

# Автоматическая расстановка рейтинга музыкальным произведениям на основе неявных оценок

Сергей Смагин

Тульский государственный университет, Тула, Россия  
smaginsergey1310@gmail.com

**Аннотация** В настоящее время людям приходится взаимодействовать с огромным количеством различных данных. Помочь ему в этом призваны рекомендательные системы. В частности, у слушателя музыки на электронных устройствах обычно достаточно большой список воспроизведения, и не каждая композиция в таком списке ему может одинаково нравиться. В данной статье описан подход к вычислению рейтинга музыкальных композиций на основе неявных оценок.

**Ключевые слова:** рекомендательные системы, рейтинг музыки.

## 1 Введение

Программными музыкальными проигрывателями пользуется большое число человек. У многих слушателей большой список воспроизведения, который часто состоит из песен, которые пользователь не хотел бы слушать по той или иной причине (например, они надоели или их скачали вместе с другими песнями того же исполнителя и она оказалась неудачной). Одним из решений такой проблемы является методичное переслушивание всего списка воспроизведения и удаление нежелательных к прослушиванию треков. Недостатки этого подхода очевидны:

- Не каждый захочет уделить несколько часов (а в худшем случае — дней) на приведение коллекции в порядок.
- Будут появляться новые песни, старые будут надоедать и через некоторое время список воспроизведения снова будет наполнен композициями, которые не хочется слушать.

Другим решением этой проблемы может быть рекомендательная система. Пользователь сам оценит некоторые треки, чтобы система проигрывала их чаще других.

Как известно, рекомендательные системы бывают построены на явных или неявных оценках<sup>1</sup>. Пример рекомендательной системы с явными оценками — применяемая во многих проигрывателях схема со “звездами” вместо

<sup>1</sup> [https://ru.wikipedia.org/wiki/Рекомендательная\\_система/Методика](https://ru.wikipedia.org/wiki/Рекомендательная_система/Методика)

оценок. Но для большинства людей такая система также неэффективна — расставлять оценки каждый раз нужно вручную, а это ненамного лучше ручного удаления композиций.

Рекомендательная система на основе неявных оценок в этом случае выглядит гораздо предпочтительнее, поскольку прослушивание музыки в большинстве случаев — фоновая задача. Неявными оценками в данном случае будут действия пользователя музыкального проигрывателя.

## 2 Существующие решения

В Amarok<sup>2</sup> и Clementine<sup>3</sup> используется следующее решение. Добавляемая в коллекцию песня имеет средний рейтинг и далее, в зависимости от доли прослушанной композиции, рейтинг увеличивается или уменьшается.

Многие программы-проигрыватели базовой поставке или в виде расширений имеют систему оценки композиций с применением «звезд».

В Интернете существует сразу несколько известных сервисов для оценки музыки (самый известный, должно быть, Last.fm<sup>4</sup>), для оффлайновых проигрывателей нет широко известной рекомендательной системы.

## 3 Предлагаемое решение

Для решения задачи был придуман алгоритм, основанный на следующих поведенческих признаках:

- Много раз прослушанная песня нравится меньше услышанной впервые.
- Песни, которые не нравятся, переключают.

Тогда, во-первых, рейтинг должен уменьшаться с количеством прослушиваний. В разработанном алгоритме применена следующая формула:

$$r = \frac{1}{\lg(10 + i \cdot lc)}, \quad (1)$$

где  $i$  — коэффициент влияния возраста, показывает, с какой скоростью песни устаревают (по умолчанию рейтинг уменьшается вполтину после 90 прослушиваний) и  $lc$  — количество прослушиваний.

Во-вторых, рейтинг песни должен быстро (гораздо быстрее, чем при прослушивании) уменьшаться при переключении. Была применена формула:

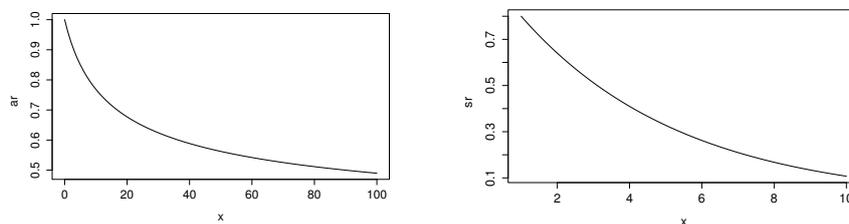
$$nr = or \cdot (1 - si \cdot (1 - sr)), \quad (2)$$

где  $nr$ ,  $or$  — новое и старое значения рейтинга песни,  $si$  — коэффициент влияния переключения песни (по умолчанию рейтинг песни может максимально уменьшиться на 0.2 от прошлого значения),  $sr$  — прослушанная доля песни.

<sup>2</sup> <http://amarok.kde.org/>

<sup>3</sup> <http://www.clementine-player.org/>

<sup>4</sup> <http://www.lastfm.ru/>



(a) Снижение рейтинга песни в зависимости от количества прослушиваний (b) Снижение рейтинга песни в зависимости от количества пропусков

Рис. 1: Графики уменьшения рейтинга песни

Формулы (1) и (2) формируют рейтинги композиции (максимальное значение по каждому критерию — единица). Для получения общего рейтинга нужно перемножить значения критериев. Начальное значение по каждому следует взять максимальным, поскольку новые песни хочется прослушать сразу после добавления. Достоинства такого подхода:

- Конфигурируемость — пользователь может задать соответствующие коэффициенты влияния по своему вкусу.
- Неявные оценки — пользователю не нужно делать ничего специального для того, чтобы система работала.

Данный алгоритм реализован в программе `autorating`<sup>5</sup> как клиент для `mpd`<sup>6</sup> — системной службы для проигрывания музыки. Данная реализация оказалась очень компактной и простой за счет клиент-серверной архитектуры `mpd`.

## 4 Оценка качества рекомендаций

Для оценки качества рекомендаций поставим следующий эксперимент. Возьмем несколько песен разной длины. Пусть одна из них условно нравится, другие — нет. Неправящиеся песни будем пропускать на первых секундах. Будем проигрывать песни 5 кругов. Перед началом проигрывания каждая песня имеет рейтинг 255 (наивысший возможный).

Как видно, песня, условно нравящаяся, по рейтингу далеко опережает те, которые не нравятся, см. таблицу 1. Кроме того, рейтинг последних крайне мало зависит от длины, если треки пропускать на первых секундах (как это обычно делается с ненавнящими композициями).

<sup>5</sup> <https://github.com/s-mage/autorating>

<sup>6</sup> <http://www.musicpd.org/>

Таблица 1: Сравнение рейтинга песен после окончания эксперимента

Название	Длина, с	Нравится?	Рейтинг
mix 25	3540	нет	70
striken	245	да	216
remember the name	220	нет	71
hybris	206	нет	71
i miss you	366	нет	70
а ну отдай мой каменный топор	120	нет	72

## 5 Заключение

Представлен алгоритм автоматического оценивания музыкальных произведений на основе неявных оценок. Данный алгоритм относительно прост и основывается на поведенческих признаках, которые подходят большинству людей. При испытании программы, реализующей данный алгоритм, была подтверждена ее способность к решению поставленной задачи и эффективность.

Направления дальнейшей работы перечислены ниже.

- Больше поведенческих признаков. Например, переключение после паузы может не учитываться (по личному опыту, не всегда хочется слушать даже хорошую песню не с начала).
- Добавление возможности пользователю явно выразить отношение к песне и учитывать это в общем рейтинге.
- Подстройка под настроение. В зависимости от настроения человек может быть расположен слушать разные песни.

## Список литературы

1. Kordumova, Suzana et al. Personalized implicit learning in a music recommender system. User Modeling, Adaptation, and Personalization. Springer Berlin Heidelberg, 2010. 351-362.
2. Kim, Hyun-Jun, and Young Sang Choi. EmoSens: Affective entity scoring, a novel service recommendation framework for mobile platform. Workshop on personalization in mobile application of the 5th international conference on recommender system. 2011.
3. The International Society for Music Information Retrieval, <http://www.ismir.net>.
4. Zaharchuk, Vasily et al. A new recommender system for the interactive radio network fmhost. Proceedings of the international workshop on experimental economics and machine learning (EEML). 2012.

# Automatic Music Rating Based on Implicit Assessments

Sergey Smagin

Tula State University, Tula, Russia  
smaginsergey1310@gmail.com

**Abstract.** Nowadays people have to interact with huge amount of data. Recommender systems are all about help them with it. Particulaly, music listener often has large playlist and not each track he likes the same. This article describes a way to evaluate music rating based on implicit assessments.

**Keywords:** music rating, recommender systems.