Національний технічний університет України «КПІ ім. Ігоря Сікорського»

Факультет Інформатики та Обчислювальної Техніки

Кафедра інформаційних систем та технологій

Лабораторна робота №5

з дисципліни «Прикладні задачі машинного навчання»

на тему

«Проектування та навчання штучної нейронної мережі для задач класифікації»

15 варіант

Виконав:

студент групи ІС-21 Костюк А. С.

Викладач:

Нестерук А.О.

Київ – 2024

**Зміст**

[**1. Постановка задачі** 3](#_Toc166938553)

[**2. Виконання** 4](#_Toc166938554)

[**3. Висновок** 16](#_Toc166938555)

**1. Постановка задачі**

1. Виконати завдання із прикладу i отримати файл із навченою моделлю для розпізнавання рукописних цифр. В будь якому графічному редакторі створити файл із рукописною цифрою i розпізнати її. Пояснити результат.

2. Спроектувати i розробити нейронну мережу на основі таких наборів даних імплементованих в Keras:

2.1. Cifar10

2.2. FMNIST

Потрібно розробити архітектуру нейронної мережі, навчити її на тестових прикладах, i продемонструвати її роботу на кількох зображеннях.

3. Зробити звіт про роботу.

**2. Виконання**

**1)** **Виконати завдання із прикладу i отримати файл із навченою моделлю для розпізнавання рукописних цифр. В будь якому графічному редакторі створити файл із рукописною цифрою i розпізнати її. Пояснити результат.**

Зображення, що містить текст, знімок екрана, програмне забезпечення, Шрифт

Автоматично згенерований опис

Переглянемо перші 25 зображень та відповідні індекси:

Зображення, що містить текст, число, Шрифт, почерк

Автоматично згенерований опис

Зображення, що містить Шрифт, типографія

Автоматично згенерований опис

Перетворимо тривимірний масив в двовимірний масив (60000, 28\*28) типу float32, а потім нормалізуємо його так, щоб отримати значення в інтервалі [0, 1]:

Зображення, що містить текст, знімок екрана, Шрифт, ряд

Автоматично згенерований опис

Тепер необхідно створити мітки для кожного тестового зображення, завантаживши масиви з відгуками. Для цього використовується функція to\_categorical(), яка перетворює масив NumPy з цілими десятковими числами на матрицю з двійковими значеннями, де кількість стовпців дорівнює кількості категорій у даних. Після цього навчальні дані train\_images і train\_labels будуть передані нейронній мережі, яка буде навчена зіставляти зображення з мітками. Після класифікації зображень у test\_images, точність класифікації мережі буде перевірена за допомогою міток test\_labels:

Зображення, що містить текст, знімок екрана, Шрифт, ряд

Автоматично згенерований опис

Модель використовує послідовну архітектуру, яку можна створити за допомогою класу Sequential. Ця архітектура складається з послідовності шарів, включаючи вихідний шар, який має 784 нейрони, тобто кожному пікселю зображення відповідає один нейрон. За допомогою методу .add() створено прихований шар, який є функцією, що приймає тензор на вході та повертає тензор на виході, можливо, іншого розміру. Використовуються щільні, повнозв’язні (Dense) шари, де кожен нейрон одного шару зв'язаний з кожним нейроном наступного шару. Прихований шар містить 512 нейронів, а вихідний шар містить 10 класифікаційних нейронів. Функція активації для прихованого шару - relu, а для вихідного шару – softmax:

Зображення, що містить текст, знімок екрана, Шрифт, програмне забезпечення

Автоматично згенерований опис

Компілюємо мережу:

Зображення, що містить текст, знімок екрана, Шрифт, ряд

Автоматично згенерований опис

Тепер треба навчити алгоритм з використанням методу .fit(). Під час навчання, модель працює з пакетами даних фіксованого розміру (batch), які мають розмір batch\_size. Алгоритм ітеративно підганяє вагові коефіцієнти моделі до відповідних міток, які знаходяться у файлі міток. Процес ітерацій називається епохами, яких може бути декілька. Це необхідно тому, що зміна вагових коефіцієнтів може вплинути на відповіді вже налаштованих тестових завдань:

Зображення, що містить текст, електроніка, знімок екрана, програмне забезпечення

Автоматично згенерований опис

Перевіримо модель на тестових даних:

Зображення, що містить знімок екрана, текст, ряд, Шрифт

Автоматично згенерований опис

Збережемо модель:





Завантажимо модель:



Робимо зображення для обробки:

Зображення, що містить гачок

Автоматично згенерований опис Зображення, що містить штатив

Автоматично згенерований опис із середнім рівнем достовірності

Зображення, що містить гачок

Автоматично згенерований опис 

Підготуємо зображення (на прикладі 5):

Зображення, що містить текст, знімок екрана, Шрифт, число

Автоматично згенерований опис

Зробимо передбачення:

Зображення, що містить текст, знімок екрана, Шрифт, ряд

Автоматично згенерований опис

Як бачимо, отриманий результат є правильним.

Перевіримо модель ще на трьох цифрах:

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Зображення, що містить текст, знімок екрана, Шрифт, число

Автоматично згенерований опис

Отримані результати та оцінка точності моделі свідчать про її ефективність у прогнозуванні рукописних цифр. Що допомогло досягнути цього результату - це якість навчального набору даних та оптимально підібрані параметри для класифікації з багатьма класами. Проте, для кращої впевненості, слід протестувати та оцінити модель на більших тестових наборах.

**2)** **Спроектувати i розробити нейронну мережу на основі таких наборів даних імплементованих в Keras:**

**1. Cifar10**

Імпорт даних:

Зображення, що містить текст, знімок екрана, Шрифт, монітор

Автоматично згенерований опис

Підготовка даних:

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Задання текстових назв класів:

Зображення, що містить текст, знімок екрана, Шрифт, ряд

Автоматично згенерований опис

Перші 15 тренувальних зображень:

Зображення, що містить текст, знімок екрана, кінь, колаж

Автоматично згенерований опис

Форматування даних:





Створення моделі:

Зображення, що містить текст, знімок екрана, монітор, програмне забезпечення

Автоматично згенерований опис

Навчання моделі:

Зображення, що містить текст, знімок екрана, монітор, Шрифт

Автоматично згенерований опис

Тестування моделі:

Зображення, що містить текст, знімок екрана, Шрифт, ряд

Автоматично згенерований опис

Як можна побачити з результату, тестова та навчальна точності виявились доволі схожими. Це означає, що модель була розроблена і натренована правильно. Точність моделі близько 70% вказує на те, що наша модель доволі часто робить похибки, проте такий результат варто очікувати, враховуючи що вхідні дані – зображення доволі низької якості – 32 на 32 пікселя.

Перевірка моделі на реальних прикладах:

Візьмемо 3 нових зображення:

Зображення, що містить кінь, знімок екрана, ссавець, просто неба

Автоматично згенерований опис

Протестуємо на них модель:

Зображення, що містить текст, знімок екрана, Шрифт, монітор

Автоматично згенерований опис

Як бачимо, всі зображення були класифіковані правильно. Візьмемо ще одне зображення:

Зображення, що містить текст, знімок екрана, дизайн

Автоматично згенерований опис

Його модель визначила неправильно, переплутавши вантажівку і автомобіль. Загальний результат тестування моделі на нових прикладах дуже схожий з отриманим при тестуванні на початковому наборі, що підтверджує визначену точність моделі.

**2. FMNIST**

Імпорт даних:

Зображення, що містить текст, знімок екрана, програмне забезпечення, Операційна система

Автоматично згенерований опис

Задання текстових назв класів:

Зображення, що містить текст, знімок екрана, Шрифт, ряд

Автоматично згенерований опис

Перші 15 тренувальних зображень:

Зображення, що містить текст, взуття, знімок екрана, одежа

Автоматично згенерований опис

Форматування даних:

Зображення, що містить текст, знімок екрана, Шрифт

Автоматично згенерований опис

Створення моделі:

Зображення, що містить текст, знімок екрана, Шрифт, програмне забезпечення

Автоматично згенерований опис

Навчання моделі:

Зображення, що містить текст, знімок екрана, Шрифт, програмне забезпечення

Автоматично згенерований опис

Тестування моделі:

Зображення, що містить текст, знімок екрана, Шрифт, ряд

Автоматично згенерований опис

Як можна побачити з результату, тестова та навчальна точності виявились доволі схожими. Це означає, що модель була розроблена і натренована правильно. Точність моделі близько 86% вказує на те, що наша модель є доволі точною, проте іноді робить помилки.

Перевірка моделі на реальних прикладах:

Візьмемо 3 нових зображення:

Зображення, що містить сукня, білий, взуття, чорно-білий

Автоматично згенерований опис

Протестуємо на них модель:

Зображення, що містить текст, знімок екрана, Шрифт, число

Автоматично згенерований опис

Як бачимо, перші 2 зображення були розпізнані правильно, а 3 зображення модель сплутала з сумкою. Візьмемо ще 2 приклади:

Зображення, що містить одежа, чобіт, взуття

Автоматично згенерований опис

Результат:

Зображення, що містить текст, знімок екрана, Шрифт, число

Автоматично згенерований опис

Все правильно. Результат тестування моделі на нових прикладах показує, що наша модель здатна розпізнавати нові зображення, проте іноді помиляється.

**3. Висновок**

В даній лабораторній роботі було розглянуто проектування та навчання штучної нейронної мережі для задач класифікації. Для цього було використано бібліотеку Keras та набори даних, які вона містить: mnist, cifar10, fmnist. Створення моделі та обробка даних була виконана за допомогою Python та його бібліотек. Отримані знання були використані для обробки даних зображень написаних вручну чисел, кольорових зображень та зображень одягу. Отримані результати показали, що створені моделі доволі точно можуть класифікувати зображення, найкращий результат показала модель класифікації рукописних чисел, найгірший – модель класифікації кольорових зображень. Більш детальні результати роботи всіх моделей представлені в лабораторній.