



О. О. Гудь, Н. Е. Кунанець

Національний університет "Львівська політехніка", м. Львів, Україна

ПРОГНОЗУВАННЯ ІНЦИДЕНТІВ ПІД ЧАС ПЛАНУВАННЯ СПРИНТІВ У ІТ-ПРОЄКТАХ

Зосереджено увагу на обґрунтуванні доцільності застосування технології машинного навчання для підвищення ефективності планування процесів, виконання яких передбачено в ітерації (Sprints) ІТ-проекту, що реалізують з використанням методології Scrum. Розглянуто проблеми, які виникають під час планування задач такого проекту. Проаналізовано причини некоректного планування та шляхи можливого вирішення проблеми. Виокремлено проблему управління незапланованими у проекті процесами та визначено вплив їх появи на коректність планування ітерацій. Проведено аналіз доцільності використання технологій машинного навчання для прогнозування кількості незапланованих завдань впродовж майбутніх ітерацій та запропоновано ці завдання трактувати як інциденти (апаратні збої). Визначено чинники, які впливають на виникнення незапланованих процесів роботи у трьох сегментах: історичні показники кількості інцидентів, апаратне забезпечення та дані мережевого навантаження. Обрано засіб прогнозування – регресор екстремального градієнтного підсилення та за допомогою нього проведено прогнозування ймовірності появи незапланованих процесів роботи. Розглянуто основні принципи роботи алгоритму. Описано переваги застосування цього методу в контексті досліджуваного середовища. Висвітлено особливості процедури порівняльного аналізу моделей регресії. Продемонстровано вплив підбору даних ознак на результат процесу прогнозування та візуалізовано результати застосування методу. Обґрунтовано вибір робочої моделі регресії та представлено результати прогнозування. Описано практичне завдання для аналізу ефективності застосування досліджуваного підходу. Сформовано контрольну та експериментальну команди для дослідження. Наведено приклад використання результатів прогнозування під час планування процесів роботи у ітерації. Проведено порівняльний аналіз підходів до планування ітерацій з урахуванням результатів прогнозування та без них прогнозування. Відображено результати аналізу та оцінено вплив прогнозування на процес прийняття рішень. Доведено ефективність застосування методу регресії екстремального градієнтного підсилення до планування процесів роботи ітерацій проекту, що реалізують з використанням методології Scrum. Наведено перспективи розвитку подальших напрямів дослідження, галузі застосування отриманих результатів.

Ключові слова: управління проєктами; машинне навчання; екстремальний градієнтно підсилений регресор; Scrum.

Вступ / Introduction

Одним із найпопулярніших підходів управління ІТ-проектами є Scrum, в основі якого закладено принципи методології Agile. Використання Scrum для реалізації проєктів із розроблення програмних продуктів дає змогу командам ефективно працювати над їх створенням та швидко вносити зміни, які враховують оновлені вимоги замовника. Одним з основних принципів підходу Scrum є розподіл проєкту на ітерації (Sprints): проєкт поділяють на короткі проміжки часу (ітерації), які зазвичай тривають від одного до чотирьох тижнів. Кожна ітерація має конкретні цілі та дає змогу досягти розроблення придатного до представлення замовнику результату. Перед кожною ітерацією відбувається процес планування завдань для наступної ітерації. Процес планування ітерацій є ключовим елементом підходу Scrum в ІТ-проектах. Його основна мета – організація та оптимізація роботи команди впродовж короткого часового проміжку (ітерації), щоб досягти конкретних цілей і проаналізувати функціонал отриманого продукту. Під

час планування ітерацій визначають завдання, обирають пріоритети, визначають обсяг ресурсів, необхідних для виконання сформованих завдань, а також забезпечують взаємодію та розподіл відповідальності (ролей) в команді. Ця процедура дає змогу забезпечити гнучкість розроблення, швидко реагувати на зміни у вимогах та ефективно досягнення поставлених завдань, забезпечуючи при цьому високу якість програмного продукту. Основними перевагами ітераційного підходу є прозорість, постійний контроль та здатність команди адаптуватися до змін вимог під час реалізації проєкту.

Важливим етапом процесу планування є потреба оцінювати наявні людські ресурси та розподіляти роботу так, щоб сприяти найефективнішому виконанню поставлених завдань. Враховуючи навантаження на кожного члена команди, менеджер проєкту та команда розробників оптимізують розподіл обов'язків (ролей), забезпечуючи баланс між освоєнням нових компетенцій, вирішенням завдань та технічною підтримкою. Окрім цього, розподіл ролей дає змогу забезпечити

Інформація про авторів:

Гудь Олекс Олександрович, аспірант, кафедра інформаційних систем та мереж.

Email: oleksa.hud@gmail.com; <https://orcid.org/0009-0007-4298-7332>

Кунанець Наталія Едуардівна, д-р наук із соц. комунікацій, професор, кафедра інформаційних систем та мереж.

Email: nek.lviv@gmail.com; <https://orcid.org/0000-0003-3007-2462>

Цитування за ДСТУ: Гудь О. О., Кунанець Н. Е. Прогнозування інцидентів під час планування спринтів у ІТ-проектах. Науковий вісник НЛТУ України. 2024, т. 34, № 2. С. 78–86.

Citation APA: Hud, O. O., & Kunanets, N. E. (2024). Incident prediction in sprint planning in IT projects. *Scientific Bulletin of UNFU*, 34(2), 78–86. <https://doi.org/10.36930/40340210>

учасників процесів реалізації ітерації проєкту завданнями, які відповідають їхнім компетентностям. Такий підхід підсилює ефективність роботи команди, зменшуючи час на виконання завдань та забезпечуючи високий рівень продуктивності їх виконання в кожній ітерації.

Однак, доволі поширеною проблемою під час розрахунків обсягів необхідних ресурсів та оцінювання складності завдань є нехтування ймовірністю появи незапланованих процесів роботи. Такими вважають несподівані та додаткові процеси, що виникають у ході розроблення програмного забезпечення і виходять за межі передбачених планом та стратегією. Це може містити потребу вирішення технічних проблем, виявлених під час реалізації проєкту, впровадження вимог замовника чи неочікуваних змін, а також інші внутрішні чи зовнішні чинники, що вимагають додаткових ресурсів для успішного досягнення мети. У разі нехтування цим складником процесу розроблення програмного продукту зазнає зниження відсоток точності планування, адже ризики можуть виникнути на будь-якому етапі життєвого циклу програмного продукту і впливати на бюджет, якість та терміни виконання заздалегідь розроблених планів.

Причиною нехтування важливих чинників під час планування зазвичай є складність прогнозування ймовірності появи незапланованих процесів та недоцільність проведення відповідних розрахунків вручну заради виявлення порівняно нетривалих проміжків часу, необхідних для виконання цих процесів. Якщо незаплановані процеси все ж враховуються під час планування Scrum ітерації, здебільшого кількісні значення не мають статистичного підґрунтя, за основу беруть показники, інтуїтивно визначені відповідальною за планування особою.

Проте, методи та засоби прогнозування часто використовують в інших галузях, таких як фінансова, метеорологічна, аграрна тощо. Зважаючи на розвиток засобів прогнозування та досвід їх використання в інших сферах для вирішення проблеми нехтування ймовірністю появи незапланованих процесів на етапі планування IT-проєкту, використання методів прогнозування не втрачає своєї актуальності. Питання підвищення точності планування та скорочення термінів реалізації процедур планування багато науковців і практиків досліджують практично безперервно, адже, вирішуючи їх, компанії мають можливість збільшувати ефективність виконання проєктів та підвищувати конкурентоспроможність.

Об'єкт дослідження – застосування систем прогнозування в процедурі планування проєктів із створення програмних продуктів.

Предмет дослідження – методи та засоби прогнозування ймовірної появи незапланованих процесів роботи під час формування плану IT-проєкту.

Мета роботи – проаналізувати особливості використання методів машинного навчання для прогнозування ймовірної появи незапланованих процесів під час управління IT-проєктами та підтвердити обґрунтованість їх використання під час планування завдань ітерації.

Для досягнення зазначеної мети визначено такі основні завдання дослідження:

- 1) проаналізувати останні публікації за темами, релевантними цьому дослідженню, такими як "Планування проєктів за методологією Scrum", "Метод градієнтного підсилення", "Машинне навчання в контексті прогнозування", "Незаплановані процеси роботи" тощо;

- 2) розробити концепцію прикладного використання процедури прогнозування під час планування процесів проєкту та обґрунтувати застосування обраного методу прогнозування як такого, що здатен зреалізувати мету дослідження;
- 3) Застосувати обраний метод прогнозування до зразка історичних даних реального IT-проєкту для виконання відповідних процедур та визначення передбачуваної кількості незапланованих процесів роботи під час окремої ітерації;
- 4) порівняти отримані прогнозовані значення з реальними показниками проєкту;
- 5) оцінити результати дослідження, вплив процедур прогнозування на планування ресурсів ітерації проєкту та обґрунтувати доцільність використання обраного алгоритму для досягнення мети.

Аналіз останніх досліджень та публікацій. Планування є одним з ключових компонентів сучасного підходу управління IT-проєктом з використанням підходу Scrum. Здебільшого перед менеджментом, заснованим на інтуїтивності та досвіді Scrum майстра або іншої уповноваженої особи, постає проблема під час планування проєкту. В емпіричному дослідженні С. Шафі, І. Вотелет, С. Поелманс та С. Хенг [11] вказано на проблему з планування проєкту, що реалізують з використанням методології Scrum, та наведено найефективніші шляхи їх вирішення, зокрема, збільшення кількості годин навчання персоналу. Розширюють спектр методів для покращення процедури планування науковці Н. Озчеліккан, Г. Тузкая, К. Алабас-Услу та Б. Сеннарроглу [9], пропонуючи для вирішення проблеми планування використовувати функції цілі для розрахунку необхідних ресурсів для виконання задач проєкту. Подібні дослідження створюють прецедент для подальшого вивчення проблеми планування у проєктному менеджменті за допомогою статистично-математичних методів оцінки компонентів. Погляд на проблему планування через призму прогнозування як допоміжного засобу для прийняття рішень пропонують М. Старон, В. Медінг та Б. Сьодерквіст [12], при цьому автори підтверджують доцільність застосування практики прогнозування у процедурі планування проєктних робіт. Дослідження проводили у межах одного з найбільших проєктів компанії Ericsson та обґрунтовують позитивний вплив найпростіших методів прогнозування.

Застосування рішень на підставі машинного навчання в межах процесу управління проєктами також висвітлено у сучасних дослідженнях. Такі рішення характерні для процесів управління проєктами, контролю, ідентифікуючи зв'язки між поставленими завданнями та бажаними результатами [10], класифікації зацікавлених сторін із зведенням до мінімуму рівнем суб'єктивності [6], прогнозування витрат в контексті управління коштами [3]. Релевантним дослідженням можна вважати також працю І. Б. Краєм, М. Б. Мабрук та Л. Д. Жод [4], в якій до уваги беруть не безпосередні процеси управління проєктами на етапі виконання, а вибір методології на нульовій стадії проєкту. У дослідженні автори тестують сім методів, відзначаючи, що найкращого результату досягають із використанням методу градієнтного підсилення, точність прогнозування досягла 94 %.

Застосування методу градієнтного підсилення в контексті прогнозування є особливо важливим у сучасних дослідженнях. У галузі машинного навчання та аналізу даних, цей метод використовують для оптимізації пара-

метрів моделей та підвищення точності прогнозів. Ітеративна природа градієнтного підсилення дає змогу ефективно апроксимувати глобальні екстремуми у просторі параметрів, що робить його важливим інструментом у завданнях прогнозування майбутніх подій, тенденцій чи значень. Приклад застосування цього методу описали науковці Д. Фернандес-Сереро, Х. А. Тро-яно, А. Якубик та А. Фернандес-Монтес [2], він містить певну кількість побічних чинників, серед яких – середовище хмарного обчислювального центру.

Використання методу градієнтного підсилення розглядають як інструмент оптимізації процедури планування розподілу ресурсів. Застосування цього методу дозволило розпізнавати шаблони використання ресурсів, а навчена регресійна модель досягла високого R^2 рівня, вищого за 0,8. Згадують даний метод і в інших сучасних дослідженнях, в основі яких є прогнозування, таких як: оптимізація використання електроенергії приватних осель [7], прогнозування цін на вуглець [1] чи прогнозування сатурації води в резервуарі [8]. Усі згадані дослідження на підставі методу градієнтного підсилення об'єднують мультифакторність об'єкта прогнозування.

Матеріали та методи дослідження. У роботі використано метод машинного навчання, який використовують для задач регресії та класифікації – регресор екстремального градієнтного підсилення (XGBoost). Як середовище реалізації алгоритму обрано Python 3, відповідну бібліотеку мови програмування "XGBoost". Візуалізацію результатів застосування алгоритму здійснено з допомогою бібліотеки "Matplotlib", методу "Pypplot".

Результати дослідження та їх обговорення / Research results and their discussion

Дослідження проводили в межах проекту підтримки апаратного забезпечення, який налічує понад п'ятдесят обчислювальних центрів та дев'ять тисяч серверів. Дані групували впродовж останніх чотирьохсот днів у періоди, які дорівнюють одній стандартній ітерації (тиждень).

Непередбаченими процесами роботи в управлінні обсягом робіт проекту вважатимемо завдання, які виникають із ситуацій, що не були заплановані заздалегідь. Зазвичай до таких робіт належать випадки виникнення неполадок чи збоїв у системах, які використовують для створення або підтримки програмного продукту. Трактуватимемо їх як інциденти (апаратні збої). Чинниками впливу обрано історичні дані, загальну кількість підтримуваного апаратного забезпечення та загальне навантаження на обчислювальні системи.

Упродовж дослідження використовували різні комбінації даних та було оцінено підходів для підбору найбільш точної моделі даних. Після обрання моделі даних, було проведено планування наступної ітерації Scrum (тиждень) двічі: один раз з урахуванням прогнозованих даних та другий раз – без урахування. Далі, після закінчення ітерації, проведено порівняльний аналіз підходів до планування та обрано ефективніший підхід.

Для початку побудовано модель регресії з урахуванням тільки історичних даних. Дані групують за періодами, в які були створені інциденти, а саме: денний проміжок часу, тижневий та місячний.

Зразок відповідних даних наведено в табл. 1, дані наведено вибірково, для демонстрації стилю та форматування даних. Повна вибірка налічує рівно чотириста

записів. Для техніки градієнтного підсилення – це репрезентативна вибірка даних, щоб продемонструвати можливість прогнозу.

Табл. 1. Історичні дані інцидентів / Historical data of incidents

Дата	Денна кількість інцидентів, шт.	Тижнева кількість інцидентів, шт.	Місячна кількість інцидентів, шт.
2023-01-02	0	5	79
2023-01-03	0	5	79
2023-01-04	1	5	61
2023-09-17	0	15	80
2023-09-18	2	10	82
2023-09-19	0	9	82
2024-01-20	1	18	36
2024-01-21	0	18	36
2024-01-22	15	32	51

В основі XGBoost є мінімізація функції цілі, яка містить дві частини – функції втрат та регуляризаційного члена. Функція втрат вимірює різницю між передбаченими значеннями та фактичними цільовими значеннями. Регуляризаційний член обмежує складність моделі регресії для запобігання перенаванчання. Загальна функція цілі J – це сума функції втрат та регуляризаційного члена:

$$J(\theta) = \sum_{i=1}^n L(y_i, \hat{y}_i) + \sum_{k=1}^K \Omega(f_k), \quad (1)$$

де: n – кількість спостережень; θ – параметри моделі, в т.ч. й параметри всіх дерев; y_i – фактичне цільове значення для i -го спостереження; $\hat{y}_i$ – передбачене значення для i -го спостереження від ансамблю слабких моделей даних; $L(y_i, \hat{y}_i)$ – функція втрат (наприклад, середньоквадратична помилка для регресії); K – кількість слабких моделей (дерев) даних; f_k – k -та слабка модель даних (дерево); $\Omega(f_k)$ – регуляризаційна функція для k -го дерева.

Остаточний прогноз – це сума прогнозів всіх слабких моделей даних, кожен з яких помножений на його коефіцієнт навчання:

$$\hat{y}_i = \sum_{k=1}^K \eta_k f_k(x_i), \forall i \in n, \quad (2)$$

де: $\hat{y}_i$ – остаточний прогноз для i -го спостереження; η – коефіцієнт навчання для кожного дерева; K – кількість дерев у моделі; $f_k(x_i)$ – прогноз, зроблений k -им деревом для i -го спостереження.

Як засіб реалізації цього методу було обрано мову програмування Python 3 з використанням відповідного пакету XGBoost. Усі дані збережено у файлі CSV та зчитано інтерпретатором. Зчитані дані поділено на ознаки або характеристики (features), мітки (labels) або цільові значення (target values). У групу ознак входять дані тижневого та місячного періодів. Цільове значення – це залежна змінна, яку модель регресії намагається передбачити на підставі вхідних ознак. Для процесу тренування мають бути наявні цільові значення, щоб алгоритм XGBoost міг оцінювати результативність дерев рішень. У цьому випадку цільовими значеннями будуть передбачені кількості інцидентів у період від кожної дати початку тижня (ітерацію). Отже, дані розширено додатковим атрибутом "Цільове значення", куди буде скопійовано дані із тижневого періоду з індексом $n-1$.

Зразок оновлених даних наведено в табл. 2. Потім їх розділено на два набори – тренувальний та тестовий.

Тренувальний набір даних – це підмножина даних, яку використовують для навчання моделі даних. Модель даних вивчає зв'язки між ознаками (features) і цільовими значеннями (target values) на цьому наборі. Тестовий набір даних – це інша підмножина, яку використовують для оцінювання ефективності моделі регресії на нових, раніше не бачених даних.

Табл. 2. Ознаки історичних даних та цільові значення інцидентів / Features of historical data and target values of incidents

Дата	Денна кількість інцидентів, шт.	Тижнева кількість інцидентів, шт.	Місячна кількість інцидентів, шт.	Цільове значення, шт.
2023-01-02	0	5	79	5
2023-01-03	0	5	79	5
2023-01-04	1	5	61	7
2023-09-17	0	15	80	10
2023-09-18	2	10	82	9
2023-09-19	0	9	82	8
2024-01-20	1	18	36	18
2024-01-21	0	18	36	32
2024-01-22	15	32	51	31

У цьому випадку тренувальним набором даних обрано записи до тридцять першого числа грудня дві ти-

сячі двадцять третього року. Масив даних за 2024 р. оголошено тестовим. Визначено кількість ітерацій алгоритму градієнтного посилення – 400. Кожна ітерація додає до ансамблю нове дерево рішень, яке коригує помилки попереднього ансамблю. Більша кількість ітерацій підвищує ризик перенавчання, за якого кількість дерев може стати зовеликою.

Швидкість навчання встановлено мінімальну (0.01), адже в дослідженні надають перевагу якості результату, а не швидкості цього процесу. Як метрику оцінювання якості моделі регресії під час тренування встановлено метод середньої абсолютної похибки. У даних спостережено кілька сплесків нетипових значень, а метод середньої похибки характеризується, зокрема стійкістю до подібних сплесків, оскільки він використовує абсолютні значення.

Візуалізацію результату процесу прогнозування подано на рис. 1. На графіку фактичні значення зображено синьою лінією, а прогнозовані – оранжевою. Для наочності було взято відрізок часу – останні 60 днів. Проаналізувавши графік, можна відзначити незначні відхилення фактичних даних від прогнозованих на часовому відрізку "23-12-31"- "24-01-11".



Рис. 1. Графік регресора XGBoost з ознаками історичних даних / XGBoost regressor graph with historical data features

Більші відхилення спостережено на часовому відрізку "24-01-12"- "24-01-21", невідповідність форми графіку на часовому відрізку "24-01-22"- "24-01-25", та певні відхилення на відрізку "24-01-29"- "24-02-01". Проте, незважаючи на кількість відхилень, оперуючи тільки історичними даними, модель регресії добре відтворила загальну тенденцію інцидентів. На останніх ітераціях алгоритму значення середньої абсолютної похибки становить 2.26499.

Логічно припустити, що кількість інцидентів залежить від стану та обсягу апаратного забезпечення. Тому до даних додано додатковий атрибут "Кількість серверів", який демонструє загальну кількість серверів, які здійснюють підтримку проекту. Зразок оновлених даних наведено в табл. 3. Для наочності динаміки показників кількісні характеристики апаратного забезпечення дещо збільшені.

Табл. 3. Ознаки історичних даних, кількості апаратного забезпечення та цільові значення інцидентів / Features of historical data, hardware counts, and incident targets

Дата	Кількість серверів, шт.	Денна кількість інцидентів, шт.	Тижнева кількість інцидентів, шт.	Місячна кількість інцидентів, шт.	Цільове значення, шт.
2023-01-02	8165	0	5	79	5
2023-01-03	8165	0	5	79	5
2023-01-04	8165	1	5	61	7
2023-01-05	8155	2	7	60	6
2023-09-17	9841	0	15	80	10
2023-09-18	9841	2	10	82	9
2023-09-19	9841	0	9	82	8
2023-09-20	9842	0	8	82	8
2024-01-20	10117	1	18	36	18
2024-01-21	10117	0	18	36	32
2024-01-22	10137	15	32	51	31
2024-01-23	10147	3	31	54	29

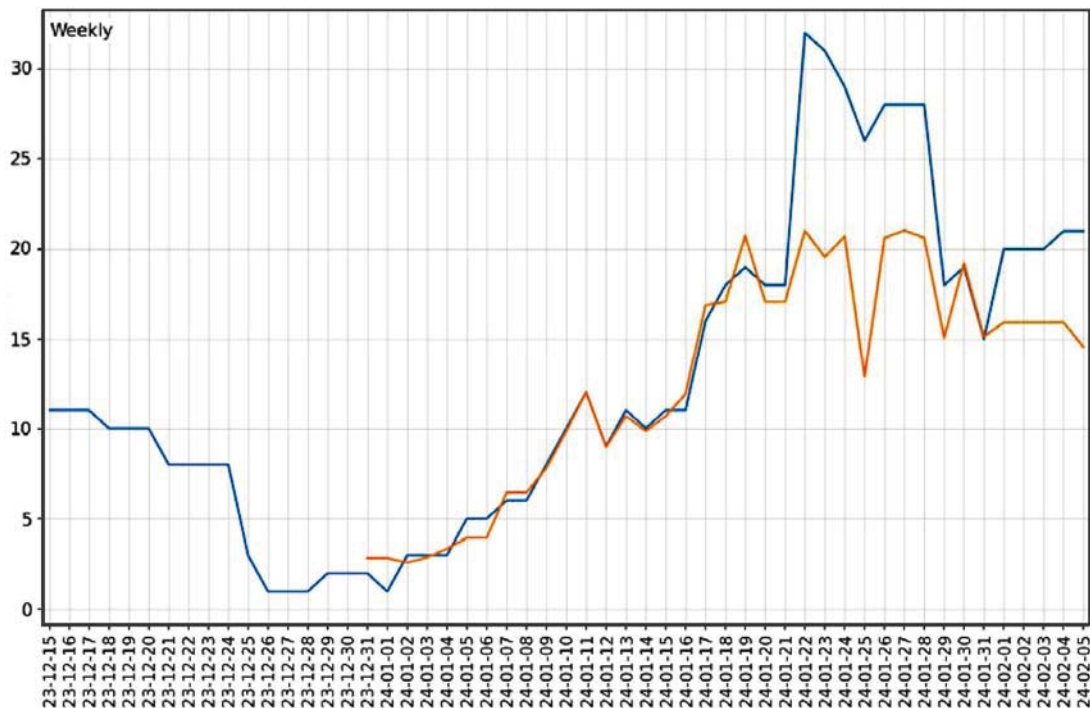


Рис. 2. Графік регресора XGBoost з ознаками історичних даних та кількості апаратного забезпечення / XGBoost regressor graph with historical data and hardware quantity features

Для урахування у моделі регресії чиннику кількості серверів, додано дані цього атрибуту в тренувальні та тестові набори даних як ознаки. Процес прогнозування запущено повторно та візуалізовано результат (рис. 2). Порівнюючи графіки на рис. 1 та 2, можна відзначити збільшення значень показників на часовому відрізку "23-12-31"- "24-01-18", де відхилення зменшилися, та значно точніше відтворюють криву інцидентів. Однак, справедливо відзначити значне погіршення точності прогнозу на часовому відрізку "23-12-21"- "24-02-05", за винятком часового відрізка "23-12-30"- "23-12-31". Форма графіку демонструє криву спостережуваних значень, проте фактичні значення значно відрізняються від прогнозованих. Причини цього відхилення передусім пояснюють перенаванчанням або відсутністю зв'язку між кількістю апаратного забезпечення та кількістю збоїв у його масиві. Проте, незважаючи на наявність незначної похибки в отриманих прогнозованих значеннях, потрібно відзначити покращену геометрію кривої прогнозування. Візуалізація ще раз підтверджує зменшення значення середньої абсолютної похибки останніх ітерацій алгоритму, яке становить 1.78991. Це значення менше за відповідний показник моделі, зображеної на рис. 1, що свідчить про вищу точність прогнозування.

На наступному етапі дослідження додано ще один чинник впливу на роботу апаратного забезпечення, а саме: мережеве навантаження. Абсолютними значеннями мережевого навантаження взято суму пікових показників прийому даних (RX) та передачі даних (TX) за добу у відповідні дати. Наведені значення вимірюють у гігабітах на секунду (Gbps). Зібрані відповідні дані було додано окремим атрибутом до даних, використаних у цьому дослідженні. Зразок розширених даних наведено в табл. 4. Використовуючи отримані дані для проведення тренування, модель регресії краще розпізнаватиме шаблони навантаження на апаратне забезпечення, такі як періоди підвищеного запиту чи періоди тиші, наприклад у вихідні дні чи свята.

Табл. 4. Ознаки історичних даних, кількості апаратного забезпечення, мережевого навантаження та цільові значення інцидентів / Features of historical data, hardware count, network load, and incident target values

Дата	Мережеве навантаження, Гб/с	Кількість серверів, шт.	Денна кількість інцидентів, шт.	Тижнева кількість інцидентів, шт.	Місячна кількість інцидентів, шт.	Цільове значення, шт.
2023-01-02	363.66	8165	0	5	79	5
2023-01-03	1125.18	8165	0	5	79	5
2023-01-04	1100.10	8165	1	5	61	7
2023-01-05	1207.26	8155	2	7	60	6
2023-09-17	507.30	9841	0	15	80	10
2023-09-18	1608.54	9841	2	10	82	9
2023-09-19	1658.70	9841	0	9	82	8
2023-09-20	1575.48	9842	0	8	82	8
2024-01-20	663.48	10117	1	18	36	18
2024-01-21	787.74	10117	0	18	36	32
2024-01-22	2348.40	10137	15	32	51	31
2024-01-23	2622.00	10147	3	31	54	29

Мережеве навантаження апаратного забезпечення корелюється зі збоями його роботи в кількох додаткових особливостях. Збільшення мережевого трафіку може призводити до низки негативних наслідків, таких як перевантаження мережевих каналів, велика затримка та втрати пакетів даних. Ці явища, зі свого боку, можуть викликати деякі несправності в роботі апаратного забезпечення.

З огляду на це прогнозування кількості апаратних збоїв на підставі значень мережевого навантаження, історичних даних збоїв та кількості апаратного забезпечення за допомогою методу XGBoost regression є цікавим і потенційно ефективним підходом.

Нові дані додано у масив загальних даних як ознаку (feature), та повторно запущено процес прогнозування. Результат прогнозування візуалізовано (рис. 3).

Проаналізувавши графік на рис. 3 та порівнявши його з двома попередніми зображеннями на рис. 1 та 2, можна відзначити дещо більші відхилення в точці "24-01-01" та на часовому відтинку "24-01-07"- "24-01-10". Проте, порівняно з попередніми моделями регресії, поточна модель значно краще відобразила поведінку кривої на відтинку "24-01-20"- "24-02-05" за винятком знач-

ного відхилення у точці "24-01-25". Без урахування незначних відхилень в абсолютних значеннях, модель доволі точно спрогнозувала два сплески інцидентів у точці "24-01-22" та на часовому відтинку "24-01-26"- "24-01-28". Більш наочно це представлено на графіку (рис. 4), де відображено весь масив даних, використаних у дослідженні.



Рис. 3. Графік регресора XGBoost з ознаками історичних даних, кількості апаратного забезпечення та мережевого навантаження / XGBoost regressor graph with historical data, hardware quantity and network utilization features

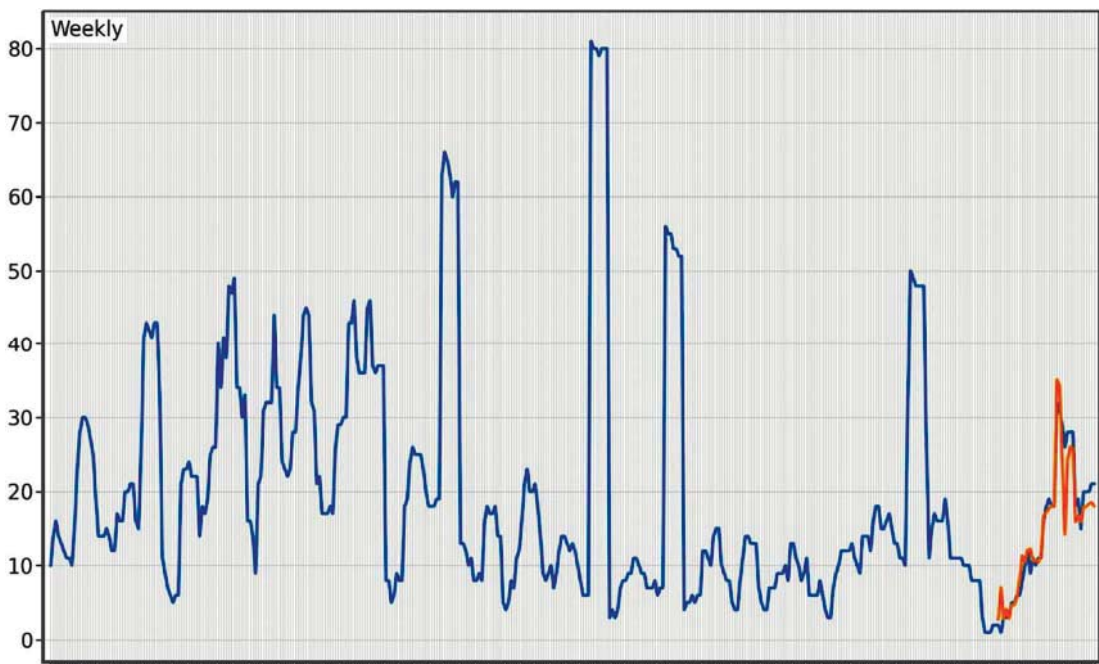


Рис. 4. Повний графік регресора XGBoost з ознаками історичних даних, кількості апаратного забезпечення та мережевого навантаження / Full XGBoost regressor graph with historical data, hardware quantity and network utilization features

Загалом, поточна модель регресії є більш надійною та найуспішнішою з випробовуваних, про що свідчить і значення отриманої середньої абсолютної похибки 1.70149. Згідно з прогнозом останньої моделі XGBoost Regressor, на наступну ітерацію, тобто тиждень з 2024-02-05 по 2025-02-09, варто очікувати $18.034161 \approx 18$ інцидентів. Основні переваги XGBoost полягають у значній швидкості як навчання, так і точності прогнозу-

вання, здатності виконувати згенеровані функції втрат і багатоваріантної структури, а також у можливості узагальнення результатів для різних типів даних та завдань.

Далі, використовуючи метод А/В для перевірки коректності прогнозування ймовірності появи інцидентів, обрали дві команди: контрольну команду (А) і експериментальну команду (В). Кожною командою проведено процедуру планування процесів для однієї і тієї ж ітера-

ції: з урахуванням прогнозованих моделлю регресії інцидентів (А), та без їх урахування (В). Для планування було обрано такі вхідні дані:

- в резерві задач перебувало 28 планових завдань до виконання;
- команда налічує шість осіб, що еквівалентно 240 людино-годин. Впродовж ітерації не заплановано відпусток, святкових чи вихідних днів;
- середня оцінка ресурсів для одного завдання з резерву – два робочих дні, або 16 людино-годин;
- середня оцінка ресурсів для одного незапланованого завдання типу "інцидент" – 0.4 робочого дня, або 3 людино-години.

Результати планування, проведені обома командами, наведено в табл. 5. Згідно з даними цієї таблиці, проектний менеджер команди А відштовхувався від кількості прогнозованих інцидентів та середньої оцінки зусиль на їх вирішення. Тому, буфером було оголошено 54 людино-годин. Буфером вважають виділену кількість годин на виконання незапланованих процесів роботи, чи екстрені ситуації. Враховуючи це, заплановано виконання одинадцяти завдань, із залишком планування в 10 людино-годин.

Табл. 5. Результати планування ітерації Scrum / Results of planning the Scrum iteration

Планування	Взято задач із резерву до виконання, шт.	Кількість годин оголошених буфером, люд. год	Залишок людино-годин, люд. год
З урахуванням прогнозу	11	54	10
Без урахування прогнозу	13	24	8

Проектний менеджер команди В підійшов до планування із стандартним методом буферизації 10 % усього ресурсу, в такий спосіб виділивши 24 людино-годин на виконання незапланованих процесів роботи. Це дало змогу запланувати виконання більшої кількості завдань (тринадцяти) із меншим залишком – 8 людино-годин.

Після проведення ітерації, отримані результати було зібрано та наведено в табл. 6.

Табл. 6. Результати ітерації Scrum / Scrum iteration results

Планування	Взято задач із резерву до виконання, шт.	Виконано задач протягом ітерації, шт.	Виконано незапланованих процесів, шт.	Кількість ресурсу оголошених буфером, люд. год	Витрачено буферних ресурсів на інциденти, люд. год	Витрачено залишок людино-годин, тип
З урахуванням прогнозу	11	12	14	54	78 %	Планові роботи
Без урахування прогнозу	13	12	14	24	100 %	Незаплановані процеси

Проаналізувавши результати виконання ітерації можна зробити такі висновки:

- плануванню роботи команди А притаманне недооцінювання її можливостей, а саме запланувало виконання меншої кількості завдань ніж було виконано в дійсності. Також є надлишкове планування буферу, проте додаткові ресурси, заплановані у буфері, дали змогу команді виконати ще одне додаткове завдання, взяту з резерву;
- особливості планування роботи команди В істотно відрізняються: спостерігається планування надлишку часу на вико-

нання планових процесів роботи та скорочення ресурсів, виділених на виконання процесів незапланованої роботи. Унаслідок цього, під час реалізації плану проекту, складеного без урахування прогнозованих інцидентів, команда не змогла виконати покладені на неї зобов'язання, витратила усю кількість ресурсів, які зберігалися як резервні у буфері, не виконала одного із запланованих завдань, оскільки потребували вирішення не передбачені планом проблеми.

Обговорення результатів дослідження. Вирішення проблеми оцінювання обсягу необхідних ресурсів для виконання завдань та проведення на цій основі коректного планування запропоновано доволі варіативно, залежно від об'єктів дослідження. У деяких випадках результатів досягаємо завдяки покращенню кваліфікації персоналу шляхом навчань, як зазначено у роботі [11], в інших випадках залученням додаткових чинників впливу [9, 12]. Додатковими чинниками впливу можна вважати додаткові інструменти, або ж методології планування, наприклад мозковий штурм оцінки зусиль у вигляді гри "планувальний покер", які покращують процес планування загалом, це відзначають В. Махніч та Т. Говель [5].

Проте однією із найважливіших причин некоректного планування вважають нехтування незапланованими процесами роботи, які переривають загальний процес, відволікають працівників та створюють негативний психологічний фон для праці. Існують методики резервування ресурсів саме на такі випадки, як буферизація. Проте зазвичай вони базуються радше на інтуїтивності відповідальної персони, та не враховують кількості чинників впливу на виникнення незапланованих процесів роботи. Оскільки кількість незапланованих процесів роботи, в контексті ІТ проектів, має залежність від зовнішніх чинників, таких як навантаження, кількість вузлів підтримки, сезонність і т. ін., цей показник може піддаватися прогнозуванню за умови врахування чинників впливу.

Це дослідження не піддається прямому порівнянню з попередніми роботами, через специфіку середовища, проте застосування машинного навчання, як допоміжного засобу в процесі управління проектами було досліджено раніше, зокрема науковцями І. Б. Краєм, М. Б. Мабрук та Л. Д. Жод [4], де автори вирішували завдання вибору методології на стадії планування проекту різними методами машинного навчання, з найкращим показником використання градієнтного підсилення – 94 % точності підбору.

Беручи до уваги успішність застосування методу градієнтно-підсиленої регресії в контексті прогнозування в роботах [1, 7, 8], автори застосували цей метод для прогнозування кількості незапланованих процесів роботи у проекті підтримки апаратного забезпечення, що працює за методологією Scrum. Даний метод не є вимогливим до показників різних чинників та має змогу ефективно працювати з великими наборами даних. Послідовно випробовуючи комбінації наявних даних для тренування, у цьому дослідженні обрано модель регресії, що найбільш точно прогнозує значення, з показником абсолютної середньої помилки – 1.70149. Застосовуючи модель, зроблено прогноз кількості інцидентів на чергову ітерацію Scrum.

Виконано два планування: з урахуванням результатів прогнозу (планування для команди А), та без урахування результатів прогнозу (планування для команди В). Планування А показало результат, який засвідчив

100 % виконання завдань та 100 % вирішених питань щодо інцидентів. Також наявні обґрунтовані причини обсягів ресурсів у буфері, та використання їх за призначенням. Планування В охарактеризували 92 % виконаних зобов'язань, та 100 % вирішених питань щодо інцидентів. Розмір ресурсів у буфері виявився недостатнім для використання їх за призначенням, тому довелося нехтувати часткою зобов'язань для вирішення незапланованих завдань.

Судячи з отриманих результатів, врахування прогнорозованих значень потенційних незапланованих процесів роботи позитивно впливає на оцінювання ресурсів та зменшує ризик невиконання оголошених зобов'язань.

Надалі метод можна вдосконалювати, комбінуючи інші ознаки, що є чинниками впливу на кількість незапланованих процесів роботи. Збільшуючи обсяг історичних даних для тренувальних масивів, а також комбінуючи параметри градієнтного підсилення, можна досягти якомога більшої точності прогнозування. Цей метод можна використовувати як алгоритм роботи певної інформаційної системи, покликаної надавати допомогу в процесах управління проектами.

Володіючи правильними даними та ознаками, підхід можна застосовувати у будь-якій галузі, у межах методології Scrum.

Отже, за результатами виконаної роботи можна сформулювати такі наукову новизну та практичну значущість результатів дослідження.

Наукова новизна отриманих результатів дослідження – розроблено підхід для вдосконалення процесу планування ітерацій проекту, який реалізують з використанням методології Scrum, та доведено ефективність використання технологій машинного навчання в процесах управління проектами.

Практична значущість результатів дослідження – отримані результати можна використовувати як незалежний метод підвищення ефективності планування ітерацій в методології створення програмних продуктів "Scrum", а також, на підставі результатів дослідження, можна впровадити використаний алгоритм як частину розширеної інформаційної системи, спрямованої на підвищення ефективності процесів управління IT-проектами.

Висновки / Conclusions

Проаналізовано проблематику планування ітерацій проектів, що реалізують з використанням методології Scrum. Сформовано гіпотезу про доцільність використання технологій машинного навчання як допоміжного засобу для підвищення ефективності процесу планування ітерацій проекту, і як наслідок, вдосконалення процесу управління проектами загалом. Під час дослідження, доведено цю гіпотезу та продемонстровано приклад практичного застосування підходу.

За результатами дослідження можна зробити такі висновки:

1. Висвітлено тематику особливостей планування в процесах управління проектами, що працюють з використанням методології Scrum. Досліджено наявні пропозиції щодо часткових рішень проблем планування. Виокремлено проблему нехтування незапланованими процесами роботи, як основу цього дослідження.
2. Проаналізовано публікації, дотичні до теми дослідження, зосереджено увагу на прецедентах використання технологій машинного навчання як допоміжного засобу у процесах планування. Обґрунтовано доцільність ви-

користання методу градієнтного підсилення для вирішення проблеми дослідження. Доведено, що основні переваги XGBoost полягають у високій швидкості навчання, високій точності прогнозування, здатності виконувати згенеровані функції втрат і багатопарові структури, а також у можливості узагальнення результатів для різних типів даних та завдань.

3. Представлено предмет дослідження, яким є кількість незапланованих процесів роботи (інцидентів) у проекті підтримки апаратного забезпечення, з історичними даними кількості збоїв у пристроях, загальної кількості серверів та значень мережевого навантаження за чотирьох днів (від другого січня 2023-го року по 5 лютого 2024 р.).
4. Випробувано різні комбінації історичних даних, як масивів тренувальних даних для моделі машинного навчання, та обрано модель регресії із найвищою точністю прогнозу. Модель враховувала усі наявні ознаки, а саме: історичні дані кількості збоїв у пристроях, загальну кількість серверів та значень мережевого навантаження і показала результат середньої абсолютної похибки 1.70149, середнє значення точності прогнозування становить 87 %.
5. Наведено приклад застосування результатів прогнозу у формі проведення планування завдань ітерації IT-проекту з урахуванням значень прогнозованої кількості незапланованих процесів роботи. Проведено порівняння з аналогічним процесом планування, проте без урахування результатів прогнозування.
6. Проаналізовано результати порівняння підходів та обґрунтовано доцільність використання методу градієнтного підсилення як допоміжного засобу для підвищення ефективності планування ітерацій у проектах, що реалізують за методологією Scrum, та наведено перспективи подальшого розвитку питання, галузі застосування.

References

1. Deng, S., Su, J., Zhu, Y., Yu, Y., & Xiao, C. (2024). Forecasting carbon price trends based on an interpretable light gradient boosting machine and Bayesian optimization. *Expert Systems With Applications*, 242, 122502. <https://doi.org/10.1016/j.eswa.2023.122502>
2. Fernández-Cerero, D., Troyano, J. A., Jakóbič, A., & Fernández-Montes, A. (2022). Machine learning regression to boost scheduling performance in hyper-scale cloud-computing data centres. *Journal of King Saud University – Computer and Information Sciences*, 34(6), 3191–3203. <https://doi.org/10.1016/j.jksuci.2022.04.008>
3. İnan, T., Narbaev, T., & Hazir, N. (2022). A Machine Learning Study to Enhance Project Cost Forecasting. *IFAC-PapersOnLine*, 55(10), 3286–3291. <https://doi.org/10.1016/j.ifacol.2022.10.127>
4. Kraiem, I. B., Mabrouk, M. B., & Jode, L. D. (2023). A Comparative Study of Machine Learning Algorithm for Predicting Project Management Methodology. *Procedia Computer Science*, 225, 665–675. <https://doi.org/10.1016/j.procs.2023.10.052>
5. Mahnič, V., & Hovelja, T. (2012). On using planning poker for estimating user stories. *Journal of Systems and Software*, 85(9), 2086–2095. <https://doi.org/10.1016/j.jss.2012.04.005>
6. Mariani, C., Navrotska, Y., & Mancini, M. (2023). Unsupervised machine learning for project stakeholder classification: Benefits and limitations. *Project Leadership and Society*, 4. <https://doi.org/10.1016/j.plas.2023.100093>
7. Nie, P., Roccotelli, M., Fanti, M. P., Ming, Z., & Li, Z. (2021). Prediction of home energy consumption based on gradient boosting regression tree. *Energy Reports*, 7, 1246–1255. <https://doi.org/10.1016/j.egyr.2021.02.006>
8. Otchere, D. A., Ganat, T. O. A., Ojero, J. O., Tackie-Otoo, B. N., & Taki, M. Y. (2022). Application of gradient boosting regression model for the evaluation of feature selection techniques in improving reservoir characterisation predictions. *Journal of Pet-*

- roleum Science and Engineering, 208, 109244. <https://doi.org/10.1016/j.petrol.2021.109244>
9. Ozcelikkan, N., Tuzkaya, G., Alabas-Uslu, C., & Sennaroglu, B. (2022). A multi-objective agile project planning model and a comparative meta-heuristic approach. *Information and Software Technology*, 151. <https://doi.org/10.1016/j.infsof.2022.107023>
 10. Santos, J. I., Pereda, M., Ahedo, V., & Galán, J. M. (2023). Explainable machine learning for project management control. *Computers & Industrial Engineering*, 180. <https://doi.org/10.1016/j.cie.2023.109261>
 11. Shafiee, S., Wautelet, Y., Poelmans, S., & Heng, S. (2023). An empirical evaluation of scrum trainings suitability for the model-driven development of knowledge-intensive software systems. *Data & Knowledge Engineering*, 146. <https://doi.org/10.1016/j.datak.2023.102195>
 12. Staron, M., Meding, W., & Söderqvist, B. (2010). A method for forecasting defect backlog in large streamline software development projects and its industrial evaluation. *Information and Software Technology*, 52(10), 1069–1079. <https://doi.org/10.1016/j.infsof.2010.05.005>

O. O. Hud, N. E. Kunanets

Lviv Polytechnic National University, Lviv, Ukraine

INCIDENT PREDICTION IN SPRINT PLANNING IN IT PROJECTS

The study focuses on justifying the feasibility of using machine learning technology to improve the efficiency of planning processes that will be performed in the iteration (Sprints) of an IT project implemented using the Scrum methodology. The problems that arise when planning the tasks of such a project are considered. The reasons for incorrect planning and possible solutions to the problem are analysed. The problem of managing unplanned processes in a project is highlighted and also the impact of their occurrence on the correctness of iteration planning is determined. The feasibility of using machine learning technologies to predict the number of unplanned tasks in future iterations is analyzed and unplanned tasks are proposed to be treated as incidents (hardware failures). The factors that influence the occurrence of unplanned work processes in three segments are identified as follows: the historical number of incidents, hardware, and network load data. A forecasting tool, the extreme gradient boosting regressor, was selected and used to predict the probability of unplanned processes. The basic principles of the algorithm are reviewed. The advantages of using this method in the context of the studied environment are described. The features of the procedure for comparative analysis of regression models are highlighted and the results of the method are visualized. The influence of the selection of these features on the result of the forecasting process is demonstrated. The choice of the working regression model is substantiated, and the forecasting results are presented. A practical task is described to analyse the effectiveness of the approach under study. The control and experimental teams for the study are formed. The choice of a working regression model is substantiated and the forecasting results are presented. An example of their use in planning work processes in iteration is given. A comparative analysis of approaches to planning iterations with and without forecasting results is carried out. The results of the analysis are presented and the impact of forecasting on the decision-making process is assessed. The effectiveness of applying the extreme gradient boosting regression method to the planning of project iteration processes implemented using the Scrum methodology is proved. Some prospects for the development of research areas and the application of the obtained results are presented.

Keywords: project management; machine learning; extreme gradient boosting regressor; Scrum.