

Данило Клебан, Ігор Капран*

¹ НЛТУ України, Львів, Україна, kleban.d@nltu.lviv.ua

² НЛТУ України, Львів, Україна. ORCID: 0000-0001-9666-9531,
kapran@nltu.edu.ua

Система аналізу та оцінки популярності технологій і фреймворків

***Анотація.** У роботі представлені результати дослідження популярності технологій шляхом аналізу даних із GitHub. Була розроблена система GitHub Trend Analyzer, яка використовує нові метрики – Star H-Index та Fork H-Index – для оцінки активності та актуальності фреймворків. На прикладі RPC-фреймворків (gRPC, JSON-RPC, SOAP, Thrift, Dubbo) продемонстровано можливості системи, щодо порівняння рівня використання різних технологій. Отримані результати дають змогу об’єктивніше оцінювати тенденції розвитку програмного забезпечення.*

***Ключові слова** – GitHub, програмне забезпечення, індекс, API, Google Trends.*

Вступ. У сучасному світі програмування кількість технологій, фреймворків та інструментів невпинно зростає, що ускладнює вибір оптимальних рішень для розробників. Визначити, які технології справді актуальні, а які поступово втрачають популярність, – завдання, що потребує об’єктивного підходу. Традиційні методи, зокрема аналіз пошукових запитів у Google Trends, не завжди відображають реальну активність спільноти розробників [1]. Тому був запропонований альтернативний шлях – розроблення інструменту для аналізу та оцінки популярності технологій і фреймворків. Такий підхід дозволяє об’єктивно простежити тенденції розвитку програмних рішень та визначити технології, що мають найбільший потенціал [2].

Математична модель. Для кожної технології k формується множина репозиторіїв:

$$R_k = \{r_1, r_2, \dots, r_{\{n_k\}}\} \quad (1)$$

де r_i – окремий репозиторій.

Для кожного репозиторію відомі такі показники:

- s_i – кількість зірок (stars);

- f_i – кількість форків (forks);
- p_i – кількість pull-request;
- u_i – дата останнього оновлення;
- t – поточний момент часу.

Щоб зменшити вплив застарілих репозиторіїв, вводиться коефіцієнт актуальності:

$$\omega_i = \exp(-\lambda(T - u_i)) \quad (2)$$

де $\lambda > 0$ – параметр загасання.

Чим більша різниця між поточним часом T і останнім оновленням u_i , тим менше значення ω_i , тобто тим менший внесок цього репозиторію. З урахуванням коефіцієнта актуальності обчислюються зважені величини:

$$s'_i = \omega_i \cdot s_i, f'_i = \omega_i \cdot f_i, a_i = \omega_i \cdot (\alpha_p p_i) \quad (3)$$

де α_p – ваговий коефіцієнт для активності (pull-request).

Ці показники відображають реальний вплив кожного репозиторію на сучасному етапі розвитку технології. Для оцінки рівня загальної популярності використовується зважений індекс Гірша:

$$\{H\}_k^{\{(s)\}} = \max \{ h : s'_{\{(h)\}} \geq h \} \quad (4)$$

де $s'_{\{(h)\}}$ – кількість зірок h -го репозиторію після сортування за спаданням.

Аналогічно визначається зважений індекс за форками $H_k^{(f)}$. Цей показник вказує на те, скільки репозиторіїв технології мають стабільно високу популярність. Крім індексу Гірша, обчислюються додаткові агреговані характеристики:

$$S_k = \sum_{i=1}^{n_k} s'_i, A_k = \frac{1}{n_k} \sum_{i=1}^{n_k} a_i \quad (5)$$

де S_k – сумарна кількість зважених зірок, A_k – середня зважена активність розробників.

Щоб об'єднати всі показники в одну шкалу, кожна метрика нормалізується за формулою:

$$\text{norm}(X_k) = \frac{X_k - \min_j X_j}{\max_j X_j - \min_j X_j} \quad (6)$$

Програмна реалізація. Обчислення всіх цих h -індексів – заняття досить трудомістке. GitHub має чудове API, яке дозволяє взаємодіяти з платформою як

через REST, так і через GraphQL [3]. Тому було вирішено створити невеликий вебзастосунок.

Застосунок дозволяє:

- Обчислювати Н-індекси. Введіть пошуковий запит – і програма розрахує $h_{\star}$ та $h_{\#}$ для репозиторіїв, що відповідають цьому запиту.
- Порівнювати кілька технологій. Введіть кілька запитів, розділених крапкою з комою, і отримаєте порівняльну таблицю всіх метрик.
- Аналізувати тренди. Окрім h-індексів, застосунок дозволяє візуалізувати динаміку активності на GitHub у часі – кількість репозиторіїв, pull request та issues для кожного пошукового запиту.
- Експортувати дані. Отримані результати можна зберегти у форматі CSV для подальшого аналізу.

Search Term	Star H-Index ↓	Fork H-Index	Total Repos	PRs	Discussions	Top Starred
gRPC	57	24	28,115	601,096	5,026	conductor-oss/conductor (20381 ★)
JSON-RPC	23	11	859	99,025	2,173	unum-cloud/ucall (1200 ★)
Dubbo	19	11	983	10,163	167	Chanzi-keji/chanzi (367 ★)
SOAP	13	11	7,752	12,564	783	FalconForceTeam/SOAPHound (722 ★)
Thrift	12	9	2,523	14,494	320	HuaGouFdog/Fdog-Kit (381 ★)

Рисунок 1. Таблиця з результатами аналізу популярності RPC-технологій.

Тепер з допомогою даного інструменту, подивимося детальніше на історію використання цих технологій на GitHub використовуючи аналіз трендів (рис. 2).

Система дозволяє застосувати методику h-індексів [5] та аналіз трендів не тільки до RPC, але і до будь-яких інших технологій, фреймворків або концепцій, які можна знайти на GitHub. Наприклад, можна вловити тенденції нового часу ось у такому графіку приведеному на рис. 3.

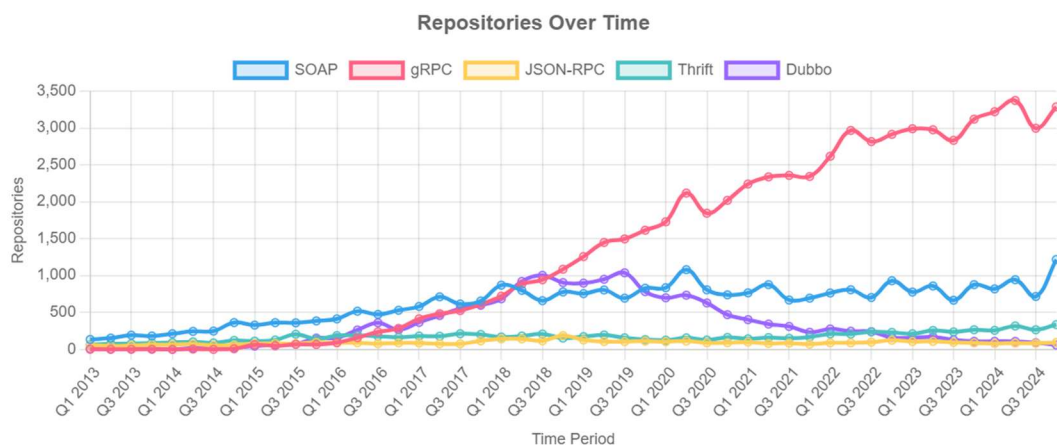


Рисунок 2. Тренд нових репозиторіїв (по кварталах, з 2013 до 2024 років)

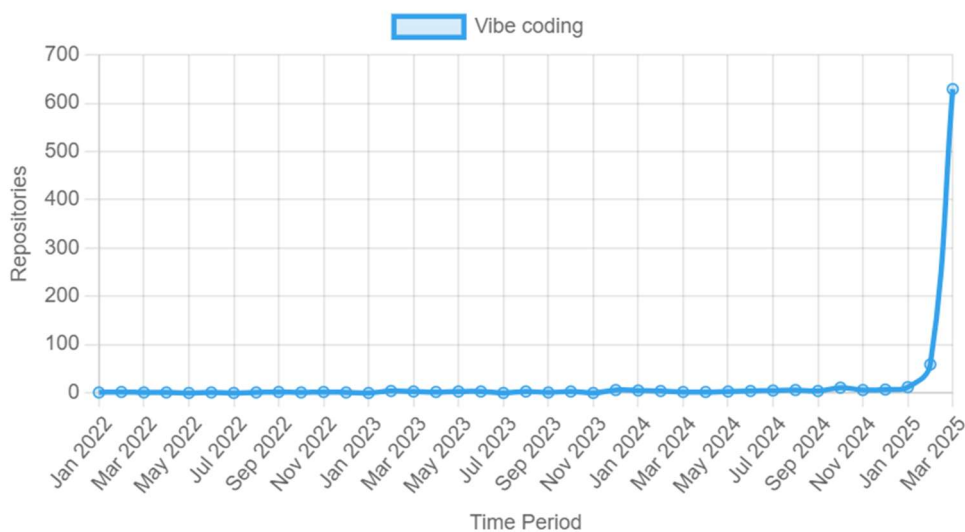


Рисунок 3. Vibe Coding

Search Terms

Analyze

Trends Analysis

☒ Show Trends

Start Year

2022

End Year

2025

Time Granularity

Yearly

Metric

Repository Count

H-Index Analysis

☐ Calculate H-Index

Рисунок 4. Інтерфейс GitHub Trend Analyzer

Висновок. В даній роботі представлено вебінструмент GitHub Trend Analyzer, який дозволяє об'єктивно оцінювати популярність технологій на основі відкритих даних GitHub. Використання метрик Star H-Index та Fork H-Index дає змогу порівнювати технології між собою та відстежувати їхню динаміку [4]. На прикладі RPC-фреймворків показано, що запропонований підхід ефективно відображає реальні тенденції розвитку технологій і може бути застосований до будь-яких галузей програмування.

Список використаних літературних джерел

- [1] Hirsch, J. E. An index to quantify an individual's scientific research output. Proceedings of the National Academy of Sciences, 2005.
- [2] Schreiber, Michael. Twenty Hirsch index variants and other indicators giving more or less preference to highly cited papers. arXiv preprint, 2010.
- [3] GitHub. GitHub REST API documentation. GitHub Docs. docs.github.com
- [4] GitHub. REST API – versioning & best practices. GitHub Docs. docs.github.com
- [5] Wikipedia contributors. H-index. Wikipedia. en.wikipedia.org

System for analysis and evaluation of the popularity of technologies and frameworks

Danilo Kleban, Ihor Kapran

This paper presents the results of a study on the popularity of technologies through the analysis of GitHub data. A system called GitHub Trend Analyzer was developed, which employs new metrics – Star H-Index and Fork H-Index – to assess the activity and relevance of frameworks. Using RPC frameworks (gRPC, JSON-RPC, SOAP, Thrift, Dubbo) as examples, the system's capabilities for comparing the usage levels of different technologies are demonstrated. The obtained results provide a more objective evaluation of software development trends.