



From Web to Map: Exploring the World of Music

Olga Goussevskaja
ETH Zurich, Switzerland
golga@tik.ee.ethz.ch

Michael Kuhn
ETH Zurich, Switzerland
kuhnmi@tik.ee.ethz.ch

Michael Lorenzi
ETH Zurich, Switzerland
mlorenzi@student.ethz.ch

Roger Wattenhofer
ETH Zurich, Switzerland
wattenhofer@tik.ee.ethz.ch



Докладчик: Жук Р.В.

Вступление

- * Большие объемы личных музыкальных коллекций
 - * Сложный поиск песен
 - * Запутанная организация папок
- * Много обновлений
- * Ошибки в заполнении метадаты

От выборки к графу

- * База данных Last.fm (290.000 пользователей)
 - * Пользователь – личный топ-50 песен
 - * У песен есть связь
- * Песни – вершины, связи – ребра с весами
- * Нормирование весов рёбер
 - * $h = \sqrt{n_i n_j} / n_{i,j}$
 - * Отсеиваем $h < 2$
- * Итог
 - * 430.000 вершин
 - * 6,3 млн ребер

Переход к евклидовому пространству

- * Метод многомерного шкалирования с опорными точками (LMDS)
 - * Основное требование: $K = \frac{d_G(i, j)}{d_e(i, j)} = const \quad \forall i, j \in G$
 - * $d_G(i, j)$ - вес ребра графа
 - * $d_e(i, j)$ - евклидово расстояние
- * Upgrade
 - * Применяем алгоритм LDMS
 - * Удаляем часть ребер с максимальным стрессом (K^{-1})
 - * Повторяем k раз

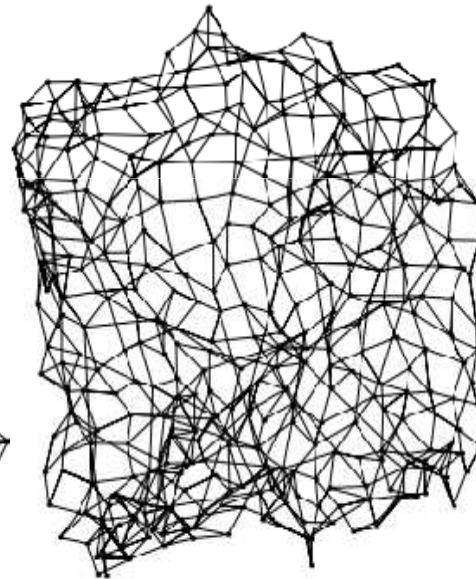
Пример эффективности алгоритма



Исходный граф

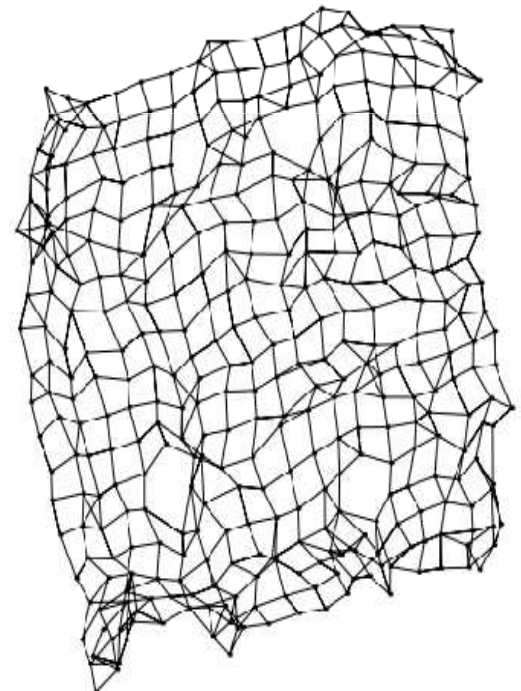


После 6 итераций



После 12 итераций

5



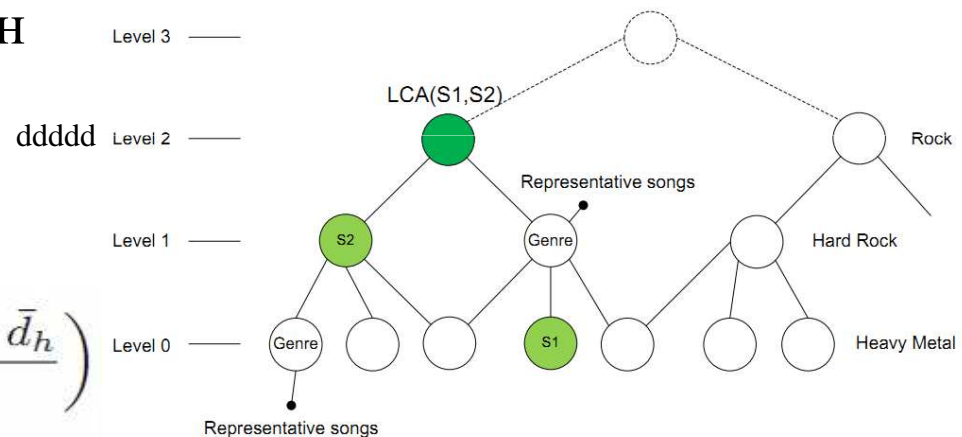
После 30 итераций

Выбор параметров

- * Allmusic.com - 7000 песен с описанием жанров
- * Мера – уровень наим. общего предка.

$$Q_L = \frac{1}{H_S} \cdot \sum_{h \in \{0 \dots H_S - 1\}} \left(\frac{\bar{d}_{h+1} - \bar{d}_h}{\bar{d}_h} \right)$$

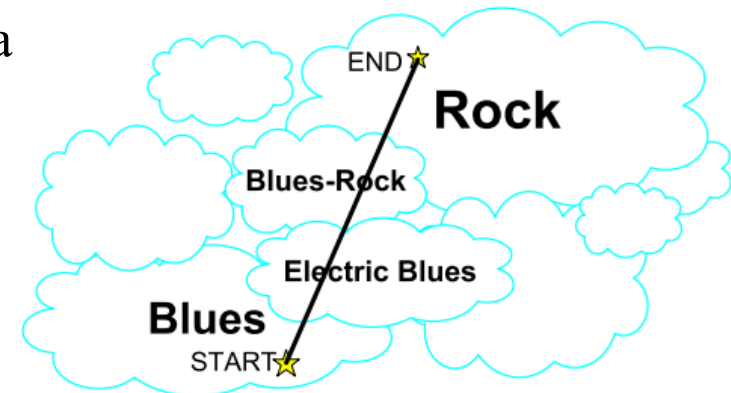
- * d_h - среднее расстояние м/у элементами пространства с жанровым расстоянием $d_s(i, j) = h$
Чем больше Q_l - тем лучше качество.



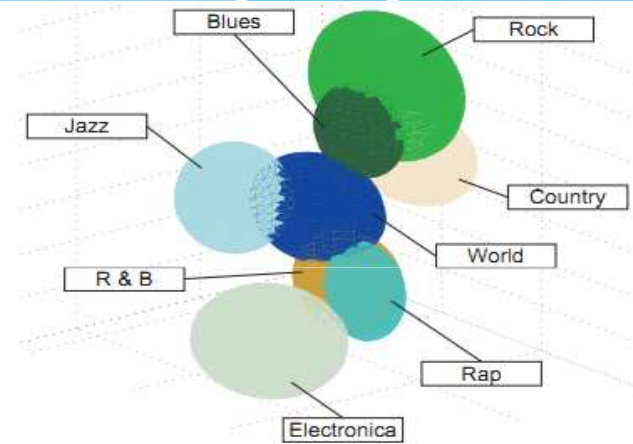
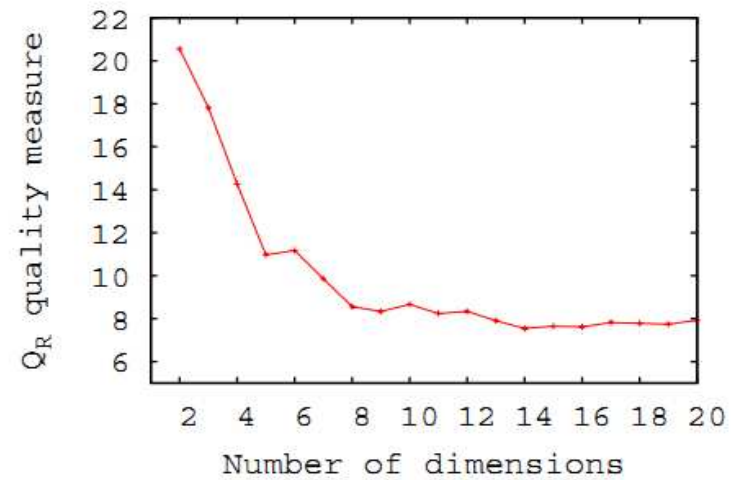
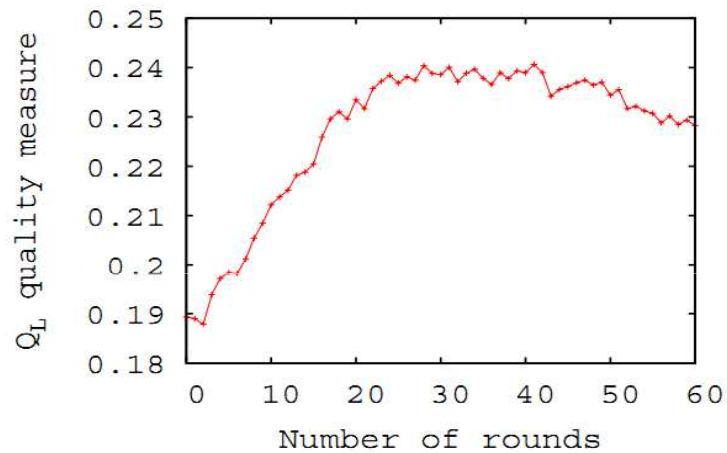
Выбор параметров

- * «Гладкость» переноса графа
 - * Хороший результат – песни одного жанра близко друг к другу
 - * Выделяем группы песен одного жанра
 - * Проводим прямую м/у двумя песнями
 - * Хороший результат - прямая пересекает жанр лишь единожды

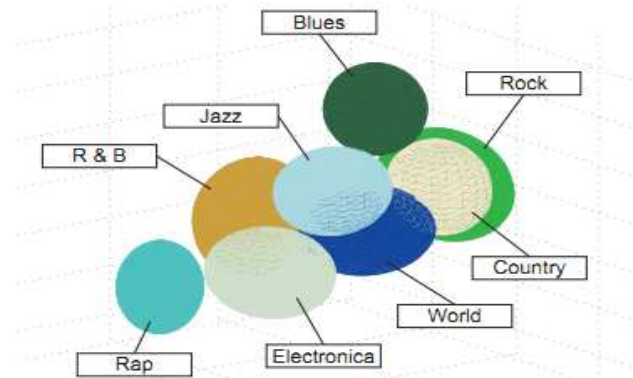
Q_i = среднее число исключений из правила



Выбор параметров



(a) Output of the original LMDS algorithm



(b) After 30 rounds of iterative embedding

Пример

Соседи

Pink Floyd (Time)

Pink Floyd (On the Run)
Pink Floyd (Any Colour...)
Pink Floyd (The Great G...)
Pink Floyd (Eclipse)
Pink Floyd (Us and Them)
Pink Floyd (Brain Damage)
Pink Floyd (Speak to Me)
Pink Floyd (Money)
Pink Floyd (Breathe)
Pink Floyd (One of These...)

Miles Davis (So What)

Horace Silver (Song For My...)
Bill Evans (All of You)
Miles Davis (Freddie Fre...)
Nat King Cole (The More I...)
Miles Davis (So Near)
Miles Davis (Flamenco Sk...)
Charles Mingus (Eat That Ch...)
Jimmy Smith (On the Sunny...)
Julie London (Daddy)
Bill Evans (My Man's Gone...)

Траектории

1. M. Davis (Milestones)
 2. M. Davis (Someday...)
 3. J. Coltrane (Wise One)
 4. R. Johnson (Kind H...)
 5. J. Hendrix (Are You...)
 6. Queen (Good Old...)
 7. J. Hendrix (Gypsy...)
 8. AC/DC (Let Me Put...)
 9. AC/DC (Givin The...)
 10. GnR (You're Crazy)
1. Eminem (Lose Yourself)
 2. Eminem (Stan)
 3. Eminem (Mockingbird)
 4. D. Bedingfield (Gotta Get...)
 5. Nelly (Dilemma ft Kelly...)
 6. Pink (Most Girls)
 7. G. Stefani (What You...)
 8. C. Aguilera (Genie in a...)
 9. B. Spears (Breathe on Me)
 10. B. Spears (Toxic)

Преимущества

- * Возможность
 - * Строить плейлисты по траекториям
 - * Определять направления дальнейшего прослушивания
 - * Выделять области интересов
 - * Выявлять общие вкусы пользователей
 - * Динамическое добавление новых элементов

