Разработка системы ссылочного ранжирования

Выполнил:

Ощепков Дмитрий Олегович

Научный руководитель доцент кафедры ПМИ:

Татаринова Александра Геннадьевна

Актуальность

- Обусловлена широким применением алгоритмов ссылочного ранжирования для решения большого спектра различных задач.
- Ранжирование результатов поиска [1]
- В социальных сетях для определения влиятельных пользователей или контента [2]
- В анализе сетей для выявления ключевых узлов [3].

Цель

Заключается в реализации известных алгоритмов ссылочного ранжирования и исследование данных алгоритмов по производительности на реальных данных, а также формирование рекомендаций по их выбору.

Задачи

1. **Выполнить обзор существующих алгоритмов** ссылочного ранжирования, описать принципы их работы на примерах. Описание и анализ методов ранжирования.
2. **Выбрать и проанализировать данные** для проведения экспериментального исследования.
3. **Реализовать алгоритмы** ссылочного ранжирования.
4. **Провести исследование** реализованных алгоритмов по производительности, проанализировать полученные результаты и сформулировать выводы.

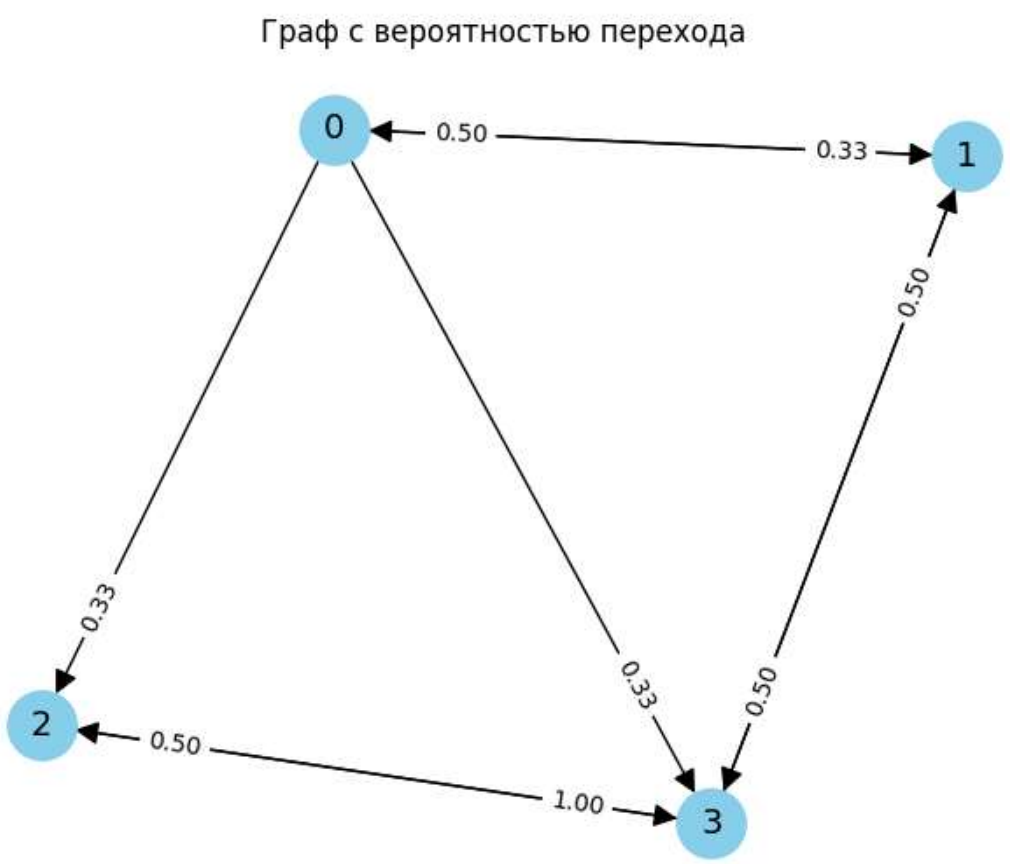
PageRank

- PageRank – это алгоритм, разработанный Google для ранжирования веб-страниц в результатах поиска. Он основан на идее, что **чем больше авторитетных страниц ссылаются на вашу страницу, тем она важнее и ценнее для пользователей**.

- $pagerank := aP^T * pagerank + b(p_u * pagerank) + c \begin{bmatrix} \frac{1}{n} \\ \dots \\ \frac{1}{n} \end{bmatrix}$,

где $a + b + c = 1$, P – матрица переходов, $pagerank$ – оценки ранжирования

PageRank



Интуитивно **узел 3 является наиболее авторитетным**, так как имеет наибольшее количество ссылок на себя

PageRank

- Одна итерация алгоритма на предыдущем графе
- $$\begin{bmatrix} 0.14375 \\ 0.213875 \\ 0.213875 \\ 0.426375 \end{bmatrix} = 0.85 \begin{bmatrix} 0 & 0.5 & 0 & 0 \\ 0.33 & 0 & 0 & 0.5 \\ 0.33 & 0 & 0 & 0.5 \\ 0.33 & 0.5 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{bmatrix} + 0.15 \begin{bmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{bmatrix}$$

- Красным вероятности перехода их входящих вершин в узел 3
- Синим вектор pagerank на предыдущем шаге
- Зеленым в контексте pagerank вектор обозначающий случайный переход к узлу 3

HITS

- HITS (Hyperlink-Induced Topic Search), как и PageRank, – это алгоритм анализа ссылок для ранжирования веб-страниц. Однако, в отличие от PageRank, HITS фокусируется на двух типах страниц: **авторитетах (authorities)** и **хабах (hubs)**.
- Итеративное правило обновление (аналогично pagerank):

$$a^+ = M^T h$$
$$h^+ = Ma,$$

где M – матрица перехода, h – вектор хабности, a – вектор авторитетности.

HITS

- Существует еще один способ найти рейтинг HITS.
- В статье [4] доказывается следующее утверждение:

a – это **главный собственный вектор** матрицы $M^T M$, а h – главный собственный вектор матрицы MM^T [4, стр. 11].

- Учитывая, что при сингулярном разложении матрицы переходов $M = U\Sigma \mathrm{V}$ столбцы матрицы V – **это собственные вектора** матрицы $M^T M$, мы можем найти a, не умножая матрицы большой размерности

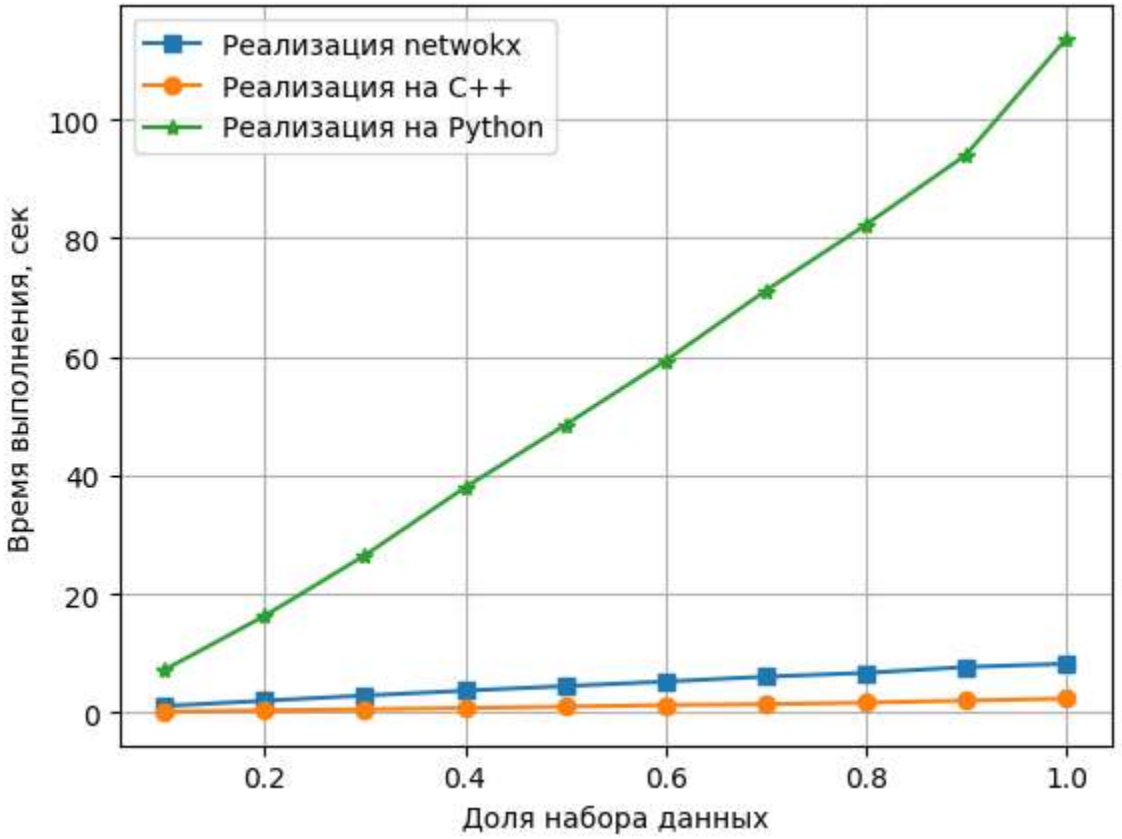
Google Web Graph

- Этот набор данных представляет собой граф веб-страниц, где узлы соответствуют веб-страницам, а ориентированные ребра – гиперссылкам между ними.
- Данные были опубликованы Google в 2002 году в рамках конкурса Google Programming Contest.
- Количество узлов: 875 713
- Количество ребер: 5 105 039

URL: https://www.kaggle.com/datasets/pappukrjha/google-web-graph (дата обращения 27.05.2024)

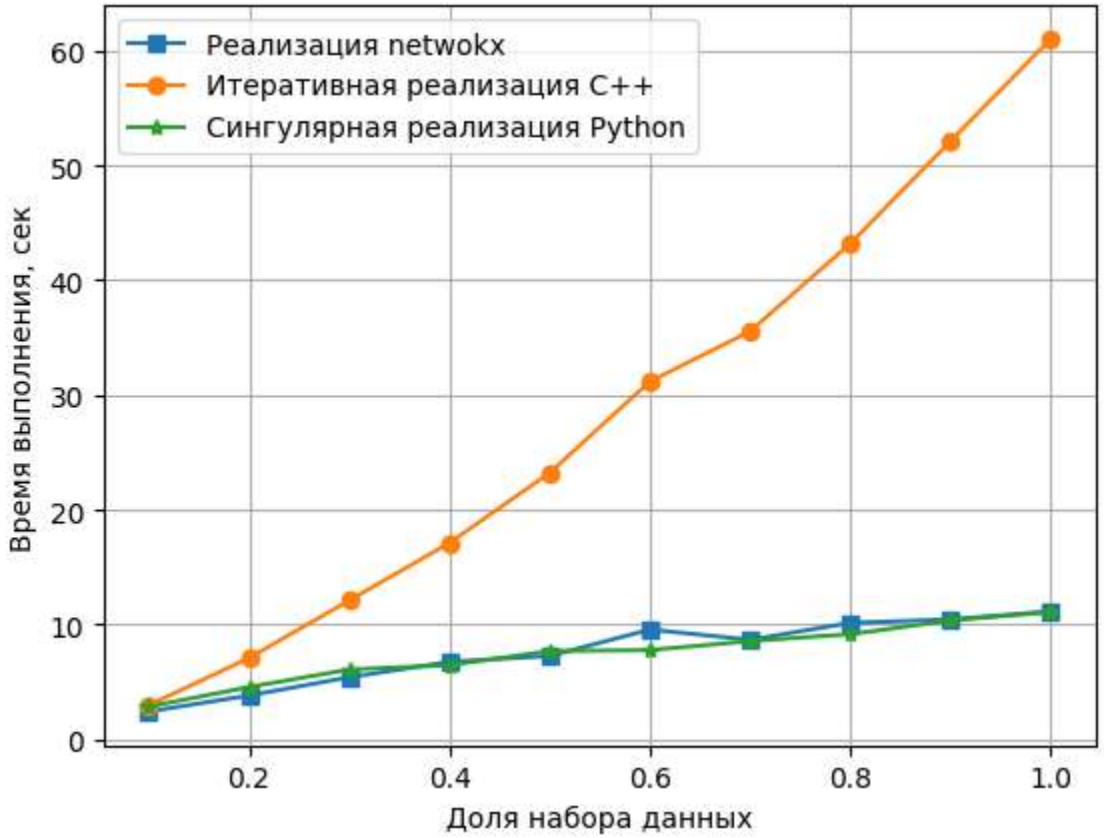
Google Web Graph

- Результаты работы Pagerank
- Эффективность C++ реализации
- Удалось получить скорость лучше, чем у networkx



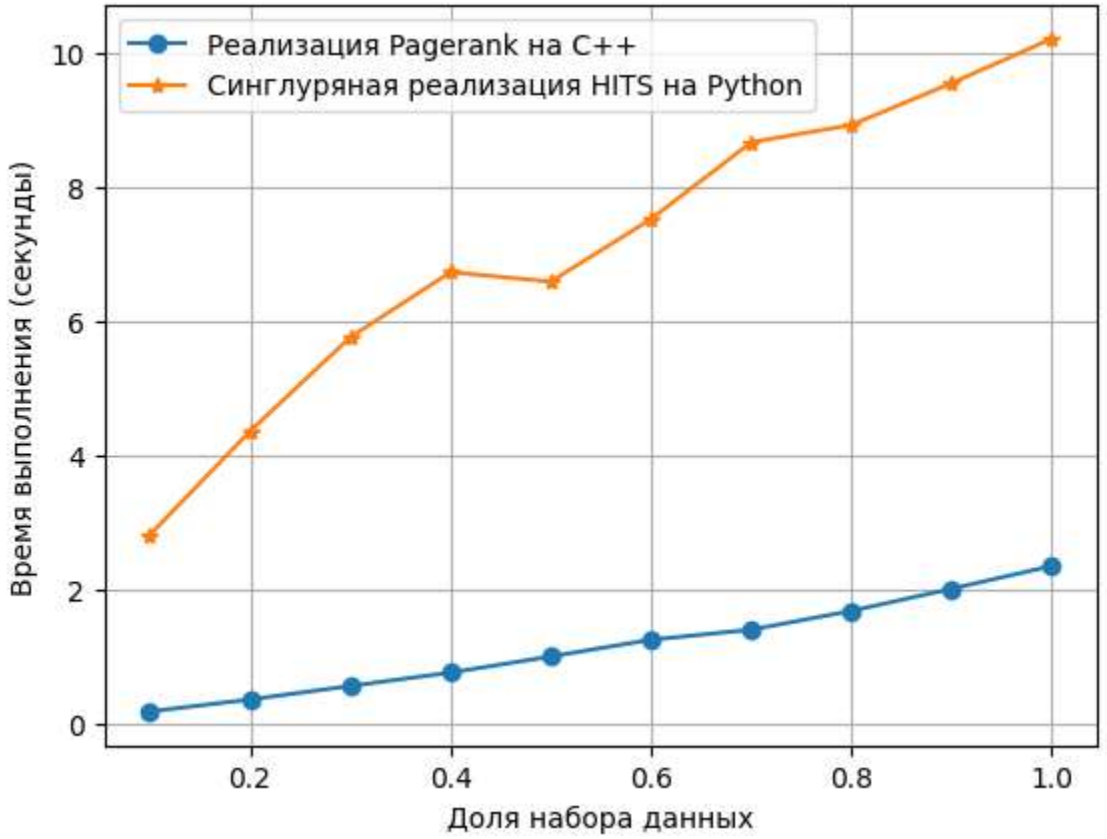
Google Web Graph

- Результаты работы HITS
- Эффективность сингулярного разложения
- Не удалось получить скорость лучше, чем у networkx (скорее всего связано с точностью сингулярного разложения)



Google Web Graph

- Сравнение производительности
- PageRank работает быстрее, чем HITS



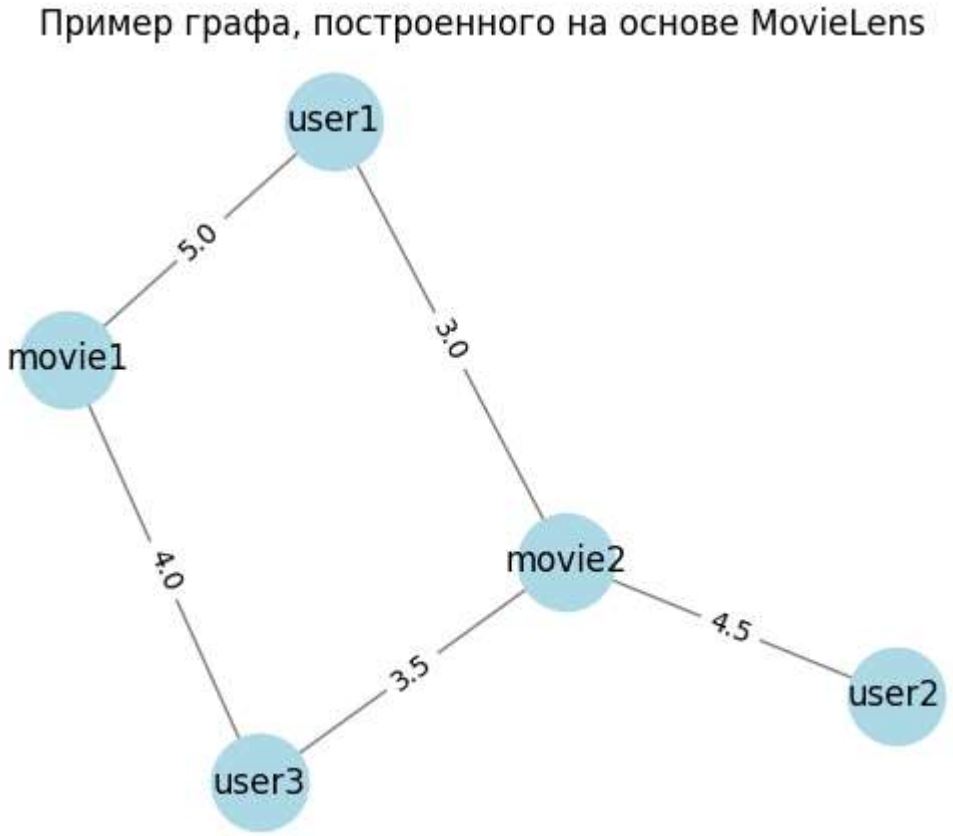
MovieLens 20M Dataset

- Набор данных описывает рейтинги и теги, присвоенные пользователями сервиса рекомендаций фильмов MovieLens.
- Он содержит 20 000 263 рейтинга и 465 564 тега, относящихся к 27 278 фильмам.
- Эти данные были созданы 138 493 пользователями в период с 9 января 1995 года по 31 марта 2015 года.
- Сам набор данных был сгенерирован 17 октября 2016 года.

URL: https://www.kaggle.com/datasets/grouplens/movielens-20m-dataset (дата обращения 27.05.2024)

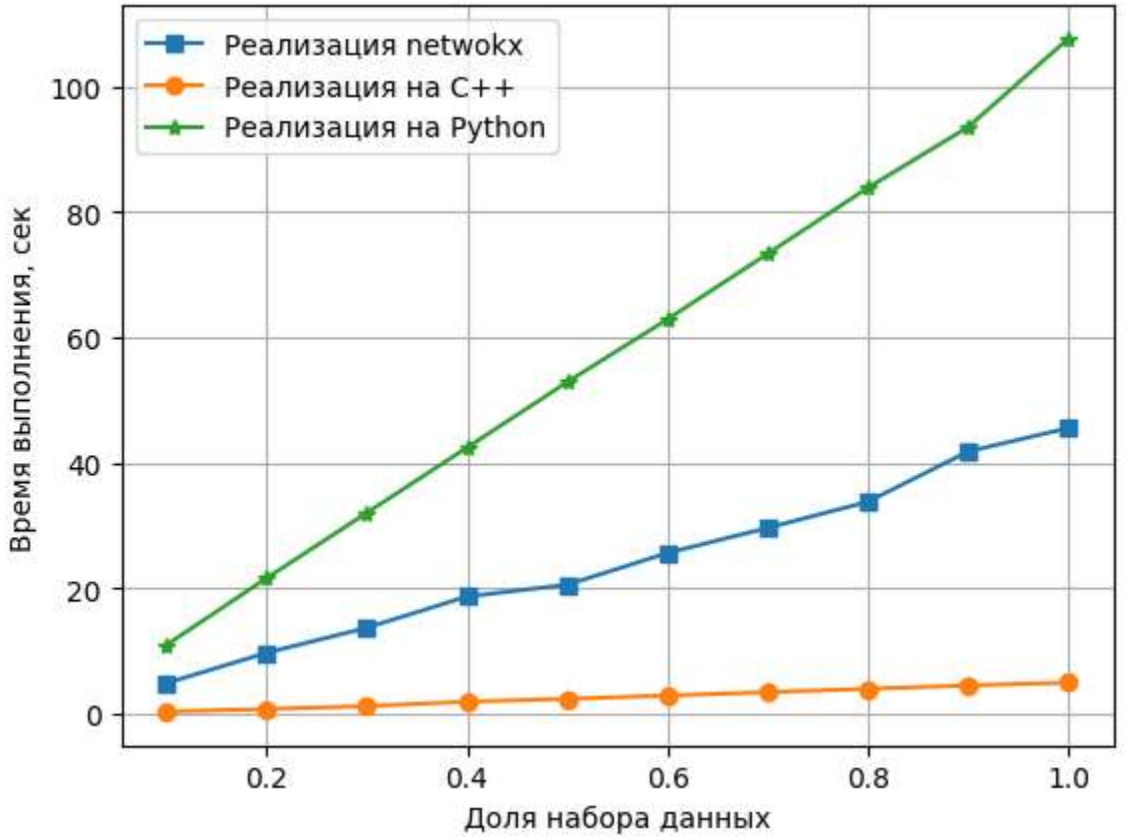
MovieLens 20M Dataset

- На основе этого датасета был построен граф подобный тому, что на рисунке
- Если пользователь поставил оценку фильму, то в графе будет ребро между пользователем и фильмом равным оценке фильма



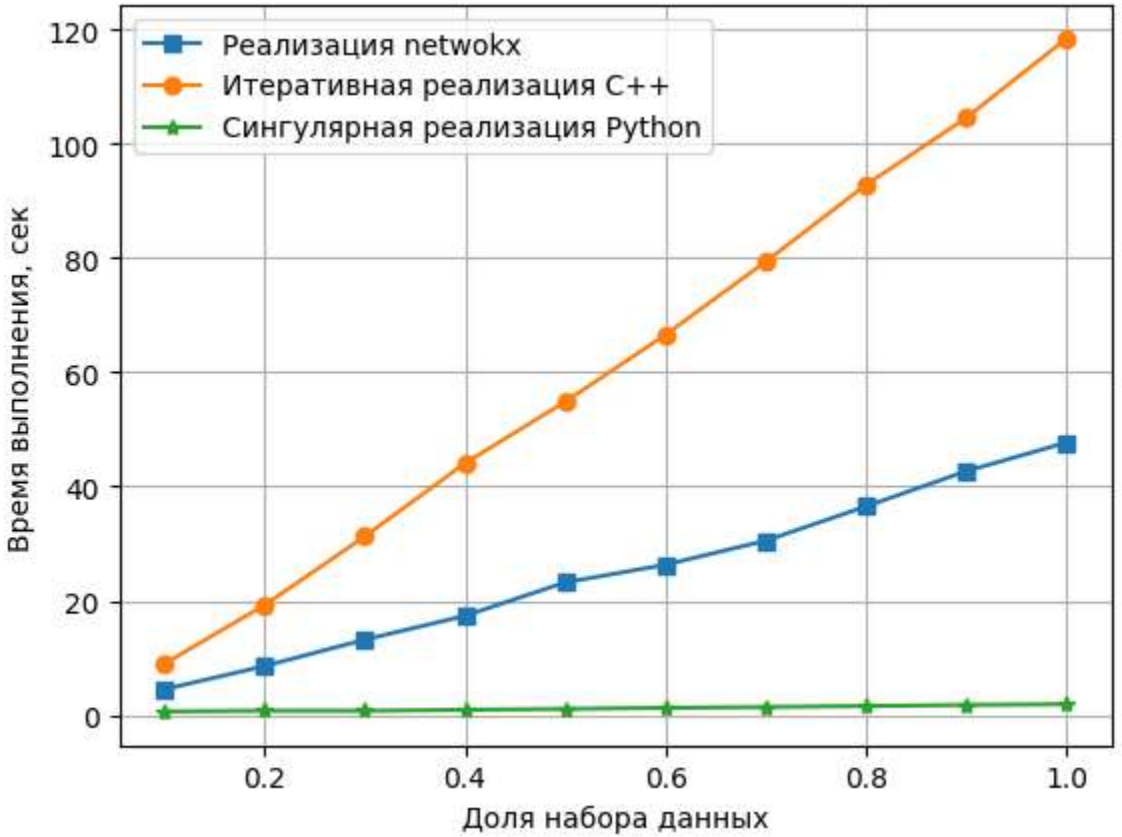
MovieLens 20M Dataset

- Результаты работы Pagerank
- Эффективность C++ реализации
- Удалось получить скорость лучше, чем у networkx



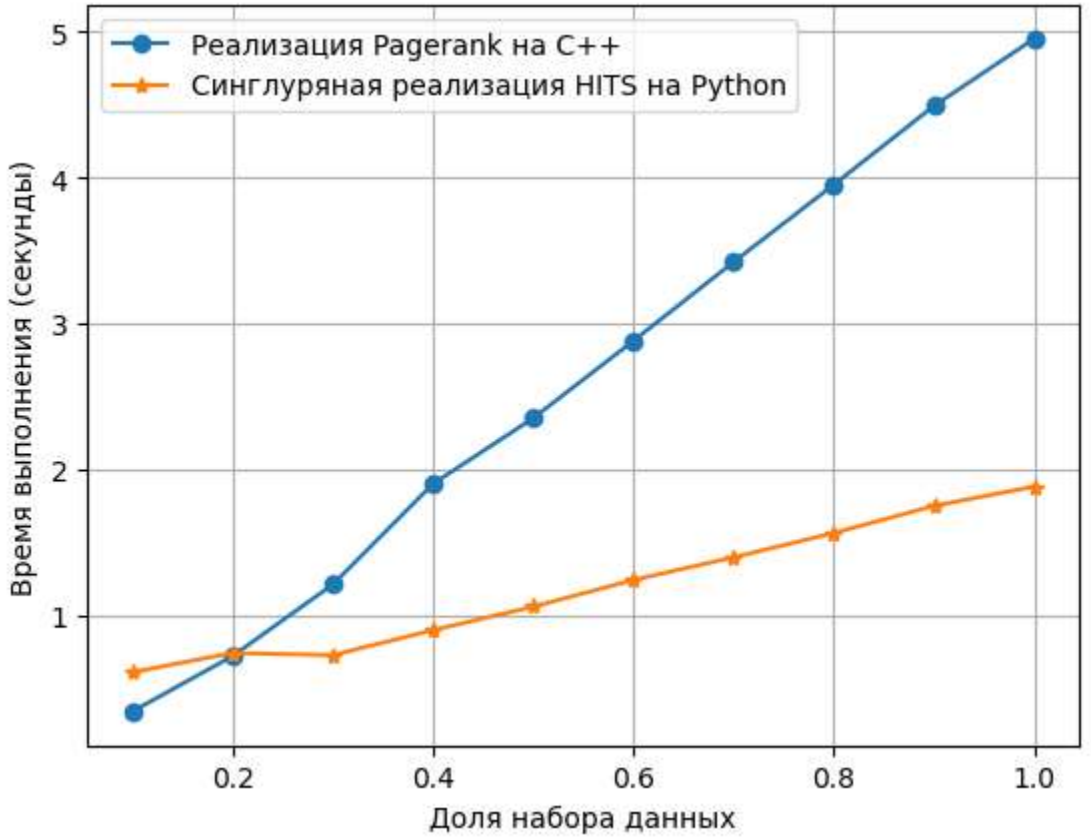
MovieLens 20M Dataset

- Результаты работы HITS
- Эффективность сингулярного разложения
- Удалось получить скорость лучше, чем у networkx



MovieLens 20M Dataset

- Сравнение производительности алгоритмов
- HITS работает быстрее, чем Pagerank



MovieLens 20M Dataset

- Метрики качества.
- Мы считаем фильм «хорошим», если пользователь оценил фильм выше, чем данная им средняя оценка всем фильмам.

Метрика	HITS	PageRank
Accuracy	**0.5681**	0.5340
Precision	**0.5492**	0.5250
Recall	0.8666	**0.9333**
F1 Score	0.6724	0.6720
Pairorder	0.6712	0.6701

Выводы

- Предпочитайте **способ с сингулярным разложением** методу простых итераций в алгоритме HITS, если только нет предпосылок, что алгоритм сойдется за малое количество итераций.
- Используйте библиотеки, которые работают с графом **в нужном формате** или реализуйте алгоритмы и хранение данных вручную так, чтобы **избежать конвертацию данных**. Это значительно ускорит процесс ранжирования.
- На датасете MovieLens **PageRank продемонстрировал более высокий recall**, в то время **как HITS продемонстрировал более высокий precision**. Выбирайте тот или иной алгоритм в зависимости от ваших требований по этим метрикам.

Спасибо за внимание

Литература

1. Schütze, Hinrich, Christopher D. Manning, and Prabhakar Raghavan. Introduction to information retrieval. Vol. 39. Cambridge: Cambridge University Press, 2008.
2. Brin, Sergey, and Lawrence Page. "The anatomy of a large-scale hypertextual web search engine." *Computer networks and ISDN systems* 30.1-7 (1998): 107-117.
3. Shahriari, Moshen, and Mahdi Jalili. "Ranking nodes in signed social networks." Social network analysis and mining 4 (2014): 1-12.

Литература

4. Kleinberg, Jon M. "Authoritative sources in a hyperlinked environment." *Journal of the ACM (JACM)* 46.5 (1999): 604-632.

Метрики

- Формализуем человеческую оценку в функциональный вид
- $\mathrm{O}(p) = \begin{cases} 0 \quad if\ p\ is\ bad \\ 1 \quad if p\ is\ good \end{cases}$
- Где $p \in V$, а V – множество вершин графа страниц

Метрики

- $I(T, O, p, q) = \begin{cases} 1 \; if \; T(p) \geq T(q) \; and \; O(p) < O(q) \\ 1 \; if \; T(p) \leq T(q) \; and \; O(p) > O(q) \\ 0 \; otherwise \end{cases}$, где
- T – функция, возвращающая ранг вершины согласно алгоритму, а p и q – это вершины графа.
- Теперь можем сформулировать pairwise orderedness
- $pairord(T, O, P) = \frac{|P| - \sum_{(p,q) \in P} I(T,O,p,q)}{|P|}$
- Если pairord равна 1, не существует случаев, когда T неверно оценил пару. Наоборот, если pairord равна нулю, то T неверно оценил все пары.

Метрики

- $\text{prec}(T, O) = \frac{\left|\{p \in \chi | T(p) > \delta \text{ and } O(p) = 1\}\right|}{\left|\{q \in \chi | T(q) > \delta\}\right|}$
- $prec = \frac{\mathrm{TP_i}}{\mathrm{TP_i + FP_i}}$
- $\text{rec}(T, O) = \frac{\left|\{p \in \chi | T(p) > \delta \text{ and } O(p) = 1\}\right|}{\left|\{q \in \chi | O(p) = 1\}\right|}$
- $\mathrm{rec} = \frac{\mathrm{TP_i}}{\mathrm{TP_i + FN_i}}$
- $\mathrm{F_\beta} = \frac{(\beta^2+1)\pi p}{\beta^2 \pi + p}$

Category		$O(p) = 1$	
		YES	NO
$T(p) > \delta$	YES	TP_i	FP_i
	NO	FN_i	TN_i