

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет
імені Петра Могили
Факультет комп'ютерних наук
Кафедра інтелектуальних інформаційних систем

ДОПУЩЕНО ДО ЗАХИСТУ
Завідувач кафедри інтелектуальних
інформаційних систем, д-р техн. наук, проф.
_____ Ю. П. Кондратенко
« ____ » _____ 2022 р.

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

ІНТЕЛЕКТУАЛЬНА СИСТЕМА АНАЛІЗУ КОНТЕНТУ
МУЗИЧНОГО ВЕБСЕРВЕРУ ДЛЯ ANDROID-
ЗАСТОСУНКУ

Спеціальність 122 «Комп'ютерні науки»

122 – МКР – 601. 21610223

Студент _____ М. Д. Скубак

«14» лютого 2022 р.

Консультант _____ І. А. Калініна
канд. техн. наук, доцент

«14» лютого 2022 р.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ.....	3
ВСТУП.....	5
1 АНАЛІЗ СУЧАСНОГО СТАНУ ОБРАНОЇ ЗАДАЧІ	7
1.1 Опис предметної сфери	7
1.2 Огляд та аналіз наявних аналогів та публікацій.....	8
1.3 Постанова задачі	19
Висновки до розділу 1	19
2 МАТЕМАТИЧНІ МОДЕЛІ, МЕТОДИ, ІФОРМАЦІЙНІ ТЕХНОЛОГІЇ ДЛЯ ВИРІШЕННЯ ПОСТАВЛЕНОЇ ЗАДАЧІ.....	21
2.1 Опис обраного методу	21
2.2 Аналіз обраних методів кластеризації.....	30
Висновки до розділу 2	35
3 МОДЕЛЮВАННЯ ТА ДОСЛІДЖЕННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ. ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ	37
3.1 Аналіз методу k-середніх.....	37
3.2 Аналіз методу пошарової кластеризації.....	45
Висновки до розділу 3	48
4 МОДЕЛЮВАННЯ ТА ДОСЛІДЖЕННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ. ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ	50
4.1 Проектування інформаційного забезпечення	50
4.1.1 Діаграма прецедентів. Опис прецедентів.....	50
4.1.2 Логічна схема даних	60
4.2 Програмна реалізація.....	63
Висновки до розділу 3	75
ВИСНОВКИ.....	77
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	79
ДОДАТОК А.....	80
ДОДАТОК Б.....	82

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

ВУ – відношення узгодженості.

ІС – інформаційні системи.

ІУ – індекс узгодженості.

ЛПР – людина, що приймає рішення.

Прецедент (від лат. *praecedens* – той, що передує) – випадок або подія, що відбулася в минулому та є прикладом або підставою для аналогічних дій у сьогоденні.

СППР – система підтримки прийняття рішень.

Сутність – це збірне поняття, деяка абстракція реального об'єкта, процесу, явища чи деякого уявлення про об'єкт.

НТТР (Назва скорочена від *Hyper Text Transfer Protocol*) – протокол передачі даних, що використовується в комп'ютерних мережах.

Пояснювальна записка

до магістерської кваліфікаційної роботи

на тему:

«ІНТЕЛЕКТУАЛЬНА СИСТЕМА АНАЛІЗУ МУЗИЧНОГО ВЕБСЕРВЕРУ ДЛЯ АНДРОЇД ДОДАТКУ»

Спеціальність 122 «Комп'ютерні науки»

122 – МКР – 601.21610223

Студент _____ М. Д. Скубак

«__» _____ 20__ р.

Консультант _____ Є. В. Сіденко

канд. техн. наук, доцент

«__» _____ 20__ р.

Миколаїв – 2022

ВСТУП

Музикальні додатки використовуються людьми у всіх куточках планети. Музика є предметом організації музичних звуків, насамперед у часовій (ритмічній), звуковисотній та тембровій шкалі. Музичним може бути практично будь-який звук з певними акустичними характеристиками, які відповідають естетиці тієї чи іншої епохи, та може бути відтвореним. Джерелами такого звуку можуть бути: людський голос, музичні інструменти, електричні генератори тощо.

Музика діє вібраційно та створює незвичний вплив на організм людини. Зміни відбуваються на рівні клітин і органів, почуттів та думок. Сила музики в тому, що ці зміни здійснюються незалежно від бажань і намірів конкретної людини і тому здаються чудодійними.

Музикотерапія є частиною оздоровлення та гармонізації особистості із середовищем і світом загалом. Так Піфагор уважав, що музика як прояв космічної гармонії може створювати у людини такий самий внутрішній порядок, гармонію, як у Космосі, а французький композитор Марен Маре (1656-1726) 200 років тому писав музику, спрямовану на лікування різних захворювань. Цикл із 12 сонатів був призначений для лікування подагри, соната для альтя і клавесина використовувалася для супроводу операцій. Тому музику і застосовують у вихованні, навчанні, лікуванні та навіть в управлінні державою.

Задачею моєї роботи є аналіз музики, що слухає користувач та визначення можливих музикальних композицій, що можуть сподобатись користувачеві на базі прослуханого матеріалу. Аналогічним додатком з майже подібним аналізом є додаток Spotify, проте він лише рекомендує пісні та жанри, що були обрані користувачем, а не те, що може бути спільним до композиції чи жанру 2.1.

Отже, **об'єктом дослідження** магістерської кваліфікаційної роботи є система аналізу музичних творів.

Кафедра інтелектуальних інформаційних систем
Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку
Предметом дослідження є засоби і технології аналізу для створення

рекомендацій на основі прослуханого матеріалу.

Мета роботи полягає у підвищенні ефективності рекомендаційних систем андроїд додатків для прослуховування музичного матеріалу.

До складових **задач роботи** варто віднести:

- дослідження існуючих систем рекомендацій музики;
- огляд сучасних методів, що можуть підвищити ефективність управління;
- аналіз та вибір найкращого методу для підвищення ефективності;
- проектування системи рекомендації музичних творів;
- програмна реалізація системи, створення андроїд додатку;
- розробка документації (у вигляді пояснювальної записки до магістерської кваліфікаційної роботи).

1 АНАЛІЗ СУЧАСНОГО СТАНУ ОБРАНОЇ ЗАДАЧІ

1.1 Опис предметної сфери

Дослідники вважають, що однією з найважливіших функцій музики є створення відчуття згуртованості чи соціального зв'язку.

Вчені стверджують, що люди могли розвинути залежність від музики як засобу спілкування, тому що наші предки походили від деревних видів — мешканців дерев, які кликали один до одного через крони.

Музика залишається потужним засобом об'єднання людей:

- державні гімни об'єднують натовп на спортивних змаганнях;
- пісні протесту викликають відчуття спільної мети під час маршів;
- гімни формують групову ідентичність у молитовних будинках;
- пісні про кохання допомагають майбутнім партнерам зблизитися під час залицяння;
- колискові пісні дозволяють батькам і немовлятам розвинути надійну прихильність.

Лікарі Johns Hopkins рекомендують слухати музику, щоб стимулювати мозок. Вчені знають, що прослуховування музики задіює ваш мозок — вони бачать, що активні ділянки світяться на МРТ.

Тепер дослідники знають, що лише обіцянка слухати музику може викликати бажання дізнатися більше. В одному дослідженні 2019 року люди були більш мотивовані вчитися, коли очікували послухати пісню як нагороду.

Музика також позитивно впливає на вашу здатність запам'ятовувати. В одному дослідженні дослідники давали людям завдання, які вимагали від них прочитати, а потім пригадати короткі списки слів. Ті, хто слухав класичну музику, перевершували тих, хто працював у тиші або з білим шумом.

У тому самому дослідженні відстежувалося, як швидко люди можуть виконувати прості завдання обробки – зіставляти числа з геометричними фігурами – і виявилось подібна перевага. Моцарт допомагав людям виконувати завдання швидше і точніше.

Кафедра інтелектуальних інформаційних систем
Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку
Клініка Мауо зазначає, що хоча музика не повертає втрату пам'яті, яку

відчувають люди з хворобою Альцгеймера та іншими формами деменції, проте було встановлено, що музика уповільнює зниження когнітивних функцій, допомагаючи людям з легкою або помірною деменцією запам'ятати епізоди зі свого життя.

Музична пам'ять є однією з функцій мозку, найбільш стійких до деменції. Ось чому деякі опікуни успішно використовують музику, щоб заспокоїти пацієнтів з деменцією та побудувати з ними довірчі зв'язки.

З огляду на дослідження, можна стверджувати, що створення системи рекомендації музичних творів зможе підвищити ефективність пошуку композицій, які допомагають людині в різних сферах діяльності. Загалом це підвищення ефективності може покращити стан людини та його якості.

Оскільки даний вид систем вже існує, слід переглянути дослідження по даній темі та знайти можливе вдосконалення існуючої системи чи навіть створення нової, більш сучасної, показники вподобання якої будуть вище за існуючі. Ця тема є досить актуальною, оскільки мільярди людей користуються засобами зв'язку, які мають технічну можливість програвати музичний контент.

Для покращення рекомендаційної частини музичних сервісів слід дослідити можливість створення системи, що використовує кластеризацію даних.

1.2 Огляд та аналіз наявних аналогів та публікацій

Виконуючи кластеризацію існує можливість не обмежуватися класифікацією жанрів, виявляючи схожість між музикою замість того, щоб бути пов'язаними суб'єктивністю жанрів. Кластеризація на основі аудіофункцій доповнює музичні жанри і може виводити більш описові піджанри на основі аудіо-функцій доріжки.

Об'єднання музики в жанри на основі аудіо-функцій дозволяє описувати музику по-новому. Поєднуючи ці жанри з умовностями, які вже

Кафедра інтелектуальних інформаційних систем
Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку
використовуються в жанрах, позначених людьми, можна створити нові й більш
описові жанри та позначити їх на музиці.

Хоча створені жанри дозволяють розглядати музику з іншого ракурсу, було виявлено, що створені музичні жанри сумісні з жанрами, позначеними людьми. Наприклад, шанувальнику поп-музики, імовірно, сподобаються «Життєрадісні пісні з веселим вокалом», які вони інакше не виявили б, якби дотримувались звичайних жанрів.

Серед усіх публікацій про кластеризації даних слід виділити статтю «Кластеризація k-середніх і PCA для класифікації музики за схожими звуковими характеристиками», в якій аналіз музикальних композицій було зроблено за допомогою машинного навчання та методу кластеризації k-середніх [1]. Розроблена автором система дозволяє проаналізувати музику за її акустичним звучанням та рекомендувати контент з певного сегмента, кластеру (Рис. 1.1).

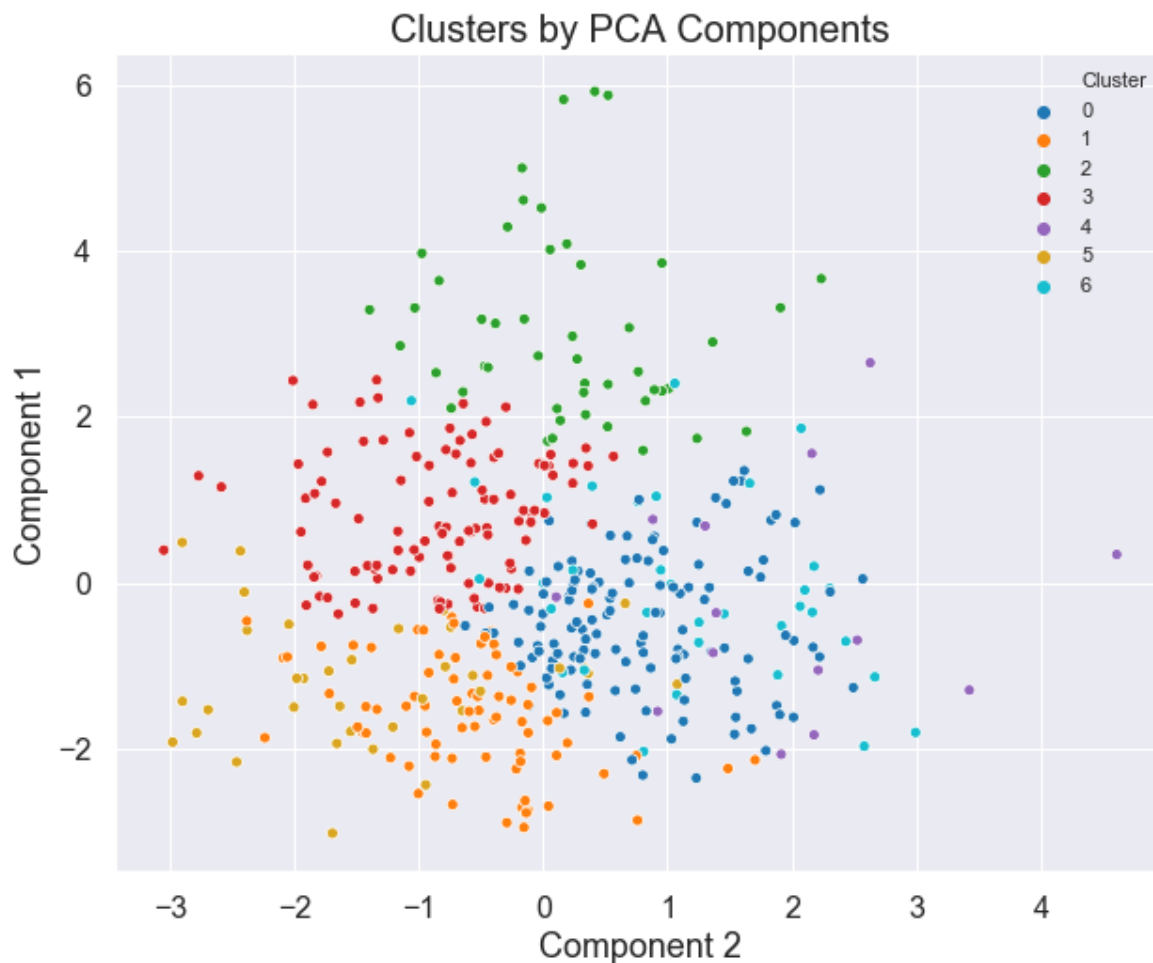


Рис. 1.1. Розбиття музикальних композицій на кластери

На рис. 1.2 побудована радіолокаційна траса для середніх значень звукових характеристик у кожному кластері після нормалізації всього кадру даних. Акустичність - це обумовлена Spotify змінна від 0 до 1, тоді як темп може бути не більше 100 секунд. Нормалізація важлива, тому що вона ефективно масштабує всі змінні до значень від 0 до 1, роблячи наступну візуалізацію набагато значущою.

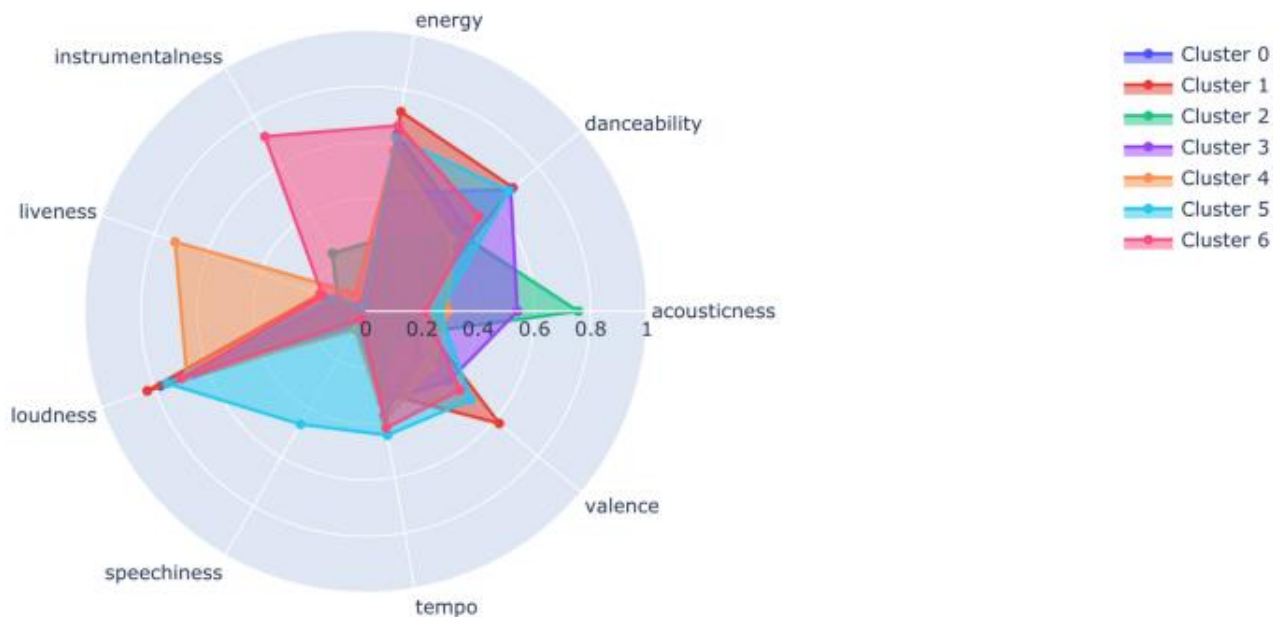


Рис 1.2. Радіолокаційна траса для середніх значень звукових характеристик для кожного кластеру

За отриманими даними автор статті стверджує, що може створювати нові списки відтворення Spotify на основі кластерних груп треків після PCA та k-середніх.

Іншим прикладом публічної роботи є праця Бакалавра «Гібридна система класифікації музикальних треків» [2]. За цією роботою дані кластеризують за допомогою методу «Виділення зв'язкових компонентів». Цей алгоритм є модифікацією процесу випадкового блукання на графі. Для реалізації алгоритму необхідно:

- вибрати m вершин графа A . Також визначити число a – ймовірність;
- почати обхід наново на кожному кроці алгоритму з ймовірністю $(1 - a)$;
- блукаючий об'єкт рівноймовірним випадковим чином повертається в одну з вибраних m вершин, з ймовірністю a відбувається перехід у сусідню вершину (у цьому випадку ймовірність переходу у вершину j дорівнює $P_{i,j}$).

Таким чином, після закінчення процесу блукання, можна виділити вершини, які відвідували найбільшу кількість разів. Ці вершини будуть рекомендацією для користувача, що відмітив m вершин.

Популярними онлайн-сервісами, що мають реалізовану рекомендаційну систему є «Spotify», який є популярним у всьому світі. Цей сервіс має платну підписку для відкриття повного функціоналу, яку потрібно оплачувати кожного місяця.

Spotify – це стрімінговий сервіс, який дозволяє легально прослуховувати музичні твори, аудіокниги та подкасти, не завантажуючи їх на пристрій. Доступний у вигляді веб-сайту, програм для перегляду всіх систем, смартфонів, смарт-пристроїв та медіа-систем автомобілів. У службі підтримки понад 36% аудіострімінгу, доступного в більшості країн Північної та Південної Америки, практично у всіх країнах Європи, деяких країнах Азії та Південної Австралії, Австралії та Океанії.

Щоб зрозуміти як працює рекомендаційна система в «Spotify» потрібно протестувати власними зусиллями додаток, оскільки інформація щодо рекомендаційної системи є закритою, тому що є комерційно успішною вижче.

На головній сторінці додатку можна побачити плейлісти, що створив користувач, рекомендації, що створюються на основі прослуханої музики (Рис. 1.3).

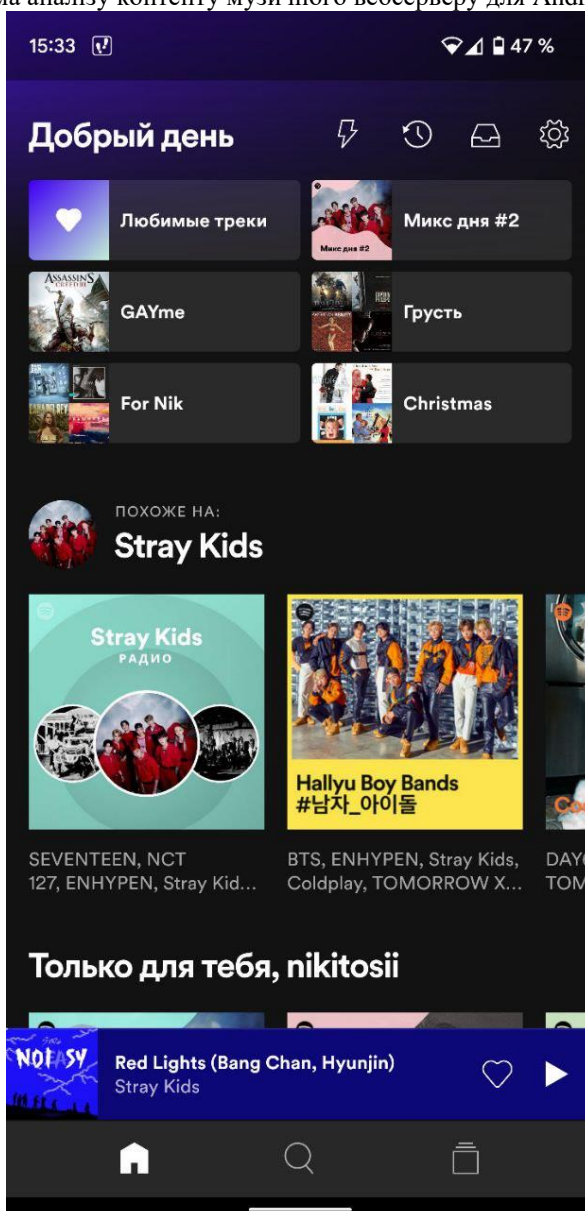


Рис 1.3. Головна сторінка додатку «Spotify»

За допомогою нижнього меню можна відкрити сторінку пошуку пісні, альбому, виконавця тощо (Рис. 1.4). Щоб відкрити дану сторінку, потрібно натиснути на кнопку 3 із зображенням лупи.

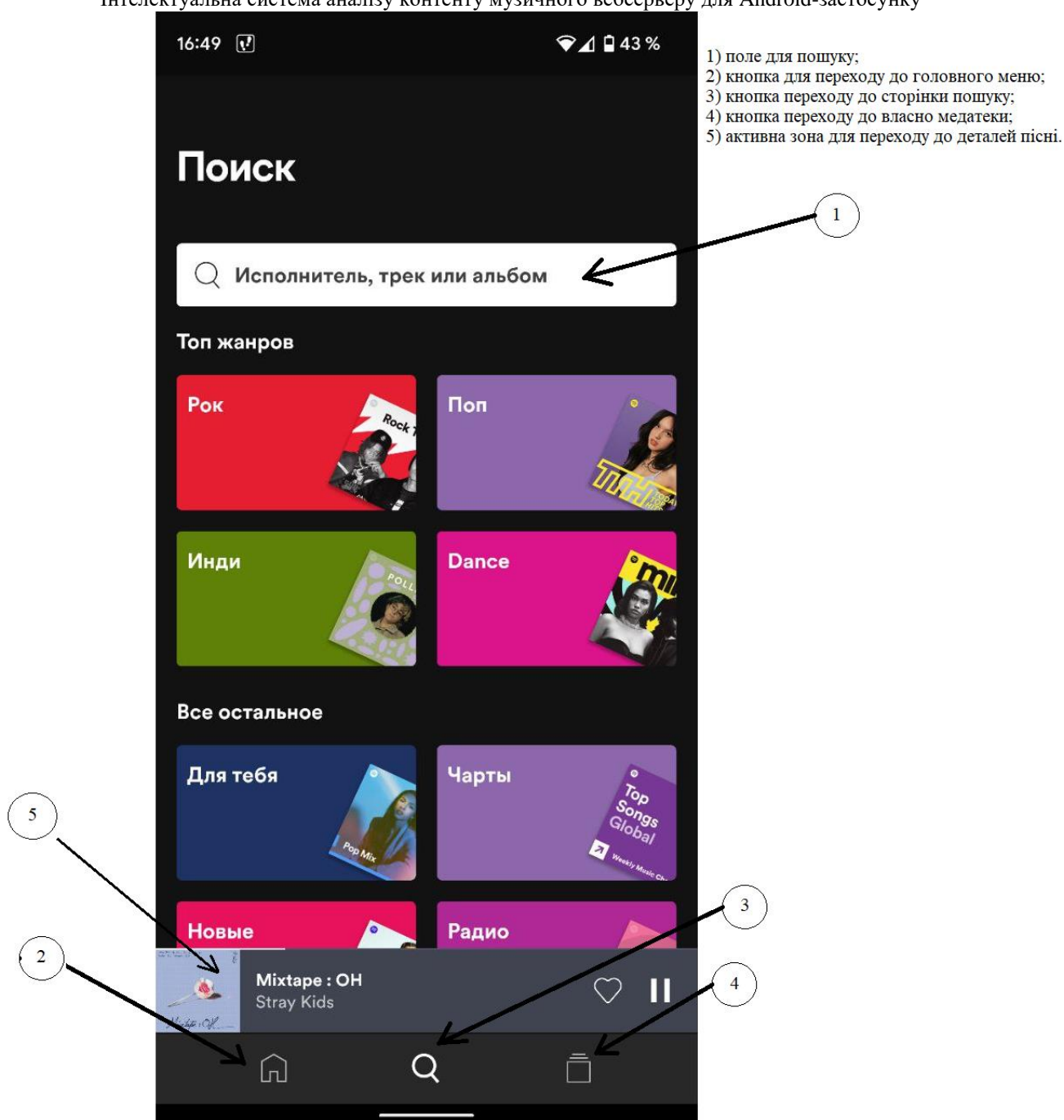


Рис. 1.4. Сторінка пошуку в додатку «Spotify»

Також, за допомогою кнопки 4 можна відкрити сторінку, на якій буде заходитись інформація щодо власних створених та на які користувач підписався плейлістів (Рис 1.5).

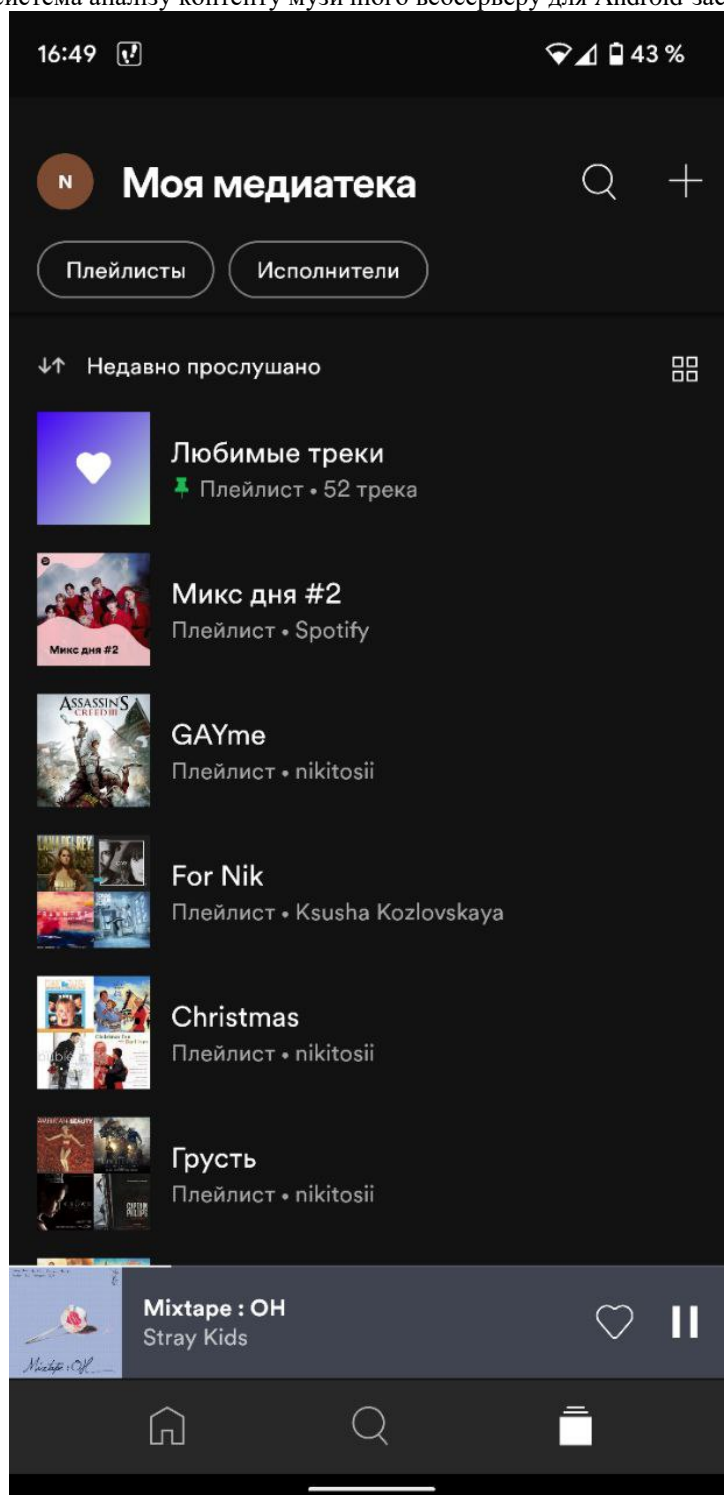


Рис. 1.5. Власна медіатека додатку «Spotify»

Натиснувши на будь який плейліст, відкриється список пісень, який наявний у цьому плейлісті (Рис 1.6).

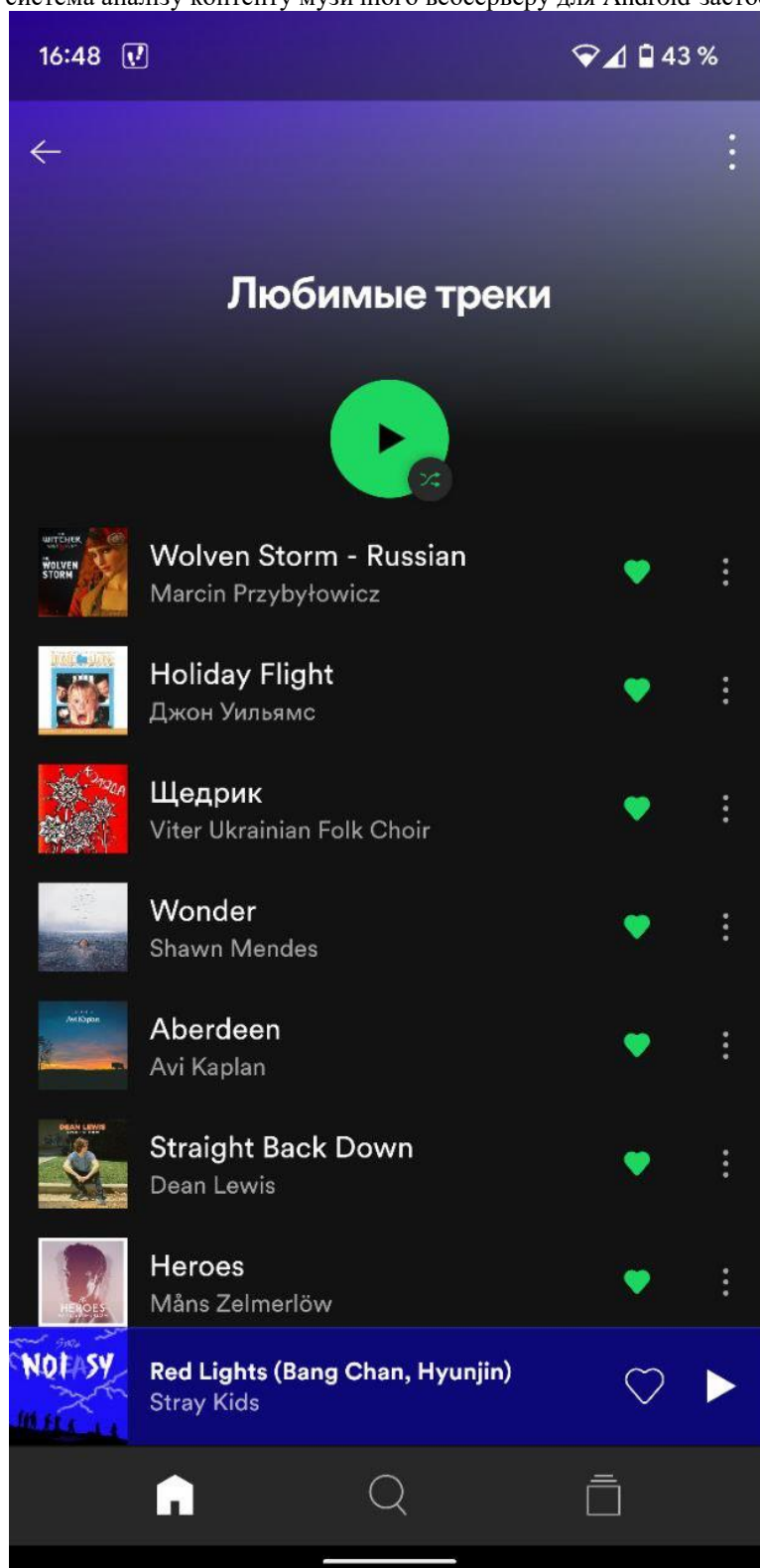


Рис. 1.6. Список пісень в плейлісті «Улюблені треки» додатку «Spotify»

Натиснувши на активну зону 5, на якій зображено картинка альбому та інформація щодо пісні, відбувається перехід до сторінки «Деталі пісні» (Рис 1.7).



Рис. 1.7. Сторінка «Деталі пісні» додатку «Spotify»

Згідно рекомендаціям пісень додатку, він рекомендує все, що прослуховував користувач, навіть якщо ввімкнув щось помилково. Всі ці рекомендації можуть бути трохи хибними, проте, на мій погляд, всі пісні

Кафедра інтелектуальних інформаційних систем
Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку
обробляються за певним алгоритмом, що використовує метод кластеризації.
Якщо точніше, то це є ієрархічна кластеризація (рис. 1.8).

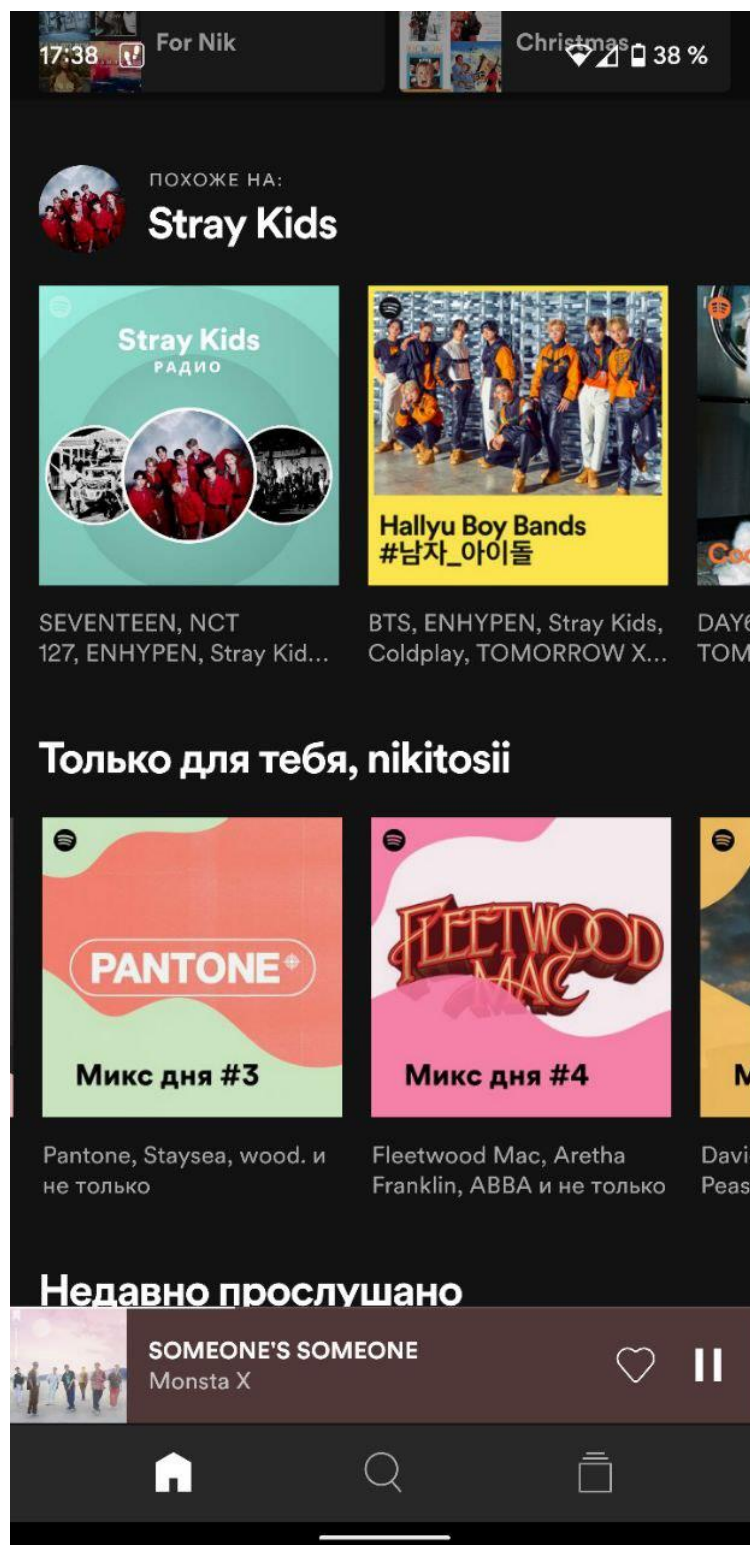


Рис. 1.8. рекомендації додатку «Spotify»

1.3 Постановва задачі

Слід покращити існуючі засоби та системи аналізу музичних композицій для підвищення ефективності створення рекомендацій. Розглянути можливі існуючі варіанти серед алгоритмів та методів аналізу даних.

Для підвищення ефективності рекомендується застосувати методи кластеризації даних та проаналізувати результати методів та підходи, які вони потребують для застосування. Кластеризація даних допоможе згрупувати музичні композиції за спільними властивостями чи характеристиками та рекомендувати саме ті, що мають більшу популярність у користувача.

Для тестування системи слід розробити проектну реалізацію додатку для операційної системи Андроїд. Оскільки розробка системи запропонована для музичного вебсерверу, вимоги до серверу є досить високими (повинно бути багато контенту, створити зв'язок між бек-частиною та мобільною, слідкувати за працездатністю серверної частини проекту) можна розробити лише демо версію, яка працюватиме із локальними даними на смартфоні (близько 50 пісень), таким чином можна протестувати рекомендації звузивши рівень розробки та її складність до мінімуму. Проаналізувавши отримані с демо версії результати, можна буде припустити щодо актуальності створеної нової системи для комерційного застосування.

Висновки до розділу 1

Під час аналізу сучасного стану обраної теми було проаналізовано сферу застосування рекомендаційних систем музичних вебсерверів для андроїд-додатків, їх вигоду у використанні та у створенні нової системи для покращення ефективності рекомендацій.

Для створення нової інтелектуальної системи було проаналізовано останні публікації та дослідження у галузі «кластеризація даних» за темою «аналіз музичних композицій». За даними з цих публікацій відібрано кластеризацію даних як засіб для подальшого аналізу та застосування в системі.

Якщо створена система зможе поліпшити продуктивність аналізу музичних композицій не вводячи додаткових проблем, тоді слід розробити дану програму.

Проаналізувавши альтернативи, які існують та мають високу популярність серед інших, було сформовано вимоги до розробляемого проекту:

- необхідно створити можливість переглянути наявні музичні твори на девайсі;
- необхідно розробити можливість програвати музику на девайсі;
- створити можливість збирати власні плейлісти;
- потрібно сформувати систему помітки музичних творів за допомогою тегів, за цими тегами розробити інтелектуальную систему аналізу музичних творів.

Для легкої та комфортної розробки проекту обрано мову програмування Kotlin. Із існуючих систем з управління базами даних для Android-застосунків обрано Room. Для роботи із потоками було використано RxJava. Із середовищ розробки обрано AndroidStudio.

2 МАТЕМАТИЧНІ МОДЕЛІ, МЕТОДИ, ІФОРМАЦІЙНІ ТЕХНОЛОГІЇ ДЛЯ ВИРІШЕННЯ ПОСТАВЛЕНОЇ ЗАДАЧІ

У даному розділі розглянуто та проаналізовано відомі методи кластеризації даних.

Проблема кластеризації музичного контенту полягає в тому, що важко оцінити об'єктивно технічно-музикальні характеристики в чисельному еквіваленті, а техніки машинного навчання приймають на вхід чітко визначені параметри.

Не всі методи кластеризації працюють із музичним контентом. Для того, щоб використовувати дані пісень в алгоритмах кластеризації, необхідно представити їх у вигляді числових датасетів, або векторів. Пошарова кластеризація дозволяє обробляти контент за допомогою спільних характеристик, в даному випадку, існує можливість створення тегів, які будуть використовуватись в обробці як спільні елементи.

2.1 Опис обраного методу

Кластеризація (або кластерний аналіз) - це завдання розбиття безлічі об'єктів на групи, які називаються кластерами. У середині кожної групи повинні виявитися «схожі» об'єкти, а об'єкти різних групи мають бути якомога відміннішими. Головна відмінність кластеризації від класифікації у тому, що перелік груп чітко не заданий й у процесі роботи алгоритму. Спектр застосувань кластерного аналізу дуже широкий: його використовують в археології, антропології, медицині, психології, хімії, біології, державному управлінні, філології, маркетингу, соціології та інших дисциплінах. Однак універсальність застосування привела до появи великої кількості несумісних термінів, методів і підходів, що ускладнюють однозначне використання і несуперечливу інтерпретацію кластерного аналізу (Рис 2.1).

Кафедра інтелектуальних інформаційних систем
Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку

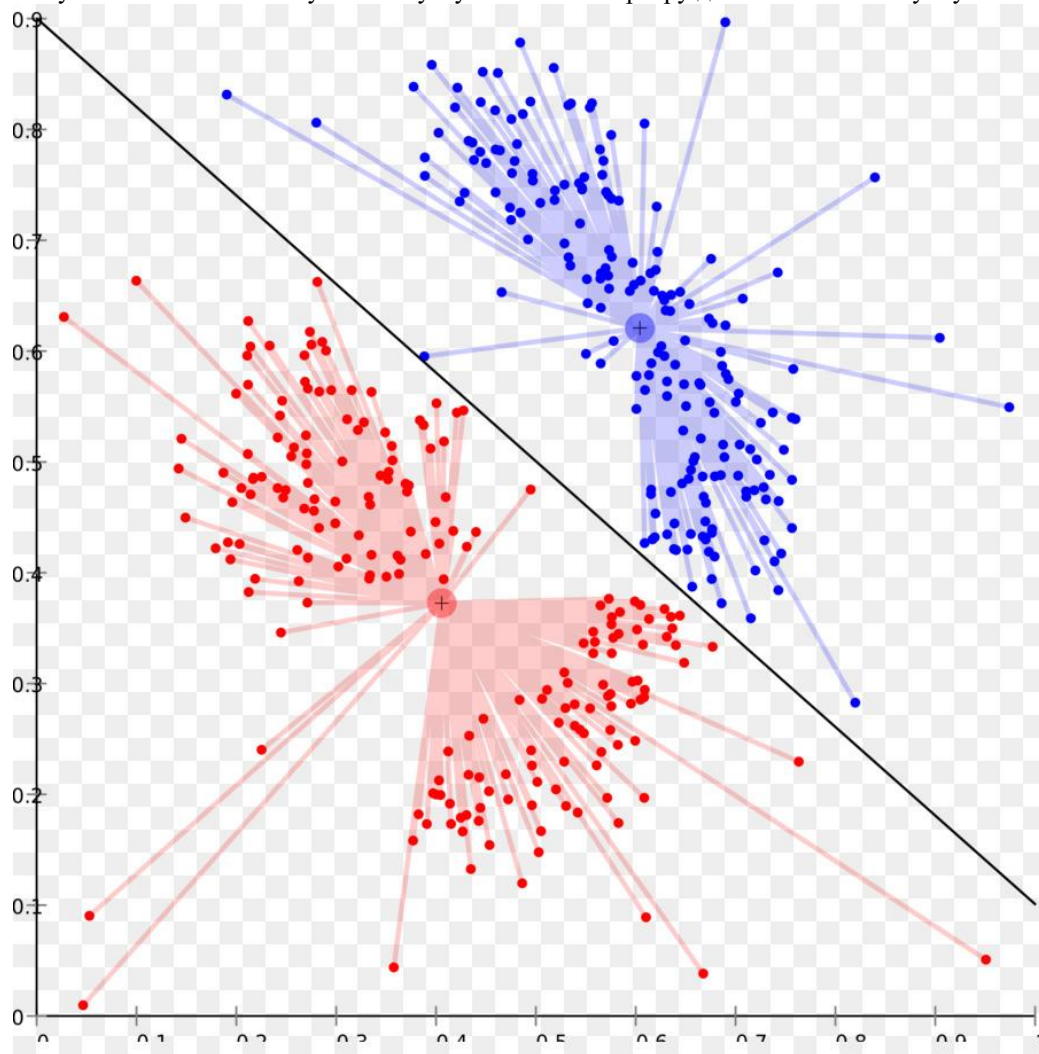


Рис. 2.1. Приклад кластерного аналізу

Застосування кластерного аналізу зводиться до наступних етапів:

1. Вибір вибірки об'єктів для кластеризації;
2. Визначення безлічі змінних, якими будуть оцінюватися об'єкти у вибірці. За потреби – нормалізація значень змінних;
3. Обчислення значень міри схожості між об'єктами;
4. Застосування методу кластерного аналізу створення груп подібних об'єктів (кластерів);
5. Подання результатів аналізу.

Після отримання аналізу результатів можливе коригування обраної метрики та методу кластеризації до отримання оптимального результату.

Кластерний аналіз виконує наступні основні завдання:

Кафедра інтелектуальних інформаційних систем
Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку

- розробка типології або класифікації;
- дослідження корисних концептуальних схем групування об'єктів.
- породження гіпотез на основі дослідження даних;
- перевірка гіпотез або дослідження для визначення, чи дійсно групи, виділені тим чи іншим способом, присутні в наявних даних.

Кластеризація актуальна, якщо вихідна вибірка дуже велика. В результаті кожного кластера залишається по одному типовому представнику. Кількість кластерів може бути будь-яким – тут важливо забезпечити максимальну схожість об'єктів усередині кожної групи.

Розбиття об'єктів на кластери дозволяє додати додаткову ознаку кожному об'єкту. Так, якщо в результаті кластерного аналізу виявилось, що певний покупець відноситься до першого кластера, і ми знаємо, що перший кластер - це кластер людей, які витрачають велику кількість грошей на покупки по середах, то можна сказати, що це покупець купує продукти в основному по середам.

Кластерний аналіз застосовують у різних сферах:

- у маркетингу - для сегментування клієнтів, конкурентів, дослідження ринку;
- медицині – для кластеризації симптомів, захворювань, препаратів;
- біології - для класифікації тварин та рослин;
- соціології - для розбиття респондентів на однорідні групи;
- комп'ютерні науки - для групування результатів при пошуку сайтів, файлів та інших об'єктів.

Оскільки поняття «кластеру» не може бути точно визначено, то це є однією з причин чому існує так багато різних методів кластеризації. Але є і спільна риса — це об'єднання схожих об'єктів у групи. Однак, різні дослідники використовують різні моделі кластерів і для кожної з цих моделей можуть бути застосовані різні алгоритми.

Поняття кластера, які отримуються у різних алгоритмах, різняться властивостями. Розуміння цих «кластерних моделей» є ключовим для розуміння відмінностей між різними алгоритмами.

Популярними категоріями (моделями) кластеризації вважаються часткова кластеризація, ієрархічна та кластеризація на основі щільності (Рис. 2.2).

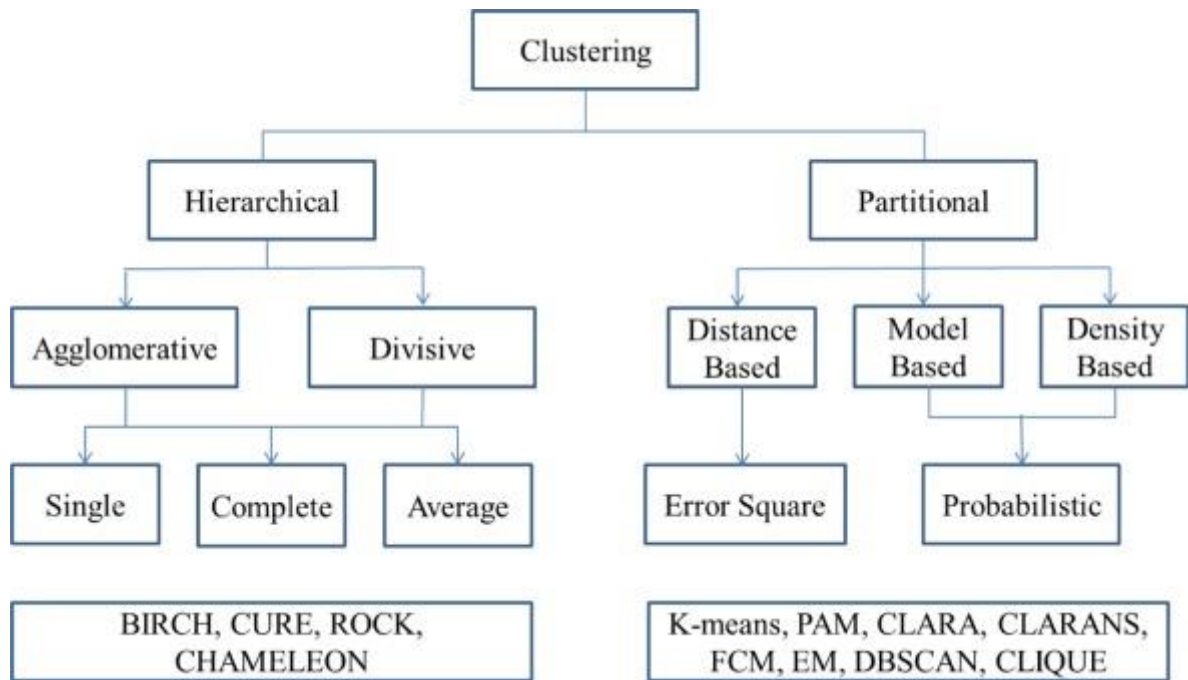


Рис. 2.2. Категорії кластеризацій

Часткова кластеризація ділить об'єкти даних на групи, що не перекриваються. Іншими словами, жоден об'єкт не може бути членом більш ніж одного кластера, і кожен кластер повинен мати принаймні один об'єкт (Рис. 2.3).

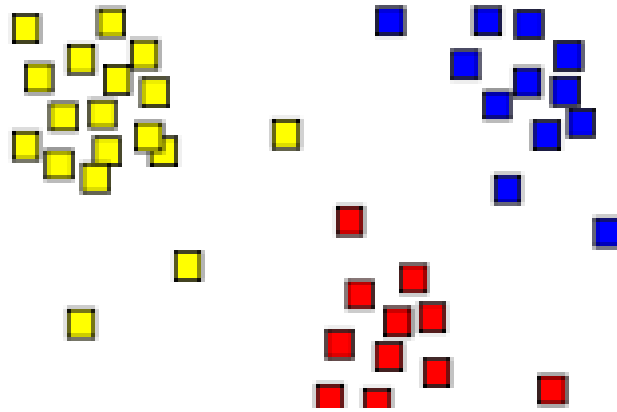


Рис. 2.3. Часткова Кластеризація

Ці методи вимагають від користувача вказати кількість кластерів, позначену змінною k . Багато алгоритмів роздільної кластеризації працюють за допомогою ітераційного процесу, щоб призначити підмножини точок даних у k кластерів. Два приклади алгоритмів роздільної кластеризації - k -середніх і k -медоїдів.

Обидва ці алгоритми є недетермінованими, що означає, що вони можуть давати різні результати від двох окремих запусків, навіть якщо запуски базувалися на одному вхідному даних.

Методи роздільної кластеризації мають кілька сильних сторін:

- добре працюють, коли кластери мають сферичну форму;
- масштабовані за складністю алгоритму.

У них також є кілька недоліків:

- не дуже підходять для кластерів зі складними формами та різними розмірами;
- вони руйнуються при використанні з кластерами різної щільності.

Ієрархічна кластеризація (Рис. 2.4) визначає призначення кластерів шляхом побудови ієрархії. Це реалізується за допомогою підходу знизу вгору або зверху вниз:

- агломеративна кластеризація – це підхід знизу вгору. Він об'єднує дві найбільш схожі точки, поки всі точки не будуть об'єднані в один кластер;

- розділова кластеризація – це підхід зверху вниз. Він починається з усіх точок як одного кластера і розбиває найменш подібні кластери на кожному кроці, поки не залишаться лише окремі точки даних.

Ці методи створюють деревоподібну ієрархію точок, яка називається дендрограмою. Подібно до роздільної кластеризації, в ієрархічній кластеризації кількість кластерів (k) часто заздалегідь визначається користувачем. Кластери призначаються шляхом розрізання дендрограми на заданій глибині, що призводить до k груп менших дендрограм.

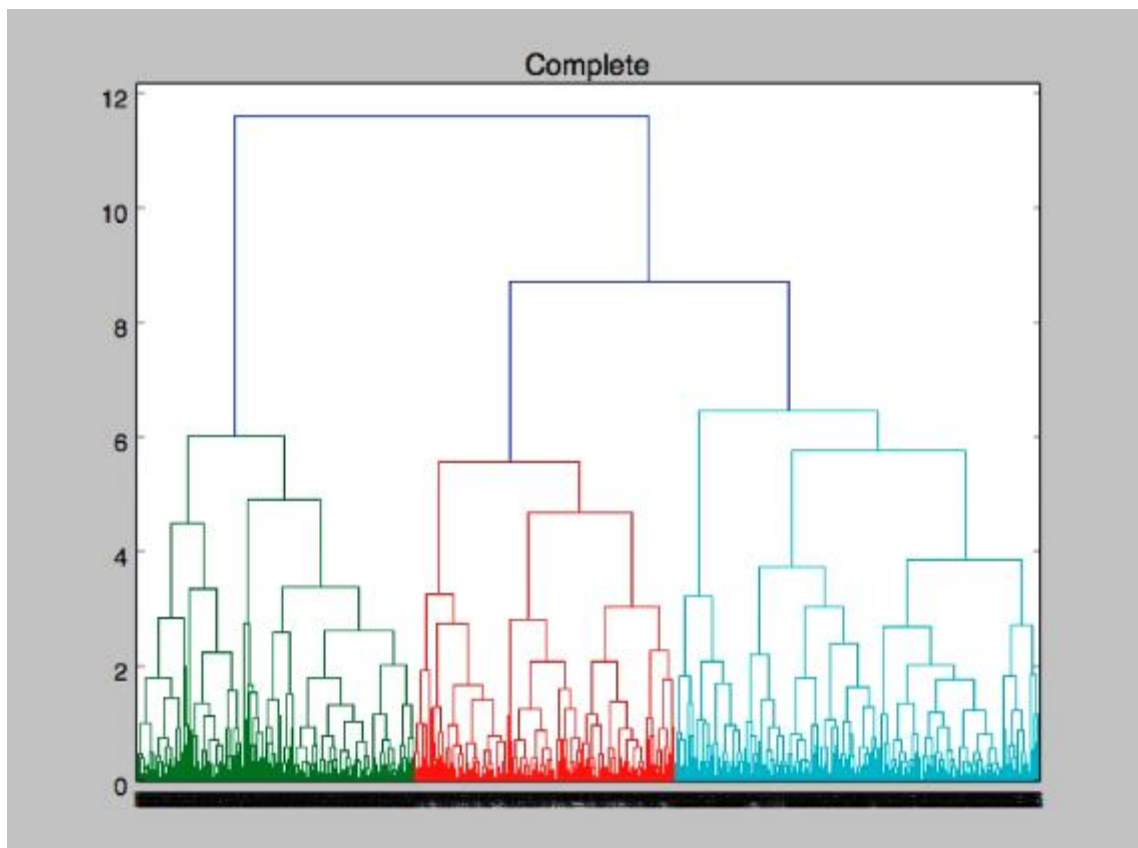


Рис. 2.4. Приклад ієрархічної кластеризації

На відміну від багатьох методів роздільної кластеризації, ієрархічна кластеризація є детермінованим процесом, що означає, що призначення кластерів не зміняться, якщо ви двічі запустите алгоритм для одних і тих самих вхідних даних.

Кафедра інтелектуальних інформаційних систем
Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку
До сильних сторін методів ієрархічної кластеризації можна віднести

наступне:

- часто розкривають найдрібніші деталі зв'язків між об'єктами даних;
- забезпечують інтерпретаційну дендрограму.

Слабкими сторонами методів ієрархічної кластеризації є наступне:

- дорогі в обчислювальному відношенні щодо складності алгоритму;
- чутливі до шуму та викидів.

Кластеризація на основі щільності визначає призначення кластерів на основі щільності точок даних у регіоні. Кластери призначаються там, де є висока щільність точок даних, розділених областями з низькою щільністю (Рис. 2.5).

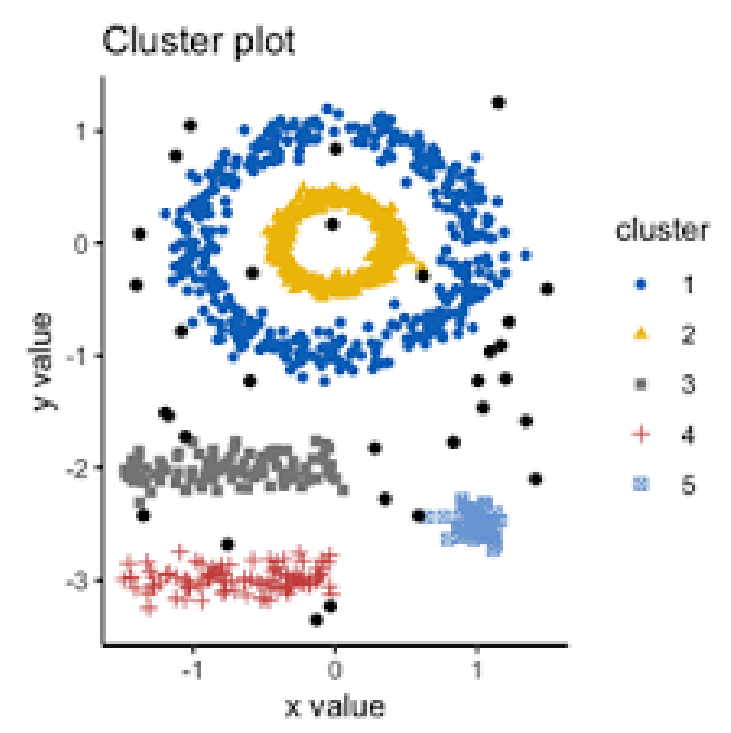


Рис. 2.5. Приклад кластеризації на основі щільності

На відміну від інших категорій кластеризації, цей підхід не вимагає від користувача вказувати кількість кластерів. Натомість існує параметр на основі

Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку відстані, який діє як настроюваний поріг. Цей поріг визначає, наскільки близькими повинні бути точки, щоб вважатися членом кластера.

Приклади алгоритмів кластеризації на основі щільності включають просторову кластеризацію додатків із шумом на основі щільності, або DBSCAN, і точки впорядкування для визначення структури кластеризації, або OPTICS.

Сильні сторони методів кластеризації на основі щільності включають наступне:

- відмінно визначають скупчення несферичних форм;
- стійкі до зовнішніх факторів.

Слабкі сторони методів кластеризації на основі щільності включають наступне:

- погано підходять для кластеризації в просторах з великою вимірністю;
- важко визначити скупчення різної щільності.

Кластеризація на основі моделі розподілу. У цьому типі кластеризації кластери техніки формуються шляхом ідентифікації за ймовірністю всіх точок даних кластера, що надходять від одного і того ж розподілу (нормального, гауссового). Найпопулярнішим алгоритмом у цьому виді техніки є кластеризація очікування-максимізації (EM) за допомогою Гауссових моделей сумішей (GMM).

Нормальні методи кластеризації, такі як ієрархічна кластеризація та кластеризація розділів, не базуються на формальних моделях, KNN при кластеризації розділів дає різні результати з різними значеннями K . Оскільки KNN та KMN вважають середніми для центру кластера, це не найкраще підходить в деяких випадках з Гауссовими моделями сумішей, існує припущення, що точки даних розподіляються по Гауссу, з цього випливає є два параметри для опису форми середнього кластеру та стандартного відхилення. Таким чином, для кожного кластеру призначається одна гауссова розподіл, щоб отримати оптимальні значення цих параметрів (середнє та стандартне

Кафедра інтелектуальних інформаційних систем
 Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку
 відхилення), використовується алгоритм оптимізації, який називається
 Максималізація очікування. Так виглядає EM - GMM після тренувань.

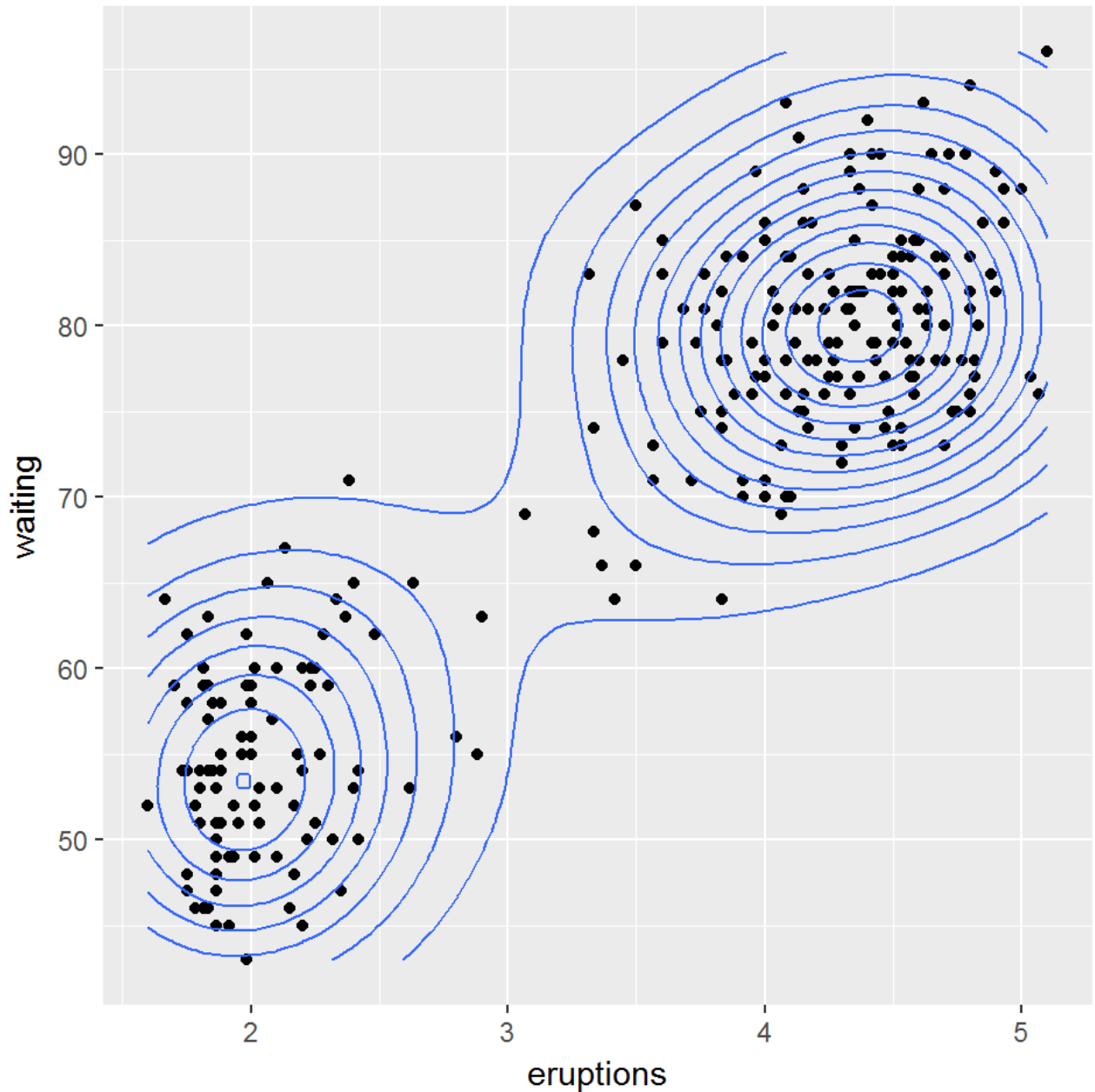


Рис. 2.6. Приклад кластеризації на основі моделі розподілу

Нечітке скупчення. Належить до гілки методів кластеризації м'яких методів, тоді як всі вищезазначені методи кластеризації належать до методів кластеризації жорстких методів. У цьому типі техніки кластеризації точок, близьких до центру, можливо, частина іншого кластера вищою мірою, ніж точки на краю цього ж кластера. Ймовірність приналежності точки до даного

Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку кластеру - це значення, яке лежить від 0 до 1. Найпопулярнішим алгоритмом у цьому виді техніки є FCM (нечіткий алгоритм C-means). Тут центроїд кластера обчислюється як середнє значення усіх балів, зважених на їх вірогідність приналежності до кластеру (Рис. 2.7).

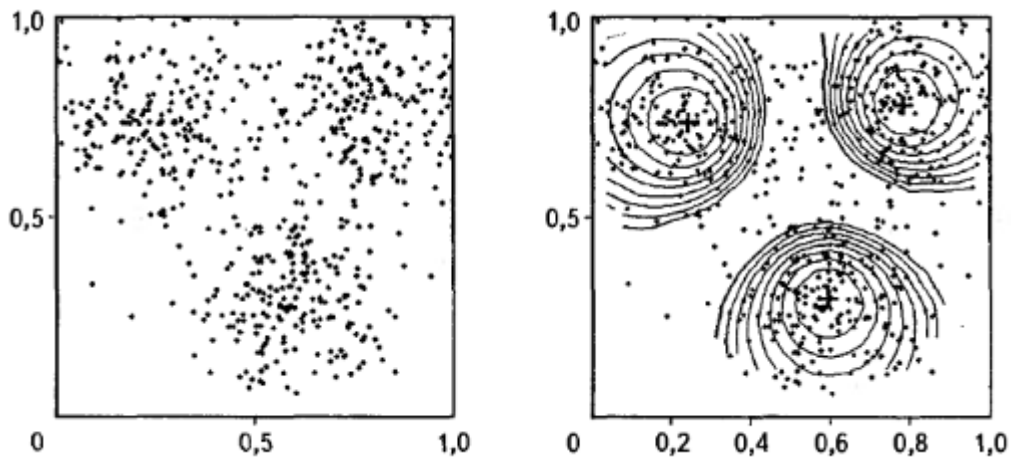


Рис. 2.7. Приклад нечіткого скупчення

Серед основних алгоритмів кластеризації обрано такі алгоритми для дослідження: k -середніх (модель зв'язності) та пошарова кластеризація (графова модель).

2.2 Аналіз обраних методів кластеризації

Метод k -середніх – це метод кластерного аналізу, мета якого є поділ m спостережень (з простору R^n) на k кластерів, при цьому кожне спостереження відноситься до кластера, до центру (центроїду) якого воно найближче.

Як міру близькості використовується Евклідова відстань:

$$p(x, y) = \|x - y\| = \sqrt{\sum_{p=1}^n (x_p - y_p)^2},$$

де $x, y \in R^n$

Отже, розглянемо низку об'єктів $(x^{(1)}, x^{(2)}, \dots, x^{(m)}), x^{(j)} \in R^n$.

Метод k -середніх поділяє m об'єктів на k груп (або кластерів) ($k \leq m$) $S = \{S_1, S_2, \dots, S_k\}$, щоб мінімізувати сумарне квадратичне відхилення точок кластерів від центроїдів цих кластерів:

$$\min \left[\sum_{i=1}^k \sum_{x^{(j)} \in S_i} \|x^{(j)} - \mu_i\|^2 \right],$$

Де $x^{(j)} \in R^n, \mu_i \in R^n$,

μ_i - центроїд для кластера S_i .

При визначеній мірі близькості до центроїду розбиття об'єктів на кластери зводиться до визначення центроїдів цих кластерів. Число k кластерів задається дослідником заздалегідь.

Слід розглянути початковий набір k -середніх μ_1, \dots, μ_k у кластерах S_1, S_2, \dots, S_k . На першому етапі центроїди кластерів вибираються випадково або за певним правилом (наприклад, вибрати центроїди, що максимізують початкові відстані між кластерами).

Відносимо спостереження тим кластерам, чиє середнє (центроїд) до них найближче. Кожне спостереження належить лише одного кластеру, навіть якщо його можна віднести до двох і більше кластерів.

Потім центроїд кожного i кластеру перераховується за таким правилом:

$$\mu_i = \frac{1}{S_i} \sum_{x^{(j)} \in S_i} x^{(j)}$$

Таким чином, алгоритм k -середніх полягає у перерахуванні на кожному кроці центроїду для кожного кластера, отриманого на попередньому кроці (Рис. 2.6).

Алгоритм зупиняється, коли значення μ_i не змінюються $\mu_i^{\text{шаг } t} = \mu_i^{\text{шаг } t+1}$.



Рис. 2.6. Процес обробки даних а) початковий набір даних; б) проміжний етап кластеризації; в) кінцевий результат кластеризації даних

Важливо: Неправильний вибір початкової кількості кластерів k може призвести до некоректних результатів. Саме тому при використанні методу k -середніх важливо спочатку провести перевірку відповідного числа кластерів для набору даних.

Особливості методу k -середніх:

1. Як метрика використовується Евклідова відстань;
2. Число кластерів заздалегідь не відоме і вибирається дослідником заздалегідь;
3. Якість кластеризації залежить від початкового розбиття.

Переваги:

Головні переваги методу k -середніх — його простота та швидкість виконання. Метод k -середніх більш зручний для кластеризації великої кількості спостережень, ніж метод ієрархічного кластерного аналізу (у якому дендограми стають перевантаженими і втрачають наочність).

Недоліки:

Незважаючи на очевидні переваги методу, він має суттєві недоліки. Одним із недоліків даного методу є порушення умови зв'язності елементів одного кластера, тому розвиваються різні модифікації методу, а також його нечіткі аналоги (англ. *fuzzy k-means methods*), у яких на першій стадії алгоритму допускається приналежність одного елемента множини до декількох кластерів (із різним ступенем приналежності).

Також, існують інші проблеми:

- результат класифікації сильно залежить від випадкових початкових позицій кластерних центрів;
- алгоритм чутливий до викидів, які можуть викривлювати середнє;
- Кількість кластерів повинна бути заздалегідь визначена дослідником.

Кластеризація k -середніх потребує чисельних даних, за роботами дослідників видно, що використовувались такі дані при аналізі: рівень звучності, тембр, акустичність, енергетика тощо. Також існує проблема із вхідним значенням кількості кластерів, на які поділятиметься музика, один кластер з котрих стане рекомендацією для додатку.

Ця система може бути ненадійною, оскільки існуює безліч музикальних композицій, які можуть мати спільні характеристики, проте є різними по жанру та виконанню. Через цю різницю можливі помилкові відношення певних пісень до кластеру та їх рекомендацій як оптимальний варіант. Існує складність урахування вподобань окремого користувача, оскільки дана змінна ускладнює роботу методу кластеризації k -середніх. Для підтвердження ефективності слід обрати досить велику групу тестувальників та час на цей тест.

Пошарова кластеризація. Алгоритм пошарової кластеризації заснований на виділенні зв'язкових компонентів графа (Рис. 2.7) на певному рівні відстаней між об'єктами (вершинами) . Рівень відстані визначається порогом відстані c . Наприклад, якщо відстань між об'єктами $0 \leq p(x, x') \leq 1$, та $0 \leq c \leq 1$.

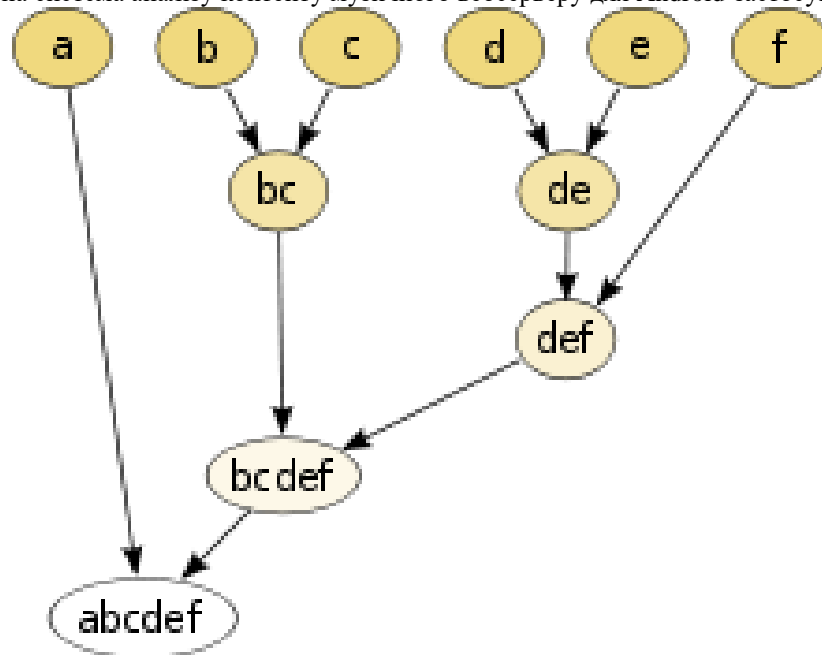


Рис. 2.7. Пошарова кластеризація

Алгоритм пошарової кластеризації формує послідовність підграфів графа G , які відображають ієрархічні зв'язки між кластерами:

$$G^0 \supseteq G^1 \supseteq \dots \supseteq G^m,$$

де $G^t = (V, E^t)$ - Граф на рівні t ,

$$E^t = \{e_{ij} \in E: p_{ij} \leq c_t\},$$

c_t – t -ий поріг відстані,

m – кількість рівнів ієрархії,

$G^0 = (V, \emptyset)$, \emptyset – порожня множина ребер графа, одержуване при $t^0 = 1$,

$G^m = G$, тобто граф об'єктів без обмежень на відстань (довжину ребер графа), оскільки $t_m = 1$.

За допомогою зміни порогів відстані $\{c^0, \dots, c^m\}$, де $0 = c^0 < c^1 < \dots < c^m = 1$, можна контролювати глибину ієрархії одержуваних кластерів. Таким чином, алгоритм пошарової кластеризації здатний створювати як плоске розбиття даних, так і ієрархічне (Рис. 2.5).

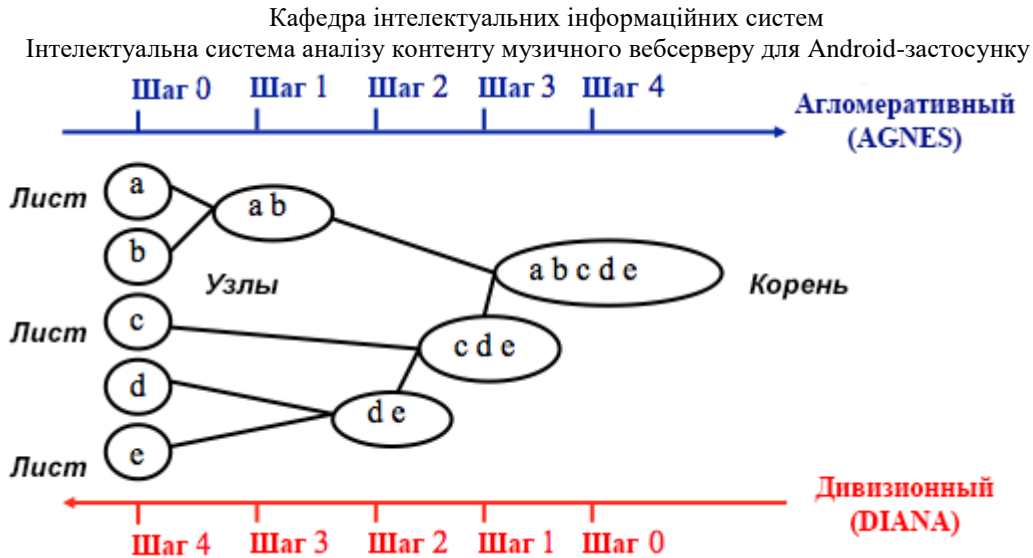


Рис. 2.5. Способи одержання кластерів

Пошарова кластеризація є досить ефективним варіантом для покращення рекомендаційних систем. Він не має суттєвих недоліків та підходить для опрацювання даних від окремого користувача, оскільки можна використовувати для аналізу дані по тегам із музики, що прослуховує користувач самостійно, не потребує великої кількості музичних композицій та багато часу для тестування ефективності системи. Проте, потрібно створювати музичні додатки, які заповнюють музику тегам.

Цей метод підходить для кластеризації будь яких речей, які не мають якихось чисельних характеристик та не має складності в аналізі як при застосуванні методу кластеризації k -середніх.

Отже, для рекомендаційної системи музичних композицій слід застосувати метод пошарової кластеризації.

Висновки до розділу 2

Розглянуті математичні моделі алгоритмів кластеризації. Більш детально розглянуто центроїдну та графову моделі, їх методи застосування. Визначено, що метод k -середніх не підходить для аналізу музичних композицій за певними тегамі, оскільки існує складність їх чисельного класифікування для задіяння в алгоритмі фільтрування музики за цим методом, оскільки він підходить для детального аналізу саме музичних технічних характеристик. Для аналізу

Кафедра інтелектуальних інформаційних систем
Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку
застосування даного алгоритму залучено дані музичного контенту із
чисельними характеристиками із допомогою API Spotify.

Для інтелектуальної системи рекомендацій, кластери яких не мають якихось чисельних характеристик було обрано метод графової моделі, а саме пошарову кластеризацію. Цей метод підходить для кластеризації будь яких речей, які не мають якихось чисельних характеристик та не має складності в аналізі як при застосуванні методу кластеризації k-середніх.

3 МОДЕЛЮВАННЯ ТА ДОСЛІДЖЕННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ. ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ

3.1 Аналіз методу k-середніх

Щоб краще зрозуміти, за чим групуються жанри, необхідно спочатку визначити аудіо-функції. Використовувались такі аудіофункції, визначені @Spotify:

акустичність: міра довіри від 0,0 до 1,0 щодо того, чи є трек акустичним. 1.0 означає високу впевненість, що трек є акустичним.

танцювальність: танцювальність описує, наскільки композиція підходить для танцю, на основі комбінації музичних елементів, включаючи темп, стабільність ритму, силу удару та загальну регулярність. Значення 0,0 є найменш танцювальним, а 1,0 — найбільш танцювальним.

Енергія є мірою від 0,0 до 1,0 і являє собою перцептивну міру інтенсивності та активності. Як правило, енергійні треки відчуються швидко, гучно та шумно. Наприклад, дет-метал має високу енергію, тоді як прелюдія Баха має низькі бали за шкалою. Характеристики сприйняття, що сприяють цьому атрибуту, включають динамічний діапазон, сприйману гучність, тембр, швидкість початку та загальну ентропію.

інструментальність: передбачає, чи не містить трек вокал. Звуки «Ох» і «Ааа» в цьому контексті розглядаються як інструментальні. Реп або розмовні композиції є явно «вокальними». Чим ближче значення інструментальності до 1,0, тим більша ймовірність, що трек не містить вокального вмісту. Значення вище 0,5 призначені для представлення інструментальних треків, але впевненість вища, коли значення наближається до 1,0.

живість: Визначає присутність аудиторії у записі. Більш високі значення живучості представляють підвищену ймовірність того, що трек було виконано наживо. Значення вище 0,8 забезпечує високу ймовірність того, що трек працює.

мовленнєвість: Speechiness виявляє наявність вимовлених слів у доріжці. Чим більше виключно мовленнєвий запис (наприклад, ток-шоу,

Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку аудіокнига, вірші), тим ближче до 1,0 значення атрибута. Значення вище 0,66 описують треки, які, ймовірно, повністю складаються з вимовлених слів. Значення від 0,33 до 0,66 описують треки, які можуть містити як музику, так і мовлення, як у розділах, так і пошарово, включаючи такі випадки, як реп-музика. Значення нижче 0,33, швидше за все, представляють музику та інші треки, які не схожі на мовлення.

темп: загальний оцінений темп треку в ударах за хвилину (BPM). У музичній термінології темп — це швидкість або темп певної п'єси і впливає безпосередньо із середньої тривалості удару.

валентність: показник від 0,0 до 1,0, що описує музичний позитив, який передає трек. Доріжки з високою валентністю звучать більш позитивно (наприклад, радісно, весело, ейфорично), тоді як треки з низькою валентністю звучать більш негативно (наприклад, сумний, пригнічений, злий).

Дані з темпу як і інші дані було нормалізовано в межах 0,0–1,0.

Застосування кластеризації K-середніх на всіх напрямках. Деякі жанри були об'єднані в один, оскільки активність їх прослуховування були на низькому рівні та заважали б в проходженні аналізу (наприклад, у фолку є багато пісень, а у блюзу його немає. Блюз найближчий до народного за відстанню, тому їх було об'єдано в фолк/блюз). Після застосування кластеризації K-середніх було створено теплову карту значень звукових функцій кожного центроїда кластера (Рис. 3.1).

Кафедра інтелектуальних інформаційних систем
Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку

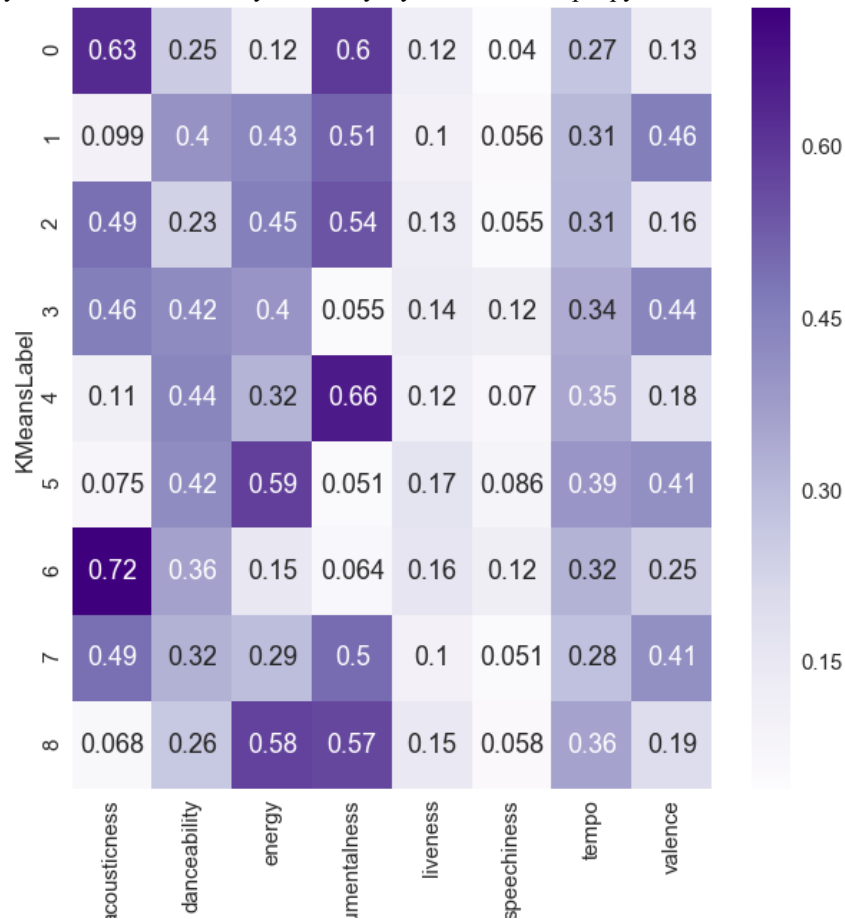


Рис. 3.1. Теплова карта значень звукових функцій за міткою K-середніх.

Враховуючи центроїди кластерів із K-Means, можна побачити значення, які характеризують кожну мітку K-Means. На основі значень з карти тепла було створено 9 груп музичних композицій:

- КМ0: високоакустичний та інструментальний. Низька танцювальність, енергетичний темп, валентність. «Повільна й похмура акустика»;
- КМ1: Дуже інструментальний і валентний. Середній темп, середина енергії. Низька акустичність і мовленнєвість. «Щасливі та танцювальні інструменти»;
- КМ2: Дуже інструментальна. Низька валентність, мовленнєвість. «Сумні інструментали»;
- КМ3: високовалентний. Мовленнєвий. Низька інструментальність. «Жорсткі пісні з веселим вокалом»;

- КМ4: високоінструментальний, танцювальний, швидкий. Низька акустичність. «Швидкі та танцювальні інструменти»;
- КМ5: високоенергійний, валентний і швидкий. Відносно висока живучість. Низька акустичність та інструментальність. «Швидкий, бадьорий і веселий»
- КМ6: Дуже акустичний. Середньо-високі танцювальні. Мовленнєвий. Низький рівень енергії. «Повільний танець».
- КМ7: дуже валентний та інструментальний. Низький темп і мовленнєвість. «Щасливий і повільний»;
- КМ8: Висока енергія, темп та інструментальність. Низька акустичність і мовленнєвість. «Щасливі та оптимістичні інструменти».

Далі отриманий результат проходив через аналіз основних компонентів, щоб зменшити набір даних до двох вимірів. Вісь ознак є оціночним візуальним орієнтиром на основі величини та напрямку поясненої дисперсії (Рис. 3.2-3.3).

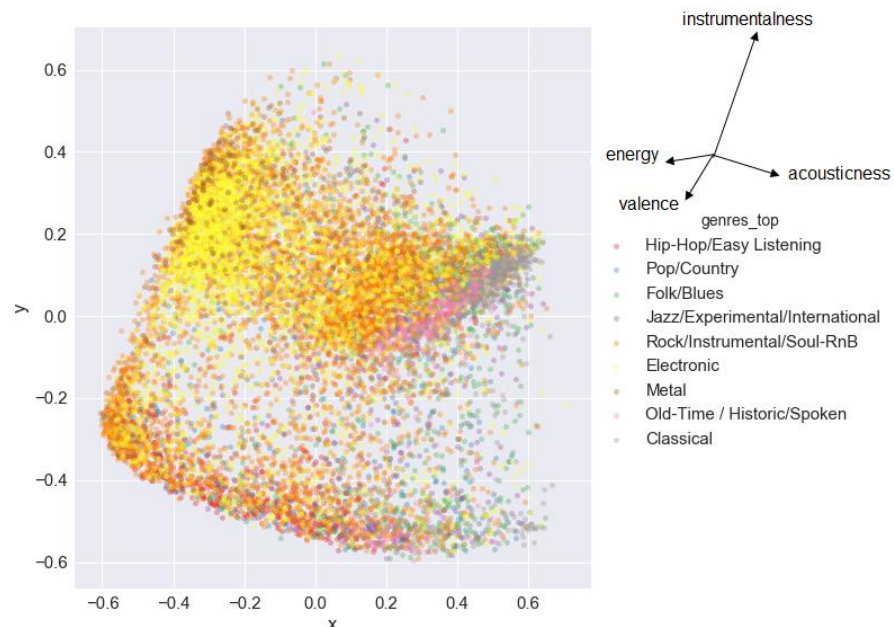


Рис. 3.2. Теплова карта значень звукових функцій за міткою K-Means. Порівняння між треками, згрупованими за звичайними музичними жанрами

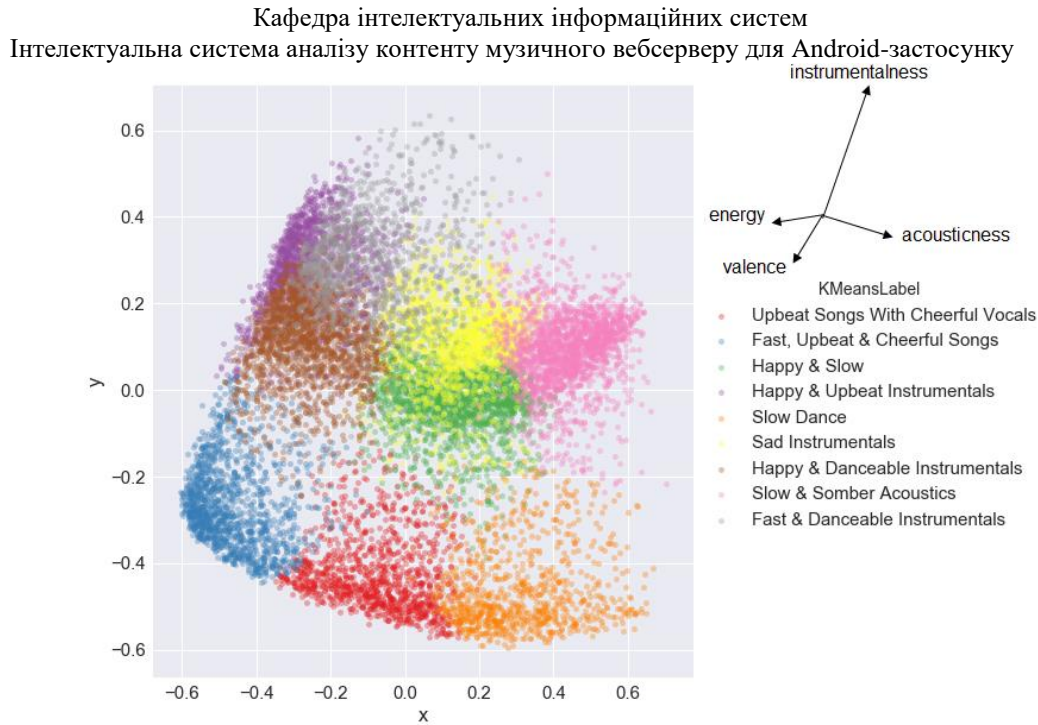


Рис. 3.3. Теплова карта значень звукових функцій за міткою K-Means.
Порівняння між треками, згрупованими за кластерами

При групуванні за звичайними жанрами (Рис. 3.2) видно деякі структуровані кластери, але в цілому досить не формативно. Фолк/блюз, класична та старовинна об'єднані разом із сильними акустичними цінностями та слабкими енергетичними цінностями. Здавалося, що метал розташовується вздовж осі інструментальності, але відхиляється у бік вищих значень енергії.

Правий сюжет з мітками K-Means (Рис. 3.3) помітно показав більше структури, ніж його звичайний аналог. Етикетки K-Means обґрунтовано описують деякі зі спостережуваних жанрів. Класика розумно поєднується з «Slow & Somber Acoustics». Фолк/блюз розділений на «Slow & Somber Acoustics» та «Happy & Danceable Instrumentals». Іншим є Metal — «Happy & Upbeat Instrumentals», що розташовуються вздовж верхнього лівого краю, що може вказувати на невелику кількість Power Metal у цьому наборі даних.

Зіставляючи деякі екземпляри кожної мітки, можна підтвердити деякі спостереження на графіку вище та побачити деякі, які приховані. K-Means розумно назвав велику кількість електронних, хіп-хоп, поп і кантрі «бадьорими піснями з веселим вокалом». Нижня клітинка точно говорить про те, що

Кафедра інтелектуальних інформаційних систем
 Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку
 електронна музика домінує у варіантах «Danceable Instrumentals».
 Застосування K-Means Clustering на скельних треках

Для поглибленого дослідження виконано поділ між піджанрами всередині певного жанру: рок. Було обрано $K = 5$ піджанрів: поп, інді-рок, психоделічний рок, панк і решту треків «звичайний» рок, які не підпадають під ці піджанри.

Як і раніше, я демонструю свої бентежно нетворчі інтерпретації найкращих припущень:

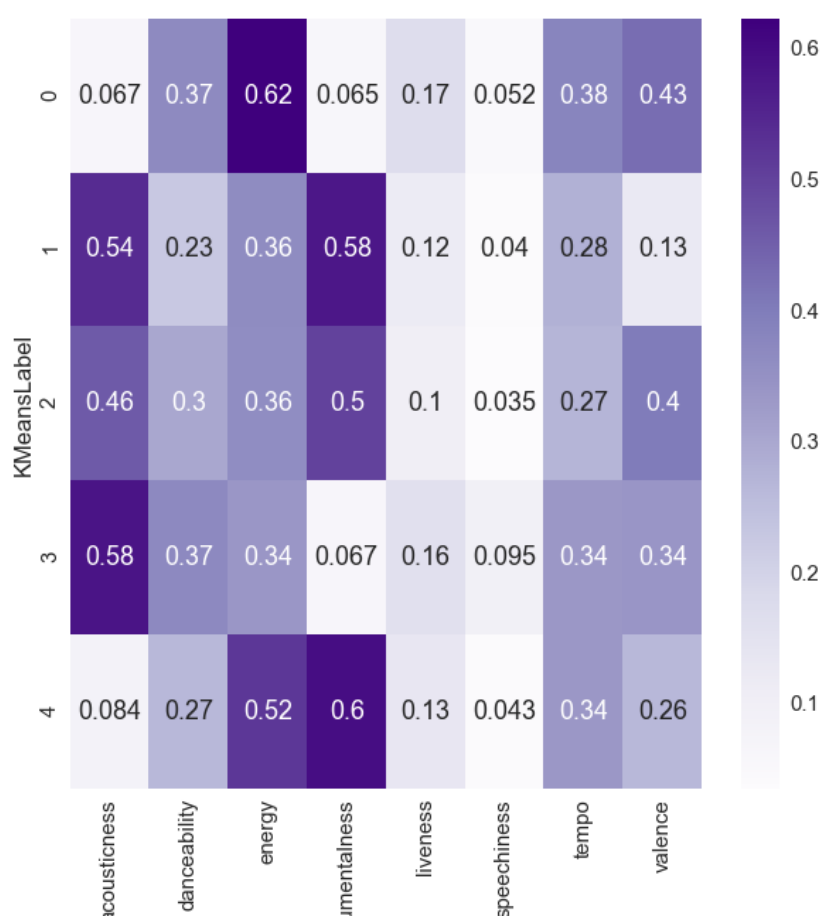


Рис. 3.4. Теплова карта значень звукових функцій від K-Means Label

- КМ0: Висока енергія, валентність, темп, танцювальність. Низька акустичність, інструментальність. Низьку мовленнєвість та низьку інструментальність я припишу синтезаторам. «Життєрадісний рок із синтезаторами, під які можна танцювати»

- КМ1: Висока акустичність та інструментальність. Не під силу танцю. Низька мовність, темп і валентність. «Повільний і депресивний рок»
- КМ2: Висока акустичність, інструментальність і валентність. Найнижча мовність і темп. «Повільний і веселий рок»
- КМ3: Висока акустичність і танцювальність. Низький рівень енергії. «Повільний танцювальний рок»
- КМ4: Висока інструментальність, темп і енергія. Низька акустичність, танцювальність. «Швидкий та енергійний рок»
- Застосувавши PCA ще раз, щоб уявити це, ми отримаємо такі графіки:

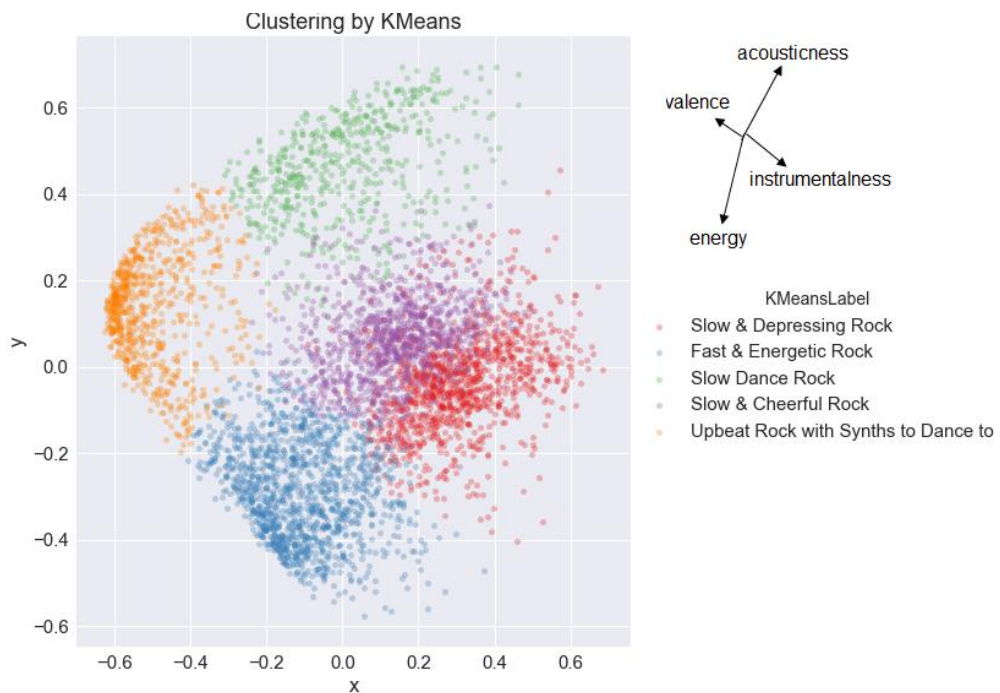


Рис. 3.5. Теплова карта значень звукових функцій від K-Means Label

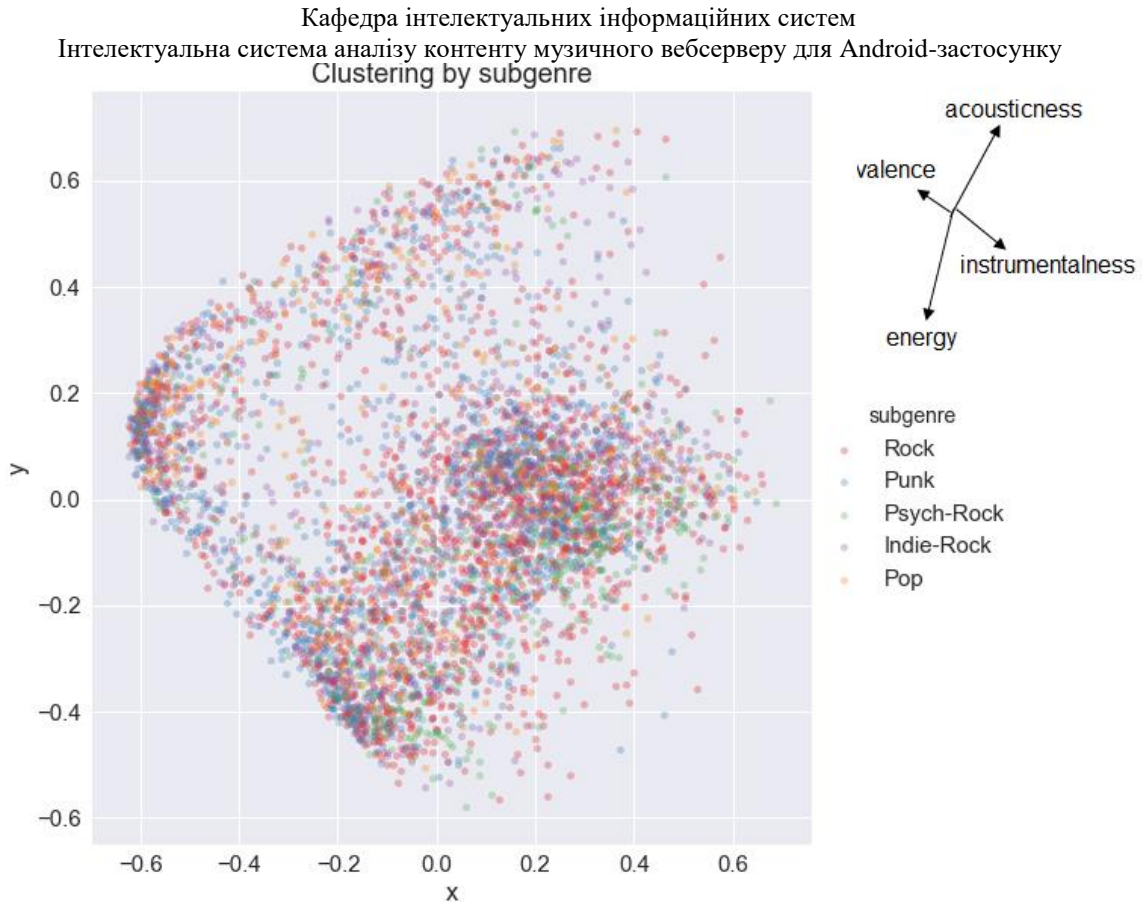


Рис. 3.6. Теплова карта значень звукових функцій від K-Means Label

Слід зазначити певну логічну групування піджанрів, наприклад, Psych-Rock, що розташовуються вздовж нижнього правого / низької валентності. У нижньому лівому куті дуже багато панку, що межує з високою енергією. Хоча Pop можна побачити на верхньому лівому / високому валентному та нижньому правому / низькому валентному краю, є дещо щільне скупчення Pop на самому лівому носі сюжету.

Правильний сюжет, позначений кластерними лейблами K-Means, підтверджує деякі інтуїтивні пари: Psych-Rock з «Slow & Depressing Rock», інді-рок в основному Psych-Rock, але більш веселий, Punk розділений на «Fast & Energetic Rock» та «Життєрадісний рок з синтезаторами, щоб танцювати».

З огляду на результати, можна припустити, що алгоритм працює, проте існують деякі складнощі в роботі, оскільки існує велика залежність від значень аудіофункцій, неправильно визначені параметри можуть знизити ефективність роботи, проте алгоритм досить непогано кластеризує музичний контент та

Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку об'єднує у піджанри (масив жанрів). Але не всім може сподобатись результат роботи, оскільки об'єднуючи в піджанри алгоритм може додати жанр, який не відповідає смаковим якостям людини, проте по характеристикам підходить.

3.2 Аналіз методу пошарової кластеризації

Техніка пошарової кластеризації спочатку кожна точка даних розглядається як окремий кластер. На кожній ітерації подібні кластери зливаються з іншими, поки не утвориться один кластер або K кластерів.

Основний алгоритм простий:

- Обчисліть матрицю близькості
- Нехай кожна точка даних є кластером
- Повторити: об'єднайте два найближчі кластери та оновіть матрицю близькості
- Поки не залишиться лише один кластер

Ключовою операцією є обчислення близькості двох кластерів

Щоб краще зрозуміти, давайте подивимося на графічне представлення техніки агломеративної ієрархічної кластеризації. Скажімо, у нас є шість точок даних $\{A, B, C, D, E, F\}$.

- Крок 1: На початковому етапі обчислюється близькість окремих точок і розглядаються всі шість точок даних як окремі кластери, як показано на зображенні нижче. Крок 2: На другому кроці подібні кластери об'єднуються разом і формуються як єдиний кластер. Для прикладу слід розглянути кластери «B,C» і «D,E» — подібні кластери, які об'єднані на другому кроці. Тепер у нас залишилося чотири кластери: A, BC, DE, F.
- Крок 3: Знову обчислюється близькість нових кластерів і об'єднуються подібні кластери, щоб утворити нові кластери A, BC, DEF.

- Крок 4: Обчисліть близькість нових кластерів. Кластери DEF і BC подібні і об'єднані разом, щоб утворити новий кластер. Тепер у нас залишилися два кластери A, BCDEF.
- Крок 5: Нарешті, всі кластери об'єднуються разом і утворюють єдиний кластер.

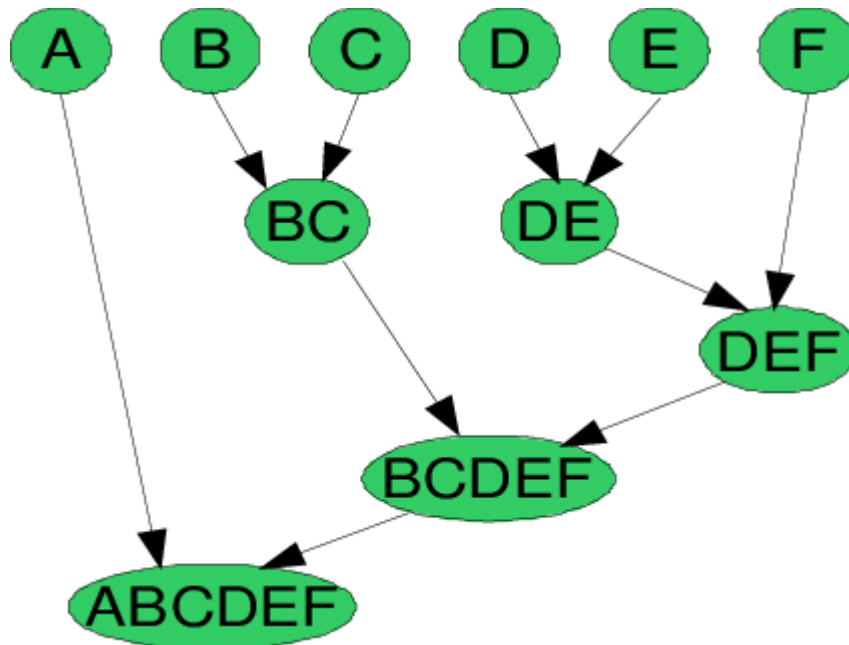


Рис. 3.7. Агломеративна ієрархічна техніка кластеризації

Техніку ієрархічної кластеризації можна візуалізувати за допомогою дендрограми.

Дендрограма — це деревоподібна діаграма, яка записує послідовності злиття або розбиття (Рис. 3.8-9).

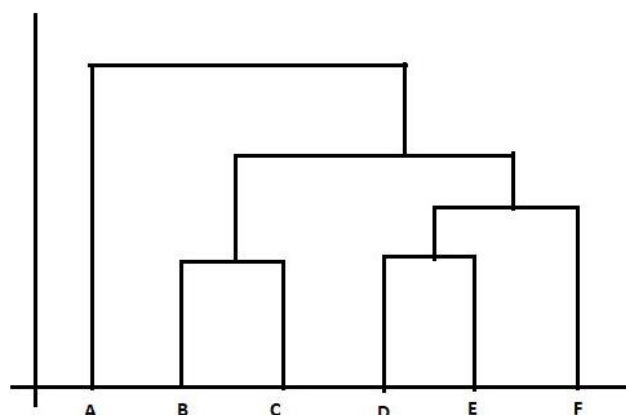


Рис. 3.8. Дендограма

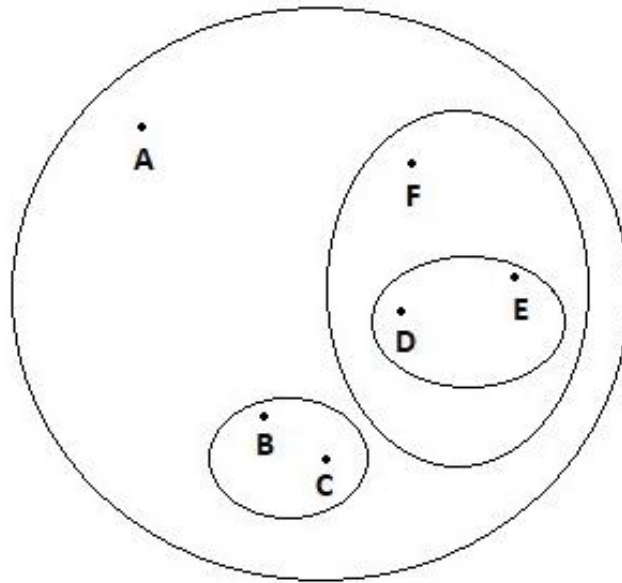


Рис. 3.9. Дендограма

Просторова та часова складність техніки ієрархічної кластеризації:

Складність простору: простір, необхідний для техніки ієрархічної кластеризації, дуже великий, коли кількість точок даних велика, оскільки нам потрібно зберігати матрицю подібності в ОЗП. Складність простору дорівнює порядку квадрата n .

Складність простору = $O(n^2)$, де n – кількість точок даних.

Часова складність: оскільки нам потрібно виконати n ітерацій і на кожній ітерації нам потрібно оновити матрицю подібності та відновити матрицю, часова складність також дуже висока. Часова складність дорівнює порядку куба n .

Часова складність = $O(n^3)$, де n – кількість точок даних.

Обмеження техніки ієрархічної кластеризації:

Для ієрархічної кластеризації немає математичної мети.

Усі підходи до розрахунку подібності між кластерами мають свої недоліки.

Висока складність простору та часу для ієрархічної кластеризації. Тому цей алгоритм кластеризації не можна використовувати, коли у нас є величезні дані. Проте використовуючи потужність серверних машин, цього можна уникнути, оскільки робота проводитиметься не на пристрої користувача.

За допомогою даного алгоритму можна з легкістю визначити найбільш активні групи жанрів чи піджанрів, автора, виконавця пісень, що наявні на сервері чи у користувача (Рис. 3.10)

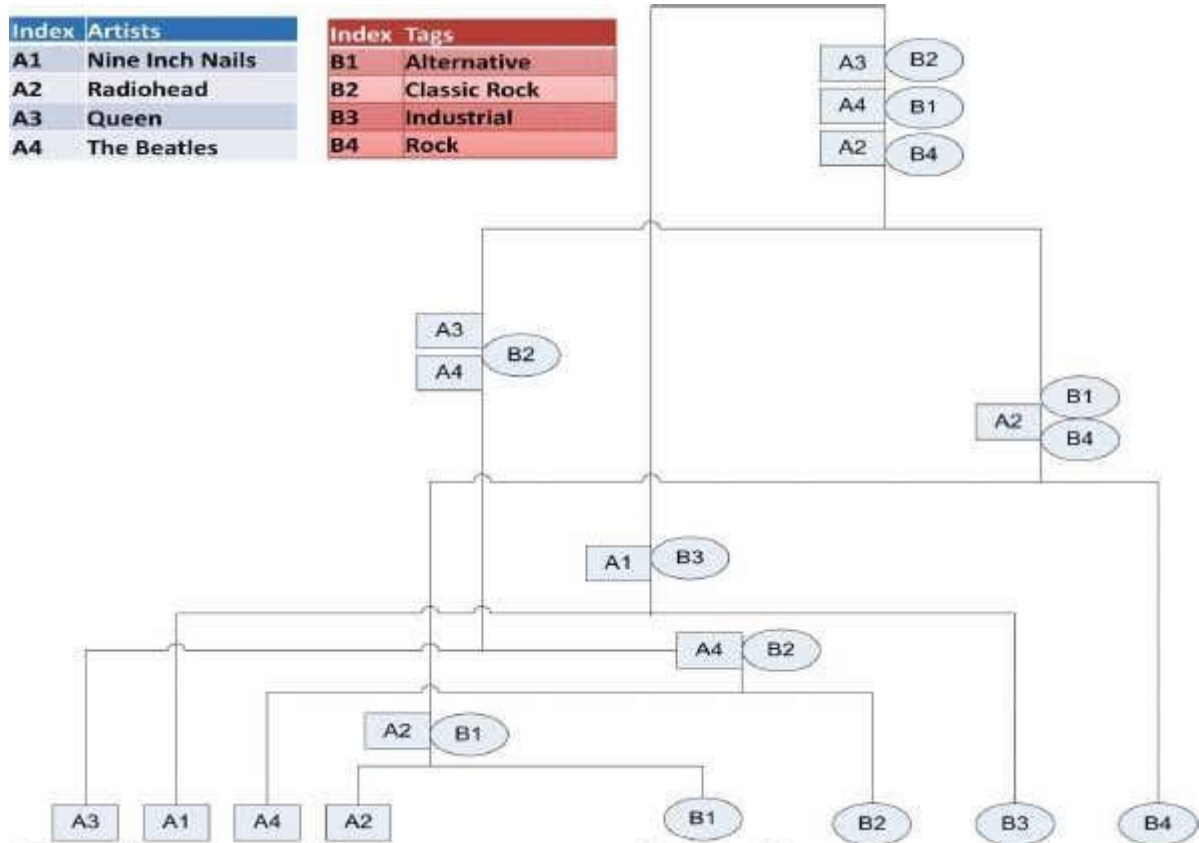


Рис. 3.10. Дендограма пісень

Визначивши кластери, існує можливість аналізувати об'єднані масиви кластерів для застосування в роботі рекомендаційних систем. Ця система потребуватиме даних щодо активності користувача. Далі за отриманими даними та даними активностей можна порівнювати масиви кластерів між собою для визначення найбільш активних масивів жанрів та рекомендувати контент із серверу, який має спільні масиви жанрів.

Висновки до розділу 3

В даному розділі проводилось дослідження обраних методів кластеризації музичних даних.

Розглянуто застосування методу k -середніх, оскільки існує велика залежність від значень аудіофункцій, неправильно визначені параметри можуть знизити ефективність роботи, проте алгоритм досить непогано кластеризує музичний контент та об'єднує у піджанри (масив жанрів). Проте не всім може сподобатись результат роботи, оскільки об'єднуючи в піджанри алгоритм може додати у кластер пісні, що не відповідають смаковим якість користувача, незважаючи на відповідні для кластеру аудіопараметри.

Розглянуто застосування пошарової кластеризації. За допомогою даного алгоритму визначаються активні групи жанрів чи піджанрів, автора, виконавця пісень, що наявні на сервері чи у користувача. За даними активностей можна порівнювати масиви кластерів між собою для визначення найбільш активних масивів жанрів та рекомендувати контент із серверу із спільними жанрами.

Кафедра інтелектуальних інформаційних систем
 Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку

4 МОДЕЛЮВАННЯ ТА ДОСЛІДЖЕННЯ ОТРИМАНИХ РЕЗУЛЬТАТІВ. ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ

4.1 Проектування інформаційного забезпечення

4.1.1 Діаграма прецедентів. Опис прецедентів

Для визначення можливостей ПЗ потрібно визначити прецеденти. Всі необхідні прецеденти відображені у рис. 4.1-4.11 та описані в табл. 4.1-4.8.

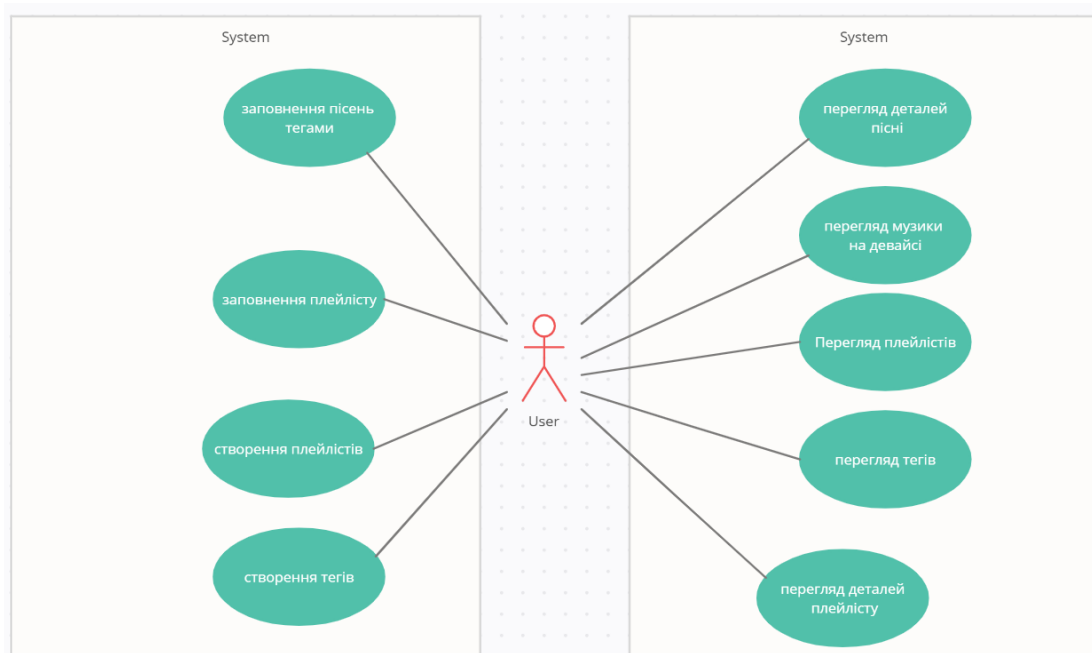


Рис. 4.1. діаграма прецедентів

Прецеденти – це випадок або подія, що відбулися у минулому та є прикладом або підставою для аналогічних дій у сьогоденні. За допомогою визначених прецедентів створюються плани та дії, які може виконати користувач при застосуванні ПЗ.

На рис. 3.2 та табл. 3.1 охарактеризовано поведінку користувача та ПЗ при його запуску. В цей час відбувається завантаження музичних даних з пам'яті телефону на головний екран.

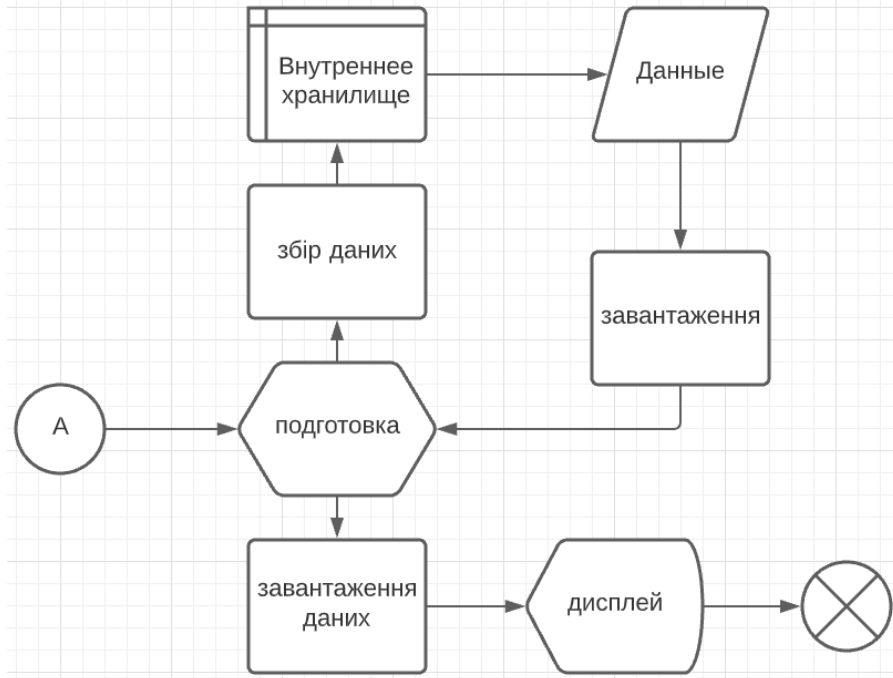


Рис. 4.2. діаграма активності прецеденту «перегляд пісень» прецеденту
«Перегляд музики на девайсі»

Таблиця 4.1

Прецедент «Перегляд пісень»

Характеристика	Опис
Короткий опис	Процес завантаження музичних композицій
Актори	Активні суб'єкти.
Передумови	На девайсі знаходяться музичні композиції.
Основний потік подій	Завантаження даних з музичних композицій та їх відображення на дисплеї
Альтернативний потік подій	Не має пісень, на дисплеї відображено пусті списки
Послідуюче	Дані пісень відображено у головному екрані, подальша робота додатку йде задовільно.

На рис 4.3 та табл. 4.2 описано поведінку системи при натисненні на певну пісню. При натисненні відкриється екран із деталями пісні.

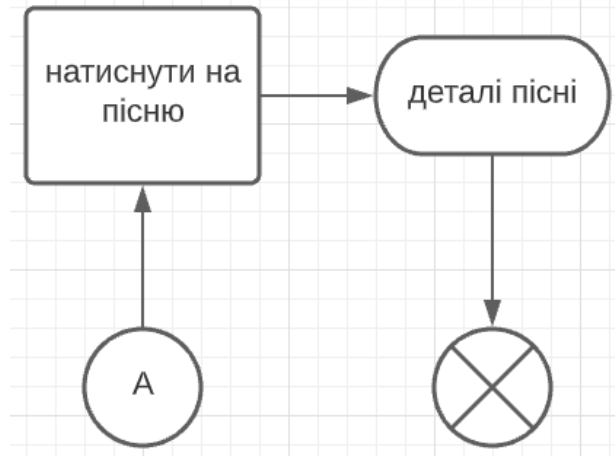


Рис. 4.3. діаграма активності «Відкрити деталі пісні» прецеденту «Перегляд деталей пісні»

Таблиця 4.2

Прецедент «Перегляд деталей пісні»

Характеристика	Опис
Короткий опис	Перегляд деталей пісень
Актори	Активні суб'єкти.
Передумови	На девайсі знаходяться музичні композиції.
Основний потік подій	Завантаження деталей пісні на дисплеї
Альтернативний потік подій	
Послідуюче	Детальна інформація пісні дисплеї, є можливість додати тег

Деталі роботи із плейлістами описується на рис. 4.4-4.7 та в табл. 4.4-4.5. за цими даними можна зробити висновок, що існує у користувача можливість створити, редагувати, переглядати плейліст.

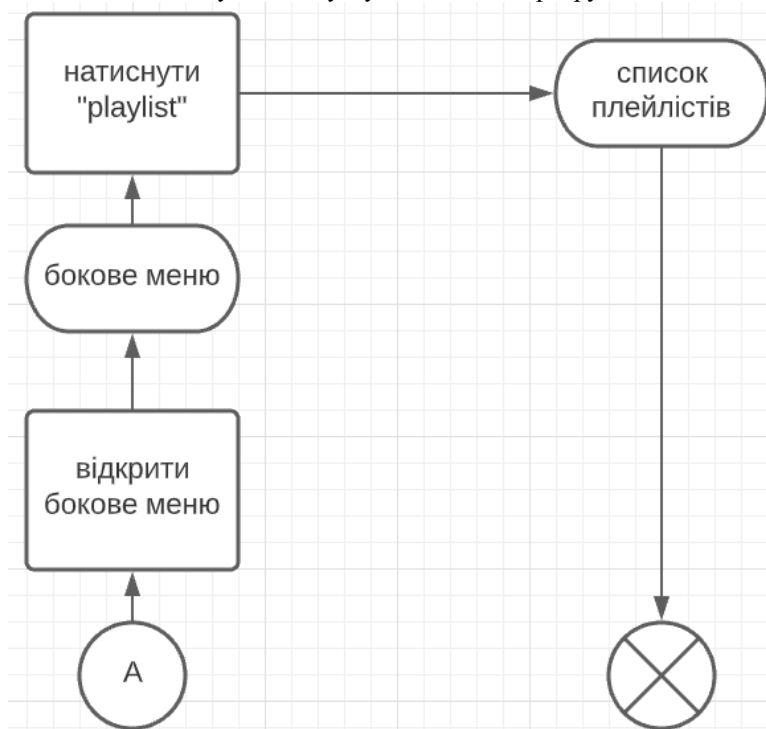


Рис. 4.4. діаграма активності «Переглянути наявні плейліст» прецеденту
«Перегляд плейлістів»



Рис. 4.5. діаграма активності «Відкрити деталі плейлісту» прецеденту
«Перегляд плейлістів»

Прецедент «Перегляд плейлістів»

Характеристика	Опис
Короткий опис	Перегляд плейлістів
Актори	Активні суб'єкти.
Передумови	Додаток запущено
Основний потік подій	Перегляд плейлістів
Альтернативний потік подій	Існуючих плейлістів немає
Послідує	Відкрити деталі плейлісту, програвати плейліст

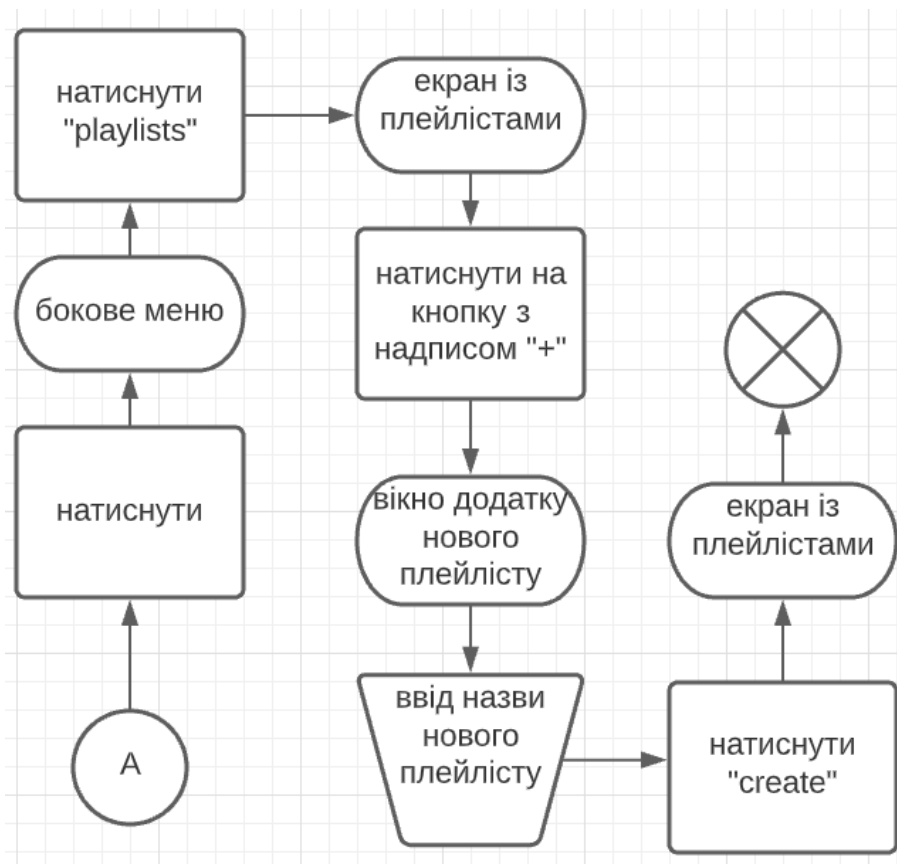


Рис. 4.6. діаграма активності «Створити новий плейліст» прецеденту
«Створення плейлістів»

Прецедент «Створення плейлістів»

Характеристика	Опис
Короткий опис	Створення нового плейлісту
Актори	Активні суб'єкти.
Передумови	Додаток запущено
Основний потік подій	Створення нового плейлісту
Альтернативний потік подій	Створення нового плейлісту відхилено
Послідуюче	Пісні можна додати в новий плейліст, його програвати

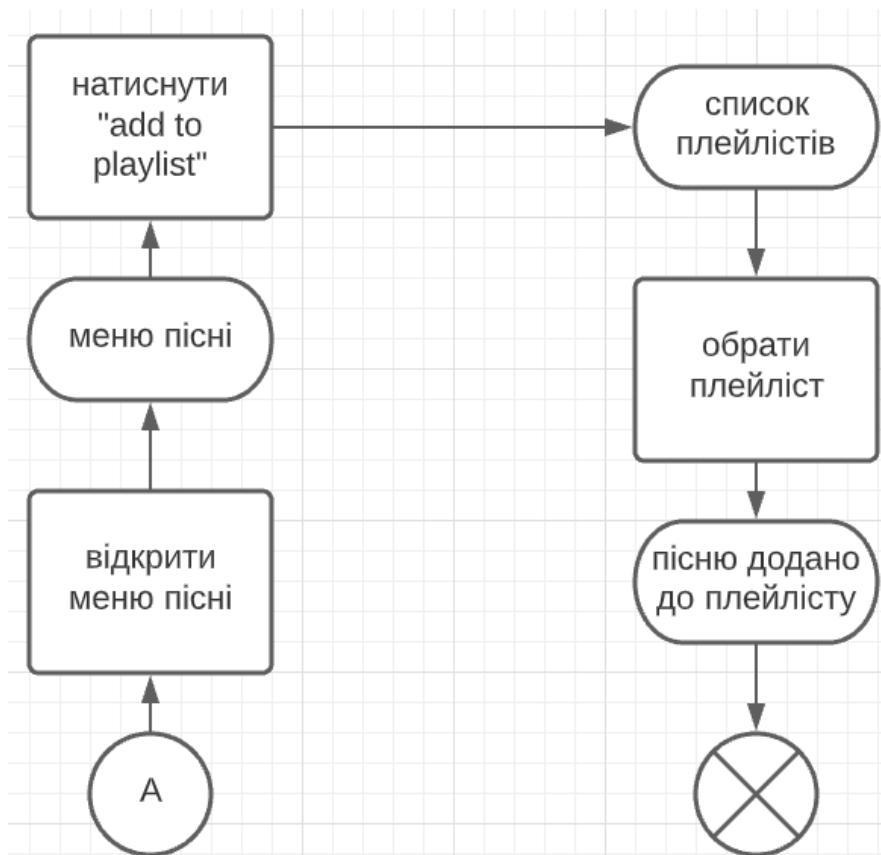


Рис. 4.7. діаграма активності «Додати пісню до плейлісту» прецеденту
«Заповнення плейлісту піснями»

Прецедент активності «Заповнення плейлісту піснями»

Характеристика	Опис
Короткий опис	Заповнення плейлісту
Актори	Активні суб'єкти системи.
Передумови	Додаток запущено
Основний потік подій	Відкриття меню пісні, вибір до якого плейлісту додати пісню
Альтернативний потік подій	
Послідує	Існує можливість програвати пісні певного плейлісту

Робота із тегами описана на рис. 4.8-4.10 та в табл. 4.6-4.8 описано поведінку роботи ПЗ із тегами. Теги можна створити, переглядати пісні за ними, заповнювати пісні тегами. За допомогою цих тегів і відбувається робота рекомендаційної системи. Для кожної пісні можна додати скільки завгодно тегів.

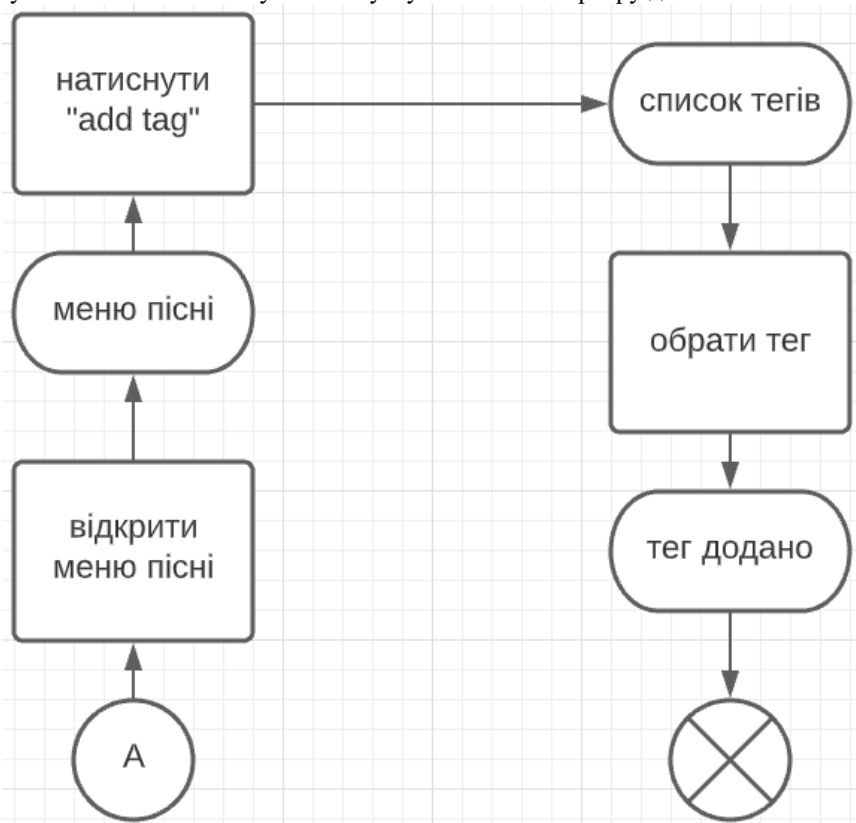


Рис. 4.8. діаграма активності «Додати тег до пісні» прецеденту «Заповнення пісень тегами»

Таблиця 4.6

Прецедент активності «Заповнення пісень тегами»

Характеристика	Опис
Короткий опис	Додавання тегу до пісні
Актори	Активні суб'єкти системи.
Передумови	Додаток запущено
Основний потік подій	Відкриття меню пісні, додавання тегу до пісні
Альтернативний потік подій	
Послідує	Якщо пісня вже була прослухана, то рекомендаційна система може відобразити спільні за тегом даної пісні інші.



Рис. 4.9. діаграма активності «Переглянути теги» прецеденту «Перегляд тегів»

Таблиця 4.7

Прецедент активності «Перегляд тегів»

Характеристика	Опис
Короткий опис	Перегляд наявних тегів
Актори	Активні суб'єкти системи.
Передумови	Додаток запущено
Основний потік подій	Відкриття бокового меню, відкриття списку тегів
Альтернативний потік подій	
Послідуюче	Можливість переглянути список пісень за будь-яким тегом.

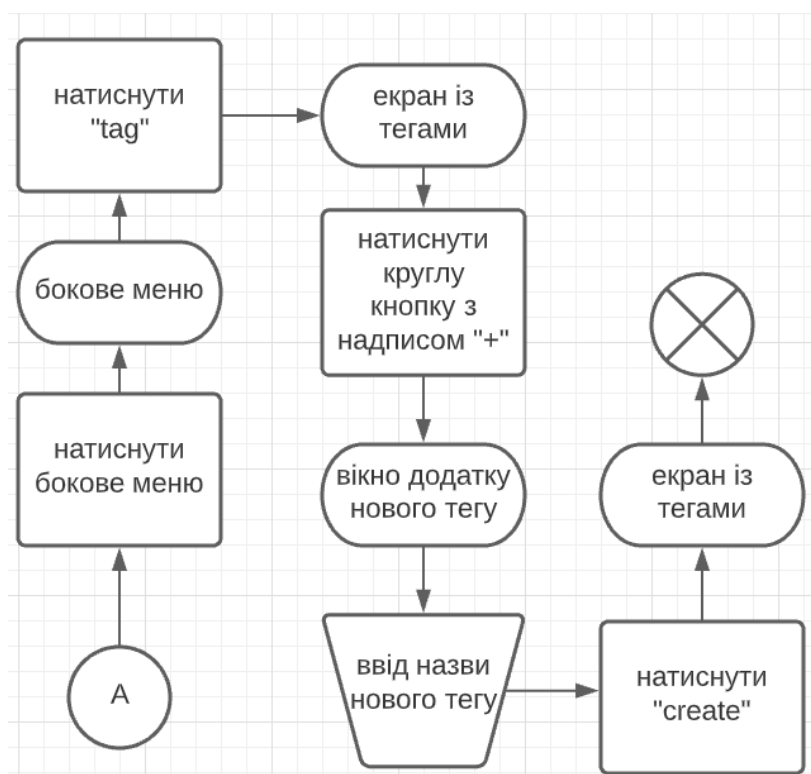


Рис. 4.10. діаграма активності «Створити новий тег» прецеденту «Створення тегів»

Таблиця 4.8

Прецедент «Створення тегів»

Характеристика	Опис
Короткий опис	Створення нового тегу.
Актори	Активні суб'єкти системи.
Передумови	Додаток запущено.
Основний потік подій	Відкриття бокового меню, відкриття списку тегів, створення нового тегу
Альтернативний потік подій	Тег не додано до бази, даний тег вже існує
Послідуюче	Тег можна додати до будь-якої пісні

4.1.2 Логічна схема даних

Для проекту потрібно визначитись із структурою бази даних, з якою ПЗ буде працювати спільно.

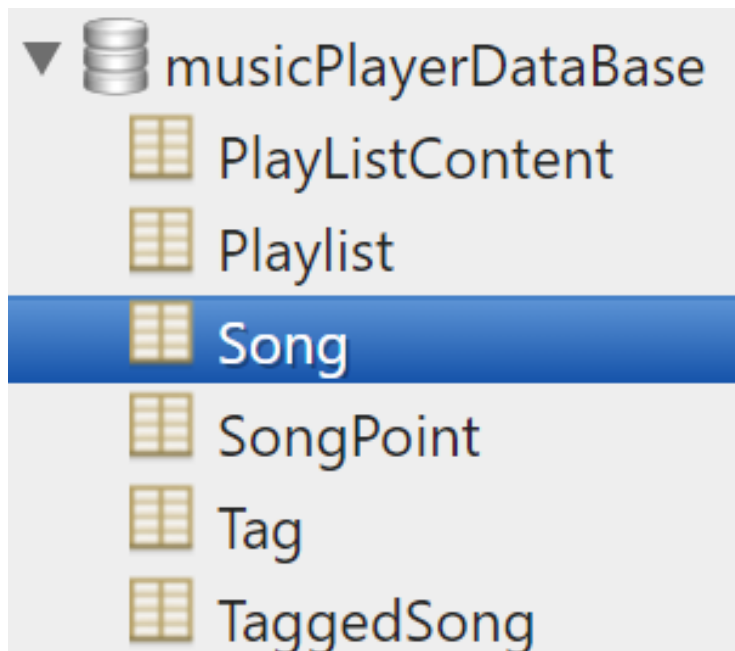


Рис 4.11. Логічна схема бази даних

Для кожної використаної таблиці з бази даних слід описати атрибути, які були створені для роботи проекту. Таблиця «PlayListContent» містить сутності, які відповідають за контент плейлісту, тобто додаючи пісню до плейлісту, заповнюється даними ця таблиця. Її характеристика описана в табл. 4.9.

Таблиця 4.9

Сутність «PlayListContent» (контент плейлісту)

Атрибут	Опис	Тип даних	Значення (приклад)
id	Автоматично згенерований унікальний ідентифікатор	Ціле число	1
songId	Ідентифікатор пісні	Ціле число	4343
playListName	Назва плейлісту	Текст	Star wars

Опис сутності «Playlist» наявний в табл. 4.10, ця сутність відповідає контенту «Playlist», кожне значення назви якого є унікальним.

Таблиця 4.10

Сутність «Playlist»

Атрибут	Опис	Тип даних	Значення (приклад)
name	унікальний ідентифікатор Назва плейлісту	Текст	Star wars

Сутність «Song» (Табл. 4.11) містить багато атрибутів, оскільки існує можливість удосконалення створеної ПЗ, яка може застосовувати дані з атрибутів.

Таблиця 4.11

Сутність «Song»

Атрибут	Опис	Тип даних	Значення (приклад)
id	унікальний ідентифікатор	Ціле число	4876
title	Назва пісні	Текст	Hands up
artist	Ім'я автора	Текст	Ed Sheeran
album	Назва альбому	Текст	The Very Best of...
albumId	Id альбому	Ціле число	-840129354
filePath	Шлях до файлу	Посилання	/storage/emulated/0/1Video Maker/blank.aac
imgUri	Шлях до картинки альбому	Посилання	content://media/external/audi o/media/19141

Сутність «SongPoint» (Табл. 4.12) відповідає за запис активності користувача, ці дані потрібні для роботи рекомендаційної системи, оскільки за цими даними відбудеться робота із тегами.

Таблиця 4.12

Сутність «SongPoint» (прослухана пісня)

Атрибут	Опис	Тип даних	Значення (приклад)
id	Автоматично згенерований унікальний ідентифікатор	Ціле число	1
songId	ID пісні	Ціле число	17340

Сутність «Tag» (Табл. 4.13) використовується для тегування пісень, за цими тегами йде обробка пісень. Всі сутності повинні бути унікальними для уникнення однакових тегів за назвою.

Таблиця 4.13

Сутність «Tag»

Атрибут	Опис	Тип даних	Значення (приклад)
title	Унікальний ідентифікатор, Назва тегу	Текст	Star wars

Сутність «TaggedSong» відповідає за інформацію щодо тегу, який додали до пісні.

Таблиця 4.14

Сутність «TaggedSong» (тег пісні)

Атрибут	Опис	Тип даних	Значення (приклад)
id	Автоматично згенерований унікальний ідентифікатор	Ціле число	1
songId	Назва команди	Рядок	Cool brothers
Tag_title	Назва тегу	Текст	Star wars

Кафедра інтелектуальних інформаційних систем
Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку
Логічна схема даних була створена в телефоні для зберігання, редагування чи видалення даних.

Для покращення розуміння як буде працювати система рекомендацій, були розроблені декілька ескізів, які я використовував під час розробки (додаток В).

4.2 Програмна реалізація

Проаналізувавши існуючі технології та розглянувши інструменти під час роботи, було обрано створення андроїд-додатку для покращення навичок розробки андроїд додатків. Також, для комфортної розробки обрано середовище розробки Android Studio, яка була розроблена в Google саме для таких випадків, оскільки Андроїд є досить поширеною платформою, що її використовують навіть в самрт-годинниках та телевізорах. Для проектування додатку обрані спеціальні інструменти, що покращують можливості розробки, а саме:

- RxJava для мультипоточності (підвищує швидкість роботи додатку),
- DataBinding для покращення роботи із UI елементами;
- Room для роботи із базою даних мобільних додатків;
- Dagger 2.0 для інжектування залежностей (покращує чистоту коду),
- Timber логування історій;
- TedPermission для роботи із забезпеченням доступів від системи андроїд для додатку;
- Stetho для слідкування за змінами в базі даних, Gson – для роботи із json даними;
- MediaFacer для роботою із медіа даними телефону, Glide – для роботи з картинками в телефоні.

Створено певний план, за яким послідовно слідували етапи розробки (програмування) проекту (рис. 1.4). Хоча і дедлайни деяких етапів зривались

Кафедра інтелектуальних інформаційних систем
 Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку
 через технічні та фізичні проблеми, все таки вдалося відновитись та розробити
 проект своєчасно.

Для роботи проекту задіяно 15 xml та 4 jpg файлів сторінок (рис. 3.1), 127
 kotlin файлів (93 kotlin класів, 31 інтерфейс 6 об'єктів).

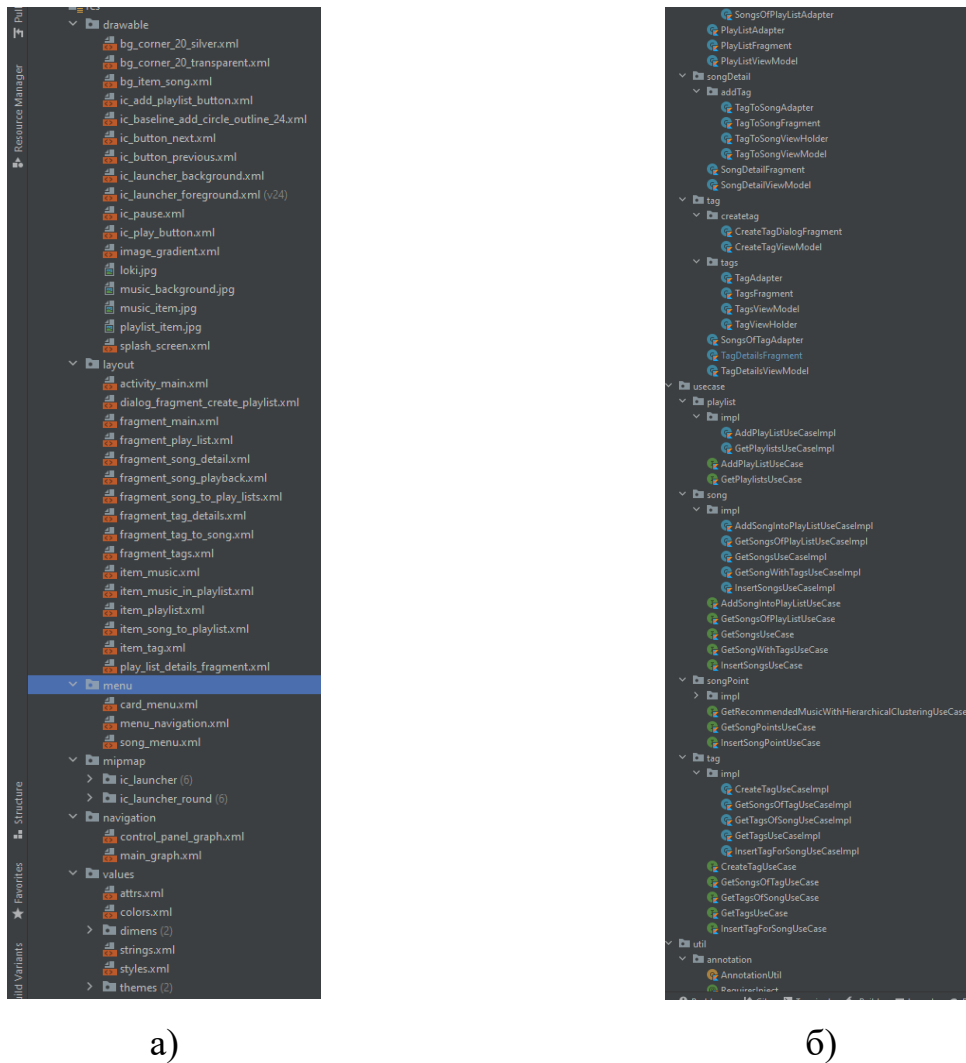


Рис. 4.1. Задіяні у проекті: xml, jpg (а) та kotlin файли (б)

Інтерфейс є досить простим, на Рис. 4.2 зображено головне меню андроїд додатку.

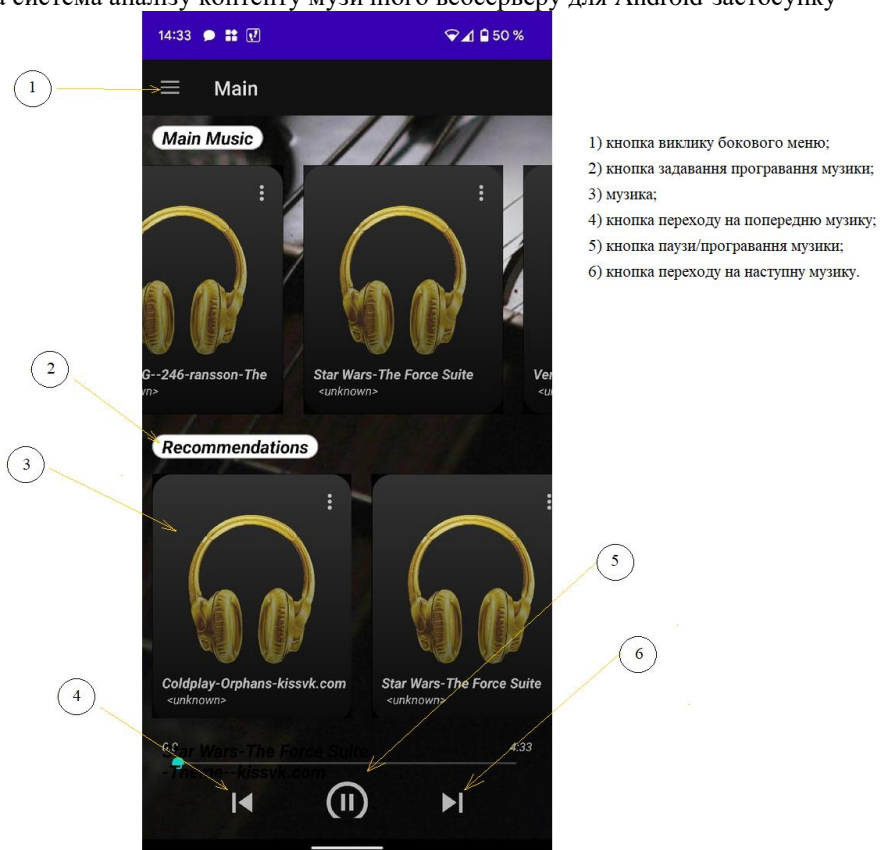


Рис. 4.2. Головне меню

Натиснувши на бокоменю меню (елемент 1) відкриється бокове меню, за допомогою якого можна перейти до екрану тегів та плейлісту (Рис. 4.3).

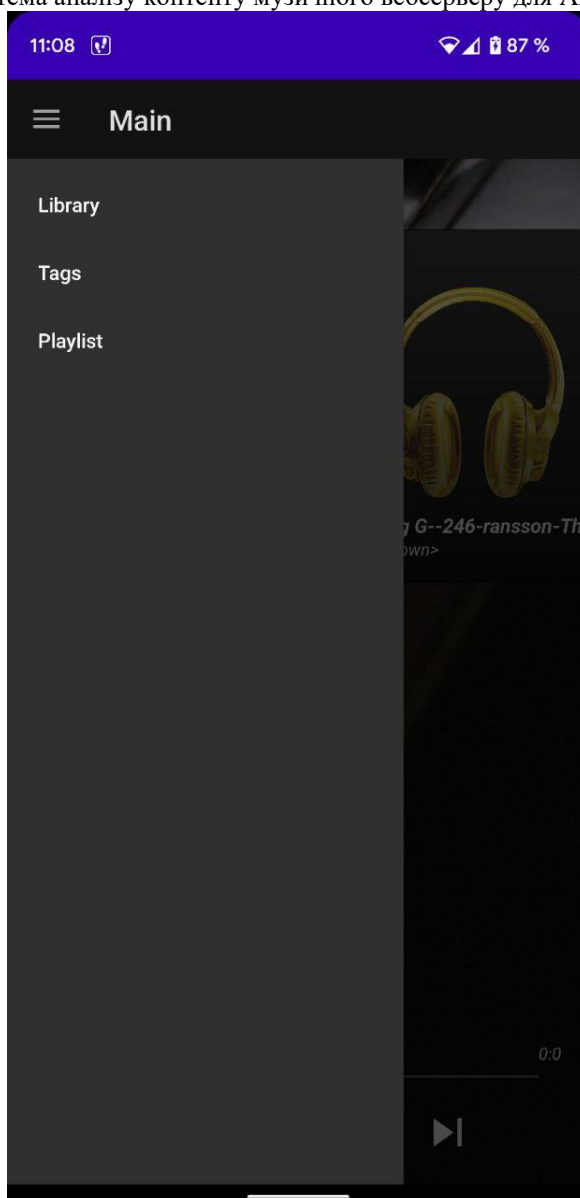
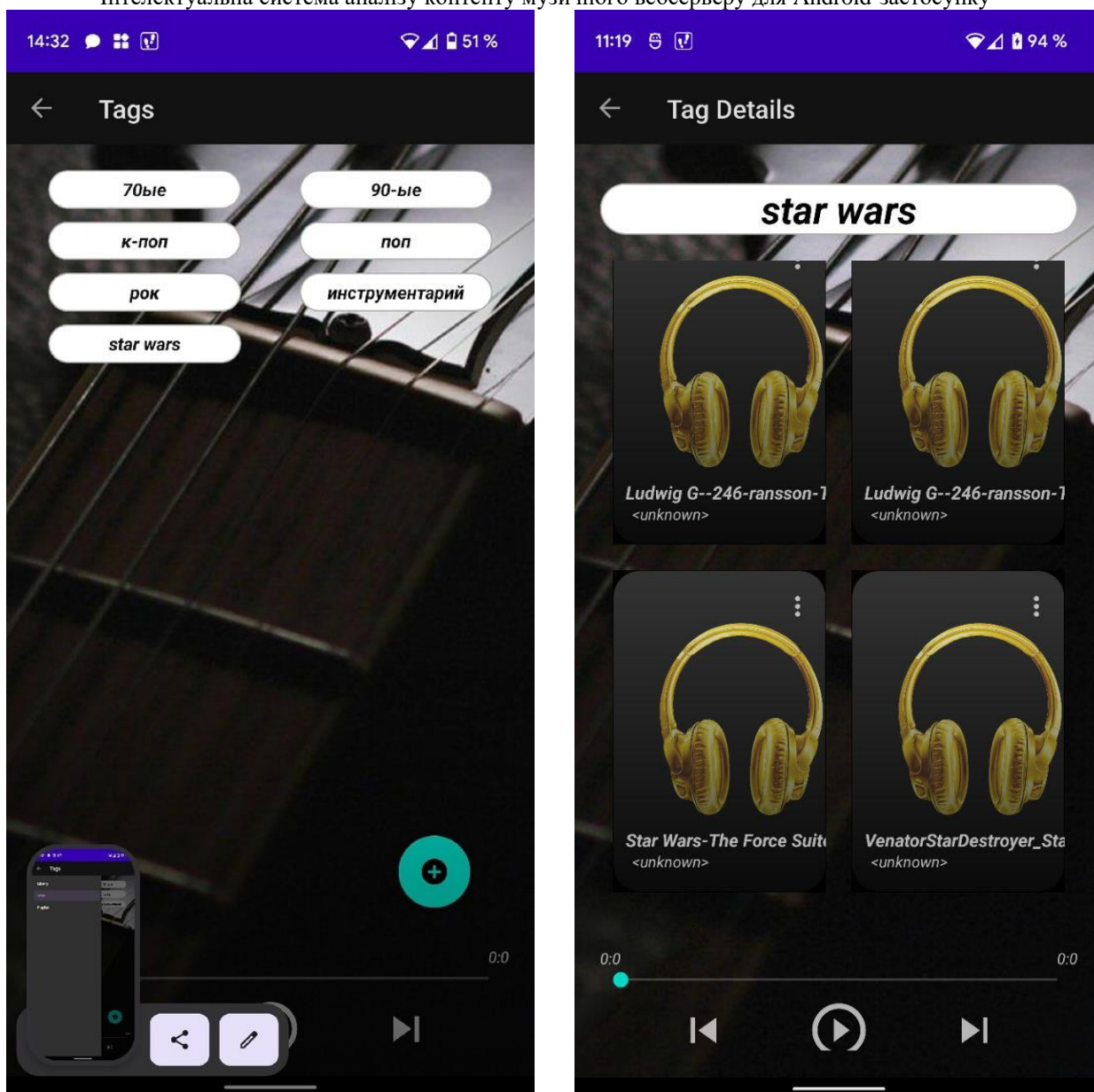


Рис. 4.3. Бокове меню додатку

Перейшовши до екрану тегів можна побачити усі наявні теги (Рис. 4.4а), натиснувши на будь-який з них відкриється екран із інформацією, яка вмістить список музикальних композицій, що мають обраний тег (Рис. 4.4б).



а)

б)

Рис. 4.4. Інтерфейс додатку а) екран тегів; б) пісні з однаковим тегом

Відкривши екран плейлістів можна побачити список створених користувачем плейлістів (Рис. 4.5а). натиснувши на будь-який, можна побачити список пісень, яких було додано до плейлісту (Рис. 4.5.б).

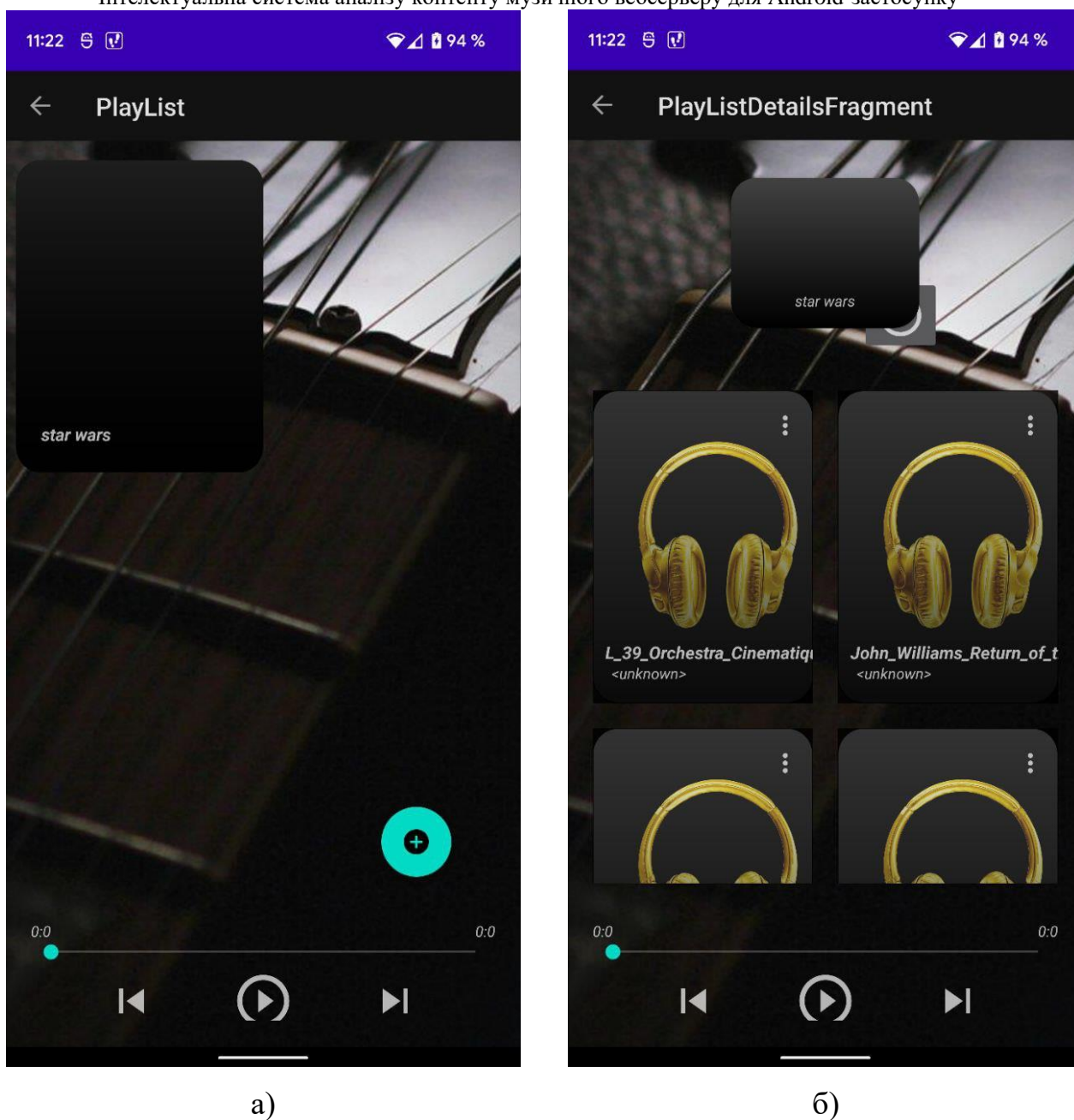


Рис. 4.5. Інтерфейс додатку а) Список плейлістів додатку; б) деталі плейлісту

Додаючи тег до пісні, в базу додається сутність TaggedSong, потім при вибірці даних дані з девайсу по пісні об'єднуються з даними з бази по тегам, всі теги, що були додані в базі до певної пісні доповнюють інформацію щодо деталей цієї пісні (Рис. 4.6).

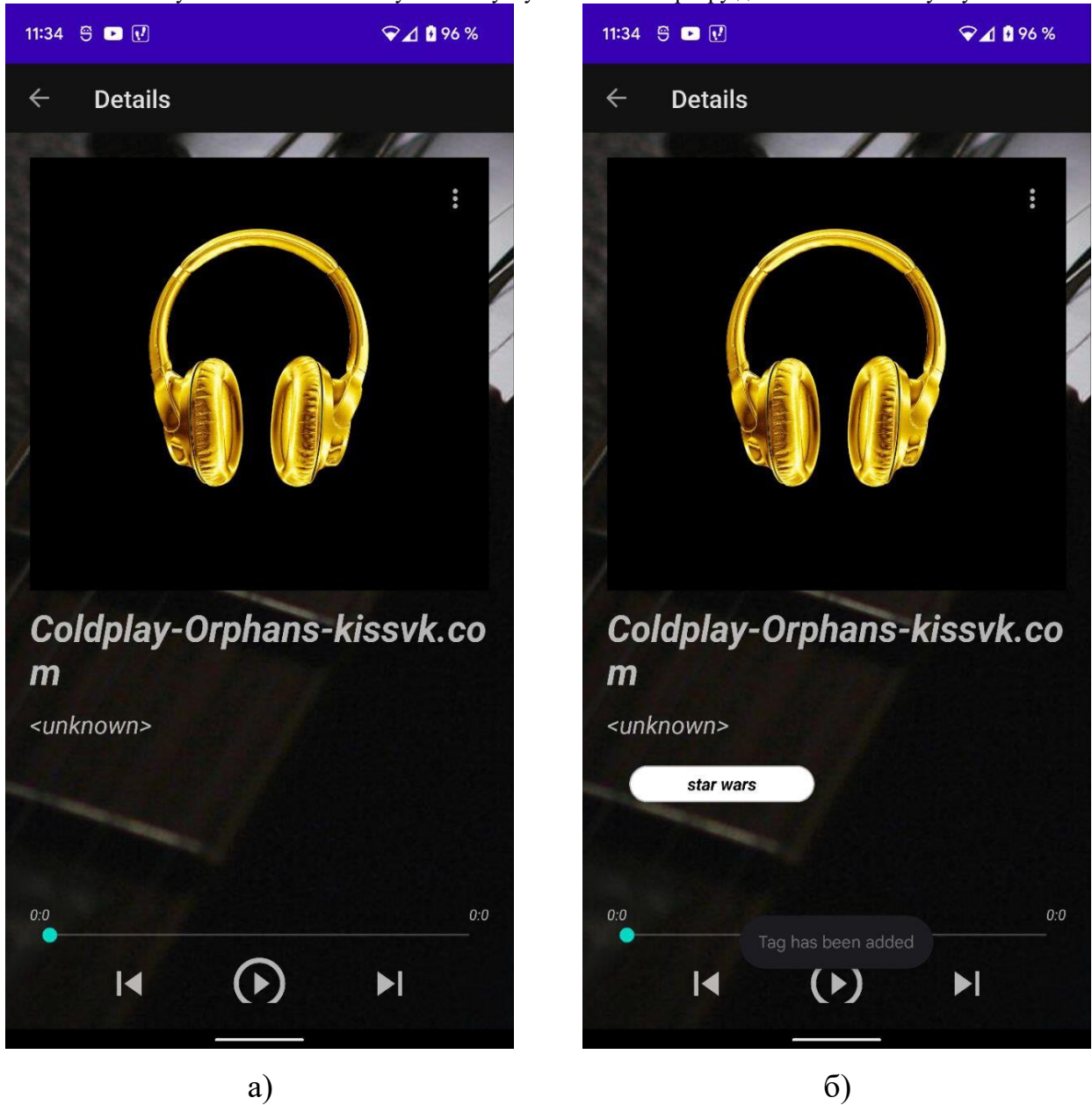


Рис. 4.8. Інтерфейс додатку а) деталі пісні без тегів; б) деталі пісні з тегом

Розроблена мною рекомендаційна система використовує пошарову кластеризацію. Вона працює за таким алгоритмом:

1. За допомогою даних з таблиці TaggedSong БД фільтрується наявний список музики (*id* музики має бути в списку даних з таблиці TaggedSong);
2. Списки тегів, що наявні в отриманому списку пісень сортуються за чисельністю тегів у списку для полегшення наступного кроку, від більшого значення до меншого;

3. Списки тегів, що мають однакову кількість елементів, фільтруються та залишається той список, що більш активно зустрічається у списку прослуханих пісень. Формується новий список, відфільтрований від менш активних;
4. Відбирається елемент списку списку тегів (`List<List<Tag>>`) та за ним (`List<Tag>`) фільтрується список наявних пісень та додається до спільного списку рекомендацій;
5. Обирається наступний елемент списку тегів та проходить крок 4 повторно. Вже додані пісні не додаються повторно до списку, оскільки він не повинен містити дублікатів;
6. Після проходження кожного елементу зі списку списку тегів (`List<List<Tag>>`) остаточний набір композицій відображається на дисплеї як рекомендації.

Лістинг коду міститься у Додатку А.

Для перевірки роботоспроможності системи необхідно мати хоча б малу кількість пісень, які мають теги. Розробка із серверною частиною потребує фінансування, тому для вирішення даної задачі було обрано рішення створити демо та використовувати локальні дані пристрою, тобто, пісні що наявні в пам'яті телефону. Демо версії з матеріалом на девайсі вистачить для аналізу результатів та визначення ефективності.

Для роботи системи на девайсі музику в додатку потрібно тегувати. Щоб забезпечити роботу було створено декілька тегів, що наявні на Рис. 4.9. Ці теги потім застосовувались та додавались до пісні.

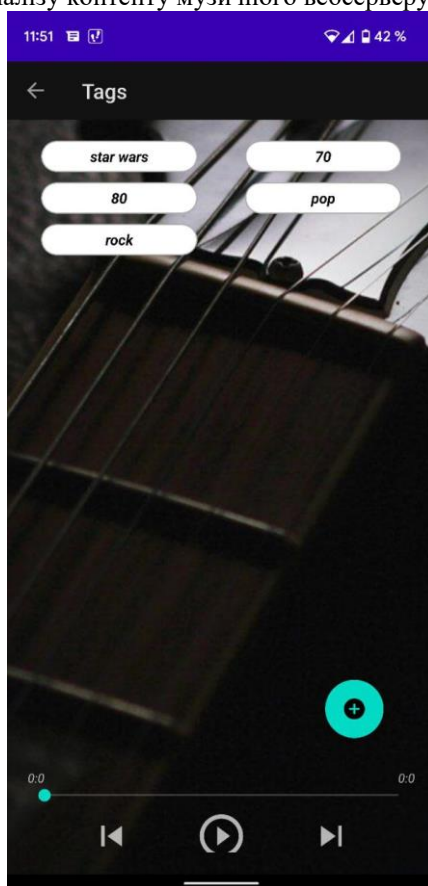


Рис. 4.9. Створені теги для тестування додатку

Список пісень, що мають теги описано в Табл. 3.15

Табл. 3.15

Таблиця тегів пісень

<i>Тег</i>	<i>Пісня</i>
Star wars	The baby: Ludwig G Ransson
	The Mandalorian: Ludwig G Ransson
	The Force Suite Theme: Star wars
	Venator Star Destroyer
	Return of the Jedi: John Williams
70	The Mandalorian: Ludwig G Ranssong
	Orphans: Coldplay
	Hands up: Merk & Kremont feat. DNCE
80	The Force Suite: Star wars
	Survivor: 2WEI Edda Hayes

<i>Тег</i>	<i>Пісня</i>
80	The baby: Ludwig G Rannsong
Pop	Hands up: Merk & Kremont feat. DNCE
	I See Fire: Ed Sheeran
	Coastline: Hollow Coves
	Moskau: Dschighis Khan
	Survivor: 2WEI Edda Hayes
Rock	Orphans: Coldplay

З таблиці видно, що деякі пісні зустрічаються декілька разів, це зроблено навмисно, оскільки мій алгоритм перевіряє пісні не лише за всіма тегами, що наявні в пісні а й по окремому тегу також. Це означає, що, наприклад, користувач слухає пісню «Orphans: Coldpaly», яка має 2 теги: «70» та «Rock», тоді йому буде пропонуватись музика не лише з 2 відповідними тегами, а ще й музика, яка має у списку свої тегів «70» чи «Rock». Список прослуханого матеріалу можна побачити на Рис. 4.10.

id	songId
1	4876
2	4878
3	23574
4	4876
5	4876
6	4878
7	4880
8	4878
9	4880
10	4878
11	23574
12	4876
13	4878
14	4880
15	5150
16	4880
17	4878
18	23574
19	4876
20	4878
21	4880
22	4878
23	23574
24	4876
25	4878
26	4880
27	4876
28	4878
29	23574
30	4876
31	4878
32	23574
33	23548
34	23574

id	id	title
4876	4876	Coldplay-Orphans-kissvk...
4878	4878	Ludwig G--246-ransson-...
4880	4880	Coldplay-Fix You-kissvk.c...
5150	5150	Ludwig G--246-ransson-...
5151	5151	Star Wars-The Force Suit...
23548	23548	HOIST THE COLOURS
23574	23574	Bandoleros (РИНГТОН)

а)

б)

Рис. 3.10. Дані з базу даних а) таблиці прослуханого матеріалу; б) таблиця пісень, які були прослухані

За активністю видно, що активно прослуховувались пісні, що мають теги:

- Orphans: Coldplay – «70», «Rock»;
- The baby: Ludwig G Ransson – «70», «star wars»;
- The Mandalorian: Ludwig G Ranssong – «star wars», «70»;
- The Force Suite: Star wars – «80», «star wars»;
- Інші – тегів не мають.

Це означає, що при врахуванні активності цих пісень (на тег більше ніж 3 пісні записано активності, показник потребує коригуванню) система буде рекомендувати спочатку пісні із більшою кількістю співпадінь, потім із

Кафедра інтелектуальних інформаційних систем
Інтелектуальна система аналізу контенту музичного вебсерверу для Android-застосунку меншою, так до одного тегу. Результати тестування зображено на Рис. 3.11. та Табл. 3.16.

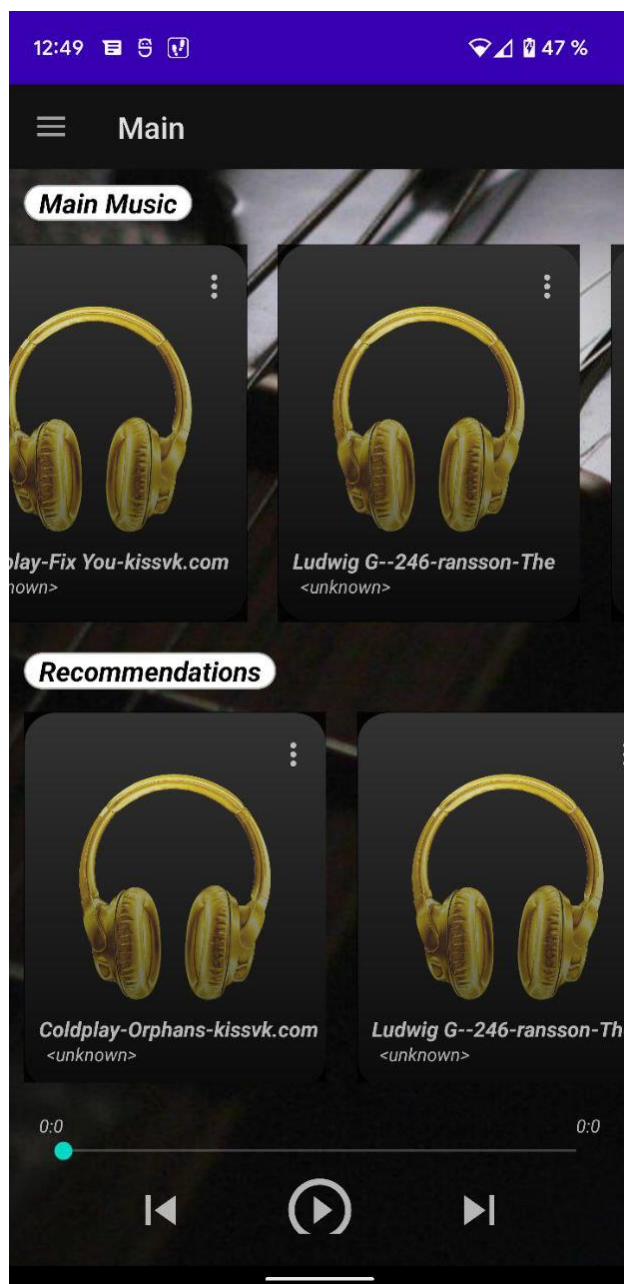


Рис. 3.11. Рекомендації на основі прослуханого матеріалу.

Рекомендації на основі прослуханого матеріалу

Пісня	Теги
Orphans: Coldplay	«Rock», «70»
The baby: Ludwig G Rannson	«Star wars», «80»
The Force Suite Theme: Star wars	«Star wars», «80»
The Mandalorian: Ludwig G Rannson	«Star wars», «70»
Venator Star Destroyer	«Star wars»
Return of the Jedi: John Williams	«Star wars»
Hands up: Merk & Kremont feat. DNCE	«pop», «70»
Survivor: 2WEI Edda Hayes	«80», «pop»

За результатами видно, що алгоритм працював правильно, оскільки: пісні шукались за тегами «Rock», «70», потім – за «Star wars», «80» - тегами, далі – окремо за тегами: «Star wars», «70» та «80». Пісні «Hands up: Merk & Kremont feat. DNCE» та «Survivor: 2WEI Edda Hayes» додані останніми, оскільки в них за алгоритмом було знайдено лише один спільний тег: «70» та «80» відповідно. Подальше опрацювання та модифікація алгоритму рекомендується проводити після глобального тестування із великою кількістю тестувальної групи. Код алгоритму представлено в Додатку Б.

Висновки до розділу 3

Даний розділ містить інформацію щодо моделювання прецедентів проекту із рекомендаційною системою, розробку дизайну та перевіркою результатів роботи системи. Під час його опису були розглянуті прецеденти, які були визначені завдяки процесній моделі системи. За кожним прецедентом визначено актори, зв'язки між ними та діаграми активностей. За кожною діаграмою розроблено детальний аналіз у вигляді таблиць, в яких зазначено детальний опис короткий опис, актори, основні події, передумови, наслідки, послідує.

За логічною схемою даних, створено та описано кожну таблицю бази даних, яка була задіяна в системі. Кожна sql-таблиця описана за її характеристиками: атрибути таблиці, їх опис, тип даних та приклади.

Під час розробки проекту були створені шаблони, на базі яких розроблялось ПЗ (Додаток Б).

Для роботи алгоритму, який оцінює музичний контент за тегами, розроблено певний алгоритм, за яким аналізується прослуховуваний контент та знаходяться більш активні кластери (теги), які приймають участь у створенні рекомендацій.

Розроблена система була перевірена, результати роботи вважаються успішними. Новостворену систему можна розглядати для комерційного застосування.

ВИСНОВКИ

Під час аналізу сучасного стану обраної теми було проаналізовано сферу застосування рекомендаційних систем музичних вебсерверів для андроїд-додатків, їх вигоду у використанні та у створенні нової системи для покращення ефективності рекомендацій.

Для створення нової інтелектуальної системи було проаналізовано останні публікації та дослідження у галузі «кластеризація даних» за темою «аналіз музичних композицій». За даними з цих публікацій відібрано кластеризацію даних як засіб для подальшого аналізу та застосування в системі.

Якщо створена система зможе поліпшити продуктивність аналізу музичних композицій не вводячи додаткових проблем, тоді слід розробити дану програму.

Були розглянуті математичні моделі алгоритмів кластеризації. Більш детально розглянуто центроїдну та графову моделі, їх методи застосування. Визначено, що метод k-середніх не підходить для аналізу музичних композицій за певними тегами, оскільки існує складність їх чисельного класифікування для задіяння в алгоритмі фільтрування музики за цим методом, оскільки він підходить для детального аналізу саме музичних технічних характеристик. Для аналізу застосування даного алгоритму залучено дані музичного контенту із чисельними характеристиками із допомогою API Spotify.

Для інтелектуальної системи рекомендацій, кластери яких не мають якихось чисельних характеристик було обрано метод графової моделі, а саме пошарову кластеризацію. Цей метод підходить для кластеризації будь яких речей, які не мають якихось чисельних характеристик та не має складності в аналізі як при застосуванні методу кластеризації k-середніх.

Були розглянуті прецеденти, які були визначені завдяки процесній моделі системи. За кожним прецедентом визначено актори, зв'язки між ними та діаграми активностей. За кожною діаграмою розроблено детальний аналіз у вигляді таблиць, в яких зазначено детальний опис, короткий опис, актори, основні події, передумови, наслідки, послідує.

Був створений певний план, за яким послідовно слідували етапи розробки (програмування) проекту. Під час розробки проекту були створені шаблони, на базі яких розроблялась система з управління проектами.

Під час розробки проекту були створені шаблони, на базі яких розроблялось ПЗ.

Для роботи алгоритму, який оцінює музичний контент за тегами, розроблено певний алгоритм, за яким аналізується прослуховуваний контент та знаходяться більш активні кластери (теги), які приймають участь у створенні рекомендацій.

Під час роботи над дипломною роботою не було виявлено жодних порушень з питань охорони праці. Робоче місце було оснащено належним чином. Технічний стан обладнання відповідав стандартам безпеки і нормам охорони праці, ніяких дефектів обладнання під час виконання роботи не виявлено.

В результаті написання спеціальної частини з охорони праці було досягнуто поставленої мети, а саме створення безпечних і здорових умов праці на робочих місцях, в робочих зонах, у виробничих приміщеннях.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Кластеризация K-средних и PCA для классификации музыки по схожим звуковым характеристикам. URL: <https://ichi.pro/ru/klasterizacia-k-srednih-i-pca-dla-klassifikacii-muzyki-po-shozim-zvukovym-harakteristikam-245233124019044> (дата звернення: 14.12.2021)
2. Костянтин Пічущкін. Гибридная система классификации музыкальных треков. URL: https://dspace.spbu.ru/bitstream/11701/10624/1/Kostya_Pichushkin_-_diploma.pdf (дата звернення: 14.12.2021)

ДОДАТОК А

Лістинг коду алгоритму

```
object Algorithm {
object HierClustering {

fun getRecommendedMusic(songPoints: List<SongPoint>, songList: List<Song>): List<Song> {
    val tags = getPopularTags(getSortedTags(getListenedSongs(songPoints, songList)))
    return getFilteredSongs(tags, songList)
}

private fun getListenedSongs(songPoints: List<SongPoint>, songList: List<Song>): List<Song>
{
    return songPoints
        .flatMap { point -> songList.filter { point.songId == it.id } }
}

private fun getSortedTags(listenedSongs: List<Song>): Map<List<SongTag>, Int> {
    val tagsOfSongs = listenedSongs
        .map { it.tags }
        .filter { it.isNotEmpty() }
        .sortedBy { it.size }
        .toSet()

    return tagsOfSongs
        .map { tags -> Pair(tags, listenedSongs.count { it.tags.containsAll(tags) }) }
        .toMap()
        .toSortedMap { o1, o2 ->
            when (o1.size == o2.size) {
                true -> 0
                else -> if (o1.size > o2.size) 1 else -1
            }
        }
}
}
```



```
private fun getPopularTags(tags: Map<List<SongTag>, Int>): List<List<String>> {  
    return tags.flatMap { (songTags, _) ->  
        val tagsWithSameCount = tags.filterKeys { songTags.size == it.size }  
        val maxValue = tagsWithSameCount.values.maxOrNull { it }  
        tagsWithSameCount  
            .filter { (_, value) -> maxValue == value && maxValue > 3 }  
            .keys  
    }  
    .toList()  
    .map { it.map { it.tagTitle }.toList() }.toList()  
}
```

```
private fun getFilteredSongs(tags: List<List<String>>, songList: List<Song>): List<Song> {  
    return songList.filter { song -> tags.contains(song.tags.map { it.tagTitle }) }  
}  
}  
}
```

ДОДАТОК Б

Ескізи проекту

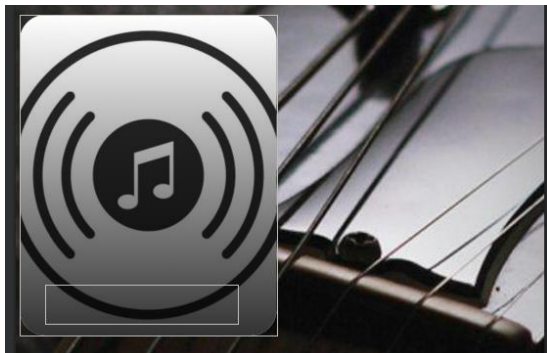


Рис. В.1. Ескіз плейліста

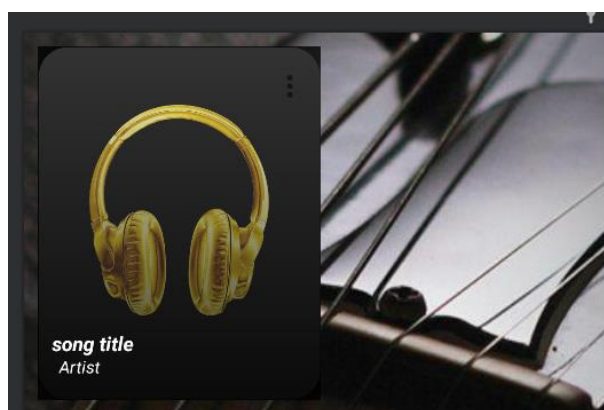


Рис. В.2. Ескіз демонстрації музичної композиції

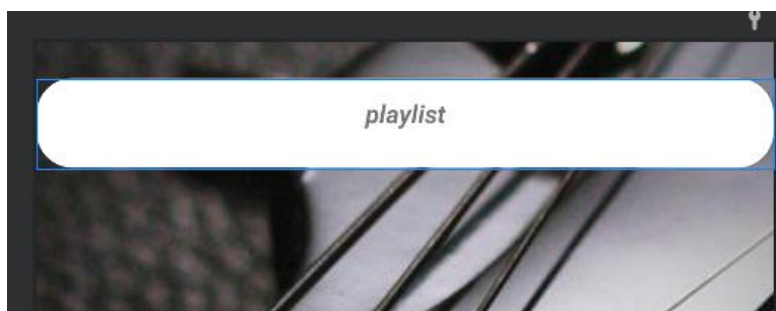


Рис. В.3. Ескіз вибору плейліста для додавання пісні

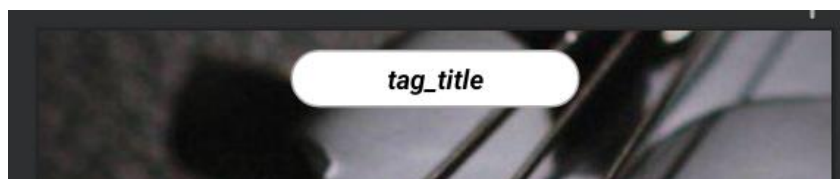


Рис. В.4. Ескіз тегу



Рис. В.5. Ескіз вікна «Додати новий плейліст»

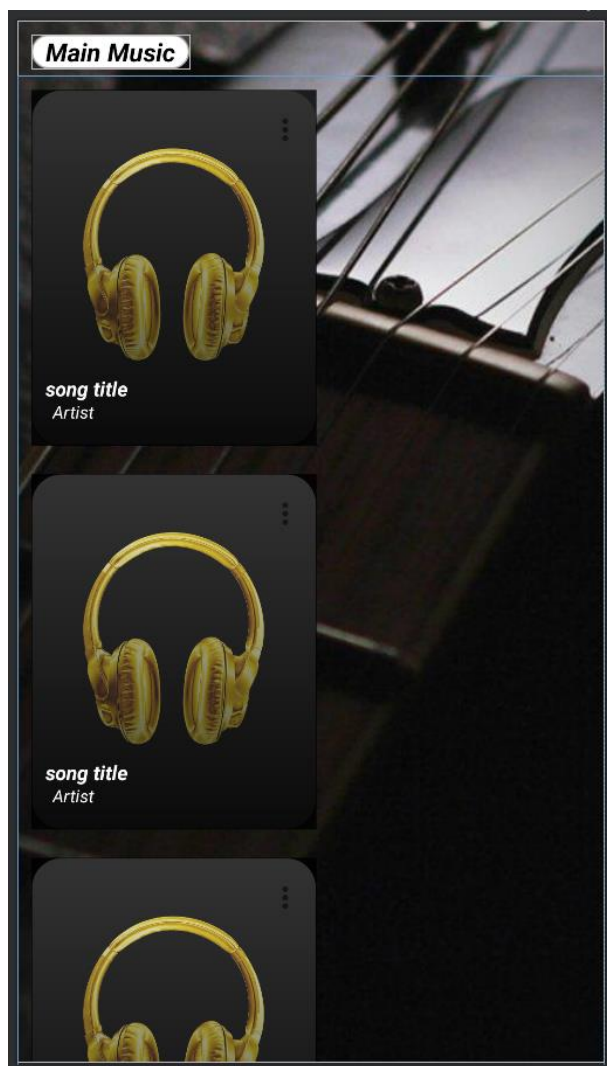


Рис. В.6. Ескіз головної сторінки додатку