УДК 004.42:004.85

Кучер М.О., студент 2 курсу магістратури, спеціальність 122 «Комп'ютерні науки»
Бабаков Р.М. к.т.н., доцент, доцент кафедри інформаційних технологій

АНАЛІЗ РОЗРОБКИ ПРОГРАМНОГО ДОДАТКУ, ЯКИЙ ВИКОРИСТОВУЄ НЕЙРОННІ МЕРЕЖІ

Донецький національний університет імені Василя Стуса, м. Вінниця

Оскільки нейронні мережі все частіше використовуються для різноманітних задач, дедалі більше дослідників ставлять собі за мету створення свого додатку для виконання їх задач, оскільки значна частина дослідників не є програмістами розглянуто етапи розробки програмного додатку, який використовує нейронні мережі.

В якості прикладу проаналізуємо створення додатку для аналізу супутникових зображень на наявність і розташування на них кораблів.

Окрім аналізу теоретичних відомостей і інших складових теоретичного етапу не менш важливим є знаходження датасету та його обробка, так як якість додатку напряму пов'язана не тільки з якістю підібраного чи написаного власноруч рішення для створення моделі, а і від якості взятого для навчання датасету [1].

Розглянемо набір даних «Ships in Satellite Imagery»[2] та проаналізуємо їх обробку.

Для обробки датасету з картинок їх необхідно перетворити в формат матриць, що можна реалізувати, наприклад за допомогою opencv-python пакета, завдяки маніпуляції з картинками отримали датасет з закодованими в ньому фото в трьох каналах кольору.

Оскільки датасет має дисбаланс класів, що може негативно вплинути на навчання мережі, так як мережа буде частіше схилятись до передбачення в бік відсутності корабля, через те, що звичайним вгадуванням, не дивлячись на малюнок вона зможе досягти приблизно 75% точності, дисбаланс необхідно усунути за допомогою, наприклад, аугментації даних.

Оброблені дані прийнято розділяти на три групи: навчальні дані, валідаційні і тестові дані.

Відповідно до назви, навчальні дані використовуються для навчання, а тестові для фінальної оцінки якості моделі, в той час як валідаційні необхідні для оцінки моделі в процесі навчання і використовуються для унеможливлення «запам'ятовування» нейромережею прикладів замість знаходження закономірності.

Використано розділення даних 6:2:2, для тренувальних, валідаційних і тестових вибірок відповідно.

Після аналізу представлених топологій нейромереж слід обрати одну чи декілька найбільш перспективних з точки зору дослідника, в даному випадку залишались моделі сімейств YOLO та RNN, для самого ж додатку було обрано друге сімейство, так як воно краще працює з зображеннями, на аналіз яких є достатньо часу [3].

Розроблена модель, що є різновидом згорткової мережі має 74818 параметрів, більшість з яких є гнучкими для тренування.

```
Epoch 1/45
221/226 [============================>.] - ETA: 0s - loss: 0.4448 - accuracy: 0.7963
Epoch 1: val_accuracy improved from -inf to 0.78394, saving model to model_weights.h5
226/226 [==============================] - 4s 11ms/step - loss: 0.4400 - accuracy: 0.7990 - val_loss: 0.4476 - val_accuracy: 0.7839
Epoch 2/45
226/226 [==============================] - ETA: 0s - loss: 0.2540 - accuracy: 0.9041
Epoch 2: val_accuracy improved from 0.78394 to 0.92384, saving model to model_weights.h5
226/226 [==============================] - 2s 9ms/step - loss: 0.2540 - accuracy: 0.9041 - val_loss: 0.2273 - val_accuracy: 0.9238
Epoch 3/45
226/226 [==============================] - ETA: 0s - loss: 0.1972 - accuracy: 0.9258
Epoch 3: val_accuracy improved from 0.92384 to 0.94205, saving model to model_weights.h5
226/226 [==============================] - 2s 10ms/step - loss: 0.1972 - accuracy: 0.9258 - val_loss: 0.1432 - val_accuracy: 0.9421
Epoch 4/45
222/226 [============================>.] - ETA: 0s - loss: 0.1642 - accuracy: 0.9388
Epoch 4: val_accuracy improved from 0.94205 to 0.96275, saving model to model_weights.h5
226/226 [==============================] - 2s 10ms/step - loss: 0.1641 - accuracy: 0.9391 - val_loss: 0.0997 - val_accuracy: 0.9627
```

Рисунок 1 – Фрагмент навчання нейромережі

Після початкового навчання моделі і імплементації її в більш складну систему може з’ясуватись, що на реальних даних мережа показує себе гірше ніж по показникам навчання. В даному випадку готовій мережі подавались кандидати-області знайдені за допомогою бібліотеки OpenCV, на виході отримано відповідь для кожної області чи є вона кораблем, результатом обробки супутникового зображення є приклад на малюнку 2.

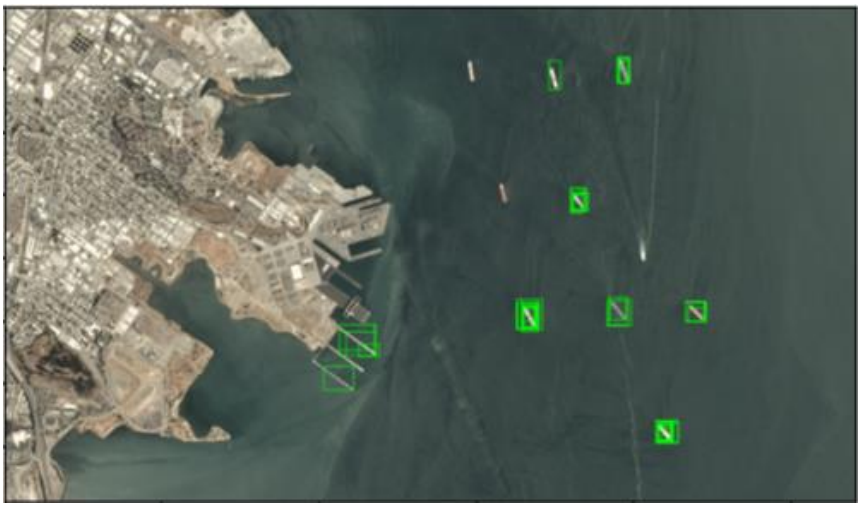

Рисунок 2 – Результат обробки зображення

Аналогічно до середньоквадратичної помилки використовують ще й метрику точності «accuracy», яка продемонстрована на рисунку 2.

На рисунку бачимо результат обробки зображення, бачимо що більшість кораблів успішно виділені, не виділеними залишаються 2 кораблі з 9, крім того є моторний човен, однак його модель і не мала виділяти. Одначе, окрім кораблів

були виділені елементи інфраструктури порту, розглянемо частину оригіналу зображення, для кращого розуміння помилки.

Рисунок 3 – Помилково визначена інфраструктура

З рисунку видно, що інфраструктура досить схожа по формі на кораблі, ймовірно це і стало причиною помилкового визначення, подібні помилки важно вчасно виявляти і аналізувати, проаналізувавши їх – можна додати більше прикладів портової інфраструктури в датасет, для збільшення точності.

Подібне донавчання мережі часто є необхідним, оскільки датасети здебільшого не дають необхідної повноти уявлення про дані для нейронної мережі, особливо це стосується невеликих датасетів один з яких і був продемонстрований.

Після остаточного навчання моделі її можна імплементувати в додаток, додано інтерфейс через бібліотеку Qt, яка добре підходить для створення інтерфейсів.

Отже, продемонстровані етапи розробки додатку, який використовує нейронну мережу, що дозволяє дослідникам краще розуміти процес створення готового рішення на базі їх моделі, що було продемонстровано на прикладі обраного датасету.

Список використаних джерел

1. *Preparing Your Dataset for Machine Learning: 10 Basic Techniques That Make Your Data Better [Електронний ресурс]. Режим доступу до ресурсу: https://www.altexsoft.com/blog/datascience/preparing-your-dataset-for-machine-learning-8-basic-techniques-that-make-your-data-better/*
2. *Ships in Satellite Imagery [Електронний ресурс] // Online Journal for Research and Education. – 2021. – Режим доступу до ресурсу: https://www.kaggle.com/datasets/rhammell/ships-in-satellite-imagery*
3. *R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms [Електронний ресурс]. Режим доступу до ресурсу: https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e*