

– Освіта: використовуйте генеративний ШІ для створення персоналізованих навчальних матеріалів.

– Бізнес: Generative AI можна використовувати не тільки для розробки нових продуктів і послуг, але й для оптимізації існуючих бізнес-процесів.

– Мистецтво: використовуйте генеративний ШІ для створення нових форм мистецтва та розваг.

Важливо, щоб цю технологію досліджували, відповідально розробляли та використовували на благо людства.

Generative AI – це захоплююча сфера штучного інтелекту, яка швидко розвивається. Чим більше ми дізнаємося про генеративний ШІ, тим більше зможемо використовувати його на благо людства. Генеративний ШІ зробить наш світ кращим, але лише якщо ми будемо використовувати його з розумом..

ІНТЕГРАЦІЯ ШІ У WEB

Годлевський М.Р., студент МК-61-23

Науковий керівник – *Попова А.В.*, доц.. к.т.н.

Харківський національний автомобільно-дорожній університет

Вступ.

Штучний інтелект (AI) і нейронні мережі (NN) стали частиною технологічного світу, відкриваючи багато можливостей у багатьох аспектах нашого життя. Нейронні мережі, частина штучного інтелекту, є моделями, натхненними тим, як працює людський мозок.

У мереж є здатність вчитися та вирішувати проблеми без програмування. Значні досягнення в цій галузі дозволили використовувати його в таких сферах, як медицина, фінанси, технології, аналіз даних і веб-розробка. Нейронні мережі добре розпізнають шаблони в даних і адаптуються до мінливих умов для вирішення завдань. Ці функції роблять його дуже ефективним у багатьох програмах.

Одним із найбільш актуальних застосувань нейронних мереж є веб-розробка. Вони відіграють певну роль у покращенні взаємодії користувачів, персоналізації пропозицій контенту, аналізі інформації про поведінку користувачів і автоматизації робочих завдань, які традиційно виконувалися вручну. Наприклад, системи рекомендацій на основі нейронної мережі можуть вибирати контент відповідно до індивідуальних уподобань користувача.

Інтеграція інструментів штучного інтелекту в процес розробки веб-сайтів – це не просто минуша тенденція, а стає необхідністю в сучасному швидкому цифровому світі. Використання цих інструментів зазвичай дає багато переваг як розробникам, так і компаніям.

У сучасних дослідженнях нейронна мережа (NN), яку також називають штучною нейронною мережею (АНМ), є моделлю з дуже гнучкою структурою та різними параметрами. Натхненні біологічними механізмами людського мозку, нейронні мережі можуть розпізнавати шаблони в наборах даних і вирішувати складні завдання, включаючи аналіз мови, обробку візуальної інформації та багато інших аспектів.

Нейронні мережі можна поділити на три основні типи шарів, які їх структурно складають [1]:

1) Вхідний шар: Це початковий шар мережі, куди вводяться вхідні дані для подальшої обробки. Приклад реалізації: функція - приймає вхідні дані для подальшої обробки, специфіка навчання - інтеграція веб-даних у вхідний шар, забезпечуючи мережі доступ до контексту веб-сайту.

2) Приховані шари: Приховані шари розташовані між вхідним та вихідними шарами, і при цьому, вони називаються "прихованими," оскільки їхні вхідні та вихідні дані не є безпосередньо доступними для спостереження. В цих шарах мережа набуває здатність розпізнавати шаблони в наборах даних, оновлюючи свої параметри (ваги) під час навчання на основі вхідних даних. Приклад реалізації: функція - розпізнавання шаблонів у вхідних даних, специфіка навчання - автоматичне вивчення патернів поведінки відвідувачів сайту та адаптація до їх вимог.

3) Вихідний шар: Вихідний шар відповідає за надання кінцевого результату обробки мережі, такого як класифікація слів у тексті або виявлення об'єктів на зображенні. Приклад реалізації: функція - подання кінцевого результату обробки. Специфіка навчання - автоматичне адаптування до змін у поведінці користувачів для покращення точності прогнозування та рекомендацій.

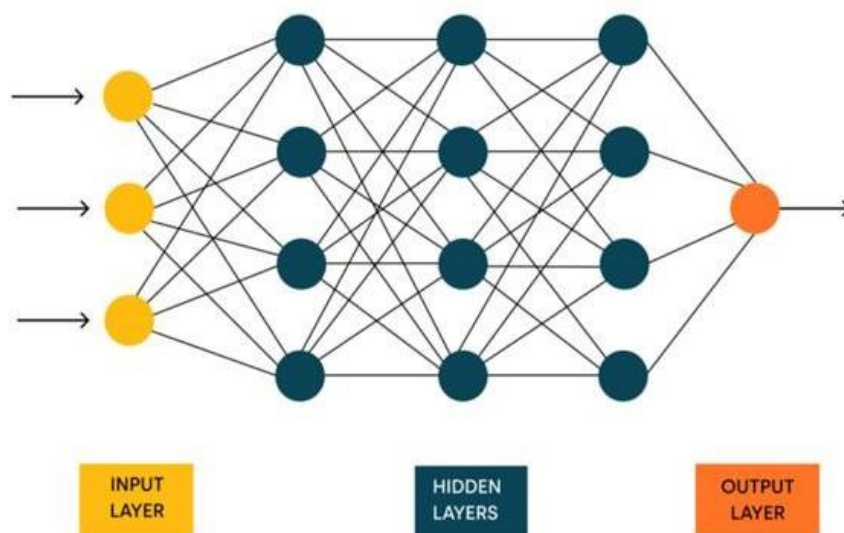


Рисунок 1 – Архітектура Нейромережі

Однією з ключових особливостей нейронних мереж є їх здатність дізнаватися, які аспекти вхідних даних впливають на вихідні результати, без

необхідності вручну визначати особливості даних. Цей процес навчання дозволяє мережі підвищувати продуктивність під час обробки нових даних і розвивати її властивості з часом, що робить її універсальним інструментом для різноманітних завдань.

Інтеграція нейронних мереж у веб-додатки. По-перше, інструменти штучного інтелекту дозволяють автоматизувати рутинні завдання, що економить багато часу та зусиль, витрачених на процес веб-розробки. Зокрема, інструменти штучного інтелекту можуть покращити функціональність веб-сайту, забезпечуючи персоналізований досвід користувача. Він аналізує поведінку користувачів, надає рекомендації щодо відповідного контенту, продуктів і послуг, а також надає відповіді в режимі реального часу завдяки чат-ботам на основі штучного інтелекту [1].

Крім того, інструменти штучного інтелекту можуть значно покращити видимість веб-сайту в пошукових системах, допомагаючи розробникам застосовувати найкращі практики SEO. Тому важливо впроваджувати штучний інтелект та його інструменти в процес веб-розробки, щоб створювати динамічні, зручні та оптимізовані веб-сайти, які сприяють розвитку бізнесу.[2].

Використання штучного інтелекту в веб-розробці включає аспекти, відмінні від дизайну та кодування. Важливим напрямком є використання чат-ботів на основі штучного інтелекту. Ці чат-боти покращують обслуговування клієнтів, надаючи персоналізовані відповіді в реальному часі на запити користувачів.

Завдяки обробці природної мови (NLP) ці боти можуть розуміти та вести розмови природною мовою, створюючи інтерактивний досвід, який нагадує спілкування зі справжньою людиною. Інше застосування штучного інтелекту в веб-розробці – це створення персоналізованого досвіду користувача.

Інструмент на основі штучного інтелекту аналізує поведінку користувачів, уподобання та моделі взаємодії, щоб надавати персоналізований контент і рекомендації. Це збільшує залучення користувачів і значно підвищує ймовірність конверсії. У міру розвитку штучного інтелекту його впровадження у веб-розробку буде продовжувати збільшуватися, змінюючи процес розробки та розпізнавання веб-сайтів.

Впровадження нейронної мережі на вашому веб-сайті вимагає кількох кроків і може відрізнятись залежно від конкретного використання.

Однак основні кроки можна підсумувати так:

1. Вибір та Навчання Моделі:

- Вибір Фреймворку: Визначте, який фреймворк для машинного навчання ви будете використовувати. TensorFlow та PyTorch - популярні вибори для веб-розробки.

- Навчання Моделі: Навчіть вашу нейронну мережу на відповідних даних. Використовуйте набір даних, що відповідає завданню, яке ви хочете вирішити.

2. Експорт Моделі:

- Збереження Моделі: Після навчання збережіть модель у відповідному форматі. У TensorFlow це може бути формат SavedModel, а у PyTorch - TorchScript.

3. Інтеграція з Веб-Сайтом:

- Реалізуйте серверний бекенд, який надає API для взаємодії з навченою моделлю.

Застосуйте фреймворк для створення API.

- На стороні клієнта веб-сайту використовуйте JavaScript [14] або інші мови для створення запитів до API для передачі даних на обробку моделі та отримання результатів.

- Використовуйте AJAX-запити або Fetch API [15] для асинхронного обміну даними з сервером.

4. Обробка Результатів:

- Обробляйте результати, отримані від нейронної мережі, на стороні клієнта для відображення або подальшого використання.

- Використовуйте JavaScript або інші мови для маніпулювання DOM та оновлення вмісту сторінки.

5. Оптимізація та Захист:

- Забезпечте оптимізацію продуктивності вашої моделі, особливо якщо вона запускається на клієнтському браузері.

- Забезпечте захист даних, переданих між веб-сайтом і сервером, особливо якщо вони містять конфіденційну інформацію.

Архітектури та бібліотеки для впровадження нейронних мереж. Бібліотеки машинного навчання мають величезний вплив на IT-індустрію та веб-розробку. Вони надають розробникам потужні інструменти для створення та навчання нейронних мереж і моделей штучного інтелекту, які використовуються для різноманітних завдань обробки даних і розуміння вмісту. Вибір між цими фреймворками залежить від конкретних потреб розробника та деталей проекту веб-розробки зі штучним інтелектом. Обидва фреймворки мають переваги та недоліки, і вибір між ними важливий на основі конкретних завдань і ресурсів, доступних для розробки.

Хоча TensorFlow вражає своєю масштабованістю та потенціалом розподіленого навчання, PyTorch забезпечує спрощену архітектуру та зручний інтерфейс для налагодження моделей. Незалежно від обраного вами фреймворка, інтеграція штучного інтелекту в веб-розробку відкриває багато можливостей для створення персоналізованого досвіду користувача, покращення функціональності веб-сайту та оптимізації видимості в пошуковій системі. Він надає розробникам інструменти для розробки інтерактивних та інтелектуальних веб-додатків, які можуть значно підвищити конкурентоспроможність компаній у цифровому світі.

TensorFlow. Бібліотека TensorFlow [3], розроблена компанією Google, є однією з найбільш визнаних та використовуваних бібліотек для чисельних

обчислень. Ця відкрита бібліотека написана на мовах програмування Python та C++, і здобула впізнаваність завдяки своїй потужності та гнучкій архітектурі. Індустріальні гіганти, такі як DeepMind, Uber, Airbnb та Dropbox, вибрали TensorFlow для використання в своїх проектах.

TensorFlow ідеально підходить для складних завдань, таких як створення глибоких нейронних мереж. Його програми широко використовуються в таких сферах, як розпізнавання мовлення та зображень і обробка текстової інформації, одним із прикладів є програма Google Translate.

Серед переваг TensorFlow варто виділити наступні моменти:

- Велика кількість документації та навчальних матеріалів для розробників.

- Інтегровані інструменти Tensorboard для моніторингу та візуалізації процесу навчання моделі.

- Підтримуйте спільноту розробників і технологічних компаній.

- Можливість розподіленого навчання.

- TensorFlow Lite оптимізовано для виконання з низькою затримкою на мобільних пристроях [4].

Однак важливо звернути увагу на деякі обмеження.

- Порівняно з іншими бібліотеками, такими як CNTK [5] і MXNet [6], TensorFlow може бути трохи повільнішим у деяких тестах.

- Для початківців TensorFlow може здатися незручним, оскільки вимагає багато шаблонного коду та функції «визначити та запустити», що може заплутати під час налагодження.

PyTorch. PyTorch [7] є конкурентом TensorFlow і наступником бібліотеки Torch, написаної на мові програмування Lua.

Ця бібліотека, розроблена Facebook, також використовується багатьма іншими компаніями та організаціями, включаючи Twitter, Salesforce і Оксфордський університет.

PyTorch відзначається наступними перевагами:

- Його архітектура робить процес створення моделей досить простим та прозорим.

- Можливість використовувати режим "визначити за запуском" спрощує налагодження та підтримує популярні інструменти для дебагування.

- Підтримка декларативного паралельного обробника даних.

- Наявність великої кількості попередньо навчених моделей та готових модульних частин, які легко комбінувати.

- Можливість розподіленого навчання, яка стала доступною з версії 0.4.

- За недоліки PyTorch можна відзначити:

- Певну відсутність підтримки деяких моделей.

- Не повну готовність до впровадження в продакшн, хоча шлях до версії 1.0 обіцяє багато нововведень.

- Відсутність вбудованих інструментів для моніторингу та візуалізації, які присутні у TensorFlow.

Приклади. Нейронні мережі мають великий вплив на основні галузі, такі як фінанси, охорона здоров'я та автомобілебудування. Ці штучні нейрони функціонують подібно до людського мозку та використовуються в різних сферах. Давайте розглянемо деякі з різних застосувань нейронних мереж.

Розпізнавання облич: Системи розпізнавання облич стали надійними системами спостереження. Він порівнює людські обличчя з цифровими зображеннями та використовується в різних сферах, таких як вибірковий запис і автентифікація. Згорточні нейронні мережі (CNN) [7] використовуються для розпізнавання облич і обробки зображень і вивчають велику кількість зображень для точного розпізнавання.

Наприклад, ви можете створити просту модель CNN для класифікації облич або модель для обробки зображень і розпізнавання облич за допомогою OpenCV і Dlib. Код прикладу розпізнавання обличчя:

```
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense #
Створення моделі CNN
model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=(64, 64, 3)),
    MaxPooling2D(pool_size=(2, 2)),
    Conv2D(64, (3, 3), activation='relu'),
    MaxPooling2D(pool_size=(2, 2)), Flatten(),
    Dense(128, activation='relu'), Dense(1, activation='sigmoid')

    # Компіляція моделі model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
    # Завантаження та підготовка даних для тренування # (зображення обличчя та їх мітки)
    # Тренування моделі
    model.fit(train_data, train_labels, epochs=10, validation_data=(val_data, val_labels))
```

Прогнозування фондового ринку. Нейронні мережі використовуються для прогнозування коливань фондового ринку. Багатошарові перцептрони (MLP) [8] та інші алгоритми використовуються для аналізу фінансових даних і прогнозування руху цін на акції, допомагаючи інвесторам приймати обґрунтовані рішення. Наприклад, ви можете створити просту модель MLP для прогнозування цін на нерухомість на основі розміру квартири або витрат клієнтів у роздрібному магазині на основі кількості покупок.

Соціальні медіа. Нейронні мережі використовуються для вивчення поведінки користувачів соціальних мереж і аналізу даних, якими обмінюються під час віртуальних розмов. Це допомагає нам вивчати звички користувачів і те, як вони витрачають свої гроші. Наприклад, уявіть, що у вас є дані про корис-

тувачів соціальної мережі, такі як кількість друзів, кількість публікацій, кількість лайків і час, проведений онлайн. Нейронні мережі можна використовувати для прогнозування можливих витрат.

Медицина. У медицині нейронні мережі використовуються для виявлення аномалій на таких зображеннях, як рентген, магнітно-резонансна томографія (МРТ) і комп'ютерна томографія (КТ).

Це може допомогти виявити захворювання на ранній стадії, підвищити точність діагностики та полегшити розробку індивідуальних планів лікування.

Безпілотних автомобілів.

Нейронні мережі використовуються в автомобільній промисловості для розробки автопілотів і систем виявлення руху. Вони допомагають безпілотним автомобілям приймати рішення на основі великої кількості даних, аналізувати оточення та уникати аварій на дорогах.

Варто відзначити, що нейронні мережі вважаються особливо перспективними для вирішення завдань веб-розробки завдяки їх здатності розпізнавати шаблони в даних і адаптуватися до мінливих умов. Такі фреймворки, як TensorFlow і PyTorch, розглядаються для роботи, і розробники повинні враховувати конкретні потреби своїх проектів і власні ресурси.

Таким чином, інтеграція штучного інтелекту в веб-розробку визнана необхідним кроком розвитку в цифровому світі, що дає розробникам можливість створювати інтелектуальні та інтерактивні веб-додатки.

Використання нейронних мереж відкриває багато можливостей для підвищення конкурентоспроможності компаній і покращення досвіду користувачів в онлайн-середовищі. Аналітичні та прогнозні можливості нейронних мереж виявилися дуже корисними в різних сферах, оскільки вони надають нові можливості та вдосконалюють процеси.

Література

1. Google Assistant [Електронний ресурс] – Режим доступу до ресурсу: <https://assistant.google.com/intl/eu-eu/>

2. Amit S. R. Layers in Neural network [Електронний ресурс] / Singh Rathore Amit // Nerd For Tech. – 2021. – Режим доступу до ресурсу: <https://medium.com/nerd-for-tech/layers-in-neural-network-90d48a5a42fb>.

3. Introduction to TensorFlow [Електронний ресурс] – Режим доступу до ресурсу: <https://www.tensorflow.org/learn>.

4. TensorFlow Lite [Електронний ресурс] – Режим доступу до ресурсу: <https://www.tensorflow.org/lite>.

5. What is a Neural Network? [Електронний ресурс] – Режим доступу до ресурсу: <https://aws.amazon.com/what-is/neural-network/>.

6. APACHE MXNET: A FLEXIBLE AND EFFICIENT LIBRARY FOR DEEP LEARNING [Електронний ресурс] – Режим доступу до ресурсу: <https://mxnet.apache.org/versions/1.9.1/>.

7. PYTORCH [Електронний ресурс] – Режим доступу до ресурсу: <https://pytorch.org/>.

8. Neural network models [Електронний ресурс] // scikit-learn developers – Режим доступу до ресурсу: https://scikit-learn.org/stable/modules/neural_networks_supervised.html.

9. OpenCV [Електронний ресурс] – Режим доступу до ресурсу: <https://opencv.org/>.

10. What are neural networks? [Електронний ресурс] // IBM – Режим доступу до ресурсу: <https://www.ibm.com/topics/neural-networks>.

11. Web Neural Network API [Електронний ресурс] – Режим доступу до ресурсу: <https://www.w3.org/TR/webnn/>.

12. Nielsen M. Neural Networks and Deep Learning / Nielsen Michael., 2019. – 224 с.

13. David S. AI is helping to make better software [Електронний ресурс] / David Schatsky // deloitte. – 2020. – Режим доступу до ресурсу: <https://www.deloitte.com/global/en/our-thinking/insights/topics/business-strategy-growth/signals-for-strategists/ai-assisted-software-development.html>.

14. JavaScript [Електронний ресурс] – Режим доступу до ресурсу: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>.

15. Fetch API [Електронний ресурс] – Режим доступу до ресурсу: https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API

НЕЙРОМЕРЕЖІ ІЗ ВИКОРИСТАННЯМ АЛГЕБРИ ПРЕДИКАТІВ

Супрун В.М., студент МК51-24

Науковий керівник – *Пронін С.В.*, доц., к.т.н.

Харківський національний автомобільно-дорожній університет

Алгебра предикатів відіграє важливу роль у формалізації та розв'язанні різноманітних завдань штучного інтелекту. Її основні поняття та методи використовуються для опису властивостей об'єктів та їхніх зв'язків у формі формальних логічних виразів.

1. Основні поняття алгебри предикатів

Алгебра предикатів фокусується на описі властивостей об'єктів у формі предикатів. Предикат - це вираз, який є істинним або хибним для різних значень змінної. Основні поняття включають істину, хибність, квантори (універсальні та екзистенціальні) і логічні операції (комбінація, розділення, заперечення).