



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ
УНІВЕРСИТЕТ ІМЕНІ СЕМЕНА КУЗНЕЦЯ

КАФЕДРА КІБЕРБЕЗПЕКИ
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

МАТЕРІАЛИ

XV-ої МІЖНАРОДНОЇ НАУКОВО-ПРАКТИЧНОЇ КОНФЕРЕНЦІЇ
«FREE AND OPEN SOURCE SOFTWARE»



Дякуємо за підтримку



IDCMP
IDEA DEVELOPMENT CONSULTING MANAGEMENT



13-14 лютого 2024 р.
м. Харків

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ СЕМЕНА КУЗНЕЦЯ

КАФЕДРА КІБЕРБЕЗПЕКИ
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

МАТЕРІАЛИ

XV-ої МІЖНАРОДНОЇ НАУКОВО-ПРАКТИЧНОЇ КОНФЕРЕНЦІЇ

«FREE AND OPEN SOURCE SOFTWARE»

13-14 лютого 2024 р.

ХАРКІВ 2024

УДК 004
БК 32.973.202

Матеріали XV-ої Міжнародної науково-практичної конференції «Free and Open Source Software», Харків, 13-14 лютого 2024 р. – Харків: Харківський національний економічний університет імені Семена Кузнеця, 2024. – 148 с.

Представлено матеріали пленарних та секційних засідань XV-ої Міжнародної науково-практичної конференції «Free and Open Source Software». Обговорено основні проблеми, науково-технічні досягнення, впровадження і досвід використання сучасних технологій в області безкоштовних програмних продуктів, а також з відкритим вихідним кодом. Висвітлено основні питання безкоштовного прикладного, серверного програмного забезпечення та прикладного програмного забезпечення з відкритим вихідним кодом, безкоштовних сервісів, в тому числі в контексті кібербезпеки, ліцензування та правові аспекти використання безкоштовного програмного забезпечення. Для фахівців науково-дослідних, комерційних організацій, аспірантів та студентів.

Редакційна колегія:

Старкова О.В. – голова, д.т.н.;

Міхєєв І.А. – к.т.н.;

Відповідальний за випуск:

Старкова О.В.

Роботи надруковані з авторських оригіналів, що надані оргкомітету, за авторської редакції.

Електронний варіант матеріалів конференції доступний на сайті конференції:

<https://foss.kn-it.info/>

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ПРОГНОЗУВАННЯ ВАРТОСТІ ТУРИСТИЧНИХ ПОДОРОЖЕЙ <i>Тарасов Я.К., Льовкін В.М.</i>	99
ОГЛЯД МОЖЛИВОСТЕЙ ЗАЛУЧЕННЯ ШТУЧНОГО ІНТЕЛЕКТУ В БУДІВЕЛЬНІЙ ГАЛУЗІ <i>Тоотс Р.В., Шаповалова О.О.</i>	100
ОГЛЯД БІБЛІОТЕКИ PUPPETEER ДЛЯ ВИРІШЕННЯ ЗАДАЧІ DATA SCRAPING З ЦІЛЮ ЗБОРУ ЦІННИХ ДАНИХ <i>Топчій М.А., Яковлева О.В.</i>	101
ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ОБРОБЛЕННЯ ПРИРОДНОЇ МОВИ ДЛЯ ВИЗНАЧЕННЯ ТЕМ СТАТЕЙ БЛОГУ <i>Харитонов Д.О., Льовкін В.М.</i>	102
SPRING BOOT: ЛЕГКІСТЬ ТА ПОТУЖНІСТЬ РОЗРОБКИ НА JAVA <i>Широкопад К.А., Яковлева О.В.</i>	103
ОГЛЯД ІНСТРУМЕНТІВ ДЛЯ ГНУЧКОГО УПРАВЛІННЯ ПРОЄКТАМИ <i>Шишкін М.С., Назаров Д.Л., Старкова О.В.</i>	105

СЕКЦІЯ 3

R-STUDIO CAPABILITIES FOR BIG DATA ANALYSIS <i>Dolgova N.G.</i>	108
IMPROVING DATA QUALITY FOR A THREE-FACTOR NONLINEAR REGRESSION MODEL FOR ESTIMATING THE SIZE OF WEB APPLICATIONS CREATED USING THE REACT FRAMEWORK <i>Makarova L., Hashko D.</i>	109
FREE EDUCATION SERVICES WITH INTERACTIVE CONTENT <i>Mikheiev I.</i>	111
ANALYSIS OF FREE INSTRUMENTS FOR GENERATING AND MANIPULATING STATISTICAL DATA <i>Zhuravka A.V., Ivanov A., Laetitia Villeneuve</i>	112
FREE TECHNOLOGIES FOR DEVELOPMENT OF USER INTERFACES <i>Zhuravka A.V., Snihurov A.V., Ivanov A.</i>	113

IMPROVING DATA QUALITY FOR A THREE-FACTOR NONLINEAR REGRESSION MODEL FOR ESTIMATING THE SIZE OF WEB APPLICATIONS CREATED USING THE REACT FRAMEWORK

Makarova L., Hashko D.

E-mail: lidiia.makarova@nuos.edu.ua, dimagashko@gmail.com

Mykolaiv, Admiral Makarov National University of Shipbuilding

Software development is a complex system of different steps and stages from initial planning, analysis and design to development, testing and maintenance. Estimating labor intensity at each step, especially in development, is crucial for accurately allocating resources, setting realistic timelines, and managing workloads effectively. There are many ways to estimate the labor intensity of development, but most of them are based on the size of the application, for example, COCOMO II (Constructive Cost Model II) [1].

But to effectively apply these methods, first we need to estimate the size of the application being developed. One of the most accurate and powerful approaches for this purpose involves using a regression model [2] specifically designed for estimating the size of such applications.

In conference thesis of VI All-Ukrainian Scientific and Practical Internet Conference of Students, Postgraduates, and Young Scientists "Modern Information Systems and Technologies" (Kherson, 30.11.2023) we have presented our three-factor nonlinear regression model for estimating the size of web applications created using the React framework [3].

React is an open-source, front-end JavaScript framework for building web applications using component-based architecture [4]. It's one of the most popular frameworks in web development that allows developers to create scalable applications of any size with high performance.

The presented model is shown below:

$$Y = 10^{\varepsilon - 1.4302} X_1^{0.3568} X_2^{0.6256} X_3^{-0.1419}, \quad (1)$$

where: X_1 – the number of components,

X_2 – the average number of methods,

X_3 – reusability index (how often components are used in other components).

The model shows acceptable results on test data, but we have concerns about its quality metrics so we want to improve the quality of our model.

To measure the quality of the model we use such metrics as the coefficient of determination (R^2), the Mean Magnitude of Relative Error (MMRE) and $PRED$ (0.25). For the initial model these metrics are 0.7862, 0.3213 and 0.5526 respectively.

First, it's important to note that the main difficulties in building a model for estimating the size of web applications created using the React framework is that React and JavaScript itself allows significant variability in how to write components and the code in general, so it makes it harder to build a quality model compared to more strict languages and technologies.

One of the possible ways to improve the quality of our model is to use another type data transformation. Currently, we're using logarithmic transformation to normalize source data, but we could try to use other transformations like Johnson's or Box-Cox transformations and see how it would affect the quality of our model.

But before changing how we process our data, we need to make sure that the data is clean and representative, which allows us to improve the quality of our model and have more solid base for further improvements. Because if the data is inconsistent or lacks representativeness then the model's quality may not reach its full potential whatever methods we use to process the data.

There are two ways to improve the quality of our data – first we can rethink what type of metrics we use to build our model (use other metrics or add more) or we can improve the way we take this data, for example, by adding more restrictions for source projects so the data we get from them is more consistent, relevant and representative.

For now, we don't want to change the metrics we use, because all the current metrics can be retrieved from UML Class Diagram and so the model is clear and easy to use. Also, the metrics were chosen based on specifics of the React framework and JavaScript language so they should be representative enough, but by using other metrics or adding more we can increase complexity of our model and make it harder to use.

First, we decided to add more restrictions for source projects, so we get more consistent data from them and have more solid base for further improvements.

Initially we wanted to make our model more universal, so it fits for estimating the size of any web application created using the React framework and because of this for our data we used any open-source projects created using the React framework available on GitHub [5] with no any other requirements. But the problem here is that both React and JavaScript evolve rapidly and so over time developers are using different methodologies, styles of code, best practices, etc. So, three metrics to build a universal model for projects created using the React framework may not be enough to build a model with a higher quality. But at the same time there also might be no reason for this – the model is intended to be used to estimate the size of newly created applications and its more likely that they would develop their applications using the latest practices. So, including older projects in the data just makes the data inconsistent and less representative for real-life usage.

The other problem with initial data is that it includes projects of different types. Mostly it contains web applications, but also some libraries, small games and test applications from some tutorials. But all of them have their own specifics that can affect their size significantly.

We want to build our model for estimating the size of real-life web applications, so we should not use other types of projects and even test web applications for our data. What makes it harder to find enough open-source projects, is that most of real-life web applications are proprietary and are not available for public use. We could also just build our model for estimating the size of libraries instead, because they are usually open-source and there are mostly unlimited number of them available, but they can be harder to analyze and be less consistent.

So, considering the above, we decided to add the following requirements to our data:

- 1) project age: created less than 2 years ago;
- 2) last commit: less than 6 months ago;
- 3) project type: real-life web applications.

Then based on these new requirements we found a new set of projects for our source data, and rebuilt the model with updated data. We also calculated the quality metrics for updated model and all of them showed that the quality of the model increased. The coefficient of determination (R^2), the Mean Magnitude of Relative Error (*MMRE*) and *PRED* (0.25) of the updated model are 0.8132, 0.3012 and 0.6527 respectively.

So, by introducing more requirements for source data we actually improved the quality of our model. Now, with a more solid base, we can use other methods to improve it ever further.

References

- [1] Boehm B. Software Cost Estimation with Cocomo II. New Jersey, Prentice Hall. 2000. 544 p.
- [2] Douglas C. Montgomery, Elizabeth A. Peck, G. Geoffrey Vining. Introduction to Linear Regression Analysis. John Wiley & Sons, Inc., Hoboken, New Jersey. 2012. 872p.
- [3] Dmytro Hashko, Lidiya Makarova. A three-factor nonlinear regression model for estimating the size of web applications created using the React framework. Proceedings of the VI All-Ukrainian Scientific and Practical Internet Conference of Students, Postgraduates, and Young Scientists "Modern Information Systems and Technologies" (November 30, 2023, Khmelnytskyi, Kherson). Edited by Hryhorova A. Kherson: FOP Vyshemyrskyi V. Publishing House, 2023. Pp. 147-148.
- [4] React. The library for web and native user interfaces. [Electronic resource]. – Access mode: <https://reactjs.org/>
- [5] GitHub. [Electronic resource]. – Access mode: <https://github.com>.

**Матеріали XV-ої Міжнародної науково-практичної конференції
«FREE AND OPEN SOURCE SOFTWARE»**

Харківський національний економічний університет імені Семена Кузнеця

Відповідальний за випуск: Старкова О.В.

Редактор: Міхєєв І.А.

Затверджено засіданням кафедри кібербезпеки та інформаційних технологій
ХНЕУ імені С. Кузнеця
протокол № 11 від «16» лютого 2024 р.

Видавець і виготовлювач – ХНЕУ імені С. Кузнеця, 61166, м. Харків, просп.
Науки, 9-А
Свідоцтво про внесення суб'єкта видавничої справи до Державного реєстру
ДК № 4853 від 20.02.2015 р.