

УДК 004.8

DOI: 10.31673/2412-9070.2025.023411

І. Ю. КОЛОМІЄЦЬ, студент;

ORCID: 0009-0002-4696-1619

І. В. ЗАМРІЙ, доктор техн. наук, професор,

ORCID: 0000-0001-5681-1871

Б. С. КАЛИНЮК, аспірант;

ORCID: 0009-0009-9569-9304

Ю. П. БАЖАН, аспірант;

ORCID: 0009-0000-6388-3267

Т. П. ДОВЖЕНКО, канд. техн. наук,

ORCID: 0000-0002-0352-8391

Державний університет інформаційно-комунікаційних технологій, Київ

СУЧАСНІ ПІДХОДИ ДО АВТОМАТИЗАЦІЇ УПРАВЛІННЯ ЗАВДАННЯМИ ДЛЯ МАЛИХ КОМАНД З ВИКОРИСТАННЯМ МЕТОДІВ МАШИННОГО НАВЧАННЯ

Сучасні технології управління проектами та завданнями пропонують численні інструменти, які забезпечують базові функції для організації роботи команд. Однак, більшість із цих рішень орієнтовані на великі компанії, що часто робить їх надто складними для малих команд. Відсутність інструментів для автоматизованого прогнозування часу виконання завдань і визначення їхніх пріоритетів обмежує ефективність таких систем. Малі команди потребують гнучких рішень, які зменшують навантаження на організацію роботи, автоматизують рутинні завдання та сприяють раціональному розподілу ресурсів. Тому, проблемою є відсутність автоматизованих засобів для точного прогнозування часу виконання завдань і визначення їх пріоритетності, що призводить до неефективного розподілу ресурсів та затримок у роботі.

Проаналізовано основні підходи до автоматизації управління завданнями, зокрема, методи, які можуть бути застосовані для прогнозування часу виконання завдань і визначення пріоритетів за допомогою машинного навчання. Особливу увагу приділено алгоритмам TF-IDF, Random Forest Regressor і K-means. TF-IDF дозволяє ефективно обробляти текстові описи завдань, перетворюючи їх у числові ознаки, що забезпечує аналітичну базу для роботи моделей машинного навчання. Random Forest Regressor використовується для точного прогнозування часу виконання завдань, що допомагає командам планувати робочий процес. Алгоритм K-means застосовується для кластеризації завдань за рівнем їхньої важливості та складності, забезпечуючи автоматичне визначення пріоритетів.

Впровадження штучного інтелекту (AI) у системи керування завданнями стало руйнівною силою у сучасному стрімкому робочому середовищі, змінивши те, як команди планують, розставляють пріоритети та виконують завдання. Цю тенденцію найкраще ілюструють такі платформи, як Trello, Monday.com і Asana, які використовують технологію штучного інтелекту для оптимізації робочих процесів проєктів, автоматизації стомлюючої роботи та пропонують рекомендації.

Популярні інструменти, такі як Trello, Asana, Jira, Wrike забезпечують базовий функціонал для управління завданнями, але не використовують методи машинного навчання для автоматизації. Розглянуті підходи можуть бути інтегровані в існуючі системи або реалізовані як окреме рішення для малих команд, що прагнуть підвищити продуктивність без значних витрат на складні платформи.

Результати дослідження підкреслюють важливість використання машинного навчання для автоматизації управління завданнями. Такий підхід дозволяє зменшити залежність від людського фактору, знизити час на організацію завдань та оптимізувати процес планування. Крім того, це сприяє підвищенню ефективності командної роботи, що особливо актуально для малих колективів, які працюють в умовах обмежених ресурсів.

Ключові слова: автоматизація управління завданнями; машинне навчання; прогнозування часу; кластеризація; TF-IDF; Random Forest Regressor; K-means.

© Коломієць І. Ю., Замрій І. В., Калинюк Б. С., Бажан Ю. П., Довженко Т. П., 2025

Вступ

У сучасному світі людям часто доводиться стикатися із зростаючим обсягом роботи, напруженим графіком і потребою ефективно керувати часом [1]. Відсутність ефективних інструментів для автоматизації процесів управління часто стає перешкодою для підвищення продуктивності. Впровадження штучного інтелекту (AI) у системи керування завданнями стало руйнівною силою у сучасному стрімкому робочому середовищі, змінивши те, як команди планують, розставляють пріоритети та виконують завдання [2]. Цю тенденцію найкраще ілюструють такі платформи, як Trello, Monday.com і Asana, які використовують технологію штучного інтелекту для оптимізації робочих процесів проєктів, автоматизації стомлюючої роботи та пропонують рекомендації [2]. Але, ці системи не завжди адаптовані до потреб малих команд, які мають обмежені ресурси і прагнуть максимальної ефективності при мінімальних витратах часу та зусиль.

Основною проблемою є відсутність спеціалізованих інструментів для автоматизованого прогнозування часу виконання завдань і визначення їх пріоритетності. Ці функції є важливими для малих команд, оскільки дозволяють не лише оптимізувати розподіл завдань, але й мінімізувати людські помилки, які можуть призводити до зриву дедлайнів та зниження продуктивності. У великих командах та корпораціях такі функції часто реалізуються за рахунок додаткових ресурсів або спеціалізованих рішень, однак для малих команд ці підходи недоцільні через їх високу вартість і складність впровадження.

Науковий підхід до вирішення цієї проблеми передбачає впровадження методів машинного навчання, які дозволяють автоматизувати ключові аспекти управління завданнями. Зокрема, використання алгоритмів для прогнозування часу виконання завдань може значно покращити процес планування. Це забезпечує точніші оцінки витрат часу на виконання завдань, що є критично важливим для малих команд, де ефективний розподіл ресурсів є пріоритетом. Також важливою є автоматична кластеризація завдань за їх пріоритетністю, що дозволяє оперативно визначати, які завдання потребують першочергового виконання і уникати недоцільні витрат часу на менш важливі завдання.

Інтеграція алгоритмів машинного навчання, таких як TF-IDF, Random Forest Regressor і K-means дозволяє досягнути цієї мети.

Розв'язання цих проблем є важливим не лише з точки зору практичного використання, але й у контексті наукових досліджень. Застосування машинного навчання для управління завданнями відкриває нові можливості для вивчення взаємозв'язків між характеристиками завдань, їх складністю, пріоритетністю та часом виконання. Також це дозволяє вдосконалити існуючі алгоритми та методи аналізу даних, адаптувавши їх до специфічних умов малих команд.

Аналіз останніх досліджень і публікацій

Актуальні дослідження у сфері управління завданнями свідчать, що для оптимізації процесів необхідна автоматизація таких аспектів, як пріоритизація завдань, прогнозування часу виконання, аналіз складності та відстеження прогресу. Зокрема, машинне навчання відіграє важливу роль у вирішенні цих завдань, дозволяючи здійснювати прогнозування та класифікацію даних для більш ефективного управління проєктами.

Один з таких методів - TF-IDF (Term Frequency-Inverse Document Frequency) [3], активно використовується для перетворення текстових даних, таких як описи завдань, у числові ознаки, що можуть бути проаналізовані моделями машинного навчання. Цей метод дозволяє виділити найважливіші слова та фрази у текстах, що допомагає при класифікації завдань та прогнозуванні часу виконання завдань, що, в свою чергу, може покращити ефективність управління проєктами.

У поєднанні з кластеризацією K-means [4], яка дозволяє групувати завдання за схожістю, можна автоматично визначати пріоритетність завдань. Це дозволяє командам зосередитися на найбільш важливих і термінових завданнях, а також допомагає у розподілі ресурсів.

Крім того, Random Forest Regressor [5] є методом для прогнозування часу виконання завдань на основі даних, допомагаючи точно оцінити, скільки часу знадобиться для завершення кожного завдання.

Ці методи машинного навчання створюють основу для побудови ефективних автоматизованих систем управління завданнями, що дозволяє мінімізувати ручне управління та знизити ймовірність помилок при прогнозуванні або відстеженні завдань. Таким чином, впровадження машинного навчання в управління завданнями покращує точність, ефективність та швидкість роботи малих команд, сприяючи їхньому росту та розвитку.

Це підкреслює необхідність інтеграції таких інноваційних підходів для досягнення більш точних та ефективних результатів у малих командах, де ресурси обмежені, а ефективність є ключовим фактором для успіху.

Результати дослідження

Метою статті є огляд сучасних підходів до автоматизації управління завданнями для малих команд з акцентом на методах машинного навчання, що можуть забезпечити оптимізацію часу виконання завдань та пріоретизації для підвищення продуктивності та ефективності роботи команди.

Автоматизація управління завданнями є невід'ємною частиною сучасних підходів до організації робочих процесів, особливо у малих командах. Із зростанням складності проєктів і обмеженістю ресурсів, необхідність у впровадженні автоматизованих систем стає дедалі очевиднішою. Відсутність автоматизації значно ускладнює процеси планування, прогнозування та аналізу прогресу, що призводить до втрати часу, ресурсів і зниження продуктивності.

У сучасному світі автоматизація забезпечує значну перевагу, дозволяючи командам досягати більш високих результатів із мінімальними витратами. Вона забезпечує не лише підвищення ефективності, але й зменшення кількості помилок завдяки автоматичному прогнозуванню часу виконання завдань, визначенню пріоритетів і моніторингу прогресу. Зокрема, використання алгоритмів машинного навчання відкриває нові можливості для аналізу та управління завданнями, пропонуючи більш адаптивний і гнучкий підхід.

Машинне навчання, зокрема методи TF-IDF, Random Forest Regressor і K-means кластеризація, пропонує ефективний підхід до вирішення цих проблем.

TF-IDF є одним із найпопулярніших методів обробки текстових даних. Цей метод використовується для визначення важливості термінів у тексті на основі частоти їхнього вживання. Він складається з двох основних компонентів: частота терміну (TF), яка визначає, наскільки часто слово зустрічається у конкретному документі і зворотна частота документу (IDF), що враховує, наскільки рідкісним є слово у корпусі текстів. У системах управління завданнями TF-IDF дозволяє аналізувати описи завдань, виділяючи ключові слова, які визначають суть задачі. Це спрощує процес класифікації завдань і їхню подальшу обробку алгоритмами машинного навчання. Наприклад, у тексті завдання "Створення логіну та авторизації на PHP" TF-IDF може визначити, що "логін" і "авторизація" є ключовими термінами, на яких потрібно зосередитися. Крім того, TF-IDF дозволяє створювати числові вектори для текстових даних, що є основою для побудови моделей прогнозування або класифікації.

Random Forest Regressor є методом ансамблевого машинного навчання, який застосовується для задач регресії, тобто прогнозування числових значень. Цей алгоритм працює шляхом створення ансамблю дерев рішень, кожне з яких робить свій прогноз, а кінцевий результат отримується як середнє значення прогнозів усіх дерев. У контексті управління завданнями Random Forest Regressor дозволяє прогнозувати час виконання задач на основі їхнього опису та історичних даних. Наприклад, завдання "Налаштування бази даних" може бути пов'язане з даними про час виконання схожих задач у минулому, що дозволяє алгоритму точно визначити, скільки часу знадобиться для його реалізації. Завдяки своїй здатності обробляти великі обсяги даних і враховувати взаємозв'язки між різними ознаками, Random Forest Regressor

є одним із найточніших методів прогнозування. Крім того, цей метод має високу стійкість до перенавчання, що робить його ефективним для роботи з реальними даними.

Метод K-means кластеризації використовується для автоматичного групування даних за схожістю. У системах управління завданнями цей метод дозволяє кластеризувати задачі за їхньою складністю, важливістю або іншими характеристиками. Наприклад, завдання можуть бути розділені на кластери, такі як "високий пріоритет", "середній пріоритет" і "низький пріоритет", на основі їхнього опису та інших ознак. Алгоритм K-means працює шляхом визначення центротидів кластерів і поступового оновлення їхніх положень до досягнення стабільності. У процесі управління завданнями це дозволяє командам зосередитися на виконанні найбільш важливих задач, водночас ефективно розподіляючи ресурси для вирішення менш критичних завдань. Метод також є корисним для візуалізації даних, оскільки групування завдань може бути представлено у вигляді графіків або діаграм, що сприяє прийняттю обґрунтованих рішень.

Для більш детального поняття методології розглянемо формальний опис моделі та алгоритму.

Вхідними даними є:

$D = \{d_1, d_2, \dots, d_n\}$ – множина текстових описів завдань;

$T = \{t_1, t_2, \dots, t_n\}$ – відповідний вектор фактичного часу виконання (у днях).

За допомогою методу TF-IDF кожне завдання d_i перетворюється у числовий вектор ознак x_i . Для відображення залежності між ознаками завдань x_i та часом їх виконання t_i використовується модель Random Forest Regressor, яка апроксимує функцію:

$$f(x) \approx t,$$

де t – прогнозований час виконання. Для визначення пріоритетності завдань застосовується K-means кластеризація, що групує вектори x_i у k кластерів, кожен з яких відповідає певному рівню важливості.

Покроковий алгоритм роботи моделі у вигляді блок-схеми наведено на рис. 1.

Практичне застосування методів машинного навчання для автоматизації управління завданнями можна навести з використанням бібліотеки Sklearn [6], яка надає широкий спектр інструментів для обробки тексту, побудови моделей та кластеризації. На першому етапі необхідно створити список описів завдань та відповідний список фактичного часу виконання завдань у днях. Ці дані слугуватимуть базовим набором для тренування та тестування моделей. Код наведено на рис. 2.

Далі текстові описи завдань потрібно перетворити у числові вектори за допомогою методу TF-IDF. Це дозволяє виділити ключові слова з описів і розрахувати їхню вагомність, що є критичним для подальшого аналізу. Код наведено на рис. 3.

```
# Приклад даних (опис завдання і фактичний час виконання у днях)
tasks = [
    "Створення логіну та авторизації на PHP",
    "Реалізація системи пошуку на сайті",
    "Впровадження нового дизайну на сторінці продукту",
    "Налаштування бази даних та інтеграція з API",
    "Використання docker",
    "Використання apache",
    "Підняття веб серверу на вдс",
    "Перезапуск серверу",
]
times = [1, 2, 3, 4, 1, 6, 1, 10] # фактичний час виконання у днях
```

Рис. 2. Код створення списку завдань і часу їх виконання у днях

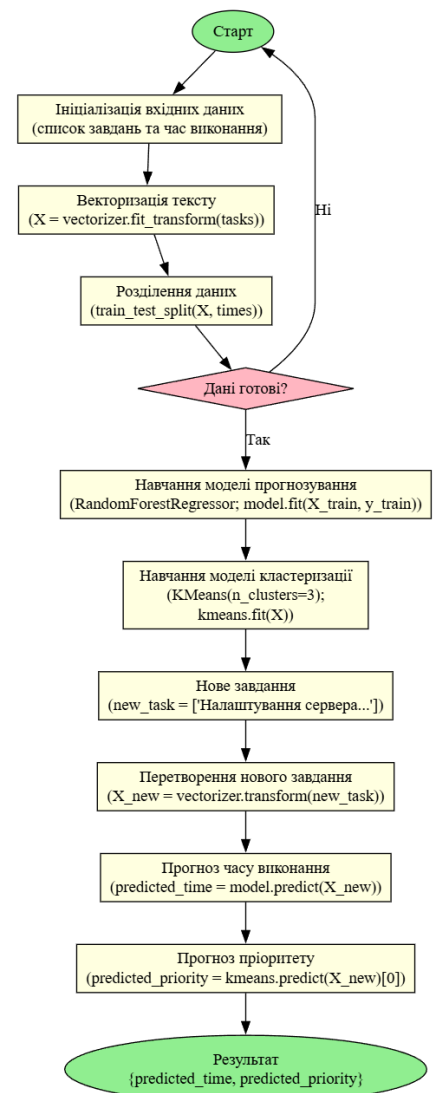


Рис. 1. Блок-схема роботи алгоритму моделі

```
# Перетворення тексту на числові ознаки з допомогою TF-IDF
vectorizer = TfidfVectorizer()
X = vectorizer.fit_transform(tasks)
```

Рис. 3. Код перетворення тексту завдань у числові ознаки методом TF-IDF

Наступним кроком є навчання моделі Random Forest Regressor для прогнозування часу виконання завдань. Для цього дані потрібно розділити на тренувальну та тестову вибірки, щоб забезпечити точність і валідацію моделі. Код алгоритму тренування моделі наведено на рис. 4.

```
# Розділення даних на тренувальну і тестову вибірки для RandomForest
X_train, X_test, y_train, y_test = train_test_split(X, times, test_size=0.2, random_state=42)

# Використання RandomForest для передбачення часу виконання
model = RandomForestRegressor()
model.fit(X_train, y_train)
```

Рис. 4. Код тренування моделі Random Forest Regressor

Для визначення пріоритетності завдань використовується метод кластеризації K-means. Він дозволяє групувати завдання за рівнями важливості та складності, сприяючи більш ефективному плануванню ресурсів. Розрахунки, що виконуються на цьому етапі, представлені на рис. 5.

```
# Використання KMeans для кластеризації завдань (пріоритетність)
num_clusters = 3 # Кількість кластерів для визначення рівнів пріоритету
kmeans = KMeans(n_clusters=num_clusters)
kmeans.fit(X) # Навчання на всіх завданнях для визначення кластерів
```

Рис. 5. Код кластеризації завдань методом K-means

Для перевірки працездатності алгоритму виконано тестування на наборі завдань із заданими текстовими описами та історичними даними щодо часу їх виконання з тестовим завданням «Налаштування сервера і реалізація безпеки підняття веб сервера та перезапуск серверу». Код тестування алгоритму наведено на рис. 6.

```
# Передбачення часу виконання нового завдання
new_task = ["Налаштування сервера і реалізація безпеки підняття веб сервера та перезапуск серверу"]
X_new = vectorizer.transform(new_task)
predicted_time = model.predict(X_new)
predicted_priority = kmeans.predict(X_new)[0]
print(f"Очікуваний час виконання завдання: {predicted_time[0]:.2f} днів")
print(f"Рівень пріоритету: {predicted_priority + 1}")
```

Рис. 6. Код передбачення часу виконання та пріоритету нового завдання

Для тестування ефективності алгоритмів машинного навчання було використано створений набір даних із завдань та часу їх виконання. Варто зазначити, що дані для моделі було сформовано навмання та не відображають реальних сценаріїв, а слугують виключно для демонстрації теоретичного застосування методів TF-IDF, Random Forest Regressor і K-means кластеризації. Метою тестування було оцінити, наскільки алгоритми здатні передбачати час виконання завдань і класифікувати їх за рівнем пріоритетності. На рис. 7 представлені результати виконання алгоритму для прогнозування часу виконання завдання та визначення його пріоритету.

```
Очікуваний час виконання завдання: 5.83 днів
Рівень пріоритету: 2
```

Рис. 7. Результати роботи алгоритмів Random Forest Regressor та K-means кластеризації

Ці значення демонструють, що модель успішно передбачає часові показники на основі текстового опису завдань та визначає кластер, до якого належить завдання, для оцінки його важливості.

Для оцінки продуктивності та точності моделі може бути використано метрики, такі як середньоквадратична похибка (MSE) для Random Forest Regressor та інерція для K-means кластеризації. Проте, у даному випадку мета полягала не у створенні виробничої системи, а у демонстрації базової концепції застосування методів машинного навчання до задач автоматизації управління завданнями.

Застосовані алгоритми показали свою ефективність, проте для впровадження у реальних сценаріях необхідно створити більш обширний і релевантний датасет, який враховуватиме специфічні характеристики завдань для кожної команди.

Використання цих методів машинного навчання дозволяє створювати інтелектуальні системи управління завданнями, які не лише оптимізують процеси планування, але й забезпечують більш високу точність у прогнозуванні та класифікації завдань.

Подальші дослідження у цьому напрямку можуть зосереджуватись на розробці гібридних моделей, які поєднують різні підходи, адаптовані до специфічних потреб команд і галузей.

Висновки

Сучасні підходи до автоматизації управління завданнями для малих команд з використанням методів машинного навчання демонструють значний потенціал у підвищенні ефективності робочих процесів та зниженні навантаження на учасників команд. Запропонований підхід дозволяє не лише описати перспективні технології, але й конкретно вирішити проблему автоматизованого прогнозування часу виконання завдань та визначення їх пріоритетів. Розвиток таких систем надає нові можливості для малих команд, дозволяючи їм зосереджуватися на більш важливих аспектах роботи, таких як стратегічне планування та виконання складних завдань.

Застосування TF-IDF дозволяє ефективно обробляти текстову інформацію, що спрощує аналіз завдань, визначення їхніх ключових аспектів і подальше групування за подібністю. Random Forest Regressor, як метод прогнозування часу виконання, дає змогу зменшити невизначеність при плануванні завдань, враховуючи різні фактори. Метод кластеризації K-means сприяє правильному визначенню пріоритетів завдань, допомагаючи оптимізувати розподіл ресурсів у команді.

Водночас подальші дослідження повинні спрямовуватись на удосконалення цих методів, зокрема через комбінування машинного навчання з іншими передовими технологіями.

У результаті, автоматизація управління завданнями з використанням машинного навчання має великий потенціал для покращення продуктивності, ефективності та адаптивності малих команд у різних сферах діяльності.

Список літератури

1. Гарбич-Мошора О., Столярчук І., Лешко Р., Британ В., Гольський В., Біліонок Я. Розробка мобільного додатку для трекінгу та контролю роботи над завданнями. *Наука і техніка сьогодні*. 2024. № 4(32). С. 882-893.
2. Orkhan Masimov, Yashar Hajiyeve Orkhan Masimov, Yashar Hajiyeve. *Integration of artificial intelligence (ai) in task management systems PAHTEI-Proceedings of Azerbaijan High Technical Educational Institutions*. 2024. 41, № 06. P. 501–509.
3. Celestine Iwendi, Suresh Ponnann, Revathi Munirathinam, Kathiravan Srinivasan and Chuan-Yu Chang. *An Efficient and Unique TF/IDF Algorithmic Model-Based Data Analysis for Handling Applications with Big Data Streaming. Electronics*. 2019. 8, № 11. P. 1331.
4. Muhammad Saad Sheikh, Rabia Noor Enam, Rehan Inam Qureshi. *Machine learning-driven task scheduling with dynamic K-means based clustering algorithm using fuzzy logic in FOG environment. Frontiers in Computer Science*. 2023. Vol.5. P. 1293209.
5. André O. Sousa; Daniel T. Veloso; Henrique M. Gonçalves; João Pascoal Faria; João Mendes-Moreira; Ricardo Graça. *Applying Machine Learning to Estimate the Effort and Duration of Individual Tasks in Software Projects. in IEEE Access*, 2023. Vol. 11. P. 89933-89946.
6. *Scikit-learn: machine learning in Python – scikit-learn 1.5.2 documentation*. URL: <https://scikit-learn.org/stable/>

I. Kolomiets, I. Zamrii, B. Kalyniuk, Yu. Bazhan, T. Dovzhenko

MODERN APPROACHES TO THE AUTOMATION OF TASK MANAGEMENT FOR SMALL TEAMS USING MACHINE LEARNING METHODS

Text of annotation translation – Modern project and task management technologies offer numerous tools that provide basic functions for organizing team work. However, most of these solutions are focused on large companies, which often makes them too complex for small teams. The lack of tools for automated forecasting of task completion times and determining their priorities limits the effectiveness of such systems. Small teams need flexible solutions that reduce the burden on work organization, automate routine tasks, and promote rational resource allocation. Therefore, the problem is the lack of automated tools for accurately predicting task completion times and determining their priority, which leads to inefficient resource allocation and work delays.

The main approaches to task management automation are analyzed, in particular, methods that can be used to forecast task completion times and determine priorities using machine learning. Special attention is paid to the TF-IDF, Random Forest Regressor, and K-means algorithms. TF-IDF allows for efficient processing of text descriptions of tasks, converting them into numerical features, which provides an analytical basis for the operation of machine learning models. Random Forest Regressor is used to accurately predict task completion times, which helps teams plan the workflow. The K-means algorithm is used to cluster tasks by their importance and complexity, providing automatic prioritization.

Popular tools such as Trello, Asana, Jira, Wrike, provide basic functionality for task management, but do not use machine learning methods for automation. The considered approaches can be integrated into existing systems or implemented as a separate solution for small teams seeking to increase productivity without significant costs for complex platforms.

The results of the study emphasize the importance of using machine learning to automate task management. This approach allows you to reduce dependence on the human factor, reduce the time for organizing tasks and optimize the planning process. In addition, it helps to increase the efficiency of teamwork, which is especially important for small teams working in conditions of limited resources.

Keywords: task management automation; machine learning; time forecasting; clustering; TF-IDF; Random Forest Regressor; K-means.