МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**
Кафедра математической кибернетики и компьютерных наук

КЛАССИФИКАЦИЯ МУЗЫКАЛЬНЫХ ПРОИЗВЕДЕНИЙ ПО ЖАНРАМ С ПОМОЩЬЮ НЕЙРОННЫХ СЕТЕЙ

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студентки 4 курса 411 группы
направления 02.03.02 — Фундаментальная информатика и информационные технологии
факультета КНиИТ
Яшиной Татьяны Александровны

Научный руководитель
доцент, к. ф.-м. н. _____ Д. Ю. Петров

Заведующий кафедрой
к. ф.-м. н. _____ А. С. Иванов

Саратов 2020

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

В связи с быстрым развитием мультимедийных технологий, количество цифровых аудиозаписей, загруженных в Интернет, стремительно увеличивается. Поскольку доступность данных возрастает, необходимость классификации аудио файлов для их эффективного использования крайне важна.

Музыкальный информационный поиск (*Music Information Retrieval*, *MIR*) представляет собой одно из направлений исследования, позволяющих значительно улучшить взаимодействие с аудио данными. Классификация является основополагающим инструментом для анализа и обработки музыкальной информации. Одним из примеров применения классификации музыки является создание рекомендательных сервисов. Так, классификация музыкальных жанров, заключающаяся в присвоении определенного жанра (классический, рок, джаз и т. д.) неизвестному музыкальному произведению, является одной из основных задач MIR [1]. Поскольку экспертное аннотирование заведомо дорого в использовании, а также трудоемко для больших каталогов, возможность автоматической классификации востребована для сервисов потоковой передачи звука. Методы машинного обучения оказались весьма успешными в сфере анализа и обработки данных, извлечении тенденций и характерных особенностей.

Целью дипломной работы является применение методов машинного обучения в задаче музыкального анализа, а именно для классификации аудио файлов по различным жанрам.

В ходе работы были поставлены следующие задачи:

- Изучить особенности обучения нейронных сетей при работе со звуком;
- Реализовать, обучить и провести сравнительный анализ различных моделей и способов представления аудио файлов для решения задачи классификации (спектограммы, MFCC, т.д.);
- Реализовать веб-интерфейс для демонстрации работы нейронной сети, показавшей наиболее высокую точностью на тестовых данных.

1 Обзор нейронных сетей

1.1 Сверточные нейронные сети

Сверточные нейронные сети ($ConvNet$, или $CNNs$) получили широкое применение в области распознавания и классификации изображений [2]. Так, каждое входное изображение представляется матрицей размерности (h, w, d), где h и w обозначают высоту и ширину изображения в пикселях, а d — глубину (количество цветовых каналов). Роль CNN состоит в том, чтобы преобразовать изображения в форму, которую легче обрабатывать, сохранив при этом признаки (функции), имеющие решающее значение для получения хорошего прогноза [3].

1.1.1 Структура сверточной нейронной сети

Сверточный слой является основным строительным блоком *CNN* и выполняет большую часть вычислительной работы. Параметры *CONV* слоя состоят из набора обучаемых фильтров, каждый из которых имеет небольшую размерность и простирается на всю глубину. Свертка представляет собой математическую операцию, принимающую на вход два параметра, такие как матрица изображения и фильтр или ядро. Так, во время прямого хода мы сворачиваем ядро по ширине и высоте входной матрицы, вычисляя поточечные произведения в соответствующих позициях фильтра и входного изображения. В результате данной операции мы получаем двухмерную матрицу активации. Таким образом, свертка позволяет изучить особенности изображения, сохраняя при этом связь между пикселями. Скалярный результат каждой свертки поступает на вход функции активации, представляющей собой некую нелинейную функцию, определяющую выходное значение нейрона в зависимости от результата взвешенной суммы входов и порогового значения. Распространенными функциями активации являются логистическая (*sigmoid*), гиперболический тангенс, *ReLU*, *Leaky ReLU*, *Softmax*.

Подобно сверточному слою, *субдискритизирующий слой (Pooling)* отвечает за постепенное уменьшение размерности изображения, что в свою очередь позволяет сократить вычислительную мощность, необходимую для обработки данных. Так, поступившая на вход матрица делится на блоки, для каждого из которых вычисляется некоторая функция. Чаще всего используется функция максимума или среднего. Совокупность сверточного и субдискрити-

зирующего слоев образует i-й слой сверточной нейронной сети.

Полносвязные слои являются важными компонентами сверточных нейронных сетей. В данном слое каждый нейрон соединен со всеми нейронами на предыдущем уровне, причем каждая связь имеет свой весовой коэффициент. Добавление полносвязного слоя представляет собой способ изучения нелинейных комбинаций высокоуровневых характеристик, представленных выходными данными сверточного слоя.

Для решения проблемы переобучения, используются слои регуляризации (*Dropout*), основная идея которых состоит том, что каждый нейрон исключается с некоторой вероятностью p, в результате чего происходит изменение структуры сети.

1.2 Рекуррентные нейронные сети

Рекуррентные нейронные сети (RNN) представляют собой тип искусственных нейронных сетей, предназначенных для распознавания паттеров в последовательностях данных. Зачастую RNN применяются при решение таких задач, как распознавание рукописного текста, распознавание речи, языковое моделирование, перевод и т.д. Отличительной же особенностью RNN от других нейронных сетей является то, что они имеют временное измерение, а также обладают внутренней памятью, которая используется для обработки последовательностей произвольной длины [4], [5].

Каждое скрытое состояние содержит значения не только предыдущего скрытого состояния, но также и всех тех, которые предшествовали h_{t-1} до тех пор, пока сохраняется память. Однако, зачастую разрыв между соответствующей информацией и точкой, где она необходима, становится очень большим и по мере того, как этот разрыв увеличивается, RNN становятся неспособными научиться соединять информацию.

1.3 Сети с долгой кратковременной памятью

Сети с долгой кратковременной памятью, обычно называемые $LSTM$ (от англ. Long Short Term Memory), представляют собой особый тип рекурентных нейронных сетей, разработанных с целью предотвращения проблемы долгосрочной зависимости. В настоящее время данные сети получили широкое распространение, отлично проявив себя при решение большого разнообразия задач.

Все рекуррентные нейронные сети имеют форму цепочки повторяющихся модулей. В стандартных RNN этот повторяющийся модуль имеет довольно простую структуру, например, один слой с функцией активации $tanh$. $LSTM$ сети также имеют цепочечную структуру, однако структура самого повторяющегося модуля отличается.

Ключевой особенностью $LSTM$ является такой компонент, как состояние ячейки (cell state), напоминающий конвейерную ленту, проходящую по всей цепочке, принимая участие при этом лишь в небольшом количестве линейных преобразований. Информация может храниться, записываться или считываться из ячейки, как данные в памяти компьютера. Данный процесс тщательным образом регулируется специальными структурами, называемыми фильтрами (gate), которые состоят из слоя сигмоидальной нейронной сети и операции поточечного умножения, и определяют, пропустить информацию или нет.

Первым шагом в $LSTM$ является решение о том, какую информацию необходимо удалить из состояния ячейки, которое принимается сигмоидальным слоем, так называемым «слоем фильтра забывания» (f_t).

На следующем шаге определяется, какая новая информацию будет сохранена в состоянии ячейки. Данный процесс состоит из двух частей. Во-первых, сигмоидальный слой решает, какие значения необходимо обновить (i_t). Затем $tanh$-слой создает вектор новых значений-кандидатов $\hat{C}_t$, которые необходимо добавить в состояние.

Теперь, когда определены значения f_t, i_t и $\hat{C}_t$, необходимо обновить состояние ячейки C_{t-1}, полученное на предыдущем шаге, на новое C_t. Так, мы умножаем старое состояние на f_t, тем самым «забывая» то, что было решено забыть ранее, и добавляем $i_t * \hat{C}_t$ (новая информация).

На заключительном шаге в результате применения нескольких фильтров к состояниям ячейки определяется выходная информация. Так, сначала применяется сигмоидальный слой, решающий, какую именно информацию из состояния ячейки нужно выводить. Затем значения приводятся к диапазону [-1, 1], после чего перемножаются со значениями сигмоидального слоя, в результате чего на выходе мы получаем только необходимую информацию.

2 Обработка звуковых сигналов

Звук представляется в форме аудиосигнала, имеющего такие параметры, как частота, ширина пропускания, децибел и т. д. Типичный аудиосигнал может быть выражен как функция от амплитуды и времени.

Амплитуда волны в определенном временном интервале называется *семплом*. *Семплинг* же представляет собой преобразование непрерывного сигнала в серию дискретных значений. *Частотой дискретизации* называется количество семплов за определенный фиксированный промежуток времени. Так, высокая частота дискретизации приводит к меньшей потере информации, но к большим вычислительным затратам, в то время, как при низких частотах дискретизации происходит большее искажение информации.

Преобразование Фурье играет важную роль при обработке аудио сигналов, позволяя разложить функцию времени (сигнал) на составляющие частоты. На практике при работе с аудио сигналами наиболее часто применяется оконное преобразования Фурье, включающее в себя разбиение аудиосигнала на кадры с последующим вычислением преобразования Фурье для каждого кадра.

Наиболее известными форматами представления аудио сигналов при решении задач машинного обучения являются мел-спектограммы и мел-кепстральные коэффициенты. *Мел* представляет собой единицу измерения высоты тона, в основе которой лежит психо-физиологические особенности восприятия звука человеком. Мел-спектрограммой называется спектрограмма, в которой частота выражена не в Гц, а в мелах. Для перехода к мелам к исходной спектрограмме применяются мел-фильтры, представляющие собой треугольные функции, равномерно распределенные на мел-шкале.

3 Практическая часть

3.1 Постановка задачи

Классификация музыкальных жанров — одна из областей *музыкального информационного поиска* (MIR), популярность которой возрастает. Тем не менее, данная задача затруднена рядом факторов, среди которых отсутствие четкого и формального определения понятия *жанр*. Также границы между жанрами все еще остаются размытыми, что делает проблему распознавания музыкальных жанров (MGR) нетривиальной задачей.

Целью данной работы является изучение новых методов, тенденций в машинном обучении, применяемым к проблеме музыкальной аннотации, а также проведение эксперимента по классификации жанров музыки, позволяющего выполнить сравнительный анализ различных подходов, а также моделей машинного обучения в контексте поставленной задачи.

3.2 Описание технологий

Для решения поставленной задачи был выбран язык программирования Python [6], а также следующие технологии:

- $NumPy$ — библиотека с открытым исходным кодом для языка программирования Python, обладающая такими возможностями, как поддержка многомерных массивов (включая матрицы), а также поддержка высокоуровневых математических функций, предназначенных для работы с многомерными массивами;
- $Keras$ — открытая библиотека, написанная на языке Python, представляющая собой надстройку над фреймворками Deeplearning4j, TensorFlow и Theano, и нацеленная на оперативную работу с сетями глубинного обучения [7];
- $Librosa$ — это пакет Python для анализа музыки и аудио, предоставляющий строительные блоки, необходимые для создания музыкальных информационно-поисковых систем [8].

Для обучения моделей был использован $GoogleColab$ — облачный сервис на основе Jupyter Notebook.

3.3 Подготовка данных

В качестве обучающего набора данных был выбран датасет $GTZAN$ [9], содержащий одну тысячу музыкальных фрагментов продолжительностью по

тридцать секунд с частотой 22050 Гц, сгруппированных по десяти различным жанрам: *Блюз, Классика, Кантри, Диско, Хип-Хоп, Джаз, Метал, Популярная музыка, Регги, Рок*.

Для решения проблемы недостатка данных была применена аугментация, в результате которой каждая аудио запись была разбита на десять треков по три секунды каждый, что позволило увеличить количество обучающей выборки в десять раз, и как следствие, значительно повысить точность моделей.

Также на данном этапе было реализовано преобразование данных `.au` формата в формат, подходящий для машинного обучения. В качестве признаков было решено использовать мел-спектограммы и мел-кепстральные коэффициенты. Преобразование аудио файла непосредственно в мел-спектограмму, а также извлечение мел-кепстральных коэффициентов было реализовано средствами библиотеки $Librosa$. Мел-спектрограммы и диаграммы $MFCC$, построенные для различных жанров, имеют существенные различия, что позволяет использовать сверточные нейронные сети для классификации [9].

Наиболее важными параметрами, используемыми в преобразовании, являются — длина окна, которая указывает окно времени для выполнения преобразования Фурье, и переменная сдвига ($hoplength$), представляющая собой число значений (семплов) между последовательными фрагментами. В представленной работе для данных параметров были выбраны значения 2048 и 512 соответственно, количество мел-фильтров составило 128.

Поскольку решение данной задачи относится к типу задач контролируемого обучения, на этапе подготовки данных каждому вектору признаков были сопоставлены соответствующие метки, представляющие собой названия жанров. Затем получившаяся структура была записана на диск для последующего использования.

3.4 Описание моделей

Сверточные нейронные сети показывают отличные результаты в широком спектре задач, среди которых компьютерное зрение, распознавание речи и обработка естественного языка [10]. Рекуррентные нейронные сети, среди которых наиболее известны сети с долгой кратковременной памятью ($LSTM$), также широко распространены и успешно используются в ряде задач, для которых необходим захват долгосрочных зависимостей. В данной работе применяются различные варианты CNN, RNN моделей для прогнозирования

музыкального жанра, описание которых представлена далее.

3.4.1 Модель CNN-1D

Использование одномерных сверточных нейронных сетей при работе с аудио данными обосновано тем, что мы можем представлять звук в виде данных временного ряда. Модель, представленная в данной главе, состоит двух $1d\ Conv$ слоев с количеством фильтров 256, 128 соответственно; ширина ядра в обоих случаях равна трем. Между данными слоями выполняется $MaxPooling$ с фактором 2. После второго слоя свертки для создания репрезентативного вектора следует $GlobalMaxPooling$, который используется в полносвязном слое с 512 фильтрами. После каждого сверточного слоя следует слой $BatchNormalization$, позволяющий значительно ускорить процесс обучения [11]. Для стабилизации обучения и избежания проблемы переобучения применяются такие методы, как L_2-регуляризация (называемая также сокращением весов) с коэффициентом регуляризации λ, равным 0.001, а также $Dropout$ с параметром 0.5. Для всех моделей используется функция активации $ReLU$ на скрытом слое и $Softmax$ на выходном слое. Модель обучалась в течении пятидесяти эпох с использованием оптимизатора Адама с коэффициентом обучения 0.0001.

Точность данной модели на тестовых данных при классификации по десяти различным жанрам составила $\sim 90.5\%$.

3.4.2 Модель CNN-2D

Основным отличием данной модель от предыдущей является то, что здесь применяются $2D\ Conv$ слои. Так, была добавлена третья размерность — один цветовой канал, что позволило работать с мел-спектрограммами и с диаграммами MFCC, как с черно-белыми изображениями. Представленная модель состоит из трех $Conv2D$ слоев с количеством фильтров 128, 64, 64 соответственно. Для первого и второго слоя применяются фильтры размерностью $(3, 3)$, а для третьего — $(2, 2)$. Аналогично первой модели в качестве субдискретизирующего слоя здесь используется $MaxPooling2D$. Полносвязный слой содержит 128 фильтров. Также используется L_2-регуляция, $BatchNormalization$ и $Dropout$ с параметром 0.4.

Точность модели в этом случае составила $\sim 77.8\%$, что существенно уступает одномерной CNN модели.

3.4.3 LSTM модель

Модель, представленная в данной главе, содержит два $LSTM$ слоя, содержащих 128, 64 единиц памяти. После второго $LSTM$-слоя следует полносвязный $Dense$ слой с 64 нейронами, применяется функция активации $ReLU$, а на выходном слое используется функция активации $Softmax$, число нейронов равно десяти по количеству классов.

Точность модели после 50 эпох обучения составила $\sim 72\%$.

3.4.4 CRNN модель

В предыдущих секциях были рассмотрены CNN, RNN модели. Так, использование сверточной нейронной сети при решении задачи классификации музыкальных жанров, обосновано тем, что спектрограммы, представляющие собой визуальное представление звука по частоте и времени, подобны изображениям, у каждого из которых есть свои отличительные признаки. RNN же в свою очередь лучше понимают последовательные данные, поскольку скрытое состояние в момент времени t зависит от скрытого состояния, полученного на предыдущем шаге.

Сверточная рекуррентная модель, представленная в данной главе, включает в себя $Conv1D$ слои, выполняющие операции свертки только по оси времени. Каждый слой $1d$ свертки извлекает признаки из небольшого фрагмента спектограммы. После операции свертки применяется $BatchNormalization$ и функция активации $ReLU$. Затем выполняется $MaxPooling1D$, уменьшающий пространственные размеры изображения. Данная цепочка операций — $Conv1D - BatchNormalization - ReLU - MaxPooling1D$ выполняется три раза. Затем, вывод одномерного сверточного слоя подается на вход $LSTM$, которая использует 128 скрытых нейронов. Далее следует полносвязный слой с 64 нейронами. И, наконец, итоговый выходной слой модели представляет собой полносвязный слой с функцией активации $Softmax$ и 10 нейронами для присвоения вероятности 10 классам. Также для предотвращения проблемы переобучения между всеми слоями используется L_2-регуляризация.

Точность модели составила $\sim 88\%$ на тестовых данных, что превосходит как CNN-$2D$ модель, так и $LSTM$ модель, однако проигрывает CNN-$1D$ модели с одномерной сверткой.

Таблица 1 – Сравнение точности классификации (%) для набора данных GTZAN по рассмотренным методам (лучший результат выделен жирным шрифтом).

Название модели	мел-спектограммы	MFCC
CNN-1D	78.9%	**90.5%**
CNN-2D	73%	77.8%
LSTM	47.62	72.2%
CRNN	78.1%	87.9%
Parallel CNN RNN	64.6%	83%

3.4.5 Параллельная CNN-RNN модель

Данная модель имеет параллельную структуру, состоящую из сверточной и рекуррентной нейронных сетей. Так, сверточная сеть состоит из цепочки операций $-Conv1D-BatchNormalization-ReLU-MaxPooling1D$, используемой также в $CRNN$, которая повторяется дважды, а рекуррентная нейронная сеть представлена GRU. Ключевая идея подхода заключается в том, что, хотя сверточная нейронная сеть и содержит RNN слой, он может извлекать временную информацию только из выходных данных CNN, однако временные отношения оригинальных музыкальных сигналов не сохраняются при выполнении операций со сверточными нейронными сетями.

Модель, описанная в этой секции, передает спектограмму (диаграмму $MFCC$) через слои CNN и RNN параллельно, объединяя данные, получившиеся на выходе, и затем отправляя их через полносвязный слой с функцией активацией $Softmax$.

Валидационная точность представленной модели на тестовых данных составила $\sim 83\%$.

3.5 Результаты эксперимента

В ходе эксперимента были спроектированы и обучены различные виды моделей нейронных сетей, предназначенные для решения задачи классификации музыкальных жанров. Результаты эксперимента представлены в таблице 1, где строки соответствуют конкретным моделям, определенным в предыдущей части, столбцы — методам обработки данных (в данном случае мел-спектограммы, мел-кепстральные коэффициенты). На пересечении располагаются значения точности классификации (%) для набора данных $GTZAN$.

Так, было получено, что применение мел-кепстральных коэффициентов показывает себя намного лучше по сравнению с мел-спектограммами. Также было выявлено, что точность классификации достигает более высоких значений при использование менее глубоких нейронных сетей (одномерные сверточные сети в данном эксперименте).

3.6 Описание веб-приложения

Для демонстрации работы нейронной сети, показавшей наилучшие результаты при решении задачи классификации музыкальных жанров, было реализовано веб-приложение.

Для написания клиентской части были использованы $HTML$, CSS, $JQuery$, $Bootstrap$, $JavaScript$; для серверной части был выбран $Flask$ — легковесный фреймворк для создания веб-приложений на языке программирования $Python$. В качестве среды программирования был выбран `IDE PyCharm`.

При переходе на главную страницу приложения, сервер возвращает шаблон `templates/index.html`. На странице загрузки пользователю предоставляется возможность выбрать аудио файл, для которого требуется определить жанр. Нажатие на изображение аудио файла сопровождается открытием файлового диалога. После загрузки песни необходимо нажать на кнопку «Предсказать жанр», в результате чего загруженный аудио файл будет отправлен на сервер `POST` запросом по адресу `/predict`. В случае успешного ответа сервера проигрывание загруженной музыкальной композиции приостановится, а на экран будет выведена диаграмма жанров, а также предсказанный результат.

Взаимодействие работы приложения с нейронной сетью на сервере осуществляется через сервис `_Genre_Prediction_Service`. Так, получив `POST` запрос, сервер инициирует функцию `predict` данного сервиса и возвращает клиенту полученный ответ в формате json.

Непосредственная логика по работе с моделью реализована в классе `_Genre_Prediction_Service`. С помощью функции `extract_feature` мы получаем сигнал, который затем разбивается на сегменты, для каждого из которых извлекается набор MFCC. Затем по суммарным значениям вероятностей принадлежности каждого сегмента тому или иному жанру определяется финальный рейтинг для всей композиции.

3.7 Тестирование модели

Для определения жанра, на котором модель работает наиболее точно, было проведено тестирование модели с помощью 100 музыкальных произведений из онлайн библиотеки *Jamendo Licensing*, где для каждого жанра было выбрано по десять композиций. Так, нейронная сеть предсказала верно все аудио файлы для следующих жанров: *Классика*, *Блюз*, *Джаз*, *Диско*. Наименьшая доля верных результатов была получена для музыкального жанра *Рок* (50%), который нейронная сеть зачастую определяла к категории *Метал*, представляющей собой разновидность рок-музыки, что подтверждает нетривиальность данной задачи ввиду смешения стилей.

ЗАКЛЮЧЕНИЕ

В ходе дипломной работы для решения задачи классификации аудио файлов по десяти различным жанрам: *Блюз, Классика, Кантри, Диско, Хип-Хоп, Джаз, Метал, Популярная музыка, Регги, Рок* были спроектированы следующие типы нейронных сетей: *CNN-1D, CNN-2D, LSTM, CRNN, Parallel CNN-RNN*. Сложность данной темы обусловлена тем, что зачастую музыкальным произведениям присущи черты сразу нескольких жанров, более того не существует достаточно четкого определения данного понятия.

Обучение моделей проводилось с помощью мел-спектограмм, а также мел-кепстральных коэффициентов, которые наиболее часто применяются при решении подобных задач. Длительность обучения каждой модели составила пятьдесят эпох. В ходе эксперимента было получено, что точность моделей при использовании MFCC значительно превышает результаты, полученные с применением мел-спектограмм. Более того, сравнительный анализ показал, что обучение менее глубоких сетей в контексте данной задачи при небольшом объеме обучающей выборки (1000 файлов по 30 секунд) проходит более эффективно. Так, наиболее высокий результат был показан моделью *CNN-1D*, состоящей из двух сверточных $Conv1D$ слоев, а также двух полносвязных слоев, и составил $\sim 90\%$ на тестовой выборке. Для демонстрации работы данной модели был реализован веб-интерфейс, для серверной части которого был использован фреймворк Flask.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1 *M. Schedl, E. G.* Music information retrieval: Recent developments and applications. / E. G. M. Schedl, J. Urbano. — 2014. — Pp. 127–261. — URL: http://www.deeplearningbook.org. (Дата обращения 01.05.2020).

2 *Goodfellow Ian, B. Y.* Deep learning. / B. Y. Goodfellow, Ian, A. Courville. — 2016. — URL: http://www.deeplearningbook.org. (Дата обращения 01.05.2020).

3 *Shujian Yu Student Member, I. K. W. R. J. M. I.* Understanding convolutional neural networks with information theory: An initial exploration / I. K. W. R. J. M. I. Shujian Yu, Student Member, L. F. Jose C. Principe // *arXiv:1804.06537v5 [cs.LG]*. — 2020. — URL: https://arxiv.org/pdf/1207.0580.pdf. (Дата обращения 18.04.2020).

4 *Сергей Николенко А. Кадурин, Е. А.* Глубокое обучение. Погружение в мир нейронных сетей / Е. А. Сергей Николенко, А. Кадурин. — Питер СПб, 2018.

5 *Саймон, Х.* Нейронные сети. Полный курс / Х. Саймон. — Вильямс, 2019.

6 Python Documentation [Электронный ресурс]. — URL: https://docs.python.org/3/ (Дата обращения 01.05.2020).

7 Keras Documentation [Электронный ресурс]. — URL: https://keras.io/documentation/ (Дата обращения 01.05.2020).

8 LibrosaDocumentation [Электронный ресурс]. — URL: https://librosa.github.io/librosa/ (Дата обращения 01.05.2020).

9 *Tzanetakis, G.* Musical genre classification of audio signals / G. Tzanetakis, P. Cook // *IEEE Transactions on speech and audio processing*. — 2002. — Vol. 10.

10 *Kim, Y.* Convolutional neural networks for sentence classification / Y. Kim // *IEEE Transactions on speech and audio processing*. — 2014.

11 *Ioffe, S.* Batch normalization: Accelerating deep network training by reducing internal covariate shift. / S. Ioffe, C. Szegedy. — 2015.