, CICIDS 2017 и др. анализируются особенности сбора статистики трафика, предварительной обработки, выделения временных параметров атак.

Третья часть книги (главы 5–7) посвящена примерам использования алгоритмов МО при решении задач сетевой безопасности.

В частности, рассматриваются особенности бинарного и многоклассового обнаружения и классификация сетевых атак методами машинного обучения на примере баз данных NSL-KDD и UNSW-NB15. Рассматривается особенности обнаружения и классификация сетевых атак с помощью алгоритмов IForest, Random Forest, гибридных искусственных нейронных сетей. Анализируются методы расширения состава признаков сетевых атак за счет введения дополнительных параметров. В частности, оценивается влияние фрактальной размерности на качество бинарной классификации сетевых атак. Рассмотрены примеры реализации нечеткой классификации сетевых атак на примере алгоритмов Мамдани и Такаги–Сугено.

В заключительной главе рассматриваются искусственные иммунные системы (ИИС) в информационной безопасности. Рассматриваются базовые принципы, области и подходы применения ИИС в системах информационной безопасности. Анализируются особенности построения ИИС для обнаружения компьютерных атак.

Авторы выражают глубокую признательность Ю.Н. Чернышову за огромный и кропотливый труд по редактированию книги.

1 Классические парадигмы машинного обучения и интеллектуального анализа данных

1.1. Основные понятия. Технологии KDD и Data Mining

По способу решения задачи интеллектуального анализа данных можно разделить на два класса: обучение с учителем (supervised learning) и обучение без учителя (unsupervised learning).

В первом случае требуется обучающий набор данных, на котором создается и обучается модель интеллектуального анализа данных. Готовая модель тестируется и впоследствии используется для предсказания значений в новых наборах данных. Иногда в этом же случае говорят об управляемых алгоритмах интеллектуального анализа.

Во втором случае целью является выявление закономерностей, имеющихся в существующем наборе данных. При этом обучающая выборка не требуется. В качестве примера можно привести задачу анализа потребительской корзины, когда в ходе исследования выявляются товары, чаще всего покупаемые вместе. К этому же классу относится задача кластеризации.

Также можно говорить о категоризации задач интеллектуального анализа данных по назначению, в соответствии с которой они делятся на описательные (descriptive) и предсказательные (predictive). Цель решения описательных задач — лучше понять исследуемые данные, выявить имеющиеся в них закономерности, даже если в других наборах данных они встречаться не будут. Для предсказательных задач характерно то, что в ходе их решения на основании набора данных с известными результатами строится модель для предсказания новых значений.

Можно выделить несколько основных задач интеллектуального анализа данных.

Задача регрессии во многом схожа с задачей категоризации, но в ходе ее решения производится поиск шаблонов для определения числового значения. Иными словами, предсказываемый параметр здесь, как правило, число из непрерывного диапазона. Задачи категоризации и регрессии относятся к типу обучения с учителем.

Задача категоризации заключается в том, что для каждого варианта определяется категория или класс, которому он принадлежит. В качестве примера можно привести оценку кредитоспособности потенциального заемщика: здесь всего два назначаемых класса — «кредитоспособен» и «некредитоспособен». Необходимо отметить, что для решения задачи необходимо, чтобы множество классов было известно заранее и было бы конечным и счетным.

Задача прогнозирования новых значений на основании имеющихся значений числовой последовательности (или нескольких последовательностей, между значениями которых наблюдается корреляция).

Задача кластеризации заключается в делении множества объектов на группы (кластеры), схожие по параметрам. При этом, в отличие от категоризации, число кластеров и их характеристики могут быть заранее неизвестны и определяться в ходе построения кластеров, исходя из степени близости объединяемых объектов по совокупности параметров. Кластеризацию можно назвать сегментацией. Например, Интернет-магазин может быть заинтересован в проведении подобного анализа базы своих клиентов для того, чтобы потом сформировать специальные предложения для выделенных групп, учитывая их особенности. Кластеризация относится к задачам обучения без учителя.

Задача определения взаимосвязей, также называемая задачей поиска ассоциативных правил, заключается в определении часто встречающихся наборов объектов среди множества подобных наборов. Классическим примером является анализ потребительской корзины, который позволяет определить наборы товаров, чаще всего встречающиеся в одном заказе (или в одном чеке). Эта информация может потом использоваться при размещении товаров в торговом зале или при формировании специальных предложений для группы связанных товаров. Данная задача также относится к классу обучения без учителя.

Анализ последовательностей, или секвенциальный анализ, одними авторами рассматривается как вариант предыдущей задачи, другими выделяется отдельно. Целью в данном случае является обнаружение закономерностей в последовательностях событий. Подобная информация позволяет, например, предупредить сбой в работе информационной системы, получив сигнал о наступлении события, часто предшествующего сбою подобного типа. Другой пример применения — анализ последовательности переходов по страницам пользователей web-сайтов.

Анализ отклонений позволяет отыскать среди множества событий те, которые существенно отличаются от нормы. Отклонение

может сигнализировать о каком-то необычном событии (неожиданный результат эксперимента, мошенническая операция по банковской карте, ...) или, например, об ошибке ввода данных оператором.

Для решения подобных задач существует две популярные методологии ведения проектов интеллектуального анализа данных: Knowledge Discovery in Databases (KDD) [1] и Cross Industry Standard Process for Data Mining (CRISP-DM) [2].

Методика извлечения знаний, зародившаяся в 1989 г., получила название Knowledge Discovery in Databases — извлечение знаний из баз данных, и описывает не конкретный алгоритм или математический аппарат, а последовательность действий, которую необходимо выполнить для обнаружения полезного знания. Методика не зависит от предметной области, а представляет собой набор атомарных операций, комбинируя которые, можно получить нужное решение. KDD включает в себя этапы подготовки данных, выбора информативных признаков, очистки, построения моделей, постобработки и интерпретации полученных результатов. Ядром этого процесса являются методы Data Mining, позволяющие обнаруживать закономерности и знания.

Определение. Knowledge Discovery in Databases — процесс получения из данных знаний в виде зависимостей, правил, моделей, состоящий обычно из таких этапов, как выборка данных, их очистка и трансформация, моделирование и интерпретация полученных результатов (рис. 1.1).

Таким образом KDD — это процесс поиска полезных знаний в «сырых» данных.

Выборка данных. Первым шагом в анализе является получение исходной выборки. На основе отобранных данных строятся модели. На этом этапе требуется активное участие экспертов для выдвижения гипотез и отбора факторов, влияющих на анализируемый процесс. Желательно, чтобы данные были уже собраны и консолидированы. Необходимы удобные механизмы подготовки выборки, как правило, это запросы, фильтрация данных и сэмплинг. Чаще всего в качестве источника рекомендуется использовать специализированное хранилище данных, консолидирующее всю необходимую для анализа информацию.

Очистка данных. Реальные данные для анализа редко бывают хорошего качества, поэтому независимо от того, какие технологии и алгоритмы используются, возникает необходимость в предварительной обработке при анализе данных. Эта задача может представлять самостоятельную ценность в областях, не имеющих непосредственного отношения к анализу данных. К задачам очистки

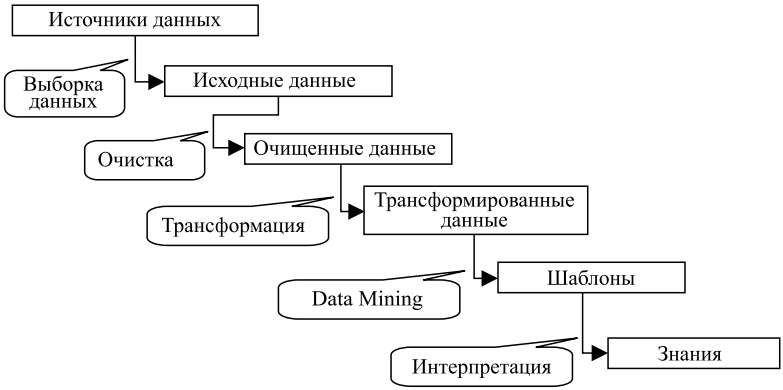

Рис. 1.1. Этапы KDD

данных относятся: заполнение пропусков, подавление аномальных значений, сглаживание, исключение дубликатов и противоречий и пр.

Трансформация данных. Этот шаг необходим для тех методов, при использовании которых исходные данные должны быть представлены в каком-то определенном виде. Дело в том, что различные алгоритмы анализа требуют специальным образом подготовленных данных. Например, для прогнозирования необходимо преобразовать временной ряд при помощи скользящего окна или вычислить агрегированные показатели. К задачам трансформации данных относятся: скользящее окно, выделение временных интервалов, квантование, сортировка, группировка и пр.

Data Mining. На этом этапе строятся модели.

Термин *Data Mining* дословно переводится как «добыча данных» или «раскопка данных» и имеет в англоязычной среде несколько определений.

Определение. Data Mining — обнаружение в «сырых» данных ранее неизвестных, нетривиальных, практически полезных и доступных интерпретации знаний, необходимых для принятия решений в различных сферах человеческой деятельности.

Data Mining — это не один метод, а совокупность большого числа различных методов обнаружения знаний [3, 4]. Существует несколько условных классификаций задач Data Mining. Как правило, говорят о четырех базовых классах задач.

1. *Классификация* — установление зависимости дискретной выходной переменной от входных переменных.

2. *Регрессия* — установление зависимости непрерывной выходной переменной от входных переменных.

3. *Кластеризация* — группировка объектов (наблюдений, событий) на основе данных, описывающих свойства объектов. Объекты внутри кластера должны быть похожими друг на друга и отличаться от других объектов, которые вошли в другие кластеры.

4. *Ассоциация* — выявление закономерностей между связанными событиями. Примером такой закономерности служит правило, указывающее, что из события X следует событие Y. Такие правила называются ассоциативными. Если же интерес представляет последовательность происходящих событий, то можно говорить о последовательных шаблонах — установлении закономерностей между связанными во времени событиями. Примером такой закономерности служит правило, указывающее, что из события X спустя время t последует событие Y.

Интерпретация. В случае, когда извлеченные зависимости и шаблоны непрозрачны для пользователя, должны существовать методы постобработки, позволяющие привести их к интерпретируемому виду. Для оценки качества полученной модели нужно использовать как формальные методы, так и знания аналитика. Именно аналитик может сказать, насколько применима полученная модель к реальным данным. Построенные модели являются по сути формализованными знаниями эксперта, а следовательно, их можно тиражировать. Найденные знания должны быть применимы и к новым данным с некоторой степенью достоверности

Кроме перечисленных задач, часто выделяют анализ отклонений (deviation detection), анализ связей (link analysis), отбор значимых признаков (feature selection), хотя эти задачи граничат с очисткой и визуализацией данных.

Задача классификации отличается от задачи регрессии тем, что в классификации на выходе присутствует переменная дискретного вида, называемая меткой класса. Решение задачи классификации сводится к определению класса объекта по его признакам, при этом множество классов, к которым может быть отнесен объект, известно заранее.

В **задаче регрессии** выходная переменная является непрерывной — множеством действительных чисел, пример — прогнозирование временного ряда на основе исторических данных.

Кластеризация отличается от классификации тем, что выходная переменная не требуется, а число кластеров, в которые необходимо сгруппировать все множество данных, может быть неизвестным. Выходом кластеризации является не готовый ответ (например, плохо/удовлетворительно/хорошо), а группы похожих объектов — кластеров. Кластеризация указывает только на схожесть объектов