Министерство образования и науки Российской Федерации
Санкт-Петербургский политехнический университет Петра Великого
Высшая школа программной инженерии

Работа допущена к защите
Директор ВШ ПИ
_____П.Д. Дробинцев
"___"_____ 2018г.

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

Разработка и реализация алгоритма идентификации жанра музыкального произведения на основе аудиозаписи

По направлению *02.03.02 «Фундаментальная информатика и информационные технологии»*
по образовательной программе
02.03.02_ 02 «Информатика и компьютерные науки»

Выполнил
студент гр. 43504/6 О. А. Сашко
Руководитель
д.т.н., проф. Ю.Б. Сениченков

Санкт-Петербург
2018

САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО
Институт компьютерных наук и технологий

Утверждаю
Директор ВШ ПИ
_____П.Д. Дробинцев
"___"_____ 2018г.

ЗАДАНИЕ

по выполнению выпускной квалификационной работы
студенту О.А. Сашко гр. 43504/6

1. Тема: *Разработка и реализация алгоритма идентификации жанра музыкального произведения на основе аудиозаписи*

2. Срок сдачи работы 08.06.18.

3. Исходные данные к проекту (работе).

   В ходе выполнения работы необходимо разработать и реализовать адаптивный алгоритм классификации, получающий в качестве исходных данных цифровую аудиозапись музыкального произведения в формате MP3 и выдающий в качестве результата идентификатор, соответствующий жанру этого произведения. Необходимо обеспечить возможность настройки классификатора на различные множества жанров методом машинного обучения с использованием размеченного набора примеров. Основным критерием качества работы алгоритма является точность классификации (доля правильно классифицированных примеров).

4. Содержание расчетно-пояснительной записки (перечень подлежащих разработке вопросов).

   - Введение, описание задачи определения жанра музыкального произведения.
   - Обзор алгоритмов распознавания жанра и достигаемых ими показателей точности.
   - Постановка задачи распознавания жанра танцевальной музыки.

- Описание разработанного алгоритма распознавания жанра музыкального произведения.
- Оценка точности, обеспечиваемой разработанным алгоритмом на наборе данных «Ballroom».
- Заключение: краткое изложение полученных результатов и выводы.

5. Перечень графического материала с точным указанием обязательных чертежей.

   Обязательный графический материал не предусмотрен.

6. Консультанты по проекту (с указанием относящегося к ним разделов проекта, работы).

   Тимофеев Д. А., ведущий программист лаборатории «Системы управления мобильными устройствами» ИКНТ: разделы 3-6.

Дата выдачи задания: _____ г.
Руководитель ВКР: _____д.т.н., проф.Ю.Б. Сениченков
Задание принял к исполнению _____ г.
Студент _____О. А. Сашко

Реферат

На 39 с., , рис. 17 , табл. 1

Музыкальный жанр — это некая описательная категория, которая чаще всего применяется для характеристики музыки, и несущая в себе значимую информацию для поиска музыкальной записи. В данной работе представлена реализация алгоритма идентификации жанра музыкального произведения с использованием модифицированного подхода классификации KNN для набора данных «Ballroom». В качестве векторов признаков использовались коэффициенты MFCC, их производные первого и второго порядка и темп, вычисленный с помощью Librosa, одной из библиотек Python. Предлагаемый подход обеспечивает точность классификации 92,1%, что превосходит результаты, представленные в обзоре.

Ключевые слова: машинное обучение, классификация жанра, алгоритм K ближайших соседей, мел-частотные кепстральные коэффициенты (MFCC).

Abstract

39 pages , 17 figures , 1 tables

A musical genre is some category, which is most often used to define music, and carries important information for searching for a musical record. we present an algorithm for identifying the genre of a musical work using the modified KNN classification approach for the «Ballroom» data set. The MFCC coefficients, their first and second-order derivatives, and the tempo which is computed with Librosa, one of the Python libraries, were used as the feature vectors. The proposed approach ensures the accuracy of the classification of 92.1%, which exceeds the results, the level in the survey.

Keywords: machine learning, genre classification, nearest neighbor algorithm, mel-frequency cepstral coefficients (MFCC).

Оглавление

Список обозначений

BPM	Beats Per Minute (биты в минуту)
1NN	Nearest Neighbor (ближайший сосед)
SVM	Support Vector Machine (метод опорных векторов)
STFT	short-time Fourier transform (кратковременное преобразование Фурье)
WT	Wavelet transform (вейвлет-преобразование)
MFCC	Mel-frequency cepstral coefficients (мел-частотные кепстральные коэффициенты)
MP3	MPEG LayerIII
GMM	Gaussian Mixture Modeling (модель гауссовских смесей)
TreeQ	Tree-based Vector Quantization
AdaBoost	Adaptive boosting algorithm
DWCH	Daubechies Wavelet Coefficient Histograms (гистограммы вейвлет-коэффициентов Добеши)
KNN	k Nearest Neighbor (алгоритм k ближайших соседей)
wKNN	weighted k Nearest Neighbor (взвешенный алгоритм k ближайших соседей)
IGS	Inter-genre similarity modeling (моделирование межжанрового сходства)
IIGS	Iterative Inter-genre similarity modeling
DFT	Discrete Fourier Transform (дискретное преобразование Фурье)
ACF	Auto-Correlation Function (функция автокорреляции)

Введение

Музыка играет важную роль в повседневной жизни многих людей. В связи с широким распространением цифровой музыки, формируются большие коллекции музыкальных данных. По мере увеличения объема информации, связанной с музыкой, необходимо эффективное средство для ее поиска и организации. Появились задачи: визуализация данных, интеллектуальная интеграция, кластеризация, поиск подобия и классификация, которые можно решать методами интеллектуального анализа данных.Поиск музыкальной информации должен соответствовать вкусам и потребностям каждого слушателя. Существует несколько способов определения этих потребностей. В [11] Д. Гурон обращает внимание на тот факт, что поскольку выдающиеся функции музыки являются социально-психологическими, то наиболее полезная характеристика будет основана на четырех типах информации: жанре, эмоции, стиле и сходстве.

Данная работа посвящена автоматическому распознаванию музыкальных жанров. Это популярный метод структурирования и организации большого количества музыкальных файлов, доступных в Интернете, так как даже в официально приобретенных музыкальных композициях издатели нередко забывают указать информацию о каждой композиции в метаданных, которые были бы полезны для поисковых систем интернет-музыки, музыковедов и слушателей, чтобы найти музыку из множества опций.

Работа организована следующим образом. В главе 1 приводится обзор существующих алгоритмов, используемых в задаче классификации жанров. В конце первой главы сформулирована точная постановка задачи и цели данной работы. В главе 2 приведено описание набора признаков, извлеченных из аудиозаписи, а также описание алгоритма

классификации k ближайших соседей. В главе 3 рассмотрены вопросы практической реализации алгоритма, и приведены численные результаты, полученные с использованием предлагаемого подхода.

Глава 1

Задача определения жанра

1.1 Постановка задачи определения жанра

Жанровую классификацию музыки традиционно производили вручную, основываясь на манере исполнения, ритмической структуре, наборе музыкальных инструментов и составе исполнителей, а также учитывая сложившиеся представления о жанре и субъективное восприятие. Тем не менее, в работе Р.Гьердингена и Д. Перротта [7] сообщалось, что студенты колледжа достигли не более 70% точности классификации при прослушивании трех секунд аудиозаписи, тем самым показывая, что техника автоматического определения жанра является значительным добавлением к системам контекстного аудио-поиска.

Отсутствие согласия в выборе жанра является типичной проблемой классификации музыки, поскольку релевантность различных категорий крайне субъективна. С одной стороны, мы можем сравнивать жанры, имеющие достаточно четкие границы, к примеру классическую музыку и хип-хоп (обзор работ, решающих данную задачу, приведен в разделе 1.2). С другой, существуют жанры, обладающие очень схожими наборами признаков, что делает их трудно различимыми.

Следовательно, необходимо определить набор жанров, по которым будет производиться классификация. В данной работе под жанром понимаются такие виды танцевальной музыки, как: джайв, квикстеп,

Таблица 1.1. Поджанры бальных танцев

Жанр	Кол-во экземпляров
джайв	60
квикстеп	82
танго	86
вальс	111
венский вальс	65
ча-ча-ча	111
самба	86
румба	97

танго, вальс, венский вальс, ча-ча-ча, самба и румба. Таким образом, задача идентификации жанра сводится к задаче многоклассовой классификации с определением принадлежности исследуемого объекта к одному классу.

Чтобы проверить корректность своего алгоритма, необходимо использовать стандартный набор данных, для которого известно достаточное количество результатов. Для этих целей был выбран набор данных «Ballroom» [3], который содержит отрывки из 698 произведений музыки, около 30 секунд. Данные охватывают восемь музыкальных поджанров бальных танцев, перечисленных в таблице 1.1.

Кроме того, также доступен эталонный темп (в битах в минуту, BPM) каждой записи. Диапазон темпа: от 60 до 224 BPM (рис.1.1) [8]. В таблице 1.2 представлены результаты работ, производящих классификацию жанра на наборе данных, описанном выше.

Ф.Гуйон и др.[6] оценивают актуальность набора из 73 признаков, характеризующих ритм, измерив их успешность в экспериментах по жанровой классификации. К ним относятся функции, полученные из темпа и из гистограммы периодичности. Авторы применяют алгоритм ближайшего соседа (1NN) и сообщают о распознавании с точностью 90,1% с использованием эталонного темпа и 78,9% с использованием расчетного темпа.

Г.Питерс [16] сравнивает различные спектральные и временные представления периодичности для описания ритма музыкального произведения. Он сначала извлекает onset функцию из сигнала, затем вычисляет три вектора признаков на основе амплитуды дискретного преобразования Фурье (DFT), функции автокорреляции (ACF) и

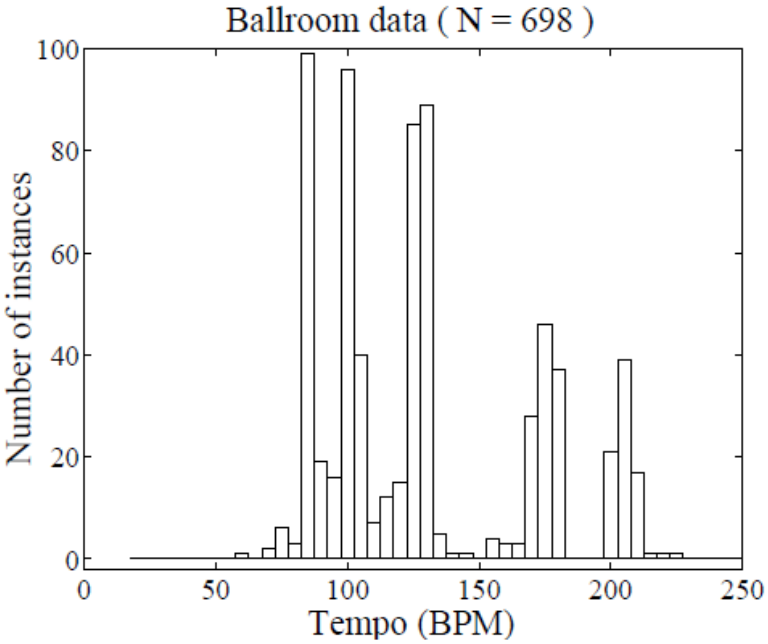

Рис. 1.1. Диапазон темпа в наборе данных «Ballroom»

Таблица 1.2. Работы, использующие набор данных «Ballroom»

Работа	Эталонный темп		Расчетный темп	
Ф.Гуйон и др. [6]	90,1%	1NN	78,9%	1NN
Г.Питерс [16]	96,13%		87,96%	AdaBoost
А.Хольцапфель [10]	-		85,5%	wKNN (k=7)
			84,5%	KNN (k=3)
Т.Поле [15]	-		90,2%	1NN
			89,2%	KNN (k=5)

представления гибридной оси-автокорреляции. При оценке эталонного темпа классификатором SVM наилучший результат (96,13%) получается с использованием 40-мерного вектор-функции, состоящей из дискретизированных значений DFT, темпа и небольшого набора характеристик ритма. Питерс повторно применил свой метод для расчетного темпа, наилучший результат (87,96%) получается с использованием 28-мерного вектор-функции, состоящей из дискретизированных значений DFT и ACF, темпа и небольшого набора характеристик ритма.

А.Хольцапфель и Я.Стилиану [10] представляют новый способ измерения ритмического сходства между двумя аудиозаписями с использованием спектров периодичности. Чтобы обнаружить сходство для кусков разных темпов, линейность траектории деформации между их спектрами служит мерой их ритмического сходства. Используя классификацию wKNN для двух наборов данных, предлагаемая мера обеспечивает сопоставимую точность классификации по жанру 82,1% до лучших широко используемых мер 85,5% для первого набора данных. Для второго набора данных, который характеризуется большой дисперсией темпов, предлагаемая мера превосходит все контрольные показатели, достигая точности 69,0%, тогда как лучшие из других показателей достигают 53,8%.

1.2 Обзор существующих решений распознавания жанра

В данном разделе приведен обзор существующих методов, используемых в задаче классификации жанров, имеющие достаточно четкие границы различия.

Д.Цанетакис, П.Кук и Г.Эссль были одними из первых, кто представил проблему автоматической классификации музыки [17]. Они утверждают, что, хотя разделение музыки на жанры субъективно, существуют критерии, связанные с текстурой, инструментовкой и ритмической структурой музыки, которые могут быть использованы для характеристики музыки. Статистика спектрального распределения по времени используется для представления музыкальной поверхности — характеристик музыки, связанных с текстурой, ром и инструментовкой. Они включают в себя функции, такие как среднее значе-

ние спектрального центроида, среднее значение спектрального спада, среднее значение спектрального потока, среднее значение нулевых пересечений, стандартное отклонение спектрального центроида, стандартное отклонение спектрального спада, стандартное отклонение спектрального потока, стандартное отклонение нулевых пересечений и низкой энергии. Эти характеристики вычисляются в окне «текстуры» продолжительностью 1 секунда, состоящем из 40 кадров, с использованием кратковременного преобразования Фурье (STFT). Расчеты признаков для представления ритмической структуры музыки основаны на вейвлет-преобразовании (WT). Набор ритмических признаков основан на выявлении наиболее заметных периодичностей сигнала. Используя дискретное вейвлет-преобразование, сигнал сначала разлагается на несколько октавных частотных диапазонов, а амплитудная огибающая временной области каждого диапазона извлекается отдельно. После этого огибающие каждого канала суммируются и вычисляется функция автокорреляции. Пики автокорреляционной функции соответствуют различным периодичностям огибающей сигнала. Производительность этих наборов признаков была оценена путем обучения гауссовских классификаторов, с использованием реальных аудио коллекций. Набор данных, используемый в этой работе, включал в себя 6 жанров: классическую музыку, кантри, диско, хип-хоп, джаз и рок.

Исходя из работы, описанной выше, автоматическая классификация основана на трех наборах признаков, связанных с ритмическими, тональными и тембральными характеристиками. В литературе тембр определяется как характеристика звука, позволяющая двум звукам. с одинаковой высотой и громкостью звучать по-разному[1].

Так Д.Пай в своей работе по классификации аудиосигналов, сжатых в формате MP3 [4], использует мел-частотные кепстральные коэффициенты (MFCC), одни из самых широко используемых тембральных характеристик. Автор сравнивает два подхода в отношении производительности и стоимости вычислений. В первом случае MFCC используются как функции, для которых требуется предыдущая декомпрессия MP3. Другой предложенный метод состоит в получении подобного MFCC набора функций, выполняющих только частичную декомпрессию, и называются MP3CEP. В [4] изучаются и оцениваются два метода: GMM и TreeQ. Классификация музыки осуществляется в следующих 6 классах: блюз, легкая музыка, классиче-

ская, опера, танец (техно) и инди-рок. Лучший уровень классификации и для MFCC и для MP3CEP был получен, используя классификатор GMM. В то время как MFCCs выполнял немного лучше (92%), чем MP3CEP (90.9%) в последнем случае, вычисление было более, чем в пять раз быстрее.

У.Багчи и Э.Эрзин [5] исследуют использование динамических тембральных текстурных признаков. Термин «тембральная текстура» применяется для обозначения более общего качества смеси звуков, присутствующих в музыке, которая не зависит от ритма и гармонии. Так как это понятие характеризует кратковременные спектральные свойства (обычно около 20-40 мс), для получения признаков из аудиозаписи в течение более длительного периода времени, в работе [5] определяется «текстурное окно» размером 25 мс для каждого кадра в 10 мс. Таким образом, из окна извлекаются 13-мерный вектор MFCC, 4-мерный вектор спектральной формы со спектральным центроидом, спектральным спадом, спектральным потоком и частотой нулевых пересечений аудиосигнала. Для измерения изменения краткосрочных спектров с течением времени итоговый вектор признаков расширен производными первого и второго порядка. Авторы предлагают два новых классификатора, использующих моделирование межжанрового сходства (IGS) для захвата сходного спектрального содержания среди различных жанровых типов. Как только кластеры IGS статистически смоделированы, кадры IGS могут быть захвачены и удалены из процесса принятия решения, чтобы уменьшить межжанровую путаницу. Для оценки предложенных алгоритмов классификации авторы используют набор музыкальных жанров GTZAN [9], который включает в себя десять различных жанров: блюз, классика, кантри, диско, хип-хоп, джаз, металл, поп, регги и рок. IGS и IIGS достигли 88,60% и 92,40% правильных показателей классификации, соответственно.

В качестве признаков, характеризующих музыкальное содержание для автоматической классификации чистой и вокальной музыки, в статье Д. Бергстра и др. [2] исследованы коэффициенты Быстрого преобразования Фурье, реальные кепстральные коэффициенты, мелчастотные кепстральные коэффициенты, частота переходов аудиосигнала через ноль, спектральный разброс, спектральный центроид, спектральный спад и коэффициенты линейного предсказания. Авторы используют алгоритм AdaBoost. Основная идея алгоритма заключается в построении «сильного» классификатора за счет объединения

«слабых» классификаторов. Для этого слабый алгоритм вызывается несколько раз, заданных пользователем с различными подмножествами обучающих данных. В качестве слабых классификаторов применялись многоклассовые деревья Хэмминга. Для оценки представленных материалов два набора данных: «Magnatune» [13] и «USPOP» [18]. База данных «Magnatune» имеет иерархическую жанровую таксономию с 10 классами: эмбиэнт, блюз, классический, электронный, этнический, народный, джаз, нью-эйдж, панк, рок, тогда как база данных «USPOP» имеет 6 жанров: кантри, электронная и танцевальная, нью-эйдж, рэп и хип-хоп, регги, рок. Предложенный метод на наборе данных «Magnatune» получил точность 77,75%, в случае USPOP точность равна 86,92%.

К. Вест и С. Кокс [19] изучают несколько факторов, влияющих на автоматическую классификацию музыкальных звуковых сигналов. Они описывают и оценивают эффективность классификации двух различных мер спектральной формы, используемых для параметризации звуковых сигналов, мел-частотных фильтров, используемых для получения мел-частотных коэффициентов (MFCC) и спектрального контраста. Завершающим этапом расчета функции классификации является снижение ковариации между различными компонентами вектора признаков. Для MFCC это выполняется дискретным Косинусным преобразованием, а для спектрального контраста — преобразованием Кархунена-Лоева. Затем музыкальные звуковые сигналы разделяются на шесть жанров: рок, классика, хэви-метал, барабан и бас, регги и джангл. Оцененными классификаторами являются одиночные Гауссовские модели, 3-х компонентные гауссовские модели смеси, критерий Фишера линейный дискриминантный анализ и новые классификаторы, основанные на неконтролируемой конструкции бинарного классификатора дерева решений либо с линейным дискриминантным анализом, либо с парой одиночных Гауссиан с измерениями расстояния Махаланобиса, используемыми для разделения каждого узла в дереве. Бесконтрольное построение больших деревьев решений для классификации кадров из музыкальных звуковых сигналов — это новый подход. Это позволяет классификатору изучать и идентифицировать различные группы звуков, которые встречаются только в определенных типах музыки. Результаты, полученные этими классификаторами, значительно увеличивают точность классификации музыкальных аудиосигналов.

Д.Ли, М.Огихара [12] используют тот же набор функций, что и Цанетакис, Кук и Эссль [17], но, кроме того, они предлагают новый метод извлечения признаков, гистограммы вейвлет-коэффициентов Добеши (DWCH). Авторы применяют программное обеспечение Marsyas [14] для извлечения функций. Эффективность этой функции оценивается с помощью алгоритмов машинного обучения, таких как метод опорных векторов, K ближайших соседей, модели Гауссовой смеси. Показано, что DWCHs значительно повышают точность классификации музыкальных жанров. На данных, предоставляемых в [17], точность классификации была увеличена с 65% до почти 80%.

1.3 Уточненная постановка задачи

Целью данной работы является разработка и реализация алгоритма идентификации жанра музыкального произведения на основе аудиозаписи. Для достижения поставленной цели должны быть решены следующие задачи:

1. Реализовать алгоритм извлечения признаков, который сопоставляет каждой аудиозаписи вектор признаков, описывающий данную запись.

2. Выбрать и реализовать алгоритм классификации векторов признаков, который обеспечивает точность, превосходящую наилучший опубликованный результат 90,2%.

3. Провести оценку точности, достигаемую разработанным алгоритмом на наборе данных «Ballroom» и сравнить с опубликованными результатами.

Глава 2

Алгоритмы извлечения признаков и классификации жанра

Как уже отмечалось в главе 1, извлечение признаков и использование данных признаков в алгоритмах машинного обучения для классификации — две важные проблемы автоматической классификации музыкальных жанров.

В этой главе приведены описание используемых в данной работе функций извлечения признаков из аудиозаписи и описание алгоритма классификации K ближайших соседей.

2.1 Извлечение векторов признаков из аудиозаписи

Данный набор признаков выбирался, исходя из популярности соответствующих признаков в обзоре литературы главы 1.

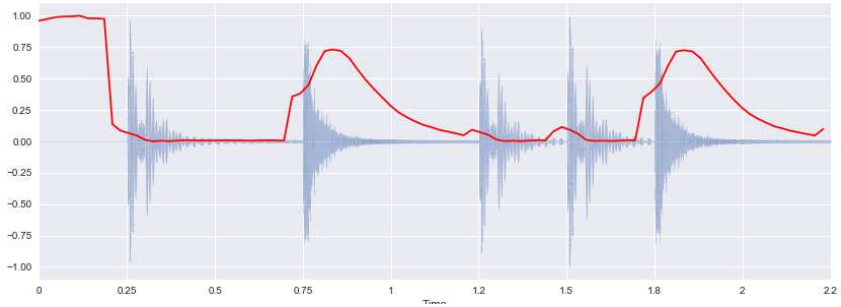

Рис. 2.1. Спектральный центроид

2.1.1 Спектральные признаки

Спектральный центроид (Spectral Centroid) определяется как средняя частота, взвешенная по величине спектра:

$$C_n = \frac{\sum_{k=0}^{N-1} X[k]_n k}{\sum_{k=0}^{N-1} X[k]} \tag{2.1}$$

где n — номер кадра, подлежащий анализу, k — номер ячейки частоты, $|X[k]|_n$ — значение спектра в кадре n на частоте k.

Рассмотрим спектральный центроид на графике (рис.2.1). Подобно частоте переходов аудиосигнала через ноль, в начале сигнала возникает ложное повышение спектрального центра тяжести. Это связано с тем, что тишина в начале имеет такую малую амплитуду, что высокочастотные компоненты имеют шанс доминировать. Чтобы обойти это, необходимо добавить небольшую константу перед вычислением центроида, тем самым перемещая его к нулю на спокойных участках (рис. 2.2).

Спектральный спад (Spectral RollOff) — это частота R_n, ниже которой остается заданный процент общей энергии спектра. По умолчанию используется значение 85%. Математически он определяется как:

$$\sum_{n=0}^{R_n-1} X[k]_n = 0,85 * \sum_{k=0}^{N-1} X[k]_n \tag{2.2}$$

где n — номер кадра, подлежащий анализу, k — номер ячейки частоты,

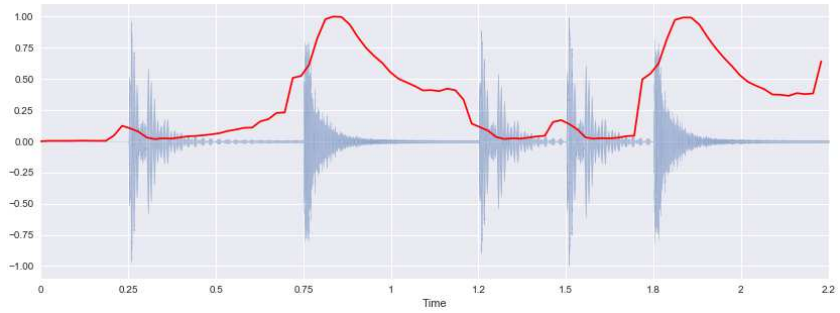

Рис. 2.2. Спектральный центроид+константа

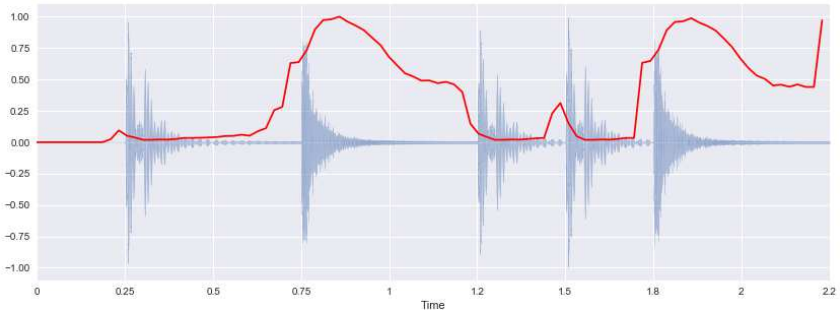

Рис. 2.3. Спектральный спад

$|X[k]|_n$ — значение спектра в кадре n на частоте k.

Спектральный спад часто используется для того, чтобы отделить шум от значимого содержимого (шумовая часть находится выше частоты спада).

Частота переходов аудиосигнала через ноль (Zero Crossing Rate) — это количество пересечений оси времени аудио сигналом.

$$ZeroCross = \frac{1}{2}\sum_{n=1}^{N} |sign(x(n)) - sign(x(n-1))| \tag{2.3}$$

Нулевые пересечения полезны для определения количества шума в сигнале.

Спектральный поток (Spectral Flux) кадра является суммой квадратов расстояний между нормированными величинами последовательных частотных бинов. Это мера показывает насколько быстро спектр изменяется по частоте.

$$F_r = \sum_{k=1}^{\frac{N}{2}} \left(|X_r[k]| - |X_{r-1}[k]|\right)^2 \tag{2.4}$$

2.1.2 Мел-частотные кепстральные коэффициенты (MFCC)

Мел-частотные кепстральные коэффициенты (MFCC) являются наиболее распространенным представлением спектра звука, которое часто используется в системах автоматического распознавания речи, а также в задаче классификации музыкальных произведений по жанрам.

Этот метод является более чувствительным к звукам низкой частоты, и менее чувствительным к звукам в области высоких частот, поэтому становится очевидным недостаток представления звука в виде частотной спектрограммы, в связи с этим используется частотная шкала мел.

Основные этапы расчета мел-частотных кеспральных коэффициентов являются следующими (рис. 2.4):

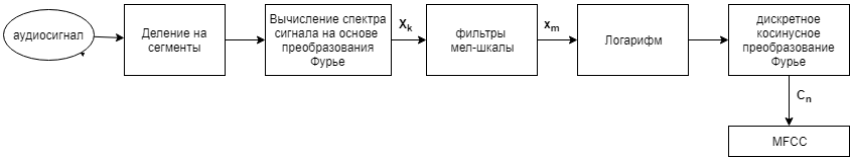

Рис. 2.4. Этапы вычисления коэффициентов MFCC

1. *Деление исходного сигнала на сегменты.*

   Известно, что аудио сигнал непрерывно меняется во времени, в целях упрощения работы допустим, что сигнал неизменен на некотором небольшом временном интервале, для этого необходимо «поделить» этот сигнал на сегменты («фреймы») некоторой

длины, спектры которых остаются относительно неизменными в течение выбранного периода времени.

2. *Вычисление спектра сигнала для каждого фрейма на основе преобразования Фурье.*

   После разбиения сигнала на фреймы, к каждому отрезку применяется весовая функция, а затем преобразование Фурье (2.6). В качестве весовой функции обычно используется окно Хэмминга (2.5) :

$$w_n = 0,54 - 0,46 * \cos\left(\pi * \frac{n}{N-1}\right), n = 0, \ldots, N-1 \qquad (2.5)$$

   где N — длинна окна, выраженная в отсчетах.

$$x_k = \sum_{0}^{N-1} x[n] * \exp\left(-\frac{2\pi i}{N} kn\right) \qquad (2.6)$$

   Если подставить формулу 2.5 в 2.6, то получим:

$$X_k = \sum_{0}^{N-1} w_n x[n] * \exp\left(-\frac{2\pi i}{N} kn\right), \qquad (2.7)$$

   где $x[n]$ — отсчёт сигнала во временной области, N — количество отсчетов в одном сегменте, w_n — оконная функция, X_k — отсчёт сигнала в спектральной области.

   Значения индексов k соответствуют частотам (2.8):

$$f_k = \frac{f_s}{N} k, \qquad (2.8)$$

   где f_s — частота дискретизации сигнала.

3. *Отображение энергетического спектра на мел-шкалу.*

   Полученное представление сигнала в частотной области разбивают на диапазоны с помощью блока треугольных фильтров(рис.2.6). Границы фильтров рассчитывают в шкале мел.

$$m = 1127 \ln 1 + \frac{f}{700}, \qquad (2.9)$$

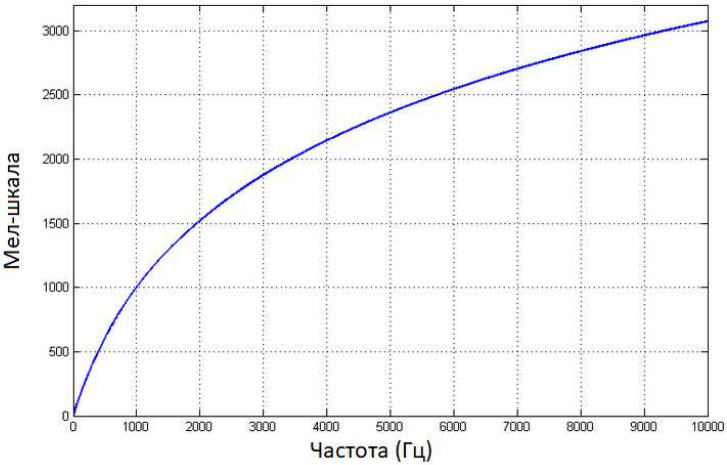

Рис. 2.5. Зависимость мел от герц

где, m — высота звука в мелах, f — высота звука в герцах.

Зависимость мел от герц приведена на рисунке 2.5.

Далее, проводится перенос фильтров из мел-области обратно в частотную, используя выражение:

$$f = 700\left(\exp\left(\frac{Mel(f)}{1127}\right) - 1\right), \tag{2.10}$$

где f — частоты по обычной (линейной) шкале, $Mel(f)$ — частоты по мел-шкале.

Полученные на частотной оси амплитудно-частотные характеристики фильтров будут собираться в области низких частот, тем самым обеспечивая более высокое разрешение там, где оно необходимо для распознавания (рис.2.7).

Расположение фильтров определяется следующей формулой:

$$F_k = f\left(M_{min} - k\frac{M_{max} - M_{min}}{K+1}\right), k = 0, \ldots, K-1 \tag{2.11}$$

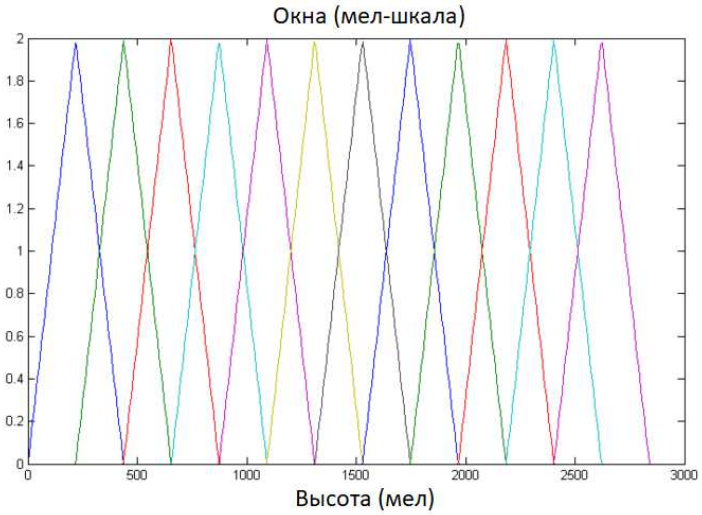

Рис. 2.6. Расположение треугольных фильтров на мел-шкале

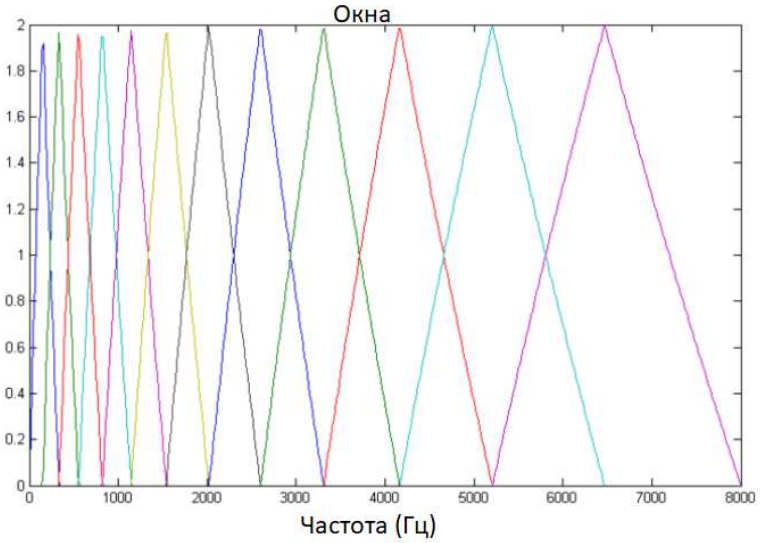

Рис. 2.7. Расположение треугольных фильтров на частотной оси

где M_{max} — мел-представление максимальной частоты, M_{min} — мел-представление минимальной частоты, K — количество фильтров.
Для вычисления весовых коэффициентов можно использовать полученные частоты, или же перейти к индексам семплов:

$$F_n(k) = \frac{N}{f_s} F(k), \tag{2.12}$$

где f_s — частота дискретизации, N – количество отсчетов.

Переход к индексам семплов упрощает дальнейшие вычисления, так как избавляет от надобности определять частоту каждого семпла. Полученные индексы подставляем в формулу вычисления функции k-го фильтра:

$$\begin{cases} 0, & n < F_n(k-1) \\ \dfrac{n - F_n(k-1)}{F_n(k) - F_n(k-1)}, & F_n(k-1) \le n < F_n(k) \\ \dfrac{n - F_n(k+1)}{F_n(k) - F_n(k+1)}, & F_n(k) \le n < F_n(k+1) \\ 0, & F_n(k+1) \le n \end{cases}$$

Перемножим вектора энергетического спектра сигнала и оконную функцию найдем энергию сигнала, попадающую в каждое из окон анализа:

$$x_m = \sum_{n=0}^{N-1} |X_k|^2 W(k, n) \tag{2.13}$$

где x_m — энергетический коэффициент от m-ого фильтра, $m = 1, \ldots, M$, M — количество фильтров; X_k — амплитудные коэффициенты спектра сегмента, $W(k, n)$ — функция k-ого фильтра.

В результате получается небольшой набор коэффициентов x_m, которые содержат спектральную информацию аудио сегментов.

4. Следующим шагом является *вычисление логарифма спектральной плотности* (2.14) для сжатия пространства при-

знаков.

$$L(k) = \ln\left(\sum_{n=0}^{N-1} |x_k|^2 W(k,n)\right), k = 0, \ldots, K-1 \qquad (2.14)$$

5. Завершающий шаг — *выполнение дискретного косинусного преобразования Фурье* (2.15) с целью уменьшения степени корреляции коэффициентов.

$$C_i(n) = \ln\left(\sum_{k=0}^{K-1} L(k) \cos\frac{\pi n}{K}\left(k + \frac{1}{2}\right)\right), n = 0, \ldots, N-1 \quad (2.15)$$

Темпоральные изменения спектров играют важную роль в восприятии слуха человеком. Одним из способов получения этой информации является извлечение производных признаков (дельта значений), которые измеряют изменение краткосрочных спектров с течением времени. Дельта значения для кепстральных коэффициентов вычисляются по формуле (2.16):

$$d(n) = \frac{C(n+1) - C(n-1)}{2}, \qquad (2.16)$$

где $d(n)$ — коэффициент дельта, n — индекс кадров анализа.

Двойные дельта значения вычисляются аналогично, с разницей в том, что вместо кепстральных коэффициентов $C(n)$ используются вычисленные дельта значения.

2.1.3 Метод K ближайших соседей (KNN)

Классификатор K ближайших соседей (KNN) является одним из простейших алгоритмов машинного обучения. Он относится к классу методов, основанных на сравнении данных, хранящихся в памяти, с новыми объектами. При появлении неизвестного объекта для предсказания метод выделяет среди всех k известных объектов, похожих на этот объект. Выделение k объектов происходит по степени близости к искомому объекту. Для случайного объекта выборки $w \in X$ необходимо расположить элементы обучающей выборки $x_1, \ldots, x_m$ в

порядке возрастания расстояний до w:

$$\rho(w, x_{1,w}) \leq \rho(w, x_{2,w}) \leq \ldots \leq \rho(w, x_{l,w}) \tag{2.17}$$

где $x_{i,w}$ — i-й сосед элемента w, при этом каждый из элементов порождает собственную перенумерацию выборки $x_{1,w}, x_{2,w}, \ldots, x_{m,w}$. Для i-го соседа то же самое расстояние обозначается как
$y_{i,w} = y * (x_{i,w})$.

Главной задачей данного метода является выбор коэффициента k — это количество соседей, сравниваемое с объектом классификации. Обычно выбирается пользователем. Когда количество соседей равно 1, то этот метод называется алгоритмом ближайшего соседа (1NN). Он классифицирует объект $w \in X^l$ к тому классу, к которому относится ближайший обучающий объект:

$$a\left(w, X^l\right) = y_{1,w} \tag{2.18}$$

Обучение сводится к обычному запоминанию выборки. Существует опасность, что среди объектов из обучающей выборки есть такой, который находится в окружении объектов другого класса, тогда не только он сам будет классифицирован неправильно, но и окружающие объекты, для которых он будет ближайшим, тоже будут классифицированы неправильно. Чтобы уменьшить воздействие таких объектов, применяется модификация алгоритма ближайшего соседа — k ближайших соседей, в котором объекты классифицируются путем «голосования», когда каждый из соседей $x_{i,u}, k = \overrightarrow{1,k}$ голосует за отнесение объекта w к своему классу. Данный алгоритм работает по достаточно простой схеме:

1. *Вычислить расстояние от классифицируемого объекта до каждого из объектов обучающей выборки.*

   В данной работе используется Евклидово расстояние, которое вычисляется по формуле (2.19):

$$d_{ab} = \sqrt{\sum_{i=1}^{n} (x_{ai} - x_{bi})^2}, \tag{2.19}$$

   где a и b — точки в n—мерном пространстве, i — порядковый номер признака, x_{ai} и x_{bi} — координаты точек a и b по признаку

i.

2. *Выбрать k элементов, расстояние до которых минимально.*

3. *Класс классифицируемого объекта — это класс, который наиболее часто встречается среди k ближайших соседей.*

В итоге, выигрывает тот класс, который наберет большее число голосов, которые определяются по формуле (2.20):

$$a\left(w, X, k\right) = \arg\max_{y \in Y} \sum_{i=1}^{k} \left[y_{i,w} = y\right] \tag{2.20}$$

где w — новый объект, X — обучающая выборка, y — класс, Y — множество классов, $y_{i,w}$ — класс i-го соседа w, k — количество соседей. Недостаток способа заключается в том, что два и более классов могут набирать одинаковую максимальную сумму голосов. Во избежание такой ситуации, каждому учебному образцу присваивается вес, задающий вклад i-го соседа в классификацию.

В данной работе $w_i = 1 - \frac{d_i}{d_{k+1}}$, где d_{k+1} — расстояние от $k+1$ — ближайшего соседа к тестовому образцу.

$$a\left(w, X, k\right) = \arg\max_{y \in Y} \sum_{i=1}^{k} \left[y_{i,w} = y\right] w_i \tag{2.21}$$

Таким образом, объекты из обучающей выборки, находящиеся вдали от исследуемого объекта, меньше способствуют классификации.

Глава 3

Программная реализация и экспериментальная оценка точности классификатора жанров

3.1 Программная реализация классификатора жанров

В данном разделе рассмотрена программная реализация системы классификации жанра. Разработка происходила на языке программирования Python с использованием функций таких библиотек, как: IPython, Numpy, Scipy, Pandas, Scikit-learn, Librosa, Matplotlib. Структура системы идентификации жанра показана на рисунке 3.1.

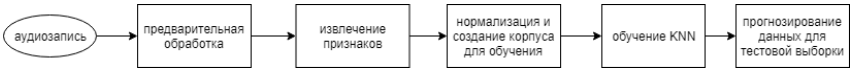

Рис. 3.1. Структура системы идентификации жанра

1. *Предварительная обработка.*

Поданная на вход системы аудиозапись проходит предварительную обработку: *get_ sample_ arrays* — функция, представляющая аудиофайл в виде временного ряда с плавающей запятой (рис. 3.2).

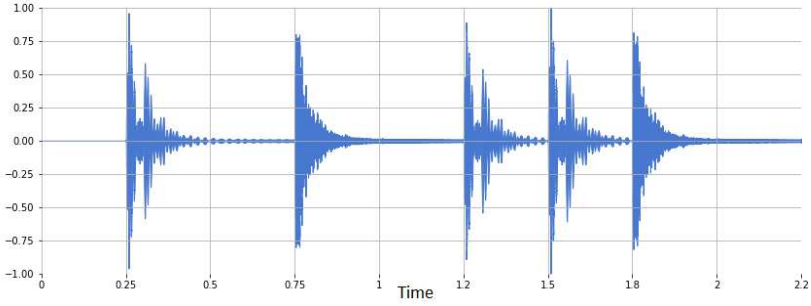

Рис. 3.2. Предварительная обработка аудиозаписи

2. *Извлечение признаков.*

   Далее из обработанного сигнала извлекается набор признаков, описанный в главе 2: спектральный центроид, спектральный спад, спектральный поток, частота переходов через ноль, MFCC, с помощью функции *extract_features*.

3. *Нормализация.*

   В процессе нормализации все значения приводятся к единому диапазону [-1, 1]. В данной работе для этого был использован класс MinMaxScaler из библиотеки Scikit-learn.

   Результат работы пунктов 1-3 представлен на рисунке 3.3 Полученные массивы извлеченных признаков заносятся в таблицу. Структура DataFrame из библиотеки Pandas отлично подходит для представления реальных данных: каждая строка представляет собой признаковое описание объекта исследования. Столбцы — признаки объекта. Данная таблица сохраняется в csv-файл.

4. *Создание и обучение модели.*

   Данные выгружаются из csv-файла. Создается объект классификатора *KNeighborsClassifier* из библиотеки Scikit-learn. Этот

```
Extracting features from audios...
.....Working in folder: ChaChaCha
.....Working in folder: Jive
.....Working in folder: Quickstep
.....Working in folder: Rumba-American
.....Working in folder: Rumba-International
.....Working in folder: Rumba-Misc
.....Working in folder: Samba
.....Working in folder: Tango
.....Working in folder: VienneseWaltz
.....Working in folder: Waltz
Normalizing the data...
data_set.csv has been created and sent to the project folder!
```

Рис. 3.3. Предобработка сигнала, извлечение признаков и нормализация

класс содержит в себе все необходимые для обучения и тестирования методы и функции. Для обучения модели используется метод fit. С помощью метода dump из библиотеки joblib сохраняем модель в pkl-файл. На рисунке 3.4 приведен результат выполнения функции.

```
Dataset shape:(698, 46)
Training the model.....
Trained and saved the model to project folder successfully.
```

Рис. 3.4. Создание и обучение модели

5. *Прогнозирование.*

Тестовые данные представляются в виде временного ряда с плавающей запятой. Функция *extract_features* извлекает вектора признаков, и вектора нормируются. Загружается обученная модель из pkl-файла. Для предсказания результатов тестовой выборки используется метод predict (рис. 3.5).

```
Extracting sample arrays for files...
DONE!
Extracting features from sample arrays...
DONE!
---------------------------------- Predicted Labels ----------------------------------

Albums-AnaBelen_Veneo-03.wav : Rumba-International
Albums-AnaBelen_Veneo-03.wav : Rumba-International
Albums-AnaBelen_Veneo-11.wav : Rumba-International
Albums-AnaBelen_Veneo-11.wav : Rumba-International
Albums-AnaBelen_Veneo-13.wav : Rumba-American
Albums-AnaBelen_Veneo-13.wav : Rumba-American
Albums-Ballroom_Classics4-05.wav : Waltz
Albums-Ballroom_Classics4-05.wav : Waltz
Albums-Ballroom_Classics4-11.wav : Rumba-International
Albums-Ballroom_Classics4-11.wav : Rumba-International
Albums-Ballroom_Magic-04.wav : Waltz
Albums-Ballroom_Magic-04.wav : Waltz
Albums-Ballroom_Magic-06.wav : Tango
```

Рис. 3.5. Прогнозирование тестовых данных

Также была создана функция *tempo_ comparison*, которая используется для проверки автоматически вычисленного темпа с эталонным. Результат работы данной функции приведен в разделе 3.4.

Для проверки эффективности алгоритма идентификации жанра музыкального произведения были проведены эксперименты, описанные ниже.

3.2 Оценка качества работы алгоритма

Для независимой и устойчивой оценки качества работы алгоритма применяется 10-блочная перекрестная проверка. При выполнении десятиблочной перекрестной проверки данные сначала разбиваются на десять частей (примерно) одинакового размера, называемых блоками (folds). Из этих 10 подмножеств для обучения выбираются 90% случайно выбранных образцов, а оставшиеся 10% используются для тестирования. Так повторяется для всех 10 блоков. Финальная оценка считается как среднее оценок для каждого блока.

Причина, по которой необходимо разбивать данные на обучающий и тестовый наборы, заключается в том, что нас интересует, насколько хорошо наша модель обобщает результат на новые, ранее неизвестные данные. Нас интересует не качество подгонки модели к обучающим данным, а правильность ее прогнозов для данных, не участвовавших

в обучении.

3.3 Подбор параметров классификатора жанров

При обучении любого классификатора существует проблема инициализации начальных параметров модели. При этом результат работы системы распознавания существенно зависит от начальных значений параметров модели.

Выбор оптимального количества ближайших соседей осуществляется путем перебора и оценки точности распознавания при заданном параметре k.

В процессе подбора оптимального k было выявлено, что точность распознавания повышается при увеличении параметра k до 7, а затем этот процесс значительно убывает. Это позволяет судить о том, что $k = 1$, $k = 7$ в алгоритме KNN и $k = 12$, $k = 15$ в wKNN, наилучшим образом подходят для решения поставленной задачи.

Рисунки 3.6 и 3.7 иллюстрируют процесс подбора параметра k для алгоритма kNN и его модификации wKNN.

3.4 Подбор эффективных признаков аудиозаписи

Результаты, полученные в таблице 3.1, сообщают о том, что выбор векторов признаков оказывает большое влияние на качество классификации. Наилучшее качество классификации жанра было достигнуто при применении коэффициентов MFCC, их производных и расчетного темпа.

Добавление информации о темпе улучшает характеристики для всех признаков. На рисунке 3.8 показано, насколько автоматически вычисленный темп отличается от эталонного. Было установлено, что самую большую погрешность имеют жанры с большим темпом: квикстеп, венский вальс и джайв. Танцевальные стили с медленным темпом: медленный вальс, самба, румба, ча-ча-ча,напротив определяются с большей точностью.

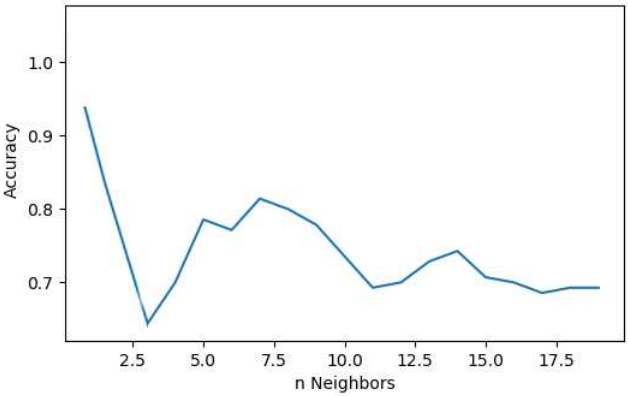

Рис. 3.6. Выбор параметра k для алгоритма KNN

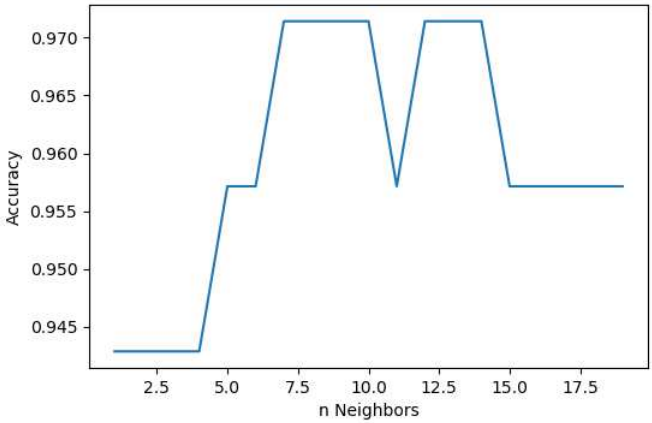

Рис. 3.7. Выбор параметра k для алгоритма wKNN

Для классификации жанра авторы работ из таблицы 1.2 использовали лишь ритмические признаки аудиозаписи. Представленные в

Таблица 3.1. Оценки точности классификации

Признаки	1NN	KNN	wKNN
спектральный центроид, спектральный спад, спектральный поток, частота переходов аудиосигналов через ноль	70,1%	71% (k=4)	73,5% (k=3)
спектральный центроид, спектральный спад, спектральный поток, частота переходов аудиосигналов через ноль +BPM	71,2%	72% (k=4)	75% (k=3)
MFCC	84,8%	80% (k=7)	85,7% (k=15)
MFCC + BPM	87,5%	81% (k=7)	88,3% (k=15)
MFCC + производные 1-го и 2-го порядка	87,3%	84% (k=7)	88,9% (k=12)
MFCC + производные 1-го и 2-го порядка + BPM	90,7%	88% (k=7)	92,1% (k=12)
спектральный центроид, спектральный спад, спектральный поток,частота переходов аудиосигналов через ноль + MFCC	83%	79,9% (k=8)	84,5% (k=15)
спектральный центроид, спектральный спад, спектральный поток,частота переходов аудиосигналов через ноль + MFCC + BPM	85,6%	81,8% (k=7)	86,7% (k=15)
спектральный центроид, спектральный спад, спектральный поток,частота переходов аудиосигналов через ноль + MFCC + производные 1-го и 2-го порядка	75,1%	74% (k=6)	76% (k=15)
спектральный центроид, спектральный спад, спектральный поток,частота переходов аудиосигналов через ноль + MFCC+ производные 1-го и 2-го порядка + BPM	77,3%	72% (k=3)	77,9% (k=15)

данной работе результаты показывают, что, используя различные настройки параметров (параметра классификатора k и набора векторов признаков (см. табл. 3.1), точность, полученная с помощью алгоритма KNN и смешивания спектральных и ритмических признаков,составляет 92,1%, что превосходит лучший опубликованный результат из таблицы 1.2.

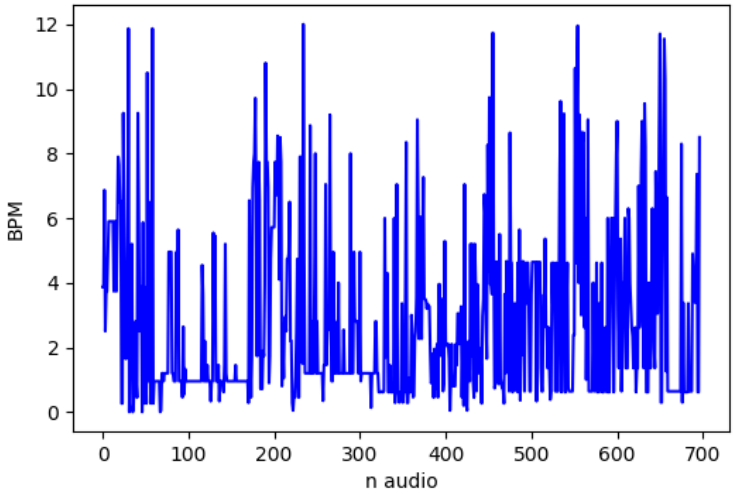

Рис. 3.8. Сравнение эталонного и расчетного темпа

```
ChaChaCha\Albums-Cafe_Paradiso-05: Estimated tempo:  [ 123.046875]
 Annotated tempo:  124

[ 0.953125]
ChaChaCha\Albums-Cafe_Paradiso-05: Estimated tempo:  [ 123.046875]
 Annotated tempo:  124

[ 0.953125]
ChaChaCha\Albums-Cafe_Paradiso-06: Estimated tempo:  [ 123.046875]
 Annotated tempo:  124

[ 0.953125]
ChaChaCha\Albums-Cafe_Paradiso-06: Estimated tempo:  [ 123.046875]
 Annotated tempo:  124

[ 0.953125]
ChaChaCha\Albums-Cafe_Paradiso-07: Estimated tempo:  [ 123.046875]
 Annotated tempo:  124
```

Рис. 3.9. Результат работы функции *tempo_ comparison*

Заключение

В ходе данной работы были решены следующие задачи:

1. Реализован алгоритм извлечения признаков из аудиозаписи. Для исследования использовались: спектральный центроид, спектральный спад, частота переходов аудиосигнала через ноль, мел-частотные кепстральные коэффициенты и темп, вычисленный с помощью библиотеки Librosa.

2. Реализован алгоритм идентификации жанра музыкального произведения с использованием модифицированного подхода классификации kNN на языке программирования Python.

3. Проведена оценка точности, достигаемая разработанным алгоритмом на наборе данных Ballroom. Результат (92,1%) был получен при применении коэффициентов MFCC, их производных и расчетного темпа, что превосходит опубликованный результат 90,2%.

Литература

[1] *Ромацкий Д. Б.* Автоматическая классификация музыкальных произведений по жанрам // *Томск. гос. ун-т. Факультет информатики.* — 2014.

[2] Aggregate features and a da b oost for music classification / J. Bergstra, N. Casagrande, D. Erhan et al. // *Machine learning.* — 2006. — Vol. 65, no. 2-3. — Pp. 473–484.

[3] Ballroom. http://mtg.upf.edu/ismir2004/contest/tempoContest.

[4] *D.Pye.* Content-based methods for the management of digital music. // Proc. Int. Conf. Acoust., Speech Signal Processing (ICASSP). — 2000.

[5] *Erzin E. et al.* Automatic classification of musical genres using intergenre similarity // *IEEE Signal Processing Letters.* — 2007. — Vol. 14, no. 8. — Pp. 521–524.

[6] Evaluating rhythmic descriptors for musical genre classification / F. Gouyon, S. Dixon, E. Pampalk, G. Widmer // Proceedings of the AES 25th International Conference. — 2004. — Pp. 196–204.

[7] *Gjerdingen R. O., Perrott D.* Scanning the dial: The rapid recognition of music genres // *Journal of New Music Research.* — 2008. — Vol. 37, no. 2. — Pp. 93–100.

[8] *Gouyon F.* A computational approach to rhythm description-Audio features for the computation of rhythm periodicity functions and their use in tempo induction and music content processing. — Universitat Pompeu Fabra, 2006.

[9] Gtzan. http://marsyasweb.appspot.com/download/data_sets.

[10] *Holzapfel A., Stylianou Y.* Rhythmic similarity of music based on dynamic periodicity warping // Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on / IEEE. — 2008. — Pp. 2217–2220.

[11] *Huron D.* Perceptual and cognitive applications in music information retrieval // *Perception*. — Vol. 10, no. 1. — Pp. 83–92.

[12] *Li T., Ogihara M.* Toward intelligent music information retrieval // *IEEE Transactions on Multimedia*. — 2006. — Vol. 8, no. 3. — Pp. 564–574.

[13] Magnatune. http://magnatune.com/.

[14] Maryas. http://marsyas.info/about/projects.html.

[15] On rhythm and general music similarity. / T. Pohle, D. Schnitzer, M. Schedl et al. // ISMIR. — 2009. — Pp. 525–530.

[16] *Peeters G.* Spectral and temporal periodicity representations of rhythm for the automatic classification of music audio signal // *IEEE Transactions on Audio, Speech, and Language Processing*. — 2011. — Vol. 19, no. 5. — Pp. 1242–1252.

[17] *Tzanetakis G., Cook P.* Musical genre classification of audio signals // *IEEE Transactions on speech and audio processing*. — 2002. — Vol. 10, no. 5. — Pp. 293–302.

[18] Uspop. http://www.ee.columbia.edu/dpwe/research/musicsim/uspop2002.html.

[19] *West K., Cox S.* Features and classifiers for the automatic classification of musical audio signals. // ISMIR. — 2004.