УДК 618.12

М.С. Вараниця

АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ВИЯВЛЕННЯ АНОМАЛІЙ У ДАНИХ ІЗ ЗАСТОСУВАННЯМ МОВИ ПРОГРАМУВАННЯ PYTHON

Вінницький національний технічний університет

Abstract

The paper examines the requirements analysis for software solutions implementing statistical methods for anomaly detection in data using the Python programming language. It explores approaches to requirements elicitation, specification of data processing needs, and algorithms for automated anomaly detection, focusing on both functional and non-functional requirements.

Keywords: Requirements analysis, software requirements, anomalies, statistical methods, Python, machine learning, data analysis, anomaly detection

Анотація

У роботі розглядається аналіз вимог до програмного забезпечення для реалізації статистичних методів виявлення аномалій у даних із використанням мови програмування Python. Досліджуються підходи до збору та специфікації вимог, обробки даних, аналізу статистичних відхилень, з фокусом на функціональні та нефункціональні вимоги, що забезпечують високу якість програмного продукту.

Ключові слова: Аналіз вимог, програмні вимоги, аномалії, статистичні методи, Python, машинне навчання, аналіз даних, виявлення аномалій

Вступ

Виявлення аномалій у даних є важливим завданням у сфері аналізу даних, машинного навчання та статистики. Аномалії можуть свідчити про наявність помилок, шахрайських операцій або технічних несправностей. Розробка ефективних методів для виявлення таких відхилень потребує ретельного аналізу вимог до відповідного програмного забезпечення.

Статистичні методи відіграють ключову роль у процесі виявлення аномалій, оскільки дозволяють оцінювати відхилення від нормального розподілу даних. У поєднанні з можливостями мови програмування Python, яка має широкий набір бібліотек для аналізу даних (numpy, pandas, scipy, scikit-learn), ці методи можуть бути ефективно реалізовані та застосовані у різних сферах.

Ця робота присвячена дослідженню процесу аналізу вимог до програмного забезпечення для виявлення аномалій та їх практичному застосуванню в середовищі Python. Особлива увага приділяється визначенню функціональних і нефункціональних вимог, а також зв'язку з ефективністю алгоритмів та можливостями автоматизації процесу обробки даних.

Функціональні вимоги до програмного забезпечення для виявлення аномалій

Функціональні вимоги до програмного забезпечення для виявлення аномалій охоплюють кілька ключових аспектів. Система повинна підтримувати імпорт даних з різних джерел та здійснювати їх попередню обробку. Це включає автоматичне опрацювання пропущених значень, нормалізацію та стандартизацію даних, а також виявлення та обробку викидів на етапі підготовки.

Програмне забезпечення повинно реалізовувати різноманітні статистичні методи виявлення аномалій: методи на основі статистичних показників (Z-score), методи на основі щільності розподілу (LOF, KDE), методи на основі відстані (кластеризація, k-NN), а також методи машинного навчання (ізоляційний ліс, автокодери) [1].

Важливою вимогою є можливість візуалізації та інтерпретації результатів. Програмне забезпечення має забезпечувати візуальне представлення виявлених аномалій та надавати інтерактивні інструменти для глибшого аналізу даних. Користувачі повинні мати можливість налаштовувати пороги для визначення аномалій відповідно до специфіки їхніх даних.

Програмне забезпечення має забезпечувати інтеграцію з іншими системами через API та працювати з великими наборами даних. Автоматизація процесів виявлення аномалій, включаючи планування регулярних перевірок та оповіщення, є важливою для систем моніторингу в реальному часі.

Нефункціональні вимоги до систем виявлення аномалій

Нефункціональні вимоги визначають якісні характеристики системи. Продуктивність програмного забезпечення повинна забезпечувати обробку даних у визначених часових межах, особливо для систем реального часу. Алгоритми мають бути оптимізовані для ефективного використання ресурсів та роботи з великими даними.

Надійність системи передбачає стабільну роботу з різними типами даних та належну обробку виключних ситуацій. Збереження цілісності даних протягом усього процесу є критично важливим, особливо для систем, що працюють з конфіденційною інформацією.

Зручність використання є важливим аспектом. Інтерфейс має бути інтуїтивно зрозумілим навіть для користувачів без глибоких знань у статистиці. Система повинна надавати документацію та контекстні підказки, а також можливість налаштування параметрів алгоритмів без програмування [2].

Безпека є критичною вимогою для систем, що працюють з чутливими даними. Програмне забезпечення повинно забезпечувати контроль доступу, аудит дій користувачів та шифрування даних для захисту від несанкціонованого доступу.

Методологія аналізу вимог для систем виявлення аномалій

Процес аналізу вимог для програмного забезпечення виявлення аномалій вимагає структурованого підходу. Збір вимог починається з інтерв'ю з експертами предметної області та аналізу існуючих систем. Дослідження потреб користувачів та визначення бізнес-цілей допомагає узгодити функціональність системи з стратегічними цілями організації.

Документування вимог включає створення специфікації, розробку прототипів інтерфейсу та визначення метрик для оцінки якості виявлення аномалій. Валідація та верифікація вимог забезпечує їх відповідність реальним потребам та технічну реалізованість. Керування змінами вимог є важливим аспектом, особливо для довготривалих проектів, і включає відстеження змін, оцінку їх впливу та пріоритизацію [3].

Перспективи розвитку

Перспективи розвитку пов'язані з удосконаленням як процесів збору вимог, так і алгоритмів для їх реалізації. Використання нейромережевих підходів для виявлення аномалій, таких як автокодери та генеративні змагальні мережі, демонструє високий потенціал. Під час аналізу вимог необхідно враховувати особливості цих технологій та їх вплив на архітектуру системи [4].

Розробка гнучких вимог, що дозволяють системі адаптуватися до змін у даних, є важливим напрямком. Це включає визначення вимог до самонавчання системи та автоматичного налаштування параметрів. Сучасні методи візуалізації даних та інтерактивні інструменти аналізу стають все важливішими для ефективної роботи з аномаліями.

Автоматизовані системи моніторингу в реальному часі та вдосконалення методів попередньої обробки даних є перспективними напрямками. Впровадження оновлюваних моделей, здатних адаптуватися до змін без ручного налаштування, особливо важливе для систем, що працюють з даними, характеристики яких змінюються з часом [5].

Висновки

Якісний аналіз вимог до програмного забезпечення для виявлення аномалій є критично важливим етапом розробки. Визначення чітких функціональних та нефункціональних вимог забезпечує основу для ефективної реалізації статистичних методів та їх інтеграції в бізнес-процеси.

Застосування структурованого підходу до збору та валідації вимог дозволяє створювати програмні рішення, які точно відповідають потребам користувачів. Поєднання методів інженерії вимог з розумінням статистичних підходів забезпечує розробку ефективних та надійних систем.

Статистичні методи у поєднанні з алгоритмами машинного навчання дозволяють підвищити ефективність виявлення аномалій. Подальший розвиток передбачає вдосконалення підходів до аналізу вимог, впровадження нейронних мереж та створення інтегрованих рішень для моніторингу в реальному часі.

Таким чином, ретельний аналіз вимог до програмного забезпечення для виявлення аномалій з використанням Python відкриває широкі можливості для вдосконалення аналізу даних та розвитку автоматизованих систем детекції аномальних ситуацій.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Aggarwal C. C. Outlier Analysis / C. C. Aggarwal. – 2nd ed. – Cham: Springer, 2017. – 492 p.
2. Chandola V. Anomaly Detection: A Survey / V. Chandola, A. Banerjee, V. Kumar // ACM Computing Surveys. – 2009. – Vol. 41, No. 3. – P. 1–58.
3. Raschka S. Python Machine Learning / S. Raschka, V. Mirjalili. – 3rd ed. – Birmingham: Packt Publishing, 2019. – 770 p.
4. Sommerville I. Software Engineering / I. Sommerville. – 10th ed. – Pearson, 2016. – 816 p.
5. Wiegers K. Software Requirements / K. Wiegers, J. Beatty. – 3rd ed. – Microsoft Press, 2013. – 672 p.

Varaanytsia Mykola **–** student of group 4PE-21b, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: pro100spaderket@gmail.com

Lishchynska Lyudmyla B. **–** professor of the Department of Computer Science, Vinnytsia National Technical University, Vinnytsia

Вараниця Микола Сергійович – студент групи 4ПІ-21б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: pro100spaderket@gmail.com

Ліщинська Людмила Броніславівна. – професор кафедри ПЗ, Вінницький національний технічний університет, м. Вінниця