



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

---

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**  
*К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ*  
*НА ТЕМУ:*

«Метод автоматического определения ритмического рисунка и  
темпа цифровой музыкальной записи с использованием  
байесовского иерархического моделирования»

Студент ИУ7-86Б  
(Группа)

\_\_\_\_\_  
(Подпись, дата)

А. А. Петрова  
(И.О. Фамилия)

Руководитель

\_\_\_\_\_  
(Подпись, дата)

К. А. Кивва  
(И.О. Фамилия)

2023 г.

# РЕФЕРАТ

Расчетно-пояснительная записка ?? с., ? рис., ? табл., ?? ист., ? прил.

TODO

## КЛЮЧЕВЫЕ СЛОВА

*темп музыки, ритм музыки, bpm, вейвлет, марковская модель, байесовская модель, нейросети.*

## СОДЕРЖАНИЕ

<b>РЕФЕРАТ</b>	<b>3</b>
<b>ВВЕДЕНИЕ</b>	<b>6</b>
<b>1 Аналитический раздел</b>	<b>7</b>
1.1 Темп, ритм и метр . . . . .	7
1.2 Проблема определения ритма и темпа . . . . .	8
1.3 Дискретное вейвлет-преобразование . . . . .	9
1.3.1 Общие сведения . . . . .	9
1.3.2 Определение ритма и темпа . . . . .	11
1.4 Скрытые модели Маркова . . . . .	12
1.4.1 Стохастическое моделирование . . . . .	12
1.4.2 Определение ритма . . . . .	14
1.5 Байесовское иерархическое моделирование . . . . .	14
1.5.1 Общие сведения . . . . .	14
1.5.2 Определение темпа и ритма . . . . .	15
1.6 Использование сверточных нейросетей . . . . .	17
1.6.1 Представление сигнала . . . . .	17
1.6.2 Архитектура сети . . . . .	17
1.7 Сравнение методов . . . . .	19
<b>2 Конструкторский раздел</b>	<b>22</b>
2.1 Определение переменного темпа . . . . .	22
2.2 Определение переменного тактового размера . . . . .	26
2.3 Структуры данных . . . . .	29
2.4 Структура ПО . . . . .	29
<b>3 Технологический раздел</b>	<b>31</b>

3.1	Выбор средств программной реализации . . . . .	31
3.2	Детали реализации . . . . .	31
3.3	Пользовательский интерфейс . . . . .	32
3.4	Тестирование приложения . . . . .	32
<b>ЗАКЛЮЧЕНИЕ</b>		<b>33</b>
<b>СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ</b>		<b>34</b>

# ВВЕДЕНИЕ

Автоматическая транскрипция музыки (АТМ) — это процесс преобразования акустического музыкального сигнала в ту или иную форму нотной записи [1]. Данную задачу можно разделить на несколько подзадач, к которым в том числе относятся задачи выделения информации о ритме и темпе музыки. Несмотря на то, что задачу АТМ для монофонических сигналов можно считать решенной [1], проблема создания автоматизированной системы, способной транскрибировать полифоническую (многоголосую) музыку без ограничений по степени полифонии или типу инструмента, остается открытой.

Цель данной работы – реализовать метод автоматического определения темпа и ритма музыки на основе байесовского иерархического моделирования.

Чтобы достигнуть поставленной цели, требуется решить следующие задачи:

- провести анализ предметной области и сформулировать проблему;
- проанализировать и сравнить основные методы определения темпа и ритма;
- разработать алгоритм решения поставленной задачи;
- спроектировать архитектуру разрабатываемого программного обеспечения;
- реализовать разработанный алгоритм;
- протестировать и сравнить результаты работы реализованного метода с результатами, полученными с помощью известных аналогов.

# 1 Аналитический раздел

## 1.1 Темп, ритм и метр

**Темп** – мера времени в музыке, упрощенно – «скорость исполнения музыки» [2].

Существует несколько способов измерения темпа. В классической музыке чаще всего используется словесное описание (как правило, на итальянском). Этот метод является неточным и дает лишь примерное представление о «скорости» исполнения музыкального произведения. Примеры такого описания: адажио, ленто (медленные темпы); анданте, модерато (средние темпы); аллегро, виво (быстрые темпы).

Второй, более точный способ измерения темпа – это число ударов в минуту (beats per minute, сокращенно bpm). Данный метод напрямую связан с частотой колебания маятника в метрономе (устройстве, предназначенном для точного ориентира темпа при исполнении музыки). Стандартным темпом считается 120 bpm, т. е. 2 Гц.

В данной работе будет использоваться второй способ измерения темпа (в bpm).

**Ритм** – организация музыки во времени [3]. Ритмическую структуру музыки образует последовательность длительностей – звуков и пауз.

Ритм в музыке принадлежит к числу терминов, дискуссии о которых ведутся в науке последние два столетия. Единого мнения по вопросу его определения нет. Чаще всего ритм определяется как регулярная, периодическая последовательность акцентов. Такое понимание ритма фактически идентично метру.

**Метр** в музыке – это чередование сильных и слабых долей в определенном темпе [2]. Обычно метр фиксируется с помощью тактового размера и тактовой черты (рис. 1.1).

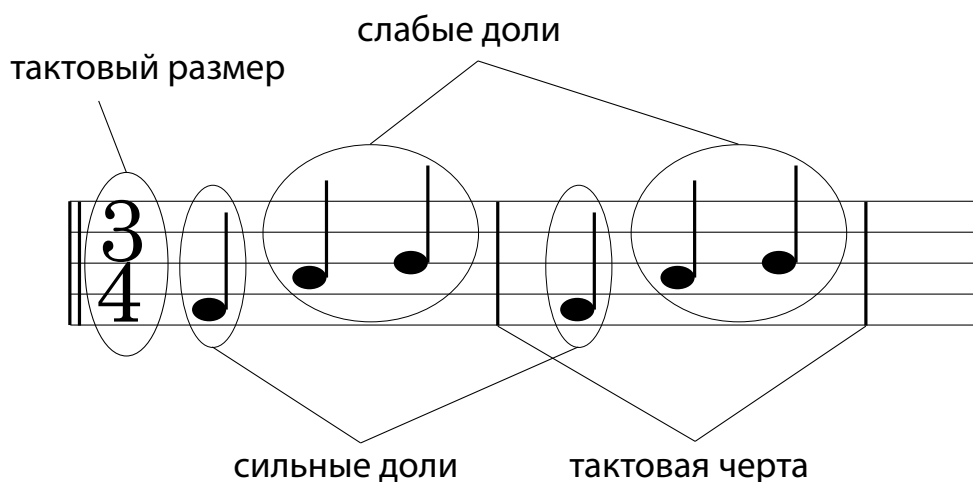


Рис. 1.1 – Обозначение метра

Размер задает относительную длительность каждой доли. Например, размер «3/4» говорит о том, что в такте 3 доли, каждая из которых представлена четвертной нотой. Можно сказать, что размер – числовое представление метра с указанием длительности каждой доли. Такт в свою очередь – единица метра, начинающаяся с наиболее сильной доли и заканчивающаяся перед следующей равной ей по силе (рис. 1.1).

В данной работе не будут учитываться тонкости различия ритма и метра. Соответственно, для измерения ритма будет использоваться числовое представление метра в виде тактового размера.

## 1.2 Проблема определения ритма и темпа

Основной проблемой автоматического определения ритма и темпа музыки является наличие некоторых особенностей в музыкальных записях с живыми инструментами, затрудняющих это определение. Одна из таких особенностей – это нечеткое попадание инструмента в ритмическую сетку. Такие небольшие отклонения на живых записях присутствуют всегда [4]. Они не заметны для уха человека, но могут осложнять автоматическое распознавание.

Также в некоторых случаях темп и ритм может изменяться в течение музыкального произведения. Пример переменного темпа приведен на рис. 1.2 (темп

обозначается числами сверху в bpm). На рис. 1.3 приведен пример переменного ритма (размера).

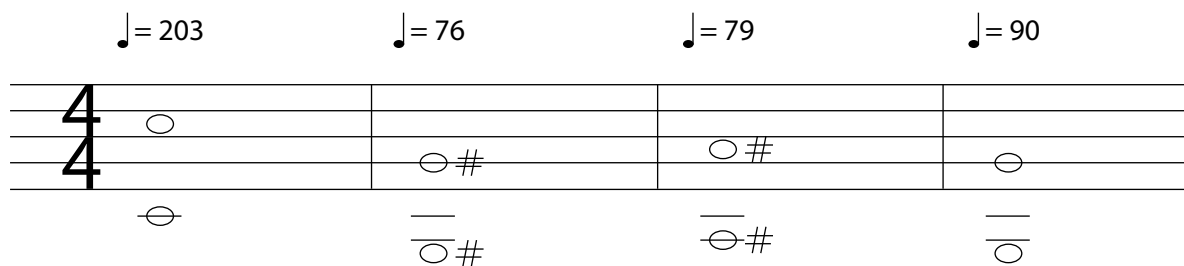


Рис. 1.2 – Пример переменного темпа (System of a down «Aerials»)

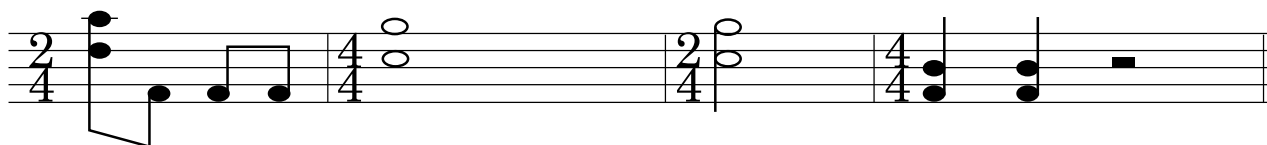


Рис. 1.3 – Пример переменного размера (Metallica «Master of puppets»)

В качестве критериев сравнения рассматриваемых далее методов выделе-  
ны следующие:

- точность результатов применения метода;
- возможность определения переменного темпа и ритма;
- ограничения на формат входного аудиофайла;
- размер использовавшегося для обучения датасета (если обучение необходимо).

### 1.3 Дискретное вейвлет-преобразование

### 1.3.1 Общие сведения

Так как преобразование Фурье не позволяет получить частотно-временное представление сигнала, оно подходит только для стационарных сигналов (т. е. сигналов, частотное наполнение которых не меняется во времени). Большинство же реальных аудио-сигналов являются нестационарными. Основная же проблема оконного преобразования Фурье (ОПФ) заключается в невозможности получить произвольно точное частотно-временное представление сигнала,



то есть нельзя определить для какого-то момента времени, какие спектральные компоненты присутствуют в сигнале. Эта проблема называется проблемой разрешения.

В качестве альтернативы ОПФ было разработано вейвлет-преобразование.

Основная идея вейвлет-преобразования – это разделение сигнала на высокие и низкие частоты с помощью фильтров [5]. После применения фильтров полученные низкие частоты снова пропускаются через два фильтра и т. д. При этом высокие частоты остаются неизменными. Эта операция называется декомпозицией.

На высоких частотах лучше разрешение по времени, а на низких – по частоте.

Фильтры для высоких и низких частот определяются следующими уравнениями [6]:

$$y_{high}[k] = \sum_{n=-\infty}^{\infty} x[n]g[2k - n], \quad (1)$$

$$y_{low}[k] = \sum_{n=-\infty}^{\infty} x[n]h[2k - n], \quad (2)$$

где  $x[n]$  – пропускаемый через фильтр сигнал (последовательность),  $h[n]$  и  $g[n]$  – импульсные характеристики (отклик на единичный импульс) низкочастотного и высокочастотного фильтров соответственно,  $k$  и  $n$  – целые числа, соответствующие отсчетам (теорема отсчетов [7]).

Выражение  $2k - n$  в формулах 1 и 2 позволяет обрезать сигнал, тем самым увеличив его масштаб в два раза (т. к. половина частот удаляется в результате фильтрации) [5].

Само ДВП (дискретное вейвлет-преобразование) описывается формулой:

$$W(j, k) = \sum_j \sum_k x(k)2^{-j/2}\psi(2^{-j}n - k), \quad (3)$$

где  $\psi(t)$  – функция преобразования, называемая материнским вейвлетом,  $j$  и  $k$  связаны с параметрами сдвига  $\tau$  (местоположение окна) и масштаба  $s$  (величина, обратная частоте).  $s = s_0^j$ ,  $\tau = ks_0^j\tau_0$ . В данном случае  $s_0 = 2$ ,  $\tau_0 = 1$ .

### 1.3.2 Определение ритма и темпа

Алгоритм определения ритма с помощью ДВП основан на обнаружении наиболее заметных периодов сигнала.

Сигнал сначала раскладывается на несколько частотных полос с помощью ДВП. Для этого сигнал «делится» пополам на высокие и низкие частоты, после чего низкие частоты снова разделяются пополам и т. д. Так продолжается до тех пор, пока не останутся два отсчета. Эта операция необходима, т. к. для высоких частот можно точнее указать их временную позицию, а для низких – их значение частоты [5]. После этого огибающая амплитуды во временной области каждой полосы извлекается отдельно. Это достигается за счет фильтрации нижних частот каждой полосы, применения полноволнового выпрямления и понижения частоты дискретизации [6]. Затем огибающие каждой полосы суммируются и вычисляется автокорреляционная функция. Пики автокорреляционной функции соответствуют различным периодам огибающей сигнала.

Фильтрация нижних частот:

$$y[n] = (1 - \alpha)x[n] - \alpha y[n], \quad (4)$$

где  $\alpha = 0,99$ .

Полноволновое выпрямление:

$$y[n] = \text{abs}(x[n]). \quad (5)$$

Понижение частоты дискретизации:

$$y[n] = x[kn]. \quad (6)$$

Нормализация в каждой полосе (удаление среднего значения) для исключения аномальных данных:

$$y[n] = x[n] - E[x[n]], \quad (7)$$

где  $E[x[n]]$  – среднее значение последовательности  $x[n]$ .

Автокорреляция:

$$y[k] = \frac{1}{N} \sum_{n=0}^{N-1} x[n]x[n+k]. \quad (8)$$

Из результата берутся первые пять пиков автокорреляционной функции, после чего рассчитываются и добавляются в гистограмму соответствующие им периодичности в bpm. Этот процесс повторяется в процессе прохождения по сигналу. Периодичность, соответствующая наиболее заметному пику конечной гистограммы, является предполагаемым темпом аудиофайла в bpm.

Основными недостатками рассмотренного метода определения темпа являются неточные (в некоторых случаях даже ошибочные) результаты на музыке определенных жанров (например, на классической музыке), а также невозможность определить переменный темп.

## **1.4 Скрытые модели Маркова**

### **1.4.1 Стохастическое моделирование**

Как уже было упомянуто выше, практически во всех музыкальных записях имеет место небольшое отклонение нот от ритмической сетки. Рассматриваемый метод рассчитан именно на работу с такими случаями. Также в данном методе подразумевается, что входные данные представлены в формате MIDI (Musical Instrument Digital Interface, стандарт обмена данными между цифровыми музыкальными инструментами). В MIDI файлах указывается информация о высоте ноты, ее длительности и силе нажатия [8].

Исследования показывают, что отклонения нот можно смоделировать с помощью распределения Гаусса относительно их идеальной длительности [9].

Тогда, если  $i$  – идеальная длительность ноты («намерение») в момент времени  $t$ , то ее исполненная длительность  $x_t$  моделируется функцией плотности вероятности  $f_i(x_t)$ .

Пусть  $Q = \{q_1, q_2, \dots, q_N\}$  – последовательность «намерений» в соответствующие моменты времени. Тогда наблюдаемая последовательность длительностей  $X = \{x_1, x_2, \dots, x_N\}$  определяется как:

$$P(X|Q) = \prod_{t=1}^N f_{q_t}(x_t). \quad (9)$$

В данном методе используются два типа моделей генерации ритмических рисунков для получения возможных ритмов:

- $n$ -граммная модель (длина ноты предсказывается исходя из предыдущих  $n-1$  нот в вероятностном смысле. Эта модель охватывает любые ритмические рисунки и может выдавать точную вероятность);
- «ритмический словарь» (состоит из всех известных ритмических рисунков за единицу времени. Хорошо представляет известные ритмические рисунки, в то время как неизвестные заменяются аналогичными существующими ритмами).

Обе модели можно представить в виде вероятностных сетей перехода состояний, где каждое состояние связано с предполагаемой длительностью ноты. Вероятность того, что номер состояния изменится в последовательности  $Q = \{q_1, q_2, \dots, q_N\}$  определяется как  $P(Q) = p_{q_0} \prod_{t=1}^N a_{q_{t-1}q_t}$ , где  $p_i$  – вероятность изначального нахождения в состоянии  $i$ , а  $a_{ij}$  – вероятность перехода из состояния  $i$  в состояние  $j$ .

Колеблющиеся длительности и возможные последовательности нот могут быть объединены в рамках скрытой модели Маркова как вероятности перехода  $A = \{a_{ij}\}$  и наблюдаемые вероятности  $B = \{b_i(x_t)\}$  соответственно. В таком случае вероятность наблюдения последовательности длительностей  $X$  определяется как:

$$P(X) = P(X|Q)P(Q). \quad (10)$$

### 1.4.2 Определение ритма

Задача заключается в том, чтобы найти временную последовательность  $Q$  номеров состояний, которая дает максимальную апостериорную вероятность  $P(Q|X)$  при заданной последовательности наблюдаемых длительностей  $X$  [9].

По теореме Байеса:

$$P(Q|X) = \frac{P(X|Q)P(Q)}{P(X)}. \quad (11)$$

Значит, максимизация апостериорной вероятности эквивалентна нахождению  $\operatorname{argmax} P(X|Q)P(Q)$  среди всех возможных  $Q$ .

Оптимальная последовательность состояний находится с помощью алгоритма Витерби для поиска наилучшего пути в вероятностной сети переходов.

Основной недостаток представленного метода заключается в необходимости входных данных быть в формате MIDI. Также к недостаткам можно отнести периодические неточности в результатах. Например, музыкальные фрагменты с разным темпом (к примеру, 116 bpm и 127 bpm) могут быть определены как имеющие одинаковый темп (в данном случае 120 bpm [9]).

## 1.5 Байесовское иерархическое моделирование

### 1.5.1 Общие сведения

Байесовская иерархическая модель - это метод статистического анализа, который использует байесовский подход для оценки параметров. Она состоит из нескольких уровней, где каждый уровень описывает определенный аспект данных [10].

Для каждого уровня иерархии определяются соответствующие параметры, которые описывают данные. Для оценки параметров используются априорные распределения, которые описывают предположения об этих параметрах до получения данных. Затем с помощью формулы 11 оцениваются парамет-

ры модели, учитывая как данные ( $P(X|Q)$ ), так и априорные распределения ( $P(Q)$ ). Таким образом, байесовская иерархическая модель позволяет учитывать неопределенность в данных и параметрах модели, а также учитывать различные уровни влияния на данные.

Термин  $P(X|Q)$  играет две роли в статистике, в зависимости от контекста [11]. Когда она рассматривается как функция от  $X$  для известного  $Q$ , то она называется **функцией массы вероятности**. Функция массы вероятности описывает вероятность любого исхода  $X$  при заданном значении  $Q$ . Но этот термин также можно рассматривать как функцию параметра  $Q$  для фиксированного  $X$ . В этом случае она называется **функцией правдоподобия** и описывает вероятность определенного значения параметра  $Q$  при заданном фиксированном значении  $X$ . В теореме Байеса используется апостериорное распределение ( $P(Q|X)$ ) как функция параметра  $Q$ . Следовательно,  $P(X|Q)$  рассматривается как правдоподобие параметра  $Q$ , а не масса вероятности  $X$ .

Член  $P(X)$  является константой по отношению к  $Q$  и служит нормирующим числом, которое гарантирует, что апостериорная вероятность не превышает 1.

Чтобы выполнить байесовский анализ, необходимо выбрать априорное распределение  $P(Q)$ . Этот выбор отражает мнение о возможных значениях  $Q$  до сбора данных.

Когда априорная и апостериорная вероятность относятся к одному и тому же распределению, они называются **сопряженными** [11]. Сопряженные значения удобны, они часто облегчают получение апостериорных распределений. Однако сопряженность вовсе не обязательна для байесовского анализа.

### 1.5.2 Определение темпа и ритма

В случае с темпом музыки формула 11 принимает вид:

$$P(t|d) = \frac{P(d|t)P(t)}{P(d)}, \quad (12)$$

где  $d$  – это собранные данные с известным темпом (датасет), а  $t$  – искомый темп.

Аналогично и с ритмом (тактовым размером).

Однако темп музыки имеет также некоторую корреляцию с жанром. Поэтому в таком случае лучше всего применять иерархический вариант рассмотренной выше байесовской модели. Тогда на первом уровне будут располагаться так называемые **гипераприорные** распределения [10] для темпа и для жанра ( $P(t)$  и  $P(g)$  соответственно). На основе этих распределений вычисляются параметры, располагающиеся на втором уровне и необходимые для задания распределения функции правдоподобия (likelihood\_params) (см. рис. 1.4). На третьем же уровне находится функция правдоподобия, которая задается на основе вычисленных ранее параметров.

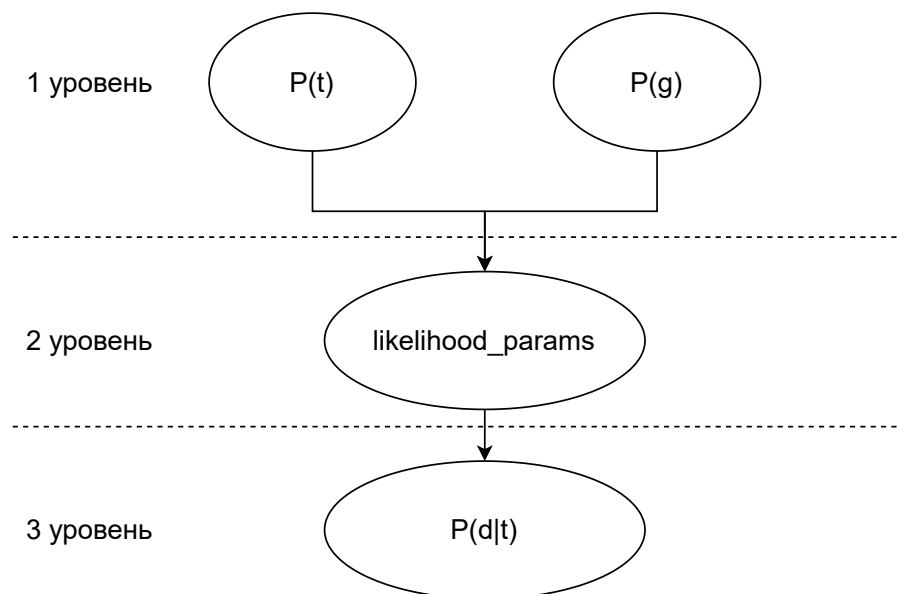


Рис. 1.4 – Схема иерархической байесовской модели

Таким образом, байесовское иерархическое моделирование позволяет немного увеличить точность определения ритма по сравнению с марковскими моделями (примерно на 2% [12]). Однако главным недостатком байесовского моделирования является возможное неправильное задание априорного распределения. Также к недостаткам можно отнести определение только постоянного темпа и ритма.

## **1.6 Использование сверточных нейросетей**

### **1.6.1 Представление сигнала**

Сигнал представляется в виде спектрограммы по шкале мела, чтобы снизить объем данных, который должен быть обработан нейросетью (мел, от слова «мелодия», - психофизическая (субъективная) единица высоты звука [13]). Шкала мела выбрана вместо линейной шкалы из-за ее связи с человеческим восприятием и диапазонами частот инструментов.

Чтобы создать спектрограмму, сигнал конвертируется в моно, его дискретизация понижается до 11025 Гц, после чего используются полуперекрывающиеся окна из 1024 отсчетов [14]. Это эквивалентно частоте кадров 21,5 Гц, что (согласно теореме отсчетов) достаточно для представления темпа до 646 bpm. Каждое окно преобразуется в 40-полосный спектр в шкале мел, охватывающий диапазон от 20 до 5000 Гц. В качестве длины спектрограммы выбрано 256 кадров, что примерно равняется 11,9 с.

### **1.6.2 Архитектура сети**

Архитектура рассматриваемой сети представлена на рис. 1.5.

Сначала входные данные обрабатываются тремя сверточными слоями, каждый из которых состоит из 16 фильтров размера 1x5. С помощью этих фильтров сопоставляется ритмическая структура сигнала.

После этого идут четыре модуля с несколькими фильтрами. Каждый из модулей состоит из среднего слоя пулинга («avg pooling»), шести параллельных сверточных слоев с фильтрами разных размеров (от 1x32 до 1x256), слоя конкатенации и т. н. «узкого» («bottle-neck») слоя, предназначенного для уменьшения размерности. С помощью этих модулей достигаются две цели:

- 1) Пулинг по оси частот для суммирования диапазонов мел.
- 2) Сопоставление сигнала с различными фильтрами, способными обнаруживать длительные временные зависимости.



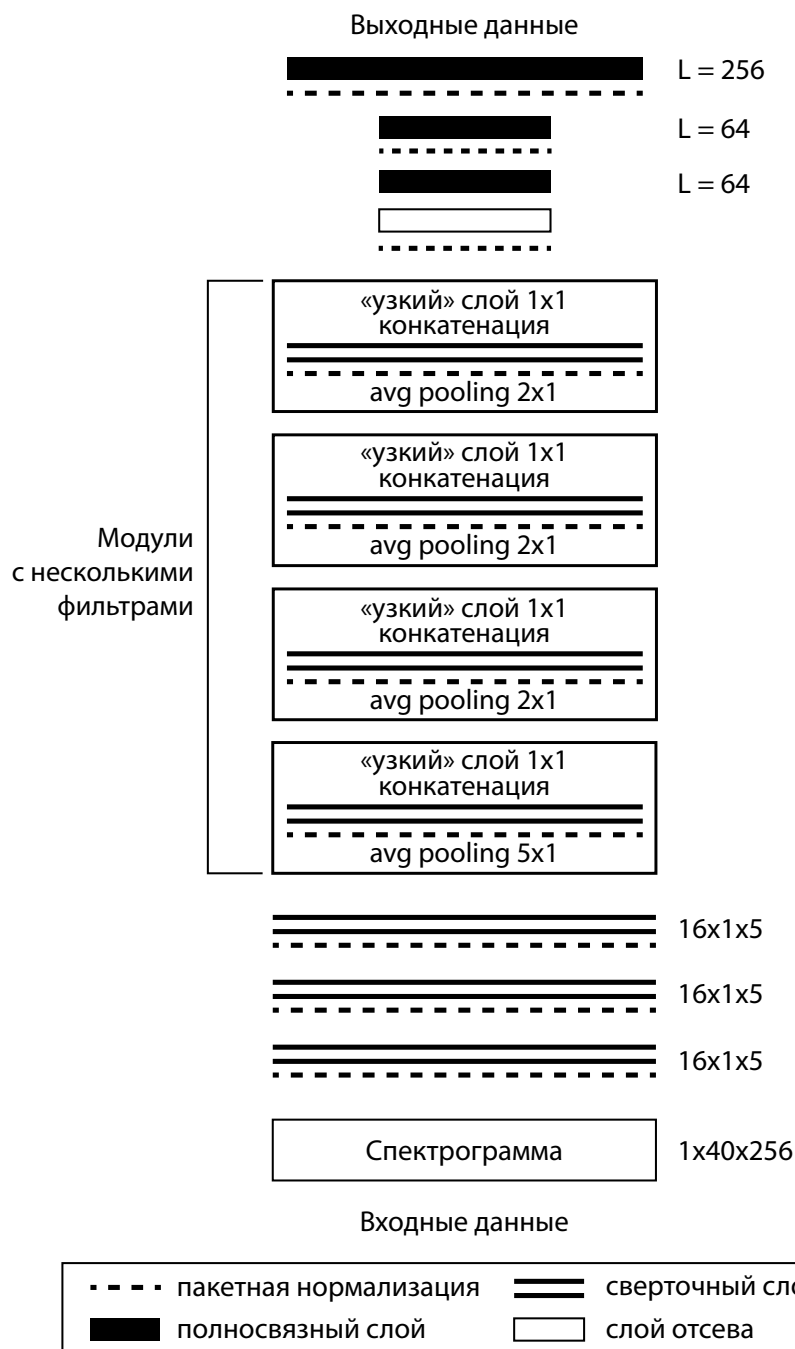


Рис. 1.5 – Схема архитектуры нейросети

Чтобы классифицировать свойства, полученные из сверточных слоев, добавляются два полносвязных слоя (по 64 единицы каждый), за которыми следует выходной слой с 256 единицами. Выходной слой использует softmax в качестве функции активации, а все остальные слои используют ELU [15]. Каждому сверточному или полносвязному слою предшествует пакетная нормализация [16]. Первому полносвязному слою также предшествует слой отсева с

$p = 0,5$  («dropout») для противодействия переобучению.

Всего сеть имеет 2921042 обучаемых параметра.

В результате выбирается один из 256 вариантов темпа от 30 до 285 bpm.

Таким образом, сверточные нейросети позволяют определять темп с достаточно высокой точностью (процент правильных оценок с допустимой погрешностью в 4%) (до 92% на основе комбинированной выборки, состоящей из аудиофайлов различных жанров с темпом от 44 до 216 bpm [14]). Также данный метод можно использовать и при определении глобального темпа, не только для фрагментов. Но он по-прежнему не позволяет определить переменный темп, а также не предназначен для определения ритма. Помимо этого нейросетевые методы имеют такие недостатки, как необходимость обучающих датасетов больших объемов, зависимость от исходных данных и долгое время обучения.

## 1.7 Сравнение методов

По результатам рассмотрения перечисленных выше методов была составлена таблица 1.1.

Как видно из таблицы, ни один метод в своем изначальном варианте не предполагает определение переменного темпа и ритма. Однако метод скрытых марковских моделей при небольшой модификации может позволить определить переменный темп и ритм [9].

Также стоит заметить, что все методы, кроме ДВП, содержат обучаемые параметры. В скрытых марковских моделях – это множество  $\{a_{ij}\}$ , а в байесовском иерархическом моделировании – множество  $\{\pi_{kk'}\}$ . В обоих методах обучение происходит с помощью статистической оценки. Размеры датасетов в таблице были указаны исходя из данных, использовавшихся для обучения соответствующих моделей в исследованиях.

Таблица 1.1 – Сравнение рассмотренных методов

Метод	Точность результатов	Переменный темп и ритм	Формат входного аудиофайла	Размер использованного датасета (кол-во аудиофайлов)
ДВП	Примерно 65% (13 верных из 20) [6]	Не определяются	Нет ограничений	Обучение не требуется
Скрытые марковские модели	Примерно 80% (при допустимой погрешности 4%)	Могут определяться при модификации метода	MIDI	88 [9]
Байес	Примерно 82%	Не определяются	Нет ограничений	100 [12]
Сверточная нейросеть	До 92%	Не определяются	Нет ограничений	8596 [14]

## Выводы

В этом разделе была проанализирована предметная область и сформулирована проблема. А также была проведена классификация и сравнение основных существующих методов решения поставленной задачи.

Таким образом, необходимо разработать метод определения переменного ритма и переменного темпа музыки на основе байесовского иерархического моделирования. Для этого требуется составить модели, которые будут использовать статистические данные о музыке, такие как темп и тактовый размер. Модель для определения темпа должна также учитывать жанр анализируемой му-

зыки. Составленные модели должны быть обучены на наборе данных, включающих в том числе различные жанры [17]. После обучения моделей, они должны быть протестированы на новых данных, чтобы оценить их точность и эффективность.

Входные данные:

- аудиофайл;
- жанр музыки (строка).

Выходные данные:

- набор темпов (целые числа, в BPM);
- набор тактовых размеров (обыкновенные дроби в формате «a/b»).

Оба набора выходных данных должны сопровождаться временем, соответствующим каждому темпу или размеру.

Ограничения:

- загружаемые аудиофайлы должны быть в формате mp3;
- знаменатель тактового размера принимается равным 4.

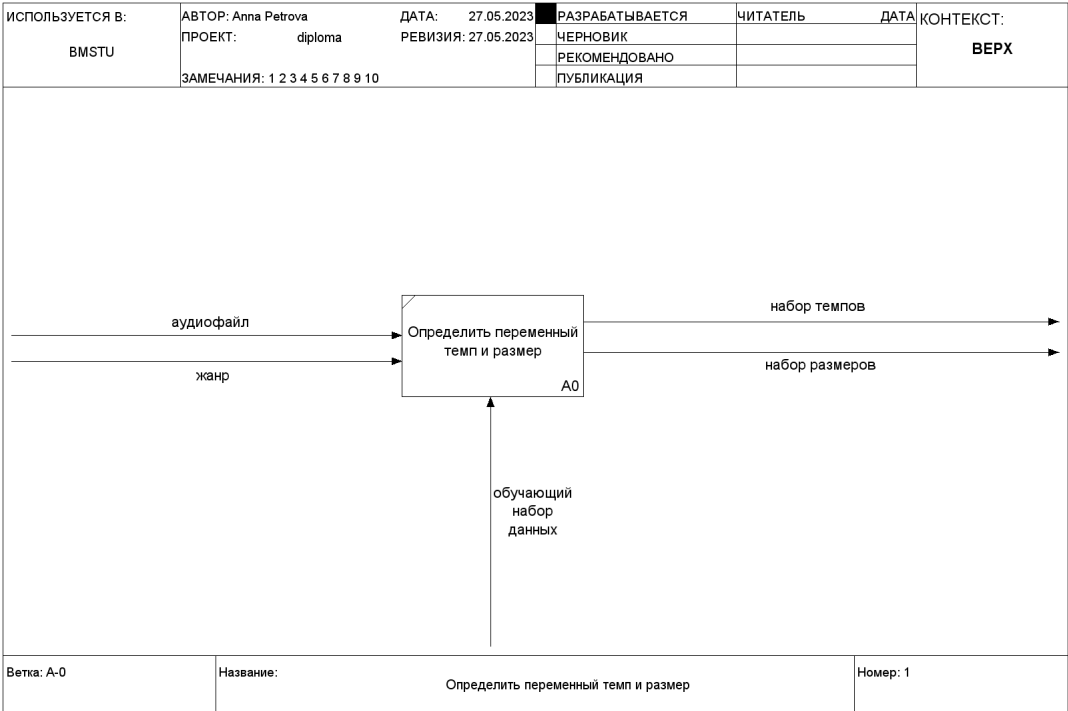


Рис. 1.6 – IDEF0 нулевого уровня для поставленной задачи

## 2 Конструкторский раздел

### 2.1 Определение переменного темпа

На рисунках 2.7 – 2.10 приведены IDEF0-диаграммы для алгоритма определения переменного темпа музыки.

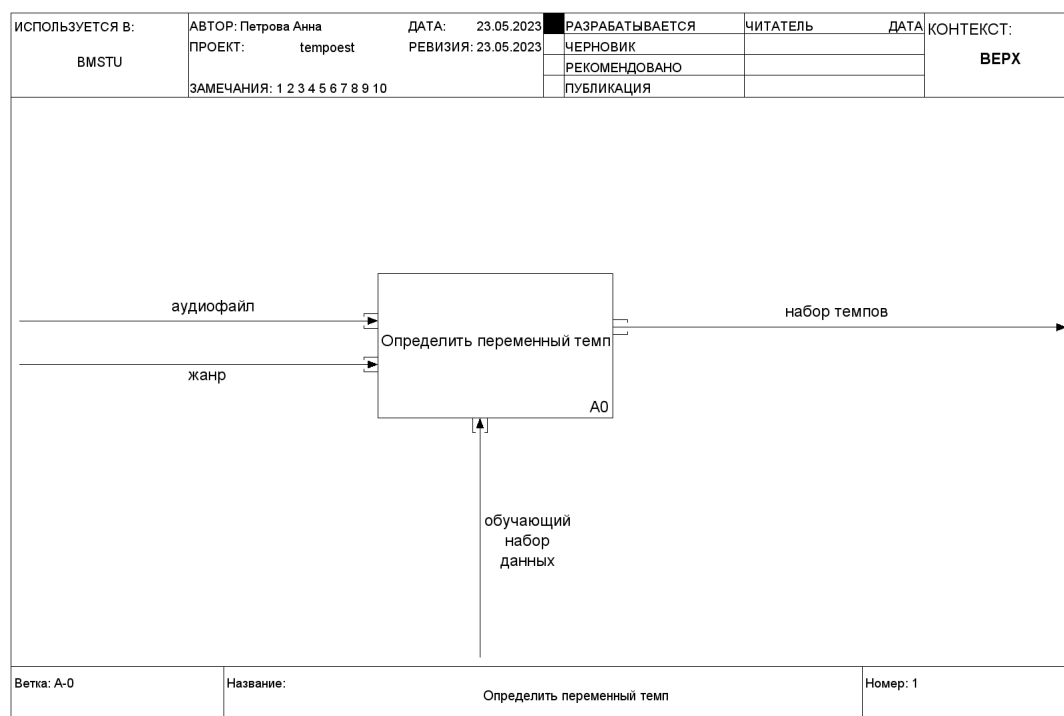


Рис. 2.7 – IDEF0 нулевого уровня

Для адаптации используемого метода к определению переменного темпа музыки анализируемый аудиофайл разделяется на равные фрагменты. Опытным путем было выявлено, что оптимальной длиной фрагментов является 5 секунд. Если взять более короткие фрагменты, то методу начинает не хватать данных для определения темпа из-за слишком маленькой длины аудио. При этом если разделять на более крупные фрагменты, то увеличивается риск пропуска изменений темпа.

При реализации байесовской иерархической модели в качестве априорного распределения темпа анализируемой музыки было выбрано равномерное распределение с границами от минимального возможного темпа до максимального. Границы темпа выбираются на основе указанного жанра музыки. Такое

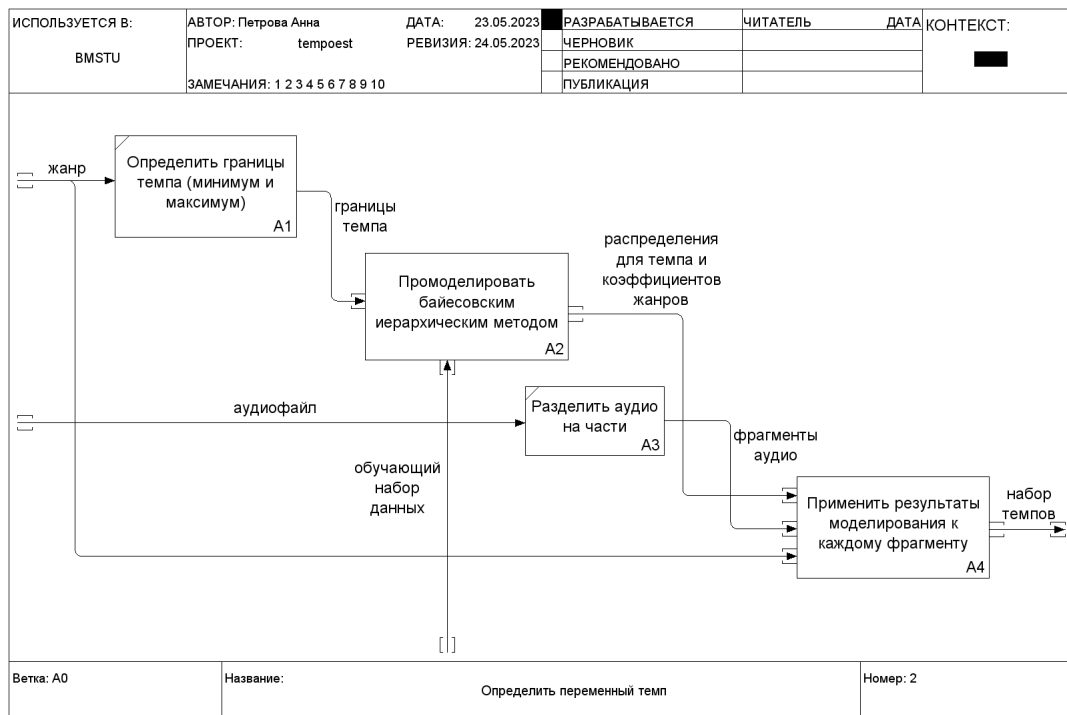


Рис. 2.8 – Определение переменного темпа

распределение было выбрано, так как предполагается, что все темпы в указанных пределах равновероятны для указанного аудиофайла.

Коэффициенты жанров необходимы для корректировки получаемой оценки темпа в зависимости от жанра музыки. Если коэффициент отрицательный, то темп должен быть уменьшен, если положительный – увеличен. В качестве априорного распределения коэффициентов для всех жанров было выбрано нормальное распределение с математическим ожиданием, равным 0, и дисперсией, равной 1.

На основе информации из датасета было выявлено, что распределение темпов различной музыки близко к нормальному. Поэтому в качестве распределения функции правдоподобия темпа было выбрано нормальное распределение. Математическое ожидание для каждого жанра для этого распределения рассчитывается на основе мат. ожидания и дисперсии априорного распределения темпа и априорных распределений коэффициентов жанров по формуле:

$$\mu = \mu_{\text{prior}} + \text{coef} * \sigma_{\text{prior}}. \quad (13)$$

В качестве дисперсии при этом используется дисперсия априорного распределения темпа.

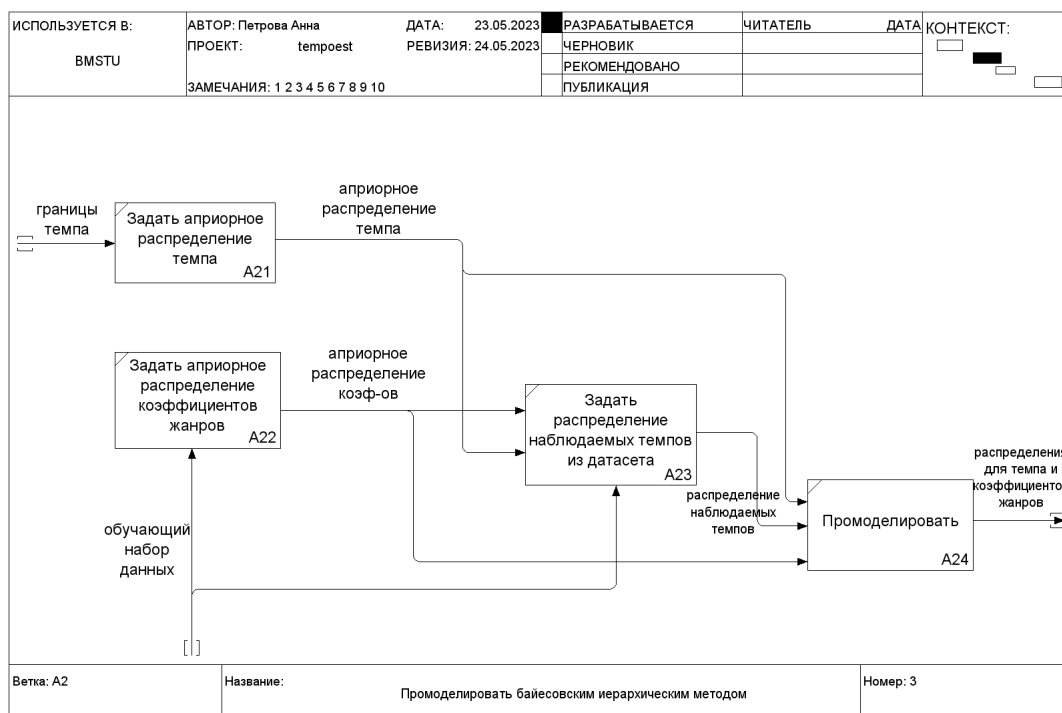


Рис. 2.9 – Байесовское моделирование

Важно заметить, что байесовская модель считается до разделения аудио-файла на фрагменты. В результате получают апостериорные распределения темпа и коэффициентов жанров.

После моделирования для каждого фрагмента аудиофайла рассчитывается примерный диапазон возможных темпов. Этот диапазон «накладывается» на полученное распределение темпа и получается наиболее вероятный темп (максимум плотности распределения) в данном диапазоне (tempo) (рис. 2.11).

После этого среди апостериорных распределений коэффициентов жанров находится распределение для указанного жанра. В этом распределении ищется наиболее вероятный коэффициент (coef). После чего применяется формула:

$$\text{result} = \text{tempo} + \text{coef} * \sigma_{\text{prior}}. \quad (14)$$

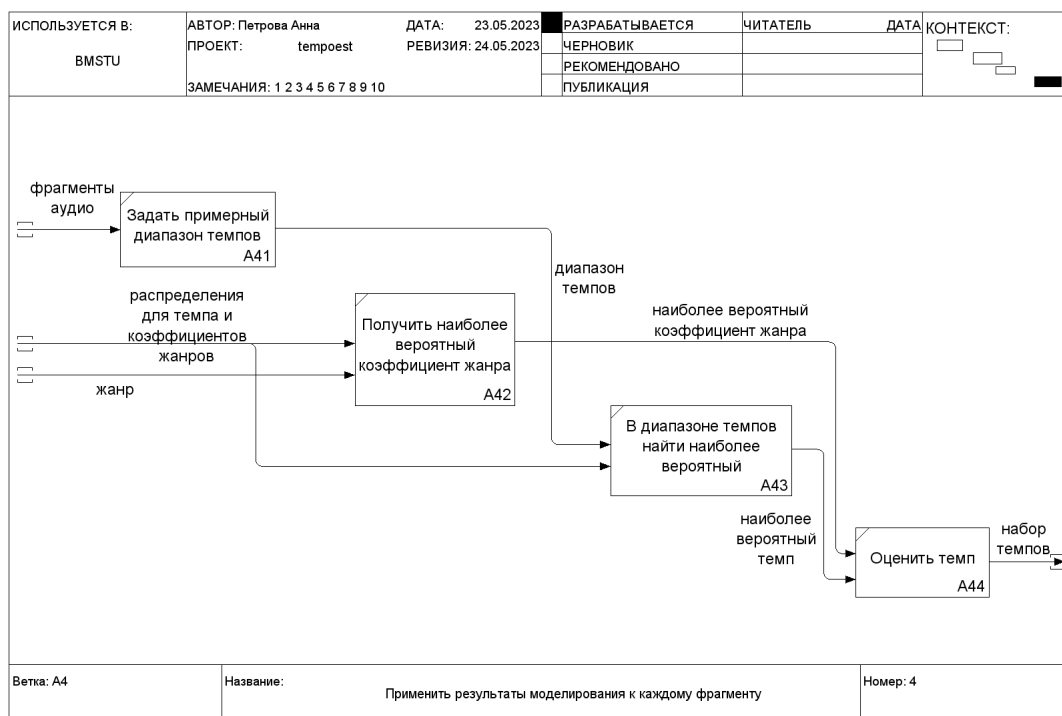


Рис. 2.10 – Применение результатов к фрагментам аудио

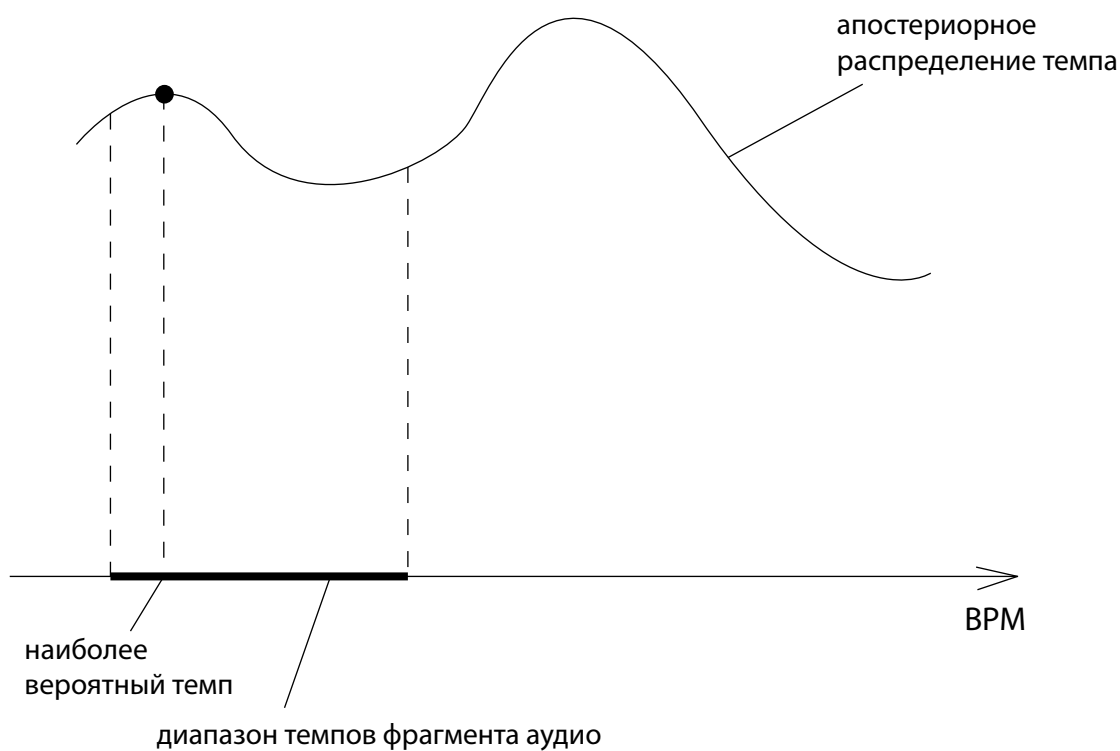


Рис. 2.11 – Нахождение наиболее вероятного темпа



## 2.2 Определение переменного тактового размера

На рисунках 2.12 – 2.15 приведены IDEF0-диаграммы для алгоритма определения переменного ритма (тактового размера) музыки.

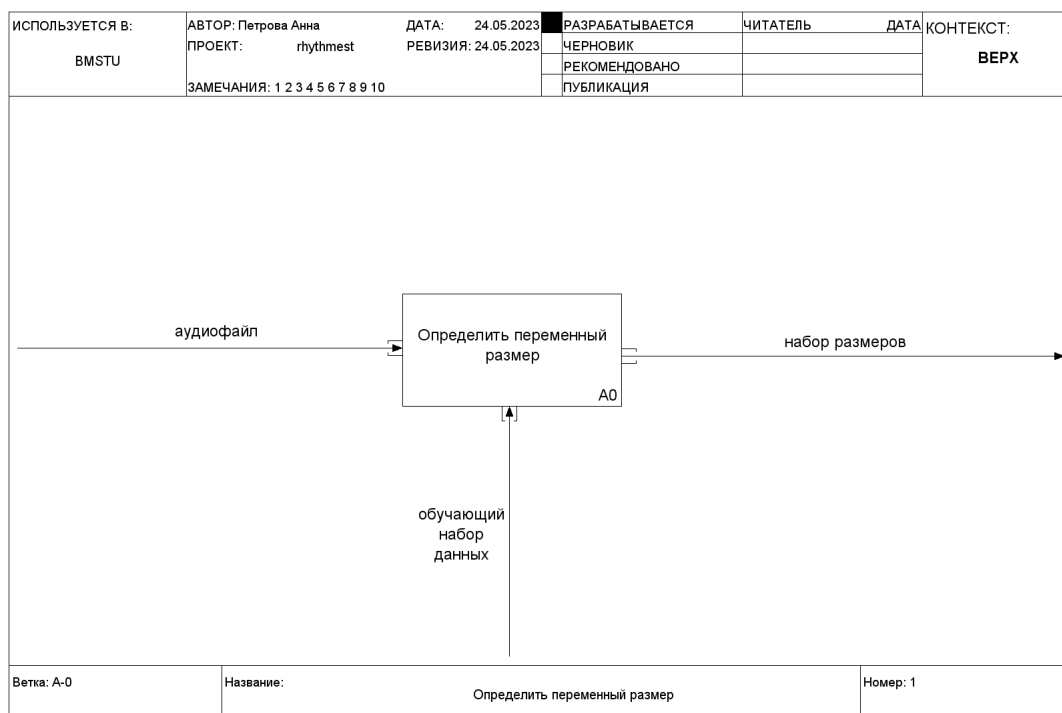


Рис. 2.12 – IDEF0 нулевого уровня

По аналогии с определением темпа в качестве априорного распределения тактового размера в байесовской модели выбрано равномерное распределение. А в качестве распределения функции правдоподобия размера – нормальное распределение с математическим ожиданием, равным мат. ожиданию априорного, и дисперсией, также равной априорной.

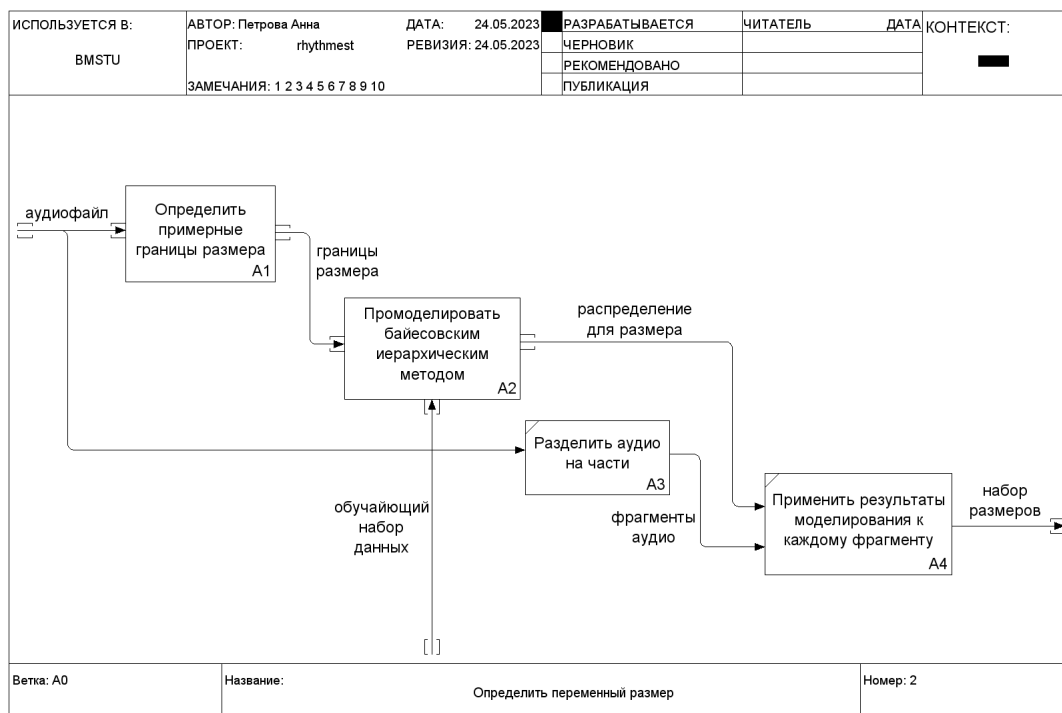


Рис. 2.13 – Определение переменного ритма

Для задания априорного распределения необходимо определить границы размера. Для этого во всем аудиофайле находятся пики амплитуды и последовательность «ударов» (на основе темпа) во временной области. После чего эти две последовательности «накладываются» друг на друга и получается последовательность сильных «ударов» (долей) (предположительные начала тактов). Количество «ударов» между сильными долями и есть тактовый размер. Таким образом определяются примерные границы размеров рассматриваемого аудиофайла.

Остальное аналогично определению темпа. Считается модель. Далее аудиофайл разделяется на фрагменты по 5 секунд, для каждого фрагмента рассчитывается диапазон размеров по принципу, описанному выше. После чего полученное в результате моделирования апостериорное распределение размеров применяется к данному диапазону, и ищется максимум функции плотности, т. е. наиболее вероятный тактовый размер фрагмента.

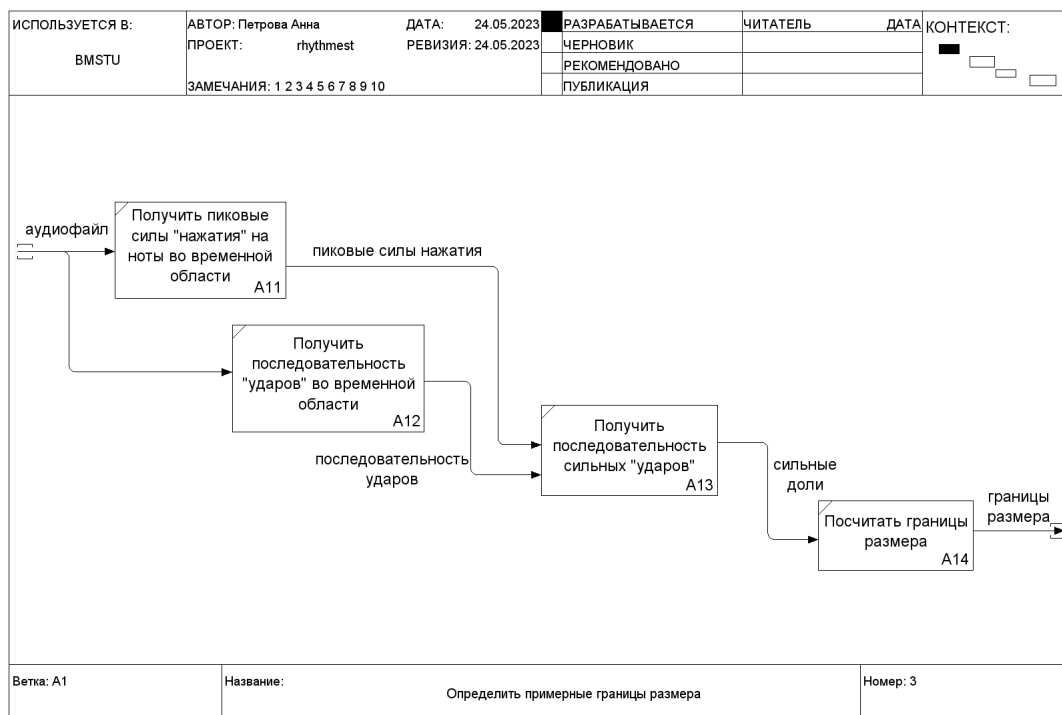


Рис. 2.14 – Определение границ размера

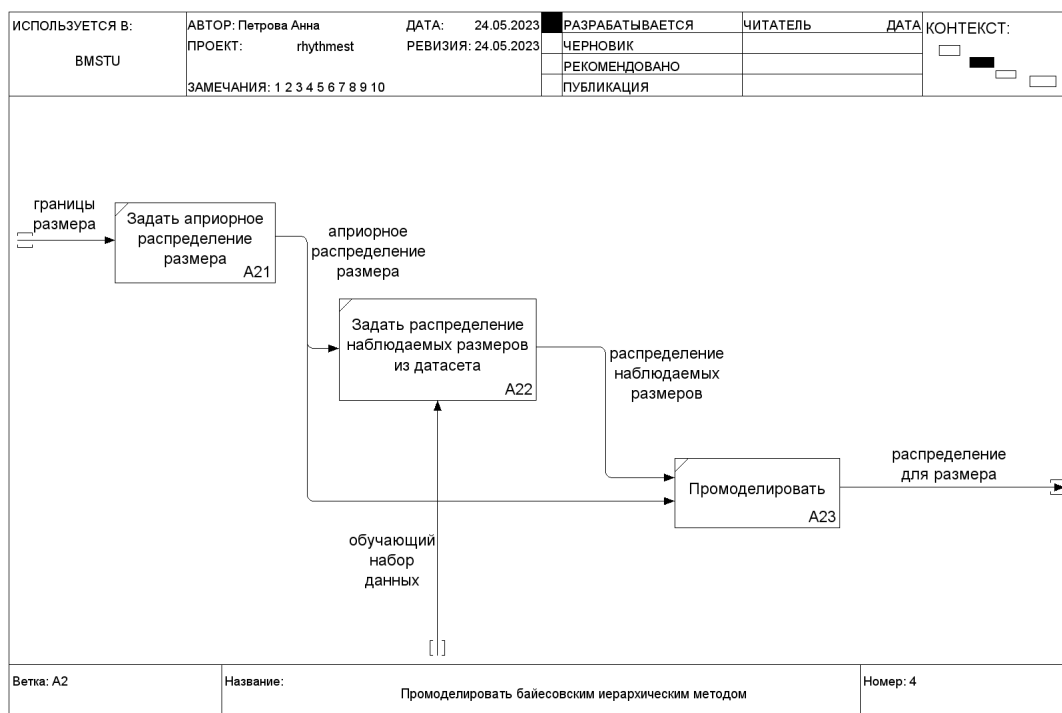


Рис. 2.15 – Байесовское моделирование

## 2.3 Структуры данных

Результаты работы программы будут представлены в виде словарей, где в качестве ключей – время от начала аудиофайла в секундах, а в качестве значений – темпы или тактовые размеры соответственно.

Все распределения, значения из датасета, последовательности «ударов» и пики амплитуды будут представлены в виде одномерных массивов. Массивы «ударов» и пиков амплитуды представляют собой последовательность секунд от начала аудиозаписи, в которые происходят «удары» или пики амплитуды соответственно.

## 2.4 Структура ПО

На рисунке 2.16 показана структура приложения, из которой наглядно видно, как происходит взаимодействие компонентов.

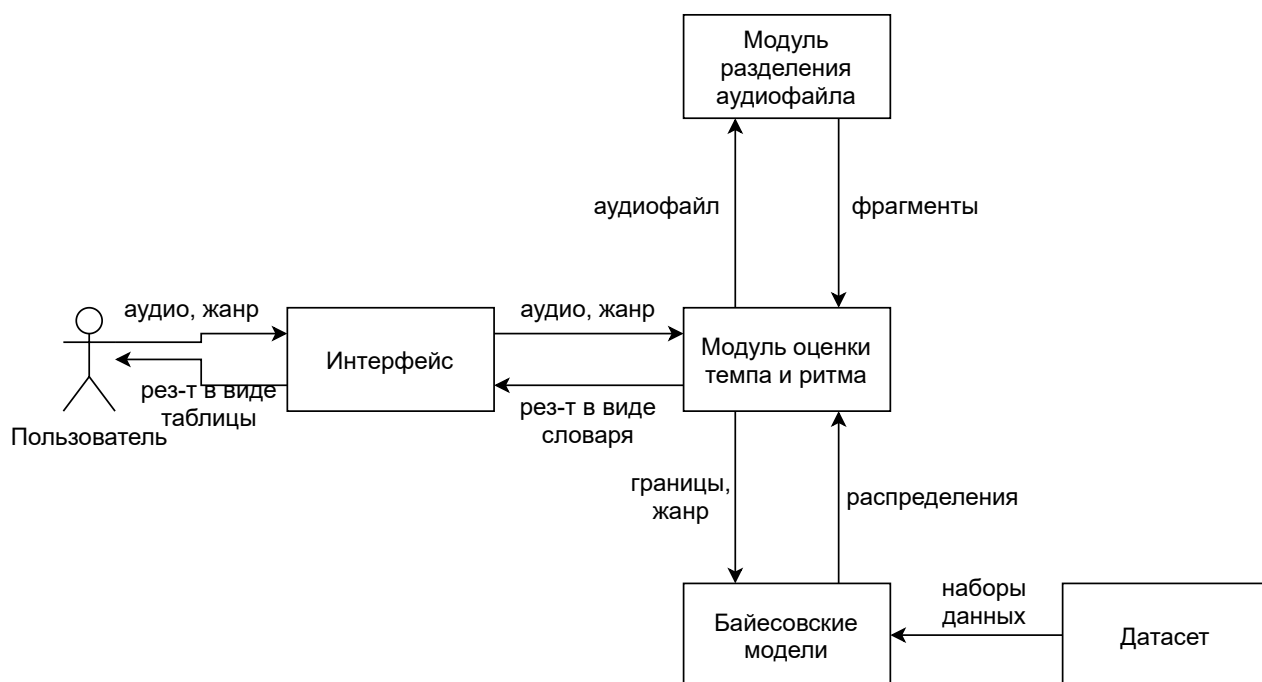


Рис. 2.16 – Структура приложения

Модуль оценки темпа и ритма включает в себя применение результатов байесовского моделирования к фрагментам заданного аудиофайла.

## **Выводы**

На основе теоретических данных, полученных в аналитическом разделе, был разработан метод определения переменного темпа и ритма музыки. Также были приведены IDEF0-диаграммы, схема структуры приложения и описаны основные структуры данных.

## 3 Технологический раздел

### 3.1 Выбор средств программной реализации

В таблице 3.2 произведено сравнение наиболее популярных СУБД, которые могут быть использованы для реализации хранения данных в разрабатываемом программном продукте.

Таблица 3.2 – Особенности различных СУБД

Особенность	MySQL	PostgreSQL	MS SQL Server	Oracle Database
Open source	GNU GPL с открытым исходным кодом	Открытый исходный код	Коммерческая	Коммерческая
Соответствие ACID	Частичное (зависит от версии) [?]	Полное	Полное	Полное
NoSQL / JSON	Поддержка некоторых функций	JSON, NoSQL	JSON, NoSQL	JSON, NoSQL
Поддержка MERGE	Да	Нет	Да	Да

Таким образом, для реализации проекта в качестве СУБД будет использоваться PostgreSQL, так как он является свободно распространяемым и удовлетворяет всем необходимым требованиям в рамках поставленной задачи.

### 3.2 Детали реализации

В приложениях А – Б представлен пример реализации frontend части приложения.

В приложении В представлена реализация соединения с базой данных в соответствии с ролевой моделью.

Помимо этого, на уровне базы данных была реализована процедура, необходимая для удаления реп. базы (приложения Г – Д).

### **3.3 Пользовательский интерфейс**

На рисунках ниже представлен основной интерфейс приложения.

TODO

### **3.4 Тестирование приложения**

#### **Выводы**

В данном разделе были выбраны СУБД, язык программирования и среда разработки.

В качестве СУБД выбрана PostgreSQL, так как она:

- свободно распространяемая;
- полностью соответствует требованиям ACID;
- поддерживает все необходимые в рамках поставленной задачи функции (такие, как вложенные селекты, транзакции, триггеры, процедуры).

В качестве языка программирования был выбран Python, так как он:

- поддерживает работу с PostgreSQL;
- объектно-ориентированный;
- обладает необходимыми расширениями для работы с графическим интерфейсом.

В качестве среды разработки был выбран PyCharm, так как он:

- имеет свободно распространяемую версию;
- содержит множество удобств для написания и отладки кода, а также для работы с СУБД.

Помимо этого, в данном разделе был разработан и протестирован исходный код программы. Программа тестировалась в соответствии с этапами, приведёнными в разделе 2.4. Все тесты были успешно пройдены.

## **ЗАКЛЮЧЕНИЕ**

TODO



## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Benetos E. / Automatic music transcription: challenges and future directions / Benetos E., Dixon S., Giannoulis D., Kirchhoff H., Klapuri A. // Journal of Intelligent Information Systems. – 2013. – С. 407-434.
2. Музыкальный словарь Гроува. // Москва. – 2007. – С. 858.
3. Чехович Д. О. Ритм музыкальный // Большая российская энциклопедия. Москва. – 2015. – Том 28. – С. 541.
4. Cemgil A. T., Desain P., Kappen B. Rhythm quantization for transcription // Computer Music Journal. – 2000. – С. 60-76.
5. Polikar R. The wavelet tutorial // 2-е изд. – 2001. – 67 с.
6. Tzanetakis G., Essl G., Cook P. Audio analysis using the Discrete Wavelet Transform. – 2001.
7. Биккенин Р. Р., Чесноков М. Н. Теория электрической связи. – 2010. – 329 с.
8. Pleshkova S., Panchev K., Bekyarski A. Development of a MIDI synthesizer for test signals to a wireless acoustic sensor network. – 2020.
9. Takeda H., Saito N., Otsuki T. Hidden Markov model for automatic transcription of MIDI signals. – 2002. – С. 428-431.
10. Kübler R. Bayesian Hierarchical Modeling in PyMC3. – 2021.
11. Rouder J. N., Lu J. An introduction to Bayesian hierarchical models with an application in the theory of signal detection // Psychonomic Bulletin & Review. – 2005. – С. 573-604.

12. Nakamura E., Itoyama K., Yoshii K. Rhythm transcription of MIDI performances based on Hierarchical Bayesian Modelling of repetition and modification of musical note patterns. – 2016.
13. Stevens S.S., Volkman J., Newman E.B. A scale for the measurement of the psychological magnitude pitch // Acoustical Society of America. – 1937. – С. 188.
14. Schreiber H., Muller M. A single-step approach to musical tempo estimation using a convolutional neural network // 2018. – С. 98-105.
15. Clevert D.A., Unterthiner T., Hochreiter S. Fast and accurate deep network learning by exponential linear units (elus). – 2015.
16. Ioffe S., Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. – 2015.
17. Spotify Tracks Dataset [Электронный ресурс]. – Режим доступа: <https://www.kaggle.com/datasets/maharshipandya/-spotify-tracks-dataset> (15.05.2023).