Київський національний університет імені Тараса Шевченка

Факультет комп’ютерних наук та кібернетики

Кафедра інтелектуальних програмних систем

**ЗВІТ**

**з виробничої практики**

**Виконав:**

Студент 4-го курсу

Шатохін Максим Сергійович

**Керівник практики:**

від кафедри

Ліндер Ярослав Миколайович

від підприємства

**------------------------------**

Київ - 2020

**ЗМІСТ**

[РОЗДІЛ 1. Постановка задачі 3](#_Toc50951982)

[РОЗДІЛ 2. Результати 4](#_Toc50951983)

[**2.1** **Пошук та збір даних** 4](#_Toc50951984)

[**2.2** **Обробка даних** 4](#_Toc50951985)

[**2.3** **Згорткова нейронна мережа** 4](#_Toc50951986)

[**2.4** **Побудова згорткової нейронної мережі для розпізнавання зображень** 5](#_Toc50951987)

[**2.5** **Навчання нейромережі** 6](#_Toc50951988)

[**2.6** **Під’єднання до додатку Телеграм** 6](#_Toc50951989)

[**2.7** **Фінальний результат** 7](#_Toc50951990)

[ВИСНОВОК 8](#_Toc50951991)

[ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ 9](#_Toc50951992)

# **РОЗДІЛ 1. Постановка задачі**

Задача даної практики полягає у вивченні, дослідженні та подальшому використанні отриманих знань у написанні алгоритму розпізнавання зображень за допомогою згорткових нейромереж для виявлення та відрізнення котів від не котів у мобільному додатку Телеграм.

# **РОЗДІЛ 2. Результати**

## **Пошук та збір даних**

Для навчання нейромережі треба використовувати великий набір даних. Так за результатами безмежної кількості пошуків в інтернеті картиночок із котиками, було знайдено ресурс Kaggle[1]. Він містить багато різноманітних наборів даних на будь-який смак, тому за невеликий час, там був помічений і завантажений набір із дев’яти тисяч фотографій пухнастих. Для протилежного набора ідеально підійшов пошуковий рушій від компанії Google – DataSetSearch[2]. З нього було завантажено даних на стільки ж фотографій із усіляким, що не представляє собою котів.

## **Обробка даних**

Для усіх завантажених фотографій був розроблений конвертер, що обрізає їх по центру, та потім трансформує до розміру 256х256 (двісті п’ятдесят шість на двісті п’ятдесят шість). Також він сортує їх за папками та перейменовував за порядком, для більш зручного використання надалі. Таким чином всі зображення виходять однакового розміру, а потім конвертуються до лінійного масиву , що і потрібно для користування нейронною мережею. Також цей конвертер потім був використаний для аналізатора нових зображень, та припущень по ним.

## **Згорткова нейронна мережа**

Згорткова нейронна мережа (з англ. «convolutional neural network») – це спеціальний вид нейронної мережі для обробки даних, яка в основі має математичне поняття згортки. Її наближена архітектура зображена на рисунку 2.1. Основна суть полягає, що замість загальної операції множення на матрицю використовується принаймні в одному шарі згортка вигляду [3]:



де - результат згортки на позиції ,

- вхідна матриця,

- ядро згортки,

- розмірність ядра згортки.

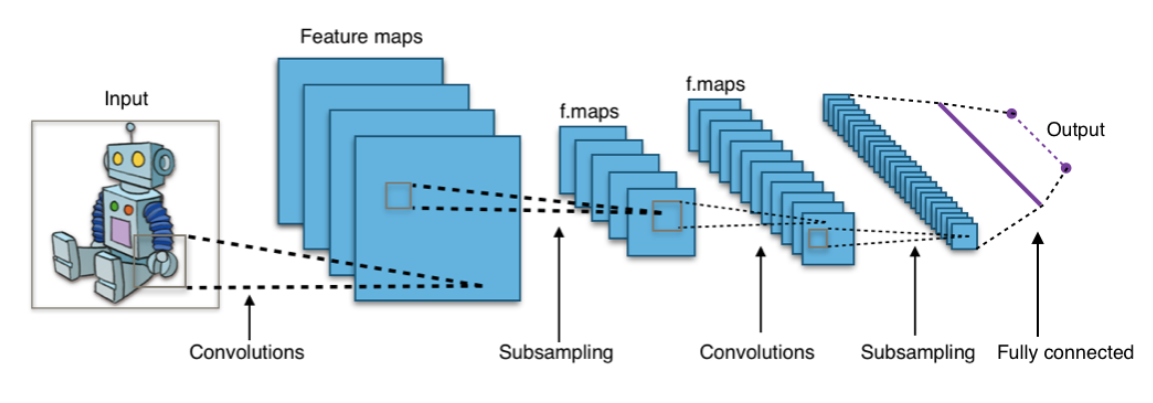


Рис. 2.1. Зображення процесу роботи згорткової нейромережі

## **Побудова згорткової нейронної мережі для розпізнавання зображень**

На вхід нейромережі подається набір фотографій у вигляді масиву. Згорткова мережа складається з п’яти згорткових шарів. Кожен згортковий шар має ядро розміру 3х3 чи 5х5 на передостанньому. Після кожного з перших трьох шарів була операція об’єднання 2х2, та 4х4 після останнього.

Кожний згортковий шар відповідно зберігав по 32, 64, 128, 256, 1 згорток. Останній шар робить перетворення матриці на вектор з якого на виході отримувалося значення. Процес обробки шарів для вхідного зображення розміром 28х28х1 продемонстрований на рисунку 2.2.

Код, який навчив модель, був написаний на мові програмування Python 3.8. Були використані бібліотеки, такі як Tensorflow[4] та Keras[5].

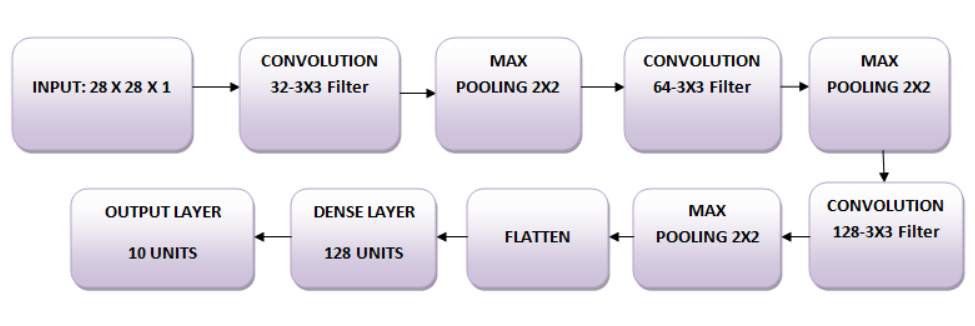


Рис. 2.2 Зображення процесів обробки зображення нейромережею

## **Навчання нейромережі**

Для навчання моделі на чотирьох тисячах фотографій знадобилась ціла ніч, що в процесі написання складало близько трьох годин та до десяти гігабайтів оперативної пам’яті.

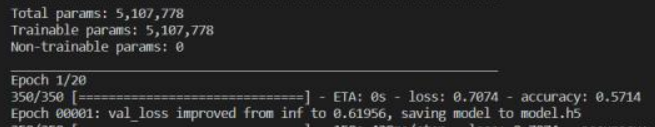


Рис. 2.3 Початок обробки вхідних фотографій нейромережею

На перших ітераціях точність ледве перевищувала п’ятдесят відсотків (рис. 2.3), проте на двадцятій епосі шанс розпізнання кота сягав майже вісімдесяти.

## **Під’єднання до додатку Телеграм**

Оскільки розробляти графічний інтерфейс для зручного користування нейронною мережею було б занадто складно і затратно за часом, було вирішено зробити це шляхом використання BOT-API від Telegram[6]. Він надає можливість надіслати будь яке зображення в чат, і скориставшись командою “/is\_cat” запустити алгоритм розпізнання нейронної мережі (рис. 2.4).

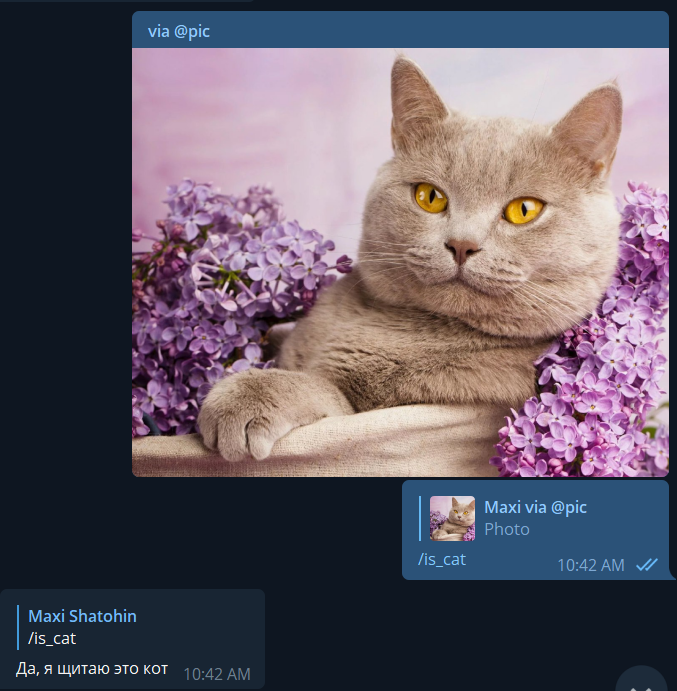


Рис. 2.4 Демонстрація роботи нейронної мережі у додатку

## **Фінальний результат**

Код нейронної мережі було інтегровано до мобільного додатку Телеграм, та протестована багатьма його користувачами. Ви і самі можете скористуватись, надіслав будь-яке зображення до специального боту[7]. Сам код програми та усі початкові дані можна переглянути за посиланням на GitHub[8].

# **ВИСНОВОК**

В результаті виконання практичної роботи було створено засіб, що дає відповідь на одне із найважливіших питань людства, що має можливість заощадити час багатьом адміністраторам каналів із картинками для котів, та людям що їх полюбляють, засіб, що дає відповідь на питання, зображений на фотографії кіт, чи ні.

У ході виконання практичної роботи було досліджено та вивчено процес роботи згорткової нейронної мережі. Створено додаток на її основі. Та переглянуто дев’ять тисяч фотографій котів.

# **ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ**

1. Kaggle [Електронний ресурс], – Режим доступу до ресурсу: https://www.kaggle.com/crawford/cat-dataset.
2. DataSetSearch [Електронний ресурс], – Режим доступу до ресурсу: https://datasetsearch.research.google.com/.
3. Я. Гудфелоу, Глубокое обучение / Я. Гудфелоу, И. Бенджио, А. Курвилль – Litres, 2018. – P. 282-284.
4. Tensorflow [Електронний ресурс], – Режим доступу до ресурсу: <https://www.tensorflow.org/>.
5. Keras [Електронний ресурс], – Режим доступу до ресурсу: <https://keras.io/>;
6. Chastis\_bot [Електронний ресурс], – Режим доступу до ресурсу: <https://t.me/chastis_bot>.
7. Github [Електронний ресурс], – Режим доступу до ресурсу:  [https://github.com/chastis/UniversityStuff/blob/master/Practice/](https://github.com/symphony-of-nirfolio/Weather-anomaly-detection).