

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
МІЖНАРОДНИЙ ГУМАНІТАРНИЙ УНІВЕРСИТЕТ

МІЖНАРОДНА КОНФЕРЕНЦІЯ
**“Передові технології в інформаційно-
комунікаційній інженерії”**
(ATICE'2023)

Одеса, Україна, 18 липня 2023 р.

Матеріали конференції

Одеса

2023

INTERNATIONAL CONFERENCE
“Advanced Technology in Information and
Communication Engineering”
(ATICE’2023)

Odesa, Ukraine, July 18, 2023

Conference Proceedings

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
МІЖНАРОДНИЙ ГУМАНІТАРНИЙ УНІВЕРСИТЕТ**

**ПЕРЕДОВІ ТЕХНОЛОГІЇ
В ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНІЙ
ІНЖЕНЕРІЇ
МАТЕРІАЛИ КОНФЕРЕНЦІЇ**

Одеса, Україна, 17-20 липня 2023 р.

**MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
INTERNATIONAL HUMANITARIAN UNIVERSITY**

**International Conference “Advanced Technology
in Information and Communication Engineering”
(ATICE’2023)**

Odesa, Ukraine, July 17-20, 2023

Conference Proceedings

**Odesa
2023**

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
МІЖНАРОДНИЙ ГУМАНІТАРНИЙ УНІВЕРСИТЕТ**

**Міжнародна конференція «Передові технології
в інформаційно-комунікаційній інженерії»
(ПТІКІ'2023)**

Одеса, Україна, 17-20 липня 2023 р.

Матеріали конференції

**Одеса
2023**

International Conference “Advanced Technology in Information and Communication Engineering” (ATICE’2023): Conference Proceedings. Odesa : International humanitarian university, 2023, : Bondarenko M. O., 2023. – 102 p.

Міжнародна конференція «Передові технології в інформаційно-комунікаційній інженерії» (ATICE’2023). Матеріали конференції/Міжнародний гуманітарний університет, Одеса, 2023 : Бондаренко М.О., 2023. – 102 с.

ISBN 978-617-8327-06-4

Збірка містить праці Міжнародної конференції з інформаційних систем, технологій захисту інформації, використання сучасних інформаційних технологій в управлінні системами за різними галузями народного господарства.

Матеріали публікуються в авторській редакції.

ISBN 978-617-8327-06-4

ОРГАНІЗАЦІЙНИЙ КОМІТЕТ

Голова

КІВАЛОВ С.В. – д.т.н., проф., Міжнародний гуманітарний університет, м.Одеса, Україна.

Співголови

СТРЕЛКОВСЬКА І.В. – д.т.н., проф., Міжнародний гуманітарний університет, м.Одеса, Україна.

ГЛОБА Л.С. – д.т.н., проф., Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», м.Київ, Україна.

Члени організаційного комітету

БЕРКМАН Л.Н. – д.т.н., проф., Державний університет телекомунікацій, м.Київ, Україна.

КЛИМАШ М.М. – д.т.н., проф., Національний університет «Львівська політехніка», м.Львів, Україна.

ЛЕМЕШКО А. В. – д.т.н., проф., Харківський національний університет радіоелектроніки, м. Харків, Україна.

РОЗЕНВАССЕР Д.М. – к.т.н., доц., Міжнародний гуманітарний університет, м.Одеса, Україна.

РИХЛІК А. – к.т.н., Лодзький технічний університет, м.Лодзь, Польща

ЗМІСТ

СЕКЦІЯ 1. ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ІНФОРМАТИКА.....	11
Brylliantova Anastasiia. Software implementation of air quality indicator forecasting.....	11
Viktoriia Volodymyrivna Tkach. Comparative analysis of feature selection methods for heart disease diagnosis using machine learning algorithms.....	14
Шабатура Костянтин Вікторович. Вибір програмного забезпечення для організації дистанційного навчання при підготовці фахівців з інформаційних технологій.....	18
Стрелковська І.В., Соловська І.М., Снігур Н., Малюга В. Параметричні сплайни в 3D-моделюванні.....	22
Слатвінська Валерія Миколаївна. Аналіз використання змінних середовища в конфігураційних файлах APACHE2.....	26
Курюкін Ілля Сергійович, Баришовець Артем Олександрович. Віртуальна лабораторія для дослідження нейронних мереж.....	28
Прокоп Юлія Віталіївна, Клімішина Ірина Вікторівна. Використання GPT для автоматичного оцінювання стилю коду студентів.....	30
Andriy Luntovskyy. Integration aspects between advanced networking and modern AI tools.....	34
Філін Олег Васильович. Розрахунок в програмному комплексі ansys з використанням моделі імпортованої з «Autodesk 3Ds Max».....	42
СЕКЦІЯ 2. КІБЕРБЕЗПЕКА ТА КІБЕРПСИХОЛОГІЯ.....	45
Григор'єва Т.І., Йона Л.Г., Мазур Г.Д., Кравченко І.А. Захист конфіденційної інформації шляхом шифрування на основі тензорних методів.....	45
Щур Наталія Олександрівна. Огляд протоколів та сервісів наскрізного шифрування	49
Andrii Shpilkin. Psychology of cyberspace. Social media advantage for threat actors.....	51

Література

1. Найкращі інтернет-ресурси та віртуальні лабораторії для проведення дистанційних уроків. URL: <https://myronivka-osvita.gov.ua/najkraschi-internetresursi-ta-virtualni-laboratorii-dlya-provedennya-distancijnih-urokiv-10-16-41-25-03-2021/>
2. Бохан Ю.В., Форостовська Т.О. Віртуальний лабораторний практикум як засіб вивчення природничих дисциплін. Наукові записки Серія: Педагогічні науки. 2021. Вип. 194. С. 74–78.
3. Neural Network Developer. URL: <https://www.profguide.io/en/professions/neural-network-developer.html>

УДК 004.032.26:378.147

Прокоп Юлія Віталіївна
Національний університет «Одеська політехніка»
prokop.y.v@op.edu.ua
Клімішина Ірина Вікторівна
Міжнародний гуманітарний університет

ВИКОРИСТАННЯ GPT ДЛЯ АВТОМАТИЧНОГО ОЦІНЮВАННЯ СТИЛЮ КОДУ СТУДЕНТІВ

***Анотація.** Перевірка якості коду в студентських програмах і формування зворотного зв'язку з поясненням зазвичай вимагає великих витрат часу, тому автоматизація цього процесу є вельми актуальною. Доведено ефективність технології GPT для перевірки правил стилю коду, які не можуть бути перевірені автоматично стандартними засобами.*

Низька якість програмного коду не лише підвищує вартість підтримки і супроводу, а й може спричинити серйозні проблеми програмного забезпечення. Тому вміння писати якісний код є однією з найважливіших компетентностей майбутнього програміста, формування якої вимагає приділення значної уваги викладачів. Якщо від самого початку не привчити студентів писати якісний код, то у майбутньому на це доведеться витратити додатковий час та зусилля [1]. Пошук ефективних методів навчання студентів написання чистого коду є актуальною задачею для сучасних викладачів програмування.

Проблеми, пов'язані з формуванням у студентів IT-спеціальностей навичок створення якісного коду, в останнє десятиліття активно обговорюється викладачами по всьому світу. Перевірка, виявлення та опис порушених правил стилю коду в студентських програмах є нетривіальною задачею, яка вимагає великої кількості часу на її виконання. Отже, гостро стоїть потреба в автоматизації цього процесу.

Автоматичне оцінювання стилю коду було започатковано приблизно 20 років тому [2] і відтоді дослідники докладають зусиль для його вдосконалення. На думку авторів [3], найкращі результати дає така комбінація: викладач, який пояснює правила і помилки, автоматична перевірка AutoStyle й оцінка стилю зі зворотним зв'язком. Дисертаційне дослідження [4] показало, що найкращий результат у процесі навчання дає використання запропонованої А. Коєхлером автоматичної оцінки стилю кодування, яка супроводжується зворотним зв'язком. У роботі [5] описуються детектори дефектів якості коду, які можуть бути використані для автоматичної генерації цінного зворотного зв'язку для студентів. Автори [6] дійшли висновку, що наявних інструментів статичного аналізу загального

призначення недостатньо для оцінки всіх критеріїв стилю коду. Зокрема, стандартні засоби не можуть точно й чітко оцінити імена змінних та функцій.

Дослідники вивчають можливості та обмеження використання технології GPT в освітньому процесі [7; 8]. Автори дійшли висновку, що ця технологія має потенціал революціонізувати навчальний процес, уможливлючи персоналізоване та інтерактивне навчання, а також надаючи студентам доступ до величезної кількості знань та інформації. Однак при цьому використання технології GPT створює виклики й обмеження, включаючи занепокоєння щодо списування, конфіденційності даних та упередженості, які обов'язково треба враховувати та вживати відповідні заходи для їх подолання.

Метою роботи є дослідження можливості залучення технології GPT для перевірки окремих правил стилю кодування, автоматизація яких стандартними засобами обмежена.

Розглянемо найпростіший код (рис. 1). Програма пропонує користувачеві ввести ціну і кількість товару та обчислює його вартість. Цей код компілюється і виконується без помилок, але містить численні порушення правил стилю.

Спробуємо залучити штучний інтелект для виявлення цих помилок. Запитаємо в чат-бота: «Look at the code below. Check the coding style. Describe errors in the form: Line ... Error (its description)».

```
1  #include <stdio.h>
2  int main(void) {
3  float a, c; int b;
4  printf("Price: "); scanf("%f", &a);
5  printf("Amount: "); scanf("%d", &b);
6  c = a * b;
7  printf("You have to pay: %.2f\n", c);
8  return 0;
9  }
```

Рисунок 1 – Приклад коду мовою C для перевірки стилю

Для перевірки спочатку застосували Bing Chat від Microsoft, який використовує мовну модель GPT-4 від OpenAI. Його відповідь подана на рис. 2.

The code is functional and there are no syntax errors. However, there are some improvements that can be made to the coding style:

- Line 1: It is recommended to include a comment at the beginning of the code to describe its purpose.
- Line 3: It is recommended to declare each variable on a separate line for better readability.
- Line 4-7: It is recommended to use more descriptive variable names instead of single letters.
- Line 4-7: It is recommended to add spaces around operators for better readability.
- Line 4, 5, 7: It is recommended to place each statement on a separate line for better readability.

Рисунок 2 – Оцінка стилю коду від Bing Chat

Крім того, до відповіді Bing Chat додав виправлену версію коду – з правильними відступами та пробілами, відсутніми у коді, запропонованому для перевірки.

Отже, Bing Chat виявив такі порушення базових правил стилю коду: відсутні необхідні коментарі (рядок 1), оголошення кожної змінної не розташовані на окремому рядку (рядок 3), назви змінних не описують призначення цих змінних (рядки 4-7), відсутність пробілів навколо операторів (рядки 4-7) та відступів на початку рядків, а також розташування кількох команд в одному рядку, що знижує читабельність коду (рядки 4, 5, 7).

Аналогічне питання щодо перевірки стилю коду також було задано чат-боту ChatGPT (chat.openai.com). Однак він виявив менше помилок, зокрема не вказав на недостатньо зрозумілі назви змінних. Тому надалі використовували саме Bing Chat як такий, що краще задовольняє потребам і виконує якіснішу перевірку.

Слід зазначити, що перевірку частини з виявлених порушень легко автоматизувати стандартними засобами. Зокрема, це стосується коментарів, розташування оголошень змінних та команд в окремих рядках, правильності відступів і пробілів. Однак встановити автоматично, чи відповідають правилам назви змінних, зазвичай доволі складно. Водночас вимога до надання змінним та функціям зрозумілих назв є однією з базових вимог до якісного коду.

Розглянемо інший фрагмент коду (рис. 3). Цей код теж компілюється та виконується без помилок, але містить порушення деяких правил стилю коду. Зокрема, назва функції `func` не вказує на призначення цієї функції, як і назви змінних `a` та `s`, які не описують їхнього призначення. Запропонуємо штучному інтелекту оцінити якість назв змінних і функції у цьому коді.

```
1  #include <stdio.h>
2  #include <math.h>
3  double func(float *a, int n) {
4      float s=0;
5      for (int i=0; i<5; ++i)
6          s+=a[i];
7      return s;
8  }
9
10 int main() {
11     float a[5], s;
12     int i;
13     printf("Input double array:\n");
14     for (i=0; i<5; ++i)
15         scanf("%f", &a[i]);
16     s = func(a, 5);
17     printf("Sum of array elements = %.2f",s);
18     return 0;
19 }
```

Рисунок 3 – Приклад коду мовою C для перевірки назв змінних та функції

Відповідь Bing Chat подана на рис. 4. Він встановив, що назви змінних `a`, `s` та функції `func` не описують їхнє призначення, що порушує правила стилю коду. Це повною мірою збігається з очікуваним результатом.

Here is my evaluation of the names of variables and function in the provided code:

- Line 3: The function name `func` is not descriptive. It is recommended to use a more descriptive name that reflects the purpose of the function.
- Line 4: The variable name `s` is not descriptive. It is recommended to use a more descriptive name that reflects the purpose of the variable.
- Line 11: The variable name `a` is not descriptive. It is recommended to use a more descriptive name that reflects the purpose of the variable.
- Line 11: The variable name `s` is not descriptive. It is recommended to use a more descriptive name that reflects the purpose of the variable.

Рисунок 4 – Оцінка назв змінних та функцій Bing Chat

Змінимо незрозумілі назви. Функцію `func` назвемо `sumElements`, змінну `a` назвемо `array`, а змінній `s` в обох функціях дамо назву `sum`. Після цього знову відправимо код для перевірки. Цього разу відповідь штучного інтелекту є схвальною (рис. 5). Bing Chat встановив призначення функції та змінних і зазначив, що імена, які використовуються в коді, пояснюють це призначення.

The names of the variables and function in the provided code are more descriptive and reflect their purpose. The function name `SumElements` clearly describes its purpose of summing the elements of an array. The variable names `array` and `sum` are also more descriptive and reflect their purpose.

Рисунок 5. Схвальна оцінка назв змінних та функцій штучним інтелектом

Отже Bing Chat може оцінювати якість коду, зокрема правильність назв змінних і функцій, що майже неможливо автоматизувати стандартними засобами. Водночас є певне обмеження. Розмір тексту, який надсилається в чат, має ліміт 4000 символів і може бути меншим за розмір коду для перевірки. З боку користувача це обмеження можна обійти, збільшивши значення ліміту в інспекторі браузера. Проте, якщо очікуваний результат – це виправлена програма, то відповідь чат-бота в будь-якому разі обмежиться лише чотирма тисячами знаків.

Висновки

Технологію GPT можна використовувати для перевірки стилю кодування в студентських програмах. Особливе значення це має для тих правил, автоматизувати перевірку яких стандартними засобами складно або неможливо. Зокрема, технологію GPT можна використовувати для перевірки правильності назв змінних і функцій. Завдяки штучному інтелекту можна скоротити час, який витрачають викладачі на перевірку якості студентського коду та створення коментарів до помилок, а також підвищити ефективність такої перевірки.

Література

1. Keuning, H., Heeren, B., Jeuring, J.: Code quality issues in student programs. In: 22nd ACM Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE'2017, pp. 110–115. Bologna, Italy (2017). doi: 10.1145/3059009.3059061.
2. Ala-Mutka, K., Uimonen, T., Jarvinen, H.: Supporting students in C++ programming courses with automatic program style assessment. Journal of Information Technology Education: Research, vol. 3(1), pp. 245-262 (2004). doi: 10.28945/300

3. Wiese, E., Yen, M., Chen, A., Santos, L., Fox, A. 2017. Teaching Students to Recognize and Implement Good Coding Style. In: 4th ACM Conference on Learning @ Scale, L@S '17, pp. 41–50. [file:///Users/yulia/CTU/Coding%20Style/. New York, NY, USA \(2017\). doi: 10.1145/3051457.3051469](https://doi.org/10.1145/3051457.3051469)
4. Koehler, A.: A Methodology for Teaching from Student Errors in Computer Science Education. A Dissertation submitted in partial satisfaction of the requirements for the degree of Doctor of Philosophy in Computer Science by University of California (2020), https://escholarship.org/content/qt9x472353/qt9x472353_noSplash_24e49bb14b8e5c530578c32c02186ddd.pdf, last accessed 2023/06/21.
5. Effenberger, T., Pelánek, R.: Code Quality Defects across Introductory Programming Topics. In: 53rd ACM Technical Symposium on Computer Science Education, SIGCSE'2022, vol. 1, pp. 941–947. New York, NY, USA (2022). doi: 10.1145/3478431.3499415
6. Perretta, J.S., Weimer, W., DeOrio, A.: Human vs. Automated Coding Style Grading in Computing Education. In: ASEE Annual Conference & Exposition, pp.1–13 (2019). doi: 10.18260/1-2--32906, <https://peer.asee.org/32906>
7. Božić, Velibor. (2023). Chat GPT and education. 10.13140/RG.2.2.18837.40168.
8. Rahman, Md & Watanobe, Yutaka. (2023). ChatGPT for Education and Research: Opportunities, Threats, and Strategies. 10.20944/preprints202303.0473.v1.

UDC 519.6:658.5; 378; 37.037; 658

Andriy Luntovskyy, Prof. Dr. habil.
BA Dresden University of Technology, Saxon Study Academy
Andriy.Luntovskyy@ba-dresden.de

INTEGRATION ASPECTS BETWEEN ADVANCED NETWORKING AND MODERN AI TOOLS

Abstract. *This work is aimed to explore the potential of advanced network technologies to support Artificial Intelligence (AI) applications and vice versa. A duality slogan acts herewith as follows: “Networks meet AI as well as AI meets Networking!”.*

In 1950, as AI was defined for the first time, AI should solve the classes of problems, per definition, which were previously only intended for humans. However, the term remained controversial till nowadays. It is still unclear what exactly "intelligence" as a term encompasses – and to what extent it requires its own separate "consciousness".

ML is an important part of AI. With the big variety of AI approaches, there the knowledge is generated from large amounts of data. As a next topic, for instance, by using photos, AI assists to learn what a tree or car looks like. However, some experts do not yet see herewith intelligent behavior in this approach, called Computer Vision or Pattern Recognition.

The classical AI theory approaches include furthermore, Neural Networks (NN), Machine Learning (ML, refer to Fig. 1), Fuzzy Logic, Deep Learning (DL), Knowledge Bases, Ontologies, Image Recognition, Computer Vision, List Processing, Logical, and Rule-Based Output and Reasoning, Bayes Probability Calculation as well as modern Generative Large Language Models (LLMs), refer to Fig. 2. Furthermore, Genetic Algorithms (GA) and Computational Intelligence can belong to AI aspects too. GA operators include mutation, crossover, and selection, which used to support and generate high-quality solutions. They are metaheuristics which map natural genetic processes to achieve Computational Intelligence.