НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

Факультет компьютерных наук
Образовательная программа бакалавриата «Прикладная математика и информатика»
Программный проект
Сложные сети. Граф связей и граф данных

Выполнил студент группы БПМИ-165
Леонов Алексей Олегович
Научный руководитель:
доцент ДАДИИ, д. ф.-м. н., проф.
Громов Василий Александрович

ОСНОВНЫЕ ПОНЯТИЯ, ОПРЕДЕЛЕНИЯ, ТЕРМИНЫ

Сложные сети – графы, обладающие определёнными свойствами (высокий коэффициент кластеризации, малый диаметр, малое среднее расстояние между вершинами (“свойства малого мира”), степенной закон распределения различных характеристик).
(примеры: сеть нейронов головного мозга, графы дорог, социальные графы, трофические сети)

Граф связей –- граф явно заданных связей между парами вершин.

Граф данных –- граф, построенный на основе метаданных, ассоциированных с вершинами.

Сообщества – группы вершин с высокой плотностью рёбер внутри групп и низкой плотностью рёбер между ними.
(здесь и далее рассматриваются разбиения на непересекающиеся сообщества)

ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

Предметная область

Различные способы представления больших данных.

Неформальная постановка задачи:

Разработать программное обеспечение для анализа данных с помощью построения графа данных и поиска сообществ в них. С его помощью сравнить свойства различных графов данных на примере нескольких наборов данных.

ЦЕЛЬ И ЗАДАЧИ РАБОТЫ

Цель работы

Написать библиотеку для построения графов данных и поиска сообществ в них.

Задачи работы

1. Реализовать алгоритмы поиска сообществ.
2. Реализовать алгоритмы построения графов данных.
3. Найти наборы данных с метаданными и явно заданными связями.
4. Выполнить построение графов данных по ним.
5. Осуществить поиск сообществ в исходных и построенных графах.
6. Проанализировать характеристики полученных данных, провести сравнительный анализ [исполнитель – студент 4 курса ПМИ, Станислав Рыбин]

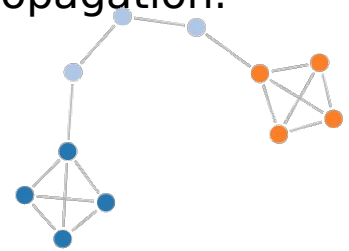
Аналоги

Графы данных:

Proxi – библиотека с открытым исходным кодом на языке Python.
Недостатки: низкая производительность, малое количество графов данных.

Поиск сообществ:

Networkx – библиотека с открытым исходным кодом на языке Python.
Недостатки: низкая производительность, отсутствие настраиваемого критерия остановки в алгоритмах. Только асинхронная версия алгоритма label propagation.



NetworkX

ВЫБОР МОДЕЛЕЙ, МЕТОДОВ И АЛГОРИТМОВ

Требования к алгоритмам поиска сообществ:

1. Малое число параметров, слабая зависимость результата от них.
2. Масштабируемость.

Выбранные алгоритмы:

1. Алгоритм проталкивания меток (Label propagation).

2. Алгоритм Клосета-Ньюмана-Мура (Clauset-Newman-Cristopher Moore, CNM)

Label propagation

1) Алгоритм назначает каждой вершине своё сообщество.

2) Далее на каждом шаге присваивает вершине сообщество, наиболее часто встречающееся среди её соседей (если таких несколько, случайное из них).

3) Процесс повторяется несколько итераций (есть различные критерии остановки).

Существуют асинхронная и синхронная модификации алгоритма.

Label propagation (псевдокод)

Синхронная версия:

```
function label_propagation(graph, n_iter):
  labels = [i for i in range(len(graph))]
  for  iter in range(n_iter):
    labels = [most_common_community(graph, labels, i)
                                    for i in range(len(graph))]

  return labels
```

Асинхроная версия:

```
function label_propagation(graph, n_iter):
  labels = [i for i in range(len(graph))]
  for iter in range(n_iter) {
    for i in range(len(graph)):
      labels[i] = most_common_community(graph, labels, i)
   return labels
```

Clauset-Newman-Moore

1) Алгоритм назначает каждой вершине своё сообщество.

2) На каждом шаге объединяет 2 сообщества, так чтобы максимизировать увеличение целевой функции на данном шаге.

3) Объединения происходят, пока такое увеличение возможно.

Сложность $O(n \log^2 n)$ для сложных сетей, где n – размер графа (суммарное количество вершин и ребёр). Нужно использовать при реализации правильные структуры данных (например, хранить разреженную матрицу ΔQ_{ij} , для каждого сообщества количество рёбер с концами в нём и т.п.).

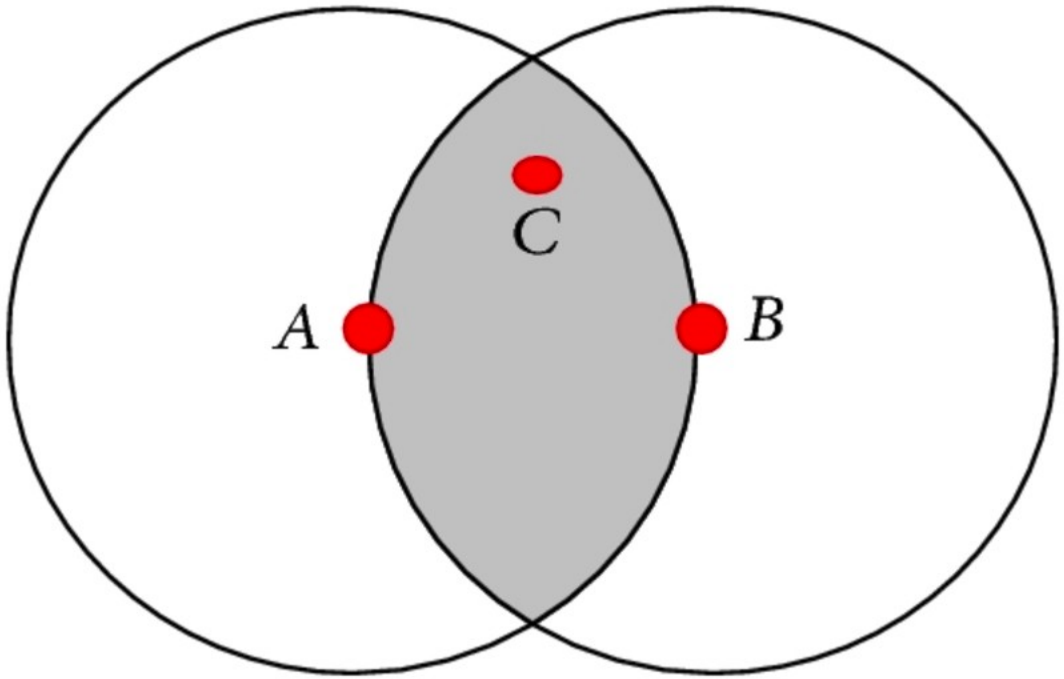
Построение графов данных

Выбранные методы построения:

1. Эпсилон-граф (с автоматическим поиском значения эпсилона)
2. Граф k-ближайших соседей.
3. Граф сфер влияний.
4. Граф относительного соседства (RNG, relative neighborhood graph).
5. Граф Габриеля.

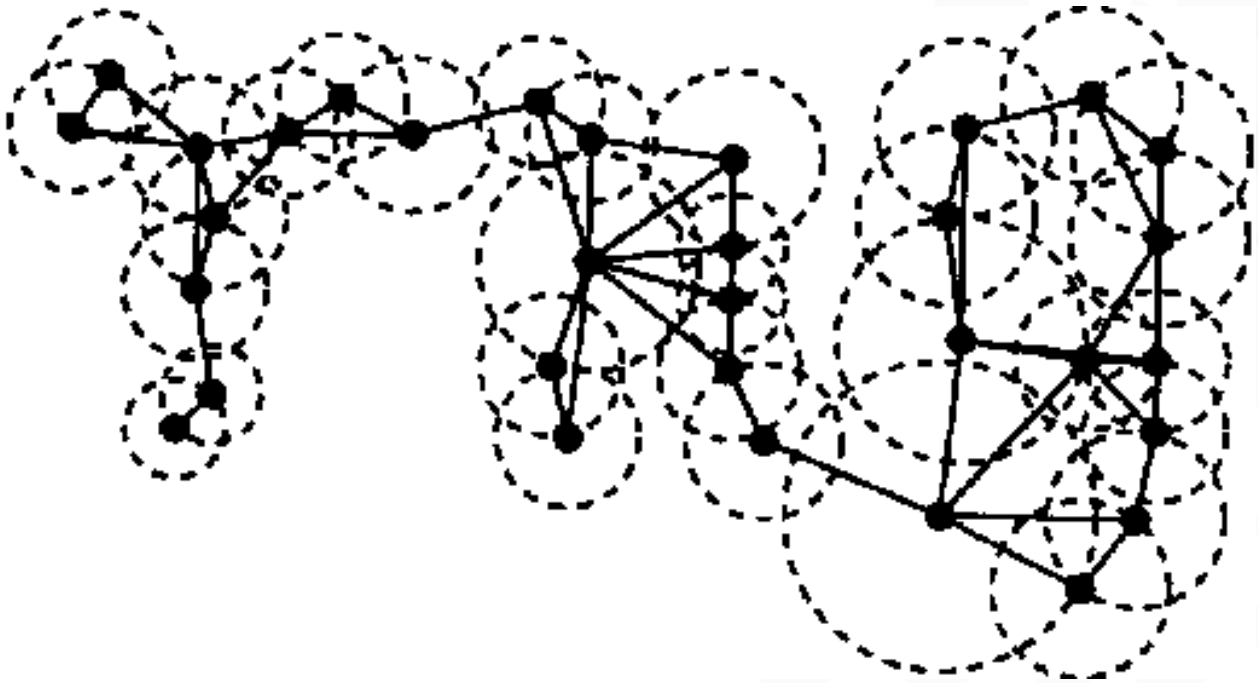
Граф относительного соседства.

Две вершины соединяются ребром, если в пересечении шаров с центрами в них и радиусами, равными расстоянию между ними, нет других точек.



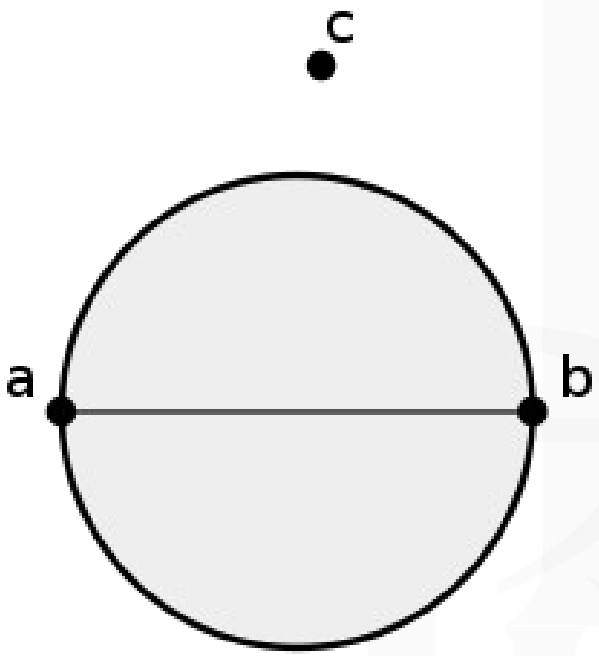
Граф сфер влияния

Для каждой вершины строится шар с центром в этой вершине и радиусом, равным расстоянию до её ближайшего соседа. Далее, две вершины соединяются ребром, если соответствующие им шары пересекаются.



Граф Габриеля

Две вершины соединяются ребром, если внутри шара построенного на отрезке между ними как на диаметре, нет других вершин.



ТЕХНОЛОГИИ И ИНСТРУМЕНТЫ РЕАЛИЗАЦИИ

Языки разработки:

C++, Python3.

Архитектура:

1) Библиотека на языке C++.
2) Интерфейс для неё на языке Python3.

Обмен данными между ними по текстовому протоколу.
Интерфейс использует структуру данных networkx.Graph из библиотеки networkx, версии 2.2.

ТЕХНОЛОГИИ И ИНСТРУМЕНТЫ РЕАЛИЗАЦИИ

Датасеты, для которых использовалась написанная программа.

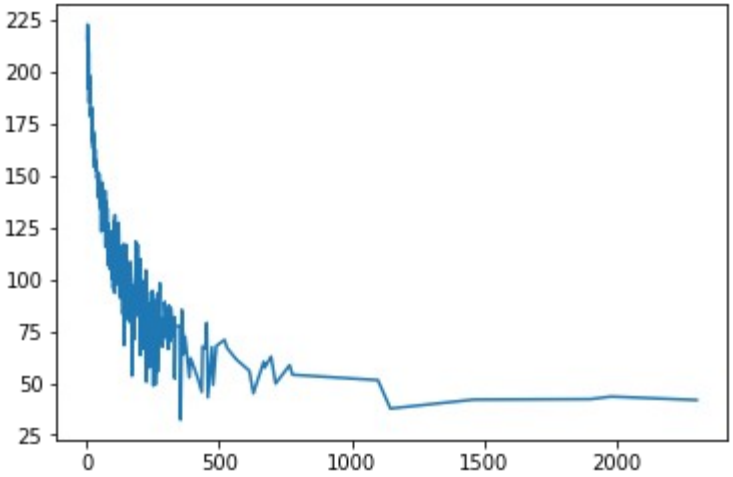
1) 6 degrees of Francis Beacon.
Социальный граф, основанный на исторических данных. 15.801 вершин, 171.408 рёбер.
Метаданные: годы жизни, пол, род занятий, имя и титул (если есть).

2) Граф подписок Твиттера.
112.416 вершин, 308927 рёбер.
Метаданные: твиты пользователя – их текст, устройство, с которого они отправлялись, время и т.п.

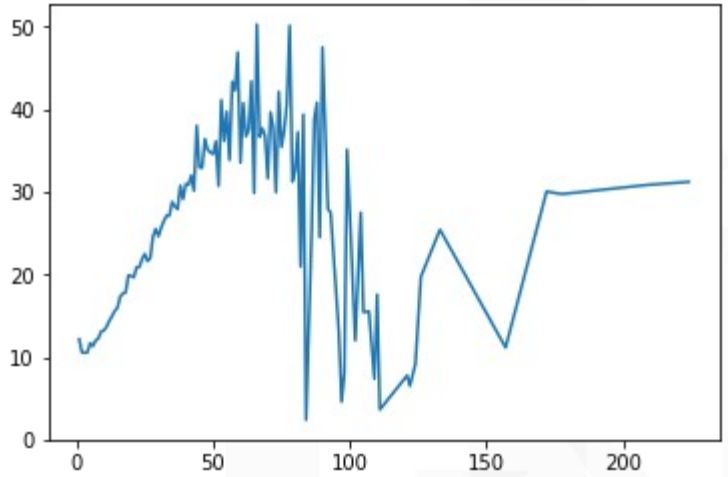
3) Граф товаров на сайте Amazon.com, часто покупаемых вместе.
548552 вершин, 987942 рёбер.
Метаданные: название и категории товара, количество покупок и отзывов, рейтинг.

Ассортативность данных

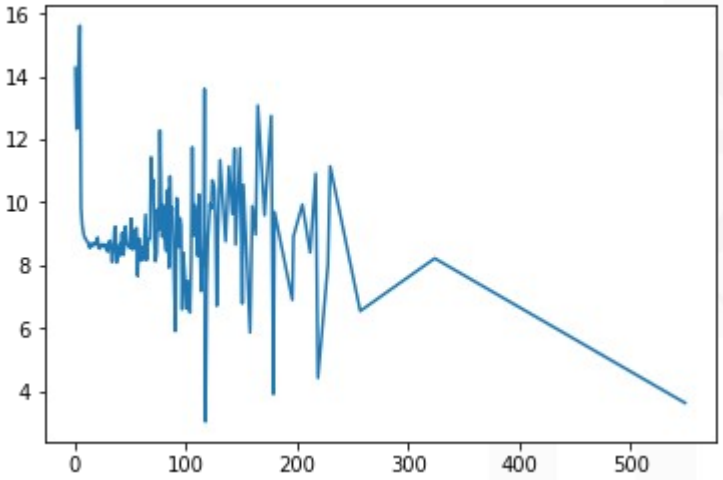
Социальный граф



Twitter



Amazon



Время работы

Таблица 2. Время построения графов данных, в секундах.

--	**6 degrees of Francis Bacon**	**Twitter**	**Amazon**
Граф ε-шаров (|E| ≈ 8 |V|)	57	291	12884
Граф k ближ. сосед. (k=8)	297	6533	95690
Граф сфер влияния	46	697	15012
Граф отн. соседства	924	--	--
Граф Габриеля	2324	--	--

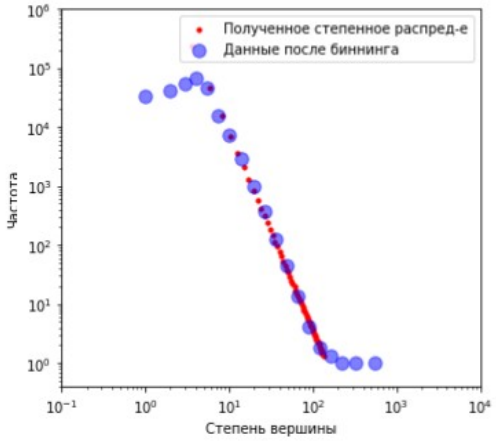
Время работы

Таблица 3. Время работы label propagation (100 итераций), в секундах

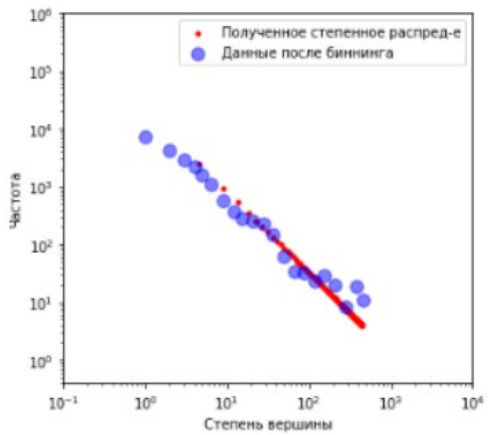
--	**6 degrees of Francis Bacon**	**Twitter**	**Amazon**
Исходный граф	14	52	229
Граф ε-шаров ($\|E\| \approx 8\|V\|$)	9	69	448
Граф k ближ. сосед. (k=8)	7	81	451
Граф сфер влияния	3	42	313
Граф отн. соседства	5	--	--
Граф Габриеля	11	--	--

Распределение степеней вершин
(Граф amazon, двойной лог. масштаб)

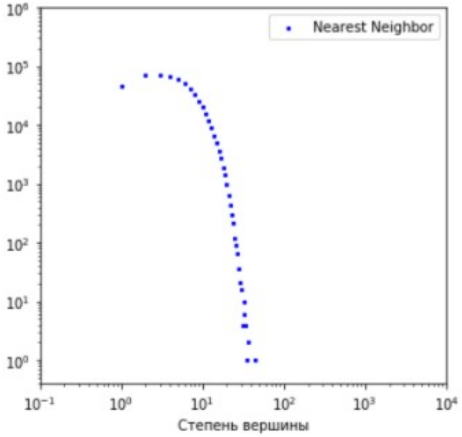
Исходный граф



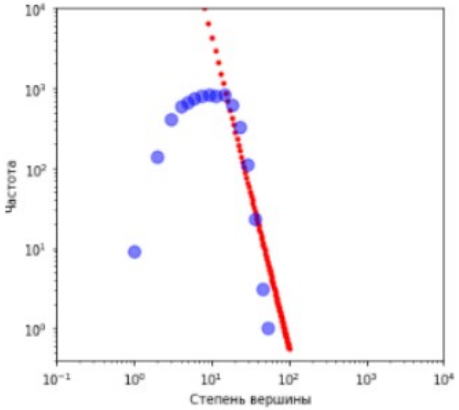
Эпсилон-граф



Граф k ближ. соседей

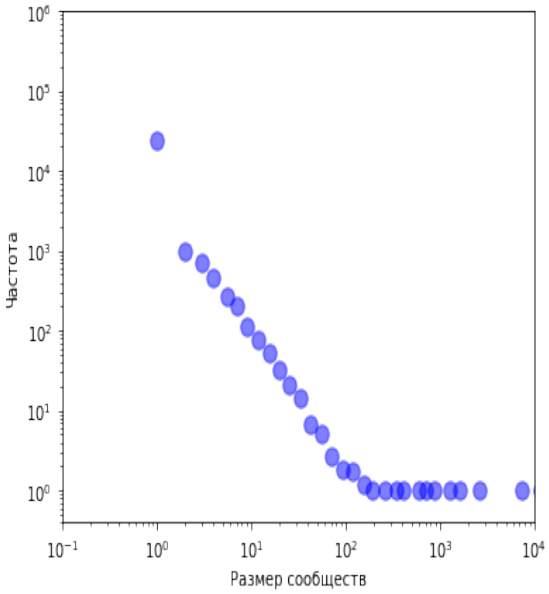


Граф Габриеля

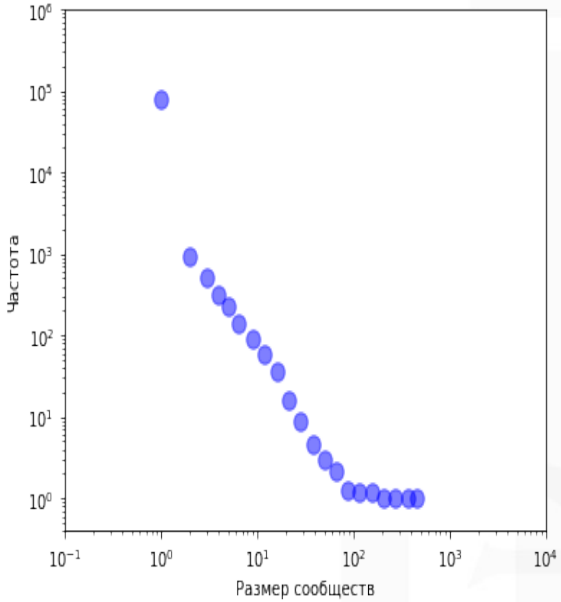


Распределение размеров сообществ
(Граф Twitter, двойной лог. масштаб)

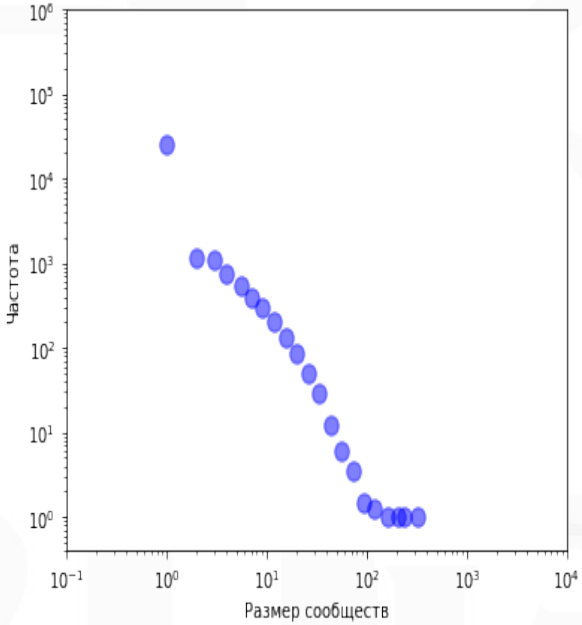
Граф связей



Эпсилон граф



Граф k ближ. соседей



Другие характеристики
(Граф 6dfb)

	Средний коэф. кластеризации	Средняя степень вершины	Диаметр
Граф связей	0.218	21.69	9
Граф эпсилон	0.13	14.91	29
Граф Габриэль	0.31	13.81	17
Граф сфер влияния	0.18	1.95	30
Граф ближайших соседей	0.60	8.52	81
Граф относительного соседства	0.20	3.64	37

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Raghavan, U. Near linear time algorithm to detect community structures in large-scale networks / Usha Nandini RagTavan, Reka Albert, Soundar Kumara // Physical Review E 76, 2007 .
2. Clauset, A. Finding community structure in very large networks / Aaron Clauset, M. E.
J. Newman,and Cristopher Moore // Phys. Rev. E 70, 2004.
3. Proxi [Электронный ресурс]. Режим доступа: https://proxi.readthedocs.io/en/latest/index.html, свободный. (дата обращения: 1.06.19)
4. Networkx [Электронный ресурс]. Режим доступа: https://networkx.github.io, свободный. (дата обращения: 1.06.19)
5. Six Degrees of Francis Bacon [Электронный ресурс]. Режим доступа: http://www.sixdegreesoffrancisbacon.com, свободный. (дата обращения: 1.06.19)
6. Datasets for Social Network Analysis [Электронный ресурс]. Режим доступа: https://aminer.org/data-sna#Twitter-Dynamic-Net, свободный. (дата обращения: 1.06.19)

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ (продолжение)

7. Stanford Large Network Dataset Collection [Электронный ресурс]. Режим доступа: https://snap.stanford.edu/data/, свободный. (дата обращения: 1.06.19)
8. Fortunato, S. Community detection in networks: A user guide / Santo Fortunato, Darko Hric // Physics Reports 659, 1-44, 2016.
9. Fortunato, S. Community detection in grapTs / Santo Fortunato // Physics Reports 486, 75-174, 2010.
10. Beck, M. Computational Discrete Geometry / Matthias Beck.

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

Спасибо за внимание!

Леонов Алексей Олегович,
aoleonov@edu.hse.ru

Москва - 2019