Міністерство освіти і науки України
Державний ВНЗ «Національний гірничий університет»

Факультет інформаційних технологій
(факультет)

Кафедра програмного забезпечення комп'ютерних систем
(повна назва)

ПОЯСНЮВАЛЬНА ЗАПИСКА
дипломної роботи

магістра
(назва освітньо-кваліфікаційного рівня)

галузь знань *12 Інформаційні технології*
(шифр і назва галузі знань)

спеціальність *122 Комп'ютерні науки*
(код і назва спеціальності)

спеціалізація *Інформаційні управляючі системи та технології*
(назва спеціалізації)

освітній рівень *магістр*
(назва освітнього рівня)

кваліфікація *інженер з комп'ютерних систем*
(назва кваліфікації)

на тему: *Обґрунтування методики класифікації аудіо файлів для музичного програмного забезпечення та веб порталів*

Виконавець:
студент *6* **курсу , групи** *122М-16-1*

(підпис) *Терехов В.А.* (прізвище та ініціали)

Керівники	Посада, прізвище, ініціали	Оцінка	Підпис
проекту	*проф. Алексєєв М.О.*		
розділів:			
Спеціальний	*проф. Алексєєв М.О.*		
Економічний	*доц. Касьяненко Л.В.*		

Рецензент			

Нормоконтроль	*доц. Коротенко Л.М.*		

Дніпропетровськ
2018

Міністерство освіти і науки України
Державний вищий навчальний заклад
«Національний гірничий університет»

ЗАТВЕРДЖЕНО:
завідувач кафедри

програмного забезпечення комп'ютерних систем
(повна назва)

_____ І.М. Удовик
(підпис) (прізвище, ініціали)

« » _____ 20 ____ року

ЗАВДАННЯ
на виконання кваліфікаційної роботи магістра

спеціальності *122 Комп'ютерні науки*
(код і назва спеціальності)

студенту *122М-16-1* *Терехов В.А.*
(група) (прізвище та ініціали)

Тема дипломної роботи *Обґрунтування методики класифікації аудіо файлів для музичного програмного забезпечення та веб порталів*

1 ПІДСТАВИ ДЛЯ ПРОВЕДЕННЯ РОБОТИ

Наказ ректора Державного ВНЗ «НГУ» від *26.12.2017 р.* № *2127 -л*

2 МЕТА ТА ВИХІДНІ ДАНІ ДЛЯ ПРОВЕДЕННЯ РОБІТ

Об'єкт досліджень – аудіо файли та способи їх класифікації.

Предмет досліджень – нейро-мережева класифікація аудіо файлів за різними групами.

Мета НДР – підвищення швидкості та вдосконалення процесу пошуку та пропозицій аудіо файлів, у тому числі музичних композицій, у різних сферах застосування за допомогою методів глибинного навчання.

Вихідні дані для проведення роботи – теоретичні й експериментальні дослідження та програмні розробки у сфері дослідження хвильових спектрів та глибинного навчання.

3 ОЧІКУВАНІ НАУКОВІ РЕЗУЛЬТАТИ

Актуальність даної теми зумовлена наявністю значних недоліків у загально прийнятому способі поділу аудіо файлів на групи: значні затрати часу на пошук необхідних файлів, неточність класифікації.

Наукова новизна результатів, що очікуються, полягає у проведенні аналізу і виявленні недоліків емпіричного підходу класифікації аудіофайлів, а також у створенні методики глибинного навчання нейронних мереж з метою розпізнавання класів музичних файлів з високою точністю.

Практична цінність результатів полягає в ефективному навчанні нейронної мережі на підставі отриманої методики й подальше використання навченої системи в реалізації програмного забезпечення для роботи з аудіо файлами.

4 ВИМОГИ ДО РЕЗУЛЬТАТІВ ВИКОНАННЯ РОБОТИ

Результати магістерської роботи повинні відповідати вимогам паспорту наукової спеціальності 05.13.06 – «Інформаційні технології».

Результати досліджень мають бути подані у вигляді, що дозволяє побачити та оцінити безпосереднє використання методики розробки систем штучного інтелекту (нейронних мереж) та вилучення характерних ознак аудіо файлів. Згідно виробничих функцій та професійних задач магістра, повинна бути розроблена ефективна методика та алгоритм розробки системи класифікації.

5 ЕТАПИ ВИКОНАННЯ РОБІТ

Найменування етапів робіт	**Строки виконання робіт (початок-кінець)**
1	2
Аналіз стану питання	11.09.2017 26.09.2017
Основи нейронних мереж	30.09.2017 23.10.2017
Конфігурація та результати моделювання	27.10.2017 13.12.2017

6 РЕАЛІЗАЦІЯ РЕЗУЛЬТАТІВ ТА ЕФЕКТИВНІСТЬ

Економічний ефект від реалізації результатів роботи очікується позитивним завдяки підвищенню конверсії покупок на музичних веб порталах, а також поліпшенню просування пропозицій менш відомих музичних авторів у пропозиціях користувачеві.

Соціальний ефект від реалізації результатів роботи очікується позитивним завдяки прискоренню і вдосконаленню процесу пошуку необхідних музичних композицій у власній або веб бібліотеці музики.

7 ДОДАТКОВІ ВИМОГИ

Відповідність оформлення ДСТУ 3008-95. Документація. Звіти у сфері науки і техніки. Структура і правила оформлення.

Завдання видав _____ *Алексєєв М.О.*
(підпис) (прізвище, ініціали)

Завдання прийняв до виконання _____ *Терехов В.А.*
(підпис) (прізвище, ініціали)

Дата видачі завдання: *09.09.2017р.*

Термін подання дипломного проекту до ДЕК *23.01.2018*

Реферат

Пояснительная записка: 53 с., 23 рис., 3 прил., 29 источников.

Объект исследования: аудио файлы и способы их классификации

Цель магистерской работы: повышение скорости и усовершенствование процесса поиска и предложений аудио файлов, в том числе музыкальных композиций, в разных сферах применения с помощью методов глубинного обучения.

Методы исследования. При решении поставленной задачи использовались научные достижения в исследовании волновых спектров и в областях машинного обучения.

Научная новизна полученных результатов состоит в проведении анализа и выявлении недостатков эмпирического подхода классификации аудио файлов, а также в создании методики глубинного обучения нейронных сетей с целью распознавания классов музыкальных файлов.

Практическое значение работы заключается в обучении на основании полученной методики нейронной сети и дальнейшем использовании обученной системы в реализации программного обеспечения для работы с аудио файлами.

Область применения. Разработанный программный продукт, реализуемый предложенную методику, может применяться в основе формирования библиотеки аудио файлов как музыкальных порталов, так и клиентских программ и мобильных приложений.

Значение работы и выводы. Применение реализации методики позволяет ускорить и усовершенствовать процесс поиска необходимой аудио файлов в музыкальной библиотеке, что может повысить конверсию покупок в музыкальных веб магазинах, а также улучшить продвижение менее известных авторов, издающих музыку, похожую на ту, которую предпочитает пользователь.

Прогнозы по развитию исследований. Разработать серверное приложение с нейронной сетью, обученной по сотням терабайт аудио данных, с предоставляемым открытым API для частных и коммерческих целях.

В разделе «Экономика» проведены маркетинговые исследования рынка сбыта созданного на основании методики программного продукта и проанализирован социальный эффект от введения в использование данной методики.

Список ключевых слов: ГЛУБИННОЕ ОБУЧЕНИЕ, СВЕРТОЧНЫЕ НЕЙРОННЫЕ СЕТИ , MP3, МУЗЫКА, АУДИО ДАНЫЕ, ПРЕОБРАЗОВАНИЕ ФУРЬЕ, СПЕКТРОГРАММА, КЛАССИФИКАЦИЯ

Реферат

Пояснювальна записка: 53 с., 23 рис., 3 дод., 29 джерел.

Об'єкт дослідження: аудіо файли та способи їх класифікації

Мета магістерскої роботы: підвищення швидкості і вдосконалення процесу пошуку і пропозицій аудіо файлів, в тому числі музичних композицій, в різних сферах застосування за допомогою методів глибинного навчання.

Методи дослідження. При вирішенні поставленого завдання використовувалися наукові досягнення в дослідженні хвильових спектрів і в областях машинного навчання.

Наукова новизна отриманих результатів полягає у проведенні аналізу і виявленні недоліків емпіричного підходу класифікації аудіофайлів, а також у створенні методики глибинного навчання нейронних мереж з метою розпізнавання класів музичних файлів з високою точністю.

Практичне значення роботы полягає в ефективному навчанні нейронної мережі на підставі отриманої методики й подальше використання навченої системи в реалізації програмного забезпечення для роботи з аудіо файлами.

Область застосування. Розроблений програмний продукт, що реалізує запропоновану методику, може застосовуватися в основі формування бібліотеки аудіо файлів як музичних порталів, так і клієнтських програм і мобільних додатків.

Значення роботи та висновки. Застосування реалізації методики дозволяє прискорити і вдосконалити процес пошуку необхідних аудіо файлів в музичній бібліотеці, що може підвищити конверсію покупок в музичних веб магазинах, а також поліпшити просування менш відомих авторів, які видають музику, схожу на ту, яку довподоби користувачеві.

Прогнози щодо розвитку досліджень. Розробити серверний додаток з нейронною мережею, навченої за сотнею терабайт аудіо даних, з наданням відкритого API для приватних і комерційних цілей.

У розділі «Економіка» проведені маркетингові дослідження ринку збуту створеного на підставі методики програмного продукту і проаналізовано соціальний ефект від введення в експлуатацію даної методики.

Список ключових слів: ГЛИБИННЕ НАВЧАННЯ, ЗГОРТКОВІ НЕЙРОННІ МЕРЕЖІ , MP3, МУЗИКА, АУДІО ДАНІ, ПЕРЕТВОРЕННЯ ФУР'Є, СПЕКТРОГРАМА, КЛАСИФІКАЦІЯ

The abstract

Explanatory note: 53 page., 23 fig., 3 app., 29 sources.

Object of research: audio files and ways to classify them.

The purpose of the master's work is to speeding up and improving the search process and suggestions for audio files, including music compositions, in different fields of application using methods of in-depth training.

Methods of research. When solving this problem, scientific achievements were used in the study of wave spectra and in the areas of machine learning.

The scientific novelty of the results is the conducting analysis and revealing the shortcomings of the empirical approach to the classification of audio files, as well as in creating a technique for deep learning of neural networks in order to recognize the music files classes.

The practical importance of the work lies in training based on the received technique of the neural network and further use of the trained system in the implementation of software for working with audio files.

Application area. The developed software product, implemented the proposed methodology, can be used in the basis of the library of audio files as music portals, and client programs and mobile applications.

The meaning of the work and conclusions. Application of the implementation of the methodology allows to speed up and improve the process of searching for the necessary audio files in the music library, which can increase the conversion of purchases in music web shops, and also improve the promotion of lesser known authors who publish music similar to the one that the user prefers.

Forecasts for the development of research. Develop a server application with a neural network trained in hundreds of terabytes of audio data, with an open API for private and commercial purposes.

In the section "Economics" carried out marketing research on the market created based on the methodology of the software product and analyzed the social effect of the introduction of this method.

List of key words: DEEP LEARNING; CONVOLUCIONAL NEURAL NETWORK, MP3, MUSIC, AUDIO DATA, FOURIER TRANSFORMATION, SPRECTROGRAM, CLASSIFICATION.

Зміст

Перелік скорочень

API	Application Programming Interface (програмний інтерфейс продукту для використання у зовнішніх застосунках)
НМ	Нейронна мережа
ЗНМ	Згорткова нейронна мережа
ReLU	Rectified Linear Units (зрізані лінійні вузли)
ELU	Exponential Linear Units (експоненціальні лінійні вузли)

Вступ

Актуальність роботы. Цифрова музика стає все більш популярною в житті людей. У наші дні досить часто людина володіє тисячами цифрових музичних творів, і користувачі можуть створювати свою власну музичну бібліотеку через системи управління музикою або користуючись можливостями веб порталів. Однак для професійних музичних баз даних часто наймають співробітників, щоб вручну класифікувати і індексувати музичні активи відповідно до зумовленими критеріями; більшість користувачів не мають часу і терпіння, щоб переглядати свої особисті музичні колекції і вручну індексувати музичні фрагменти по одному. З іншого боку, якщо музичні активи не класифікуються належним чином, це може стати великим головним болем, коли користувач хоче знайти певну частину музики серед тисяч штук в музичній колекції. Ручні методи класифікації не можуть відповідати розвитку цифрової музики. Класифікація музики - це проблема розпізнавання образів, яка включає в себе функції вилучення і створення класифікатора.

Штучна нейронна мережа знайшла великий успіх в області розпізнавання образів, її можна навчити розрізняти критерії, які використовуються для класифікації, щоб узагальнювати образи, повторюючи покази входів нейронної мережі, класифіковані за групами. Нейронна мережа пропонує нове рішення для класифікації музичних аудіо файлів.

Мета та задача досліження. Метою даної магістерської роботи є створення методики класифікації аудіо файлів із застосуванням технологій машинного навчання.

Для досягнення поставленої мети в роботі сформульовані і вирішені такі завдання:

1. Проведення аналізу та виявлення недоліків існуючого підходу до класифікації аудіо файлів
2. Аналіз відмінних характеристик аудіо файлів різних класів та жанрів.

3. Проектування нейронної мережі, здатній навчитись розпізнавати віявлені характеристики

Об'єкт дослідження – аудіо файли та способи їх класифікації.

Предмет дослідження – нейро-мережева класифікація аудіо файлів за різними групами.

Ідея роботи полягає в підвищенні швидкості та вдосконалення процесу пошуку та пропозицій аудіо файлів, у тому числі музичних композицій, у різних сферах застосування.

Методи дослідження. При вирішенні поставленого завдання використовувалися наукові досягнення в дослідженні хвильових спектрів і в областях машинного навчання.

Наукові положення, очікувані наукові результати:

1. Методика классификації аудіо файлів на основі використання технологій глибинного навчання.

Обгрунтованність достовірність научних положень

Обгрунтованість і достовірність наукових положень, висновків і рекомендацій магістерської роботи обгрунтована коректністю поставлених проблем і прийнятих припущень при математичному описі процесів, обґрунтованістю вихідних посилок, достатнім обсягом вибірки даних і верифікованими на модельних об'єктах результатами обчислень.

Наукова новизна отриманих результатів полягає у проведенні аналізу і виявленні недоліків емпіричного підходу класифікації аудіофайлів, а також у створенні методики глибинного навчання нейронних мереж з метою розпізнавання класів музичних файлів з високою точністю.

Практичне значення отриманих результатів полягає в ефективному навчанні нейронної мережі на підставі отриманої методики й подальше використання навченої системи в реалізації програмного забезпечення для роботи з аудіо файлами.

Зв'язок роботи з державними програмами, планами науково-дослідних робіт.

Результати дипломної роботи можуть бути використані як інструмент для дослідження сприйняття людиною певних класів музики, вміння їх розрізняти, а також характер впливу на людину прослуховування окремих груп. Також можна скористатися запропонованою методикою для створення програмного продукту організації аудіо бібліотеки.

Особливий внесок магістра складається в:

- выборі методыв досліджень та техноголій реалізації;
- створенні технології реалізуючої механізми формування параметрів характеризації аудіо файлів;
- разрабці теоретичної частини роботы, в якій досліджені та систематизовані знання про існуючі підходи розробки информаційних систем, вирішуючих задачу класифікації;
- оцінці отриманих результатів.

Апробація результатів магістерської роботи.

Основні положення і результати були докладені та обговорені на студентській науковій конференції.

Структура і обсяг роботи. Робота складається з вступу, чотирьох розділів і висновків. Містить 57 сторінок друкованого тексту, в тому числі 35 сторінок тексту основної частини з 23 рисунками, списку використаних джерел з 29 найменуваннями на 2 сторінках, 3 додатків на 7 сторінках.

РОЗДІЛ 1. АНАЛІЗ СТАНУ ПИТАННЯ

1.1. Про музику вцілому

Музика вже не одне століття знаходиться в побуті людини. Музика початку зароджувати в порядку тисячі років до н.е. Вона вдосконалювалася, створювалося все більше інструментів, характерних різним народам. Шукали все більш цікаві комбінації звучання, а музичні композиції ставали все більш складними і віртуозними. Дійшовши до наших днів музика перестала бути лише звуком, вона знайшла цифровий формат запису. Відкрились можливості електронної генерації і відтворення звуків, а їх сукупності знаходять сенс в музичних композиціях.

За своєю природою звук - це деяка характерна хвиля. Музика, будучи комбінацією звуків, визначається як гармонійне накладення хвиль різної довжини і частоти. Тому, як і будь-які інші хвильові коливання в електротехніці і математиці, вона легко піддається обробці та аналізу. Пошук закономірностей між різними типами музики став цікавити все більшу кількість наукових діячів.

1.2. Необхідність класифікації музики

Потреба чіткої класифікації музичних композицій з'явилася сама собою із зростанням кількості та різноманітності аудіо творів. Із появленням цифрового запису звуку люди почали коллекціонувати твори різних авторів. З поширенням музичного мистецтва почали з'являтися нові виконавці, нові жанри – кількість композицій в аудіо бібліотеках зростали, що призвело до проблеми пошуку необхідної композиції без попереднього ознайомлення з нею. Цей процес використовую принцип бажаних пропозицій, коли користувачеві пропонується ознайомитись із новими для нього аудіо записами, група яких заснована на схожості із улюбленими композиціями людини. Подібні процеси використовують в маркетингу, демонструючи рекламу цільовій аудиторії.

У віці інформаційних технологій ручний розподіл музики на жанри досить затратний за часом. Автоматичний розподіл також потребую початкової інформації про жанр, але після тренування системи можливо буде оцінити позитивний вплив даної технології. Але щоб автоматизувати розподіл аудіо файлів, необхідно виділити характерні відзнаки один від одного.

1.3. Розпізнання емоційності музики

Однією з найскладнішою з точки зору точності є класифікація музики за настроєм. Музиці, через свою неоднорідну природу, вкрай не просто зіставити мітку того чи іншого емоційно забарвленого класу. В першу чергу це обумовлено безпосередньо сприйняттям людини окремих гармонік, а також емоційною реакцією на різні звуки. Але загальні риси все ж виділити можна. Для цього доведеться відійти від звичних дискретних категорій і звернутися до роботи Джеймса Русселя, який пропоную вимірювати настрій в безперервному багатовимірному просторі. Він виділив характерні емоційні характеристики і відобразив їх на моделі Valence-Arousal (Валентність-Бадьорість), зображеної на рис. 1.1.

Рис.1.1. Модель Валентність-Бадьорість Русселя

У будь-якому випадку для вирішення задачі класифікації за настроєм вкрай складно підібрати вірно помічені дані для впевненості збіжності

розрахунків нейронної мережі. І навіть добре навчена мережа однією групою людей може видавати не задовольняють вихідні дані іншій групі людей через різне психоакустичного сприйняття.

1.4. Разподіл на жанри

Основним питанням магістерської роботи є класифікація музичних композицій за жанром. На відміну від поділу за настроєм, жанрові категорії маю як супер класи (classic, hip-hop, rock, electronic, jazz і т.д.), так і підкласи (piano, beats, drum & bass, swing і т.д.), відроджуються характеристики батьківського класу з додатковими емоційними забарвленнями і варіативністю виконання [17]. У нашому випадку, маючи невеличку вибірку аудіо файлів, ми вимушені відмовитись від классифікування за піджанрами. Але це можливо впровадити як етап вдосконалення методики, яка буде сформована.

1.5. Постановка завдання

З розповсюдженням мобільних технологій та розвитку музичної індустрії зростає і попит на якісні послуги надання доступу до аудіо композицій, потреби у структурованому зберіганні музичної бібліотеки. Вцілому завдання магістрської роботи – вирішити проблему класифікації аудіо файлів.

Будуть описані технології представлення аудіо даних за подальшим їх використанням як вхідних даних навчання нейронної мережи. Також проаналізується структура, правила та порядок формалізування нейронної мережі. Визначеться з типом НМ, необхідним для вирішення поставленої задачі.

1.6. Висновки

У зв'язку з вищевикладеної інформацією, яка існує на даному етапі, про основні концепції і методи дослідження структур аудіо даних та їх класифікації, були сформульовані такі напрямки завдань подальшого дослідження:

- Розгляд існуючих технологій і методів для аналізу аудіо даних
- Вибір конкретних технологій та бібліотек для вирішення поставленого завдання;
- Створювання методики представлення та класифікації аудіо файлів

РОЗДІЛ 2. ОСНОВИ ХАРАКТЕРИЗАЦІЇ АУДІО ФАЙЛІВ ТА МОДЕЛЮВАННЯ НЕЙРОННИХ МЕРЕЖ

2.1. Цифрове відображення аудіо сигналу

Реальний аудіо сигнал - це складне за формою коливання, деяка складна залежність амплітуди звукової хвилі від часу. Перетворення аналогового звукового сигналу в цифровий вигляд називається аналогово-цифровим перетворенням або оцифруванням.

Процес дискретизації за часом - це процес отримання миттєвих значень перетворюємого аналогового сигналу з певним часовим кроком, званим кроком дискретизації (рис 2.1.).

Рис. 2.1. Дискретизація сигналу

Кількість здійснюваних в одну секунду замірів величини сигналу називають частотою дискретизації або частотою вибірки, або частотою семплювання (від англ. «sampling» - «вибірка»). Очевидно, що чим менше крок, тим правдивіше більша їх кількість (тобто, тим частіше реєструються значення амплітуди), і, отже, тим більш точне уявлення про сигнал ми отримуємо. Це міркування підтверджується доведеною теоремою, теоремою Котельникова (в зарубіжній літературі зустрічається як теорема Шеннона, Shannon). Відповідно до цієї теореми, аналоговий сигнал з обмеженим спектром може бути точно

описаний дискретної послідовністю значень його амплітуди, якщо ці значення слідують з частотою, як мінімум удвічі перевищує найвищу частоту спектра. Інакше кажучи, аналоговий сигнал, в якому частота найвищої складової спектра дорівнює F_m, може бути точно описаний послідовністю дискретних значень амплітуди, якщо для частоти дискретизації F_d виконується: $F_d \geq 2F_m$. На практиці це означає наступне: для того, щоб оцифрований сигнал містив інформацію про всьому діапазоні чутних людиною частот вихідного аналогового сигналу (0 - 20 кГц) необхідно, щоб вибране значення частоти дискретизації при оцифрування сигналу становила не менше 40 кГц.

Здавалося б, для завершення процесу оцифровки тепер залишилося лише записати виміряні миттєві значення амплітуди сигналу в чисельній формі. Отримана послідовність чисел (по одному результату виміру амплітуди сигналу на кожен крок) і утворює цифрову форму вихідного аналогового сигналу - так званий імпульсний сигнал. Тут, однак, виявляється основна проблема оцифровки, яка полягає в неможливості записати виміряні значення сигналу з ідеальною точністю.

2.2. Стиснення з втратами

Кодування з втратами тому і називається «з втратами», що призводить до втрати деякої частини аудіо інформації. Таке кодування призводить до того, що перекодованим сигнал при відтворенні звучить схоже на оригінальний, але фактично перестає бути йому ідентичним. В основі більшості методів кодування з втратами лежить використання психоакустических властивостей слуховий системи людини, а також різних хитрощів, пов'язаних з переквантованіем і передіскретізацією сигналу. У частотності, в процесі компресії аудіо дані аналізуються кодером на предмет виявлення різних деталей звучання, якими можна знехтувати. Замасковані частоти, нечутні і слабослишімие деталі звучання - всім цим можна пожертвувати з метою досягнення більш високого значення коефіцієнта компресії. Там, де в звучанні важлива лише розбірливість (наприклад, в телефонії, де наявність частот вище

4 кГц не є необхідним), аудіо інформація в процесі кодування піддається серйозному «спрощенню», що укупі з використанням «розумних» квантователей і вдалих «жадібних» алгоритмів компресії даних дозволяє досягти найвищих ступенів компресії (1: 50 і вище). Там, де якості звучання пред'являються більш високі вимоги (наприклад, в портативних і побутових аудіо пристроях), аудіо матеріали піддають більш щадному кодування. Треба відзначити, що ступінь агресивності кодера по відношенню до деталей звучання може регулюватися (ця здатність, втім, залежить від конкретної реалізації). В середньому, сучасні кодери навіть при настільки високого ступеня компресії, як 1: 10 дозволяють забезпечити відмінне звучання, якість якого середнім слухачем на середньої апаратурі оцінюється як рівне якості звучання початкових аудіо даних.

2.3. MP3 файл

MP3 - формат файлу для зберігання аудіоінформації. У ньому використовується алгоритм стиснення з втратами, розроблений для істотного зменшення розміру даних, необхідних для відтворення запису і забезпечення якості відтворення звуку, що точно відповідає оригінальному (на думку більшості слухачів), але з відчутними втратами якості при прослуховуванні на якісної звукової системи.

Файл MP3 має стандартний формат, який являє собою кадр, що складається з 384, 576 або +1152 семплів (в залежності від версії і рівня MPEG), і всі кадри мають пов'язану інформацію заголовка (32 біта) і додаткову інформацію (9, 17, або 32 байта, в залежності від версії MPEG і стерео / моно).

У стандарті MPEG-1 рівня 3 вказані кілька бітових швидкостей: 32, 40, 48, 56, 64, 80, 96, 112, 128, 144, 160, 192, 224, 256 і 320 кбіт / с і доступні частота дискретизації становить 32 , 44,1 і 48 кГц. Частота дискретизації 44.1 кГц майже завжди використовується, тому що це також використовується для CD audio, основного джерела, використовуваного для створення MP3-файлів. В Інтернеті використовується більша розмаїтість бітових швидкостей. 128 кбіт / с

є найбільш поширеним явищем, оскільки він зазвичай забезпечує адекватне якість звуку у відносно невеликому просторі. 192 кбіт / с часто використовують ті, хто помічає артефакти при більш низьких швидкостях передачі. У міру збільшення доступності смуги пропускання в Інтернеті і розміру жорсткого диска файли з повільною швидкістю 128 кбіт / с повільно замінюються більш високими швидкостями передачі даних, такими як 192 кбіт / с, причому деякі з них кодуються до максимального MP3-файлу 320 кбіт / с.

Навпаки, нестислий звук, що зберігається на компакт-диску, має бітову швидкість 1411,2 кбіт / с (16 біт / вибірка × 44100 вибірок / сек. × 2 канали / 1000 біт / кілобіт).

Деякі додаткові швидкості передачі бітів і частоти дискретизації були доступні в стандартах MPEG-2 і (неофіційних) стандарту MPEG-2.5: швидкість передачі 8, 16, 24 і 144 кбіт / с і частота дискретизації 8, 11.025, 12, 16, 22.05 і 24 кГц.

Нестандартні швидкості передачі даних до 640 кбіт / с можуть бути досягнуті за допомогою кодера LAME і опції freeformat, хоча кілька MP3-плеєрів можуть відтворювати ці файли. Відповідно до стандарту ISO, декодери повинні мати можливість декодування потоків до 320 кбіт / с.

2.4. Структура MP3

MP3-файл складається з декількох фрагментів (фреймів) MP3, які, в свою чергу, складаються з заголовка і блоку даних як показано на рис. 2.2. Така послідовність фрагментів називається елементарним потоком. Фрагменти не є незалежними елементами («резервуар байт»), і тому не можуть бути вилучені довільно. Блок даних MP3-файлу містить стислу аудиоинформацию у вигляді частот і амплітуд. На наведеному нижче рис 2.2. показано, що заголовок MP3 складається з маркера, який служить для знаходження вірного MP3-фрагмента. За ним слідує біт, що показує, що використовується стандарт MPEG, і два біта, що показують використання layer 3; іншими словами, це визначає MPEG-1 Audio Layer 3 або MP3. Наступні значення можуть варіюватися в залежності від типу MP3-файлу. Стандарт ISO / IEC 11172-3 визначає діапазон значень для кожної секції заголовка, разом із загальною його специфікацією. Більшість MP3-файлів зараз містять ID3-метадані, які передують або йдуть за MP3-фрагментом.

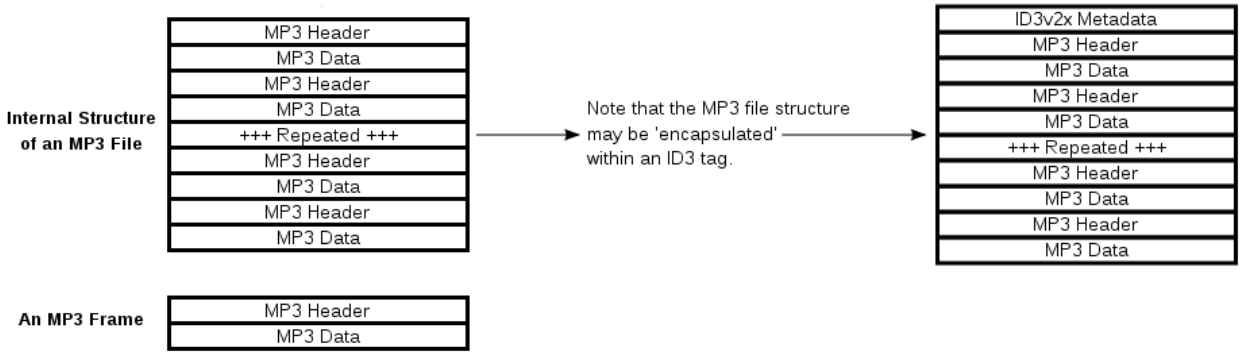

Рис. 2.2. Структура MP3 файлу

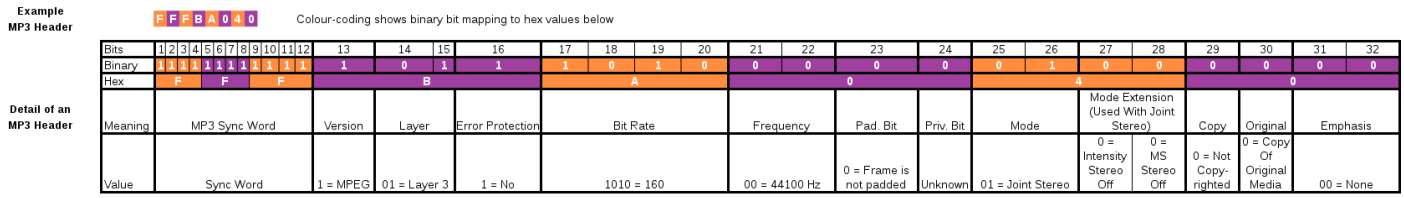

Рис. 2.3. Структура фрейму

2.5. Візуалізація блоку даних

Інформація з блоків даних MP3 файлу витягується в масив значень амплітуд у часі. У чистому вигляді цю інформацію дуже складно обробити, а тим більше характеризувати, тому потрібен спосіб представити набір даних інакше. Для нашої задачі буде зручним відобразити наявний масив у вигляді спектрограми (рис. 2.4.).

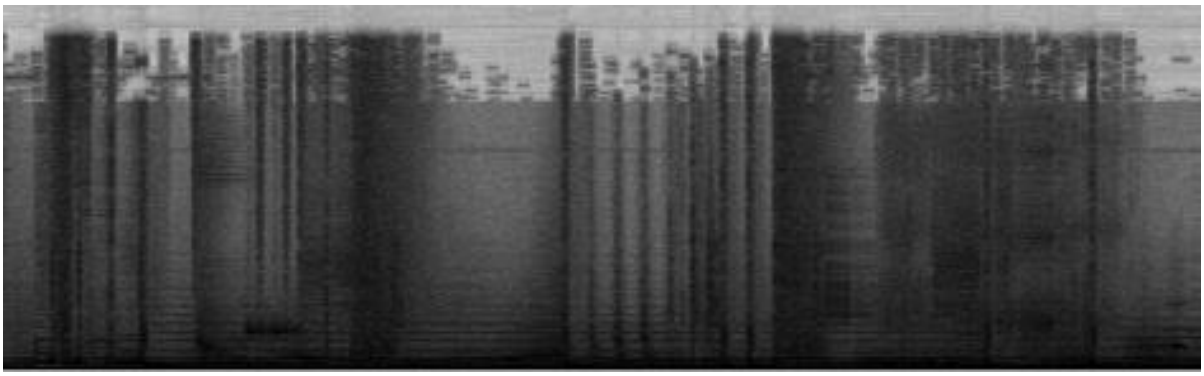

Рис. 2.4. Відрізок спектрограми аудіо файлу

Це зображення показує залежність спектральної щільності потужності сигналу від часу. Найбільш поширеним уявленням спектрограми є двовимірна діаграма: на горизонтальній осі представлено час, по вертикальній осі - частота; третій вимір із зазначенням амплітуди на певній частоті в конкретний момент часу представлено інтенсивністю або кольором кожної точки зображення. Спектрограма зазвичай створюється одним з двох способів: апроксимується, як набір фільтрів, отриманих із серії смугових фільтрів (це був єдиний спосіб до появи сучасних методів цифрової обробки сигналів), або розраховується за сигналом часу, використовуючи віконне перетворення Фур'є. Ці два способи фактично утворюють різні квадратичні частотно-часові розподілу, але еквівалентні при деяких умовах.

Для створення спектограмми сигнал розбивається на частини, які, як правило, перекриваються, і потім проводиться перетворення Фур'є, щоб розрахувати величину частотного спектра для кожної частини. Кожна частина відповідає вертикальній лінії на зображенні - значення амплітуди в залежності

від частоти в кожен момент часу. Спектри або терміни їх виконання розташовуються поруч на зображенні.

2.6. Віконне перетворення Фур'є

Перетворення Фур'є - операція, що зіставляє однієї функції дійсної змінної іншу функцію дійсної змінної. Формула розрахунку перетворення відображена на рис. 2.5. Отримана нова функція описує коефіцієнти («амплітуди») при розкладанні вихідної функції на елементарні складові - гармонійні коливання з різними частотами:

$$\hat{f}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x) e^{-ix\omega}\, dx. \tag{2.1}$$

Віконне перетворення Фур'є – це різновид перетворення Фур'є, яка визначається формулою, де $W(\tau - t)$— деяка віконна функція:

$$F(t, \omega) = \int_{-\infty}^{\infty} f(\tau) W(\tau - t) e^{-i\omega\tau}\, d\tau, \tag{2.2}$$

Існує безліч математичних формул візуально поліпшують частотний спектр на розриві кордонів вікна. Для цього застосовуються такі перетворення:

- прямокутне (рис. 2.7.)
- фрагмент синусоїди
- синус в кубі
- перетворення Гаусса
- перетворення Хеннінга (рис. 2.8.)
- перетворення Хеммінга (рис. 2.9.)
- перетворення Розенфілда
- перетворення Блекмана-Харріса (рис. 3.0.)

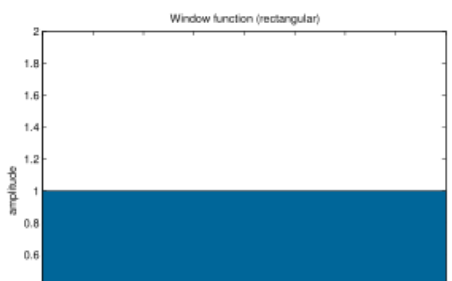

Рис. 2.7 Прямокутне перетворення

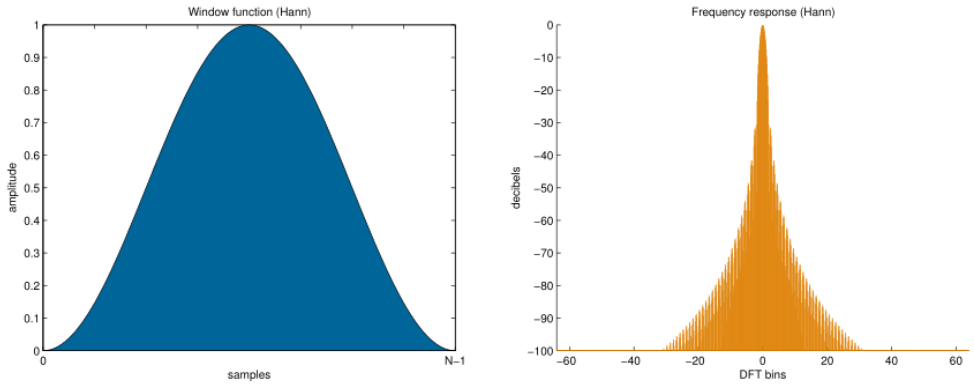

Рис. 2.8. Перетворення Хеннінга

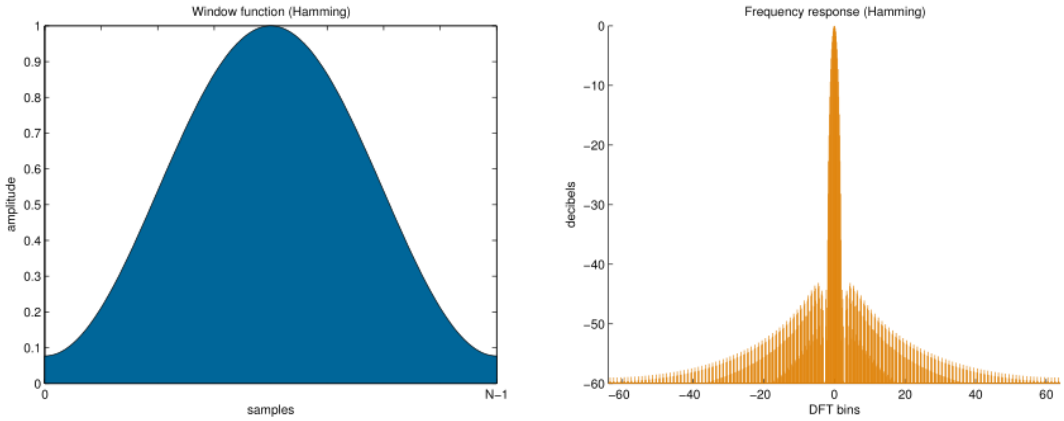

Рис. 2.9. Перетворення Хеммінга

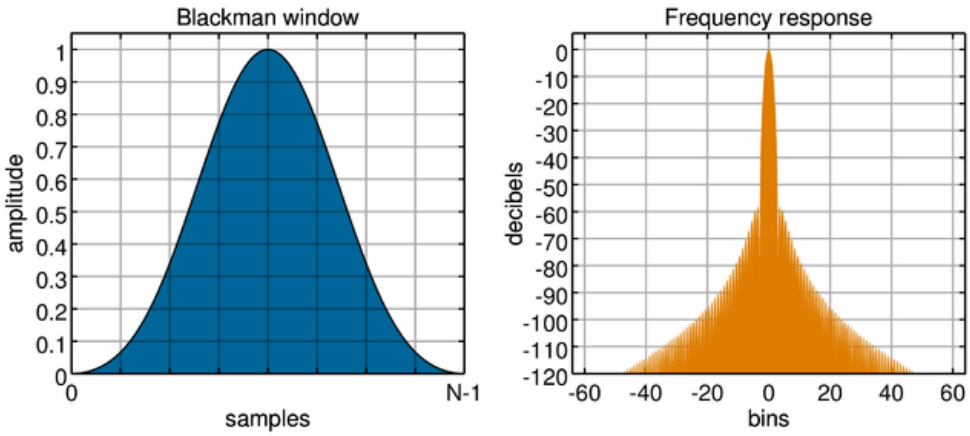

Рис. 2.10. Перетворення Блекмана

За допомогою цих перетворень ми зможемо отримати спектрограми аудіо файлів, за якими будемо навчати програму класифікувати їх..

2.7. Задача класифікації

Задачи класифікації зустрічаються дуже часто в різних областях діяльності людини. Як і задача регресійного аналізу, задача класифікації вирішується з метою подальшого прогнозування змінної відгуку (номери класу). Передбачається, що вже є якась кількість n об'єктів, для кожного з яких відомий деякий набір з m ознак (чинників) і номер класу, до якого цей об'єкт належить, тобто сирі дані, які використовуються для вирішення задачі класифікації, мають вигляд:

Таблиця 2.1

Номер спостереження, i	**Значення факторів**			**Значення змінної відгуку (номеру класу)**
1	$x_{1,1}$	…	$x_{1,m}$	y_1
…	…	…	…	…
i	$x_{i,1}$	…	$x_{i,m}$	y_i
…	…	…	…	…
n	$x_{n,1}$	…	$x_{n,m}$	y_n

Тут значення змінної відгуку - номер класу, якому належить об'єкт, тобто

$$y_i \in \{1, \dots, K\} \text{ для всіх } i = 1, .., n \qquad (2.3)$$

K – (відома) кількість класів.

Нейронні мережі є передовим способом вирішення завдань класифікації завдяки точності визначення і швидкості роботи. Для нашої задачі необхідна конфігурація мережі, що навчається з учителем.

2.8. Глибинне навчання

Глибинне навчання — це галузь машинного навчання, що ґрунтується на наборі алгоритмів, які намагаються моделювати високорівневі абстракції в даних, застосовуючи глибинний граф із декількома обробними шарами, що побудовано з кількох лінійних або нелінійних перетворень.

Глибинне навчання є частиною ширшого сімейства методів машинного навчання, що ґрунтуються на навчанні ознак даних. Спостереження (наприклад, зображення) може бути представлено багатьма способами, такими як вектор значень яскравості для пікселів, або абстрактнішим способом, як множина кромок, областей певної форми тощо. Деякі представлення є кращими за інші у спрощенні задачі навчання (наприклад, розпізнаванню облич, або виразів облич[4]). Однією з обіцянок глибинного навчання є заміна ознак ручної роботи дієвими алгоритмами автоматичного або напівавтоматичного навчання ознак та ієрархічного виділяння ознак.

Алгоритми глибинного навчання ґрунтуються на розподілених представленнях. Припущенням, що лежить в основі розподілених представлень, є те, що спостережувані дані породжено взаємодією факторів, організованих у рівні. Глибинне навчання додає припущення, що ці рівні факторів відповідають різним рівням абстракції або побудови. Для забезпечення різних ступенів абстракції можуть застосовуватися змінні кількості та розміри шарів.[5]

Глибинне навчання використовує цю ідею ієрархічних пояснювальних факторів, де з понять нижчого рівня відбувається навчання абстрактніших понять вищого рівня. Ці архітектури часто будуються за допомогою пошарового жадібного методу. Глибинне навчання дозволяє розплутувати ці абстракції й вихоплювати ознаки, що є корисними для навчання.[5]

Для задач керованого навчання методи глибинного навчання уникають проектування ознак, перетворюючи дані у компактні проміжні представлення на кшталт головних компонент, і виводять шаруваті структури, що усувають надмірність у представленні.[6]

2.9. Моделювання нейронів

Прагнучи змоделювати певні здібності мозку, Уоррен Маккаллох і Уолтер Піттс встановив спрощену модель біологічного нейрона в 1943 році, названу Маккаллох - Піттс моделлю, яка складається з декількох входів і одного виходу з центральним процесором (ЦП) [2].

На рис. 3.1 відображено модель нейрона, яка описується:

$$y=f(\sum_{i=1}^{N} w_i x_i - v_t) \qquad (2.4)$$

де x_i = вхідні сигнали, i = 1, 2, 3, ..., N;

w_i = синаптичні ваги;

v_t = поріг або зсув;

f (*) = функція активації або функція стиснення або елемент який обробляє;

y = вихідний сигнал нейрона.

Використання порогу v_t полягає в забезпеченні зміщення функції активації f (*). Маккаллох і Піттс не надали ніякого способу, за допомогою якого вузол або нейрон могли самонастроюватися або адаптувати свої синаптичні ваги в процесі навчання. У 1949 році Хебб запропонував просту математичну формулу, яка може адаптивно змінювати вагу нейронів пропорційно активності між до- і пост-сінаптикой нейрону:

$$\Delta w_i(n) = \mu y(n) x_i(n) \qquad (2.5)$$

де μ- позитивна постійна швидкість навчання за весь час n.

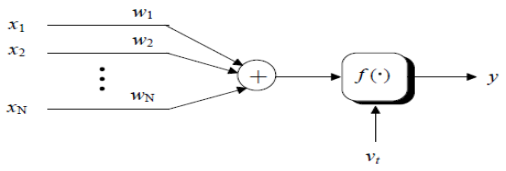

Рис. 2.11. Модель нейронів Маккаллох – Піттс

2.10. Персептрон

У 1958 році Розенблатт продемонстрував деякі практичні застосування, використовуючи персептрон [3]. Персептрон - це однорівневе з'єднання нейронів Маккаллоха-Пітта званих одношаровими мережами прямого доступу. Мережа здатна лінійно відокремлювати вхідні вектора в структуру класів гіперплоскістю. Лінійна асоціативна пам'ять є прикладом одношарової нейронної мережі. В такому додатку мережа пов'язує (вектор) з шаблоном введення (вектором), і інформація зберігається в мережі за допомогою модифікацій синаптичних ваг мережі.

Рис. 3.2 ілюструє персептрон, який описується:

$$y_i = f\left(\sum_{j=1}^{N} w_{ij}\, x_j - v_t\right) \qquad (2.6)$$

де i = 1, 2, ..., M (вихідні вузли), j = 1, 2, ..., N (входи).

Розенблатт розробив правило навчання, засноване на вагах, скоригованих пропорційно похибки між вихідними нейронами і бажаними виходами (мішенню). Вагові пристосування виходять з:

$$\Delta w_{ij}(n) = \mu[d_i(n) - y_i(n)]x_j(n) \qquad (2.7)$$

де i = 1, 2, ..., M (виходи), j = 1, 2, ..., N (входи), а d_i - бажаний вихід вузла i у часі n.

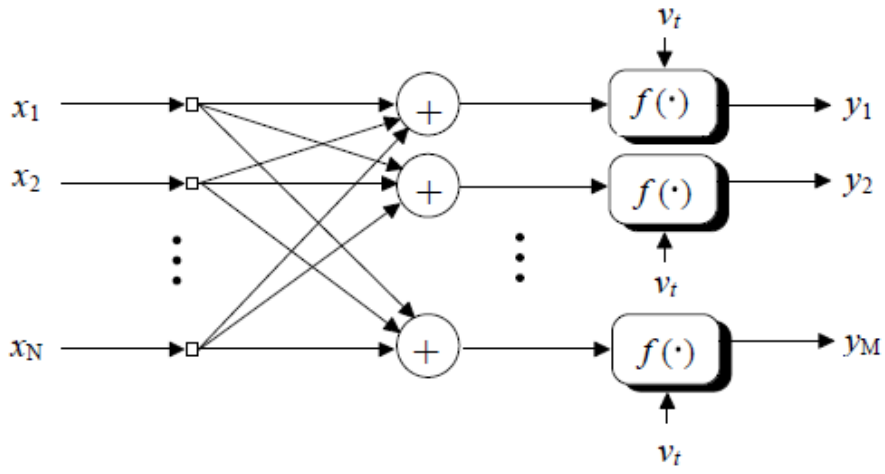

Рис.2.12. Персептрон з одним шаром (одношарова мережа фідів)

2.11. Багатошарові персептрони

Щоб витягти статистику вищого порядку, таку як реалізація простого XOR або Логічна функція XNOR (без попереднього процесора, який часто використовується в одношаровому персептрони), були запропоновані Мінським і Паперт в 1969 році нелінійні багатошарові персептрони або багатошарові мережі з прямим зв'язком, які математично показали , що існують фундаментальні обмеження на те, що може обчислити один персептрон [30]. Багатошарові персептрони (МСП) представляють один або кілька прихованих шарів, чиї обчислювальні вузли відповідно називаються прихованими нейронами. Функція прихованих нейронів полягає у втручанні між зовнішнім входом і виходом мережі. Рис. 3.3. представляє тришарові багатошарові персептрони з одним прихованим шаром і виходом. Вихідні вузли у вхідному шарі мережі складаються з N елементів шаблону, які складають вхідні сигнали, що подаються на K нейрони в другому шарі або перший прихований шар (l = 1). Вихідними сигналами M нейронів є кінцевий шар (l = L) мережі, що становить загальний відгук мережі на шаблон, наданий вихідними вузлами. Так само правильний вибір кількості прихованих вузлів може бути розрахований спочатку для кращого узагальнення.

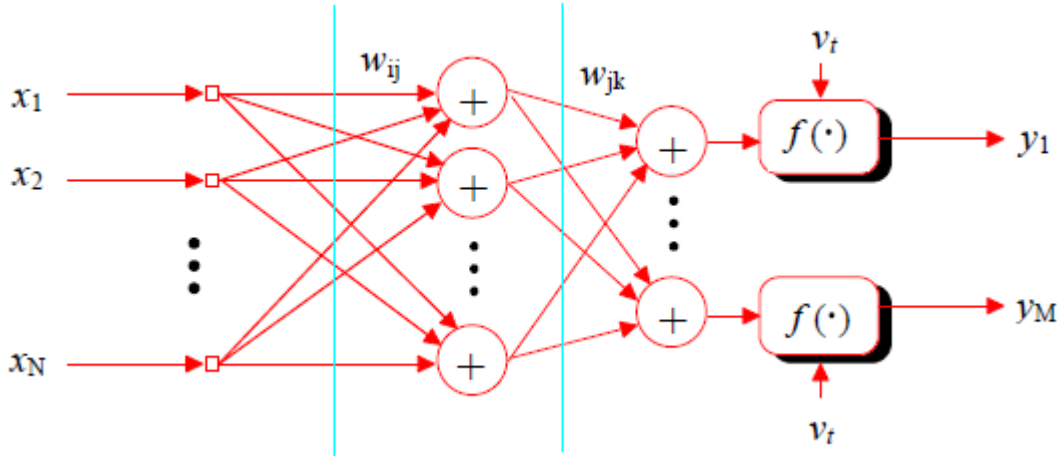

Рис.2.13. Тришарові багатошарові персептрони з одним прихованим шаром і виходом

1) Вхідний шар джерела N;

2) Прихований шар прихованих вузлів K;

3) Вихідний рівень.

Всі три архітектури нейронних мереж, описані досі, використовують активацію функція f (), яка визначається як вихід нейрона з точки зору рівня активності на його вході (від -1 до 1 або від 0 до 1). У табл. 2.1 наведені основні типи функцій активації.

Найбільш практичними функціями активації є сигмоїд і гіперболічний тангенс функції. Це тому, що вони мають похідні.

Таблиця 2.2.

Загальні функції активації

Назва	Визначення
Лінійна	$f(\mathrm{x}) = \mathrm{kx}$
Крокова (зазвичай: b = 1, d = 0, x_k = 0)	$f(\mathrm{x}) = \begin{cases} \beta & \text{якщо } \mathrm{x} \geq \mathrm{x_k} \\ \delta & \text{якщо } \mathrm{x} < \mathrm{x_k} \end{cases}$
Лінійної зміни	$f(\mathrm{x}) = \begin{cases} \rho & \text{якщо } \mathrm{x} \geq \rho \\ \mathrm{x} & \text{якщо } \lvert \mathrm{x} \rvert < \rho \\ -\rho & \text{якщо } \mathrm{x} \leq -\rho \end{cases}$
Сигмоїд	$f(\mathrm{x}) = \frac{1}{1 + \mathrm{e}^{-\mathrm{ax}}}, \alpha > 0$
Гіперболічний тангенс	$f(\mathrm{x}) = \tanh(\gamma x) = \frac{1 - e^{-2x}}{1 + e^{-2x}}, \gamma > 0$
Раціональна	$f(\mathrm{x}) = \begin{cases} \frac{x^2}{1 + x^2} & \text{якщо } x > 0 \\ 0 \end{cases}$
Гаусса	$f(\mathrm{x}) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right]$

2.12. Згорткова нейронна мережа

Згорткова нейронна мережа - це клас глибинних штучних нейронних мереж прямого поширення, який найчастіше застосовувався до аналізу візуальних зображень. ЗНМ використовують різновид багатошарових перцептронів, розроблений так, щоби вимагати використання мінімального обсягу попередньої обробки.

ЗНМ складається з шарів входу та виходу, а також із декількох прихованих шарів. Приховані шари ЗНМ зазвичай складаються зі згорткових шарів, агрегувальних шарів, повноз'єднаних шарів та шарів нормалізації. Цей процес описують в нейронних мережах як згортку за домовленістю. З математичної точки зору він є радше взаємною кореляцією, ніж згорткою. Це має значення лише для індексів у матриці, й відтак які ваги на якому індексі розташовуються.

2.12.1.Згорткові шари

Згорткові шари застосовують до входу операцію згортки, передаючи результат до наступного шару. Згортка імітує реакцію окремого нейрону на зоровий стимул[7].Кожен згортковий нейрон обробляє дані лише для свого рецептивного поля.

Хоч повноз'єднані нейронні мережі прямого поширення й можливо застосовувати як для навчання ознак, так і для класифікування даних, застосування цієї архітектури до зображень є непрактичним. Було би необхідним дуже велике число нейронів, навіть у поверхневій (протилежній до глибинної) архітектурі, через дуже великі розміри входу, пов'язані з зображеннями, де кожен піксель є відповідною змінною. Наприклад, повноз'єднаний шар для (маленького) зображення розміром 100 × 100 має 10 000 ваг. Операція згортки дає змогу розв'язати цю проблему, оскільки вона зменшує кількість вільних параметрів, дозволяючи мережі бути глибшою за меншої кількості параметрів[8]. Наприклад, незалежно від розміру зображення, області замощування розміру 5 × 5, кожна з одними й тими ж спільними вагами, вимагають лише 25 вільних параметрів. Таким чином, це розв'язує

проблему зникання або вибуху градієнтів у тренуванні традиційних багатошарових нейронних мереж з багатьма шарами за допомогою зворотного поширення.

Розмір ємності виходу згорткового шару контролюють три гіперпараметри:

- Глибина ємності виходу контролює кількість нейронів шару, що з'єднуються з однією й тією ж областю вхідної ємності. Ці нейрони вчаться активуватися для різних ознак входу. Наприклад, якщо перший згортковий шар бере як вхід сире зображення, то різні нейрони вздовж виміру глибини можуть активуватися в присутності різних орієнтованих контурів, або плям кольору.
- Крок контролює те, як стовпчики глибини розподіляються за просторовими вимірами (шириною та висотою). Коли кроком є 1, ми рухаємо фільтри на один піксель за раз. Це веде до сильного перекриття рецептивних полів між стовпчиками, а також до великих ємностей виходу. Коли ми робимо крок 2 (або, рідше, 3 чи більше), то фільтри, просуваючись, перестрибують на 2 пікселі за раз. Рецептивні поля перекриваються менше, й отримувана в результаті ємність виходу має менші просторові розміри.[10]
- Іноді зручно доповнювати вхід нулями по краях вхідної ємності. Розмір цього доповнення є третім гіперпараметром. Доповнення забезпечує контроль над просторовим розміром ємності виходу. Зокрема, іноді бажано точно зберігати просторовий розмір вхідної ємності.

2.12.2.Агрегувальні шари

Іншим важливим поняттям ЗНМ є агрегування (англ. pooling), яке є різновидом нелінійного зниження дискретизації. Існує декілька нелінійних функцій для реалізації агрегування, серед яких найпоширенішою є максимізаційне агрегування (англ. max pooling). Воно розділяє вхідне зображення на набір прямокутників без перекриттів, і для кожної такої

підобласті виводить її максимум. Ідея полягає в тому, що точне положення ознаки не так важливе, як її грубе положення відносно інших ознак. Агрегувальний шар слугує поступовому скороченню просторового розміру представлення для зменшення кількості параметрів та об'єму обчислень у мережі, і відтак також для контролю перенавчання. В архітектурі ЗНМ є звичним періодично вставляти агрегувальний шар між послідовними згортковими шарами[9]. Операція агрегування забезпечує ще один різновид інваріантності відносно паралельного перенесення.

Агрегувальний шар діє незалежно на кожен зріз глибини входу, і зменшує його просторовий розмір. Найпоширенішим видом є агрегувальний шар із фільтрами розміру 2×2 (рис. 2.14.), що застосовуються з кроком 2, який знижує дискретизацію кожного зрізу глибини входу в 2 рази як за шириною, так і за висотою, відкидаючи 75 % збуджень. В цьому випадку кожна операція взяття максимуму діє над 4 числами. Розмір за глибиною залишається незмінним.

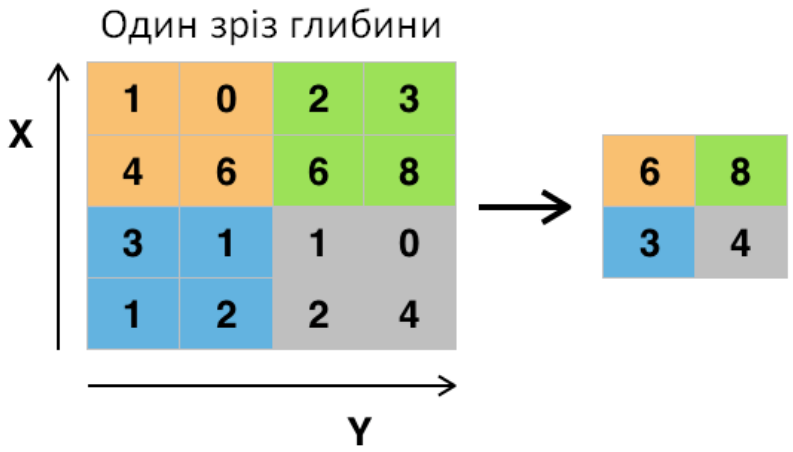

Рис. 2.14. Максимізаційне агрегування

2.12.3.Шар зрізаних лінійних вузлів (ReLU)

ReLU є абревіатурою від англ. Rectified Linear Units, зрізаних лінійних вузлів. Цей шар застосовує ненасичувальну передавальну функцію $f(x) = \max(0, x)$, посилює нелінійні властивості функції ухвалення рішення і мережі в цілому, не зачіпаючи рецептивних полів згорткового шару.

Для посилення нелінійності застосовуються й інші функції, наприклад, насичувальні гіперболічний тангенс $f(x) = \tanh(x)$, $f(x) = |\tanh(x)|$, та сигмоїдна функція $f(x) = (1 + e^{-x})^{-1}$. ReLU часто віддають перевагу перед іншими функціями, оскільки він тренує нейронну мережу в декілька разів швидше[11] без значної розплати точністю узагальнення.

2.12.4.Повноз'єднаний шар

Насамкінець, після кількох згорткових та максимізаційно агрегувальних шарів, високорівневі міркування в нейронній мережі здійснюються повноз'єднаними шарами (англ. fully connected layers). Нейрони у повноз'єднаному шарі мають з'єднання з усіма збудженнями попереднього шару, як це можна бачити у звичайних нейронних мережах. Їхні збудження відтак може бути обчислювано матричним множенням, за яким слідує зсув упередженості.

2.12.5.Шар втрат

Шар втрат визначає, як тренування штрафує відхилення між передбаченими та справжніми мітками, і є, як правило, завершальним шаром. Для різних завдань у ньому можуть використовувати різні функції втрат. Нормалізовані експоненційні втрати (англ. softmax) застосовуються для передбачення єдиного класу з K взаємно виключних класів. Сигмоїдні перехресно-ентропійні втрати застосовуються для передбачення K незалежних значень імовірності в проміжку [0,1]. Евклідові втрати застосовуються для регресії до дійснозначних міток.

2.13. Висновки

Використання ЗНМ цілком задовольняє вимоги до вирішення задачі класифікації, але показники якості та точності залежать від правильної конфігурації нейронної мережи, чіткості поділу вхідних даних на класи та наявності взаємозв’язку між параметрами цих даних.

З іншого боку спектрограма – найбільш характеризоване та зрозуміле для глибинного навчання відображення звукових хвиль. Форматування цієї спектограми, ширина вибірки даних, та використання необхідної функції віконного трансформування Фур'є ще підлягають дослідженню з метою вдосконалення матодики класифікації.

РОЗДІЛ 3. РЕАЛІЗАЦІЯ МЕТОДИКИ КЛАСИФІКАЦІЇ АУДІО ФАЙЛІВ

3.1. Загальний план вирішення задачі класифікації бібліотеки аудіо файлів

Застосування будь якої нейронної мережі у чистому вигляді не дасе ніяких результатів, бо ця мережа нічого не знає про те, що саме вона повинна класифікувати та за якими ознаками. Тому спочатку треба підготовити дані для тренування, та декілька з них виділити для тестування навченої мережі. При цьому кожен аудіо файл обов'язково повинен бути поміченим відповідним класом (наприклад, жанром) завдяки розташуванні у відповідній директорії.

Після цього треба буде сконфігорувати нейронну мережу за згортковою моделлю враховуючи кількість пікселів сформованих спектрограм для вірного порядку та кількості необхідних шарів нейронів.

Останнім кроком ми натренуємо модель глибинного навчання та запустемо тестувальні дані для прогнозування жанрів, до яких вони ймовірніше відносяться.

Отже, формулювання поетапного виконання виглядає наступним чином:

- Створення спрощеного уявлення кожної пісні в бібліотеці
- Тренування глибинної нейронної мережі для класифікації пісень
- Застосування класифікатору для заповнення жанрів в нашій бібліотеці

3.2. Уточнення вхідних даних

Так як більшість домашніх бібліотек налічуэ не більше 2000 екземплярів музики, треба враховувати, що до кожного жанру повинна відноситись велика кількість музичних файлів (хоча б більше 100), щоб нейронна мережа змогла виявити закономірності та розділити їх для різних жанрів. Тобто маючи невелику бібліотеку необхідно абстрагуватися від піджанрів і привести мітки файлів до батьківського жанру (супер-жанру), приклад яких відображений на

рис. 3.1. Більш точну класифікацію за піджанрами можна робити тільки якщо вибірка зодовльняє умову вище.

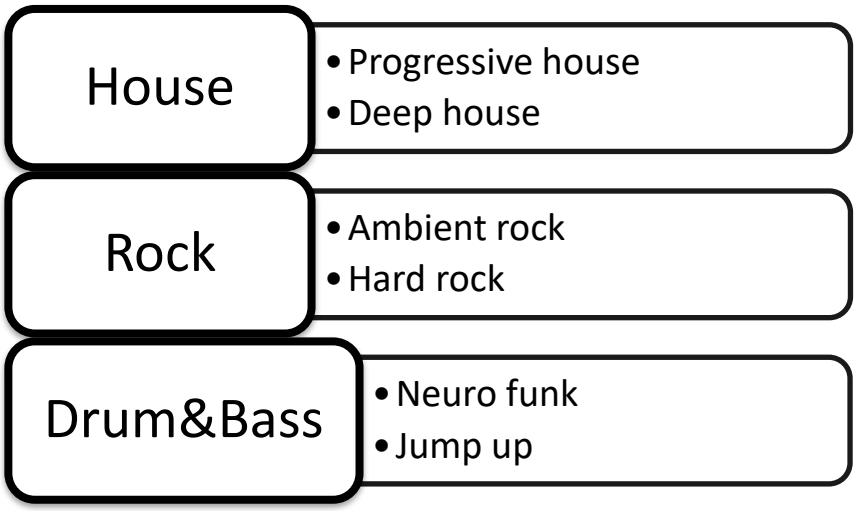

Рис. 3.1. Узагальнення піджанрів

3.3. Перетворення аудіо даних

Класична частота дискретизації становить 44100 Гц - на кожну секунду звуку записано 44100 значень і в два рази більше для стерео. Це означає, що 3-хвилинна стереофонічна пісня містить 7 938 000 зразків. Для нас це дуже великий обсяг даних, відповідно, потрібно зменшити його до більш керованого рівня, щоб ми мали можливість виробляти над нам будь-які операції. Для початку можна відкинути стереоканал, оскільки він містить надлишкову інформацію.

Ми скористаємося перетворенням Фур'є для відображення наших аудіо даних у вигляді частотної області. Це більш просте і компактне представлення даних, які ми будемо експортувати у вигляді спектрограми. В результаті цього ми отримаємо PNG-файл, який містить еволюцію всіх частот нашої пісні в часі.

Частота дискретизації 44100 Гц, про яку ми говорили раніше, дозволяє нам відновлювати частоти до 22050 Гц (за теоремою Котельникова [12]), але тепер, коли частоти витягнуті, ми можемо використовувати набагато більше значення роздільної здатності. Встановимо точність відображення на спектрограмі 50

пікселів в секунду (20 мс на піксель). Цього більш ніж достатньо, щоб бути впевненим у використанні всієї необхідної нам інформації.

На рис. 3.2. зображена відрізок музики після процесу перетвореня (зразок 12.8s).

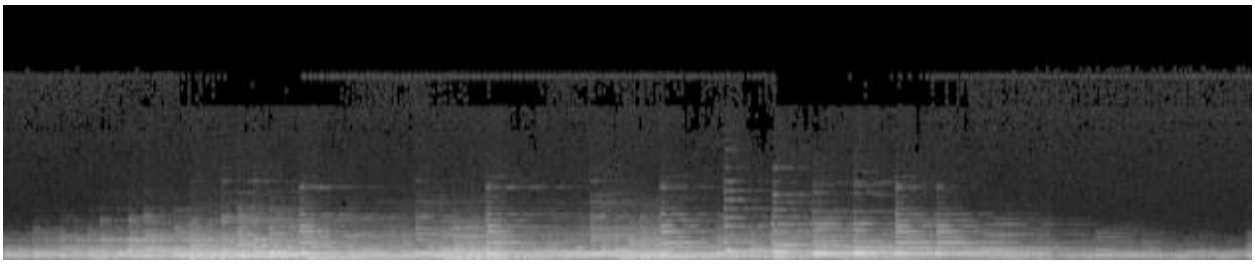

Рис. 3.2. Спектрограма витягу з пісні

Масштабована амплітуда частоти зображена у відтінках сірого, де білий є максимальним, а чорний - мінімальним. Я використовував спектрограму зі 128 частотними рівнями, тому що в ній міститься вся необхідна інформація про пісню - ми можемо з легкістю відрізнити різні частоти.

На етапі анализу найбільш зручної нейронної мережі встав вибір між рекурентною та згортковою. Для першої необхідно було б «годувати» кожен стовпчик зображення окремо по порядку , але людина здатна визначити жанр музикальної композиції за декілька секунд, тому цей варіант віявився не найкращим.

Для прискорення роботи згорткової моделі створему фрагменти фіксованої довжини спектрограми і розглянемо їх як незалежні зразки, що представляють жанр. Для цього ми можемо використовувати квадратні скибочки, скоротивши спектрограму до 128x128 пікселів (рис. 3.3.).

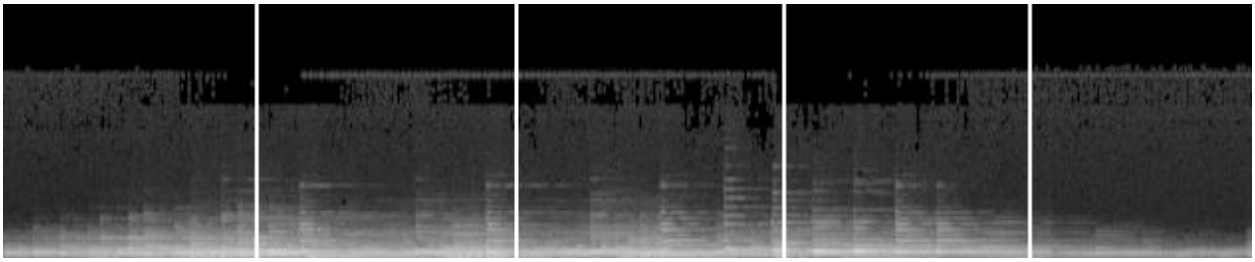

Рис. 3.3. «Нарізана» спектрограма

3.4. Побудова моделі

Після того, як ми нарізали всі наші аудіо файли на квадратні спектральні зображення, у нас є набір даних, що містить десятки тисяч зразків для кожного жанру. Тепер ми можемо навчити глибоку згоркову нейронну мережу для класифікації цих зразків. Для цієї мети я використовував обгортку Deeplearning4J.

3.4.1. Бібліотека Deeplearning4j

Deeplearning4j - бібліотека, яка реалізує глибинне навчання. Для цієї технології потрібно більше обчислювальних ресурсів, ніж для класичних нейронних мереж. Всі складні обчислення в DL4J здійснюються окремим нативним модулем, який працює незалежно від JVM, що дозволяє прискорити роботу бібліотеки. Обчислення може виконуватись як безпосередньо на процесорі, так і на розподілених ресурсах і графічних прискорювачах [14]. DL4J дозволяє створювати згорткові нейронні мережі, що важливо при обробці зображень. Все конфігурація проводиться за допомогою програмного коду. На рис.3.4. показано визначення мережі LeNet. LeNet - це класична сверточное мережу, яка складається з 5 шарів.

```
MultiLayerConfiguration conf = new NeuralNetConfiguration.Builder()
    .layer(1, maxPool("maxpool1", new int[]{2,2}))
    .layer(2, conv5x5("cnn2", 100, new int[]{5, 5}, new int[]{1, 1}, 0))
    .layer(3, maxPool("maxool2", new int[]{2,2}))
    .layer(4, new DenseLayer.Builder().nOut(500).build())
    .layer(5, new OutputLayer.Builder(LossFunctions.LossFunction.NEGATIVELOGLIKELIHOOD)
        .nOut(numLabels)
        .activation(Activation.SOFTMAX)
        .build())
    .backprop(true).pretrain(false)
    .setInputType(InputType.convolutional(height, width, channels))
    .build();
```

Рис. 3.4. Приклад конфігурації нейронної мережі у deeplearning4j

3.4.2. Конфігурація згорткової мережі

Етапи згортання зображені на рис. 3.4.

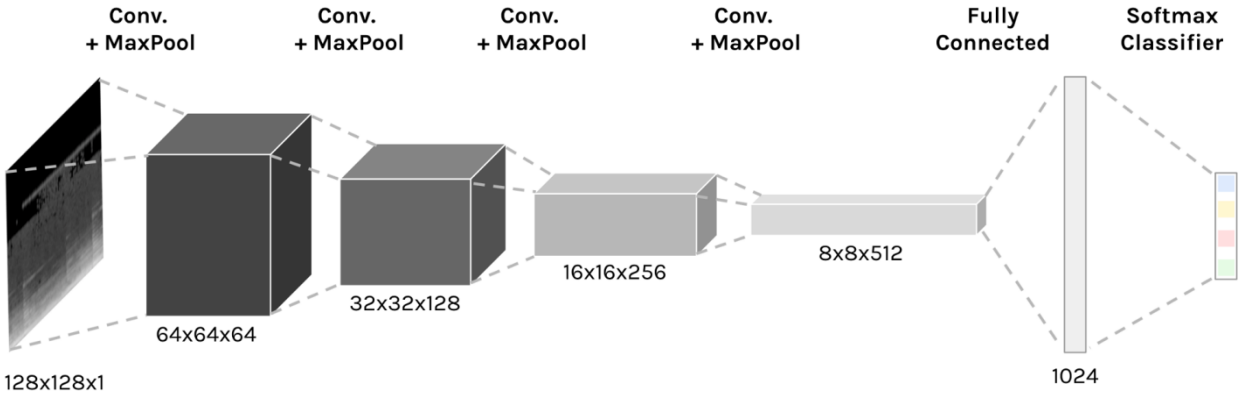

Рис.3.4. Модель згорткової нейронної мережі

Експериментальним шляхом було виведено такі параметри моделі:

1) Шар: Ядра розміром 2x2 із кроком 2
2) Оптимізатор: RMSProp.
3) Функція активації: ELU (Експоненціальний лінійный блок), за високу продуктивністю, яку він показал у порівнянні з ReLU [13]
4) Ініціалізація: XAVIER для матриць ваг на всіх шарах.
5) Регуляризація: виключення з ймовірністю 0,5

З 2 000 піснями, розділеними між 6 жанрами - Hardcore, Dubstep, Electro, Classical, Soundtrack і Rap, і загальними фрагментами більше 12 000 128x128 спектрограмм, модель досягла точності 90% на наборі перевірки. Це дуже добре, особливо якщо врахувати, що ми обробляємо пісні маленькими бітами за раз. Зауважте, що це не остаточна точність, яку ми матимемо для класифікації цілих пісень (це буде ще краще). Зараз йде річ лише за шматочки.

3.5. Тестування класифікатора

До сих пір ми перетворили наші пісні з стерео в моно і створили спектрограму, яку ми нарізали невеликими бітами. Потім ми використовували ці зрізи для навчання глибокої нейронної мережі. Тепер ми можемо використовувати модель для класифікації нової пісні, про яку нейронна мережа ще нічого не знаю.

Ми починаємо зі створення спектрограми так само, як і з даними навчання. Використовуючи нарізки ми не можемо передбачити клас пісні за один раз. Ми повинні нарізати нову пісню, а потім зібрати та проаналізувати передбачені класи для всіх фрагментів.

Для цього ми будемо використовувати систему голосувань, зразок роботи якої зображено на рис. 3.5. Кожен зразок треку буде «голосувати» за жанр, після чого ми вибираємо жанр з більшістю голосів. Це збільшить нашу точність, як ми позбудемося багатьох помилок класифікації з цим способом навчання. А так як одна аудіо доріжка розрізається десь на 70 шматків, тобто має достатню дрібність із шагом визначення 1.45, тому ми можемо гарантувати актуальність та достовірність вихідних результатів.

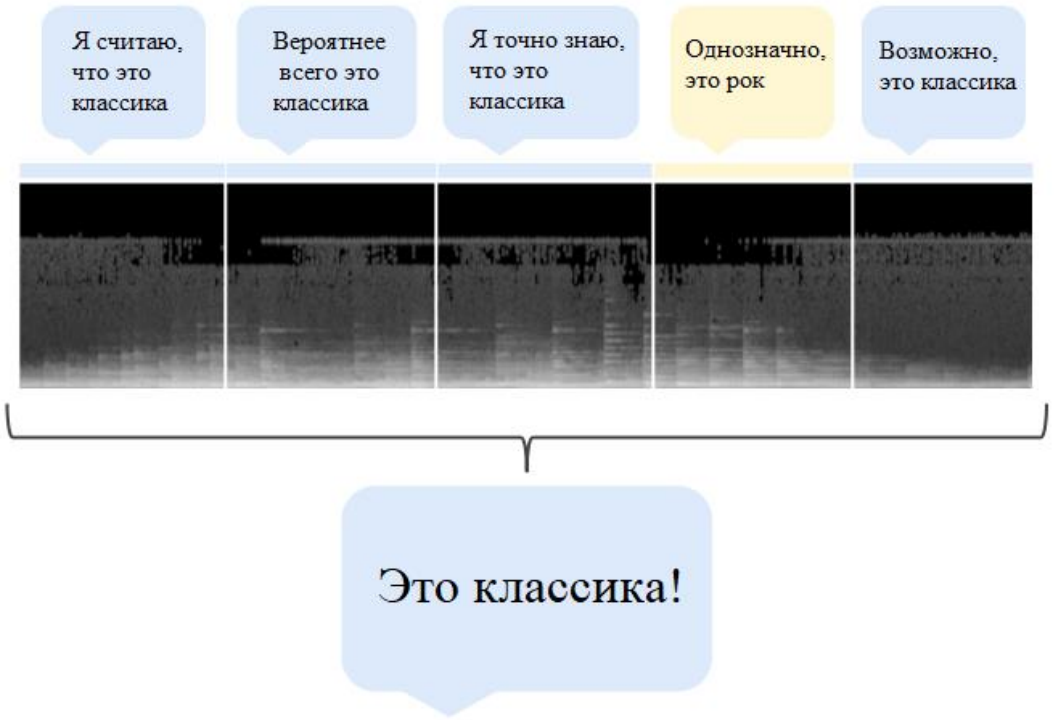

Рис. 3.5. Принцип роботи системи голосувань

З цим конвеєром (рис. 3.6.) ми тепер здатні швидко класифікувати немарковані пісні з нашої бібліотеки. Ми могли б просто запустити систему голосування на всі пісні, для яких нам потрібен жанр, і взяти вихідне слово класифікатора. Це дало б добрі результати, але все одно можуть видаватись невірні твердження відносно реального жанру музики.

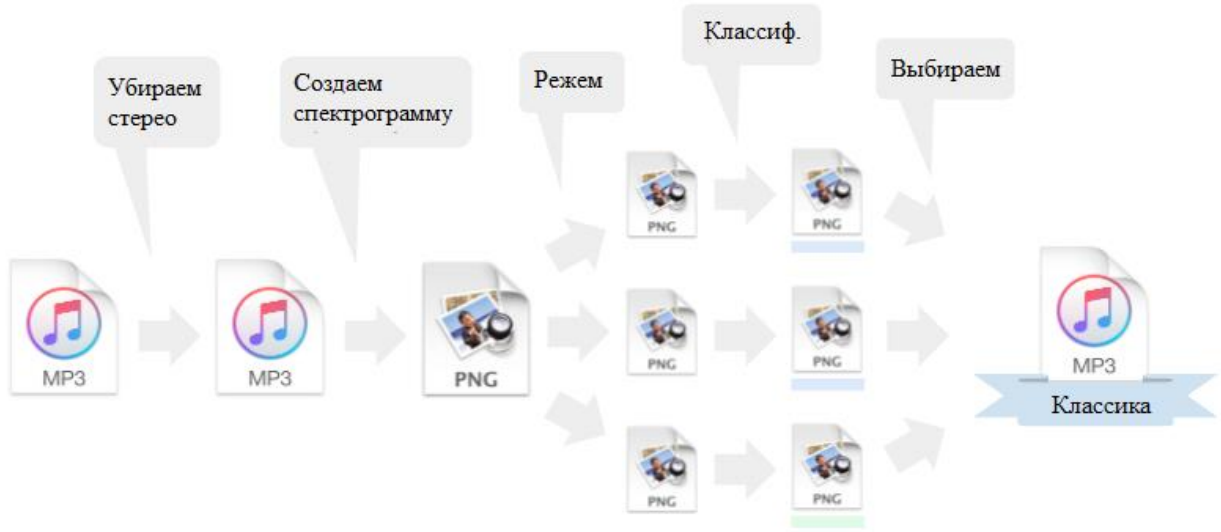

Рис. 3.6. Етапи конвеєру класифікації

3.6. Поліпшення системи голосування

Останній шар класифікатора, який ми створили, є шаром softmax. Це означає, що він виводить не виявлений жанр, а ймовірності кожного з них. Це явище називається довірою класифікації (рис. 3.7.).

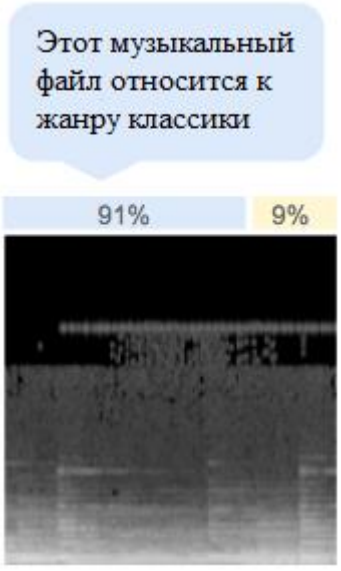

Рис. 3.7. Довіра класифікації

Корисно буде використати це для визначення впевненності видачі класифікатором відповіді відносно жанру музичної композіції. Тому введемо поріг впевненності більше 55% у процес аналізу відрізку. Таким чином ми уникнемо випадків невірного визначення класу, дотримуючись судження, що краще не видати результат зовсім аніж ввести в оману користувача (рис. 3.8.).

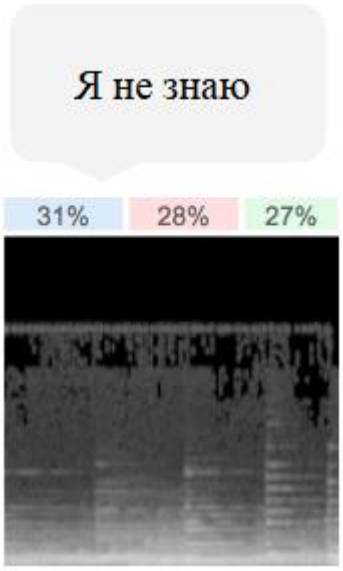

Рис. 3.8. Незадовільнення порогу впевненності

У будь-якому випадку усі нерозмічени аудіо файли доведеться власноруч позначити до того, чи іншого класу, бо вони є нестандартними.

Також необхідно встановити поріг впевненності на визначення рельтуючого жанру із відповідей аналізатору окремих відрізків музичної композиції. Задовільним значенням буде поріг у 70%. Аудіо файли, визначення жанру яких не перевищило цей поріг впевненності вважаються також немаркованими (рис. 3.9.).

Завдяки більш великої вибірки тренувальних даних можливо мінімізувати невизначеність і малу довіру класифікації нейронної мережі.

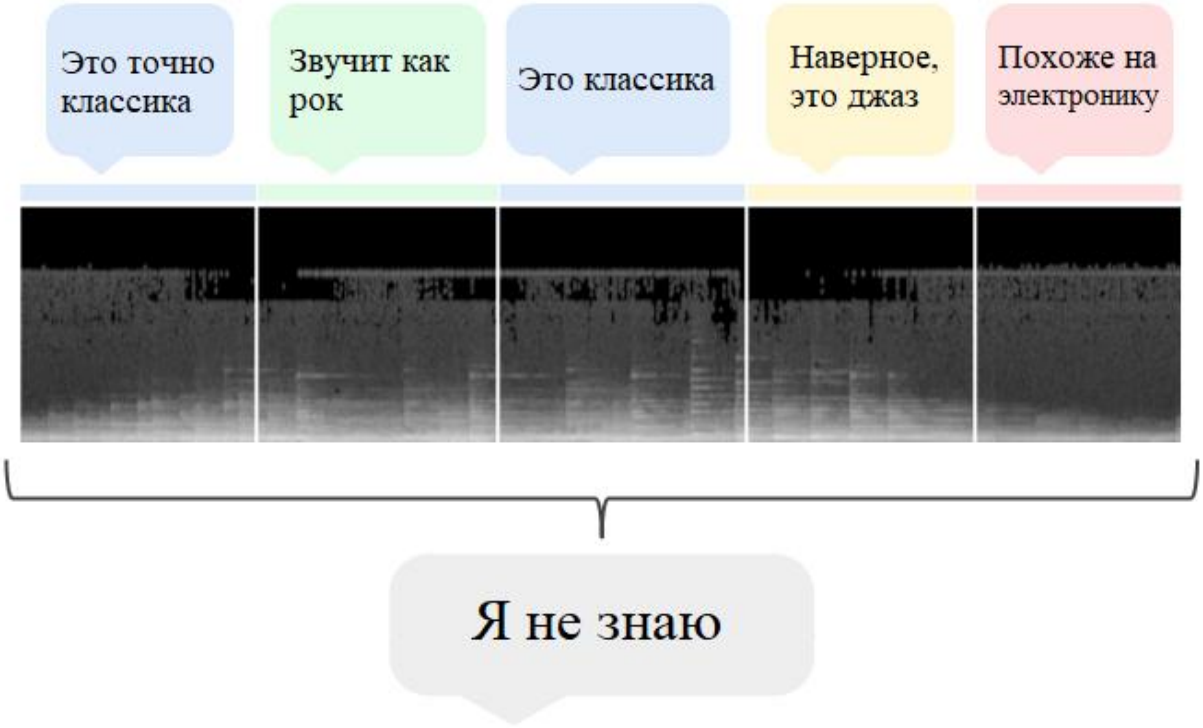

Рис. 3.9. Музична композиція залишена без відмітки жанру у зв'язку з низкою довірою до системи голосування

3.7. Висновки

Використання моделі глибинного навчання у сукупності з перетворенням вхідного аудіо сигналу на спектрограму частот повністю вирішило проблему класифікації великого масиву аудіо даних. Але й ця система не позбавлена недоліків, таких як відносно повільна процедура Фур'є перетворень та групування отриманих частотних спектрів до єдиної спектрограми; точності класифікації все одно залежна від кількості вхідної вибірки.

РОЗДІЛ 4. ЕКОНОМІЧНА ЧАСТИНА

4.1. Маркетингові дослідження ринку збуту розробленого продукту

Музика – є важливою частиною нашого життя. Багато ресурсів надають можливість прослуховувати, оцінювати, обговорювати, купляти, та навіть створювати власноруч музичні композиції. Кожен час автори по всьому світі видають нові треки, пісні та навіть альбоми. Все більше прогресивних авторів експерементують з новими жанрами, їх комбінаціями для розширення кордонів відчуття музики. У всьому цьому різноманітті жанрів, піджанрів, класів музики простіше заблукати, не знайти потрібний користувачеві аудіо продукт. Багато з нас коллекціонують свою бібліотеку, і чім більше вона стає, тим складніше власноруч її класифікувати. Рішенням циєї проблеми і стаю розроблена модель класифікації.

Вчені та програмісти вже довго вивчають процеси машинного навчання, вдосконалюють їх, виводять нові теорії та твердження, розробляють нові алгоритми. Із розвитком технічної складової компьютерів йде вперед і розвиток машинного навчання, стає біль чітким, точним, здатним до аналізу все більш великого набору даних. З'являються поняття нейронних мереж, нові алгоритми та формули розрахунку залежностей.

Як результат такого розвитку стає відомою та затребуваною галузь глибинного навчання, яка прийшла на зміну вже застарілого поняття нейронних мереж. Вона розширює звичні процеси компьютерного навчання, розподіляючи потужності розрахунку між декількома системами, переводячи відповідальність з процесорів на більш потужні та цільові у вирішенні подібних задач графічні прискорювачи. Одної з перспективних розробок у цьому напрямі є біблиотека Deeplearning4j для конфігорування та вбудовання в проекти, розроблені на язику Java. Вона відповідає всім вимогам глибинного навчання та має весь перелік доступних функцій оптимізації, активації, тонкого налаштування процесу навчання мережі.

Розроблена методика класифікації аудіо файлів використовує найсучасні досягнення у вирішенні задач класифікації та розпізнаванні образів, та

впроваджує новий метод аналізу та формування вигляду аудіо даних з подальшою класифікацією за класами, жанрами тощо. У звязку з чим очікується немалий вклад у розвиток аналізу музикальних композицій, та у глибинне навчання зокрема.

Крім того багато сервісів та застосунків не мають такого функціоналу пошуку та розподілу, що перешкоджає зручності їх використання що несе за собою малу популярність серед користувачів. Впровадження системи, заснованої на розробленій методиці та моделі класифікації може вирішити ці проблеми. Цей факт відіграє важливу роль у виділенні ринку збиту продукту.

4.2. Оцінка економічної ефективності впровадження розробленого алгоритму

У наш час існує багато музичних магазинів, що продаєть аудіо композиції різних авторів та співаків. З кожним новим альбомом відомі співаки та групи стають ще більш відомимм і бажаними до прослуховування. У тіні цих знаменитостей знаходяться меньш відомі, але не меньш талановиті автори. Та за відсутністю грошей або правильної рекламної компанії вони залишаються невідомими.

Впровадження реалізації розробленої методики на музичні веб магазини та застосунки може вирішити цю проблему. Користувач буде здатний шукати композиції, які відносяться до тих жанрів, які йому довподоби. Завдяки цьому з одного боку очікується підвищення конверсії покупок, що сприяє розвитку музичного бізнесу, та з іншого – поліпшення процесу пошуку інтересуючого аудіо контенту та якості наданих магазином послуг.

Соціальний ефект даної роботи полягає в удосконаленні порядку надання необхідних аудіо даних та поліпшенню розподілу вже наявної бібліотеки музики, що економить час, лишаючи необхідності віконання цих дій у ручному режимі.

4.3. Висновки

У цій роботі зроблено значний внесок у розвиток систем упорядкування музичної бібліотеки веб-порталів, програмного запезпечення та власних компьютерів завдяки автоматизації кропітливої праці ручного прослуховування та класифікації аудіо файлів.

Виходячи з маркетингових досліджень ринку збуту розробленого алгоритму та соціальний ефект, який він створює на теперішній час, можна дійти висновку, що ефективність даного впровадження обумовлена позитивною оцінкою та заслуговує своє місце на ринку збуту у сфері інформаційних технологій та музики.

ВИСНОВКИ

У даній роботі проведено аналіз проблем зберігання та пропозицій аудіо файлів. Підібрані технології для успішного вирішення ціх проблем. Показані приклади моделювання згорткових нейронних мереж, розкрита структура та принципи їх побудови. Проведено тренування та тестування розробленої методики на власній бібліотеці аудіо файлів.

Результати цієї роботи роблять значний внесок у розвиток глибинного навчання ідентифікації аудіо файлів за допомогою технологій розпізнання образів. Класифікація музичних композицій - дуже складна проблема, особливо коли постає завдання розподілу між усіма доступними жанрами музики. Обгрунтована методика пропонує варіант вирішення цієї проблеми і вносить свій вклад в створення основи для більш широкої класифікації музичних композицій, та навіть виведення кореляції різних масивів класів між собою з метою знаходження нових залежностей групування та розуміння природи аудіо даних.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Nikil Jayant, James Johnston, Robert Safranek. (October 1992). «Signal Compression Based on Models of Human Perception».
2. R. Lippmann, “An Introduction to Computing with Neural Networks,” IEEE ASSP Ma, 1987. C. 4-22.
3. D. R. Hush and B. G. Horne, “Progress in Supervised Neural Networks: What’s New Since Lippmann?” IEEE Signal Proc. Mag., 1993. C. 8-39.
4. Glauner, P. (2015). Deep Convolutional Neural Networks for Smile Recognition (MSc Thesis).
5. Bengio, Y.; Courville, A.; Vincent, P. (2013). Representation Learning: A Review and New Perspectives. IEEE Transactions on Pattern Analysis and Machine Intelligence
6. Deng, L.; Yu, D. (2014). Deep Learning: Methods and Applications. Foundations and Trends in Signal Processing
7. Convolutional Neural Networks (LeNet) – DeepLearning 0.1 documentation. DeepLearning 0.1. LISA Lab.
8. Habibi, Aghdam, Hamed. Guide to convolutional neural networks : a practical application to traffic-sign detection and classification.
9. Ciresan, Dan; Ueli Meier; Jonathan Masci; Luca M. Gambardella; Jurgen Schmidhuber (2011). Flexible, High Performance Convolutional Neural Networks for Image Classification. Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume Two
10. CS231n Convolutional Neural Networks for Visual Recognition. http://cs231n.github.io
11. Krizhevsky, A.; Sutskever, I.; Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems
12. Теория электрической связи: учебное пособие / К.К. Васильев, В.А. Глушков, А.В. Дормидонтов, А.Г. Нестеренко; под общ. ред. К.К. Васильева. – Ульяновск: УлГТУ, 2008

13. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs) Djork-Arne Clevert, Thomas Unterthiner & Sepp Hochreiter
14. Deeplearning4j: Open-source, Distributed Deep Learning for the JVM // Deep Learning for Java. https://deeplearning4j.org/
15. Deep learning for music genre classification.Tao Feng
16. J. A. Russell (1980), A circumspect model of affect Journal of Psychology and Social Psychology
17. Музыкальные стили. http://websound.ru/styles_r.htm
18. Царегородцев В.Г. Уточнение решения обратной задачи для нейросети-классификатора. Нейрокомпьютеры: разработка, применение. 2003
19. Глубокое обучение. Погружение в мир нейронных сетей. Николенко С. И., Кадурин А. А., Архангельская Е. О. 2018
20. Машинное обучение. Хенрик Бринк, Джозеф Ричардс, Марк Феверолф
21. Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных. Петер Флах 2015
22. Машинное обучение - состояние и перспективы. Д.П. Ветров
23. Artificial Intelligence: A Modern Approach. Stuart Russell and Peter Norvig
24. Fundamentals of Sound and Time-Frequency Representations. Juan Pablo Bello
25. Pohlmann, K.C. “Principles of Digital Audio”. 6th Edition. McGraw Hill (2011):chapter 2.
26. Deep content-based music recommendation. Aaron van den Oord, Sander Dieleman, Benjamin Schrauwen
27. Ruslan Salakhutdinov and Andriy Mnih. *Probabilistic matrix factorization*. In Advances in Neural Information Processing Systems, volume 20, 2008.
28. LeCun, Yann. LeNet-5, convolutional neural networks. November 2013
29. Ciresan, Dan; Ueli Meier; Jonathan Masci; Luca M. Gambardella; Jurgen Schmidhuber (2011). Flexible, High Performance Convolutional Neural Networks for Image Classification

ДОДАТОК А

ВІДГУК

на дипломну роботу магістра на тему:

«Обгрунтування методики класифікації аудіо файлів для музичного програмного забезпечення та веб порталів»

студента групи 122М-16-1 Терехова Володимира Андрійовича

Метою даної магістерської роботи є підвищення швидкості та вдосконалення процесу пошуку та пропозицій аудіо файлів, у тому числі музичних композицій, у різних сферах застосування за допомогою методів глибинного навчання.

Актуальність даної теми зумовлена наявністю значних недоліків у загально прийнятому способі поділу аудіо файлів на групи: значні затрати часу на пошук необхідних файлів, неточність класифікації..

Тема дипломної роботи безпосередньо пов'язана з об'єктом діяльності магістра спеціальності 122 «Комп’ютерні науки» області знань 12 «Інформаційні технології» - створення та дослідження моделей та програмних засобів.

Наукова новизна результатів, що очікуються, полягає у проведенні аналізу і виявленні недоліків емпіричного підходу класифікації аудіофайлів, а також у створенні методики глибинного навчання нейронних мереж з метою розпізнавання класів музичних файлів з високою точністю. Оригінальність технічних рішень при розробці алгоритму полягає в використанні передових методів представлення аудіо даних у зв’язці з технологією глибинного навчання.

Практична цінність результатів полягає в ефективному навчанні нейронної мережі на підставі отриманої методики й подальше використання навченої системи в реалізації програмного забезпечення для роботи з аудіо файлами.

Оформлення дипломної роботи магістра виконано на сучасному рівні і відповідає вимогам, що пред'являються до робіт даної кваліфікації. Ступінь самостійності виконання досить висока.

Дипломна робота магістра в цілому заслуговує оцінки «відмінно», а сам автор - присвоєння кваліфікації «інженер з комп’ютерних систем».

Керівник дипломної
роботи магістра, д.т.н.,
проф. кафедри ПЗКС М.О. Алексєєв

ДОДАТОК Б

РЕЦЕНЗІЯ

на дипломну роботу магістра на тему:

«Обгрунтування методики класифікації аудіо файлів для музичного програмного забезпечення та веб порталів»

студента групи 122М-16-1 Терехова Володимира Андрійовича

Більшість музичних веб ресурсів використовують стандартні методи розподілу на групи аудіо файлів: за автором, назвою, мовою виконавця тощо; а пошук взагалі можливий лише за назвою / автором композиції. Частіш за все бувають випадки, коли не відоме ні те, ні інше. Користуючись таким скромним інструментарієм потенційний покупець не зможе знайти необхідний музичний товар та зробити бажану покупку. У найкращих випадках на це може піти багато часу, який у нинішньому віці швидкої інформації дуже цінний.

Актуальність даної магістрської роботи виражена обгрунтуванням методики вирішення перелічених проблем та недоліків загально прийнятої структури зберігання та класифікації аудіо файлів.

Тема дипломної роботи безпосередньо пов'язана з об'єктом діяльності магістра спеціальності 122 «Комп’ютерні науки» області знань 12 «Інформаційні технології» - створення та дослідження моделей та програмних засобів.

Наукова новизна результатів, що очікуються, полягає у проведенні аналізу і виявленні недоліків емпіричного підходу класифікації аудіофайлів, а також у створенні методики глибинного навчання нейронних мереж з метою розпізнавання класів музичних файлів з високою точністю.

Студент В.А. Терехов досить добре розібрався в специфіці зберігання аудіо файлів та використанні технологій глибинного навчання.

Беручи до уваги вище викладене, можна зробити висновок, що дана робота цілком відповідає вимогам, що пред'являються до кваліфікаційних робіт рівня магістра.

З огляду на наукову новизну і ступінь опрацювання компонентів даної роботи, в цілому автор заслуговує оцінки «відмінно», а також присвоєння кваліфікації «інженер з комп’ютерних систем».

Програма класифікації аудіо файлів за допомогою використання згорткової нейронної мережі.

```
package org.deeplearning4j.examples.convolution.mnist;

import java.io.File;
import java.util.HashMap;
import java.util.Map;
import java.util.Random;

import org.datavec.api.io.labels.ParentPathLabelGenerator;
import org.datavec.api.split.FileSplit;
import org.datavec.image.loader.NativeImageLoader;
import org.datavec.image.recordreader.ImageRecordReader;
import org.deeplearning4j.datasets.datavec.RecordReaderDataSetIterator;
import org.deeplearning4j.eval.Evaluation;
import org.deeplearning4j.examples.utilities.DataUtilities;
import org.deeplearning4j.nn.api.OptimizationAlgorithm;
import org.deeplearning4j.nn.conf.LearningRatePolicy;
import org.deeplearning4j.nn.conf.MultiLayerConfiguration;
import org.deeplearning4j.nn.conf.NeuralNetConfiguration;
import org.deeplearning4j.nn.conf.Updater;
import org.deeplearning4j.nn.conf.inputs.InputType;
import org.deeplearning4j.nn.conf.layers.ConvolutionLayer;
import org.deeplearning4j.nn.conf.layers.DenseLayer;
import org.deeplearning4j.nn.conf.layers.OutputLayer;
import org.deeplearning4j.nn.conf.layers.SubsamplingLayer;
import org.deeplearning4j.nn.multilayer.MultiLayerNetwork;
import org.deeplearning4j.nn.weights.WeightInit;
import org.deeplearning4j.optimize.listeners.ScoreIterationListener;
import org.deeplearning4j.util.ModelSerializer;
import org.nd4j.linalg.activations.Activation;
import org.nd4j.linalg.dataset.api.iterator.DataSetIterator;
import org.nd4j.linalg.dataset.api.preprocessor.DataNormalization;
import org.nd4j.linalg.dataset.api.preprocessor.ImagePreProcessingScaler;
```

```
import org.nd4j.linalg.lossfunctions.LossFunctions;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

public class ConvClassifier {

  private static final Logger log = LoggerFactory.getLogger(ConvClassifier.class);
  private static final String basePath = System.getProperty("java.io.tmpdir") + "/mnist";
  private static final String dataUrl = "http://github.com/myleott/mnist_png/raw/master/mnist_png.tar.gz";

  public static void main(String[] args) throws Exception {
    int height = 28;
    int width = 28;
    int channels = 1; // single channel for grayscale images
    int outputNum = 10; // 10 digits classification
    int batchSize = 54;
    int nEpochs = 1;
    int iterations = 1;

    int seed = 1234;
    Random randNumGen = new Random(seed);

    log.info("Data load and vectorization...");
    String localFilePath = basePath + "/mnist_png.tar.gz";
    if (DataUtilities.downloadFile(dataUrl, localFilePath))
      log.debug("Data downloaded from {}", dataUrl);
    if (!new File(basePath + "/mnist_png").exists())
      DataUtilities.extractTarGz(localFilePath, basePath);

    // vectorization of train data
    File trainData = new File(basePath + "/mnist_png/training");
    FileSplit trainSplit = new FileSplit(trainData, NativeImageLoader.ALLOWED_FORMATS, randNumGen);
```

```
    ParentPathLabelGenerator labelMaker = new ParentPathLabelGenerator(); //
parent path as the image label
    ImageRecordReader trainRR = new ImageRecordReader(height, width, channels,
labelMaker);
    trainRR.initialize(trainSplit);
    DataSetIterator trainIter = new RecordReaderDataSetIterator(trainRR,
batchSize, 1, outputNum);

    // pixel values from 0-255 to 0-1 (min-max scaling)
    DataNormalization scaler = new ImagePreProcessingScaler(0, 1);
    scaler.fit(trainIter);
    trainIter.setPreProcessor(scaler);

    // vectorization of test data
    File testData = new File(basePath + "/mnist_png/testing");
    FileSplit testSplit = new FileSplit(testData,
NativeImageLoader.ALLOWED_FORMATS, randNumGen);
    ImageRecordReader testRR = new ImageRecordReader(height, width, channels,
labelMaker);
    testRR.initialize(testSplit);
    DataSetIterator testIter = new RecordReaderDataSetIterator(testRR,
batchSize, 1, outputNum);
    testIter.setPreProcessor(scaler); // same normalization for better results

    log.info("Network configuration and training...");
    Map<Integer, Double> lrSchedule = new HashMap<>();
    lrSchedule.put(0, 0.06); // iteration #, learning rate
    lrSchedule.put(200, 0.05);
    lrSchedule.put(600, 0.028);
    lrSchedule.put(800, 0.0060);
    lrSchedule.put(1000, 0.001);

    MultiLayerConfiguration conf = new NeuralNetConfiguration.Builder()
        .seed(seed)
        .iterations(iterations)
        .regularization(true).l2(0.0005)
        .learningRate(.01)
```

```
        .learningRateDecayPolicy(LearningRatePolicy.Schedule)
        .learningRateSchedule(lrSchedule)  //  overrides  the  rate  set  in
learningRate
        .weightInit(WeightInit.XAVIER)
        .optimizationAlgo(OptimizationAlgorithm.STOCHASTIC_GRADIENT_DESCENT)
        .updater(Updater.NESTEROVS)
        .list()
        .layer(0, new ConvolutionLayer.Builder(5, 5)
            .nIn(channels)
            .stride(1, 1)
            .nOut(20)
            .activation(Activation.IDENTITY)
            .build())
        .layer(1,                                                          new
SubsamplingLayer.Builder(SubsamplingLayer.PoolingType.MAX)
            .kernelSize(2, 2)
            .stride(2, 2)
            .build())
        .layer(2, new ConvolutionLayer.Builder(5, 5)
            .stride(1, 1) // nIn need not specified in later layers
            .nOut(50)
            .activation(Activation.IDENTITY)
            .build())
        .layer(3,                                                          new
SubsamplingLayer.Builder(SubsamplingLayer.PoolingType.MAX)
            .kernelSize(2, 2)
            .stride(2, 2)
            .build())
        .layer(4, new DenseLayer.Builder().activation(Activation.RELU)
            .nOut(500).build())
        .layer(5,                                                          new
OutputLayer.Builder(LossFunctions.LossFunction.NEGATIVELOGLIKELIHOOD)
            .nOut(outputNum)
            .activation(Activation.SOFTMAX)
            .build())
        .setInputType(InputType.convolutionalFlat(28,      28,      1))      //
InputType.convolutional for normal image
```

```
        .backprop(true).pretrain(false).build();

    MultiLayerNetwork net = new MultiLayerNetwork(conf);
    net.init();
    net.setListeners(new ScoreIterationListener(10));
    log.debug("Total num of params: {}", net.numParams());

    // evaluation while training (the score should go down)
    for (int i = 0; i < nEpochs; i++) {
      net.fit(trainIter);
      log.info("Completed epoch {}", i);
      Evaluation eval = net.evaluate(testIter);
      log.info(eval.stats());
      trainIter.reset();
      testIter.reset();
    }

    ModelSerializer.writeModel(net, new File(basePath + "/minist-model.zip"),
true);
  }

}
```