УДК 004.85

ПОРІВНЯЛЬНИЙ АНАЛІЗ МЕТОДІВ ВИЯВЛЕННЯ АНОМАЛІЙ У ЛОГАХ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Чубаров Є.Е.
e-mail: yevhen.chubarov@nure.ua
Науковий керівник – к.т.н., доц. Русакова Н.Є.
Харківський національний університет радіоелектроніки, каф. ПІ,
м. Харків, Україна

This study evaluates classical (PCA, Isolation Forest) and deep learning (DeepLog, LogAnomaly, LogBERT) models for log anomaly detection using HDFS and REST API datasets. Results show that model effectiveness strictly depends on the anomaly type. Classical models efficiently identify volumetric deviations, such as frequency and latency, via statistical outliers. Conversely, deep learning architectures, notably LogBERT, excel at detecting complex semantic and sequential logic violations by capturing system context. However, high computational costs limit neural networks in real-time monitoring, advocating for a hybrid approach.

В умовах постійно зростаючої складності програмних систем з’являється необхідність застосування засобів машинного навчання для аналізу логів, які є основним джерелом інформації про стан системи.

У попередньому теоретичному дослідженні [1] було проведено порівняння методів виявлення аномалій у логах ПЗ та висунуто припущення, що ефективність класичних моделей машинного навчання та моделей глибокого навчання суттєво залежить від природи логів. Метою даної роботи є перевірка цих теоретичних припущень шляхом проведення експериментів на датасетах різної природи: інфраструктурних логах та логах прикладного рівня.

Для досягнення поставленої мети на основі попереднього теоретичного дослідження було обрано п’ять алгоритмів навчання без вчителя: класичні методи PCA та Isolation Forest, моделі глибинного навчання на основі рекурентних нейронних мереж (DeepLog, LogAnomaly) та трансформерну модель LogBERT [2].

Експериментальне дослідження проводилося на двох наборах даних, що відрізняються за своєю структурою та природою відхилень. Перший – класичний еталонний датасет інфраструктурних логів файлової системи HDFS. Цей датасет містить понад 11 млн рядків логів, згрупованих у 575 тис. блоків, з яких 16 838 є аномальними. Аномалії мають інфраструктурний характер (апаратні збої, мережеві помилки), що дозволяє представляти логи у вигляді матриці частот подій.

Другий набір – синтетичний датасет логів прикладного рівня, згенерований на власному стенді, який являє собою Web API, побудоване на базі ASP.NET Core, що імітує роботу платформи електронної комерції.

API є горизонтально масштабованим із балансуванням трафіку через Nginx, а навантаження генерувалося Python-скриптом. На відміну від HDFS, даний датасет містить інформацію про поведінку користувачів, що відрізняється непередбачуваністю. Розмір датасету – 295 123 рядки, з яких 2386 рядків є аномаліями різних видів, тобто відсоток аномалій складає ~0,81%, що відповідає реальним середовищам, де кількість аномальних подій у порівнянні з нормальними дуже мала. Для перевірки моделей у нормальний веб-трафік було включено п'ять типів аномалій: точкові, частотні, часові, семантичні та послідовні.

Для оцінки ефективності моделей у роботі використано стандартні метрики класифікації: точність (Precision), повноту (Recall) та F1-Score.

Аналіз результатів на класичному датасеті HDFS підтвердив, що за умови коректної агрегації подій за ідентифікаторами блоків, класичні статистичні методи демонструють дуже високу роздільну здатність. Зокрема, модель PCA досягла показників F1-Score = 0.985 та Recall = 1.000. Це також збігається з висновками попередніх досліджень [3].

Така ефективність алгоритму PCA прямо випливає з природи моделей та особливостей датасету. У цьому датасеті 12 із 29 типів системних подій зустрічаються лише в аномальних блоках. Нормальні лог-сесії формують стабільний базис головних компонент, тоді як поява маркерів помилок або значних відхилень створює вектор, відмінний від нормального простору, що фіксується через зростання помилки реконструкції.

Натомість алгоритм Isolation Forest, який спирається на випадкове розбиття ознак для ізоляції точок, у багатовимірному та розрідженому просторі лічильників HDFS втрачає здатність чітко відокремлювати аномалії. При високому показнику повноти (recall 0.998), модель IF показала вкрай низьку точність (precision) 0.106, що призводить до критично великої кількості хибних спрацьовувань (false positives) та падіння підсумкової метрики F1-Score до рівня 0.191.

Попри високу ефективність PCA на датасеті HDFS, було також досліджено застосування моделей глибинного навчання. Експерименти показали різну здатність моделей виявляти аномалії. Моделі DeepLog та LogAnomaly продемонстрували нижчу ефективність (F1-Score 0.3833 та 0.5783), головним чином через низьку повноту, що призвело до пропуску значної частини аномальних блоків. Найкращі результати показала модель LogBERT (F1-Score 0.7923, precision 0.8048, recall 0.7802), що пояснюється використанням механізму self-attention для виявлення залежностей у логах.

Перехід до аналізу логів прикладного рівня, згенерованих на базі e-commerce REST API, виявив ситуації, в яких класичні методи майже або взагалі не застосовні. На відміну від датасету HDFS, де аномалії часто супроводжуються явними системними помилками, прикладні відхилення

(зокрема семантичні та послідовні) складаються з цілком нормальних подій, що не створюють викидів з точки зору статистики.

Загальні результати тестування класичних моделей підтверджують це фундаментальне обмеження. Найкращі показники серед перевірених підходів продемонструвала модель PCA з показником F1 0.5708, точності 0.6227 та повноти 0.5269. В свою чергу, Isolation Forest показав гірші показники: F1 0.4571, точність 0.5333, повнота 0.4.

Детальний аналіз типів аномалій показав, що класичні моделі ефективно виявляють кількісні відхилення. Для частотних та latency-аномалій PCA досяг повноти 1.000 та 0.78 відповідно. В той же час F1-Score становив 0.061 для семантичних, 0.117 для точкових відхилень. Послідовні аномалії виявлялися лише частково (F1 0.3763 для PCA) – це пояснюється природою алгоритмів і втратами інформації під час препроцесингу.

Моделі глибинного навчання показали вищу загальну ефективність. Найкращою стала LogBERT (F1 = 0.6929, precision = 0.9766, recall = 0.5370). DeepLog та LogAnomaly продемонстрували близькі результати (F1 = 0.6766 та 0.6709 відповідно) і досягли точності 1.0, що практично усуває проблему хибнопозитивних спрацювань.

Аналіз за типами аномалій показав, що нейромережі найкраще виявляють відхилення, які класичні ML-моделі «не бачать». Семантичні та точкові аномалії ідентифікувалися майже безпомилково (F1 та recall 0.95–1.0). Модель LogBERT також покращила виявлення послідовних збоїв (F1 = 0.6667, recall = 0.5667) завдяки здатності аналізувати логи як послідовності подій. Водночас найгірші результати отримано для аномалій затримок (recall 0.23–0.24, F1 $\leq$ 0.3944), що пов’язано з відсутністю числових характеристик у вхідних даних. Недоліком нейромереж є обчислювальна вартість: DeepLog працює 192 с проти ≈1 с для класичних моделей.

Отже, експерименти показали, що класичні та глибинні моделі мають різні переваги, тому найбільш ефективним підходом є їх комбінування.

Список використаних джерел:

1. Русакова Н., Чубаров Є. Порівняльний аналіз методів виявлення аномалій у логах програмного забезпечення. Progressive Approaches in Science and Engineering: Collection of Scientific Papers with Proceedings of the 2nd International Scientific and Practical Conference, 26 листоп. 2025 р. 2025. С. 139–143. DOI: https://doi.org/10.70286/isu-06.08.2025.001

2. Guo H., Yuan S., Wu X. LogBERT: Log Anomaly Detection via BERT. 2021 International Joint Conference on Neural Networks (IJCNN). 2021. С. 1–8. DOI: https://doi.org/10.1109/IJCNN52387.2021.9534113

3. Detecting large-scale system problems by mining console logs / W. Xu та ін. Proceedings of the ACM SIGOPS 22nd Symposium on Operating Systems Principles. New York, NY, USA, 2009. С. 117–132. DOI: https://doi.org/10.1145/1629575.1629587