Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра прикладної математики
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Математичні моделі і методи звукової класифікації об'єктів
(тема)

Виконав:
студент _2_ курсу, групи _ПМм-20-1
Кит М.О.
(прізвище, ініціали)
Спеціальність
113 Прикладна математика
(код і повна назва спеціальності)
Тип програми _освітньо-професійна
(освітньо-професійна або освітньо-наукова)
Освітня програма
Прикладна математика
(повна назва освітньої програми)
Керівник доц. Єсілевський В.С.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ПМ _____ Тевяшев А.Д.
(підпис) (прізвище, ініціали)

2021 р.

Харківський національний університет радіоелектроніки

Факультет _інформаційно-аналітичних технологій та менеджменту
Кафедра _прикладної математики
Рівень вищої освіти _другий (магістерський)
Спеціальність _113 Прикладна математика
(код і повна назва)
Тип програми _освітньо-професійна
(освітньо-професійна або освітньо-наукова)
Освітня програма _Прикладна математика
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри ПМ _____
(підпис)
"____" _____ 2021 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові _Киту Микиті Олександровичу
(прізвище, ім'я, по батькові)

1. Тема роботи _Математичні моделі і методи звукової класифікації об'єктів

затверджена наказом по університету від 05 листопада 2021 р. № 1641 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 10 грудня 2021 р.

3. Вихідні дані до роботи _набір аудіо зразків співу птахів

4. Перелік питань, що потрібно опрацювати в роботі
1. Аналіз предметної області
2. Вибір і обґрунтування методу розв'язання
3. Програмна реалізація
4. Результати обчислювального експерименту
5. Аналіз можливих застосувань

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій

1. Актуальність теми роботи
2. Постановка задачі
3. Аналіз предметної області
4. Метод чисельного аналізу
5. Результати обчислювального експерименту

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Підбір та вивчення технічної літератури за темою роботи	8 – 14 листопада 2021 р.	виконано
2	Вибір та обґрунтування методу	15 – 21 листопада 2021 р.	виконано
3	Розробка алгоритму і програми	22 – 28 листопада 2021 р.	виконано
4	Проведення аналітичних досліджень та розрахунків	29 листопада – 5 грудня 2021 р.	виконано
5	Робота над текстом пояснювальної записки	6 – 9 грудня 2021 р.	виконано
6	Представлення роботи на рецензію в ЕК	10 грудня 2021 р.	виконано

Дата видачі завдання _8_ листопада 2021 р.

Студент _____
(підпис)

Керівник роботи _____ доц. Єсілевський В.С.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 63 с., 18 рис., 1 дод., 13 джерел.

МАШИННЕ НАВЧАННЯ, НЕЙРОННА МЕРЕЖА, ЗГОРТКОВА НЕЙРОННА МЕРЕЖА, КЛАСИФІКАЦІЯ ЗОБРАЖЕНЬ, СПІВ ПТАХІВ.

Об’єкт дослідження – набір співів птахів.

Мета роботи – реалізація програмного забезпечення для звукової класифікації птахів.

Методи дослідження – згорткові нейронні мережі.

У роботі проведений аналіз стану проблеми звукової класифікації об’єктів за допомогою нейронних мереж. З використанням мови програмування Python та середовища розробки Jupyter Notebook був створений програмний продукт та проведений порівняння двох моделей, а саме Xception та EfficientNet, та отриманий результат тестування, також була виконана робота по аналізу можливих застосувань.

ABSTRACT

Introductory note: 63 pages, 18 figures, 1 appendix, 13 sources.

MACHINE LEARNING, NEURAL NETWORK, CONVOLUTIONAL NEURAL NETWORK, IMAGE CLASSIFICATION, BIRD SONG.

Object of research – set of bird singing.

Purpose of work – implementation of software for sound classification of birds.

Methods of research – convolutional neural network.

The paper analyzes the state of the problem of sound classification of objects using neural networks. Using the Python programming language and the Jupyter Notebook development environment, a software product was created and two models, namely Xception and EfficientNet, were compared and tested, and work was done to analyze possible applications.

ЗМІСТ

ВСТУП

Актуальність теми. У даній роботі розглядається проблема звукової класифікації об'єктів та її розв’язання за допомогою нейронних мереж. Оскільки ця задача дуже актуальна у наш час, оцінка присутності та чисельності птахів є важливою для моніторингу окремих видів, а також загального стану здоров’я екосистеми. Багато птахів найлегше розпізнаються за їхніми звуками, тому пасивний акустичний моніторинг є дуже доцільним. Проте акустичний моніторинг часто стримується практичними обмеженнями, такими як необхідність ручної конфігурації, використання звукових бібліотек, низька точність, низька надійність та обмежена здатність узагальнювати нові акустичні умови. Завдяки сучасному машинному навчанню, включаючи глибоке навчання, можна досягти загального акустичного виявлення птахів дуже високі показники пошуку даних віддаленого моніторингу без ручного перекалібрування та попереднього навчання детектора для цільового виду або акустичних умов у цільове середовище.

Мета і завдання кваліфікаційної роботи. Метою кваліфікаційної роботи є аналіз проблеми звукової класифікації об’єктів та аналіз існуючих методів. Для досягнення поставленої мети необхідно виконати наступні завдання:

– провести аналіз стану проблеми звукової класифікації об’єктів;

– побудувати модель згорткових нейронних мереж та розробити програму для звукової класифікації об’єктів;

– провести обчислювальний експеримент та дослідити результати.

Об’єктом дослідження є набір співів птахів.

Предметом дослідження математична модель класифікація співів птахів.

Методи дослідження. У кваліфікаційній роботі використовується технологія згорткових нейронних мереж.

1 АНАЛІЗ СТАНУ ПРОБЛЕМИ И МЕТОДІВ ЗВУКОВОЇ КЛАСИФІКАЦІЇ ОБ’ЄКТІВ

1.1 Математичні моделі звукової класифікації об'єктів

1.1.1 Нейронні мережі та класифікація звукових файлів

Штучні нейронні мережі набули широкого поширення у трьох хвилях, викликаних:

а) алгоритмом перцептрону у 1957 році [1];

б) алгоритмом зворотного поширення помилки у 1986 році [2];

в) успіхом глибокого навчання у розпізнаванні мови та класифікації зображень у 2012 році [3].

Це призвело до відродження глибокого навчання, що включає, наприклад, глибокі прямолінійні нейронні мережі, згорткові та мережі з довготривалою пам'яттю. У цій "глибокій" парадигмі архітектури з великою кількістю параметрів навчаються на основі величезної кількості даних, використовуючи останні досягнення в галузі паралельних обчислень. Нещодавній сплеск інтересу до глибокого навчання дозволив знайти практичне застосування у багатьох галузях обробки сигналів, часто перевершуючи традиційні методи у великих масштабах. У ході цієї останньої хвилі глибоке навчання спочатку набуло поширення в обробці зображень, а потім почало широко застосовуватись в обробці мови, музики та звуків навколишнього середовища, а також у багатьох інших областях, таких як вивчення квантової хімії, пошук ліків, обробка природної мови та рекомендаційні системи. В результаті методи обробки звукових сигналів, що використовувалися раніше, такі як гаусівські моделі, приховані марківські моделі і невід’ємна матрична факторизація, часто перевершували моделі глибокого навчання в тих випадках, коли було достатньо даних. Хоча багато методів глибокого навчання були запозичені з обробки зображень, між цими областями

існують важливі відмінності, які вимагають особливого розгляду аудіо. Необроблені звукові зразки утворюють одновимірний сигнал часового ряду, який відрізняється від двовимірних зображень. Аудіо сигнали зазвичай перетворюються на двомірні частотні представлення для обробки, але дві осі, часова та частотна, не є однорідними, як горизонтальна та вертикальна осі у зображенні. Зображення є миттєвими знімками об'єкта і часто аналізуються як єдине ціле або фрагментами з невеликими обмеженнями по порядку. Однак аудіо сигнали повинні аналізуватися послідовно і в хронологічному порядку. Ці властивості призвели до появи специфічних моделей в області аудіо.

1.1.2 Категорії задач роботи з аудіо сигналами

Задачі, що розглядаються в даному пункті, можна розділити на різні категорії в залежності від типу цілі, яку необхідно передбачити за вхідними даними, які завжди є часовим рядом аудіо зразків.

По-перше, ціль може бути однією глобальною міткою або локальною міткою за часовий крок, або послідовністю міток довільної довжини. По-друге, кожна мітка може бути одним класом, набором класів чи числовим значенням. Надалі називатимемо і наводитимемо приклади різних аналізованих комбінацій. Передбачення однієї глобальної мітки класу називається класифікацією послідовності. Такою міткою класу може бути передбачена мова, диктор, музичний ключ або акустична сцена, взяті із наперед визначеного набору можливих класів. У класифікації послідовності з кількома мітками ціль – це підмножина множини можливих класів. Наприклад, ціль може складатися з декількох акустичних подій. Класифікація за декількома мітками може бути особливо ефективною, коли класи залежать один від одного. У регресії послідовності ціллю є значення безперервного діапазону. Так можна сформулювати оцінку музичного темпу чи передбачення наступного аудіо зразка. Важливо, що проблеми регресії завжди можуть бути дискретизовані і перетворені на проблеми класифікації:

наприклад, коли аудіо зразок квантований на 8 біт, передбачення зразка є проблемою класифікації з 256 класами. При передбаченні мітки за крок у часі кожен крок у часі може охоплювати постійне число аудіо зразків, тому довжина цільової послідовності становить частину довжини вхідний послідовності. Знову ж таки, ми можемо розрізняти різні випадки. Класифікація за часовий крок тут називається маркуванням послідовності. Прикладами є анотування акордів та визначення вокальної активності. Виявлення подій спрямоване на передбачення часових точок настання подій, таких як зміна диктора або набуття ноти, що можна сформулювати як задачу бінарного маркування послідовності: на кожному кроці розрізняти наявність та відсутність події. Регресія на кожному часовому кроці генерує безперервні прогнози, які можуть бути відстанню до джерела звуку, що рухається, або висотою голосу, або поділом джерел.

При перетворенні послідовності довжина цільової послідовності не є функцією довжини вхідної послідовності. Не існує усталених термінів для поділу класифікації, багатозначної класифікації та регресії. Прикладами можуть бути переклад мови в текст, транскрипція музики або переклад мови. Синтез аудіо можна подати як задачу перетворення послідовності або регресії, яка передбачає аудіо зразки за послідовністю умовних змінних. Оцінка подібності аудіо – це регресійна задача, в якій безперервне значення надається парі аудіо сигналів, можливо, різної довжини.

Створення відповідного представлення ознак і розробка відповідного класифікатора цих ознак часто розглядаються як окремі проблеми у обробці звуку. Недоліком такого підходу є те, що розроблені ознаки можуть бути не оптимальними для задачі класифікації. Глибокі нейронні мережі (ГНМ) можна представити як такі, що виконують витяг ознак спільно з оптимізацією цілі, такі, як класифікація. Наприклад, для розпізнавання мови, активації на нижніх шарах ГНМ можна розглядати як адаптовані до диктора ознаки, а активації на верхніх шарах ГНМ – як дискримінацію на основі класу.

Протягом десятиліть кепстральні коефіцієнти mel-частот використовувалися, як домінуюче уявлення акустичних ознак для задач аналізу звуку. Це спе-

ктри величин, спроектовані на зменшений набір частотних смуг, перетворені на логарифмічні величини, а також приблизно вибілені та стислі за допомогою дискретного косинусного перетворення. У моделях глибокого навчання було показано, що останнє непотрібне або небажане, оскільки воно видаляє інформацію і руйнує просторові відносини. Якщо його опустити, то вийде спектр log-mel, найпопулярніша характеристика в аудіо областях. Банк фільтрів mel для проектування частот натхненний слуховою системою людини та фізіологічними даними про сприйняття мови. Для деяких завдань краще використовувати уявлення, яке фіксує транспонування як переклад. Транспонування тону полягає у масштабуванні основної частоти та обертонів на загальний коефіцієнт, який стає зсувом у логарифмічній шкалі частот. Спектр із постійною добротністю дозволяє отримати таку шкалу частот за допомогою відповідного банку фільтрів. Спектрограма – це часова послідовність спектрів. Як і в природних зображеннях, сусідні біни спектрограми природних звуків за часом та частотою корелюють. Однак, в силу фізичних особливостей виробництва звуку, існують додаткові кореляції для частот, кратних одній і тій самій базовій частоті. Щоб просторово-локальна модель могла врахувати їх, можна додати третій вимір, який безпосередньо дає величини гармонійних рядів. Крім того, на відміну від зображень, розподіл значень істотно різняться між частотними смугами. Для боротьби з цим спектрограм можуть бути стандартизовані окремо для кожного діапазону. Розмір вікна для обчислення спектрів дозволяє вибрати часову роздільну здатність (короткі вікна) проти частотної роздільної здатності (довгі вікна). Як для спектрів log-mel, так і для спектрів constant-Q можна використовувати більш короткі вікна для високих частот, але це призводить до неоднорідно розмитих спектрограм, непридатним для просторових локальних моделей. Альтернативні варіанти включають обчислення спектрів з різними довжинами вікон, які проектуються вниз на ті ж частотні смуги і розглядаються як окремі канали.

Щоб не покладатися на розроблений банк фільтрів, було запропоновано різні методи, що дозволяють спростити процес отримання ознак і відкладання

його до навчання статистичної моделі на основі даних. Замість трикутних фільтрів з рознесеними трикутниками були запропоновані фільтри, керовані даними та використовування спектру величини повного дозволу тобто безпосередньо використовувати необроблене представлення форми сигналу аудіо сигналів, як вхідні дані та вивчити керовані даними фільтри спільно з рештою мережі для цільових задач. Таким чином, фільтри, що навчаються, безпосередньо оптимізуються під поставлену задачу. Нижні шари моделі розроблені таким чином, щоб імітувати обчислення спектра log-mel, але всі параметри фільтра навчаються на основі даних. В поняття банку фільтрів відкидається, навчаючи причинно-наслідкову регресійну модель зразків форми сигналу у часовій області без будь-яких попередніх знань людини. Аудіо сигнал, представлений у вигляді послідовності або кадрів необробленого звуку, або розроблених людиною векторів ознак, матриць (наприклад, спектрограм) або тензорів (наприклад, складених спектрограм) може бути проаналізований різними моделями глибокого навчання. Як і в інших областях, таких як обробка зображень, аудіо для підвищення здатності до моделювання зазвичай використовується кілька шарів: прямолінійний, згортковий і рекурентний (наприклад, LSTM).

1.2 Огляд методів звукової класифікації об’єктів

Convolutional neural network (CNN): CNN [4] засновані на згортці вхідних даних з ядрами, що навчаються. У разі спектральних вхідних ознак зазвичай використовується 1-мірна часова згортка або 2-мірна частотна згортка, у той час як для вихідних сигналів застосовується 1-мірна згортка в часовій області. Згортковий шар зазвичай розраховує кілька карт ознак (каналів), кожна з яких заснована на відповідному ядрі. Пулінг-шари, додані поверх цих згорткових шарів, можуть використовуватися для зниження дискретизації отриманих карт ознак. CNN часто складається з ряду конволюційних шарів, що чергуються з шарами пулінгу, за якими слідує один або кілька щільних шарів. Для марку-

вання послідовності щільні шари можуть бути опущені щоб отримати повністю конволюційну мережу (FCN). Рецептивне поле (кількість зразків чи спектрів, що у обчисленні передбачення) CNN визначається її архітектурою. Його можна збільшити, використовуючи більші ядра або укладаючи більше шарів. Особливо для вихідних сигналів з високою частотою дискретизації досягнення достатнього розміру рецептивного поля може призвести до великої кількості параметрів CNN та високої обчислювальної складності. В якості альтернативи можна використовувати розширену згортку (також звану згорткою з отворами), яка застосовує фільтр згортки до області, що перевищує довжину фільтра, вставляючи нулі між коефіцієнтами фільтра. Стек розширених пакунків дозволяє мережам отримувати дуже великі рецептивні поля за допомогою всього декількох шарів, зберігаючи при цьому вхідну роздільну здатність, а також ефективність обчислень.

Рекурентні нейронні мережі (RNN): Ефективний розмір контексту, який може бути змодельований CNN, обмежений навіть при використанні розширених пакунків. RNN [5] використовують інший підхід для моделювання послідовностей: вони обчислюють вихід для часового кроку на основі вхідних даних на цьому кроці та їх прихованого стану на попередньому кроці. Це моделює часову залежність входів та дозволяє рецептивному полю простягатися на невизначений час у минуле. Для автономних додатків двонаправлені РНМ використовують другу рекурсію у зворотному порядку, розширюючи рецептивне поле в майбутнє.

Порівняно зі звичайними ПММ (Приховані Марківські моделі), при лінійному зростанні числа рекурентних прихованих одиниць в РНМ з ядрами "все до всіх" кількість станів зростає експоненційно, в той час як час навчання або виведення зростає максимум квадратично. РНМ можуть страждати від зникаючих градієнтів під час навчання. Для розв’язання цієї проблеми було розроблено безліч варіантів. Довгострокова пам'ять (LSTM) використовує механізм стробування та комірки пам’яті для пом’якшення потоку інформації та полегшення проблем із градієнтом. Додавання рекурентних шарів і розріджені реку-

рентні мережі були визнані корисними при синтезі звуку. Крім використання для моделювання часових послідовностей, LSTM були розширені для моделювання аудіо сигналів у часовій та частотній областях. Частотні LSTM (F-LSTM) та частотні LSTM (TF-LSTM) були представлені як альтернатива CNN для моделювання кореляцій за частотою. На відміну від CNN, F-LSTM враховують трансляційну інваріантність за допомогою локальних фільтрів та рекурентних зв'язків. Вони не вимагають операцій об'єднання та краще адаптуються до різних типів вхідних ознак. TF-LSTM розгортаються як за часом, так і частотою та можуть використовуватися для моделювання спектральних і часових змін за допомогою локальних фільтрів і рекурентних зв'язків. TF-LSTM перевершують CNN у деяких завданнях, але вони гірше піддаються розпаралелювання і тому працюють повільніше. Як альтернатива RNN можуть обробляти вихід CNN, формуючи конволюційну рекурентну нейронну мережу (CRNN). У цьому випадку конволюційні шари отримують локальну інформацію, а рекурентні шари об'єднують її в більш тривалому часовому контексті.

Моделі "послідовність-послідовність": Модель "послідовність-послідовність" [6] безпосередньо перетворює вхідну послідовність у вихідну. Багато задач обробки звуку насправді є задачами перетворення послідовності на послідовність. Однак через велику складність задач обробки звуку звичайні системи зазвичай поділяють задачі ряд підзадач і розв’язують кожну незалежно. Якщо взяти, як приклад, розпізнавання мови, то основна задача полягає у перетворенні вхідних часових аудіо сигналів у вихідну послідовність слів. Але традиційні системи ASR складаються з окремих компонентів акустики, вимови та моделювання мови, які зазвичай навчаються незалежно один від одного. Зі збільшенням моделюючої здатності моделей глибокого навчання, зростає інтерес до створення наскрізних систем, що навчаються безпосередньо, які безпосередньо відображають вхідний аудіо сигнал в цільові послідовності. Ці системи навчаються для оптимізації критеріїв, пов'язаних із кінцевою метрикою оцінки (наприклад, коефіцієнт помилок у словах для систем ASR). Такі моделі послідовність-послідовність є повністю нейронними і використовують перетво-

рювачі кінцевих станів, лексикон чи модулі нормалізації тексту. Акустичні, вимовні та мовні компоненти моделювання навчаються разом в одній системі. Це значно спрощує процес навчання порівняно з традиційними системами: не потрібно завантажувальна вибірка з дерев рішень, що генеруються окремою системою. Крім того, оскільки моделі навчаються для прямого передбачення цільових послідовностей, процес декодування також спрощується.

Однією з таких моделей є часова конекціоністська класифікація (ЧКК). Ця модель вводить символ пробілу для узгодження довжини вихідної послідовності з вхідною послідовністю та інтегрує всі можливі способи вставки пробілів для спільної оптимізації вихідної послідовності замість кожної окремої вихідної мітки. Базова модель ЧКК була розширена для включення окремого компонента мовної рекурентної моделі, званого рекурентним нейромережевим перетворювачем (RNNT). Моделі на основі уваги, які вивчають вирівнювання між вхідними та вихідними послідовностями спільно з цільовою оптимізацією, стають дедалі популярнішими. Серед різних моделей "послідовність-послідовність" модель listen, attend and spell (LAS) показала найкращі результати порівняно з іншими.

Генеративно-змагальна мережа (GAN): GANs – це генеративні моделі без контролю, які вчяться створювати реалістичні зразки заданого набору даних із низькорозмірних випадкових прихованих векторів. GAN [7] складаються з двох мереж, генератора та дискримінатора. Генератор зіставляє латентні вектори, взяті з деякої заздалегідь відомої множини, зі зразками, а завдання дискримінатора - визначити, чи даний зразок є справжнім або підробленим. Ці дві моделі протиставляються одна одній у рамках змагального процесу. Незважаючи на успіх GANs для синтезу зображень, їх використання у звуковій області було обмежене. GAN були використані для поділу джерел, перетворення музичних інструментів та покращення мови для перетворення зашумленого мовлення в деноізовані версії.

1.3 Змістовна та формальна постановка задачі

Розпізнавання звуків птахів у тій чи іншій місцевості може дати важливу інформацію про середовище проживання. Оскільки птахи знаходяться високо в харчовому ланцюзі, вони є відмінними індикаторами погіршення якості довкілля та забруднення. Моніторинг стану та тенденцій біорізноманіття в екосистемах – задача не проста. При правильному виявленні та класифікації звуків за допомогою машинного навчання, дослідники можуть покращити свої можливості щодо відстеження стану та тенденцій біорізноманіття у важливих екосистемах, що дозволить їм краще підтримувати глобальні зусилля щодо збереження природи.

В якості вхідних даних використовуються аудіо фрагменти, представлені у форматі WAV. Звукові хвилі оцифровуються шляхом вибірки дискретних інтервалів, відомих як частота дискретизації. Кожен семпл є амплітудою хвилі в певному часовому інтервалі, де глибина в бітах визначає на скільки деталізованим буде семпл. Процес обробки звуку включає вилучення акустичних характеристик, що відносяться до поставленої задачі, за якими слідують схеми прийняття рішень, які включають виявлення, класифікацію та поєднання знань. Для цього буде використовуватися спектрограма. Спектрограма – це візуальний спосіб представлення рівня або гучності сигналу в часі на різних частотах, присутніх у формі хвилі. Зазвичай зображується у вигляді теплової карти. Насамперед потрібно перетворити аудіо файли на зображення формату PNG (спектрограми). Використовуючи спектрограми, як вхідні дані, ви можна використовувати класичні підходи до класифікації зображень (наприклад, згорткові нейронні мережі), також можна розділити вхідний звук на кадри розміром близько 20 мс-100 мс (залежно від необхідного часового розширення) та перетворити ці кадри на спектрограми. Згорткові мережі також можуть бути поєднані з рекурентними одиницями для обліку більшого часового контексту. Та на виході моделі ми будемо отримувати вірогідність належності до класів, а на основі цього будемо робити висновки про класифікацію.

Розв’язання задачі буде здійснюватися у декілька етапів:

а) пошук даних;

б) аналіз вхідних даних;

в) попередня обробка даних;

г) тренування нейронної мережі;

д) тестування нейронної мережі.

Нехай X – множина об’єктів, а Y – скінченна множина номерів класів. Для задачі класифікації співів птахів об’єктами будуть зображення спектрограм, для яких відомі номери класів. Існує невідома цільова залежність, тобто відображення:

$$y^* : X \to Y,$$

значення якої відомі лише на об’єктах, тобто зображеннях, скінченної навчальної вибірки:

$$X^m = (x_1, y_1), \ldots, (x_m, y_m).$$

Задача класифікації полягає в тому, щоб по навчальній вибірці X^m побудувати функцію-алгоритм розв’язку вигляду:

$$a : X \to Y,$$

яка апроксимує залежність y^*. Фунцкція розв’язку повинна правильно класифікувати здатний класифікувати довільний об’єкт та усі об’єкти з множин X,тобто таке співвідношення y^* та X^m, де $y^*(x_1) = y_1$.

1.4 Постановка задач дослідження

Спираючись на аналіз проблеми, постановку задачі та досвід вивчення розв’язання задачі, сформулюємо задачі кваліфікаційної роботи:

– на основі згорткових нейронних мереж побудувати модель для класифікації звуків;

– реалізувати алгоритм мовою програмування Python та візуалізувати за допомогою графічної бібліотеки Matplotlib;

– провести обчислювальний експеримент для набору нових даних.

2 ВИБІР ТА ОБҐРУНТУВАННЯ МЕТОДУ РОЗВ’ЯЗАННЯ

2.1 Попередня обробка даних

Попередня обробка даних використовується для досягнення найкращих результатів від моделі. Тобто формат даних повинен бути відповідним. Деякі конкретні моделі машинного навчання потребують інформації в певному форматі. Через обробку даних результат буде відрізнятися від результатів, які будуть отримані без обробки або з неправильною обробкою. Етапи попередньої обробки даних:

– нормалізація, тобто поділення аудіо файлів на рівні проміжки;

– фільтрування частот;

– конвертація звукових доріжок у mel-спектрограми;

– поділ даних на тренувальний та тестовий.

2.1.1 Нормалізація даних

Так як зазвичай у реальних задачах ідеальних даних не буває, їх треба зводити до однієї шкали. Багато алгоритмів машинного навчання працюють краще, коли функції знаходяться у відносно схожих межах та близькі до нормального розподілу. Для роботи зі звуком, скоріш за все, аудіо доріжки будуть мати різну тривалість, а для подальшого перетворення потрібно розділити дані на рівні проміжки, щоб уникнути зжатих даних. Можна сказати, що масштабування даних завжди необхідно. Чим довша довжина аудіо, з якого створюється спектрограма, тим більше інформації ви отримаєте в зображенні. Якщо ваші дані мають багато шуму або тиші, є ймовірність, що відрізок звуку не вловить потрібну інформацію.

У цифровому звукозаписі під нормалізацією звуку розуміється процес

вирівнювання гучності звукового сигналу щодо будь-якого стандарту, наприклад гучності іншого звукового сигналу.

Пікова нормалізація – це спосіб нормалізації, при якому рівень звукового сигналу підіймається до максимально можливого значення для цифрового звуку без появи спотворень. В даному випадку орієнтиром служить рівень його найвищого піку. Цей спосіб дозволяє повністю виключити кліппінг, однак, якщо в звуковому файлі є хоча б один пік, сильно виділяється із загальної сигналограми звукового сигналу, то нормалізація за його рівнем може привести до того, що звуковий сигнал залишиться досить тихим, хоч і звук, на який орієнтувалися при нормалізації буде цілком гучним. Величина звуку при даному способі зазвичай вимірюється у відсотках.

RMS нормалізація (нормалізація гучності) – це нормалізація за середньоквадратичним значенням рівня звуку в файлі. Повна протилежність піковій нормалізації. При цьому способі величина звуку вимірюється в децибелах. Для людського вуха цей спосіб підходить найбільше, однак при великих значеннях гучності можливий кліппінг. Щоб від нього повністю позбутися, фахівці рекомендують нормалізувати звук до значення в 89 децибел, однак для деяких сучасних слухачів воно може здатися занадто тихим. Також слід враховувати, що якщо звукові файли мають різний динамічний діапазон, то на слух вони можуть звучати не однаково голосно навіть з однаковими значеннями RMS.

2.1.2 Фільтрування даних

High-Pass фільтр, високочастотний пропускний фільтр або фільтр високих частот (ФВЧ) – інструмент, що відрізає всі низькі частоти після певної позначки, яка називається частотою зрізу (точка обрізки). Під час роботи фільтр усуває все, що виходить за вибрану точку обрізки, але не торкається сигналу вище. Основне призначення високочастотних фільтрів – обмежити низькочастотний контент, який не потрібний. Для налаштування параметрів роботи вико-

ристовується лише один параметр – точка обрізки, що задає конкретну межу спектра, після якої відбувається відсікання частот.

2.1.3 Конвертація даних

Аудіосигнал складається з кількох одночастотних звукових хвиль. При взятті відліків сигналу у часі фіксуємо лише результуючі амплітуди. Перетворення Фур'є – це математична формула, яка дозволяє нам розкласти сигнал на окремі частоти та амплітуду частоти. Іншими словами, він перетворює сигнал із тимчасової області на частотну. Результат називається спектром. Mel-спектрограми [9] обчислюють за звичайною спектрограмою, побудованою за допомогою віконного перетворення Фур’є:

$$F\left(k,\, m\right) = \sum_{n=2}^{L-1} x\left[n + m\right] w[n] e^{-i\frac{2\pi}{L}kn} \quad .$$

Суть цієї операції у послідовному застосуванні перетворення Фур’є до коротких шматочків сигналу, домноженим на деяку віконну функцію. Результат застосування віконного перетворення це матриця, де кожен стовпець є спектром короткої ділянки вихідного сигналу, приклад на рис. 2.1:

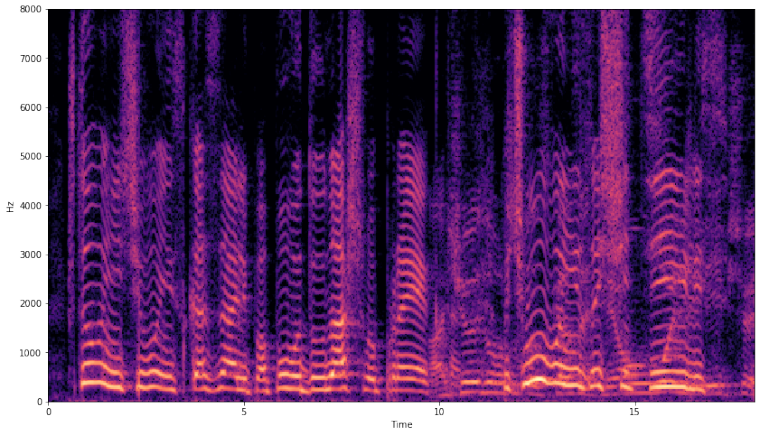

Рисунок 2.1 – Приклад спектрограми

Експерименти вчених показали, що людське вухо чутливіше до змін звуку на низьких частотах, ніж високих. Тобто якщо частота звуку зміниться зі 100 Гц на 120 Гц, людина з дуже високою ймовірністю помітить цю зміну. А от якщо частота зміниться з 10000 Гц на 10020 Гц, цю зміну навряд чи зможе вловити. У зв'язку з цим була запроваджена нова одиниця виміру висоти звуку – mel. Вона заснована на психофізіологічному сприйнятті звуку людиною, і логарифмічно залежить від частоти:

$$mel = 1127.01048 \ln\left(1 + \frac{freq}{700}\right).$$

Власне, mel-спектрограма – це звичайна спектрограма, де частота виражена не в Гц, а в mel. Перехід до mel здійснюється за допомогою застосування mel-фільтрів до вихідної спектрограми. Mel-фільтри являють собою трикутні функції, рівномірно розподілені на mel шкалі. Як приклад на рис. 2.2 зображено 10 mel-фільтрів:

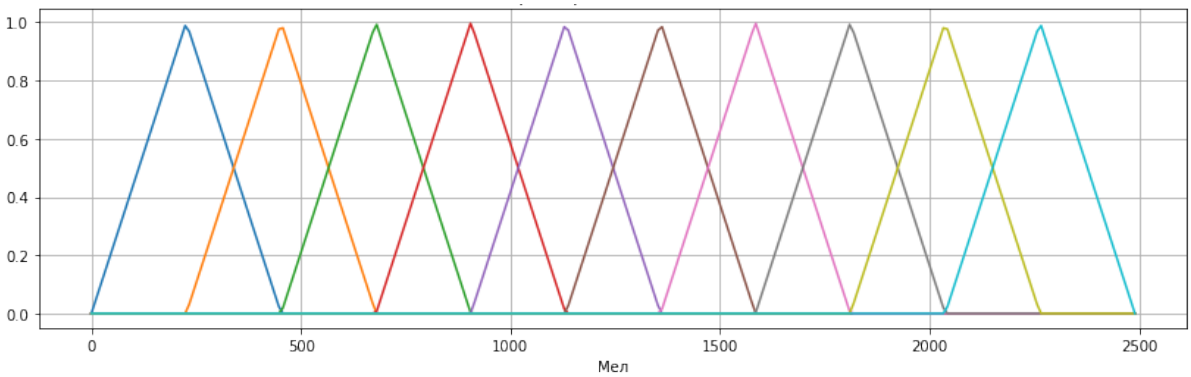

Рисунок 2.2 – Приклад mel-фільтрів

При переведенні в частотну шкали, як ті самі виглядатимуть фільтри зображено на рис 2.3:

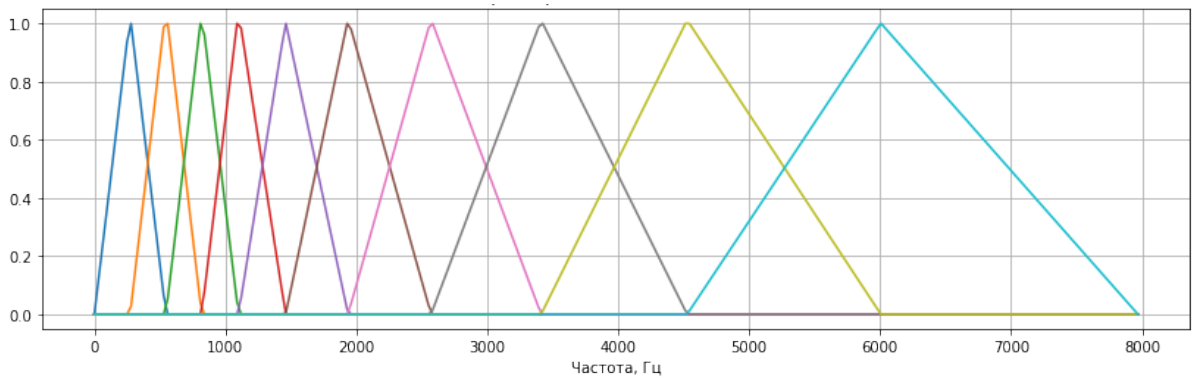

Рисунок 2.3 – Приклад mel-фільтрів переведених в Гц

Кожен стовпець вихідної спектрограми скалярно множиться на кожен mel-фільтр (розміщений на частотній шкалі), після чого виходить вектор чисел, за розміром, що дорівнює кількості фільтрів. В результаті таких перетворень значення з низьких частот спектрограми залишаються практично незмінними на mel-спектрі, а у високих частотах відбувається усереднення значень більш широкого діапазону. Резюмуючи все вищесказане: на mel-спектрограмі зберігається більше інформації, яка добре сприймається і відрізняється людиною, ніж на стандартній спектрограмі. Іншими словами, таке уявлення звуку більше сфокусовано на низьких частотах, і менше на високих.

2.1.4 Поділ даних на тренувальний та тестовий набори

Основна мета поділу набору даних на навчальний і тестовий — мати можливість тренувати модель на відомих (навчальних) даних і оцінювати її на невідомих (тестових). Втрачається деяка цінна інформація в процесі поділу даних і алгоритми можуть скористатися цим. Тому для тестового набору візьміть дуже мало даних (зазвичай 25-30%), наприклад, як показано на рис. 2.4 Для дуже великих колекцій відсоток тестових даних може бути значно зменшений (1-10%).

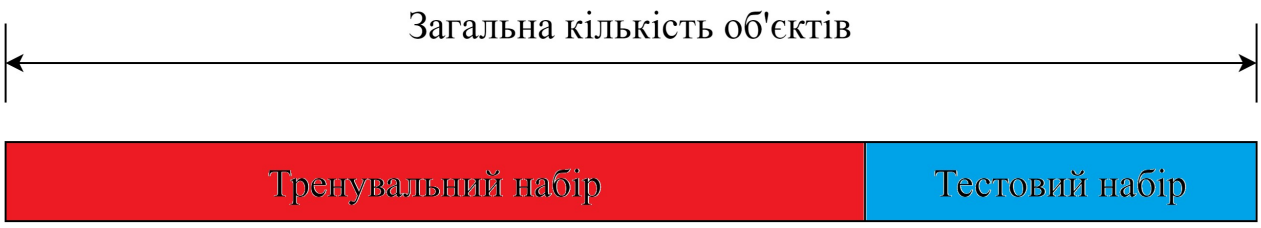

Рисунок 2.4 – Приклад поділу набору даних

2.2 Згорткові нейронні мережі

Згорткова нейронна мережа (ЗНМ) – тип багатошарової нейронної мережі, яка свою назву отримала за назвою операції – згортка, яка часто використовується для обробки зображень та може бути описана наступною формулою

$$(f \times g)[m,\ n] = \sum_{k,l} f[m-k, n-l] \cdot g[k,l],$$

де f – вихідна матриця зображення;

g – ядро (матриця) згортки.

Неформально цю операцію можна описати наступним чином – вікном розміру ядра g проходимо з заданим кроком (зазвичай 1) все зображення f на кожному кроці поелементно множимо вміст вікна на ядро g, результат сумується і записується в таблицю.

Ідея згорткових нейронних мереж полягає в чергуванні згорткових шарів (англ. Convolution layers) і субдискретизуючих шарів (англ. Subsampling layers, верств підвибірки). Структура мережі – односпрямована (без зворотних зв’язків), багатошарова, яка зображена на рис. 2.5.

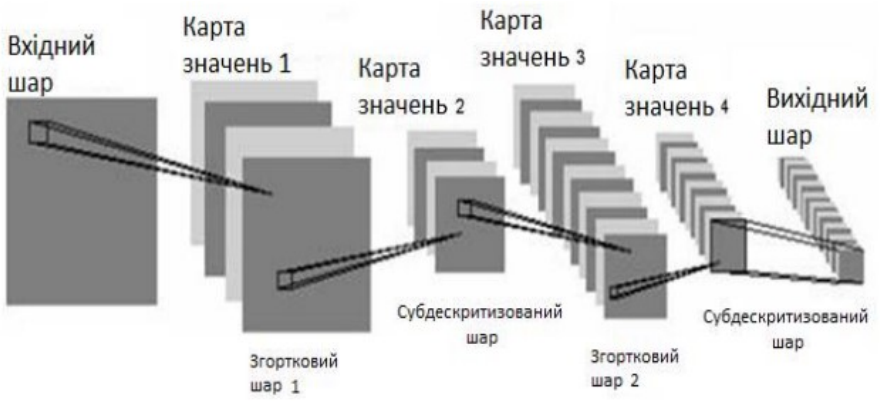

Рисунок 2.5 – Структура згорткової мережі

Модель згорткової мережі складається з трьох типів шарів: згорткові (convolutional) шари, субдискретизуючі (subsampling, підвибірка) прошарки і прошарки «звичайної» нейронної мережі – персептрона.

Архітектура згорткових нейронних мереж реалізує три ідеї, які забезпечують інваріантність мережі до невеликих зрушень, змін масштабу і спотворень:

– кожен нейрон отримує вхідний сигнал від локального рецептивного поля (local receptive fields) у попередньому шарі, що забезпечує локальну двовимірну зв’язність нейронів;

– кожен прихований шар мережі складається з безлічі карт ознак, на яких всі нейрони мають загальні ваги (shared weights), що забезпечує інваріантність до зміщення і скорочення загальної кількості вагових коефіцієнтів мережі;

– за кожним шаром згортки слідує обчислювальний шар, який здійснює локальне усереднення та підвибірку, що забезпечує зменшення розширення для карт ознак.

Робота згорткової нейронної мережі забезпечується двома основними елементами:

а) фільтри (filters) (визначники ознак);

б) карти ознак (feature maps).

Фільтр – це невелика матриця, що представляє ознаку, яку необхідно знайти на вихідному зображенні. За допомогою верхнього фільтра визначають-

ся частини вихідного зображення з вертикальними лініями, нижній фільтр служить для визначення частин зображення з горизонтальними лініями.

Безпосередньо процес визначення заснований на операції згортки фільтром оригінального зображення. Результати згортки, які визначають місце розташування ознак вихідного зображення, називаються картами ознак.

Мета процесу згортки – зменшити розмірність карти ознак до такої міри, щоб з повним набором ознак могла працювати мережа прямого поширення (в більшості випадків багатошаровий персептрон).

Згортковий шар реалізує ідею локальних рецептивних полів, тобто кожен вихідний нейрон з’єднаний тільки з певною (невеликою) областю вхідної матриці і таким чином моделює деякі особливості людського зору.

Недоліками згорткових нейронних мереж (ЗНМ) є:

– висока складність архітектури;

– повнозв’язаність;

– фіксована площа вікна шару згортки.

З метою підвищення ефективності роботи ЗНМ необхідно знайти оптимальні значення наступних параметрів:

– кількість карт ознак;

– щільність зв'язків між картами ознак;

– розмір вікна;

– площа перекриття;

– початкова ініціалізація ваг.

2.3 Застосування згорткових нейронних мереж для звукової класифікації об’єктів

Архітектура в стилі Inception була представлена в 2012 році. Модель, представлена в статті, автори назвали GoogLeNet, в якій використовувалися блоки Inception [10]. Це була нова та інноваційна архітектура. Крім того, він

привернув велику увагу, оскільки багато архітектур того часу були набори все більшої кількості шарів для збільшення пропускної здатності мережі. Inception, з іншого боку, був більш креативним і витонченим.

Замість того, щоб глибше просто додавати більше шарів, він також розширився. Inception бере вхідний тензор і паралельно виконує комбінацію згортки та об’єднання зображений на рис. 2.6.

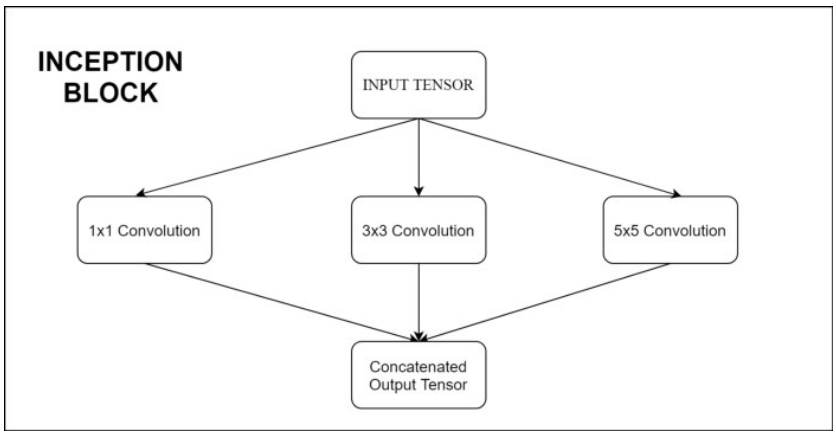

Рисунок 2.6 – Блок Inception

Тепер фактичний блок Inception дещо відрізняється з точки зору кількості згорток, їх розміру та того, як вони пошарові. Блок паралельно виконує згортку з різними розмірами фільтрів. І вихідні тензори об’єднуються вздовж розміру каналу, тобто складаються один за одним.

Вхідний тензор, який обробляється різною згорткою, буде мати форму (розмір пакету, висота, ширина, канали). У блоці Inception розмір каналу вхідного тензора зменшується за допомогою згортки 1x1 перед застосуванням згорток 3x3 або 5x5. Це зменшення розміру каналу робиться для зменшення обчислень під час подачі цього тензора на наступні шари. Вхідний тензор окремо обробляється трьома шарами згортки. І три окремі вихідні тензори об’єднуються вздовж розмірності каналу. GoogLeNet використовує кілька блоків Inception та кілька інших хитрощів і налаштувань для досягнення своєї продуктивності.

Для трансформації шару Inception в Xception замінимо три окремі шари 1x1 у кожній із паралельних веж одним шаром. Це буде виглядати приблизно,

як на рис. 2.7.

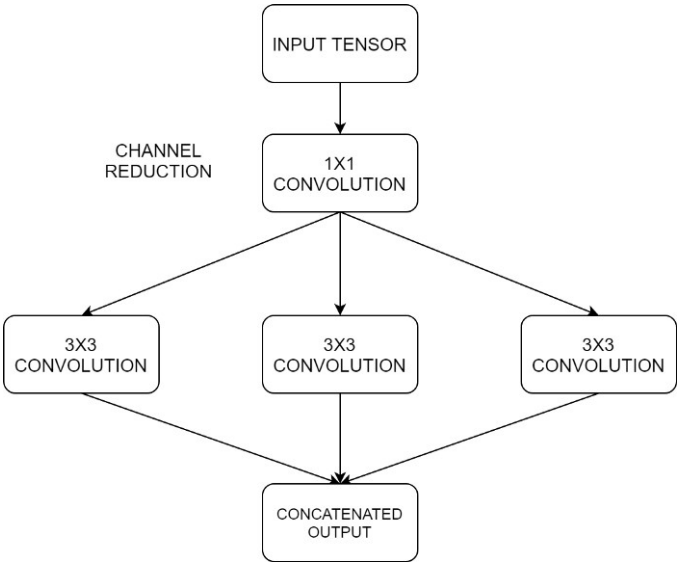

Рисунок 2.7 – Блок Xception

Розділений по глибині шар згортки – це те, що забезпечує Xception. І він активно використовує це у своїй архітектурі. Цей тип згортки схожий на крайню версію блоку Inception. Але трохи відрізняється своєю роботою. Роздільний по глибині шар має дві функціональні частини, які розділяють роботу звичайного шару згортки. Дві частини – це глибинна згортка та точкова згортка. Дослідники розділили всю архітектуру Xception на 14 модулів, де кожен модуль є лише купою шарів Depthwise Separable Convolution (DSC) та об’єднання. Ці модулі згруповано у три групи, а саме. вхідний потік, середній потік і вихідний потік. І кожна з груп має чотири, вісім і два модулі відповідно. Остання група, тобто вихідний потік, може додатково мати повністю пов’язані шари на кінці.

Перший модуль містить звичайні шари згортки, і в них немає DSC. І кожен шар згортки, як звичайний, так і DSC, супроводжується шаром пакетної нормалізації. Згортки, які мають крок 2, зменшують його майже вдвічі. Структура цієї групи зображена на рис. 2.8.

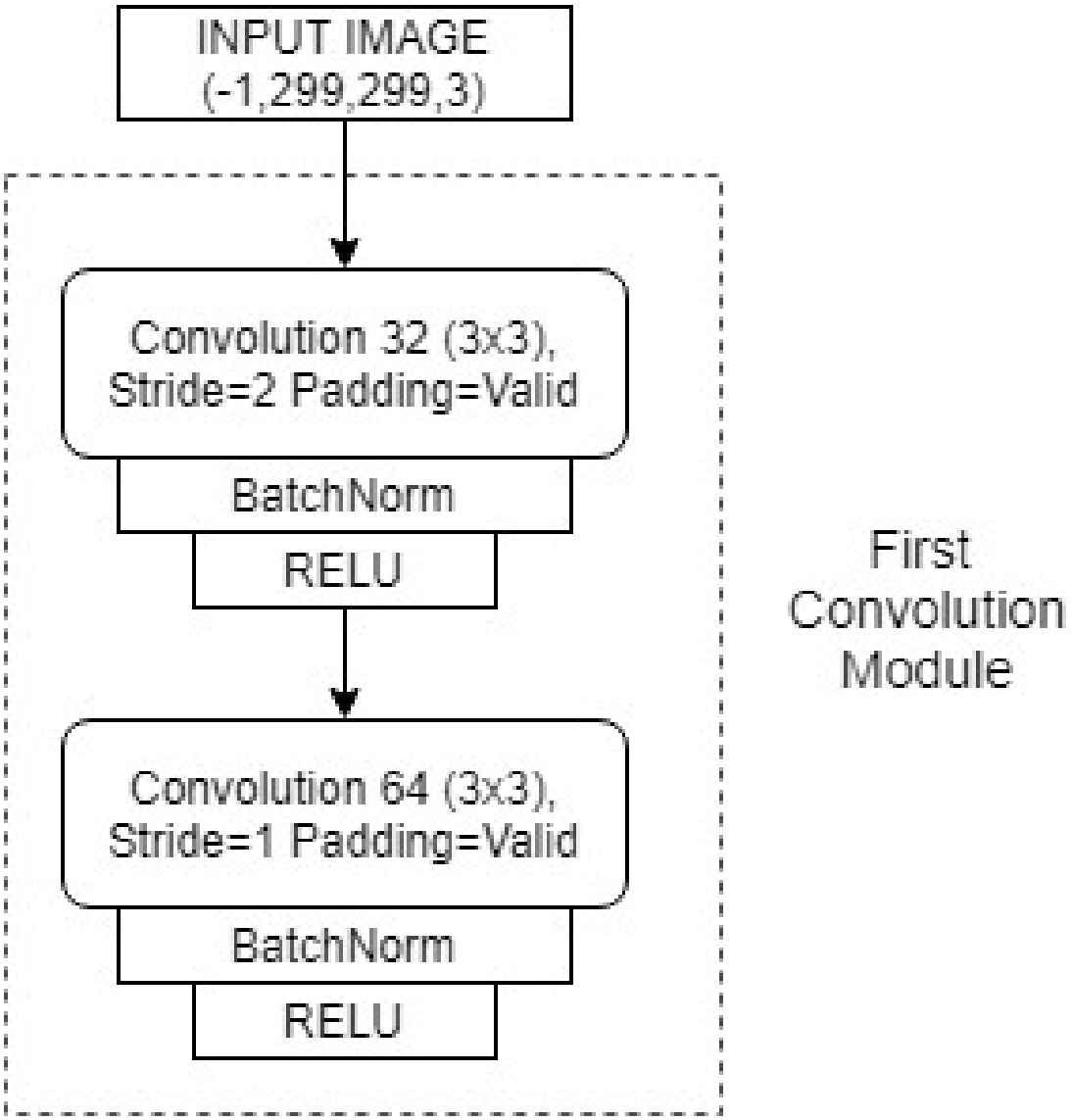

Рисунок 2.8 – Блок Xception

За винятком першого модуля, всі інші у вхідному потоці мають residual skip connections. Паралельні skip connections мають шар точкової згортки, який додається до виводу з головного шляху.

У середньому потоці вісім таких модулів, один за одним. Вищевказаний модуль повторюється вісім разів, щоб сформувати середній потік. Усі 8 модулів у середньому потоці використовують крок 1 і не мають жодних шарів об'єднання. Таким чином, просторовий розмір тензора, який передається з вхідного потоку, залишається незмінним. Глибина каналу також залишається незмінною. І це те саме, що і глибина введення. Середній шар зображений на рис. 2.9.

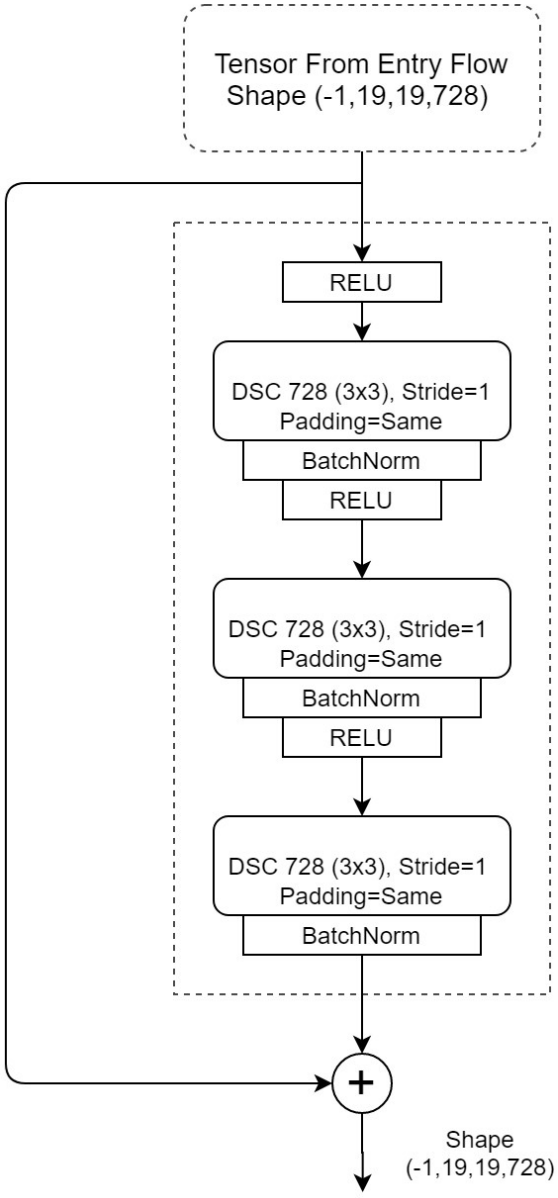

Рисунок 2.9 – Блок Xception

Вихідний потік має лише два модулі згортки, а другий не має skip connection. Другий модуль використовує Global Average Pooling, на відміну від попередніх модулів, які використовували Maxpooling. Вихідний вектор середнього рівня об’єднання може бути поданий безпосередньо до шару логістичної регресії. Але можна використовувати проміжні шари Fully Connected. Вихіднй потік зображений на рис. 2.10.

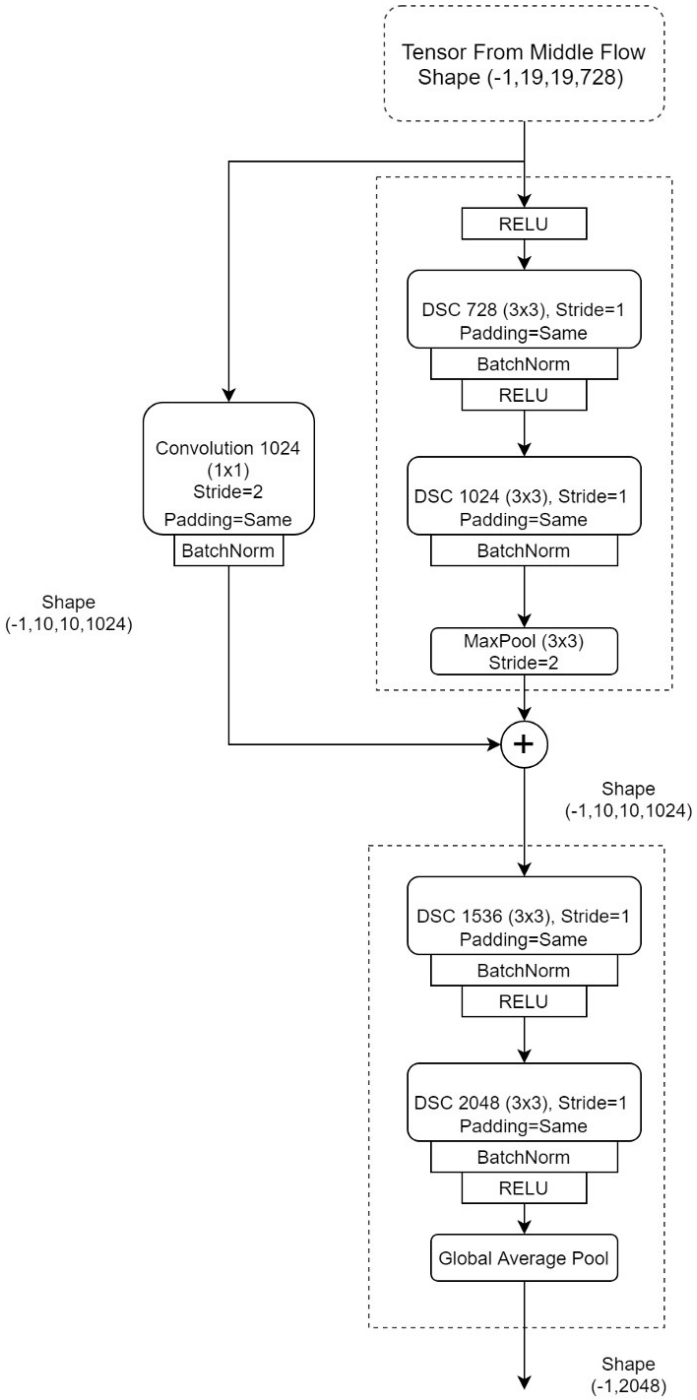

Рисунок 2.10 – Блок Xception

Модель Xception містить майже таку ж кількість параметрів, як і Inception V3, але перевершує InceptionV3 з невеликим відривом у наборі даних ImageNet. Кращу роботу з майже такою ж кількістю параметрів можна пояснити його архітектурною інженерією.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Особливості програмної реалізації задач на мові Python

В останні двадцять років мова Python все частіше використовуються і для наукових обчислень, і для аналізу даних. Сьогодні головна перевага Python і одна з основних причин його популярності полягає в тому, що він додає особливості наукових обчислень в мову загального призначення, який використовується в багатьох наукових областях і галузях промисловості. Це значно полегшує перехід від досліджень до виробництва.

Jupyter Notebook – це командна оболонка для інтерактивних обчислень. Він часто використовується для роботи з даними, статистичним моделюванням і машинним навчанням. Сьогодні Jupyter сам по собі є екосистемою, яка розуміє кілька альтернативних інтерфейсів ноутбуків, бібліотек інтерактивної візуалізації, авторських інструментів, сумісних з ноутбуками.

Крім багатого минулого ноутбуків Jupyter і багатої екосистеми, яку вона надає розробникам, ось ще десять причин, за якими можна почати використовувати її для свого наступного проекту в області машинного навчання:

– все в одному місці: Jupyter Notebook – це інтерактивна веб-середовище, яке об'єднує код, багатий текст, зображення, відео, анімацію, математичні рівняння, графіки, карти, інтерактивні фігури і віджети, а також графічні інтерфейси в єдиний документ;

– легко конвертувати: Jupyter поставляється із спеціальним інструментом, nbconvert, який конвертує ноутбуки в інші формати, такі як HTML і PDF. Інший онлайн-інструмент, nbviewer, дозволяє нам візуалізувати загальнодоступний ноутбук прямо в браузері;

– мовна незалежність: Архітектура Jupyter незалежна від мови. Розв'язка між клієнтом і ядром дозволяє писати ядра на будь-якій мові;

– легко створювати обгортку ядра: Jupyter пропонує легкий інтерфейс для

мов ядра, який можна обернути на Python. Обгортки ядра можуть реалізовувати додаткові методи, зокрема, для завершення коду і перевірки коду;

– легко налаштовується: Jupyter інтерфейс може бути використаний для створення повністю персоналізованого досвіду в Jupyter Notebook;

– розширення до призначених для користувача чарівними командами: Створюйте розширення IPython до призначених для користувача чарівними командами, щоб зробити інтерактивні обчислення ще простіше. Багато розширення сторонніх виробників і чарівні команди існують;

– ноутбуки Jupyter можуть з легкістю проводити ефективні та відтворювані інтерактивні обчислювальні експерименти. Вони дозволяють вести докладний звіт про виконану роботу. Крім того, простота використання ноутбуків Jupyter означає, що не потрібно турбуватися про відтворюваність;

– ефективний інструмент викладання-навчання: Jupyter Notebook – це не тільки інструмент для наукових досліджень і аналізу даних, а й відмінний інструмент для навчання;

– інтерактивний код і дослідження даних: Пакет ipywidgets надає безліч спільних елементів управління призначеним для користувача інтерфейсом для інтерактивного вивчення коду і даних.

Python надає лаконічний код, що легко читається. У той час як складні алгоритми та універсальні робочі процеси стоять за алгоритмами машинного навчання, простота Python дозволяє розробникам писати надійні системи. Розробникам доводиться докладати всіх зусиль для вирішення проблеми ML замість того, щоб концентруватися на технічних нюансах мови.

Крім того, Python привабливий для багатьох розробників, оскільки його легко вивчати. Код на Python зрозумілий людям, що полегшує побудову моделей для машинного навчання.

Говорять, що Python більш інтуїтивний, ніж інші мови програмування. Інші вказують на безліч фреймворків, бібліотек і розширень, які спрощують реалізацію різних функцій. Загальновизнано, що Python підходить для спільної реалізації, коли в ній бере участь кілька розробників. Оскільки Python є мовою

загального призначення, він може виконувати набір складних завдань по машинному навчання і дозволяє швидко створювати прототипи, які дозволять вам протестувати ваш продукт для цілей машинного навчання.

3.2 Алгоритм розв’язання задачі звукової класифікації об’єктів

Алгоритм розв’язання задачі складається з п’яти логічних блоків:

– парсинг даних;
– аналіз даних;
– попередня обробка даних;
– тренування моделі;
– тестування моделі.

Далі розглянемо алгоритм більш детально, він зображений на рис. 3.1.

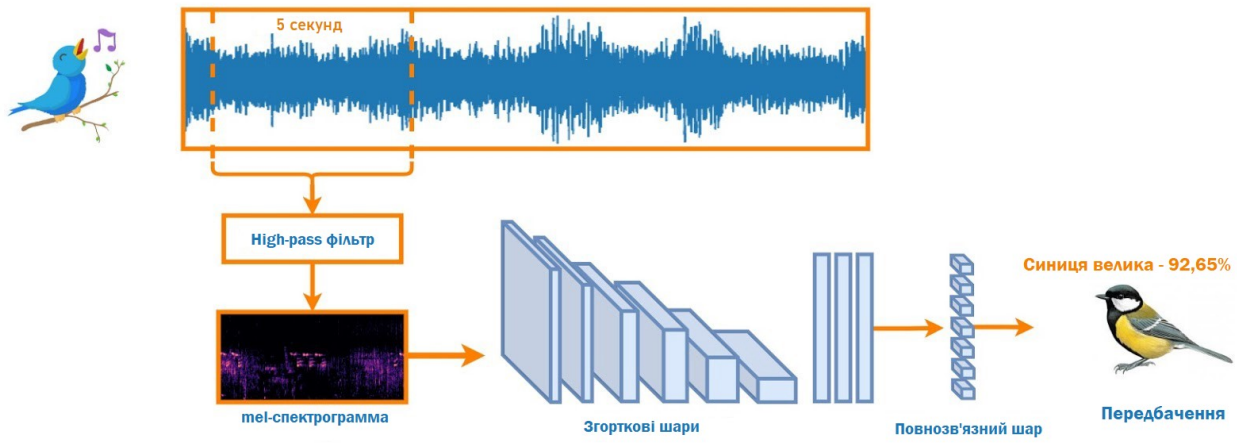

Рисунок 3.1 – Алгоритм розв’язання задачі

Для парсингу навчальних даних вони були з взяті сайту із записами звуків птахів по всьому світу. Після цього був проведений аналіз даних. Для попередньої обробки даних аудіо доріжки були поділені на рівні відрізки, відфільтровані високочастотним фільтром та перетворені у mel-спектрограми, а далі конвертовані у зображення однакового розміру. Наступним кроком дані були поді-

лені на тренувальний, тестовий набори та валідаційний.

Для швидшого и кращого тренування моделі були використані попередньо навчені моделі. Після тренування моделей вони були протестовані на валідаційному, та на основі отриманих метрик обирається найкраща модель.

3.3 Опис програми

Для реалізації програми були використані різні бібліотеки та фреймворки:

– numpy: бібліотека, яка надає можливість використовувати математичні і числові функції у вигляді «швидких функцій»;

– keras: відкрита бібліотека для нейромереж, написана мовою Python;

– librosa: пакет для аналізу музики та аудіо. Він надає функції, необхідні для створення систем пошуку інформації про музичну інформацію.

– matplotlib: бібліотека двовимірної графіки для мови програмування Python за допомогою якої можна створювати високоякісні рисунки різних форматів;

– sklearn: найпоширеніший вибір для розв’язання задач класичного машинного навчання.;

– tensorflow: фреймворк для нейромереж.

Після імпортування даних бібліотек отримуються дані з веб-сайту Xeno-canto за допомогою API сайту. Дані представляють собою аудіо доріжки в форматі WAW. Далі за допомогою бібліотеки librosa дані проходять попередню обробку, а саме розділяються на проміжки тривалістю по 5 секунд. Наступним кроком за допомогою цієї ж бібліотеки дані фільтруються високочастотним фільтром та конвертуються у зображення, а саме спектрограми розміром 432 на 432 пікселя. Потім ці дані діляться на три набори у співвідношенні 8:1:1. Після цього вже дані йдуть на data generator з бібліотеки keras, де проходять аугментацію для більшої стійкості моделі та перетворюються у тензори для тренування моделі. Що до тренування моделей, були обрані преднавчені моделі згортко-

вих нейронних мереж для класифікації зображень, а саме Xception та EfficientNetB3 [11]. Та після тренування моделей, обирається найкраща, критерієм оцінки є функція втрат loss та метрика F1 score. Де значення loss найменше, а значення метрики найбільша, та і обирається.

4 РЕЗУЛЬТАТИ ОБЧИСЛЮВАЛЬНОГО ЕКСПЕРИМЕНТУ

У світі налічується понад 10 тисяч видів птахів. Розпізнавання червонокрилих дроздів або кропив'янок, наприклад, у тій чи іншій місцевості може дати важливу інформацію про місце існування. Оскільки птахи знаходяться високо в харчовому ланцюзі, вони є відмінними індикаторами погіршення якості довкілля та забруднення. Моніторинг стану та тенденцій біорізноманіття в екосистемах – завдання не з легких. При правильному виявленні та класифікації звуків за допомогою машинного навчання дослідники можуть покращити свої можливості щодо відстеження стану та тенденцій біорізноманіття у важливих екосистемах, що дозволить їм краще підтримувати глобальні зусилля щодо збереження природи.

Останні досягнення в галузі машинного прослуховування покращили збирання акустичних даних. Проте, створення результатів аналізу з високою точністю та запам'ятовуванням залишається складною задачею. Більшість даних залишається невивченою через відсутність дійових інструментів для ефективного та надійного вилучення сигналів, що становлять інтерес (наприклад, позиви птахів).

Орнітологічне співтовариство щорічно збирає багато петабайтів акустичних даних, але більшість даних залишається невивченою. У разі успіху можжна допомогти дослідникам правильно визначати та класифікувати звуки птахів, що значно покращить їхню здатність відслідковувати стан та тенденції біорізноманіття у важливих екосистемах. Дослідники зможуть робити висновки щодо якості життя у тій чи іншій місцевості на основі зміни популяції птахів, що дозволить їм визначити, як краще підтримати глобальні зусилля щодо збереження природи.

Аналіз та моделювання пісень українських птахів. Записи були завантажені з xeno-canto.org [12], який є веб-сайтом, присвяченим звукам птахів з усьо-

го світу. Дані можна завантажити за допомогою API.

Птахи мають високу міжвидову мінливість – спів тих самих видів птахів у різних країнах може звучати зовсім по-різному.

Набір даних представляє собою записи звуків 10 видів птахів [13]:

– Chlorischloris: Зеленяк звичайний – співочий птах родини в'юркових, Україні гніздовий, кочуючий, зимуючий вид;

– Emberizacitrinella: Вівсянка звичайна – горобцеподібний птах родини вівсянкових, в Україні осілий, кочовий вид;

– Erithacusrubecula: Вільшанка – птах родини мухоловкових, в Україні гніздовий, перелітний, зимуючий птах;

– Fringillacoelebs: Зяблик звичайний – співочий птах родини в'юркових, дуже численний в Україні;

– Parusmajor: Синиця велика – невеликий птах родини синицевих, в Україні звичайний осілий вид;

– Phylloscopuscollybita: Вівчарик-ковалик – невеликий птах родини кропив’янкових, поширений на значній частині території Старого Світу, в Україні гніздовий, перелітний;

– Phylloscopustrochilus: Вівчарик весняний – дрібний широко поширений перелітний птах родини кропив'янкових, в Україні гніздовий перелітний вид;

– Troglodytestroglodytes: Волове очко – вид дрібних птахів роду волове очко родини воловоочкових, в Україні осілий, кочовий, перелітний птах;

– Turdusmerula: Дрізд чорний – вид птахів роду дроздів родини дроздових, в Україні гніздовий, перелітний, зимуючий вид;

– Turdusphilomelos: Дрізд співочий – птах з родини Дроздових, в Україні звичайний гніздовий, перелітний вид;

На рис. 4.1 зображено кількість записів по кожному класу. Як можна побачити, дані не збалансовані, тобто мають різну кількість записів.

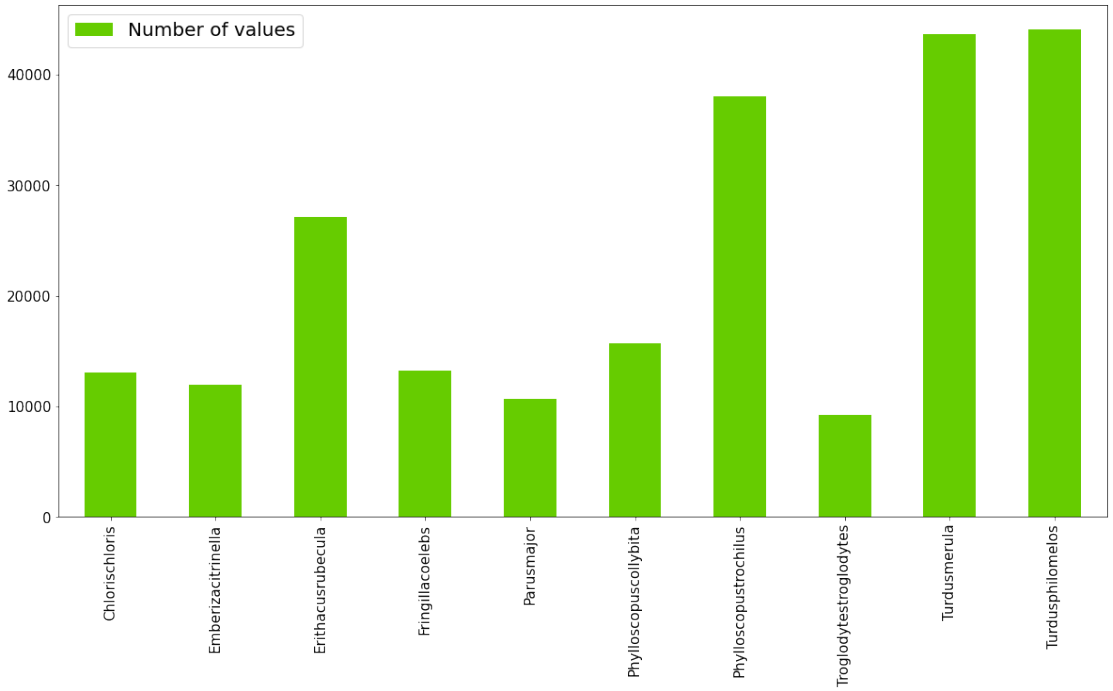

Рисунок 4.1 – Кількість даних у кожному класі

Наступним кроком буде попередня обробка даних. Кожна звукова доріжка була розрізана на 5-секундні записи та попередньо оброблена в mel-спектрограми. Мета полягає в нормалізації даних, щоб вони мали однаковий розмір по всьому набору даних, та зменшити шум записів. Крім того, дані фільтруються за допомогою фільтра високих частот.

Далі набір даних було поділено на тренувальний, валідаційний та тестовий набори у співвідношенні 8:1:1:

– train: набір на якому буде тренуватися нейронна мережа;

– valid: набір для валідації при тренуванні нейронної мережі;

– test: набір на якому буде відбуватися тестування натренованої моделі.

Разом із тим, набір даних не збалансований, було розраховані ваги для кожного класу за допомогою функції class_weight.compute_class_weight з бібліотеки sklearn з параметрами для збалансування набору даних при тренуванні моделі.

Для порівняння було побудовано 2 заздалегідь навчені моделі згорткових нейронних мереж.

Перша модель – це EfficientNetB3 з наступними параметрами:

– Loss: Categorical crossentropy;

– Optimizer: Adam;

– Scheduler: ReduceLR;

та метриками accuracy, F1-score. Її тренування зображено на рис. 4.2, 4.3.

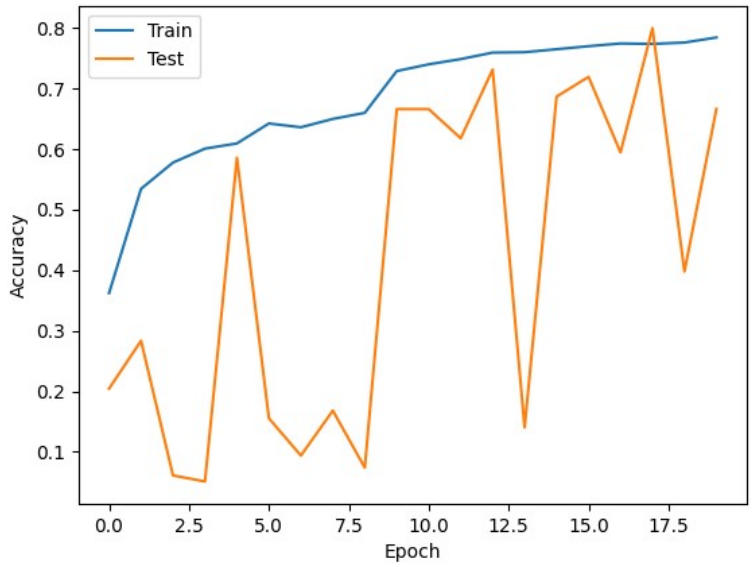

Рисунок 4.2 – Процес тренування нейронної мережі

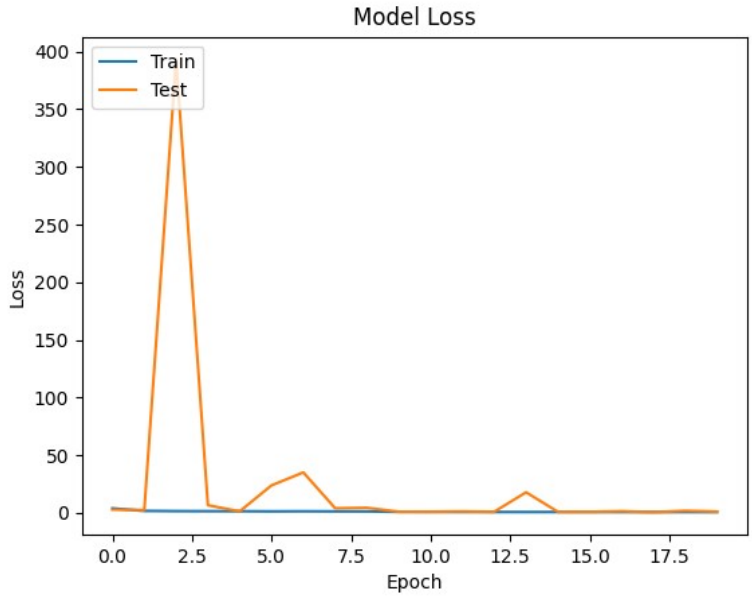

Рисунок 4.3 – Процес тренування нейронної мережі

З рис. 4.2 можна побачити, що у моделі є невелике перенавчання, так як

на тренувальних даних модель працює трохи краще, але це не є суттєвим. Максимальна точність була отримана на 20 епосі та вона дорівнює, на тренувальному наборі, 78,42% та мінімальною похибкою 0,69 і значення F1-score дорівнює 0.67, також на рис 4.4 можна побачити значення F1-score для кожного класу.

	precision	recall	f1-score
Chlorischloris	0.67	0.89	0.76
Emberizacitrinella	0.29	0.83	0.43
Erithacusrubecula	0.88	0.46	0.61
Fringillacoelebs	0.75	0.71	0.73
Parusmajor	0.31	0.91	0.47
Phylloscopuscollybita	0.75	0.81	0.78
Phylloscopustrochilus	0.91	0.68	0.78
Troglodytestroglodytes	0.71	0.64	0.68
Turdusmerula	0.80	0.72	0.76
Turdusphilomelos	0.92	0.50	0.65
accuracy			0.67
macro avg	0.70	0.72	0.66
weighted avg	0.78	0.67	0.69

Рисунок 4.4 – Процес тренування нейронної мережі

Друга модель – це Xception, з наступними параметрами:

- Loss: Categorical crossentropy;
- Optimizer: Adam;
- Scheduler: ReduceLROnPlateau;

та метриками accuracy, F1-score. Її тренування зображено на рис. 4.5 – 4.7.

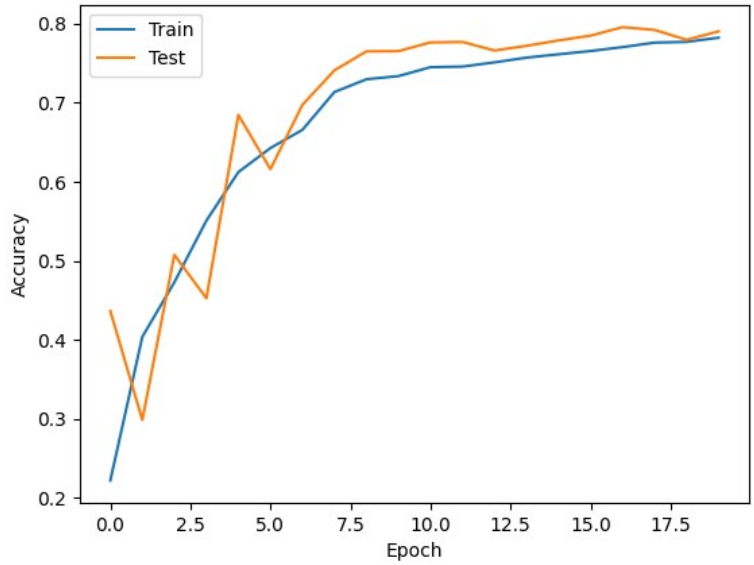

Рисунок 4.5 – Процес тренування нейронної мережі

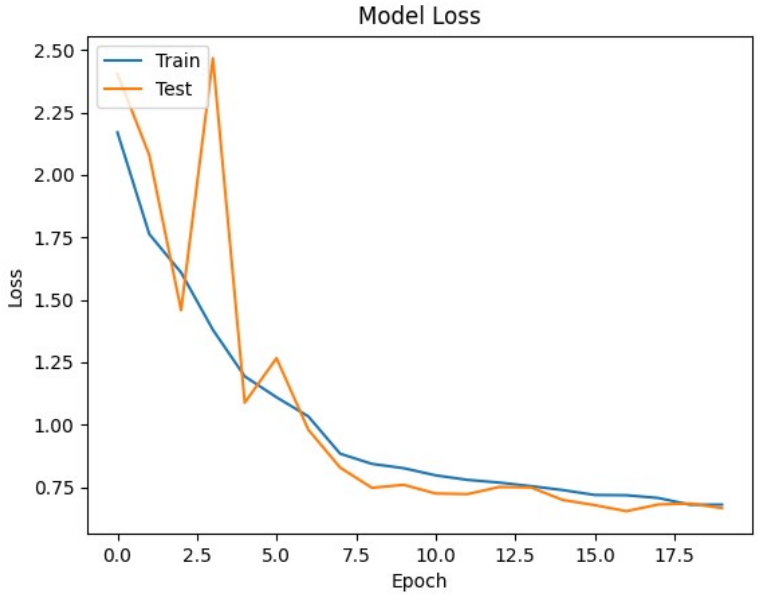

Рисунок 4.6 – Процес тренування нейронної мережі

З рис. 4.5 можна побачити, що у моделі немає перенавчання, так як на тренувальних даних модель працює так само як і на тестових. Максимальна точність була отримана на 20 епосі і вона дорівнює, на трену-вальному наборі, 80% та мінімальною похибкою 0,68 і значення F1-score дорівнює 0.793, також на рис 4.7 можна побачити значення F1-score для кожного класу.

	precision	recall	f1-score
Chlorischloris	0.93	0.86	0.89
Emberizacitrinella	0.60	0.81	0.69
Erithacusrubecula	0.77	0.79	0.78
Fringillacoelebs	0.54	0.86	0.66
Parusmajor	0.69	0.89	0.78
Phylloscopuscollybita	0.85	0.82	0.83
Phylloscopustrochilus	0.91	0.82	0.86
Troglodytestroglodytes	0.54	0.81	0.65
Turdusmerula	0.94	0.75	0.84
Turdusphilomelos	0.86	0.74	0.80
accuracy			0.80
macro avg	0.76	0.81	0.79
weighted avg	0.82	0.79	0.81

Рисунок 4.7 – Процес тренування нейронної мережі

З вищеперерахованих моделей найкраща – Xception, тобто вона має максимальні значення метрик та мінімальне значення функції втрат, на даному наборі даних. Також, для перевірки на нових даних, які моделі ще не бачили, вони тестувалися на тестовому наборі. Результати вийшли такі:

– EfficientNetB3: loss 1.13, accuracy 0.67, F1-score 0.6643;

– Xception: loss 0.65, accuracy 0.8, F1-score 0.81.

Усе вказує на те, що найкраща модель – це Xception.

Отже, для подальшого використання моделі нові дані повинні бути попередньо обробленні та подані на модель для отримання вірогідності щодо відповідності до класу.

ВИСНОВКИ

В результаті виконання кваліфікаційної роботи були досліджені проблеми класифікації об’єктів за допомогою нейронних мереж. Був проведений аналіз стану проблеми звукової класифікації об’єктів, наведені вербальна, формальна та змістовна постановки задачі. Також було проведено огляд методів звукової класифікації об’єктів.

Для експерименту були реалізовані дві моделі згорткових нейронних мереж для класифікації звуків птахів. Модель була побудована на реальних даних звуків птахів України та за допомогою мови програмування Python. Результати були задовільними та з гарною точністю при не збалансованих даних, та продемонстрували абсолютну коректність роботи моделі на різних даних.

Я вважаю, що впровадження має місце бути, так як дана проблема дуже актуальна і як продукт він може бути конкурентним на ринку, так як альтернатив не так вже багато. Стосовно продовження досліджень, мені здається, що це доцільно, бо існує ще багато параметрів, які б слід врахувати, та найкращим було б отримання більшої кількості розмічених даних. Так як у всьому світі популяція птахів різко скорочується з 1970-х років, в основному через зміни в кліматі. Також очікується, що популяції птахів зміняться в чисельності та розподілі через вплив зміни клімату в найближчі роки. Таким чином, дуже важливо здійснювати моніторинг популяцій птахів для цілей збереження видів, та управління екосистемою. Так як зараз це традиційно здійснювалося за допомогою ручної зйомки, та часто включає використання волонтерів для вирішення проблем. Однак ручне спостереження залишається обмеженим, особливо в областях, які є фізично важким доступ, або коли у фокусі є нічна поведінка. Багато видів птахів легко впізнати за їхніми звуками, часто краще, ніж баченням, і тому з сучасними дистанційними станціями моніторингу, здатними фіксувати безперервно аудіозаписів це відкриває перспективу масштабного просторово-часового моніторингу птахів. І так як результати досить не погані, я вважаю цей напрямок дуже перспективним у майбутньому.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Rosenblatt F. The perceptron: A probabilistic model for information storage and organization in the brain // Psychological Review. 1958. Vol. 65, No. 6. P. 386–408.

2. Rumelhart D., Hinton G. E., Williams R. J. Learning representations by back-propagating errors // Nature. 1986. Vol. 323. P. 533–536.

3. Krizhevsky A., Sutskever I., Hinton G. E. ImageNet Classification with Deep Convolutional Neural Networks // Advances in Neural Information Processing Systems. 2012. Vol. 25. P. 1097–1105.

4. Goodfellow I., Yoshhua B. Deep Learning. Massachusetts : MIT Press, 2016. 800 p.

5. Github. URL : https://colah.github.io/posts/2015-08-Understanding-LSTMs/ (дата звернення: 14.10.2021).

6. Chollet F. Deep Learning with Python. New York : Manning Publications, 2017. 384 p.

7. Ramahlo L. Fluent Python. Sebastopol : O`Reilly, 2015. 792 p.

8. Purwins H. Deep Learning for Audio Signal Processing // Journal for selected topics of signal processing. 2019. Vol. 13, No. 2. P. 206–219.

9. Жерон О. Прикладне машинне навчання за допомогою Scikit-Learn та TensorFlow. Київ : Вільямс, 2018. 688 с.

10. Chollet F. Xception: Deep Learning with Depthwise Separable Convolutions // IEEE Conference on Computer Vision and Pattern Recognition. 2017. P. 1251–1258.

11. Mingxing T., Quoc V. L. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks // International Conference on Machine Learning. 2019. P. 6105–6114.

12. Xeno Canto. URL : https://www.xeno-canto.org/ (дата звернення: 11.10.2021)

13. Серебряков В. В. Атлас птахів України. Київ : Фітосоціоцентр, 2012. 240 с.