

ми для действий. У данного фреймворка также есть встроенные функции, называемые трассировками (Traces), которые могут делать автоматические снимки экрана, видеозапись теста, повторную попытку теста и механизм ведения журнала. Playwright предоставляет инструмент инспектора, который помогает отслеживать и отлаживать каждый шаг исполнения, видеть точки кликов и проверять локаторы страниц на лету (во время выполнения теста) [4].

Playwright предоставляет также функцию контекста браузера, которая помогает сохранять и передавать состояние браузера в любой другой новый браузер. Имеется возможность воспользоваться функцией генерации кода, которая записывает все шаги в веб-приложении и далее преобразует в написанные тесты на любом поддерживаемом языке программирования [4].

Сегодня фреймворк Playwright для автоматизированного тестирования от Microsoft набирает популярность и активно развивается. Благодаря всем вышеперечисленным основным функциям и возможностям, Playwright является основным конкурентом других инструментов автоматизации, например, Selenium и Cypress.

Очевидно, что выбор технологии Playwright для основы фреймворка автоматизации тестирования является обоснованным. Используя набор выбранных технологий планируется создать эффективный универсальный фреймворк для автоматизированного кросс-браузерного тестирования веб-сайтов. Готовых решений не существует, т.к. каждый программный продукт индивидуален. Поэтому для успешного создания фреймворка автоматизированного тестирования необходимо учитывать логику работы, набор API веб-сервисов, а также реализацию внешнего интерфейса, что позволит ускорить процесс непрерывного тестирования веб-приложения.

Список использованных источников

1. Тестирование программного обеспечения. Базовый курс / С. С. Куликов. – 3-е изд. – Минск: Четыре четверти, 2020. – 312 с..
2. Playwright – драматургия от Microsoft и новый инструмент для тестирования [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/company/jugru/blog/487294> – Дата доступа: 15.10.2022.
3. Playwright [Электронный ресурс]. – Режим доступа: <https://playwright.dev> – Дата доступа: 15.10.2022.
4. Playwright JS Automation Testing from Scratch with Framework [Электронный ресурс]. – Режим доступа: <https://www.classcentral.com/course/udemy-playwright-tutorials-automation-testing-98095> – Дата доступа: 15.10.2022.

УДК 655.1

ОБНАРУЖЕНИЕ АНОМАЛИЙ В ДАННЫХ ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА

А.Л. Михняев, А.В. Пархоць, И.В. Абоимов, Д.А. Дворанинович
Брестский государственный технический университет, Брест, Беларусь,
bstucon1@bstu.by

The main task of this work is the detection of anomalies for their further analysis, classification and possible ways of elimination.

Временные ряды повсюду: в поведении пользователя на веб-сайте, в ценах на акции компании или в любом другом примере, связанном со временем. Данные временных рядов очевидны в каждой отрасли в той или иной форме. Естественно, это также один из наиболее исследуемых типов данных. На практике можно сказать, что временные ряды – это тип данных, выборка которых основана на некотором временном измерении, таком как годы, месяцы или секунды.

Временные ряды – это наблюдения, которые были записаны упорядоченным образом и коррелированы во времени. Анализируя данные временных рядов, мы должны следить за выбросами, как мы это делаем со статическими данными. Если вы работали с данными в любом качестве, вы знаете, как много боли причиняют аналитику выбросы. Эти выбросы называются «аномалиями». С традиционной точки зрения выброс/аномалия – это «наблюдение, которое настолько отличается от других наблюдений, что вызывает подозрение, что оно было произведено другим механизмом».

В последние годы, особенно в области данных временных рядов, многие исследователи стремились обнаруживать и анализировать необычные, но интересные явления. Хорошим примером является обнаружение мошенничества — основная цель состоит в том, чтобы обнаружить и проанализировать сам выброс. Проблема обнаружения аномалий для временных рядов обычно формулируется как идентификация точек данных с выбросами по отношению к некоторой норме или обычному сигналу.

Выбросы бывают: точечные и последующие. Точечный выброс – это данные, которые ведут себя необычно в определенный момент времени по сравнению либо с другими значениями временного ряда (глобальный выброс), либо с соседними точками (локальный выброс). Точечные выбросы могут быть одномерными или многомерными, в зависимости от того, влияют ли они на одну или несколько переменных, зависящих от времени, соответственно. Последующий выброс – это последовательные моменты времени, совместное поведение которых является необычным, хотя каждое наблюдение в отдельности не обязательно является точкой выброса. Выбросы также могут быть глобальными или локальными.

Есть несколько методов, которые аналитики могут использовать для выявления различных аномалий в данных. Он начинается с базовой статистической декомпозиции и может работать до автоэнкодеров. В нашей работе обнаружение аномалий было проведено с помощью автоэнкодеров. Поэтому рассмотрим данную область более детально.

Автоэнкодеры — это неконтролируемый метод, который воссоздает входные данные, извлекая их функции в разных измерениях. Другими словами, если мы используем скрытое представление данных из автоэнкодеров, это соответствует уменьшению размерности.

Почему мы применяем уменьшение размерности, чтобы найти выбросы? Не потеряем ли мы часть информации, включая выбросы, если уменьшим размерность? Ответ заключается в том, что после выявления основных закономерностей выявляются выбросы. Многие основанные на расстоянии методы страдают от размерности, когда они вычисляют расстояния до каждой точки данных в полном пространстве признаков. Высокая размерность должна быть уменьше-

на. Интересно, что в процессе уменьшения размерности выявляются выбросы. Можно сказать, что обнаружение выбросов является побочным продуктом уменьшения размерности.

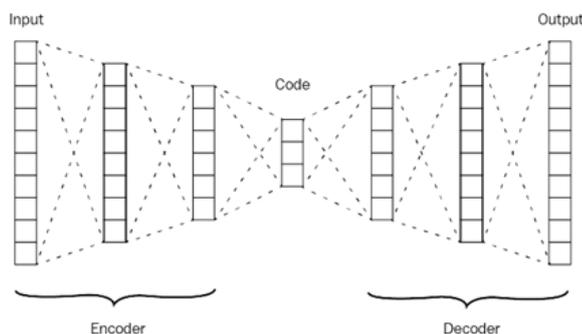


Рисунок 1 – Составные части автоэнкодера

Работа автоэнкодера состоит в том, чтобы получить некоторые входные данные, передать их через модель и получить реконструкцию входных данных. Реконструкция должна максимально соответствовать входным данным. Хитрость заключается в использовании небольшого количества параметров, поэтому ваша модель изучает сжатое представление данных. В некотором смысле автоэнкодеры пытаются изучить только самые важные функции (сжатую версию) данных.

Почему автоэнкодеры? Причина в том, что методы автоэнкодера могут выполнять нелинейные преобразования с их нелинейной функцией активации и несколькими уровнями. Более эффективно обучать несколько слоев с помощью автоэнкодера, чем обучать одно большое преобразование с помощью PCA. Таким образом, методы автокодировщика показывают свои достоинства, когда проблемы с данными сложны и нелинейны по своей природе.

Плюсы:

- Автоэнкодеры могут легко обрабатывать многомерные данные.
- Что касается его нелинейного поведения, он может находить сложные закономерности в многомерных наборах данных.

Минусы:

- Поскольку это стратегия, основанная на глубоком обучении, ей будет особенно трудно, если данных меньше.
- Затраты на вычисления резко возрастут, если увеличится глубина сети и при работе с большими данными.

Заключение. Выбросы не часто обсуждаются при тестировании, но, в зависимости от бизнеса и метрики, которые оптимизируются, они могут повлиять на результаты. Одно или два высоких значения при небольшом размере выборки могут полностью исказить тест, что приведет к принятию решения на основе ошибочных данных.

Другой способ, возможно, лучший в долгосрочной перспективе, — экспортировать данные после тестирования и визуализировать их различными способами. Определите влияние выбросов в каждом конкретном случае.

На самом деле существует множество способов справиться с выбросами в данных. Не существует быстрого решения, которое работало бы повсеместно, поэтому спрос на хороших аналитиков продолжает расти.

Список использованных источников

1. Современные проблемы математики и вычислительной техники. – Брест : БрГТУ, 2003. – 298 с.
2. Neptune.ai [Электронный ресурс]. – Режим доступа: <https://neptune.ai/blog/anomaly-detection-in-time-series>. – Дата доступа: 30.10.2022.
3. Головки, В. А. Нейросетевые технологии обработки данных : учебное пособие / В. А. Головки, В. В. Краснопрошин. – Минск : Белорусский государственный университет, 2017. – 263 с. – (Классическое университетское издание). – ISBN 978-985-566-467-4. – EDN GLVGIE.
4. Головки В. А. Нейросетевые методы обработки хаотических процессов //VII Всероссийская научно-техническая конференция «Нейроинформатика. – 2005. – С. 43-91.

UDC 004.942

A COMPARISON OF THE COVID-19 MACHINE LEARNING AUTOMATION MODEL AND SPSS TIME SERIES

Hongxu Zhu, D.O. Petrov, V.S.Razumeichik

Brest State Technical University, Brest, Belarus, zhuhongxu08@gmail.com

This paper uses publicly available data on the prediction process of Covid-19 transmission in the world to attempt to predict the time series using the SPSS exponential Holt model and the Python ARIMA model. model model to predict the epidemic development trend and key nodes, quantitative analysis of the scale of the epidemic, scientific and reliable interval estimation of the original base and effective transmission rate of the epidemic and comparative analysis of different algorithms, providing an effective basis and guide for analysis, command and decision making in the prevention and control of the epidemic.

Predicting data at a range of points in time is a common activity in real life, and research fields such as agriculture, business, climate, military and medicine all contain large amounts of time series data. Time series forecasting refers to making predictions about the likely future values of a series based on the historical data of the series, as well as other relevant series that may have an impact on the outcome. There are many real-life time series forecasting problems, including voice analysis, noise cancellation and analysis of stock and futures markets, where the essence is to derive the value of the time series at $T + 1$ based on observations at the previous T moments. For time series prediction, we can use the traditional ARIMA model, or we can use the Holt model or other models based on time series. Nowadays, machine learning methods such as deep learning can also be used for time series prediction. We are going to introduce how to implement the covid-19 prediction of time series based on two different models.

In this paper, two types of time-series data software were used for fitting:

- Holt Model (SPSS);
- ARIMA Model (Python3.7).

There were 187,801 training samples prepared, and the sample data was split into a training set and a test set with 67 test indicators. The actual training data is in 3 columns (841 items filtered according to the test specified mediation and 822 items filtered for missing values):