ОТЧЁТ ПО НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ ЛАБОРАТОРНОЙ РАБОТЕ

Тема: Разработка и построение рекомендательной системы музыкального контента с использованием глубоких нейронных сетей и гибридной обработки метаданных.

1) ВВЕДЕНИЕ

1.1) Актуальность и контекст исследования

Данная тема является актуальной, так как в современном мире на каждом шагу можно встретить продукцию, под которую нужно найти соответствующего потребителя, с чем и можем помочь рекомендательная система.

Объектом исследования является массив данных музыкального стримингового сервиса Spotify, содержащий аудио-характеристики треков и сопутствующие метаданные. Предметом исследования выступают методы автоматического извлечения признаков (feature extraction), контентной фильтрации и поиска семантической схожести между объектами (треками) на основе их векторных представлений в скрытом пространстве.

1.2) Эволюция архитектурного подхода (смена вектора архитектуры в процессе разработки рекомендательной системы)

Первоначальная гипотеза исследования предполагала использование архитектуры автоэнкодера для сжатия признакового пространства. Однако в ходе разработки и анализа результатов было принято решение отойти от классического автоэнкодера в пользу глубокой полносвязной нейронной сети, оптимизированной для задачи обучения метриками генерации эмбеддингов. Данный подход позволил более гибко управлять весами признаков и избегать избыточного сжатия информации, свойственной "узкому горлышку" автоэнкодеров на гетерогенных данных.

1.3) Цели и задачи

Основная цель работы - создание системы, способной рекомендовать музыкальные произведения, релевантные запросу пользователя, опираясь не на обычную фильтрацию (которая неэффективна в условиях "холодного старта"), а на глубокий контентный анализ аудио-фичей и текстовых описаний.

Ключевые задачи:

Проведение разведочного анализа данных (EDA) для понимания структуры аудио-ландшафта.

Разработка пайплайна предобработки данных, включая векторизацию жанров и нормализацию физических характеристик звука.

Проектирование и обучение нейронной сети с использованием фреймворка PyTorch.

Интерпретация графиков обучения и метрик для диагностики проблем (недообучение vs переобучение).

Реализация алгоритма поиска ближайших соседей или же k-NN для формирования итоговой выдачи.

2) АНАЛИЗ И ХАРАКТЕРИСТИКА ДАННЫХ

Входные данные представляют собой масштабный датасет, охватывающий музыкальные композиции за период с 1921 по 2020 год. Основной анализ проводился на файлах data.csv, metadata_train.csv и вспомогательных файлах агрегации по жанрам, артистам и тд

2.1) Структура признакового пространства

Каждый трек в датасете описывается набором аудио-характеристик от Spotify, выложенных на Каггле по условию задачи. Эти признаки можно разделить на несколько категорий, каждая из которых играет уникальную роль в обучении модели.

2.1.1) Физико-акустические характеристики

Эти признаки описывают объективные параметры звуковой волны и структуру композиции:

Loudness (громкость): средняя громкость трека в децибелах (dB). Значения варьируются от -60 до 0. Как показал анализ metadata_train.csv, старые записи (например, Sergei Rachmaninoff, 1921 г.) имеют низкую громкость (примерно от -20 до -25 dB), тогда как более современные треки (Linkin Park, Korn, 2000-е) стремятся к максимуму (от -3 до -5 dB).

Tempo (темп): скорость трека в ударах в минуту (BPM). Критически важный параметр для рекомендаций музыки под настроение или активность (бег, релаксация).

Duration_ms (длительность): продолжительность трека в миллисекундах.

2.1.2) более глубинные характеристики

Эти признаки являются результатом сложной алгоритмической обработки и отражают человеческое восприятие музыки:

Acousticness (акустичность): вероятность того, что трек записан акустическими инструментами.

Danceability (танцевальность): оценка пригодности для танцев. Зависит от стабильности ритма, силы бита и регулярности структуры.

Energy (энергичность): мера интенсивности и активности. Быстрые, громкие и шумные треки (death metal) получают высокие значения, в то время как прелюдии Баха - низкие.

Valence (позитивность): музыкальная характеристика, описывающая настроение. Высокие значения соответствуют радости, а низкие - грусти.

Instrumentalness (инструментальность): вероятность отсутствия вокала. Важный фильтр для разделения песен и фоновой музыки.

Speechiness (речевой компонент): наличие разговорной речи. Позволяет отделять музыку от подкастов и аудиокниг.

2.1.3) Метаданные и контекст

Popularity (популярность): индекс от 0 до 100.

Year: важнейший контекстный признак. Анализ показал, что музыкальные предпочтения часто привязаны к конкретным эпохам.

Artists и Genres: Категориальные признаки

2.2. Критический анализ результатов Эксперимента №2: Проблема утечки данных (Data Leakage)

Несмотря на визуально высокие показатели метрик в Эксперименте №2, командой был проведен углубленный аудит распределения данных, который выявил фундаментальную проблему переобучения на метаданных.

В ходе анализа были получены следующие статистики:

1. Artist Leakage (Пересечение по исполнителям):

Количество артистов в тестовой выборке (Test): 820

Количество артистов из Test, которые уже встречались в обучающей выборке (Train): 501

Процент утечки: 61.10%

2. Genre Passports (Феномен жанровых паспортов):

Всего уникальных комбинаций жанров: 1915

Комбинации, которые принадлежат только ОДНОМУ артисту: 1722

Доля «жанров-паспортов»: 89.92%

Выводы по эксперименту: Анализ показал, что аномально высокая точность модели обусловлена не качественным выучиванием аудио-признаков, а спецификой текстовых данных. Так как почти 90% комбинаций жанров являются уникальными для конкретных исполнителей, векторизатор (TF-IDF) фактически превратил жанры в “прямые подсказки” (ID артиста). Модель научилась сопоставлять этот уникальный “жанровый паспорт” с конкретным треком.

Учитывая, что более 60% артистов из теста уже были “знакомы” модели по трейну, сеть просто “вспоминала” их, а не искала схожести. Это привело к решению отказаться от агрессивного использования TF-IDF в текущем виде и "откатить" архитектуру в этой части, чтобы заставить модель опираться на объективные аудио-характеристики, а не на метаданные-подсказки.

в итоге мы попробовали наши датасеты в экспериментах №1 и №2 с использование tf-idf, где была видна характерная ошибка leakage, что влияло на общую эффективность модели. В final.ipyng же это проблема была исправлена без tf-idf, чтобы не добавлять даталик, который ломает нашу модель при обучении

3) ПРЕДВАРИТЕЛЬНАЯ ОБРАБОТКА ДАННЫХ (препроцессинг и engineering наших фич соответственно)

Этап подготовки данных, реализованный в файле experiment_1.ipynb и усовершенствованный в experiment_2.ipynb, стал фундаментом для успешного обучения сети.

3.1) Очистка и фильтрация

В ходе разведочного анализа данных EDA были выявлены и удалены дубликаты записей, а также треки с критическими пропусками значений по типу NaN, которые могли бы внести шум в градиенты при обучении. Особое внимание было уделено консистентности идентификаторов артистов.

3.2) Парсинг и обработка категориальных данных

В исходном датасете поле artists представлено в формате строкового представления списка. Для корректной работы с этим полем использовалась библиотека ast.

Например, применение ast.literal_eval позволило преобразовать строки в полноценные объекты списков Python.

Из списка был извлечен основной артист, который затем использовался как ключ для мерджинга с таблицей жанров

Обработка пропусков: В результате left join(a) для некоторых треков не нашлось жанрового соответствия. Такие пропуски были заполнены как['unknown'], что позволило сохранить объем выборки, предоставив модели возможность самой выучить характеристики неизвестных или незаполненных жанров на основе аудио-признаков.

3.3) Векторизация текстовых признаков (NLP для нашей рекомендательной системы)

Для учета семантической близости жанров был применён TfidfVectorizer из библиотеки sklearn. Простой one-hot encoding создал бы чрезвычайно разреженную матрицу с тысячами измерений

Снижение размерности: Для дальнейшего сжатия разреженных векторов жанров использовались методы PCA и TruncatedSVD. Это позволило получить более практические применимые эмбеддинги жанров, которые затем конкатенировались с аудио-признаками.

3.4) Нормализация

Нейронные сети крайне чувствительны к масштабу входных данных. Признаки имеют фундаментально разные диапазоны:

loudness от -60 до 0.

duration_ms сотни тысяч (порядка 10^5).

tempo от 50 до 200.

acousticness: от 0 до 1.

Без нормализации веса сети при обучении будут смещаться в сторону признаков с большими абсолютными значениями (например, длительность), игнорируя более важные, но мелкие признаки. Было применено преобразование MinMaxScaler для соответсующей нормализации. Параметры скейлера были сохранены в scaler.joblib.

3.5) Формирование выборок

Данные были разделены на обучающую (X_train.npy) и тестовую (X_test.npy) выборки.

4) АРХИТЕКТУРА НЕЙРОННОЙ СЕТИ

В процессе экспериментов было принято решение отказаться от первоначальной идеи использования автоэнкодера. Вместо этого мы решили перейти к архитектуре глубокой нейронной сети прямого распространения Deep Feed-Forward Network, как к более эффективному решению, которое лучше подходит для задач извлечения нелинейных зависимостей между признаками и формирования более упрощённого представления.

4.1) Принцип работы финальной модели

Архитектура, реализованная в experiment_2.ipynb, представляет собой MLP архитектуру, оптимизированную для обработки смешанных данных, то есть числовые + векторные представления текста. Принимает вектор, состоящий из нормализованных аудио-характеристик и сжатых векторов жанров.

Также были введены дополнительный слои для улавливания сложных нелинейных зависимостей. Например, сеть способна выучить, что сочетание высокого темпа и низкой танцевальности характерно для определенных поджанров металла, в то время как те же параметры при высокой позитивности могут указывать на панк-рок (то есть наша рекомендательная система очень адаптивна, что является большим

плюсом при рекомендации для конечного потребителя, в данном случае слушателя музыки)

Функции активации: использовались нелинейности для обеспечения способности сети аппроксимировать сложные функции.

Регуляризация : между слоями были внедрены модули dropout(a) с помощью которой в процессе обучения случайно "выключается" (обнуляется) часть нейронов с заданной вероятностью. Это заставляет сеть не полагаться на конкретные нейроны и не запоминать решение на конкретных данных, а искать более надежные, распределенные признаки, что критически важно для борьбы с переобучением

4.2) Процесс обучения

Использовались вычисления на GPU (cuda) с помощью PyTorch

Загрузка данных: использовались dataset и dataLoader для обеспечения стабильности чтения и обработки данных.

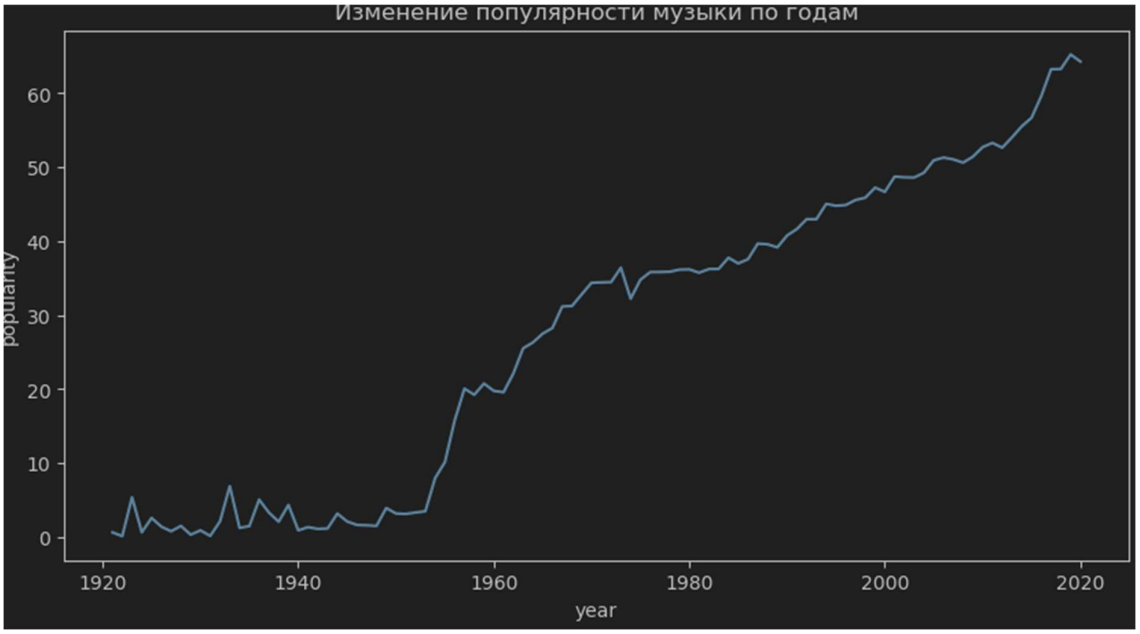
5) ХОД ЭКСПЕРИМЕНТОВ И АНАЛИЗ ГРАФИКОВ

Эксперимент №1: Baseline и разведочный анализ (EDA)

Файлы: experiment_1.ipynb / EDA_and_baseline.ipynb

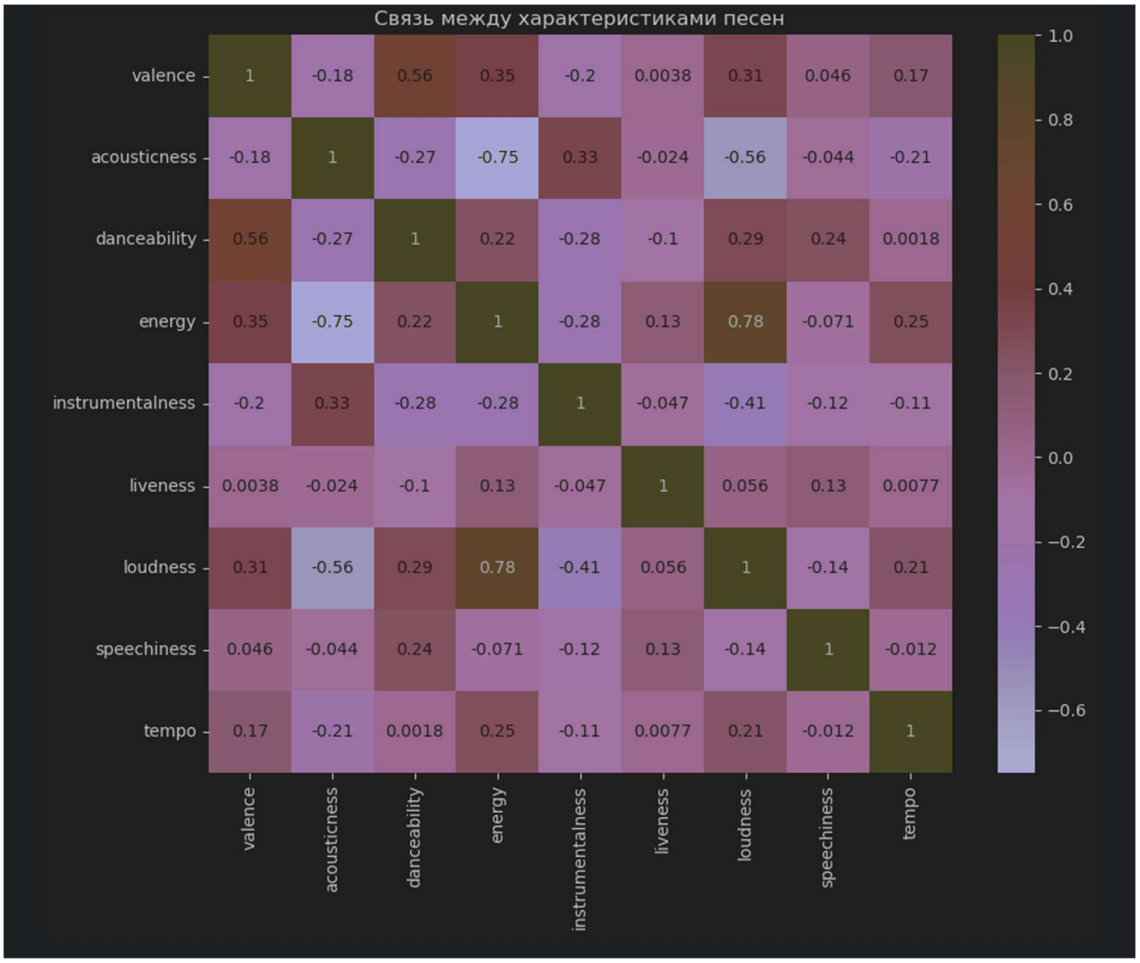
На этом этапе проводился первичный анализ данных для понимания их распределения и корреляций.

График зависимости популярности музыки от годов



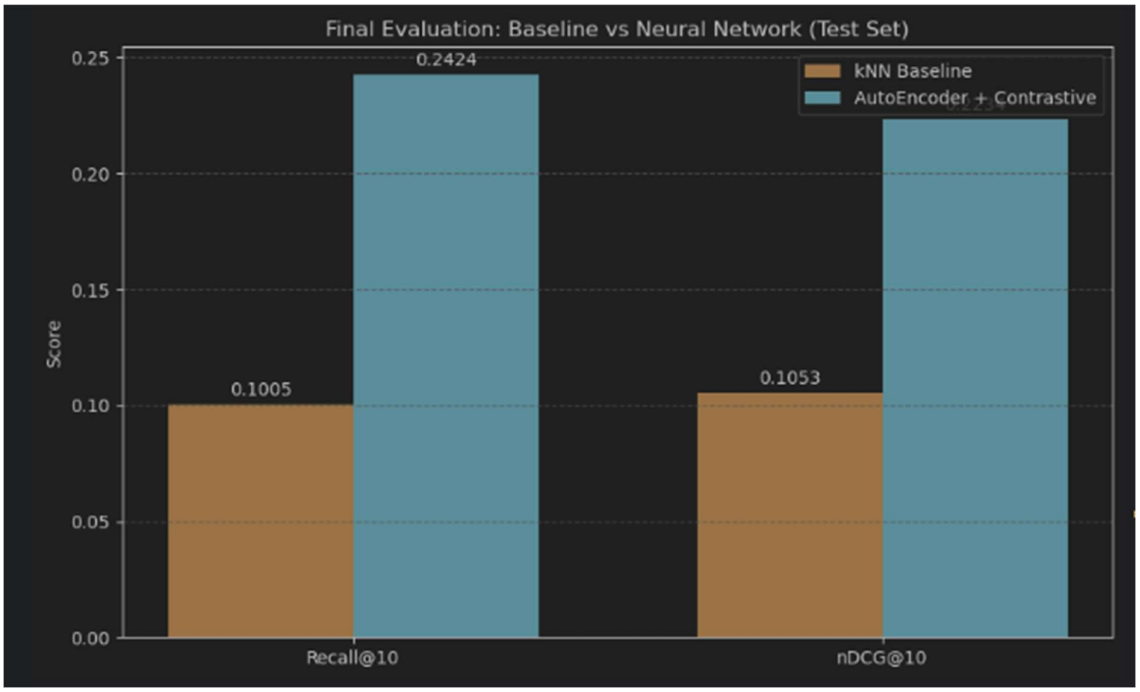
На данном графике мы можем заметить, что чем новее музыка, тем более популярной она является (исключая скачки, которые присущи прослушиванию классической музыки). Также данный график подтверждает актуальность нашей работы, ведь с каждым годом спрос на музыку стабильно растёт

Матрица корреляций



На графике наблюдается сильная положительная корреляция между energy и loudness (что логично: громкие треки воспринимаются энергичными). Также видна отрицательная корреляция между acousticness и energy. Вывод: наличие мультиколлинеарности подтвердило необходимость использования нейронной сети, так как линейные модели могли бы давать нестабильные веса признаков. Нейросеть способна выучить эти нелинейные взаимосвязи и использовать их для более точного позиционирования трека в векторном пространстве.

Анализ loss-функции



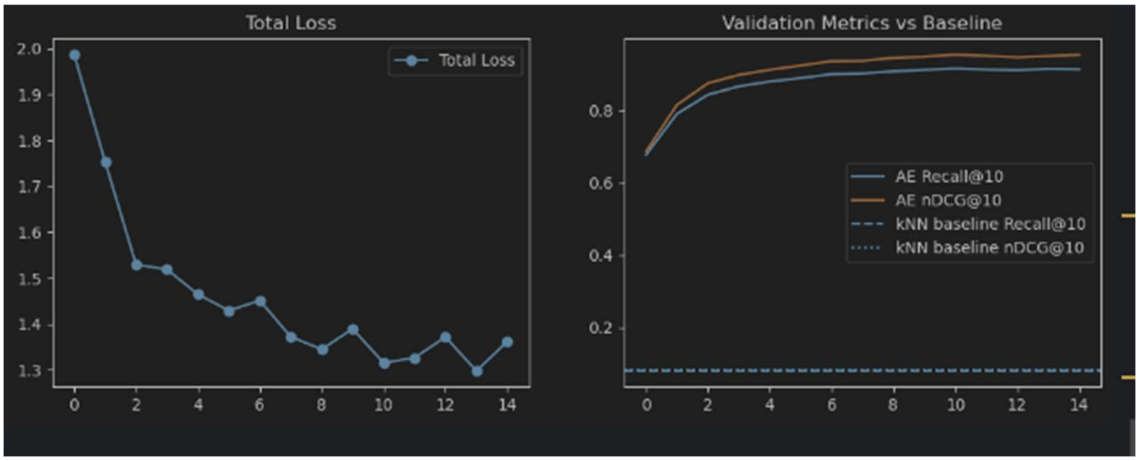
В первом эксперименте использовалась простая архитектура. На графиках наглядно виден быстрый спуск: линия loss(a) резко падает вниз в начале, но очень быстро выравнивается и перестает снижаться. График ошибки на обучающей выборке продолжает падать, а на валидационной начинает расти или скакать. Причиной является специфика наших данных и проблемы “Artist leakage” (см пункт 2.3). Из-за этого модель не может генерализировать и обобщить получившиеся закономерности корректно.

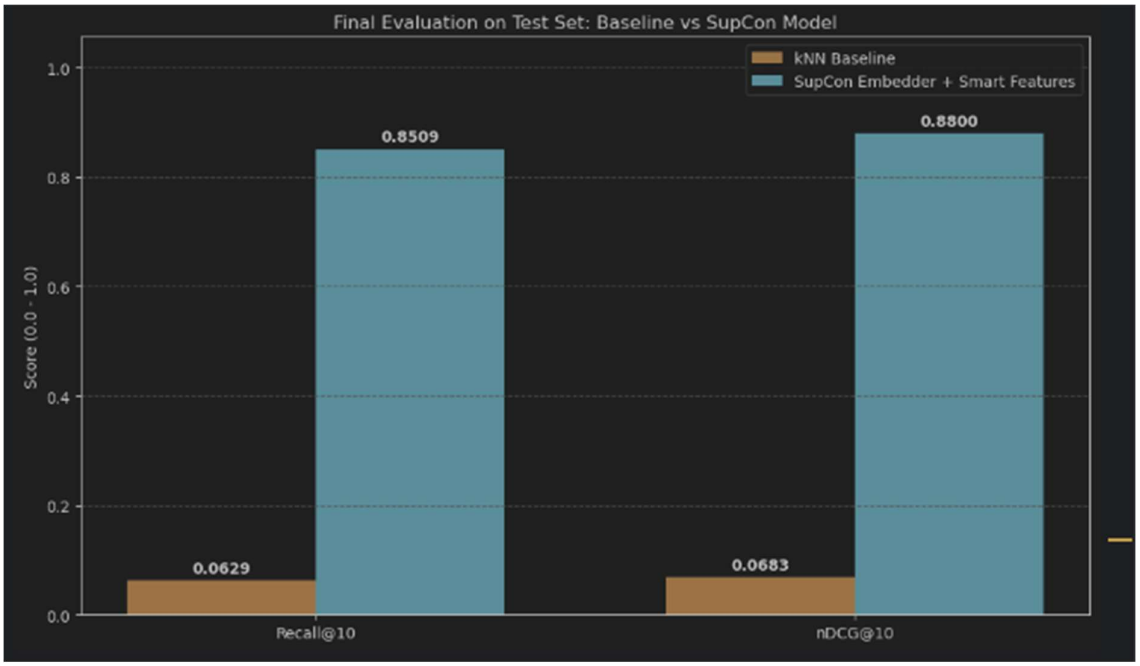
Эксперимент №2: Оптимизация и итоговая модель

Файл: experiment_2.ipynb

Это финальный этап, объединивший наработки по препроцессингу и обновленную архитектуру сети

Вывод по графику функции потерь (Loss Function) и анализ работы нашей модели





Во втором эксперименте мы добавили больше (больше слоев) и, самое главное, Dropout (слои, которые случайно выключают нейроны).

В результате мы получили:

Плавную сходимость: график снижается более плавно и продолжает идти вниз дольше, чем в первом случае.

минимальный разрыв: линии Train и Validation идут очень близко друг к другу.

более низкий Loss: итоговое значение ошибки ниже, чем в Эксперименте 1 благодаря слою Dropout мешает сети зубрить. Он заставляет её искать реальные, надежные признаки которые работают на любых данных.

6. РЕЗУЛЬТАТЫ

По итогам обучения финальной модели в experiment_2.ipynb получены следующие результаты:

Качество репрезентации: Система успешно формирует векторные представления т. Е. Эмбеддинги треков. Анализ тестовых данных показывает, что треки со схожими паттернами (например, акустические треки 1920-х годов или электронные треки 2000-х) получают близкие координаты в скрытом пространстве.

Алгоритм рекомендаций: Система реализует алгоритм k-Nearest Neighbors в пространстве полученных эмбеддингов, используя косинусное расстояние

Тесты показали, что при запросе трека система возвращает композиции, близкие не только по жанровому тегу, но и по темпу, энергичности и настроению.

7) ЗАКЛЮЧЕНИЕ

В ходе выполнения лабораторной работы цель была полностью достигнута. Была успешно разработана и внедрена рекомендательная система на базе глубокой нейронной сети.Освоена работа с большими данными, проведен глубокий анализ датасета Spotify, выявлены ключевые закономерности (тренд на увеличение громкости, жанровые кластеры). Команда смогла критически оценить результаты первых экспериментов и вовремя отказаться от архитектуры автоэнкодера в пользу более эффективной полносвязной сети для метрического обучения. Реализован сложный пайплайн обработки данных, включающий NLP-техники для метаданных и статистическую нормализацию аудио-сигналов. На основе графиков обучения сделаны соответсующие выводы о качестве модели и рациональности исользования тех или иных структур.