

Рекомендаційні системи відіграють важливу роль у сучасному світі. Вони використовуються в онлайн-магазинах, бібліотеках, на сайтах для перегляду фільмів та книжок, у соціальних мережах та інших інтернет-ресурсах. Такі системи значно спрощують життя як користувачам, так і тим, хто надає послуги на цих платформах.

Список використаних джерел:

1. Isinkaye F.O., Folajimi Y.O., Ojokoh B.A., Recommendation systems: Principles, methods and evaluation, Egyptian Informatics Journal, Volume 16, Issue 3, 2018, Pages 261-273
2. Обліковий запис [Електронний ресурс]. – Режим доступу : [https://uk.wikipedia.org/wiki %D0%9E%D0%B2%D0%B8%D0%](https://uk.wikipedia.org/wiki/%D0%9E%D0%B2%D0%B8%D0%)
3. Користувач [Електронний ресурс]. – Режим доступу : <https://uk.economy-pedia.com/11031989-user>
4. Melville, P. and Sindhvani, V. (2010). Recommender systems. Encyclopedia of Machine Learning, Vol. 1, pp. 829–838.

ПРОГРАМУВАННЯ З ВИКОРИСТАННЯМ ШТУЧНОГО ІНТЕЛЕКУ

Панікар Д.В., студент МК-61-23

Науковий керівник – *Плехова Г.А.*, доц., к.т.н.

Харківський національний автомобільно-дорожній університет

Сучасні технології дозволяють значно легше вирішувати багато проблем людства, ніж кілька десятиліть тому. Особливо швидко розвивається штучний інтелект (ШІ). Одним із найбільш перспективних напрямків його застосування є мови програмування. ШІ може спростити і автоматизувати процес розробки програмного забезпечення, забезпечити зручний та інтуїтивний інтерфейс, а також підвищити ефективність і безпеку коду.

Головними напрямками використання ШІ у мовах програмування є:

- Генерація коду, оскільки ШІ-моделі генерують код на основі заданих параметрів, що значно економить час та зусилля розробників. Сюди можна додати також і автоматичне доповнення коду. Наприклад, багато IDE (інтегрованих середовищ розробки) вже використовують ШІ для автоматичного доповнення коду, що може значно полегшити та пришвидшити процес написання коду.
- Автоматизація тестування. Інструменти на базі ШІ автоматизують процес тестування програм, допомагаючи знаходити і виправляти помилки.

- **Налагодження.** ШІ допомагає програмістам у відлагодженні, визначаючи причини помилок та пропонуючи можливі шляхи їх вирішення.

Аналіз даних. ШІ-алгоритми використовуються для аналізу даних, що сприяє прийняттю більш обґрунтованих рішень щодо дизайну та функціональності програм.

Аналізуючи всі вищеперераховані фактори, можна виділити певні переваги використання ШІ у мовах програмування. Одним з таких плюсів є автоматизація рутинних завдань, адже ШІ може автоматизувати багато рутинних завдань, таких як генерація коду, тестування та відладка. Зокрема, система генеративного штучного інтелекту може створити шаблонний код на основі заданих вхідних параметрів та специфікацій. Це значно звільняє час розробників, дозволяючи їм сконцентруватися на більш складних та творчих аспектах програмування.

Наприклад, використання модуля Tensorflow у Python для машинного навчання дозволяє ШІ автоматично створювати модель нейронної мережі, налаштовувати гіперпараметри та тренувати модель на великому наборі даних. Це значно спрощує процес розробки.

Наступна перевага – це підвищення продуктивності, бо завдяки автоматизації та іншим можливостям ШІ, розробники можуть писати код швидше та з меншою кількістю помилок. Наприклад, розглянемо використання ШІ при розробці веб-додатків. ШІ може допомогти розробникам автоматично генерувати код для стандартних функцій або повторюваних процесів. У такому випадку, система генеративного штучного інтелекту може створити шаблони для створення сторінок, обробки форм або взаємодії з базою даних. Розробникам залишається лише налаштувати ці шаблони згідно зі своїми потребами, що значно прискорює процес розробки.

Звісно, не менш вагомим плюсом є створення нових можливостей. ШІ може використовуватися для створення нових інструментів та можливостей для розробників, які раніше не були доступні. Один з прикладів цього – автоматичне генерування коду за допомогою глибокого навчання. Використовуючи нейромережі та алгоритми генерації тексту, ШІ може створювати нові частини коду або навіть повністю функціонуючі програми. Наприклад, можна навчити модель генерувати код для створення веб-сторінок або розробки алгоритмів обробки даних. Це відкриває нові можливості для швидкого прототипування та розробки програмного забезпечення.

До переваг можна віднести і покращення якості коду. ШІ може допомогти розробникам писати якісніший код, який є більш читабельним, модульним та безпечним.

Прикладом може слугувати автоматична перевірка стилю коду та виявлення потенційних проблем. ШІ може аналізувати синтаксис, структуру та стиль коду, надаючи рекомендації щодо їх вдосконалення. На додаток, він може виявляти неконсистентність в іменах змінних, неправильне форматування або відступи, а також надавати рекомендації щодо їх виправлення. Це

допомагає забезпечити чистоту та однорідність коду, полегшує співпрацю між розробниками та поліпшує загальну читабельність коду.

Важливо відзначити, що використання генеративних систем штучного інтелекту, особливо моделі генеративного попереднього підготовленого трансформатора *Generative Pre-trained AI*(ChatGPT), у дистанційному навчанні для програмістів-початківців має потенціал для покращення процесу навчання. ChatGPT діє як інтерактивний помічник, який надає інструкції з програмування, вказівки та поради для початківців.

Тому використання штучного інтелекту в навчанні програмуванню забезпечує індивідуальний підхід до кожного учня та сприяє пробудженню інтересу до предмету. Однак, використання штучного інтелекту у мовах програмування також вносить свої ризики та виклики, які потрібно враховувати.

По-перше, це складність і вартість. Розробка та впровадження систем ШІ може бути складною та дорогою справою. Навчання моделі ШІ вимагає обчислювальних ресурсів, інфраструктури, навчальних даних і часу. Досягнення високої якості та ефективності вимагає великої кількості обчислювальної потужності та може бути дорогим.

Друга тема – Етика та безпека. Використання штучного інтелекту в мовах програмування може призвести до етичних проблем і проблем безпеки, таких як упередженість алгоритмів і неправильне використання ШІ. Коли моделі штучного інтелекту навчаються з неточними або неповними даними, можуть виникнути алгоритмічні зміщення, що призведе до неточних або несправедливих результатів.

Наприклад, якщо модель штучного інтелекту для автоматизованого прийняття рішень про наймання розглядає минулі дані про прийом на роботу, які можуть містити упередження, пов'язані зі статтю, расою чи іншими факторами, модель навчиться приймати рішення, які можуть призвести до дискримінації. Це може мати серйозні соціальні наслідки та порушувати принципи справедливості.

Останнім моментом є потреба в кваліфікованому персоналі.

Для розробки та використання систем штучного інтелекту з використанням мов програмування потрібен спеціалізований персонал зі знаннями програмування та штучного інтелекту. Одним із прикладів, коли потреба в кваліфікованому персоналі стає важливою, є розробка систем обробки природної мови (NLP). NLP використовується для розуміння та обробки звукових даних, таких як текстові повідомлення, документи та голосові команди.

Таким чином, використання штучного інтелекту в мовах програмування має великий потенціал для покращення процесу розробки програмного забезпечення. Впроваджуючи системи генеративного штучного інтелекту та інші інструменти, ви можете автоматизувати та спростити процес програмування та підвищити якість і продуктивність програми.

Однак використання штучного інтелекту також має недоліки та ризики, які вимагають ретельного аналізу та контролю. Дослідження в цій галузі

сприяють розумінню можливостей і обмежень штучного інтелекту в мовах програмування, а також сприяють подальшому розвитку та вдосконаленню цих технологій.

Література

1. Шаров, С. В. (2023). Сучасний стан розвитку штучного інтелекту та напрямки його використання. *Українські студії в європейському контексті*, 6, 136. Режим доступу: http://obrii.org.ua/usec/storage/conference/zb_vol6_2023.pdf#page=137.

2. Sisilitsyn, Y. O., & Osadchyi, V. V. (2023). USING CHATGPT IN DISTANCE LEARNING FOR BEGINNERS IN PROGRAMMING. *Information Technologies and Learning Tools*, 97(5), 167-180. <https://doi.org/10.33407/itlt.v97i5.5277>.

3. Бойко, В., Василенко, М., & Слатвінська, В. (2023). Програмування за допомогою систем генеративного штучного інтелекту: ризики та виклики. *Інформаційні технології та суспільство*, (2 (8), 18-26. <https://doi.org/10.32689/maup.it.2023.2.2>.

ВПРОВАДЖЕННЯ МЕТОДІВ ШТУЧНОГО ІНТЕЛЕКТУ У ВАНТАЖОПЕРЕВЕЗЕННЯ

Чорнобай Е.І., студент МК 61-23

Науковий керівник – *Карпішен Б.С.*, асистент

Харківський національний автомобільно-дорожній університет

Штучний інтелект (ШІ) існує з 1950-х років, але завдяки технологічному розвитку сьогодні він стає дедалі популярнішим. Ця технологія дозволяє комп'ютерним програмам навчатися, імітуючи людську поведінку, та відтворювати такі характеристики, як здатність до міркування, розуміння значень і навчання на власному досвіді. Внаслідок цього ШІ може виконувати завдання розумно, без потреби в детальних інструкціях.

На сучасному етапі вважається, що ШІ допомагає зменшити кількість людських помилок і присутній практично скрізь. Використання таких систем, як Siri або Alexa, є прикладом застосування цієї технології. Сьогодні ШІ використовується в багатьох галузях, таких як реклама, охорона здоров'я, роздрібна торгівля, електронна комерція, фінанси, виробництво, транспорт і логістика [1].