Оглавление

[Введение 2](#_Toc532644558)

[1. Исследовательский раздел. 4](#_Toc532644559)

[1.1 Spotify 4](#_Toc532644560)

[Модель совместной фильтрации 5](#_Toc532644561)

[Модель контекстной фильтрации 8](#_Toc532644562)

[Модель, основанная на фильтрации содержимого контента 9](#_Toc532644563)

[1.2 Pandora Music 11](#_Toc532644564)

[Сопоставление песни к песне. 11](#_Toc532644565)

[Взвешенное соответствие 12](#_Toc532644566)

[Фокусное соответствие 14](#_Toc532644567)

[1.3 Amazon Music 14](#_Toc532644568)

[Jaccard Swap Diversity 16](#_Toc532644569)

[Submodular Diversity 16](#_Toc532644570)

[1.4 Выделение функциональных требований в виде Use-case диаграммы UML 19](#_Toc532644571)

[2. Специальный раздел 21](#_Toc532644572)

[1.5 Диаграмма деятельности 21](#_Toc532644573)

[1.6 Диаграмма классов 22](#_Toc532644574)

[1.7 Диаграмма последовательности 23](#_Toc532644575)

[1.8 Компонентная диаграмма 25](#_Toc532644576)

[3. Заключение 26](#_Toc532644577)

[Список литературы 27](#_Toc532644578)

Введение

Наступил момент, когда мобильные приложения глубоко проникли в жизнь человека. Они дополняют возможности и облегчают работу в различных сферах деятельности. Это также касается развлечений, отдыха, ухода за здоровьем и т. д.

Любое мобильное приложение – это визитка, которая вручается постоянным клиентам, и оно должно решать одну из трех задач пользователя:

* увлекательное времяпровождение;
* осуществление доступа в интернет за неимением других способов;
* возможность получить нужную информацию в короткий промежуток времени.

Мобильные пользователи получают доступ к 30 приложениям ежемесячно. Эти 30 приложений составляют примерно одну треть от половины приложений, установленных пользователями на смартфонах. И использование этих приложений - ежедневная привычка, так как люди теперь запускают в среднем по меньшей мере 9 приложений в день по данным отчета App Annie.

App Annie также отметила, что время, проведенное в приложениях, увеличилось за последние два года. Сегодня пользователи в США тратят в среднем более 2 часов и 15 минут в приложениях каждый день, что составляет более одного месяца в году. В Южной Корее, Бразилии, Мексике и Японии этот показатель еще выше, пользователи в среднем тратят около 3 часов в день.

Одним из самых распространенных способов провести время в дороге или дома является прослушивание музыки. Музыка и общество всегда были тесно связаны между собой. Музыка создает и отражает общественные настроения, а в некоторых случаях способствует или препятствует социальным изменениям. Она облегчает общение, выходящее за рамки слов, приобщает к совместному использованию, содействует групповой, культурной, национальной самоидентификации.

Любой современный телефон уже снабжен предустановленным приложением для проигрывания и управления музыкальными файлами. Однако основной функционал стандартных решений не всегда устраивает. С развитием мобильных сетей появилась возможность получать музыкальный контент с помощью потоковой передачи в любой точке земного шара.

Потоковая передача работает путем отправки информации с сервера отдельному пользователю. Песня находится на сервере как необработанный файл. Сырые файлы огромны и детализированы, поэтому их нужно сжимать, чтобы мгновенно перемещать по Интернету. Когда поток достигнет устройства, оно будет декодировать сжатую информацию с помощью приложения или плагина. Поскольку необработанные файлы сжимаются для передачи, качество музыки становится ниже, чем в других форматах, таких как компакт-диски, виниловые записи или даже цифровые загрузки. Например, потоковое качество Spotify по умолчанию - 160 кбит/с. Обычный mp3 обычно составляет 320 кбит/с. Компакт-диски обычно составляют 1411 кбит/с. Успех потоковой передачи музыки доказал, что слушателям важно не качество, а количество и удобство. Огромное количество доступных песен и простота поиска и прослушивания — это настоящая привлекательность сервисов.

**Целью** данной работы является проектирование мобильного приложения-сервиса воспроизведения музыки.

При написании курсового проекта использовались **метод теоретического исследования** (анализ) и **эмпирический метод** (изучение источников литературы).

**Актуальность** курсового проекта состоит в том, что с ростом количества пользователей мобильных смартфонов, все больше и больше слушателей уклоняются от физических записей в пользу онлайн-музыкальных сервисов.

**Практическая значимость** проекта заключается в том, что музыкальный сервис обеспечивает пользователям быстрое получение высококачественного музыкального контента по сети интернет с предоставлением рекомендационных плейлистов, а также упрощает процесс поиска треков.

1. Исследовательский раздел.

Потоковое мультимедиа— это мультимедиа, которое непрерывно получает пользователь от провайдера потокового вещания. Это понятие применимо как к информации, распространяемой через телекоммуникации, так и к информации, которая изначально распространялась посредством потокового вещания (например, радио, телевидение) или не потоковой (например, книги, видеокассеты, аудио CD).

**Протоколы потокового вещания**

Датаграмные протоколы, такие как User Datagram Protocol (UDP), отправляют поток медиаинформации как поток отдельных маленьких пакетов. Он прост и эффективен; в то же время, в спецификации протокола нет гарантии доставки данных получателю. Это очень сильно затрудняет поиск и исправление получаемых данных принимающим информацию приложением. При потере данных поток может быть отключен.

Протоколы RTSP, RTP и RTCP специально разрабатывались для передачи мультимедийной информации по сети. Последние два построены на основе UDP.

Надежные протоколы, такие как TCP, гарантируют корректность получаемых данных клиентов потокового вещания. Однако при большом количестве ошибок при соединении/подтверждении получаемой информации передаваемая информация может стать неактуальной. Это также может вызвать значительные задержки при передаче информации на время, затраченное на пересылку поврежденной информации. Одним из решений данной проблемы является буферизация информации на стороне клиента.

Протоколы Unicast отправляют отдельную копию данных каждому клиенту. Unicast подходит для большинства пользователей сети Интернет, но сильно затрудняет масштабирование сервера для бо́льшего количества клиентов.

При широковещательной передаче одна копия данных передается всем клиентам сервера

Протоколы Multicast разработаны для снижения нагрузки с серверов на подключения/ширину канала при получении потокового мультимедиа большим количеством клиентов. Эти протоколы отсылают одну порцию данных целой группе клиентов. В зависимости от типа сетевой инфраструктуры, групповая передача данных может быть доступна, а может и не быть. Одним из потенциальных недостатков групповой передачи является отсутствие возможности реализовать функцию видео по запросу. Непрерывное вещание потоковой информации также делает невозможным управление воспроизведением пользователем. Однако, эта проблема может быть решена внедрением в сеть передачи данных кэширующих серверов и буферизирующего принимаемый поток программного обеспечения.

Multicast позволяет передавать один поток информации группе клиентов по сети. Одной из проблем при реализации подобной схемы потокового вещания является корректная настройка маршрутизаторов для передачи широковещательных пакетов из одного сегмента сети в другой. Если организация, предоставляющая потоковое вещание, имеет контроль над сетью между сервером и клиентами (например, в образовательной, правительственной или корпоративной сети), то протоколы маршрутизации, такие как IGMP и PIM, могут быть использованы для доставки мультимедиа нескольким клиентам из различных сегментов LAN.

Протоколы P2P могут использоваться при распространении предварительно записанной мультимедиа между компьютерами. Это снимает нагрузку с сервера, однако сеть передачи данных между сервером и одним из клиентов становится узким местом данного варианта реализации потокового вещания информации.

**Сжатие аудиоданных**

Сжатие (компрессия) аудиоданных представляет собой процесс уменьшения скорости цифрового потока за счет сокращения статистической и психоакустической избыточности цифрового звукового сигнала.

**Сжатие с без потерь**

Методы сокращения статистической избыточности аудиоданных также называют сжатием без потерь, а, соответственно, методы сокращения психоакустической избыточности — сжатием с потерями.

Сокращение статистической избыточности основано на учете свойств самих звуковых сигналов. Она определяется наличием корреляционной связи между соседними отсчетами цифрового звукового сигнала, устранение которой позволяет сокращать объем передаваемых данных на 15…25 % по сравнению с их исходной величиной. Для передачи сигнала необходимо получить более компактное его представление, что возможно осуществить с помощью ортогонального преобразования. Важными условиями применения такого метода преобразования являются:

* возможность восстанавливать исходный сигнал без искажений
* способность обеспечивать наибольшую концентрацию энергии в небольшом числе коэффициентов преобразования
* быстрый вычислительный алгоритм

Этим требованиям отвечает модифицированное дискретно-косинусное преобразование (МДКП).

Уменьшить скорость цифрового потока позволяют методы кодирования, учитывающие статистику звуковых сигналов, например, вероятности появления уровней разной величины. Одним из таких методов является код Хаффмана, где наиболее вероятным значениям сигнала приписываются более короткие кодовые слова, а значения отсчетов, вероятность появления которых мала, кодируются кодовыми словами большей длины. Именно в силу этих двух причин в наиболее эффективных алгоритмах компрессии цифровых аудиоданных кодированию подвергаются не сами отсчеты звукового сигнала, а коэффициенты МДКП. Подобные методы применяются при архивации файлов.

**Сжатие с потерями**

Сжатие аудиоданных с потерями основывается на несовершенстве человеческого слуха при восприятии звуковой информации. Неспособность человека в определенных случаях различать тихие звуки в присутствии более громких, называемая эффектом маскировки, была использована в алгоритмах сокращения психоакустической избыточности. Эффекты слухового маскирования зависят от спектральных и временных характеристик маскируемого и маскирующего сигналов и могут быть разделены на две основные группы:

* частотное (одновременное) маскирование
* временное (неодновременное) маскирование

Эффект маскирования в частотной области связан с тем, что в присутствии больших звуковых амплитуд человеческое ухо нечувствительно к малым амплитудам близких частот. То есть, когда два сигнала одновременно находятся в ограниченной частотной области, то более слабый сигнал становится неслышимым на фоне более сильного.

Маскирование во временной области характеризует динамические свойства слуха, показывая изменение во времени относительного порога слышимости (порог слышимости одного сигнала в присутствии другого), когда маскирующий и маскируемый сигналы звучат не одновременно. При этом следует различать явления послемаскировки (изменение порога слышимости после сигнала высокого уровня) и предмаскировки (изменение порога слышимости перед приходом сигнала максимального уровня). Более слабый сигнал становится неслышимым за 5 − 20 мс до включения сигнала маскирования и становится слышимым через 50 − 200 мс после его включения.

Наилучшим методом кодирования звука, учитывающим эффект маскирования, оказывается полосное кодирование. Сущность его заключается в следующем. Группа отсчетов входного звукового сигнала, называемая кадром, поступает на блок фильтров, который разделяет сигнал на частотные поддиапазоны. На выходе каждого фильтра оказывается та часть входного сигнала, которая попадает в полосу пропускания данного фильтра. Далее, в каждой полосе с помощью психоакустической модели, анализируется спектральный состав сигнала и оценивается, какую часть сигнала следует передавать без сокращений, а какая лежит ниже порога маскирования и может быть переквантована на меньшее число бит. Для сокращения максимального динамического диапазона определяется максимальный отсчет в кадре и вычисляется масштабирующий множитель, который приводит этот отсчет к верхнему уровню квантования. Эта операция аналогична компандированию в аналоговом вещании. На этот же множитель умножаются и все остальные отсчеты. Масштабирующий множитель передается к декодеру вместе с кодированными данными для коррекции коэффициента передачи последнего. После масштабирования производится оценка порога маскирования и осуществляется перераспределение общего числа битов между всеми полосами.

Очевидно, что после устранения психоакустической избыточности звуковых сигналов их точное восстановления при декодировании оказывается уже невозможным. Методами устранения психофизической избыточности можно обеспечить сжатие цифровых аудиоданных в 10 − 12 раз без существенных потерь в качестве.

Существует множество различных стриминговых сервисов, реализующих воспроизведение музыкальных файлов с возможностью высококачественной потоковой передачи, а также использующие различные алгоритмы формирования рекомендательных плейлистов. В данном разделе мы рассмотрим такие виды приложений, как Spotify, Pandora Music, Amazon Music, поддерживаемые ими платформы, высоту битрейта, количество передаваемых данных, а также функциональные возможности и методы создания индивидуализированных плейлистов.

1. Spotify

Одним из самых популярных стриминговых музыкальных сервисов является Spotify, имеющий на 2018 год 170 миллионов зарегистрированных пользователей. В настоящее время Spotify предоставляет доступ к более чем 30 миллионам композиций. Пользователи могут осуществлять поиск по исполнителям, альбомам, лейблам, жанрам, а также имеют доступ ко многим крупным и независимым лейблам.

Пользователи могут создавать плейлисты, делиться ими с другими пользователями, а также совместно редактировать их. В настоящее время в Spotify около 2-х миллиардов плейлистов, созданных как пользователями, так и музыкантами.

Spotify работает на самых распространенных платформах: Android, Wear OS, IOS, Windows Phone. Также, если пользователь имеет персональный компьютер, либо ноутбук, сервис поддерживает OS X и Windows. Однако, большинство пользователей склонны использовать свой мобильный телефон для прослушивания Spotify на ходу.

Spotify – платный сервис. Цена подписки – 9,99 долларов в месяц. Также существует возможность пользоваться этим стриминг-сервисом бесплатно. Однако нужно отметить, что подписчики получают целый ряд преимуществ:

* прослушивание музыки в качестве 320 Кбит/с (при бесплатном использовании – 160 Кбит/с);
* доступ ко многим интересным и полезным опциям;
* отсутствие рекламы (в бесплатном режиме – 15-20 секунд рекламы через 3-4 композиции).

Музыкальное приложение обеспечивает 4 возможных битрейта вещания, в формате Ogg Vorbis, который наделяет музыкальные файлы достаточно высоким качеством звучания, однако, является форматом сжатия с потерями.

* Нормальное качество - 96 килобит/с, что позволяет передавать 40 мегабайт в час
* Высокое качество - 160 килобит/c, что позволяет передавать 70 мегабайт в час
* Экстремальное качество - 320 килобит/с, что позволяет передавать 150 мегабайт в час
* Автоматическое качество - этот параметр настраивает качество потоковой передачи на основе интернет-соединения телефона.

Основной причиной популярности Spotify является рекомендательный сервис, который предлагает пользователям новую музыку и исполнителей, учитывая их предпочтения и историю прослушанной музыки.

В качестве рекомендательного движка Spotify использует комбинацию трех уже существующих различных моделей, такая совокупность позволяет сервису создавать уникальные персонализированные плейлисты. Идеальные музыкальные рекомендации максимизируют удовлетворение пользователя от сервиса. Три вида рекомендательных моделей:

* Модель совместной фильтрации
* Модель контекстной фильтрации
* Модель, основанная на фильтрации контента

Модель совместной фильтрации

Данная модель является наиболее распространенным методом для рекомендательной системы. Она использует технику K-ближайших соседей. Музыкальный вкус пользователей рассчитывается как расстояние между различными пользователями. Метод ищет соседних пользователей, которые разделяют аналогичных интерес к музыке и рекомендуют контент на основе этих данных. Следовательно, совместная фильтрация предсказывает предпочтения пользователей, основываясь отношении пользователь-элемент. В повседневной жизни это можно сравнить с дружеским советом.

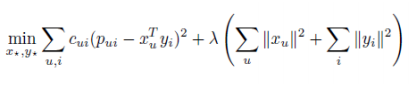
Методы совместной фильтрации работают путем построения матрицы M c n-элементами и m-пользователями, которые содержат взаимодействие, например рейтинг, проигрывание. Каждая строка представляет пользователя, а каждый столбец элемент. Этот тип представления имеет чрезвычайно большие размеры, поскольку элемент является m-мерным вектором, а пользователь n-мерным вектором. Кроме того, существует избыточность в измерениях, так как пользователи и элементы могут быть разделены на группы со схожими предпочтениями.



1. - Матрица пользователь - элемент

* Пользователь D и пользователь E имеют схожие музыкальные вкусы, обоим нравятся треки V, W и Z, а также не нравится трек X
* Пользователь E еще не послушал трек Y, но пользователю D нравится трек Y
* Spotify будет рекомендовать песню Y пользователю E.

Технический подход совместной фильтрации будет объяснен с уравнением ниже:



1. Функция минимизации стоимости

Прежде всего, двоичная переменная “pui” представляет собой предпочтение пользователя “u” элементу “i”, где пользователю нравится элемент, когда он его прослушал и когда “pui” равен единице. В противном случае пользователь никогда не слушал элемент раньше. Следующая переменная “сui” измеряет доверие в зависимости от “pui”. Однако цель состоит в том, чтобы найти вектор “xu” для пользователя “u” и “yi” для элемента “i”, который будет учитывать предпочтения пользователя.



1. Вектор элемента



1. Вектор пользователя

Расчет аналогичен методу факторизации матрицы. Последний термин предназначен для упорядочения модели, чтобы избежать чрезмерной подгонки тренировочных данных. На каждом этапе вычислений значение стоимости функции, безусловно, уменьшается. Совместную фильтрацию далее можно разделить на три типа:

* Фильтрация на основе памяти
* Метод на основе моделей
* Гибридный метод

Фильтрация на основе памяти генерирует прогноз, используя базу данных пользователь-элемент. Каждый пользователь находится в группе людей со схожими интересами. Прогнозирование предпочтений пользователя на новые элементы вычисляется как совокупность рейтингов его соседей.

Метод на основе моделей использует алгоритмы машинного обучения и интеллектуального анализа данных для обучения и моделирования предпочтений пользователей на основе сбора рейтингов. Затем модель используется для прогнозирования тестовых и реальных данных.

Гибридный метод формирует прогнозы, комбинируя алгоритмы на основе памяти и на основе модели или комбинируя с другими техниками рекомендаций.

Модель контекстной фильтрации

Контекстная фильтрация использует культурную информацию для того, чтобы описать и охарактеризовать элементы, которая затем используется для вычисления сходства исполнителя или песни. Культурная информация может включать метаданные, такие как жанр, эмоции, семантические особенности, сходство, метки и теги. Фильтрация основана на данных, собранных с помощью методов веб-интеллектуального анализа данных или данных, полученных из совместных пометок.

Методы веб-майнинга направлены на обнаружение интересной и полезной информации с помощью анализа веб-контента и его использования. Выделяют три категории веб-майнинга. Интеллектуальный анализ веб-содержимого включает в себя анализ текста, гипертекста, полу структурированных данных и мультимедиа. Интеллектуальный анализ веб-структуры анализирует структуры ссылок в Интернете. Он основан на топологии гиперссылок. Интеллектуальный анализ веб-использования анализирует журналы сеансов и взаимодействия пользователей. Информация, полученная с помощью интеллектуального анализа веб-содержимого, может использоваться для получения сходства элементов. Информация, полученная с помощью интеллектуального анализа, может использоваться для получения пользовательских привычек и предпочтений, а также сходства элементов на основе совпадений в журналах сеансов.

Совместные пометки (также известные как теги) - аннотация веб-контента с помощью тегов, которые являются свободно выбранными ключевыми словами. Рекомендательные системы могут использовать данные совместной пометки для получения сходства элементов или пользователей.

При совместных пометках теги элементов могут быть представлены в виде кортежей <пользователя, элемента, тега>, которые соответствуют трехмерной матрице или тензору.

Два основных подхода к использованию информации совместной пометки, чтобы вычислить схожесть элемента и пользователя:

* разложить 3-мерный тензор в три двумерные матрицы (пользователь-тег, элемент-тег, пользователь-элемент)
* непосредственно использовать 3-мерный тензор

Используя аналогичный матричный алгоритм совместной фильтрации, Spotify может определить, какие песни и исполнители похожи и в дальнейшем могут быть рекомендованы вместе.

Модель, основанная на фильтрации содержимого контента

Фильтрация на основе содержимого собирает сведения, описывающие содержимое элемента, и рекомендует элементы, похожие на элементы, которые нравятся пользователю. Элементы обычно представлены N-мерными векторами признаков, и признаки, которые они содержат, могут быть собраны автоматически, например, путем извлечения признаков из звукового сигнала, или использовать ручные аннотации экспертами по предметной области, т. е. музыкантами. В качестве данной модели Spotify используется аудио-анализ.

Первые два метода, которые Spotify использует для рекомендации музыки, не используют никакого анализа самой музыки. Аудио анализ — это третий и последний способ, которым Spotify улучшает свою уже тщательную модель рекомендаций. Этот метод, как Spotify может рекомендовать совершенно новую музыку для своих слушателей. В то время как совместная и контекстная фильтрация полагаются на других людей, слушающих и находящих контент на определенном исполнителе или треке, аудио анализ может найти подобную музыку на основе того, как звучит трек.

Используя технологию, основанную на искусственном интеллекте, аналогичному распознаванию лиц, Spotify может анализировать отдельные музыкальные файлы с помощью сверточных нейронных сетей (форма глубокого обучения). Анализируя маркеры в аудиофайлах, которые идентифицируют различные темы в музыке, Spotify может группировать похожие музыкальные стили и исполнителей, используя сотни фильтров, таких как темп, громкость или определенные музыкальные клавиши, и использовать эти данные для улучшения своей модели рекомендаций. Ниже представлен 30-секундный клип” Around The World " от Daft Punk.

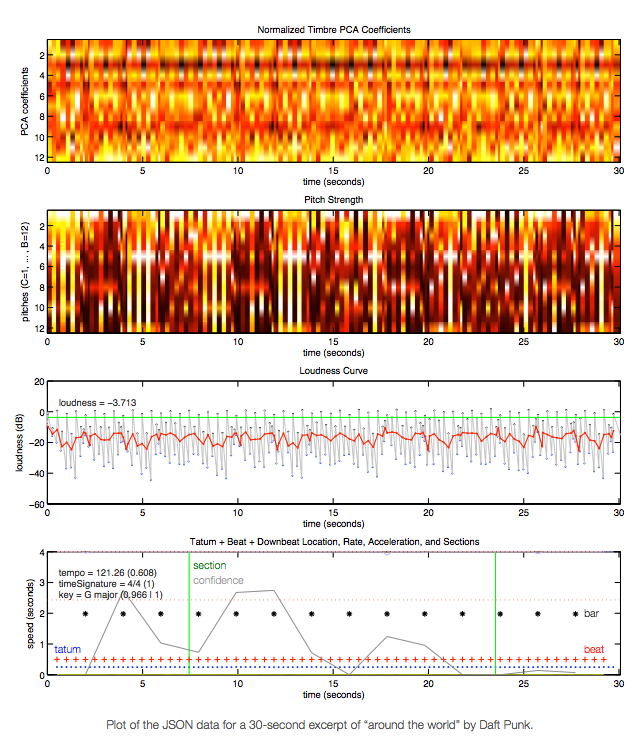


Рисунок 1. Анализ песни Daft Punk

Spotify анализирует отрывки из каждой песни с помощью фильтров, которые подбирают гармонический контент и определяют предикторы звучания песен.

Благодаря такой стратегии, Spotify создает плейлист Discover Weekly, который строит индивидуальный список 30-ти песен каждую неделю для отдельного пользователя.

В будущем Spotify будет анализировать внутренние физические показатели пользователя, таких как количество ударов сердца в минуту, а также датчик движения, на основании которых в дальнейшем будет формироваться еще один метод рекомендаций аудиозаписей и музыкальных плейлистов, соответствующие настроению и движению пользователя.

1. Pandora Music

Одним из основных конкурентов Spotify на рынке музыкальных сервисов является Pandora Music. На 2018 год по данным Statista.com в Pandora Music зарегистрировано около 80 миллионов пользователей. Сервис обеспечивает возможности поиска и воспроизведения песен, альбомов и плейлистов по запросу, создание плейлистов, скачивание музыки для прослушивания в автономном режиме.

Кроссплатформенность сервиса разрешает использовать Pandora Music на Android, IOS, Windows Phone, Wear OS, watch OS, Windows, NOOK, kindle fire, Fitbit.

Приложение позволяет транслировать аудиофайлы в следующих битрейтах, используя формат сжатия AACPlus format:

* Низкое - 64Кбит/c, потребление трафика незначительное.
* Среднее -128Кбит/c, потребляя 57,6 Мбит каждый час и 0,5 Гбит свыше 9 часов.
* Высокое - 192Кбит/с, потребляя 86,4 Мбит каждый час и 0,7 Гбит свыше 9 часов.

Pandora Music аналогично Spotify использует модели совместной фильтрации, контекстной фильтрации и фильтрации контента, однако, отличительной особенностью сервиса является его популярная модель рекомендаций “Music Genome Project”.

Сопоставление песни к песне.

**“Music Genome Project**”— это база данных песен. Каждая песня описывается набором нескольких характеристик или «генов», которые собираются в логические группы, называемые «хромосомами». Набор хромосом составляет геном. Одной из этих основных групп в геноме является хронология анализа музыки.

Соответствующий движок эффективно вычисляет расстояние между исходной песней и другими песнями в базе данных, а затем сортирует результаты, чтобы получить регулируемое количество ближайших совпадений.

Каждый ген можно рассматривать как ортогональную ось многомерного пространства и каждую песню как точку в этом пространстве. Песни, которые геометрически близки друг к другу, являются «хорошими» музыкальными парами.

Песня «S» представляется музыкальным вектором, содержащим приблизительно 150 генов. Каждый ген соответствует характеристике музыки, например, пол вокалиста, уровень искажения на электрогитаре, тип фонового вокала и т. Д. В предпочтительном варианте рок и поп-песни имеют 150 генов, рэп-песни имеют 350, а джазовые песни - приблизительно 400. В других жанрах музыки, таких как классика, есть 300-500 генов. Для получения полезных результатов система зависит от достаточного количества генов. Каждый ген «s» этого вектора представляет собой число от 0 до 5. Дробные значения разрешены, но ограничены половинными целыми числами.

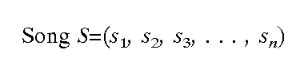


Рисунок . *Вектор песни*

Простое расстояние между любыми двумя песнями «S» и «T» в n-мерном пространстве можно рассчитать следующим образом:



Рисунок . *Расстояние между песнями*

Поскольку монотонная функция квадратного корня используется при вычислении всех этих расстояний, вычисление функции не обязательно. Вместо этого изобретение использует квадрат расстояния в сравнении песен. Принимая это, расчет расстояния пишется в упрощенном виде как:



Рисунок . *Расстояние между песнями после упрощения*

Взвешенное соответствие

Поскольку не все гены одинаково важны, чтобы установить хорошее соответствие, расстояние лучше рассчитывается в виде суммы, взвешенной в соответствии с индивидуальными данными каждого гена, который первоначально устанавливается путем эмпирической работы, например, музыкальной группой, которая анализирует песни. Вектором утяжеления можно манипулировать различными способами, которые влияют на общее поведение системы.

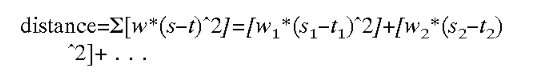


Рисунок . *Расстояние между песнями с соответствующими генами*

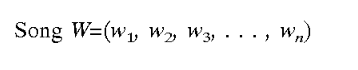


Рисунок *. Взвешенный вектор песни*

Данные, представленные многими из отдельных генов нелинейные. Другими словами, расстояние между значениями 1 и 2 не обязательно совпадает с расстоянием между значения 4 и 5. Введение функций масштабирования f(x) может подстраиваться под эту нелинейность. Добавление этих масштабирования функции изменяет соответствующую функцию:

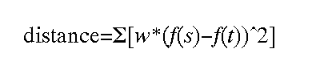


Рисунок . *Расстояние после введения функции масштабирования*

Существует практически неограниченное количество масштабирующих функций, которые могут быть применены к значениям гена для достижения желаемого результата.

В качестве альтернативы можно обобщить функцию разности квадрата на любую функцию, которая работает с абсолютной разницей двух значений гена. Общая функция расстояния:



Рисунок . Обобщение функции расстояния

Фокусное соответствие

Фокусное соответствие позволяет конечному пользователю системы, оснащенной механизмом согласования, контролировать соответствующее поведение системы. Фокусные черты могут быть использованы для повторного взвешивания соответствия песни и уточнить поиск подходящих песен, чтобы включить или исключить выбранные черты фокуса.

Фокусные черты - отличительные аспекты песни. Когда пользователь вводит исходную песню в систему, ее геном исследуется, чтобы определить, какие фокусные признаки были определены музыкальными аналитиками.

Конечный пользователь может выбрать эти признаки, чтобы сфокусироваться на любом из представленных признаков. Когда выбран признак или количество признаков, соответствующий движок изменяет свой вектор взвешивания, чтобы более точно соответствовать выбору. Это делается путем увеличения веса генов, которые являются специфичными для выбранного фокусного признака, и путем изменения значений конкретных генов, которые имеют отношение к черте. Результирующие песни будут очень похожи на исходную песню в выбранных признаках.

Используемые Pandora методы для рекомендации музыки являются очень успешными. Их система классификации обеспечивает мощный способ показать сходство между многими типами музыки и позволяет людям легко находить новые песни с небольшими усилиями.

1. Amazon Music

Завершает тройку основных лидеров рынка сервис Amazon Music. Amazon Music является платформой потоковой передачи музыки и музыкальным онлайн-магазином, управляемым Amazon.com.

При запуске Amazon предложила «более 2 миллионов песен от более чем 180 000 исполнителей и более 20 000 лейблов, включая EMI Music и Universal Music Group», только для клиентов, находящихся в Соединенных Штатах. В декабре 2007 года Warner Music объявила, что предложит свой каталог на Amazon MP3, а в январе 2008 года Sony BMG последовала их примеру. Текущий каталог составляет 29,1 миллиона песен.

Amazon Music поддерживает следующие платформы Windows, Mac OS, iOS, Android, Fire OS, устройства Alexa и Android TV.

В свою очередь сервис предлагает воспроизводить аудиозаписи в формате AAC и MP3 c битрейтом вещания 256кбит/c, используя формат сжатия звука AACPlus.

Компания Amazon повсеместно использует алгоритмы совместной фильтрации, фильтрации контента, контекстной фильтрации, рассмотренные ранее, во всех своих сервисах, в том числе и в Amazon Music.

Тем не менее, для того чтобы разнообразить рекомендации музыкального приложения, были проведены эксперименты по внедрению алгоритма Submodular Diversity, а также сравнение с Jaccard Swap Diversity, которые будут рассмотрены далее.

Необходимость использования алгоритма Submodular Diversity обусловлена тем, что в отличие от других цифровых и физических продуктов, музыкальный контент имеет тенденцию иметь явные кластеры. Альбом содержит несколько песен, каждая из которых имеет одну и ту же обложку альбома, название и описание. Кроме того, песни в одном и том же альбоме, как правило, принадлежат к одному и тому же жанру и обычно воспроизводятся вплотную. Из-за схожих особенностей рекомендательные системы обычно одинаково оценивают песни одного и того же альбома.

Обычно музыкальные рекомендации отображаются в виде списка, что упрощает просмотр пользователями на настольных, мобильных или голосовых устройствах. Наивное ранжирование рекомендуемых песен по их индивидуальному баллу приводит к снижению удовлетворенности пользователей, потому что подобные песни рекомендуются подряд. Дублирование приводит к устареванию пользовательского опыта и к потере возможностей для поставщиков музыкального контента, желающих продемонстрировать широту выбора контента. Это влияние усиливается на устройствах с ограниченными возможностями взаимодействия. Например, смартфоны имеют ограниченное пространство на экране, и, как правило, более обременительно перемещение между экранами или даже при прокручивании страницы вниз.

Помимо фактора точности, существуют другие факторы, влияющие на качество рекомендаций. К таким факторам относятся разнообразие, новизна и случайность, которые дополняют и часто противоречат точности. Данный алгоритм позволяет разнообразить контент, а также уменьшить избыточность рекомендаций.

Jaccard Swap Diversity

Для каждой музыкальной рекомендации используется метод диверсификации, чтобы сгенерировать набор взвешенных соответствующих пояснительных элементов. Пояснительные пункты являются скрытыми признаками кандидата, порожденными содержанием и поведенческими особенностями. Затем вычисляется расстояние Джакарда между двумя музыкальными рекомендациями, применяя уравнение, описанное ниже, к основным поясняющим элементам.

Расстояние Джаккарда измеряет несходство между двумя конечными наборами выборок A и B:



Рисунок . *Расстояние Джаккарда*

Затем формируется список из K = 40 рекомендаций, используя алгоритм перестановок, который итеративно максимизирует верхнее K парное расстояние Джаккарда, обусловленное релевантностью оценки.

Submodular Diversity

Для улучшения взаимодействия с клиентами формулируется набор и ранжирование разнообразного музыкального подмножества. Это задача субмодульной оптимизации. Субмодульные функции характеризуются убывающим свойством возврата. Для множества S, подмножества A ⊆ S, элементов x, y ∈ S и субмодульной функции f: {0, 1}^ S → R имеем:



Рисунок . Субмодульная оптимизация

Весь музыкальный контент разделяется на категории C, в соответствии с теми же атрибутами контента. Каждый выбранный кандидат сопоставляется с несколькими категориями в зависимости от его содержания и особенностей поведения. Каждая категория c имеет свою собственную субмодульную функцию fc. Чтобы убедиться, что кандидат не вносит больше, чем его персональный показатель используется формула:



Рисунок *. Формула ограничения показателей кандидата*

Затем проводится диверсификация, максимизируя сумму ρ всех функций категории fc:

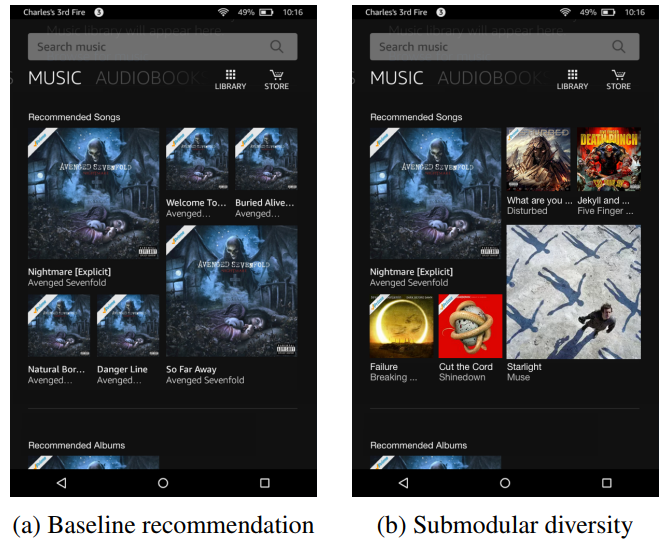


Рисунок . *Максимизация суммы всех категорий*

Почти оптимальное решение достигается с помощью итеративной процедуры:



Рисунок . *Итеративный вариант процедуры максимизации*



1. Результаты используемых алгоритмов вместе с алгоритмом совместной фильтрации.

Влияние диверсификации на персонализированные рекомендации для мобильных приложений Amazon Prime Music. Этот результат усиливает представление о том, что разнообразие влияет на качество рекомендаций. Эффективность такого метода обеспечения разнообразия улучшают взаимодействие клиента с рекомендациями, треками, альбомами и плейлистами приложения Amazon Music.

1. Выделение функциональных требований в виде Use-case диаграммы UML

Во-первых, одна из главных проблем вышерассмотренных систем – это отсутствие их на рынке России, а также высокая стоимость подписки. Во-вторых, Российские аналоги отличаются малым музыкальным каталогом, а также использованием форматов сжатия звука с потерями.

Алгоритм рекомендаций и музыкальный каталог будут иметь решающее значение для проектируемого музыкального приложения. Также, чтобы обеспечить высококачественное воспроизведение, необходимо использовать формат FLAC, являющимся популярным свободным кодеком, предназначенным для сжатия аудиоданных без потерь, имея битрейт до 1096кбит/с. Сегодня формат FLAC поддерживается множеством аудио-приложений, портативных аудиоплееров и специализированных аудиосистем и имеет большое число разнообразных аппаратных реализаций. Например, популярный музыкальный сервис Tidal, отсутствующий на рынке России, использует данный формат сжатия.

В свою очередь, проектируемое мобильное приложение должно обладать необходимым алгоритмом генерации рекомендательных каталогов. В качестве используемого алгоритма для музыкального приложения выступит “Совместная фильтрация”, рассмотренная выше.

Опять же, стоит добавить, что сервис должен минимально потреблять трафик, обеспечивать высокую скорость кеширования треков, поддерживать устойчивую работу при плохой связи, доставлять аудиозаписи пользователю с максимальным битрейтом, обладать возможностью создания персонализированных плейлистов и иметь интуитивно-понятный пользовательский интерфейс.

Исходя из рассмотренных и проанализированных решений можно формализовать функциональные требования к предлагаемой системе в виде use case диаграммы.



Рисунок . Use case диаграмма

1. Специальный раздел
2. Диаграмма деятельности

Исходя из поставленных функциональных требований можно предположить, каким образом должно вести себя музыкальный сервис и формализовать это в виде диаграммы деятельности (Activity diagram), представленной на рисунке 15.



Рисунок . Диаграмма деятельности

1. Диаграмма классов

Основываясь на диаграмме деятельности можно приступить к проектированию логической структуры системы путем построения диаграммы классов (Class diagram), представленной на рисунке 16. В данной диаграмме выделяются и описываются основные классы системы и из взаимосвязи. В описываемой системе предполагается использование следующих классов:



Рисунок . Диаграмма классов

1. Диаграмма последовательности

Для более точного и полного описания жизненных циклов всех объектов системы и взаимодействия и общения между ними используется диаграмма последовательности (Sequence diagram), реализация которой для описываемой системы представлена на рисунке 17.



Рисунок . Диаграмма деятельности

1. Компонентная диаграмма

Особенности физического представления разрабатываемой системы и её общей архитектуры можно представить в виде диаграммы компонентов (Component diagram), которая представлена на рисунке 18.



Рисунок . Компонентная диаграмма

1. Заключение

В процессе работы было проведено исследование предметной области и существующих решений и алгоритмов в сфере музыкальных мобильных систем, потоковой передачи, сжатие аудиоданных с потерей и без потерь, алгоритмов рекомендаций и диверсификации. Были построены диаграммы вариантов использования, деятельности, классов, последовательности и компонентов для наиболее полного описания предполагаемого музыкального приложения.

Алгоритм рекомендаций и музыкальный каталог будут иметь решающее значение музыкального приложения. При внедрении алгоритмов рекомендаций в систему потокового вещания вырастает количество предлагаемого контента для пользователей, что позволяет удерживать высокий интерес к приложению, его постоянному использованию и упростить процесс поиска желанных песен для прослушивания.

Также, стоит отметить, что кодирование аудиоданных в формате FLAC предоставит возможность передавать весь спектр музыкального произведения в максимальном качестве.

Список литературы

1. Ricci F., Rokach L., Shapira B. Recommender systems: introduction and challenges //Recommender systems handbook. – Springer, Boston, MA, 2015. – С. 1-34.
2. Wei V., Nguyen K. Compatible Item Recommendation: дис. – 2018.
3. Hagen A. N. The playlist experience: Personal playlists in music streaming services //Popular Music and Society. – 2015. – Т. 38. – №. 5. – С. 625-645.
4. Glaser W. T. et al. Consumer item matching method and system: пат. 7003515 США. – 2006.
5. Howe M. Pandora’s Music Recommender //A Case Study, I. – 2009. – С. 1-6.
6. Atkinson J. MP3 vs AAC vs FLAC vs CD //Stereophile, March. – 2008. – Т. 31.
7. Graham J. J. Video and audio streaming for multiple users: пат. 6732183 США. – 2004.