

## СИСТЕМА АВТОМАТИЗАЦІЇ НАПИСАННЯ ПРОГРАМНОГО КОДУ

*Вінницький Національний Технічний Університет*

### **Анотація**

В роботі проведений загальний опис системи зі штучним інтелектом. Проаналізовані можливості та перспективи розвитку подібних систем.

**Ключові слова:** *Роботизована система, штучний інтелект, програмний код, перспективи розвитку*

### **Annotation**

In the work the general description of the system with artificial intelligence is made. Possibilities and prospects of development of such systems are analyzed

**Keywords:** *Robotic system, artificial intelligence, program code, development prospects.*

### **Вступ**

Ще кілька десятиліть тому розвиток технологій штучного інтелекту гальмувала відсутність впевненості в кінцевому продукті. На це впливало чимало чинників: надмірна вартість машинного часу, вельми скромні обчислювальні ресурси, обмеженість мов програмування, громіздкість елементної бази тощо. У 1970-80-х роках процес взагалі майже зупинився на фоні фактично повного скорочення належного фінансування[1].

Однак, завдяки революційним розробкам у сфері напівпровідникової промисловості відбувся прорив у технологіях зберігання та обробки інформації і, як наслідок, – початок відродження епохи розумних машин припав на 1990-ті роки: з появою обмежених систем машинного навчання. А 2000-і роки ознаменували вже зовсім нову епоху розвитку систем штучного інтелекту.

Пристрої, попередньо запрограмовані для найпростіших міркувань, породили ранні платформи для створення цілих експертних і кваліфікованих прогностичних систем. І, не дивлячись на те, що на початкових етапах роботи з такими системами вчені зіштовхнулися з низкою проблем, які, на перший погляд, було неможливо вирішити, – результати численних досліджень принесли свої плоди.

Більшість прикладів використання штучного інтелекту, відомі сьогодні, – від комп'ютерів, що грають у шахи, до автономних систем, які все ще залежать від людського фактору і потребують глибокого навчання. Однак, навіть на етапі свого нинішнього прогресу вони глобально впливають на життєдіяльність всього суспільства, формуючи нові уявлення про майбутнє і перспективи розвитку надсучасних технологій. Штучного інтелекту надійно, системно виконує комп'ютеризовані завдання[2].

Метою роботи є розкриття потенціалу системи автоматизації написання програмного коду, що дозволяє суттєво скоротити неосяжний обсяг рутинної роботи та розробляти програми будь-якої складності. Отже завдання полягає в створенні системи, яка здатна самостійно скласти алгоритм роботи, або працювати з кодом інших програм та самостійно навчатись.

### **Дослідження системи**

Сучасна автоматизація, розмовні платформи, розумні боти та інтелектуальні машини працюють із величезною кількістю даних для вдосконалення багатьох технологій вдома або на робочому місці.[3]

Саме система DeepCoder є нейромережею, яка здатна виконувати «чорну» роботу і розробляти програми будь-якої складності, навіть не знаючи мови та не вмюючи програмувати. Достатньо скласти лише алгоритм і поставити завдання, і нейромережа самостійно напише код для їх вирішення.

Отже, DeepCoder - це програма для роботи з готовим кодом, який знаходиться у відкритому доступі, а на основі опису відповідних функцій самостійно створює робочий алгоритм[2].

DeepCoder використовує технологію під назвою програмний синтез: нові програми створюються шляхом копіювання рядків коду з уже створених чимось напрацювань. Системі дається список вхідних і вихідних значень, за яким вона визначає, яку ділянку коду їй потрібно скопіювати, щоб досягти бажаного результату. Система швидко адаптується при отриманні нових даних, що поступово призводить до повного виключення помилок у реалізації певного автоматизованого процесу та розвивається за допомогою алгоритмів прогресивного навчання і формує дані для подальшого програмування. Вона самостійно знаходить структуру та закономірності у даних, опрацьовуючи їх таким чином, що фактично сам алгоритм набуває певного вміння. Можливості такого навчання – безмежні з точки зору використання розумних машин для вирішення широкого спектру задач. Цінність полягає в тому, що штучний інтелект здатний шукати інформацію набагато швидше людини. Крім того, для дослідження наявних баз даних DeepCoder використовує машинне навчання, завдяки чому з кожним разом система знаходить більш «правильні» ділянки коду і поступово вдосконалюється. Слід зауважити, що DeepCoder створює програми за лічені доли секунди, у той час як її попередникам на рішення подібних завдань необхідно кілька хвилин. Штучний інтелект без особливих труднощів виконує будь-які завдання, що дають розробникам для оцінки їх навичок. За допомогою такої нейромережі можливо створювати програми без знань конкретної мови та вмінь програмувати. Достатньо лише скласти алгоритм і поставити завдання.

Важливо мати на увазі, що DeepCoder як і інші системи програмного синтезу використовує предметно-орієнтовану мову (DSL), спеціалізовану для конкретної області застосування, так би мовити, «урізанну мову програмування». Вона містить тільки дев'ять функцій першого порядку: HEAD, AST, TAKE, DROP, ACCESS, MINIMUM, MAXIMUM, REVERSE, SORT, SUM і п'ять функцій

Самого високого порядку: MAP, FILTER, COUNT, ZIPWITH, SCANL1. Вибір рядків для складання програми здійснюється після аналізу вхідних і вихідних значень кожного фрагмента коду.

Система машинного навчання швидко вчиться розуміти, як використовувати ці зразки - і заздалегідь складає базу. Крім того, нейромережа з'єднує знайдені рядки таким способом, яким ніколи не прийде в голову з'єднати їх живій людині. Методом проб і помилок нейромережа поступово вчиться домагатися поставленої мети. Переваги такої системи дозволяють DeepCoder писати програми набагато швидше, ніж її попередникам. Вона пише програма з трьох рядків за доли секунди, у той час як попереднім системам потрібно в рази або в десятки разів більше часу, щоб перепробувати всі можливі варіанти. По мірі навчання система розуміє, які комбінації коду працюють, а які ні. Вона вдосконалюється в програмуванні з кожним новим завданням. Таку технологію можна використовувати для рефакторингу пошуку багів в програмах. Вона знайде некоректні рядки і замінить їх на правильні рядки з інших програм. Автори вважають, що з її допомогою легко створювати рутинні програми в стилі відомої IFTTT - автоматизувати базові взаємодії між різними веб-сервісам. Навіть непрограміст може скласти прості інструкції на кшталт вибору фотографій з Facebook і сортування за заданими критеріями. Такі прості програми можна буде створювати за хвилини без знання програмування. Звичайно ж, такі інструменти поки не можуть замінити справжнього програміста. Вони розглядаються тільки як допоміжний інструмент, але у найближчому майбутньому він суттєво полегшить роботу програмістів.

## Висновки

DeepCoder вміє шукати більш масштабно, ніж програміст, і збирати фрагменти коду з різних джерел, про які програміст може не здогадуватися. Ідея роботи полягає в тому, щоб зняти розробників з "брудної роботи" конфігурації та калібрування, делегуючи все, до нейронних мереж. Ця брудна робота сьогодні займає більшу частину часу: вона складається здебільшого з випробувань та помилок, спроб з деякими алгоритмами та параметрами, налаштування їх там, де це необхідно.

Для того щоб удосконалити цю систему потрібно зібрати величезну базу алгоритмів, функцій, методів, з якими б система працювала і одночасно вчилася вдосконалюватися завдяки машинному самонавчанню. Щоб підвищити швидкість роботи, система повинна підібрати для себе алгоритми, які будуть спроможні виконувати поставлені задачі. На відміну від своїх аналогів DeepCoder підтримується компанією гігантом Microsoft, що є досить потужним фактором у наш час і знімає з системи обмеження по бюджету, оновленню, догляду та модернізуванню кращими інженерами у світі.

## СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. [Електронний ресурс ] [DeepCoder - microsoft.com](#)
2. [Електронний ресурс ] [DeepCoder - algorithm](#)
3. «Homo Roboticus. Люди і машини у пошуках взаєморозуміння», Джон Маркофф «Альпіна Паблішер», 2017
4. «Четверта промислова революція», Клаус Шваб «Ексмо», 2016
5. Васюра А.С., Мартинюк Т.Б., Куперштейн Л.М. Методи та засоби нейроподібної обробки даних для систем керування.// Монографія. – Вінниця, УНІВЕРСУМ, 2008, 175 с.

*Середюк Гліб Володимирович* – студент групи ІАКІТ - 17б, факультет комп'ютерних систем і автоматики, Вінницький національний технічний університет, Вінниця, e-mail : [glebserediuk@gmail.com](mailto:glebserediuk@gmail.com)

Науковий керівник - *Васюра Анатолій Степанович* — професор кафедри автоматизації та інтелектуальних інформаційних технологій, Вінницький національний технічний університет, м. Вінниця. e-mail: [vasanat@i.ua](mailto:vasanat@i.ua).

*Serediuk Gleb V.* - student of ІАКІТ – 17b - group, Faculty of Computer Systems and Automation, Vinnitsa National Technical University, Vinnitsia, e-mail : [glebserediuk@gmail.com](mailto:glebserediuk@gmail.com)

Supervisor: *Vasyura Anatoly S.* — Professor, academician of Ukrainian Technological Academy, Professor of automation and intelligent information technologies department, Vinnitsia National Technical University, Vinnitsia, e-mail: [vasanat@i.ua](mailto:vasanat@i.ua).