Федеральное государственное бюджетное учреждение науки
Институт системного программирования Российской академии наук

На правах рукописи

КОРШУНОВ АНТОН ВИКТОРОВИЧ

ИССЛЕДОВАНИЕ СТРУКТУРЫ СООБЩЕСТВ ПОЛЬЗОВАТЕЛЕЙ В ГРАФАХ ОНЛАЙНОВЫХ СОЦИАЛЬНЫХ СЕТЕЙ

Специальность 05.13.11 — математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей

Диссертация на соискание учёной степени
кандидата физико-математических наук

Научный руководитель:
доктор технических наук,
главный научный сотрудник ИСП РАН
Кузнецов Сергей Дмитриевич

Москва – 2015

Содержание

Введение

Актуальность

Работа посвящена актуальной теме исследования структуры сообществ пользователей онлайновых социальных сетей. Данная задача рассматривается как частный случай задачи категоризации вершин графа, основанной исключительно на его структурных свойствах. В случае социальных сетей под структурными свойствами понимается специфическая для конкретного социального графа конфигурация рёбер (социальных связей) между его вершинами (пользовательскими аккаунтами).

В современном понимании *онлайновая социальная сеть* — это Интернет-сервис, позволяющий пользователям публиковать на своих страницах персональные и иные данные и предназначенный для упрощения коммуникации и обмена информацией между пользователями сети Интернет. Социальные сети являются важным инструментом компьютерно-опосредованной коммуникации, стремительно набирающим популярность по всему миру в течение последних двух десятилетий. К примеру, в марте 2015 года социальная сеть Facebook сообщает об 1,39 миллиарде[1], а Twitter — о 288 миллионах пользователей[2], которые совершают какие-либо действия в сети хотя бы 1 раз в месяц.

Помимо коммуникационной функции, сервисы социальных сетей играют роль баз пользовательских данных, в которых с каждым пользователем ассоциирован набор персональной информации, составляющий его “виртуальную личность”. Данные всех пользователей некоторой сети образуют её *социальный граф*, — динамическую структуру, полностью описывающую состояние и поведение составляющих её пользователей, а также их отношения между собой и объектами внешнего мира в некоторый момент времени. Вершинами

[1] http://newsroom.fb.com/company-info/

[2] https://about.twitter.com/company

графа принято считать пользовательские аккаунты, а рёбрами — социальные связи типа “дружба”, “подписка”, “следование”, которыми пары пользователей явно связывают свои аккаунты.

Являясь по сути виртуальным отражением части человеческого социума, социальные сети во многих случаях наследуют характерную для него структуру социальных групп, или *сообществ*. Наличие сообществ пользователей является распространённым свойством современных онлайновых социальных сетей вне зависимости от состава аудитории, природы связей и преобладающих сценариев использования: общение, обмен контентом, поиск информации, развлечения и т.д. Например, часто можно встретить сообщества, в которых участников объединяют общие интересы, политические и религиозные предпочтения, географическая близость и т.д.

Таким образом, с точки зрения сетевого анализа сообщества представляют собой *кластеры* пользователей, связанных между собой сильнее, чем с другими пользователями сети. С функциональной точки зрения *сообщество* — это группа пользователей, выполняющая общую роль или функцию и обладающая общими свойствами, ценностями и целями.

С точки зрения интерфейса социальных сервисов возможно 2 способа объединения пользователей в сообщества. В первом случае пользователи явно указывают своё членство в группе путём вступления в неё. Во втором случае связь с некоторой группой устанавливается неявно путём образования социальных связей, основанных на общей роли, деятельности, круге общения, интересах, функциях или каких-либо других свойствах. В этом случае для определения структуры сообществ пользователей требуются специализированные методы [1–4]. Большинство методов использует для анализа только социальные связи между пользователями, поскольку они являются неотъемлемым элементом любой социальной сети. Таким образом, задачу определения структуры сообществ в социальном графе можно свести к задаче поиска особых кластеров пользователей на основе анализа социальных связей между ними. С другой стороны, данную задачу можно рассматривать как разновидность категоризации пользователей в одну или несколько заранее неизвестных групп.

Представителями ведущих научных групп, занимающихся исследованием структуры сообществ в социальных сетях, являются Jure Leskovec из Стэн-

фордского университета, США [5–12] и Santo Fortunato из Университета Аалто, Финляндия [1, 13, 14]. Среди отечественных учёных данной тематике посвящены работы С.Н. Пупырева [15], М.И. Коломейченко и соавторов [16], а также работы коллектива исследователей из Института системного программирования РАН.

Знание структуры сообществ пользователей находит применение в ряде практических приложений анализа социальных данных: определение значений скрытых атрибутов пользователей, оптимизация передачи сообщений в коммуникационных сетях, ограничение распространения вредоносного программного обеспечения, идентификация распространителей спам-сообщений, рекомендация товаров, услуг и контента пользователям социальных сетей и др.

Определение структуры сообществ является вычислительно сложной и во многих случаях плохо масштабируемой задачей. Вместе с тем, количество пользователей в современных социальных графах требует разработки масштабируемых методов. Однако временная сложность известных масштабируемых методов BigCLAM [12], SCD [17], Louvain [18] и некоторых других превышает линейную по количеству рёбер, что затрудняет их применимость к графам с $> 10^6$ вершин и средней степенью > 100, характерной для социальных сетей. Метод LPA [19], в свою очередь, обладает требуемой временной сложностью и имеет масштабируемую реализацию. Однако данный метод, как и большинство других известных методов, не позволяет присваивать пользователю более одного сообщества. Следовательно, множества вершин результирующих сообществ не пересекаются, а результатом работы является *разбиение* исходного графа.

Вместе с тем, согласно результатам проведённого в 2012 году исследования свойств групп пользователей четырёх социальных сетей [9], множества пользователей групп имеют тенденцию к значительному пересечению. Например, в социальных сетях LiveJournal и Orkut пользователь состоит в среднем в 3.09 и 95.9 публичных сообществах соответственно. Предположительно, этот факт связан с тем, что каждый пользователь образует связи в соответствии с несколькими основными ролями: семья, коллеги, люди с общими интересами и т.д.

Метод SLPA [20] позволяет находить сообщества пользователей с пересекающимися множествами вершин и имеет временную сложность, линейно зависящую от количество рёбер исходного графа. Кроме того, метод позволяет эффективную реализацию в рамках современных программных фреймворков для распределённых вычислений (Apache Hadoop, Spark, Giraph). Однако экспериментальные исследования качества метода с помощью синтетических графов с управляемыми свойствами заданной структуры сообществ выявили тенденцию к резкому ухудшению результатов с увеличением среднего количества сообществ у пользователя[3], что ограничивает применимость SLPA к реальным данным.

Таким образом, актуальной является разработка масштабируемых методов определения структуры сообществ пользователей со значительным пересечением вершин в графах онлайновых социальных сетей. Для обеспечения применимости метода в реальных приложениях зависимость времени обработки социального графа от количества рёбер должна быть близка к линейной. Кроме того, качество метода должно быть высоким вне зависимости от количества сообществ у пользователей.

Для оценки качества методов определения структуры сообществ пользователей принято оценивать близость двух множеств сообществ для некоторого графа: найденного алгоритмом и *референтного*, то есть заранее заданного или известного. Такой подход позволяет исследовать способность различных методов восстанавливать структуру сообществ, заданную особым способом, зависящим от конкретного приложения или исследовательской задачи.

При этом сбор тестовых данных из реальных социальных сетей является трудоёмким, а свойства полученных наборов данных часто далеки от желаемых. Поэтому принято использовать программные средства для генерации случайных социальных графов со структурой сообществ пользователей, заданной в соответствии с некоторой моделью.

Однако известные методы генерации таких графов не учитывают ряда важных структурных свойств сообществ. В частности, согласно результатам вышеупомянутого исследования реальных социальных графов, распределение количества сообществ у пользователя подчиняется степенному закону, что игнорируется в методах GN [21], LFR [13] и др. Кроме того, извест-

[3]Экспериментальные данные приведены в разделе 4.6.1

ные методы имеют существенные ограничения в плане производительности при генерации графов с $> 10^6$ вершин, что затрудняет оценку применимости методов определения структуры сообществ к социальным графам большой размерности.

Таким образом, недостаточное качество известных масштабируемых методов определения структуры сообществ пользователей социальной сети, а также недостаточная достоверность известных способов тестирования качества таких методов затрудняют их использование в реальных приложениях и сервисах, связанных с анализом социальных данных, что обуславливает актуальность темы диссертационной работы.

Целью диссертационной работы является разработка моделей, методов и программных средств для исследования структуры сообществ пользователей в графах онлайновых социальных сетей. Разрабатываемые модели, методы и программные средства должны сочетать низкую вычислительную сложность, хорошую масштабируемость и высокое качество работы вне зависимости от количества сообществ у пользователей.

Для достижения поставленной цели были поставлены и решены следующие **задачи**:

1) исследовать структурные свойства сообществ пользователей в графах онлайновых социальных сетей, методы определения структуры сообществ пользователей, а также методы генерации случайных графов, обладающих свойствами социальных графов и заданной структурой сообществ пользователей;

2) разработать и реализовать метод генерации случайных социальных графов с заданной структурой сообществ пользователей;

3) разработать и реализовать метод определения структуры сообществ пользователей в социальном графе;

4) провести экспериментальное исследование качества, производительности и масштабируемости разработанных методов, а также оценку их применимости для решения прикладных задач.

Основные положения, выносимые на защиту:

1) разработан распределённый метод CKB для генерации случайных социальных графов с заданной структурой сообществ пользователей;

2) разработан распределённый метод EgoLP для определения структуры сообществ пользователей в социальном графе;

3) для экспериментального подтверждения эффективности предложенных методов реализованы прототипы систем для определения структуры сообществ пользователей и генерации случайных социальных графов с заданной структурой сообществ пользователей[4]. Реализованные прототипы позволили подтвердить высокое качество предложенных методов и соответствие экспериментальных оценок производительности теоретическим оценкам вычислительной сложности.

Научная новизна

В диссертационной работе предложены два новых метода исследования структуры сообществ пользователей в социальных графах.

Метод CKB позволяет осуществлять распределённую генерацию случайных социальных графов с заданной структурой сообществ пользователей, обладающей характерным для реальных социальных сетей набором свойств. Параметрами метода являются количество пользователей и параметры распределения размеров сообществ и распределения количества сообществ у пользователя. Кроме того, предусмотрена возможность управления вероятностью ребра в сообществе в зависимости от его размера, а также регуляции среднего коэффициента кластеризации вершин в сообществе. Экспериментально продемонстрировано, что синтезируемые графы обладают всеми описанными в работе свойствами социальных графов с сообществами. Программная реализация метода обладает масштабируемостью, близкой к линейной, что при достаточном размере вычислительного кластера позволяет генерировать графы из сотен миллионов вершин за несколько часов. Таким образом, предложенный метод превосходит известные методы по совокупности масштабируемости и количества поддерживаемых свойств социальных графов с сообществами.

[4]Веб-демонстрация прототипа метода CKB доступна по адресу: http://ckb.at.ispras.ru/home/

Метод EgoLP позволяет определять структуру сообществ пользователей в социальном графе. Основой метода является итеративная пересылка меток сообществ по рёбрам графа в соответствии с установленными правилами взаимодействия вершин. Экспериментально продемонстрировано, что предложенный метод превосходит известные методы по совокупности критериев: а) близость определённой структуры сообществ с заранее известной; б) точность решения прикладной задачи определения скрытых атрибутов пользователей с использованием информации о сообществах; в) временная сложность; г) масштабируемость.

Теоретическая и практическая значимость

Теоретическая значимость работы заключается в следующем:

- предложены способы вычисления свойств используемых моделей случайных графов: вероятность ребра кратности ≥ 2 в случайном двудольном графе "пользователь-сообщество", а также средняя степень вершины в случайном социальном графе с сообществами;
- косвенно подтверждена гипотеза о значительном пересечении сообществ контактов индивидуального пользователя с сообществами социального графа, в которых состоит данный пользователь.

Разработанный в диссертационной работе метод EgoLP позволяет определять структуру сообществ пользователей в масштабе всей популяции социальной сети (сотни миллионов пользователей), обеспечивая при этом возможность решения практических задач, связанных с использованием знаний о сообществах. Одним из этапов предложенного метода является определение структуры сообществ среди непосредственных контактов каждого пользователя. Полученные сообщества могут использоваться пользователями в качестве замены ручной группировки контактов для оптимизации потоков информации на персональных страницах пользователей.

Кроме того, разработанный метод CKB для генерации случайных социальных графов с заданной структурой сообществ пользователей позволяет, в отличие от известных аналогичных методов:

- создавать в случайном социальном графе структуру сообществ, обладающую характерными свойствами сообществ пользователей реальных социальных сетей;

- исследовать качество методов определения структуры сообществ на графах из сотен миллионов вершин.

Таким образом, можно ожидать, что генерируемые с помощью предложенного метода тестовые данные будут применяться исследователями для оценки качества и усовершенствования методов определения структуры сообществ пользователей в социальных графах.

На основе предложенных методов были поданы заявки на патенты:

- заявка на патент P20140009930 “Fast and Distributed Detection for Overlapping Community”, подана в Республике Корея 27.01.2014 г.;
- заявка на патент 2014117945 “Способ и устройства для распределённой генерации случайных социальных графов со структурой пересекающихся сообществ пользователей”, подана в РФ 05.05.2014 г.

Апробация работы

Основные результаты диссертационной работы докладывались в рамках следующих мероприятий:

- сто шестьдесят третье (30 мая 2013 года) заседание Московской секции ACM SIGMOD (ВМК МГУ, г. Москва);
- 15-я Всероссийская научная конференция “Электронные библиотеки: перспективные методы и технологии, электронные коллекции” — RCDL’2013 (14-17 октября 2013 года, г. Ярославль);
- 5-й симпозиум по сложным сетям CompleNet-2014 (12-14 марта 2014 года, г. Болонья, Италия);
- 10-я Международная конференция “Интеллектуализация обработки информации-2014” (4-11 октября 2014 года, о. Крит, Греция);
- 4-й симпозиум по интеллектуальному анализу сетевых данных DaMNet-2014 (14 декабря 2014 года, г. Шэньчжень, КНР).

Кроме того, результаты работы обсуждались в рамках семинаров “Распределенные объектно-ориентированные системы” в Институте системного программирования РАН, а также на семинаре по анализу социальных сетей Института проблем управления РАН.

Диссертационная работа выполнена при поддержке гранта РФФИ №13-07-12134 офи_м “Исследование и разработка методов распределенной обработки больших баз графовых данных”.

Личный вклад

Все выносимые на защиту результаты получены лично автором. Программные реализации выполнены совместно с Кириллом Чихрадзе и Назаром Бузуном.

Публикации

Основные результаты по теме диссертации опубликованы в 8 печатных работах [22–29], из которых 2 статьи опубликованы в рецензируемых журналах, рекомендованных ВАК РФ [23, 24], 4 статьи включены в реферативную базу данных Scopus [22, 24, 25, 29].

В работах [25–27] автору принадлежат обзорные разделы и описание основных элементов разработанных методов. В статьях [28, 29] автором написаны обзорные разделы. В статье [23] автору принадлежит раздел, посвящённый генерации случайных социальных графов с сообществами пользователей, а также поиску сообществ пользователей. В работе [24] автору принадлежит раздел, посвящённый исследованию применимости эго-сообществ пользователей для решения задачи рекомендации пользователям получателей электронных сообщений.

Структура и объём диссертации

Диссертация состоит из введения, четырёх глав, заключения и двух приложений. Полный объем диссертации **134** страницы текста с **92** рисунками и **9** таблицами. Объём приложений составляет **30** страниц. Список литературы содержит **78** наименований.

Глава 1

Сообщества пользователей в социальном графе

1.1 Социальная сеть и социальный граф

Современные социальные сети являются важным инструментом компьютерно-опосредованной коммуникации, стремительно набирающим популярность пользователей по всему миру в течение последних двух десятилетий. В современном понимании *онлайновая социальная сеть* — это Интернет-сервис, позволяющий пользователям публиковать на своих страницах персональные и иные данные и служащий для упрощения коммуникации и обмена информацией между пользователями сети Интернет.

Можно выделить следующие группы социальных сетей в зависимости от основного предназначения:

- **сети общего назначения** (*Facebook*, *вКонтакте*, *Одноклассники*) — предназначены для поддержания существующих контактов из реального мира, обсуждения повседневных событий и развлечений;
- **нишевые сети** (*LinkedIn*, *Comon*) — предназначены для поддержания существующих и установления новых профессиональных контактов, а также для дискуссий по общим интересам и решения профессиональных вопросов;

- **контентные сети** (*Twitter*, *YouTube*, *Last.fm*) — предназначены для обмена контентом, распространения новостей, создания и развития сообществ по интересам и развлечений;

- **другие сети** — геосоциальные сервисы (*FourSquare*), сервисы вопросов (*StackOverflow*, *Quora*), социальные Интернет-закладки (*delicious*), онлайн-игры (*World of Warcraft*) и т.д.

Кроме того, появление средств эффективного взаимодействия между пользователями во многом определило развитие других сервисов, таких как системы рекомендаций и онлайн-энциклопедии.

Помимо коммуникационной функции, сервисы социальных сетей играют роль баз пользовательских данных, в которых с каждым пользователем ассоциирован набор персональной информации, составляющий его "виртуальную личность". Данные всех пользователей одного сервиса образуют его *социальный граф*, — динамическую структуру, полностью описывающую состояние и поведение составляющих её пользователей, а также их отношения между собой и объектами внешнего мира в некоторый момент времени.

С позиций анализа данных социальный граф представляет собой набор разнородной и слабоструктурированной пользовательской информации. Несмотря на то, что целевая аудитория и функционал социальной сети во многом определяют структуру и содержание конкретного социального графа, можно выделить следующие основные типы данных:

- *сетевые данные* (отношения связи между пользователями, а также между пользователями и объектами);

- *профили пользователей* (биография, взгляды, интересы пользователей);

- *текстовые данные* (сообщения, комментарии);

- *мультимедийные данные* (фото-, видео-, аудиоматериалы);

- *объекты уровня сети* (сообщества, приложения, таксономии контента);

- *объекты внешнего мира* (ссылки на ресурсы за пределами сети);

- *журналы активности пользователей* (записи о взаимодействии пользователей между собой и различными объектами).

В терминах сетевого анализа социальный граф можно представить в виде графа $G = (V, E)$, полностью описывающегося множеством V своих вершин (пользователи и объекты сети) и множеством E своих рёбер (взаимодействия между пользователями и объектами сети). Вершины и рёбра могут быть различных типов, а также обладать *атрибутами*: профили пользователей, метаданные объектов, временные метки, веса и др. Несмотря на многообразие семантики возможных отношений между пользователями, принято выделять 2 основных типа рёбер в социальных графах: *неориентированные* (соответствуют связям типа "дружба") и *ориентированные* (соответствуют связям типа "подписка"). Первый тип рёбер характерен для графов сетей общего назначения и нишевых сетей, тогда как ориентированные рёбра являются типичными для контентных сетей. Некоторые сети также допускают существование ребра любого из описанных типов между парой пользователей.

Кроме того, в связи с наличием различных типов вершин и рёбер принято различать подмножества множеств V и E, которые содержат соответственно вершины и рёбра одного типа. Например, в графе контентой сети для обмена фотографиями и классификации их с помощью категорий пользовательской таксономии (*тэгов*) множество вершин можно представить как $V = (U, P, T)$, где U, P и T — подмножества пользователей, фотографий и тегов соответственно. Соответственно, множество рёбер в таком графе будет состоять из связей типов "пользователь-фотография", "фотография-тэг"и "пользователь-тэг": $E = (UP, PT, UT)$.

Поскольку одно ребро социального графа может связывать более 2 вершин (например, взаимодействие типа *назначение тега фотографии* включает пользователя, фотографию и тег), то в терминах теории графов он является *гиперграфом*. Если каждое гиперребро представить в виде множества парных рёбер между k различных подмножеств вершин, то такой гиперграф может быть сведён к *k-дольному графу*.

Однако большинство методов анализа социальных сетей не предназначены для работы с гиперграфами и многодольными графами. Поэтому общепринятой является практика построения *упрощённого представления* социального графа, которое отражает отдельные аспекты более сложных взаимо-

действий оригинального графа. Как правило, такое представление содержит вершины одного-двух типов и не содержит гиперрёбер, что упрощает применение к ним различных методов анализа без существенного ухудшения качества результатов. В данной работе, в соответствии с наиболее распространённым подходом, вершинами графа считаются пользовательские аккаунты, а рёбрами — социальные связи типа “дружба”, которыми пары пользователей явно связывают свои аккаунты.

Принято выделять 3 уровня организации социальных графов: макро-, мезо- и микроскопический.

На *макроскопическом уровне* социальный граф является *безмасштабной сетью* (англ. *scale-free network*) с набором характерных свойств: степенное распределение степеней вершин с экспонентой в пределах $[1.5; 2]$, небольшой диаметр ($4-5$ для большинства сетей), типичная зависимость коэффициента кластеризации от степени вершины и т.д. В связи с резко неравномерным распределением степеней вершин для социального графа характерна довольно большая разреженность, что во многих случаях определяет выбор метода для решения тех или иных задач.

На *мезоскопическом уровне* социальный граф состоит из иерархии модулей (сообществ пользователей), структура которых часто связана с организационными и функциональными характеристиками социальной сети.

На *микроскопическом уровне* социальный граф представлен отдельными *учётными записями*, или *аккаунтами* пользователей, включающих атрибуты пользователей, их контент и связи между собой. Особенностью данного уровня является феномен *гомофилии* — повышенной вероятности образования рёбер и сообществ между пользователями с похожими свойствами и атрибутами.

1.2 Сообщества пользователей

Являясь по сути виртуальным отражением части человеческого социума, социальные сети во многих случаях наследуют характерную для него структуру социальных групп, или *сообществ*. Наличие сообществ пользователей является распространённым свойством современных онлайновых социальных сетей вне зависимости от состава аудитории, природы связей и преоблада-

ющих сценариев использования: общение, обмен контентом, поиск информации, развлечения и т.д. Например, часто можно встретить сообщества, в которых участников объединяют общие интересы, политические и религиозные предпочтения, географическая близость и т.д.

Рисунок 1.1: Визуализация социального графа Facebook в виде множества городов. Позиции городов соответствуют их географическим координатам, вес рёбер между городами соответствует количеству социальных связей между пользователями из этих городов. Рёбра с минимальным весом окрашены чёрным, с максимальным весом — белым.

На рисунке 1.1 изображёны связи между городами, в которых живут пользователи Facebook. Обращает на себя внимание наличие участков концентрации рёбер с большим весом, которые во многих случаях соответствуют странам и отдельным регионам. Следовательно, социальный граф Facebook можно рассматривать как совокупность модулей, соответствующих сообществам жителей стран, регионов, городов и т.д.

Таким образом, с точки зрения сетевого анализа сообщества представляют собой *кластеры* пользователей, связанных между собой сильнее, чем с другими пользователями сети. С функциональной точки зрения *сообщество* — это группа пользователей, выполняющая общую роль или функцию и обладающая общими свойствами, ценностями и целями.

С точки зрения интерфейса социальных сервисов возможно 2 способа объединения пользователей в сообщества. В первом случае пользователи явно указывают своё членство в группе путём вступления в неё (рисунок 1.2). Во втором случае связь с некоторой группой устанавливается неявно путём

образования социальных связей, основанных на общей роли, деятельности, круге общения, интересах, функциях или каких-либо других свойствах. В этом случае для определения структуры сообществ пользователей требуются специализированные методы (раздел 2.1). Очевидно, однако, что некоторые из алгоритмически найденных сообществ пользователей будут частично или полностью совпадать с составом существующих внутрисетевых групп.

Подобный механизм формирования социальных групп отчасти объясняется *теорией общей идентичности и общей привязанности* (англ. *common identity & common bond theory*) [30], которая утверждает, что люди присоединяются к группам либо на основании интереса к обсуждаемым темам, либо в силу социальных отношений с другими участниками группы.

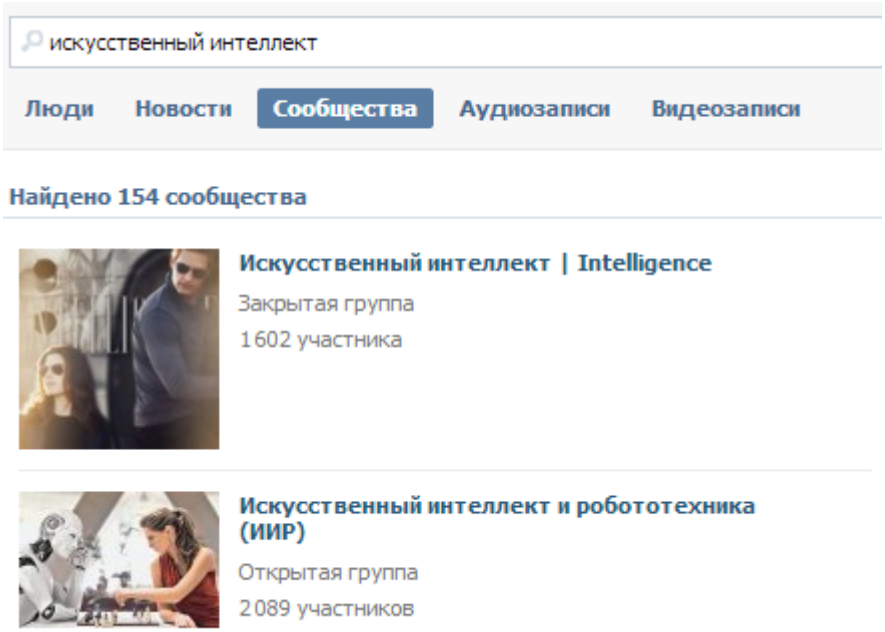


Рисунок 1.2: Примеры сообществ по интересам.

Вместе с тем, возможны ситуации, когда пользователь вступает в группу по интересам из-за личного знакомства с другими её участниками. Аналогично, пользователи могут устанавливать между собой социальные связи на основании участия в общих группах.

1.3 Структурные свойства сообществ

Ввиду отсутствия общепринятой формализации приведённого определения сообщества, важно выделить некоторые особенности сообществ пользо-

вателей, характерные для социального графа на уровне связей между пользователями. Формализация задачи и метода поиска сообществ на уровне сетевых данных позволяет использовать общепринятую терминологию теории графов и соответствующие инструменты.

Отметим, что в данной работе рассматриваются только неориентированные социальные графы со связями типа “дружба”. Это связано с предположением о том, что тип социальных связей оказывает существенное влияние как на свойства ссылочной структуры социального графа, так и на свойства пользовательских сообществ. Несмотря на это, предложенные в разделах 3 и 4 методы допускают расширения для случая ориентированных графов, что обсуждается в завершении разделов.

Рассмотрим неориентированный граф $G(V, E)$, где $|V| = n$ и $|E| = m$. Пусть V' —- некоторое подмножество V и пусть E' —- подмножество всех ребер графа G, концевые вершины которых входят в V'. Тогда граф $G' = (V', E')$ называется *вершинно-порождённым подграфом* графа G. В случае социального графа V' соответствует группе пользователей, а E' — всем связям между ними в социальном графе.

Сообществом пользователей $Z_c(V_c, E_c)$ будем называть любой вершинно-порождённый подграф социального графа $G(V, E)$.

Покрытием $\mathbb{C}$ социального графа $G(V, E)$ будем называть множество сообществ пользователей, заданных для G: $\mathbb{C} = \{Z_c\}_{c=1}^{K}$, причём $\forall c : V_c \subseteq V$, $E_c \subseteq E$.

Рассмотрим двудольный граф $B(V, \mathbb{C}, M)$, где V соответствует множеству вершин социального графа G, $\mathbb{C}$ — покрытие, а ребро $(u, c) \in M$ соединяет вершину $u \in V$ с сообществом $Z_c \in \mathbb{C}$, если $u \in V_c$. Степень m_u вершины u равна количеству сообществ, в которых состоит пользователь u. Степень $n_c = |V_c|$ вершины Z_c называется *размером сообщества* и равна количеству пользователей, которые состоят в сообществе Z_c.

Отметим, что целесообразно ограничить размер сообщества снизу 3 вершинами, $\forall c : n_c \geq 3$, поскольку идентификация и анализ групп пользователей меньшего размера не требует дополнительных методов анализа структуры сети. Кроме того, некоторые пользователи могут не состоять ни в одном сообществе, $\forall u : m_u \geq 0$.

Распространённым подходом к исследованию структурных свойств сообществ длительное время являлся анализ сообществ, найденных различными методами кластеризации графов и специализированными методами определения структуры сообществ. Исследователи предлагали различные классификации сообществ на основании структурных различий, а также делали выводы о характерных паттернах связи вершин друг с другом и с сообществами [7, 31].

В качестве альтернативного подхода Mislove et al [32] и Yang et al [9] предложили исследовать свойства так называемых *функциональных сообществ*, то есть пользовательских групп с добровольным членством. Многие сервисы социальных сетей позволяют пользователям создавать и вступать в особые группы, в которых они могут общаться и обмениваться контентом (рисунок 1.2). Эти группы обычно создаются на основе специфических тем, интересов, хобби и географического положения. Например, в LiveJournal группы категоризованы по следующим типам: культура, развлечения, игры, спорт, студенческая жизнь, технологии и др. Некоторые группы являются модерируемыми, то есть вступление в группу и размещение в ней контента контролируются выбранным или назначенным пользователем-модератором. Другие группы полностью открыты и позволяют любому пользователю присоединяться к ним и размещать информацию.

С целью исследования структурных свойств таких групп исследователи произвели сбор данных о связях между пользователями реальных социальных сетей, а также о членстве пользователей в функциональных сообществах. Yang et al опубликовали собранные данные о социальных сетях Friendster, Orkut, LiveJournal и YouTube[1].

Характеристики полученных наборов данных приведены в таблице 1.1. Для Friendster и LiveJournal полученные семплы социальных графов соответствуют $> 90\%$ популяции их пользователей на момент сбора. Так как члены групп могут быть отделены от остальной части сети, каждая компонента связности группы считается отдельным сообществом.

Отметим, что среди опубликованных наборов данных только в сетях Orkut и Friendster связи между пользователями являются неориентированными. Однако для согласованного представления всех сетей Yang et al счи-

[1] http://snap.stanford.edu/data/index.html#communities

тали каждую сеть невзвешенным неориентированным статичным графом. Таким образом, для LiveJournal и YouTube ориентация рёбер игнорировалась. Несмотря на некорректность такого подхода, некоторые из нижеперечисленных структурных свойств сообществ нашли подтверждение во всех наборах данных.

Таблица 1.1: Свойства графов социальных сетей LiveJournal, ORKUT и YouTube с известной структурой сообществ пользователей.

Социальная сеть	**Friendster**	**Orkut**	**LiveJournal**	**YouTube**
Количество вершин	65,608,366	3,072,441	3,997,962	1,134,890
Средняя степень	55.05	76.2	17.3	5.3
Количество сообществ	1,449,666	8,455,253	311,782	8385
Экспонента степенного распределения размеров сообществ	2.11	2.12	2.14	2.36
Экспонента степенного распределения принадлежности пользователей к сообществам	2.44	1.59	2.22	2.83
Экспонента степенного распределения степеней вершин	1.42	1.58	2.15	2.53
Медиана распределения размеров сообществ	2	16	2	3
Медиана распределения количества сообществ, в которых состоит пользователь	2	14	2	1
Средний коэффициент кластеризации	0.1623	0.169	0.353	0.172

Авторы исследований выявили следующие свойства социальных графов с сообществами:

1. Групповые свойства сообществ:
   - множества вершин сообществ могут пересекаться [9, 33];
   - распределение размеров сообществ подчиняется степенному закону [8];
2. Связь пользователей и сообществ:
   - распределение количества сообществ, в которых состоит пользователь, подчиняется степенному закону [8];
   - количество сообществ у пользователя прямо пропорционально количеству его связей с другими пользователями [32];
3. Связи между пользователями:

- вероятность ребра между парой вершин увеличивается с ростом количества общих сообществ, которым принадлежат обе вершины [8];
- для пары сообществ пересечение их более плотно, чем непересекающаяся часть [8];
- количество рёбер в сообществе растёт суперлинейно с размером сообщества [8];
- *связующие вершины* (англ. *connector nodes*) сообщества (имеющие среди всех вершин сообщества наибольшее количество связей с другими вершинами из этого сообщества) более вероятно находятся в пересечениях с другими сообществами, чем в непересекающейся области сообщества [8];
- средний коэффициент кластеризации вершин в сообществе (раздел 1.4) обратно пропорционален размеру сообщества [32].

Дискуссии о причинах и следствиях каждого из свойств, а также их экспериментальные подтверждения приведены в цитируемых работах. Здесь рассмотрим только самое важное из сделанных открытий, которое противоречило общепринятым представлениям: вероятность ребра между парой вершин увеличивается с ростом количества общих сообществ, которым принадлежат обе вершины. Таким образом, в пересечениях сообществ плотность рёбер между пользователями выше, чем между пользователями в непересекающихся их частях.

Yang et al объясняют данный феномен с помощью нового взгляда на *гомофилию* — склонность индивидуумов устанавливать связи с людьми, похожими на них [34, 35]. Авторы исследования утверждают, что современные взгляды на сетевые сообщества основаны на двух фундаментальных теориях, объясняющих структуру сложных сетей: *триадное замыкание* [36] и *сила слабых связей* [37]. Как следствие, механизм гомофилии реализуется через склонность к созданию связей между отдельными личностями, похожими по одному из социально значимых атрибутов: социальный статус, пол, возраст, нация и т.д. Это приводит к образованию сообществ в виде т.н. “карманов гомофилии”, которые либо полностью изолированы (рисунок 1.3, a), либо имеют небольшое пересечение с низкой плотностью рёбер (рисунок 1.3, b).

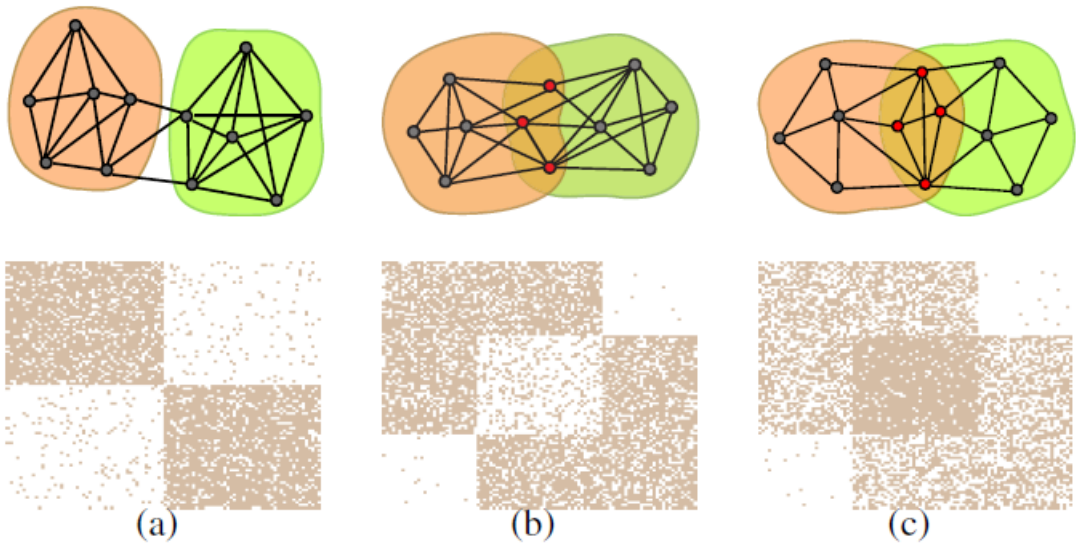


Рисунок 1.3: Различные представления о пересечениях сообществ: (a) - сообщества не пересекаются, (b) - плотность рёбер в пересечении меньше, (c) - плотность рёбер в пересечении больше, чем в непересекающихся частях.

В качестве причины значительных по размеру пересечений сообществ и повышенной плотности рёбер в них (рисунок 1.3, c) Yang et al вводят понятие *плюралистической гомофилии* (англ. *pluralistic homophily*). Согласно этой теории образования социальных связей, близость пользователей оценивается не количеством похожих атрибутов отдельных личностей, а количеством социальных групп, к которым принадлежат оба пользователя. Таким образом, механизм плюралистической гомофилии объясняет образование сообществ не в виде “карманов” с большей концентрацией социальных связей, а в виде множества пересекающихся социальных групп различной природы, в котором области с наибольшим количеством пересечений естественным образом содержат больше связей между пользователями.

Данные представления согласуются с идеями социологов Зиммеля [38] и Фельда [39] о том, что существуют внешние по отношению к социальной сети факторы, которые служат определяющими причинами образования связей. Фельд называет такие факторы *фокусами* и утверждает, что личности, чья деятельность связана с одним и тем же фокусом (человеком, местом, социальной позицией и др.), имеют повышенную вероятность образования связи между собой. Вместе с тем, отмечается, что связи могут возникать и “случайно”, то есть безотносительно общих фокусов.

В заключение отметим, что данные социологические теории, частично подтверждённые в недавних исследованиях пользовательских групп онлайновых социальных сетей, нашли отражение в предложенных в диссертационной работе методах исследования структуры сообществ. В предложенном методе генерации случайных социальных графов с сообществами связи между пользователями образуются прежде всего за счёт общих сообществ, однако существует и вероятность возникновения связи безотносительно сообществ. В предложенном методе определения структуры сообществ сначала идентифицируются сообщества непосредственных контактов каждой вершины как части неизвестных фокусов анализируемой сети, которые затем особым образом объединяются в глобальные сообщества пользователей.

1.4 Метрики качества сообществ

Интуитивная ассоциация сообщества пользователей с кластером вершин социального графа приводит к естественному предположению о том, что вершины “хорошего” сообщества компактно связаны между собой и одновременно хорошо отделены от остальных вершин графа. На этих представлениях основано большинство методов определения структуры сообществ (раздел 2.1), многие из которых предлагают различные способы формализации описанных свойств.

Yang et al [9] предложили аксиоматический подход к оценке соответствия произвольного подграфа интуитивным свойствам сообщества пользователей. Вводится набор *метрик качества сообществ* (англ. *community goodness metrics*), каждая из которых оценивает некоторое желаемое свойство отдельного сообщества.

Рассмотрим граф $G(V, E)$ и сообщество $Z_c(V_c, E_c)$. Обозначим через s_c величину разреза $C(V_c, \overline{V_c})$, $V_c \cup \overline{V_c} = V$:

$$s_c = |\{(u, v) \in E | u \in V_c, v \notin V_c\}|. \tag{1.1}$$

Проводимость $\phi(V_c)$ разреза $C(V_c, \overline{V_c})$ определяется как

$$\phi(V_c) = \frac{s_c}{\min(Vol(V_c), Vol(\overline{V_c}))}, \tag{1.2}$$

при этом

$$Vol(V_c) = \sum_{i \in V_c} d_i, \tag{1.3}$$

где d_i — степень вершины i.

Yang et al предложили следующий набор метрик качества сообществ:

1. *Отделимость* (англ. *separability*) соответствует представлению о том, что сообщество хорошо отделено от остальной части сети: $\frac{|E_c|}{s_c}$;

2. *Плотность* (англ. *density*) соответствует представлению о хорошей внутренней связности сообществ: $\frac{2|E_c|}{|V_c|(|V_c|-1)}$;

3. *Сплочённость* (англ. *cohesiveness*) характеризует внутреннюю структуру сообщества и соответствует представлению о том, что вершины сообщества должны быть связаны равномерно. Таким образом, разделить сообщества на несколько подсообществ должно быть довольно сложно. Для оценки сплочённости сообщества используется проводимость его возможных внутренних разрезов [11], которая является широко распространённой мерой оценки внутренней связности подграфа: $min_{V'_c \subset V_c} \phi(V'_c)$. Таким образом, хорошо сплочённое сообщество должно обладать большим значением минимальной проводимости среди всех возможных внутренних разрезов, поскольку для разделения сообщества на компоненты связности требуется удаление большого количества рёбер;

4. *Средний коэффициент кластеризации* $\overline{C}_c$ вершин сообщества $Z_c(V_c, E_c)$ характеризует повышенную вероятность возникновения ребра между участниками, которые имеют общих соседей в данном сообществе:

$$\overline{C}_c = \frac{1}{|V_c|} \sum_{i=1}^{|V_c|} \frac{2|\{e_{jk} : v_j, v_k \in N_i, e_{jk} \in E_c\}|}{|N_i|(|N_i| - 1)}, \tag{1.4}$$

   где $N_i \subseteq V_c$ соответствует множеству соседей вершины i в сообществе Z_c.

При этом авторы исследования отмечают, что подграф с большим значением какой-либо метрики качества не обязательно соответствует сообществу.

Однако сообщество, найденное алгоритмически, должно иметь высокие показатели по одной или более метрик качества. В частности, при сравнении явно заданных функциональных сообществ (раздел 1.3) со случайно выбранными подграфами исходного социального графа исследователи показывают, что сообщества характеризуются устойчиво более высокими значениями предложенных метрик.

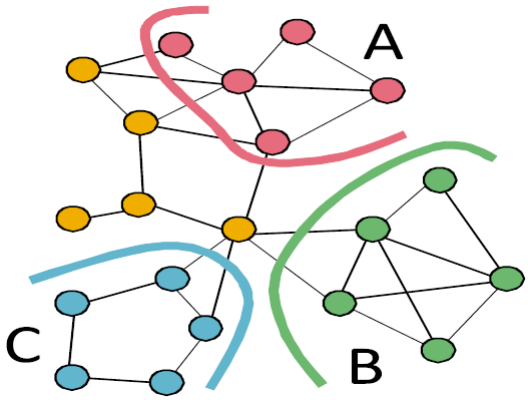


Рисунок 1.4: Сообщества пользователей как кластеры вершин социального графа.

На рисунке 1.4 изображены 3 подграфа, которые интуитивно идентифицируются как кластеры вершин представленного социального графа, которые могут соответствовать сообществам пользователей. Подграф A имеет 6 внутренних и 5 внешних рёбер, в силу чего обладает более низкой отделимостью по сравнению с другими кластерами. В подграфе B представлены 8 из 10 возможных рёбер между его вершинами, что говорит о его высокой плотности по сравнению с подграфом A и особенно подграфом C. При этом в подграфе C достаточно удалить 2 ребра, чтобы разделить его на компоненты связности, что говорит о его низкой сплочённости. Данный подграф также обладает самым низким коэффициентом кластеризации, поскольку ни одна тройка его вершин не образует 3-клику. Следовательно, подграфы C и B по совокупности значений метрик качества являются “худшим” и “лучшим” сообществами соответственно.

Таким образом, рассмотренные метрики качества позволяют оценить различные — зачастую противоречивые — аспекты сетевых сообществ.

1.5 Выводы

В данном разделе были рассмотрены основные причины и принципы формирования сообществ пользователей в социальных сетях, а так же структурные свойства сообществ как вершинно-порождённых подграфов социального графа. В заключение рассмотрены метрики качества сообществ, позволяющие оценивать соответствие алгоритмически найденных сообществ интуитивным представлениям о сообществах пользователей социального графа.

Глава 2

Определение структуры сообществ пользователей

Пользователи онлайновых социальных сетей не всегда указывают полный список сообществ, к которым они принадлежат. С одной стороны, поддержание актуального списка сообществ, к которым относит себя пользователь, требует значительного времени. С другой стороны, из соображений приватности пользователи часто не указывают некоторые типы сообществ осознанно: например, сообщества, связанные с политическими, религиозными или сексуальными предпочтениями. Наконец, пользователь может не знать о наличии некоторых сообществ (коллеги, одноклассники, соседи), которые формируются и развиваются неявно на макроскопическом уровне организации социальной сети.

При этом данные социального графа содержат достаточно информации для определения структуры как публичных, так и неявных сообществ пользователей. В последние годы было предложено множество методов для определения в графе социальной сети скрытой структуры сообществ, к которым принадлежат пользователи [1–4]. Большинство методов использует для анализа только социальные связи между пользователями, поскольку они являются неотъемлемым элементом любой социальной сети. Таким образом, задачу определения структуры сообществ в социальном графе можно свести к задаче поиска особых кластеров пользователей на основе анализа социальных связей между ними. С другой стороны, данную задачу можно рассматри-

вать как разновидность категоризации пользователей в одну или несколько заранее неизвестных групп.

В данной главе приводится краткий обзор современных методов определения структуры сообществ пользователей в социальном графе, а также критериев оценки качества таких методов.

Основные результаты главы опубликованы в работах [22, 23, 28, 29].

2.1 Методы определения структуры сообществ

По результатам исследования в рамках данной работы было выделено три основных класса алгоритмов, которые обеспечивают хорошее качество определения структуры сообществ со значительным пересечением множеств вершин [2]. Эти классы включают методы, основанные на вероятностных моделях, методы локальной оптимизации и методы распространения меток.

Ниже дано краткое описание методов, основанных на вероятностных моделях и локальной оптимизации, а также рассмотрены проблемы их масштабируемой реализации. После этого рассмотрен класс методов распространения меток, включающий алгоритмы, допускающие простую и эффективную распределённую реализацию. Отдельно рассмотрен класс методов, использующих знания о структуре эго-сообществ отдельных узлов для определения структуры глобальных сообществ, отмечены отличия известных методов от стратегии использования эго-сообществ, применяемой в предложенном методе `EgoLP` (глава 4). Наконец, перечислены известные методы определения структуры сообществ, имеющие распределённую масштабируемую реализацию.

2.1.1 Локальная оптимизация

Алгоритмы из этого класса оптимизируют некоторую локальную функцию качества для каждого сообщества независимо. При этом изменяется состав вершин сообщества, чаще всего путём добавления в расширяющееся “ядро” будущего сообщества смежных с ним вершин. Например, функция качества

сообщества в методе GCE [40] задана следующим образом:

$$F_S = \frac{k_{in}^S}{(k_{in}^S + k_{out}^S)^\alpha}, \tag{2.1}$$

где S — индуцированный подграф G (растущее "ядро" сообщества Z_c), k_{in}^S — удвоенное число рёбер, которые имеют начало и конец в S, k_{out}^S — число рёбер с одной вершиной в S, α — константа, параметр метода.

Другой популярный представитель этого класса – метод OSLOM [14], который формирует сообщество, сравнивая его с подграфом вершин сообщества с теми же степенями в случайном графе.

Создание распределённых реализаций таких методов, с одной стороны, упрощается за счёт независимой обработки каждого сообщества. С другой стороны, однако, большинство методов требуют дополнительной предили постобработки сообществ или исходного графа. Как правило, такие процедуры имеют значительную вычислительную сложность: поиск случайных окрестностей вершин [14], поиск максимальных клик [40] и других плотных подграфов [41], объединения части значительно пересекающихся сообществ и др. Кроме того, обработка больших графов данными методами чувствительна к лимиту памяти, выделенной для одного сообщества и его окрестности.

2.1.2 Вероятностные модели

Каждый алгоритм в этом классе работает с параметрическим распределением на множестве графов с фиксированным количеством узлов: MOSES [42], BCD [43], BIGCLAM [12], CESNA [44] и др. Параметр распределения включает матрицу инцидентности вершина-сообщество. В процессе оптимизации параметров модели выполняется поиск матрицы, максимизирующей правдоподобие конфигурации рёбер исходного графа.

Распределённая оптимизация параметров модели для графов большого размера затрудняется значительной временной сложностью используемых методов оптимизации (например, методы Монте Карло по схеме марковской цепи, мультистартовая оптимизация), а также наличием зависимых переменных, которые обновляются синхронно. Кроме того, передача общих парамет-

ров функции правдоподобия между узлами вычислительного кластера может привести к значительной нагрузке на сетевую инфраструктуру.

2.1.3 Распространение меток

Использование различных схем *распространения меток* (англ. *label propagation, LP*) по рёбрам графа является популярным направлением среди современных методов обнаружения сообществ. Простые и интуитивно понятные методы из данного класса сочетают приемлемую точность обнаружения сообществ, низкую вычислительную сложность и хорошую масштабируемость, а также простоту реализации с точки зрения современных вычислительных парадигм распределенной обработки графов *Pregel* [45] (раздел 4.2) и *GraphX* [46].

Основной особенностью методов данного класса является итеративный процесс обмена метками сообществ между узлами графа, которые накапливают поступающие к ним метки и отправляют сообщения с наборами меток соседним узлам, чтобы сообщить об обновленной коллекции меток. В большинстве методов вершины графа отправляют и получают метки одновременно и независимо друг от друга, что упрощает создание параллельной реализации алгоритма.

Алгоритм 1 описывает общий шаблон алгоритма распространения меток. Большинство популярных методов из данного класса (в том числе LPA [19], SLPA [20], BMLPA [41] и COPRA [47]) соответствуют этому шаблону и отличаются только способом инициализации вершин метками сообществ и стратегиями обмена метками.

Среди описанных методов SLPA (*Speaker-listener Label Propagation Algorithm*) обладает оптимальным сочетанием низкой вычислительной сложности и возможностью эффективной распределённой реализации. Вначале каждая вершина инициализируется уникальной меткой сообщества. На каждой итерации каждый узел поочерёдно выполняет роль “говорящего” и “слушающего узла” согласно простейшим стратегиям взаимодействия узлов:

- СтратегияОтправителя(v,n): случайным образом выбрать k меток из множества меток вершины v с вероятностью, пропорциональной частоте меток, затем отправить выбранные метки вершине n;

Исходные данные: Граф $G(V, E)$
Результат: Покрытие сообществ $\mathbb{C} = \{Z_c(V_c, E_c)\},\ V_c \subseteq V$

для каждого $v \in V$ **выполнить**
  инициализировать v меткой сообщества;
конец цикла
для каждого $i = 1{:}T$ **выполнить**
  для каждого $v \in V$ **выполнить**
    для каждого $n \in V$ **выполнить**
      СтратегияОтправителя(v,n);
    конец цикла
  конец цикла
  для каждого $v \in V$ **выполнить**
    СтратегияПолучателя(v);
  конец цикла
конец цикла
для каждого $v \in V$ **выполнить**
  удалить из множества меток v все метки с частотой $< r$;
конец цикла
преобразовать метки всех вершин в $\{Z_c\}$.

Алгоритм 1: Шаблон алгоритма распространения меток

- СтратегияПолучателя(v): выбрать k наиболее частых меток из множества меток, полученных вершиной v от соседей, затем добавить выбранные метки к множеству меток вершины v.

В оригинальном методе SLPA $k = 1$, однако в данной работе была исследована модификация SLPA с варьируемым значением k. Согласно экспериментальным данным, оптимальное качество результатов достигается при $k \in [3; 10]$. Далее исследуемый метод обозначается как SLPA(T,r,k).

В рамках диссертационной работы были выявлены также следующие недостатки SLPA(T,r,k), которые обуславливают ухудшение качества результатов при увеличении степени пересечения сообществ в исследуемом графе:

1) в случае наличия в графе вершин со степенью, существенно (>3 раз) превышающей среднюю степень вершины в графе, их метки активно распространяются, что ведёт к формированию очень больших сообществ (по сравнению с известными заранее сообществами);

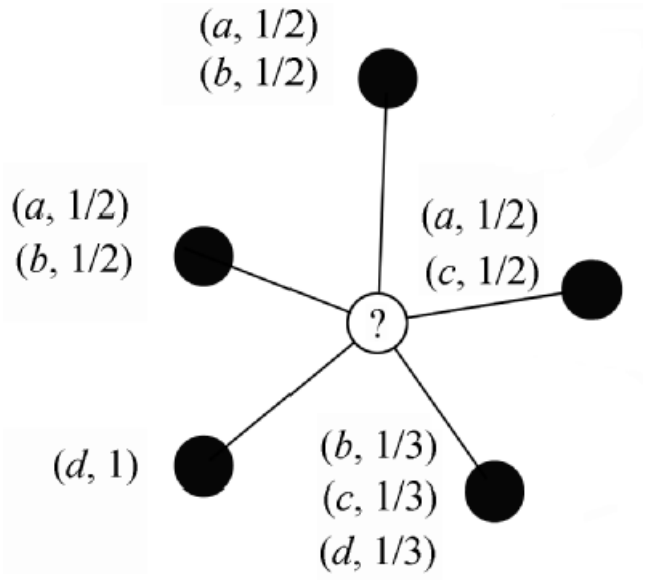


Рисунок 2.1: Обмен метками сообществ. “Говорящие” узлы окрашены чёрным, “слушающий” узел окрашен белым. На каждой итерации алгоритма каждый “слушающий” узел выбирает lr наиболее популярных из всех полученных меток окружающих его “говорящих” узлов.

2) недостаточное количество найденных сообществ по сравнению с заранее известными: с увеличением количества итераций количество сообществ в графе уменьшается, предположительно из-за доминирования одной или нескольких популярных меток сообществ среди меток большинства вершин;

3) наличие сообществ с низкой связностью между вершинами, включая несвязные сообщества;

4) если инициализировать вершины в соответствии с заранее известными сообществами (т.е. заранее задать корректное покрытие графа сообществами), то в процессе обмена метками вершины добавляют в свои множества меток много “шумовых” меток, что ведёт к ухудшению качества результатов с увеличением числа итераций.

Предложенный в главе 3 метод определения структуры сообществ пользователей также основан на распространении меток сообществ и включает шаги, направленные на устранение перечисленных недостатков SLPA(T,r,k).

2.1.4 Методы, основанные на эго-сообществах

В некоторых задачах социального анализа в качестве входных данных используется не весь социальный граф, а так называемая *эго-сеть* $Ego_v(v, V_v^{ego}, E_v^{ego})$, которая состоит из центрального пользователя v, всех его соседей V_v^{ego} и всех рёбер E_v^{ego} между участниками этой сети. Сообщества непосредственных контактов центрального пользователя v называются *эго-сообществами* $\{\mathcal{E}_{vk}(V_{vk}, E_{vk})\}$, $V_{vk} \subseteq V_v^{ego}$.

Использование знаний об эго-сообществах отдельных вершин для определения глобальной структуры сообществ в социальном графе является относительно новым подходом в области кластерного анализа сложных сетей.

Soundarajan et al. [48] предложили *Node Perception* — шаблон алгоритмов для объединения эго-сообществ в глобальные сообщества большего размера. Авторы сначала находят эго-сообщества в эго-сети каждого пользователя, а затем строят новую мета-сеть, вершинами которой являются найденные эго-сообщества. Наконец, глобальные сообщества находятся путём кластеризации полученной мета-сети. Алгоритм *DEMON*, предложенный Coscia et al. [49], является примером реализации шаблона *Node Perception* с одним отличием: эго-сообщества извлекаются не из всей эго-сети, а из эго-сети без центрального узла и всех его рёбер. Такая модификация позволяет избежать “шума”, вносимого центральным узлом, связанным со всеми остальными узлами эго-сети. Алгоритм *EgoClustering*, предложенный Rees et al. [50], похож на метод *DEMON*, однако эго-сообществами считаются все компоненты связности эго-сети, образующиеся после удаления центрального узла со всеми его рёбрами.

В более поздней работе Rees et al. [51] предложен другой способ агрегации эго-сообществ. Каждому из найденных эго-сообществ назначается уникальный идентификатор. Узлы каждого эго-сообщества начинают “обсуждать”, какой выбрать общий идентификатор. Для этого они итеративно обмениваются своими идентификаторами до тех пор, пока выбранные значения во всех эго-сообществах не перестанут меняться. Идентификатор для каждого эго-сообщества на каждой итерации выбирается как наименьшее значение из идентификаторов всех его узлов. При этом некоторые узлы с самого начала работы алгоритма не обмениваются метками.

В отличие от предыдущих подходов, предложенный в данной работе метод EgoLP использует эго-сообщества для агрегации и фильтрации меток сообществ, распространяющихся между узлами. Вместо того, чтобы взаимодействовать с каждым из соседей, узел взаимодействует со своими эго-сообществами, через которые к нему поступают сообщения от соседних узлов в графе. В результате самые популярные метки сообществ внутри одного эго-сообщества получают более высокие шансы быть принятыми и переданными далее. Результаты экспериментального сравнения методов SLPA и EgoLP в разделе 4.6 подтверждают эффективность такого подхода.

2.1.5 Масштабируемые методы

Определение структуры сообществ является вычислительно сложной и во многих случаях плохо масштабируемой задачей [2]. Вместе с тем, масштабируемость стала важным и крайне желательным свойством методов обнаружения сообществ в связи с ростом популяции пользователей социальных сетей. К примеру, в марте 2015 года социальная сеть Facebook сообщает об 1,39 миллиарде[1], а Twitter — о 288 миллионах пользователей[2], которые совершают какие-либо действия в сети хотя бы 1 раз в месяц.

Известные автору масштабируемые реализации включают:

- метод LPA [19] на основе фреймворка Hadoop MapReduce[3];
- метод SLPA [20] на основе программного интерфейса MPI (Message Passing Interface) [52];
- метод Louvian [18] на основе фреймворка Apache Giraph[4];
- метод *propinquity dynamics* [53] на основе фреймворка Hadoop MapReduce;
- метод спектральной кластеризации [54] на основе фреймворка Hadoop MapReduce;
- метод *shingling* [55] на основе фреймворка Hadoop MapReduce;

[1] http://newsroom.fb.com/company-info/
[2] https://about.twitter.com/company
[3] http://www.akshaybhat.com/LPMR/
[4] http://sotera.github.io/distributed-louvain-modularity/

- метод Scalable Community Detection [17].

Данные методы относятся к различным классам, описание которых лежит за пределами необходимого для данной работы обзора. Однако ни одна из перечисленных реализаций не обладает следующей комбинацией свойств, которые требуются для метода определения структуры сообществ пользователей социальной сети в масштабе всей её популяции:

1) высокое качество работы на социальных графах вне зависимости от количества сообществ у пользователя;

2) возможность функционирования на кластере из потребительских компьютеров с поддержкой автоматической балансировки нагрузки и устойчивостью к отказам отдельных узлов;

3) временная сложность, не превышающая линейную по количеству рёбер;

4) близкая к линейной масштабируемость;

5) возможность обрабатывать графы из сотен миллионов вершин со средней степенью >100, характерной для социальных графов.

2.2 Критерии оценки качества

Оценивание качества категоризации вершин социального графа в сообщества является нетривиальной проблемой. Стандартные критерии качества кластеризации не очень хорошо подходят для оценивания “мягкой” кластеризации вершин графа.

Общепринятые способы оценки включают использование шаблонных сетей как источников эталонной структуры сообществ, а также оценку качества приложений, использующих информацию о найденных в графе сообществах.

2.2.1 Качество восстановления эталонных покрытий

Современные исследователи в области идентификации социальных сообществ широко применяют социальные графы с эталонной структурой сооб-

ществ для оценки методов определения структуры сообществ. Такие *шаблонные сети* (англ. *benchmark networks*) состоят из набора вершин и рёбер социального графа (пользователи и связи между ними), а также списка сообществ, в которых состоит каждый пользователь. Графы и соответствующие им покрытия могут быть как синтезированы, так и получены из данных реальных социальных сервисов.

В качестве непосредственного критерия качества методов определения структуры сообществ принято использовать некоторую оценку близости двух покрытий для некоторого графа: найденного алгоритмом и *референтного*, то есть заранее заданного или известного. Такой подход позволяет исследовать способность различных методов восстанавливать структуру сообществ, заданную особым способом, зависящим от конкретного приложения или исследовательской задачи. Способы сравнения покрытий перечислены в конце текущего раздела.

Наиболее известным хранилищем данных социальных сетей со структурой сообществ является Stanford Large Network Dataset Collection[5]. Для создания этих наборов данных авторы исследования Янг и Лесковец [9] использовали различные социальные сети (LiveJournal, Friendster, Orkut, YouTube, а также 225 различных социальных сетей на платформе Ning), где пользователи создают явные группы для общения и обмена контентом (раздел 1.3).

При этом сбор тестовых данных путём семплирования и разметки реальных социальных сетей является трудоёмким, а свойства полученных наборов данных часто далеки от желаемых. Поэтому принято использовать программные средства для генерации случайных социальных графов со структурой сообществ пользователей, заданной в соответствии с некоторой моделью.

Отметим, что в теории любая генеративная модель графа с кластерами (например, из методов в разделе 2.1.2) при наличии известных значений параметров может быть использована для генерации шаблонных сетей как случайных социальных графов с сообществами в соответствии с заданным генеративным процессом.

[5] http://snap.stanford.edu/data/index.html#communities

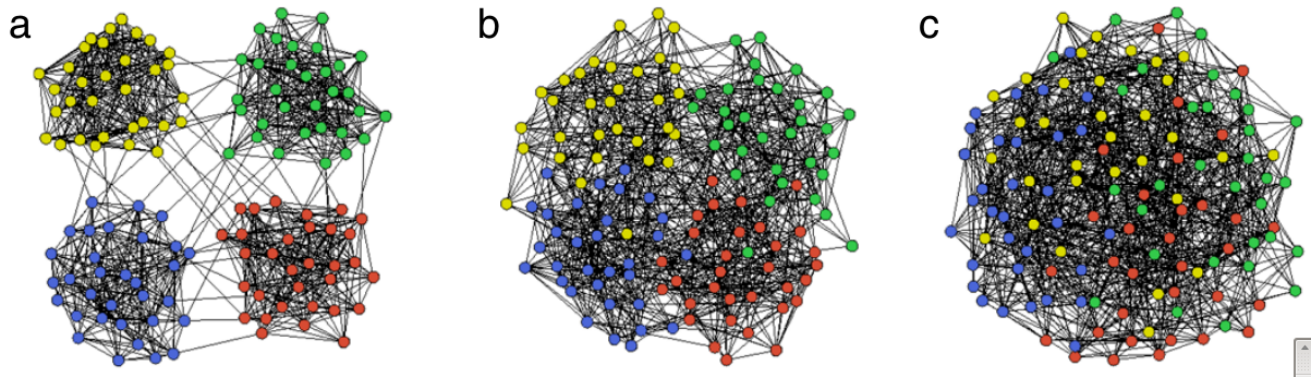


Рисунок 2.2: Шаблонные сети Гирвана-Ньюмена с различными значениями ожидаемой средней внутренней степени: $z_{in} = 15$ (a), $z_{in} = 11$ (b) и $z_{in} = 8$ (c). Можно видеть (c), что при небольшой ожидаемой внутренней степени группы становятся трудно различимы.

Тем не менее, исследователями был предложен ряд специализированных моделей случайных графов, которые могут быть использованы для генерации шаблонных сетей.

Модель Гирвана-Ньюмена (GN) [21]. В графе, созданном в соответствии с этой моделью (рисунок 2.2), вершины разбиты на l эквиразмерных групп по g вершин в каждой. Вершины из одинаковых групп соединяются ребром с вероятностью p_{in}, а вершины из разных групп — с вероятностью p_{out}. Внутри каждой группы рёбра генерируются в соответствии с моделью Эрдёша-Реньи [56] для генерации случайного графа с вероятностью связи p_{in}. Ожидаемые внутренняя и внешняя средние степени графа равны соответственно $z_{in} = p_{in}(g-1)$ и $z_{out} = p_{out}g(l-1)$. Средняя степень вершины во всём графе $\langle k \rangle = z_{in} + z_{out}$.

Также существуют и другие варианты данной модели: с сообществами разного размера [57], с пересекающимися сообществами [58], иерархическая [59] и взвешенная версии [60].

Таким образом, модель Гирвана-Ньюмена создаёт взаимно связанные между собой случайные графы модели Эрдёша-Реньи. Поэтому все вершины имеют примерно одну и ту же степень. Кроме того, все сообщества по умолчанию создаются эквиразмерными. Эти два свойства не соответствуют структуре реальных социальных сетей. Распределения степеней обычно подчинено степенному закону: вершин с маленькой степенью на порядки больше, чем вершин с большой степенью. Этим же свойством обладает и распределение размеров сообществ.

Блочная двухуровневая модель Эрдёша-Реньи (BTER) [61] позволяет генерировать социальные графы со степенными распределениями для распределения степеней вершин и размеров сообществ. Генерация проходит в 3 этапа. Прежде всего, выполняется *предварительная обработка*: каждая вершина степени 2 или выше распределяется в сообщество. Далее, в *Фазе 1* моделируется локальная структура внутри каждого сообщества как граф Эрдёша-Реньи. Вероятность ребра для сообщества G_k определяется как $\rho_k = \rho\left[1 - \eta\left(\frac{\log(d_k+1)}{\log(d_{max}+1)}\right)^2\right]$, где $d_k = min\{d_i | i \in G_k\}$, d_{max} — максимальная степень вершины во всём графе, а ρ и η — параметры. После этого во время *Фазы 2* создаются связи между сообществами. Применяется модель Чунг-Лу [62] для исходящей степени вершины e_i, которая определяется следующим образом:

$$e_i = \begin{cases} 1, & \text{если } d_i = 1; \\ d_i - \rho_{k_i}(|G_{k_i}| - 1), & \text{иначе,} \end{cases} \tag{2.2}$$

где $|G_k|$ — размер k-го сообщества.

Минусы этой модели в том, что она не позволяет создавать сети с пересекающейся структурой сообществ, а также вероятность ребра не зависит от размера сообщества.

Модель **случайных графов пересечений** [63, 64] может быть представлена следующим образом: V – множество вершин ($|V| = n$), A – множество множеств из m элементов. Для $p \in [0, 1]$ строится двудольный граф $B(n, m, p)$ с двумя долями вершин V и A включением каждого из возможных nm рёбер между элементами из V и элементами из A независимо с вероятностью p. После создаётся случайный граф пересечений $G(n, m, p)$ с множеством вершин V путём связывания двух различных вершин $i, j \in V$ если и только если существует элемент $a \in A$ такой, что и i, и j смежны с a в $B(n, m, p)$.

Если рассматривать вершины в V как пользователей, а элементы множества A как сообщества, то получается модель социальной сети, в которой пара пользователей может быть связана ребром, если они одновременно состоят в хотя бы одном общем сообществе.

Недостатком модели случайных графов пересечений является необходимость задания количества сообществ m в качестве входного параметра, а

также несоответствующее степенному закону распределение степеней вершин. Кроме того, отсутствует возможность управления вероятностью ребра в отдельных сообществах.

Генератор Ланчичинетти-Фортунато-Радиччи (LFR)[6] [13] основан на более реалистичной модели социального графа с сообществами. В этой модели распределения степеней вершин и размеров сообществ генерируются в соответствии со степенным законом с различными экспонентами τ_1 и τ_2 соответственно.

Сам граф строится следующим образом (рисунки 2.3 и 2.4):

1) генерируется последовательность размеров сообществ, подчиняющаяся степенному закону с экспонентой τ_2;

2) генерируется последовательность степеней вершин, подчиняющаяся степенному закону с экспонентой τ_1. Для каждой вершины i со степенью k_i определяется внутренняя степень $k_i^{(in)} = (1-\mu_t)k_i$, где $0 \leq \mu_t \leq 1$ — *топологический параметр смешивания*. Внутренняя степень вершины i соответствует числу её соседей, которые состоят как минимум в 1 общем сообществе с i;

3) вершины в каждом сообществе соединяются с использованием модели конфигураций [65];

4) для каждой вершины вычисляется внешняя степень $k_i^{(out)} = k_i - k_i^{(in)}$, после чего вершины случайным образом соединяются рёбрами с другими вершинами из различных сообществ. При этом сохраняется внутренняя степень $k_i^{(in)}$ для вершин, входящих в несколько сообществ.

Несмотря на тот факт, что шаблонные сети LFR являются де-факто “золотым стандартом” для оценки результатов алгоритмов поиска сообществ, у них есть несколько существенных недостатков:

- вершины делятся на пересекающиеся и непересекающиеся, а все вершины из пересечений входят в одно и то же количество сообществ. Кроме того, не выполняются некоторые другие структурные свойства сообществ (раздел 3.5.1).;

[6] https://sites.google.com/site/andrealancichinetti/files

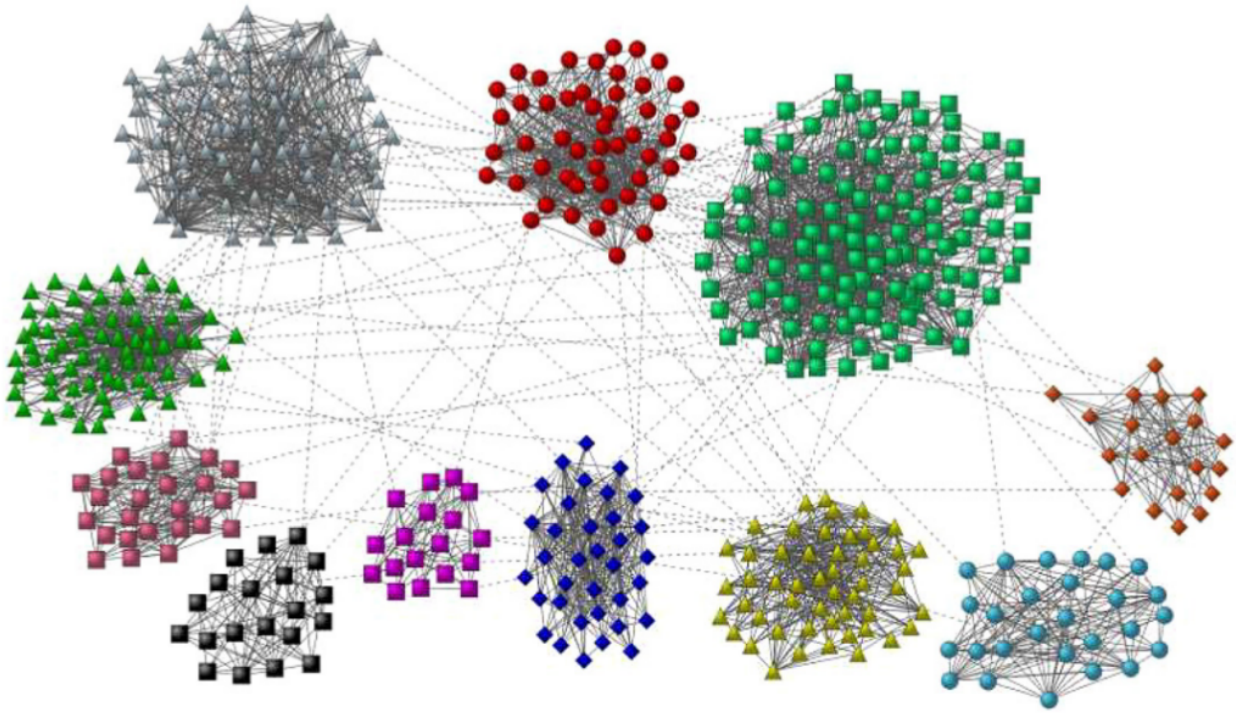


Рисунок 2.3: Шаблонная сеть LFR из 500 вершин с непересекающимися сообществами.

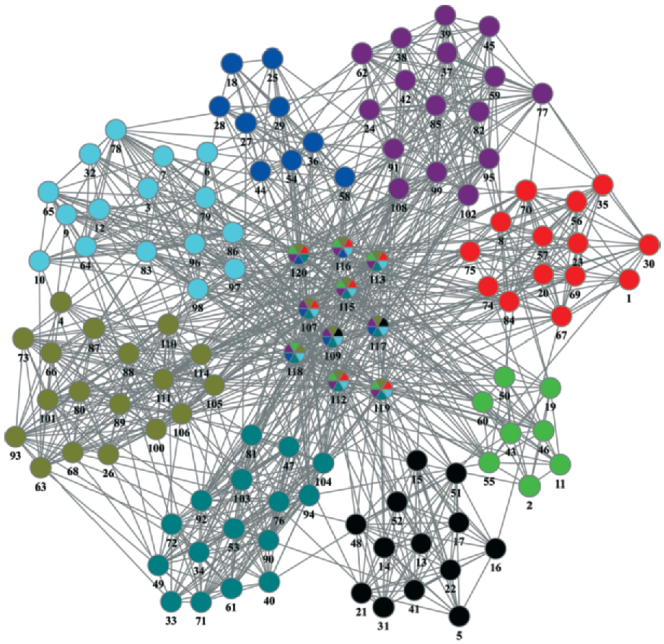


Рисунок 2.4: Шаблонная сеть LFR из 120 вершин с пересекающимися сообществами пользователей. Каждая из $O_n = 10$ вершин в центре состоит в $O_m = 6$ сообществах.

- значение топологического параметра смешивания μ_t одинаково для всех вершин;
- длительное время генерации.

Генератор agmgen[7] основан на предложенной разработчиками *графовой модели принадлежности пользователей к сообществам AGM* (раздел 3.4.1).

Генератору agmgen в качестве входных данных требуется двудольный граф, задающий принадлежность пользователей сообществам. Далее выполняется генерация ребёр независимо в каждом сообществе по модели Эрдёша-Реньи.

Вместе с тем, при генерации шаблонных сетей зачастую удобнее указать параметры генерации графа "пользователь-сообщество", нежели использовать данные из реальных сетей или генерировать такой граф отдельно. Кроме того, agmgen (как и LFR) не позволяет генерировать графы из сотен миллионов вершин, что затрудняет оценку применимости методов определения структуры сообществ к социальным графам большой размерности.

Тем не менее, AGM является одной из наиболее реалистичных на данный момент моделей социальной сети с сообществами пользователей. Поэтому предложенный в данной работе метод генерации шаблонных сетей (глава 3) также основывается на этой модели, но при этом лишён описанных недостатков agmgen.

Способы сравнения покрытий

Для того, чтобы установить близость известного и найденного покрытий, принято использовать набор специализированных мер сравнения множеств вершин: нормализованная взаимная информация [13], Омега-индекс [66], средняя F_1-мера классификации вершин [5] и др.

Наиболее распространённым способом сравнения покрытий (X, Y) является вычисление значения *нормализованной взаимной информации* (англ. *Normalized Mutual Information, NMI*):

$$NMI(X,Y) = 1 - \frac{1}{2}[H(X|Y)_{norm} + H(Y|X)_{norm}] \tag{2.3}$$

Для каждого сообщества X_k находится ближайшее Y_k в смысле неопределённости информации $H(X_k|Y_j) \to min_j$, где X_k — случайная переменная, соответствующая вероятности возникновения вершины в сообществе k,

[7] https://github.com/snap-stanford/snap/tree/master/examples/agmgen

$H(X_k|Y_j)$ — условная энтропия X_k при условии Y_j. H_{norm} вычисляется как нормализация $H(X_k|Y_j)$ от количества всей информации о X_k, усредняя по всем сообществам в X.

Значения NMI лежат в промежутке $[0; 1]$. Минимальное значение соответствует абсолютно разным покрытиям, максимальное — полностью совпадающим покрытиям.

2.2.2 Качество приложений

Знание структуры сообществ пользователей находит применение в ряде практических приложений анализа социальных данных:

1) определение значений скрытых атрибутов пользователей [67];
2) оптимизация передачи сообщений в коммуникационных сетях [68];
3) ограничение распространения вредоносного программного обеспечения [68];
4) идентификация распространителей спам-сообщений [69];
5) рекомендация товаров, услуг и контента пользователям социальных сетей [70, 71].

В данной работе покрытия, полученные различными алгоритмами, сравнивались путём косвенного оценивания их качества с помощью задачи определения скрытых атрибутов пользователей (раздел 4.6.2). Метки сообществ для каждой вершины были использованы в качестве векторов признаков для обучения классификатора. В дальнейшем точность классификации использовалась в качестве оценки качества различных методов.

Другим аспектом практического использования полученных сообществ является упрощение анализа сложных сетей больших размеров. Большинство методов анализа не предназначены для обработки графов размером $> 10^6$ вершин. Поэтому для таких графов важно сначала определить один или несколько подграфов, представляющих интерес для дальнейшего анализа.

В этом случае масштабируемый алгоритм определения структуры сообществ может помочь идентифицировать сообщества, соответствующие естественным модулям исследуемой сети (страны, организации), которые могут быть без труда идентифицированы аналитиком. В дальнейшем каждое из выбранных на предварительном этапе сообществ может быть исследовано с помощью других методов анализа структуры сетей. Например, становится возможным анализ структуры связей всего социального графа, определение целевых сообществ пользователей и детальный анализ путей распространения информации в соответствующих им подграфах.

2.3 Выводы

В данном разделе были исследованы основные классы методов определения структуры сообществ со значительным пересечением множеств вершин в социальных сетях, а также способы оценки качества результатов таких методов.

По результатам исследования класс методов, основанных на распространении меток, является оптимальным по комбинации качества и масштабируемости. Вместе с тем, качество результатов имеет тенденцию к резкому ухудшению с ростом степени пересечения сообществ. Поскольку сообщества в реальных социальных сетях имеют тенденцию к значительному пересечению, было принято решение исследовать возможность модификации метода SLPA с целью улучшения качества результатов с сохранением низкой временной сложности и хорошей масштабируемости.

Для оценки качества методов определения структуры сообществ целесообразно использование генератора шаблонных сетей LFR. Однако покрытия, синтезируемые LFR, не соответствуют некоторым из недавно открытых свойств структуры сообществ реальных социальных сетей. Кроме того, создание сетей из более 10^6 вершин на практике затруднено в силу вычислительной сложности алгоритма генерации и нераспределённой реализации. Поэтому было принято решение исследовать возможность создания метода генерации более реалистичных шаблонных сетей размером $> 10^6$ вершин.

Глава 3

Распределённый метод генерации случайных социальных графов с заданной структурой сообществ

К настоящему времени исследователями социальных сетей созданы наборы реальных данных для тестирования методов определения структуры сообществ. Такие *шаблонные сети* (раздел 2.2.1) состоят из набора вершин и рёбер социального графа (пользователи и связи между ними), а также списка сообществ, в которых состоит каждый пользователь. Однако сбор реальных данных из сервисов социальных сетей часто занимает длительное время, а свойства полученных наборов данных не соответствуют желаемым.

Поэтому важно определить фундаментальные свойства структуры сообществ пользователей, основываясь на реальных шаблонных сетях, и разработать инструмент для генерации синтетических шаблонных сетей с контролируемыми характеристиками: количество вершин и рёбер графа, количество и размеры сообществ, количество сообществ у пользователя и т.д. Используя синтетические шаблонные сети, можно выполнять более достоверное и комплексное тестирование методов определения структуры сообществ, по-

скольку различные значения параметров шаблонной сети могут оказывать существенное влияние на результаты.

Вместе с тем, многие известные генераторы шаблонных сетей не учитывают ряда важных структурных свойств сообществ (раздел 2.2.1). Это указывает на необходимость пересмотра требований к таким генераторам как методам оценки качества методов определения структуры сообществ.

Кроме того, известные генераторы шаблонных сетей имеют существенные ограничения в плане производительности при генерации графов с $> 10^6$ вершин, что затрудняет оценку применимости методов определения структуры сообществ к социальным графам большой размерности.

Таким образом, автору неизвестны методы генерации случайных социальных графов с сообществами пользователей, поддерживающие все из перечисленных в разделе 1.3 свойств и одновременно позволяющие эффективную распределённую реализацию для генерации графов из сотен миллионов вершин. В данной главе приводится описание такого метода, который был разработан, реализован и исследован в рамках диссертационной работы.

Основные результаты главы опубликованы в работах [25, 26].

3.1 Постановка задачи

Рассмотрим социальный граф $G(V, E)$, где V соответствует множеству пользователей социальной сети, а E — множеству социальных связей между пользователями.

Рассмотрим двудольный граф $B(V, \mathbb{C}, M)$, где V соответствует множеству вершин социального графа G, $\mathbb{C}$ — покрытие (множество сообществ), а ребро $(u, c) \in M$ соединяет вершину $u \in V$ с сообществом $Z_c \in \mathbb{C}$, если $u \in V_c$.

Требуется разработать метод генерации пар графов $G(V, E)$ и $B(V, \mathbb{C}, M)$ с перечисленными в разделе 1.3 свойствами, характерными для социальных графов с сообществами пользователей. Кроме того, распределение степеней вершин V должно следовать степенному закону.

Для управления свойствами генерируемых графов метод должен поддерживать следующие параметры:

1) количество вершин в V;

2) средняя степень вершины в V;

3) параметры степенного распределения количества сообществ у пользователя: минимальное и максимальное значения, экспонента;

4) параметры степенного распределения размеров сообществ: минимальное и максимальное значения, экспонента.

Кроме того, метод должен позволять разбивать задачу генерации рёбер графа $B(V, \mathbb{C}, M)$ на независимые подзадачи, каждая из которых может решаться независимо от других с последующей агрегацией результатов решения всех подзадач. Аналогичное требование предъявляется к способу решения задачи генерации рёбер графа $G(V, E)$. Как следствие, метод должен позволять распределённую реализацию с близкой к линейной масштабируемостью в зависимости от $|E|$.

3.2 Общая схема метода

Разработанный метод генерации случайных социальных графов с заданной структурой пересекающихся сообществ пользователей получил название СКВ[1] и состоит из двух последовательных этапов:

1) пользователи распределяются по сообществам;

2) генерируются связи между пользователями внутри каждого сообщества.

Предложенный метод был реализован на языке программирования Scala с использованием Apache Spark[2] — фреймворка для распределённых вычислений в распределённой среде[3]. Данный фреймворк позволяет эффективно выполнять все необходимые операции за счёт использования структурной абстракции данных, именуемой *сбоеустойчивые распределённые наборы данных* (англ. *Resilient Distributed Datasets*, *RDD*). RDD — это коллекция объектов, распределённых между множеством машин, которая может быть восстановлена, если какая-то часть объектов утеряна. По умолчанию все RDD

[1] По первым буквам фамилий разработчиков в латинской транскрипции.

[2] https://spark.apache.org/

[3] Веб-демонстрация разработанного прототипа доступна по адресу: http://ckb.at.ispras.ru/home/

хранятся в оперативной памяти. При обновлении данных старый RDD не изменяется, но создаётся новый, содержащий ссылку на предыдущую версию и список изменений. Перечисленные особенности Spark позволяют увеличивать скорость работы программы в несколько раз по сравнению с другими платформами для распределённых вычислений, в частности, Apache Hadoop.

Схема работы генератора на вычислительном кластере показана на рисунке 3.1. *Главный узел (мастер)* — главная машина в вычислительном кластере. *Ведомыми узлами (слэйвами)* называются остальные машины кластера. В процессе распределённых вычислений главный узел назначает задачи слэйвам, координирует их работу и агрегирует результаты. Нераспределённая часть вычислений также выполняется на главном узле. Синтезированные графы сохраняются в виде файлов распределённой файловой системы *Hadoop Distributed File System, HDFS*, части которых сохраняются в локальную файловую систему на слэйвах.

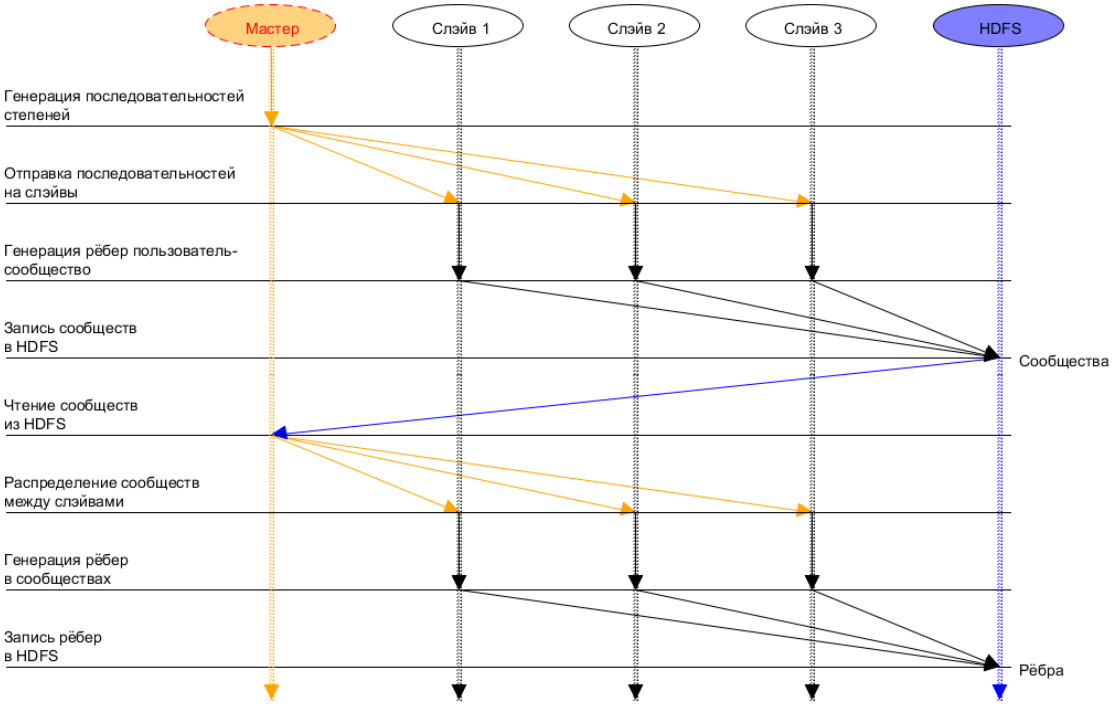


Рисунок 3.1: Схема работы распределённой реализации метода СКВ

На первом шаге процесса генерации создаются две последовательности степеней вершин для долей генерируемого графа “пользователь-сообщество”. Далее мастер отправляет эти последовательности на каждый из слэйвов. На следующем шаге каждый из s слэйвов независимо генерирует $\frac{|M|}{s}$ рёбер (раздел 3.3) между пользователями и сообществами в двудольном графе. Все

рёбра объединяются в один список рёбер, и на его основе создаётся список сообществ, который сохраняется в распределённой файловой системе HDFS.

После этого мастер считывает из хранилища сообщества, создаёт v копий каждого сообщества, равномощно группирует их и равномерно распределяет их по слэйвам. Далее каждый из ведомых узлов генерирует конфигурации рёбер в $\frac{T_c}{v}$ тройках вершин каждого полученного сообщества Z_c (раздел 3.4.2). Все рёбра объединяются в общий список и записываются в HDFS.

3.3 Генерация двудольного графа “пользователь–сообщество”

На данном этапе производится генерация графа $B(V, \mathbb{C}, M)$ с учётом параметров генерации (таблица 3.1). Для этого требуется вычислить количество генерируемых рёбер M, а также сгенерировать последовательности степеней для V и $\mathbb{C}$.

Таблица 3.1: Параметры этапа генерации двудольного графа “пользователь–сообщество”

Параметр	**Описание**	**По умолчанию**
N_1	Количество пользователей	–
x_{min}	Минимальный размер сообщества	3
m_{min}	Минимальное количество сообществ у пользователя	1
x_{max}	Максимальный размер сообщества	100
m_{max}	Максимальное количество сообществ у пользователя	100
$\beta_1 > 1$	Экспонента степенного распределения количества сообществ у пользователей	2.5
$\beta_2 > 1$	Экспонента степенного распределения размеров сообществ	2.5

Согласно модели генерации, количество сообществ $m_j \in \mathbb{N}$ пользователя j и размер $x_c \in \mathbb{N}$ сообщества Z_c являются случайными величинами со степенным распределением с экспонентами β_1 и β_2:

$$m_j \sim m^{-\beta_1}, \forall m_j : m_{min} \leq m_j \leq m_{max}, \tag{3.1}$$

$$x_c \sim x^{-\beta_2}, \forall x_c : x_{min} \leq x_c \leq x_{max}. \tag{3.2}$$

Сначала генерируются ожидаемые последовательности степеней в долях графа $B(V, \mathbb{C}, M)$: $\vec{d^1} = (d_1^1, ..., d_{N_1}^1)$ и $\vec{d^2} = (d_1^2, ..., d_{N_2}^2)$ для количества сообществ у пользователя и размеров сообществ соответственно. Количество пользователей N_1 задаётся в виде параметра, а количество сообществ N_2 сначала должно быть вычислено исходя из остальных параметров генерации. Элементы последовательностей независимо семплируются из распределений $m^{-\beta_1}$ и $x^{-\beta_2}$.

Отметим, что приведённый ниже способ генерации рёбер не обеспечивает точной реализации последовательностей степеней $\vec{d_1}$ и $\vec{d_2}$. Однако математические ожидания $\mathbb{E}[m_j]$ числа вхождений пользователей в сообщества и $\mathbb{E}[x_c]$ размеров сообществ приблизительно соответствуют математическим ожиданиям степеней вершин в долях генерируемого графа, откуда получаем следующее выражение для количества сообществ:

$$N_2 = \left\lfloor \frac{N_1 \cdot \mathbb{E}[m_j]}{\mathbb{E}[x_c]} \right\rfloor. \tag{3.3}$$

Отметим, что k-й момент случайной величины $y_i \sim y^{-\beta}$ при условии $\forall y_i : y_{min} \leq y_i \leq y_{max}$ равен:

$$\mathbb{E}[y_i^k] = \int_{y_{min}}^{y_{max}} y_i^k p(y_i) dy_i = \int_{y_{min}}^{y_{max}} y_i^k \frac{1-\beta}{y_{max}^{1-\beta} - y_{min}^{1-\beta}} y_i^{-\beta} dy_i. \tag{3.4}$$

Таким образом,

$$\mathbb{E}[y_i^k] = \frac{(1-\beta)(y_{max}^{k-\beta+1} - y_{min}^{k-\beta+1})}{(y_{max}^{1-\beta} - y_{min}^{1-\beta})(k+1-\beta)}, \tag{3.5}$$

а при $k - \beta + 1 = 0$

$$\mathbb{E}[y_i^k] = \frac{1-\beta}{y_{max}^{1-\beta} - y_{min}^{1-\beta}} \ln\left(\frac{y_{max}}{y_{min}}\right). \tag{3.6}$$

Для случая $k = 1$ и $\beta \neq 2$ в формуле 3.3 получаем формулы зависимости математических ожиданий m_j и x_c от параметров генерации:

$$\mathbb{E}[m_j] = \frac{(1-\beta_1)(m_{max}^{2-\beta_1} - m_{min}^{2-\beta_1})}{(m_{max}^{1-\beta_1} - m_{min}^{1-\beta_1})(2-\beta_1)}, \tag{3.7}$$

$$\mathbb{E}[x_c] = \frac{(1-\beta_2)(x_{max}^{2-\beta_2} - x_{min}^{2-\beta_2})}{(x_{max}^{1-\beta_2} - x_{min}^{1-\beta_2})(2-\beta_2)}. \tag{3.8}$$

Далее вычисляется количество генерируемых рёбер $|M|$ между долями графа B. В силу независимой генерации рёбер требуется поправка на вероятность кратных рёбер:

$$|M| = \lfloor (1 + P_{(c,j)}^{\geq 2}) M_0 \rfloor, \tag{3.9}$$

где $P_{(c,j)}^{\geq 2}$ — вероятность ребра (c, j) кратности ≥ 2 (раздел 3.3.1), а M_0 соответствует ожидаемому количеству рёбер:

$$M_0 = \lfloor N_1 \cdot \mathbb{E}[m_j] \rfloor = \lfloor N_2 \cdot \mathbb{E}[x_c] \rfloor. \tag{3.10}$$

Для генерации рёбер между долями графа (рисунок 3.2) согласно последовательностям степеней $\vec{d_1}$ и $\vec{d_2}$ вычисляются значения

$$D_1^1 = d_1^1, D_2^1 = D_1^1 + d_2^1, ..., D_{k+1}^1 = D_k^1 + d_{k+1}^1, ..., D_{N_1}^1 = D_{N_1-1}^1 + d_{N_1}^1,$$

$$D_1^2 = d_1^2, D_2^2 = D_1^2 + d_2^2, ..., D_{k+1}^2 = D_k^2 + d_{k+1}^2, ..., D_{N_2}^2 = D_{N_2-1}^2 + d_{N_2}^2.$$

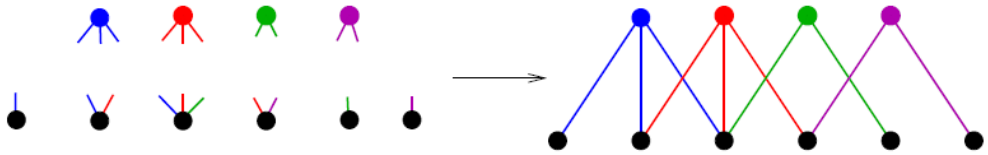


Рисунок 3.2: Генерация рёбер в двудольном графе

Далее для последовательности натуральных чисел

$$[M] = \{1, 2, 3, \ldots, |M|\},$$

выполняется в цикле для $t = 1$ до $|M|$:

1) выбираются случайные числа p и q из $[M]$ равномерно;

2) находится интервал $[D_i^1, D_{i+1}^1]$, которому принадлежит p;

3) находится интервал $[D^2_j, D^2_{j+1}]$, которому принадлежит q;

4) в M добавляется ребро (i, j).

Описанный алгоритм генерации позволяет каждому из s ведомых узлов вычислительного кластера выполнять независимую генерацию $\frac{|M|}{s}$ рёбер, имея лишь информацию об ожидаемых последовательностях степеней в долях графа. Таким образом, процесс генерации разбивается на независимые подзадачи, по завершении которых все сгенерированные рёбра агрегируются, сортируются по одному из концов, а кратные рёбра удаляются.

Временная сложность этапа генерации двудольного графа:

$$O(|M|(\log(N_1) + \log(N_2))) = O(|M| \log(N_1 N_2)), \tag{3.11}$$

поскольку для каждого из $|M|$ рёбер требуется выполнить бинарный поиск интервалов в последовательностях $D^1_{1:N_1}$ и $D^2_{1:N_2}$, содержащих N_1 и N_2 интервалов соответственно. Временная сложность бинарного поиска равна $\log(N_1) + \log(N_2)$ для каждого ребра.

Таким образом, предложенный метод обеспечивает выполнение 3 структурных свойств сообществ, описанных в разделе 1.3:

- множества вершин сообществ могут пересекаться;
- распределение размеров сообществ подчиняется степенному закону;
- распределение количества сообществ, в которых состоит пользователь, подчиняется степенному закону.

3.3.1 Кратные рёбра

Поскольку рёбра в предложенном методе генерируются независимо, то после удаления кратных рёбер из общего списка по окончании генерации количество рёбер несколько отличается от ожидаемого значения M_0, заданного в соответствии с входными параметрами. В разработанном методе в качестве поправки при расчёте количества генерируемых рёбер (формула 3.9) используется вероятность генерации ребра кратности ≥ 2. Это позволяет уменьшить погрешность количества сгенерированных рёбер, связанную с удалением кратных рёбер.

Теорема 1. *Вероятность ребра (c, j) кратности ≥ 2 в случайном двудольном графе "пользователь-сообщество" при независимой генерации M_0 рёбер между его долями удовлетворяет*

$$P_{(c,j)}^{\geq 2} \to \frac{\mathbb{E}[x_c^2]\mathbb{E}[m_j^2]}{2M_0^2} \tag{3.12}$$

при условии $\frac{x_{max}m_{max}}{M_0} \to 0$,
где x_c — случайная величина, соответствующая размеру сообщества Z_c, m_j — случайная величина, соответствующая количеству сообществ у пользователя j.

Доказательство. Вероятность ребра (c, j) кратности k между вершинами с заданными степенями x_c и m_j:

$$P_{(c,j)|x_c,m_j}^k = C_{M_0}^k \left(\frac{x_c m_j}{M_0^2}\right)^k \left(1 - \frac{x_c m_j}{M_0^2}\right)^{M_0-k}, \tag{3.13}$$

где $C_{M_0}^k$ — биномиальный коэффициент, второй множитель соответствует вероятности того, что k из M_0 рёбер будут соответствовать ребру (c, j), третий множитель соответствует вероятности того, что оставшиеся $(M_0 - k)$ рёбер не будут сгенерированы между вершинами c и j.

Найдём вероятность отсутствия ребра (c, j), используя формулу Тейлора:

$$\begin{aligned} P_{(c,j)|x_c,m_j}^0 &= \left(1 - \frac{x_c m_j}{M_0^2}\right)^{M_0} = \\ &= 1 - \frac{x_c m_j}{M_0} + \frac{1}{2}\left(\frac{x_c m_j}{M_0}\right)^2 + O\left(\left(\frac{x_c m_j}{M_0}\right)^3\right), \end{aligned} \tag{3.14}$$

и вероятность ребра (c, j) кратности 1:

$$\begin{aligned} P_{(c,j)|x_c,m_j}^1 &= \frac{x_c m_j}{M_0}\left(1 - \frac{x_c m_j}{M_0^2}\right)^{M_0-1} = \\ &= \frac{x_c m_j}{M_0} - \left(\frac{x_c m_j}{M_0}\right)^2 + O\left(\left(\frac{x_c m_j}{M_0}\right)^3\right). \end{aligned} \tag{3.15}$$

Тогда вероятность ребра кратности ≥ 2 равна

$$\begin{aligned} P^{\geq 2}_{(c,j)|x_c,m_j} &= 1 - P^0_{(c,j)|x_c,m_j} - P^1_{(c,j)|x_c,m_j} = \\ &= \frac{1}{2}\left(\frac{x_c m_j}{M_0}\right)^2 + o\left(\left(\frac{x_c m_j}{M_0}\right)^2\right). \end{aligned} \tag{3.16}$$

Найдём искомую вероятность для произвольных x_c и m_j по формуле полной вероятности:

$$\begin{aligned} P^{\geq 2}_{(c,j)} &= \sum_c \sum_j P(x_c, m_j) P^{\geq 2}_{(c,j)|x_c,m_j} = \\ &= \sum_c \sum_j P(x_c, m_j) \left(\frac{1}{2}\left(\frac{x_c m_j}{M_0}\right)^2 + o\left(\left(\frac{x_c m_j}{M_0}\right)^2\right)\right) = \\ &= \frac{\mathbb{E}[(x_c m_j)^2]}{2M_0^2} + o\left(\frac{\mathbb{E}[(x_c m_j)^2]}{M_0^2}\right). \end{aligned} \tag{3.17}$$

При переходе к пределу $\frac{x_{max} m_{max}}{M_0} \to 0$

$$P^{\geq 2}_{(c,j)} \to \frac{\mathbb{E}[(x_c m_j)^2]}{2M_0^2}. \tag{3.18}$$

Используя независимость x_c и m_j, получаем

$$P^{\geq 2}_{(c,j)} \to \frac{\mathbb{E}[x_c^2]\mathbb{E}[m_j^2]}{2M_0^2}. \tag{3.19}$$

□

Вторые моменты $\mathbb{E}[x_c^2]$ и $\mathbb{E}[m_j^2]$ вычисляются по формулам 3.5 и 3.6.

Доказанная теорема, в частности, применима при генерации социальных графов с большим количеством вершин N_1, для которых целесообразно ограничить сверху размер сообщества и количество сообществ у пользователя с помощью параметров x_{max} и m_{max}. При этом количество генерируемых рёбер $M_0 \to \infty$, что обеспечивает выполнение условия теоремы.

3.4 Генерация рёбер внутри сообществ

Вторым этапом предложенного метода является генерация рёбер в графе $G(V, E)$, которая проходит независимо в каждом из сгенерированных на предыдущем этапе сообществ согласно модели AGM.

3.4.1 Модель AGM

Графовая модель принадлежности пользователей к сообществам (англ. *community-affiliation graph model, AGM*) — вероятностная генеративная модель для графов, которая представляет организацию сложных сетей в виде пересекающихся сообществ. Она была разработана Yang et al [8] путём анализа данных о реальных сообществах пользователей в социальных сетях (раздел 1.3). Модель представляет принадлежность вершин к сообществам как двудольный граф, в котором рёбра от пользователя идут к сообществам, которым этот пользователь принадлежит.

Другая часть модели основана на том, что люди принадлежат многим сообществам (друзья, члены семьи, коллеги), но связи между ними часто возникают как результат одной доминирующей причины. Такой характер связей моделируется заданием для каждого сообщества вероятности, с которой вершина будет соединена ребром с другой вершиной из этого сообщества (рисунок 3.3). В результате каждое сообщество, которому принадлежит пара вершин, имеет независимый шанс создать ребро между этими двумя вершинами. Следовательно, чем большему количеству сообществ принадлежит пара пользователей, тем больше вероятность того, что между ними будет образовано ребро.

Рассмотрим двудольный граф $B(V, \mathbb{C}, M)$, где V соответствует множеству вершин социального графа G, $\mathbb{C}$ — покрытие, а ребро $(u, c) \in M$ соединяет вершину $u \in V$ с сообществом $Z_c \in \mathbb{C}$, если $u \in V_c$. Пусть $\{p_c\}$ соответствует множеству вероятностей для всех сообществ $Z_c \in \mathbb{C}$.

Согласно модели AGM, неориентированный граф $G(V, E)$ генерируется следующим образом. Для каждой пары вершин $u, v \in V$ создаётся ребро

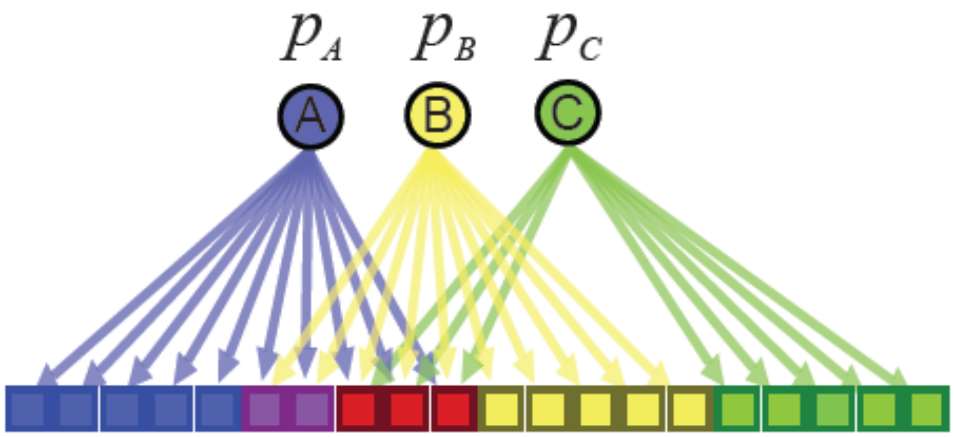


Рисунок 3.3: Модель AGM: двудольный граф “пользователь-сообщество” с заданными вероятностями рёбер в каждом сообществе.

$(u, v) \in E$ с вероятностью $p(u, v)$:

$$p(u,v) = \begin{cases} 1 - \prod\limits_{k \in \mathcal{G}_{uv}} (1 - p_k), & \mathcal{Z}_{uv} \neq \varnothing \\ \epsilon, & \mathcal{Z}_{uv} = \varnothing \end{cases} \tag{3.20}$$

где $\mathcal{Z}_{uv} \subseteq \mathbb{C}$ — множество сообществ, в которые входят одновременно вершины u и v: $\mathcal{Z}_{uv} = \{Z_c | (u, c), (v, c) \in M\}$, ϵ — константа.

Ниже приведён ряд доказанных авторами модели теорем о свойствах описываемых ей графов:

1) количество рёбер $|E_c|$ между вершинами сообщества Z_c растёт сверхлинейно как функция $|V_c|$ при условии $p_c \propto n_c^{-\beta}$, $1 > \beta > 0$;

2) доля связанных ребром вершин в подграфе O_{AB}, порождённом пересечением $V_A \cap V_B$ множеств вершин сообществ Z_A и Z_B, больше, чем доли связанных ребром вершин в подграфе сообщества Z_A, порождённом множеством $V_A \setminus V_{O_{AB}}$, и подграфе сообщества Z_B, порождённом множеством $V_B \setminus V_{O_{AB}}$;

3) для сообществ Z_A и Z_B, подграфа O_{AB} (порождённого пересечением $V_A \cap V_B$ множеств вершин сообществ Z_A и Z_B) и связующей вершины v_A (которая среди всех вершин Z_A имеет наибольшее количество связей с другими вершинами из этого сообщества) $p(v_A \in V_{O_{AB}}) > \frac{|V_{O_{AB}}|}{|V_A|}$;

4) условная вероятность ребра между вершинами $u, v \in V$ в графе $G(V, E)$ является растущей функцией от $|\mathcal{Z}_{uv}|$ при условии $\forall Z_c : p_c = p$.

Таким образом, модель обеспечивает выполнение следующих структурных свойств сообществ, описанных в разделе 1.3:

- вероятность ребра между парой вершин увеличивается с ростом количества общих сообществ, которым принадлежат обе вершины;
- для пары сообществ пересечение их более плотно, чем непересекающаяся часть;
- количество рёбер в сообществе растёт суперлинейно с размером сообщества;
- связующие вершины сообщества (имеющие среди всех вершин сообщества наибольшее количество связей с другими вершинами из этого сообщества) более вероятно находятся в пересечениях с другими сообществами, чем в непересекающейся области сообщества.

3.4.2 Схема генерации рёбер внутри сообществ

Для генерации графа $G(V, E)$ по модели AGM требуется, прежде всего, задать двудольный граф $B(V, \mathbb{C}, M)$ и параметры $\{p_c\}$ и ϵ. В разработанном методе используется случайный двудольный граф “пользователь-сообщество”, созданный на предыдущем этапе. Затем нужно сгенерировать необходимое количество ребёр между вершинами из V. Параметры этапа генерации рёбер внутри сообществ перечислены в таблице 3.2.

Рёбра в графе $G(V, E)$ генерируются независимо в каждом сообществе Z_c с вероятностью p_c, что позволяет выполнять генерацию одновременно во всех сообществах и упрощает распределённую реализацию:

$$p_c = \frac{\alpha}{x_c^{\gamma}}, \tag{3.21}$$

где x_c — размер сообщества, $\gamma \in (0; 1)$ — параметр модели, $\alpha > 0$ определяется средней степенью вершины либо задаётся в виде параметра. Способ вычисления α исходя из заданных значений γ и средней степени описан в разделе 3.4.4.

Таблица 3.2: Параметры этапа генерации рёбер внутри сообществ

Параметр	**Описание**	**По умолчанию**
$\alpha > 0$	Увеличивает вероятность ребра внутри сообщества	4
$0 < \gamma < 1$	Уменьшает вероятность ребра внутри сообщества	0.5
ϵ	Вероятность ребра между произвольной парой вершин	$2N_1^{-1}$
λ_1	Увеличивает средний коэффициент кластеризации вершин сообщества	1

Согласно оригинальной модели AGM, если пара вершин не имеет общих сообществ, то вероятность создания ребра между ними равна ϵ. Однако соблюдение этого условия требует вычисления количества общих сообществ для каждой пары вершин генерируемого графа, что занимает длительное время для графов большой размерности.

В разработанном методе ϵ соответствует вероятности ребра между парой произвольных вершин графа. На практике это достигается добавлением дополнительного ϵ-сообщества с вероятностью ребра ϵ. Таким образом, существует небольшая “фоновая” вероятность образования ребра между парой пользователей безотносительно сообществ, что соответствует положениям теории фокусов Фельда (раздел 1.3). Как показывают результаты экспериментов, данная модификация не изменяет основных структурных свойств генерируемых графов.

Для генерации рёбер внутри сообществ производится семплирование случайных графов из модели $\mathcal{G}(x_c, p_c)$ Эрдёша-Реньи для каждого сообщества. При этом для каждой пары вершин $i, j \in V_c$ создаётся ребро (i, j) с вероятностью p_c. Однако эта модель обладает известным недостатком, связанным с почти полным отсутствием кластерной структуры в создаваемых с её помощью графах [56].

Поэтому для генерации рёбер внутри сообществ используется *модель триплетов* (англ. *triplet model*) [72], которая в некотором смысле является генерализацией модели Эрдёша-Реньи. Вместо пар вершин рассматриваются все возможные комбинации троек вершин. Каждая тройка вершин может на-

ходиться в восьми *конфигурациях* (рисунок 3.4) в зависимости от наличия или отсутствия рёбер между упорядоченными вершинами.

Рисунок 3.4: Варианты конфигурации для тройки вершин.

Положим, что возможные конфигурации независимо распределены со следующими вероятностями: $P(000) = p_0$ (нет рёбер), $P(001) = P(010) = P(100) = p_1$ (одно ребро), $P(011) = P(101) = P(110) = p_2$ (два ребра), $P(111) = p_3$ (клика размера 3).

Для генерации каждого ребра внутри триплета с одинаковой вероятностью p примем $p_0 = (1-p)^3$, $p_1 = p(1-p)^2$, $p_2 = p^2(1-p)$, $p_3 = p^3$. Отметим, что при этом модель триплетов становится эквивалентной исходной модели $\mathcal{G}(x_c, p_c)$ Эрдёша-Реньи для сообщества.

Следовательно, заданная вероятность p_c ребра внутри сообщества связана с вероятностью p ребра в одной из троек сообщества следующим образом:

$$p_c = 1 - (1-p)^{x_c - 2}, \tag{3.22}$$

где $x_c = |V_c|$.

Получаем вероятность ребра внутри триплета:

$$p = 1 - e^{\frac{\ln(1-p_c)}{x_c-2}} = 1 - e^{\frac{\ln\left(1-\frac{\alpha}{x_c^2}\right)}{x_c-2}}. \tag{3.23}$$

Для генерации графа по описанной модели T_c раз выбирается случайная тройка вершин в сообществе и генерируется некоторая конфигурация рёбер в каждой выбранной тройке. T_c является случайной величиной, имеющей биномиальное распределение со следующими параметрами:

$$T_c \sim Bin(C_{x_c}^3, 3p_1 + 3p_2 + p_3), \tag{3.24}$$

Однако при независимом выборе троек вершин существует вероятность повторного выбора одной и той же тройки. Поэтому в разработанном методе выбирается $T_c(1+\delta)$ троек, где поправка δ находится следующим образом.

Обозначим $\bar{p}$ вероятность отсутствия ребра в тройке вершин:

$$\bar{p} = p_0 + 2p_1 + p_2. \tag{3.25}$$

Обозначим p_Δ вероятность выбора некоторой тройки для заданных вершин i и j:

$$p_\Delta = \frac{x_c - 2}{C^3_{x_c}}. \tag{3.26}$$

Тогда вероятность отсутствия ребра (i,j):

$$P^0_{(i,j)} = (1 - p_\Delta(1-\bar{p}))^{T_c(1+\delta)} = 1 - p_c. \tag{3.27}$$

Отсюда получаем искомую поправку:

$$\delta = \frac{\log(1-p_c)}{T_c \log(1 - p_\Delta(1-\bar{p}))} - 1. \tag{3.28}$$

Описанный алгоритм генерации позволяет разбивать задачу генерации конфигураций рёбер в $T_c(1+\delta)$ тройках вершин на независимые подзадачи, для выполнения которых требуются только значения x_c, p_c и λ_1 (раздел 3.4.3). В итоге одно, два или три ребра создаются внутри каждой тройки вершин с соответствующими нормализованными вероятностями конфигураций. В завершение кратные ребра удаляются.

Временная сложность этапа генерации рёбер внутри сообществ равна $O(\sum_{c=1}^{N_2} T_c)$.

3.4.3 Средний коэффициент кластеризации

Управление средним коэффициентом кластеризации вершин сообщества осуществляется с помощью параметров λ_1 и λ_2:

$$p_2 = \lambda_1 p^2(1-p), \tag{3.29}$$

$$p_3 = \lambda_1 p^3, \tag{3.30}$$

$$p_0 = \lambda_2(1-p)^3, \tag{3.31}$$

$$p_1 = \lambda_2 p(1-p)^2. \tag{3.32}$$

Таким образом, увеличение λ_1 повышает вероятность конфигураций триплетов с двумя и более рёбрами, что в конечном итоге ведёт к повышению среднего коэффициента кластеризации вершин сообщества.

Найдём зависимость λ_1 и λ_2:

$$\lambda_2(p_0 + 3p_1) + \lambda_1(3p_2 + p_3) = 1, \tag{3.33}$$

$$\lambda_2 p_{01} + \lambda_1(1 - p_{01}) = 1, \tag{3.34}$$

$$\lambda_2 = \frac{1 - \lambda_1(1-p_{01})}{p_{01}} \geq 0, \tag{3.35}$$

где $p_{01} = p_0 + 3p_1$.

Найдём ограничение на λ_1, который является параметром метода:

$$\lambda_1 \leq \frac{1}{1-p_{01}}. \tag{3.36}$$

3.4.4 Средняя степень

Так как средняя степень вершины является важным свойством для анализа графов и тестирования алгоритмов поиска сообществ, была исследована зависимость между средней степенью и входными параметрами α и γ, регулирующими вероятность ребра в сообществе. Используя полученную зависимость, можно задавать требуемую среднюю степень и параметр γ, а значение α рассчитывается исходя из этих условий.

Теорема 2. *Ожидаемая средняя степень вершины случайного социального графа из N вершин при независимой генерации его рёбер с вероятностью $p_c = \frac{\alpha}{x_c^\gamma}$ в каждом из сообществ удовлетворяет*

$$\mathbb{E}[d_v] \to (N-1)\sum_{r=1}^{k}(-1)^{r+1}\left[\alpha^r \sum_{c_1<\ldots<c_r} \mathbb{E}_{\xi_{ijc}}\left[\prod_{k=\{1,\ldots,r\}} \frac{x_{c_k}^{2-\gamma} m_i m_j}{M^2}\right]\right] \tag{3.37}$$

при условии $\frac{x_{max} m_{max}}{M} \to 0$,
где k — максимальная кратность ребра, ξ_{ijc} — индикаторная случайная

величина, соответствующая появлению ребра (i, j) в сообществе Z_c, x_c — случайная величина, соответствующая размеру сообщества Z_c, m_i и m_j — случайные величины, соответствующие количеству сообществ у произвольных пользователей i и j, M — случайная величина, соответствующая количеству рёбер в двудольном графе "пользователь-сообщество", $\alpha > 0$ и $\gamma \in (0; 1)$ — константы.

Доказательство. Поскольку рёбра в каждом сообществе генерируются независимо, то данный процесс генерации эквивалентен семплированию из модели Эрдёша-Реньи с N вершин и вероятностью $P(i, j)$ появления ребра между произвольной парой пользователей i и j.

В модели случайного графа $\mathcal{G}(n, p)$ Эрдёша-Реньи каждое ребро между n вершинами появляется с вероятностью p. Вероятность вершины иметь степень d описывается выражением

$$P(d_v = d) = C_{n-1}^{d} p^d (1-p)^{n-1-d}, \tag{3.38}$$

а ожидаемая средняя степень вершины графа $G \sim \mathcal{G}(n, p)$ равна

$$\mathbb{E}[d_v] = \sum_{d=0}^{n-1} d C_{n-1}^{d} p^d (1-p)^{n-1-d} = (n-1)p. \tag{3.39}$$

Рассмотрим случайную величину ξ_{ijc}:

$$\xi_{ijc} = \begin{cases} 1 & \text{, если } (i,j) \in E_c \\ 0 & \text{, если } (i,j) \notin E_c, \end{cases} \tag{3.40}$$

где E_c — множество рёбер сообщества Z_c.

Вероятность появления ребра (i, j) в Z_c:

$$P(\xi_c = 1) = p_c p_{ic} p_{jc}, \tag{3.41}$$

где p_c — вероятность ребра в сообществе, $p_{ic} = P(i \in V_c)$, $p_{jc} = P(j \in V_c)$.

При этом по аналогии с формулой 3.14:

$$p_{ic} = 1 - \left(1 - \frac{x_c m_i}{M^2}\right)^M = \frac{x_c m_i}{M} + O\left(\left(\frac{x_c m_i}{M}\right)^2\right). \tag{3.42}$$

При переходе к пределу $\frac{x_{max}m_{max}}{M} \to 0$

$$p_{ic} \to \frac{x_c m_i}{M}. \tag{3.43}$$

Используя принцип включений-исключений для рёбер кратности $r \in [1, k]$, получаем формулу вероятности ребра во всём графе:

$$P(i,j) = \sum_{r=1}^{k} (-1)^{r+1} S_r, \tag{3.44}$$

где k — максимальная кратность ребра в графе, а S_r соответствует всем рёбрам, которые появились не менее, чем в r сообществах:

$$\begin{aligned}
S_r &= \sum_{c_1<\ldots<c_r} \mathbb{E}_{\xi_{ijc}} \left[\prod_{k=\{1,\ldots,r\}} p_{c_k} p_{ik} p_{jk} \right] = \\
&= \sum_{c_1<\ldots<c_r} \mathbb{E}_{\xi_{ijc}} \left[\prod_{k=\{1,\ldots,r\}} \frac{\alpha}{x_{c_k}^{\gamma}} p_{ik} p_{jk} \right] \to \\
&\to \alpha^r \sum_{c_1<\ldots<c_r} \mathbb{E}_{\xi_{ijc}} \left[\prod_{k=\{1,\ldots,r\}} \frac{1}{x_{c_k}^{\gamma}} \prod_{t=\{i,j\}} \frac{x_{c_k} m_t}{M} \right] = \\
&= \alpha^r \sum_{c_1<\ldots<c_r} \mathbb{E}_{\xi_{ijc}} \left[\prod_{k=\{1,\ldots,r\}} \frac{x_{c_k}^{2-\gamma} m_i m_j}{M^2} \right].
\end{aligned} \tag{3.45}$$

Тогда по формуле 3.39:

$$\begin{aligned}
\mathbb{E}[d_v] &= (N-1) \cdot P(i,j) \to \\
&\to (N-1) \sum_{r=1}^{k} (-1)^{r+1} \left[\alpha^r \sum_{c_1<\ldots<c_r} \mathbb{E}_{\xi_{ijc}} \left[\prod_{k=\{1,\ldots,r\}} \frac{x_{c_k}^{2-\gamma} m_i m_j}{M^2} \right] \right]
\end{aligned} \tag{3.46}$$

□

После решения уравнения 3.37 k-ой степени с переменной α и фиксированной γ можно вычислить вероятности $\{p_c\}$, необходимые для достижения заданной средней степени (раздел 3.4.2).

При этом необходимо отметить, что

$$\sum_{c_1<c_2<\ldots<c_r} \mathbb{E}[x_{c_1}^{\theta_1} x_{c_2}^{\theta_2} \cdot \ldots \cdot x_{c_r}^{\theta_r}] = C_{N_2}^{r} \mathbb{E}[x_{c_1}^{\theta_1}] E[x_{c_2}^{\theta_2}] \cdot \ldots \cdot \mathbb{E}[x_{c_r}^{\theta_r}] \tag{3.47}$$

Вычисление моментов $\mathbb{E}[x_{c_i}^{\theta_i}]$ осуществляется по формулам 3.5 и 3.6.

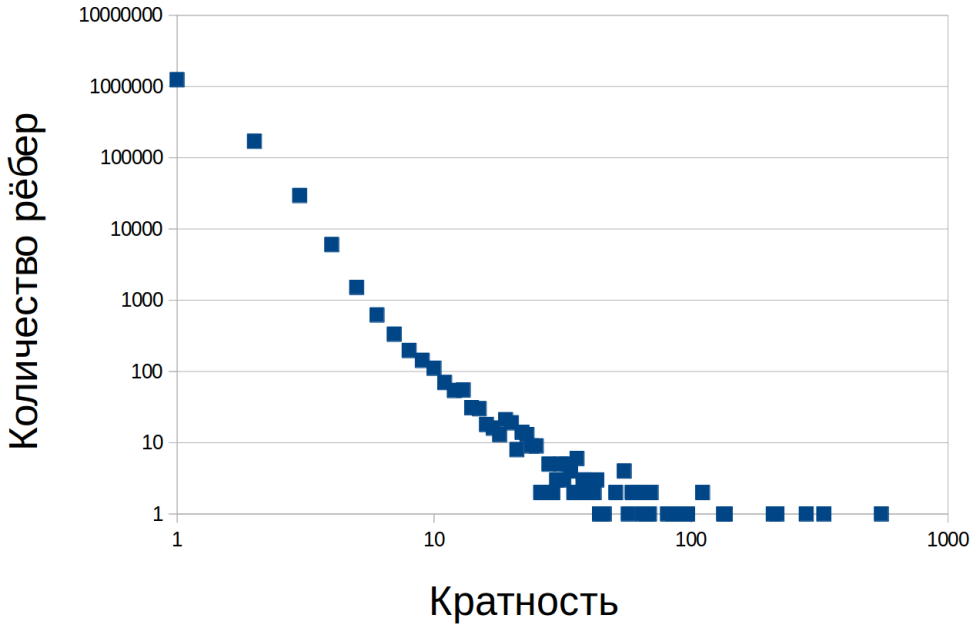


Рисунок 3.5: Распределение кратности генерируемых рёбер в социальном графе.

Для упрощения вычислений в уравнении 3.37 в реализованном прототипе рассматривается не более трёх слагаемых, так как из эмпирического анализа следует, что количество рёбер кратности более 3 незначительно по сравнению с общим количеством рёбер (рисунок 3.5).

3.5 Результаты экспериментов

В данном разделе приведены результаты экспериментального исследования программной реализации разработанного метода. Синтезированные шаблонные сети сравниваются с результатами работы популярного генератора LFR и реальными шаблонными сетями (раздел 1.3). Также оценивается производительность и масштабируемость программной реализации.

Кроме того, в разделе 4.6 представлены результаты исследования качества результатов различных методов определения структуры сообществ с помощью различных реальных и синтетических шаблонных сетей.

Все экспериментальные исследования предложенных в диссертационной работе методов выполнены с помощью разработанного в рамках работы инструмента `NetBlox`[4] для исследования кластерной структуры сложных сетей.

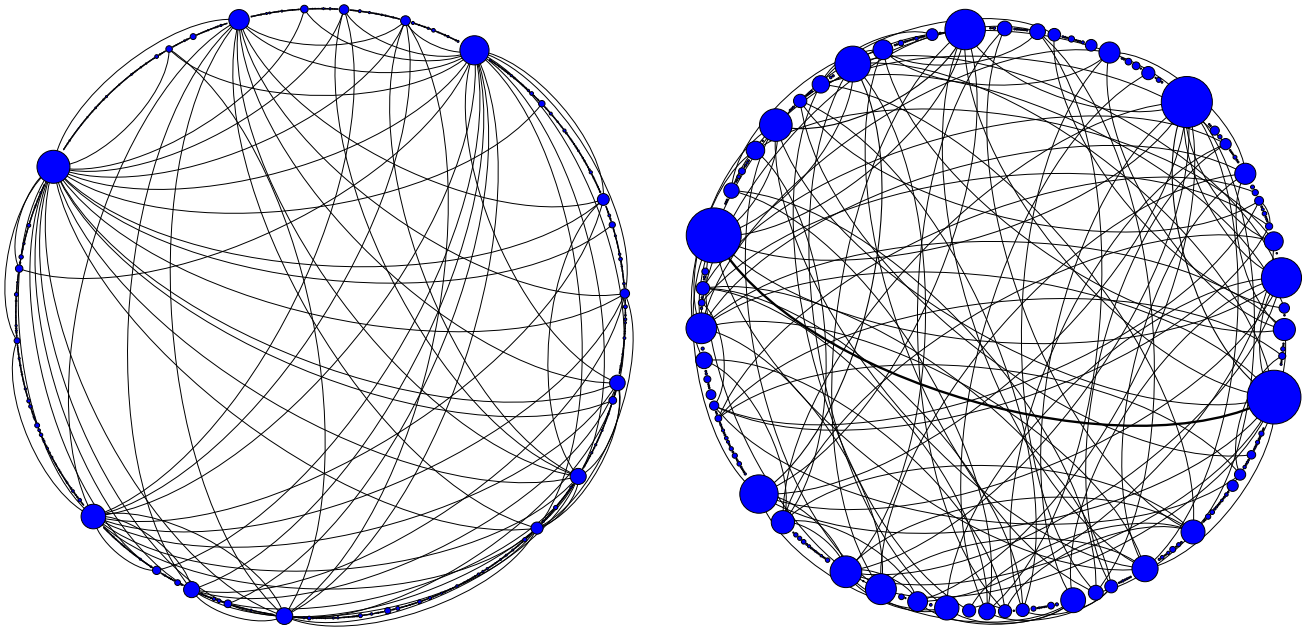


Рисунок 3.6: Графы пересечения сообществ CKB (слева) и LFR (справа).

3.5.1 Оценка свойств структуры сообществ

Таблица 3.4 содержит результаты сравнения сгенерированных графов с реальными шаблонными сетями на основе данных LiveJournal, ORKUT, Friendster и YouTube, а также с синтетическими сетями, созданными генератором LFR и разработанным методом CKB.

Используемые параметры LFR и их значения перечислены в таблице 3.3. Параметры CKB: $N_1 = 10,000$, $\beta_1 = 2.5$, $\beta_2 = 2.5$, $x_{min} = 3$, $x_{max} = 100$, $m_{min} = 1$, $m_{max} = 100$, $\gamma = 0.5$, средняя степень $= 50$, $\epsilon = 2N_1^{-1}$.

[4] https://github.com/ispras/NetBlox

Таблица 3.3: Параметры LFR.

Параметр	**Описание**	**По умолчанию**
N	Количество вершин	$10,000$
O_n	Количество вершин с количеством сообществ >1	$0.5N$
O_m	Количество сообществ у каждой из O_n вершин	4
k	Средняя степень вершины	50
μ_t	Топологический параметр смешивания	0.3
t_1	Экспонента степенного распределения степеней вершин	3.0
t_2	Экспонента степенного распределения размеров сообществ	2.5
c_{min}	Минимальный размер сообщества	3
c_{min}	Максимальный размер сообщества	100

На рисунке 3.6 изображены *графы пересечения сообществ* (англ. *community overlap graphs*) [33], где сообщества являются вершинами, а рёбра между ними существуют в случае наличия пересечения между множествами вершин пары сообществ мощностью не менее 10 вершин. Размер вершины пропорционален размеру сообщества, а вес ребра — мощности пересечения.

На рисунках 3.7-3.14 визуализированы структурные свойства сгенерированных LFR и СКВ графов с сообществами.

По результатам сравнения можно заключить, что среди сравниваемых синтетических шаблонных сетей СКВ имеют наиболее схожую структуру с реальными сетями. Все требуемые структурные свойств социального графа с сообществами (раздел 1.3) выполняются в синтезированных с помощью предложенного метода шаблонных сетях.

Отметим, что выполнение следующих свойств для СКВ-графов подтверждается экспериментально (рисунки 3.13 и 3.14):

- количество сообществ у пользователя прямо пропорционально количеству его связей с другими пользователями;
- средний коэффициент кластеризации вершин в сообществе обратно пропорционален размеру сообщества.

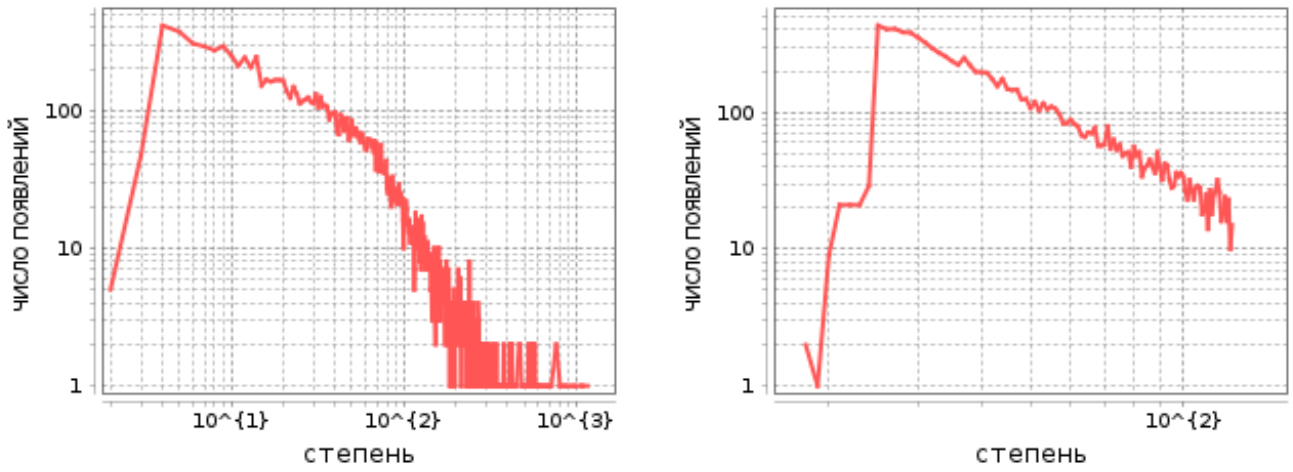


Рисунок 3.7: Распределение степеней вершин шаблонных сетей CKB (слева) и LFR (справа).

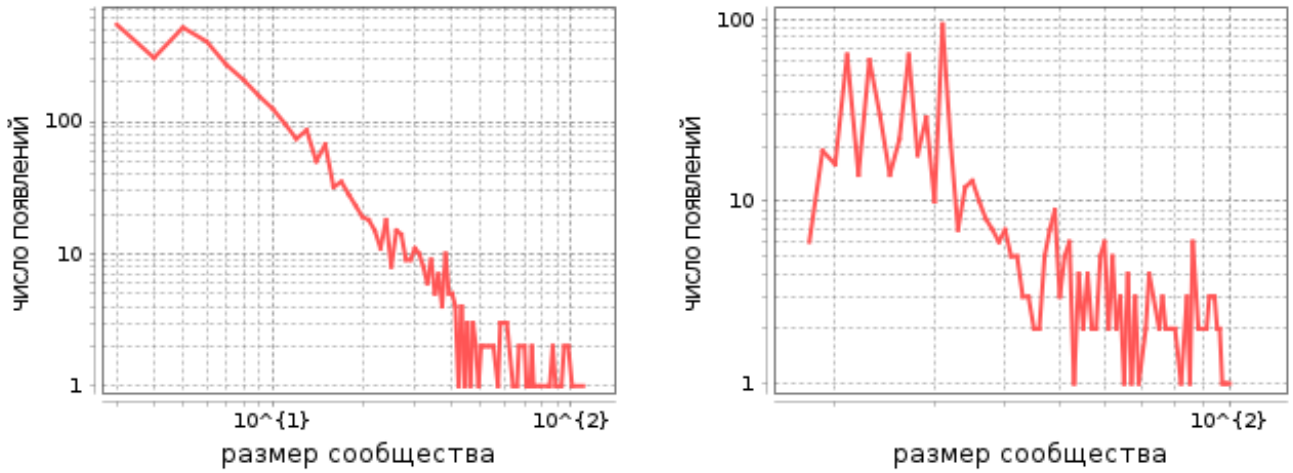


Рисунок 3.8: Распределение размеров сообществ шаблонных сетей CKB (слева) и LFR (справа).

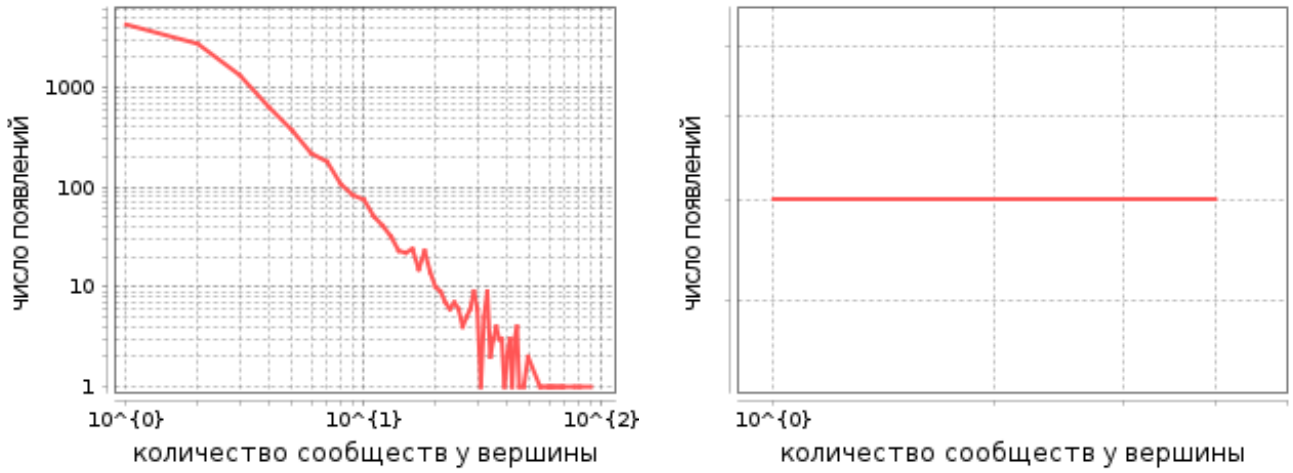


Рисунок 3.9: Распределение количества сообществ у пользователя в шаблонных сетях CKB (слева) и LFR (справа).

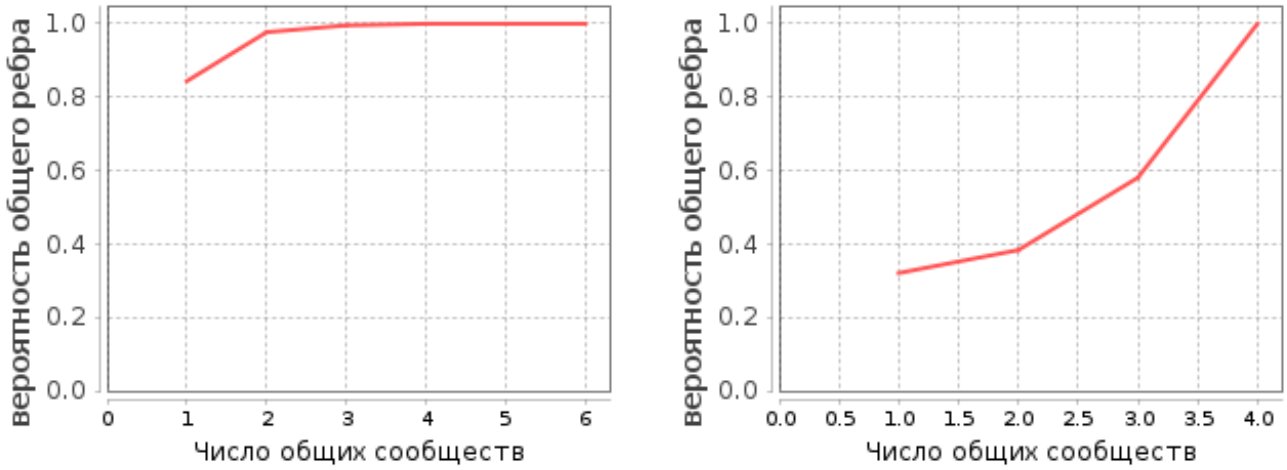


Рисунок 3.10: Зависимость вероятности ребра от количества общих сообществ его концевых вершин в шаблонных сетях CKB (слева) и LFR (справа).

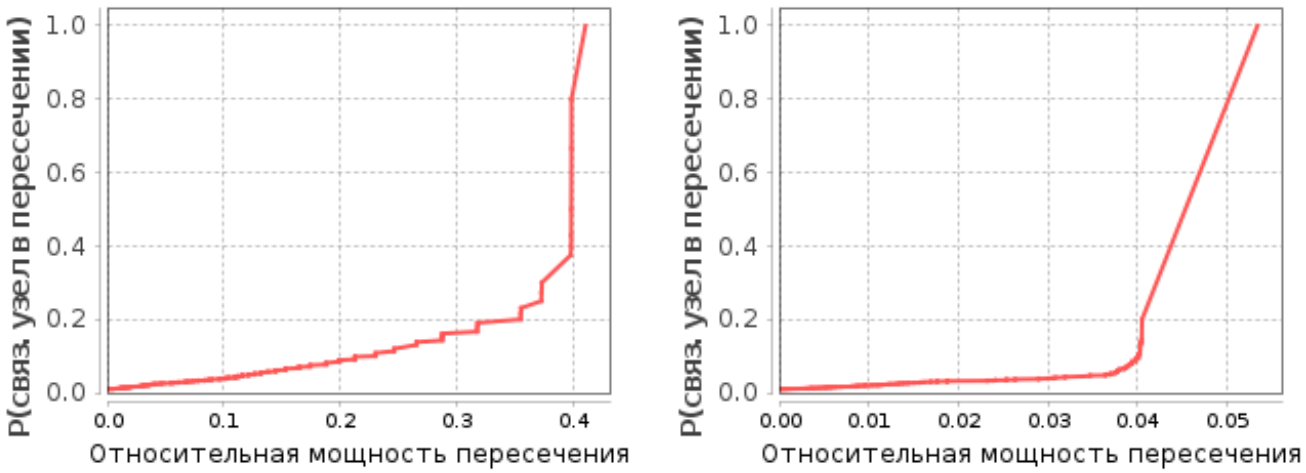


Рисунок 3.11: Зависимость вероятности появления связующей вершины в пересечении сообществ от относительной мощности пересечения в шаблонных сетях CKB (слева) и LFR (справа).

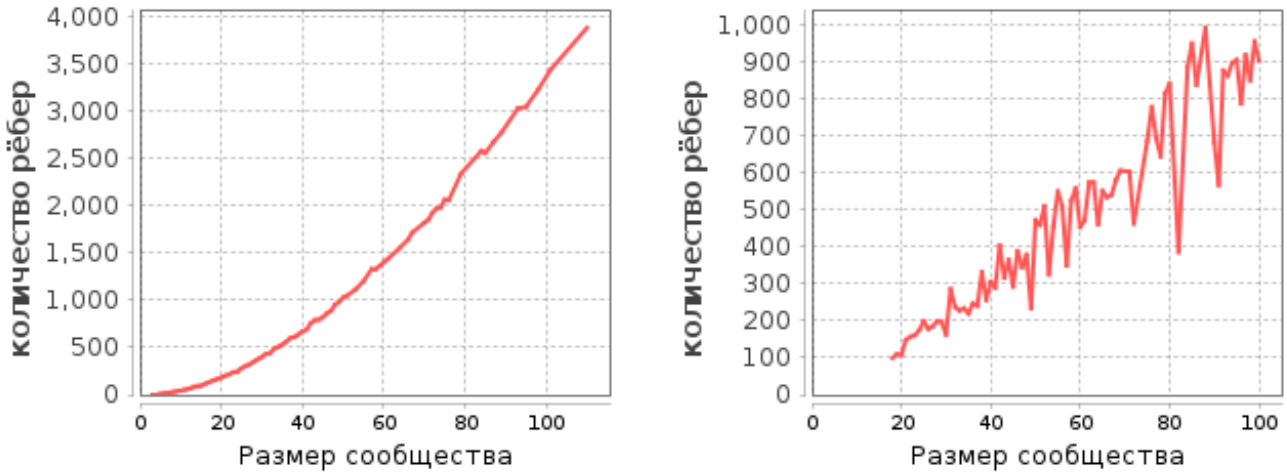


Рисунок 3.12: Зависимость количества рёбер в сообществе от его размера в шаблонных сетях CKB (слева) и LFR (справа).

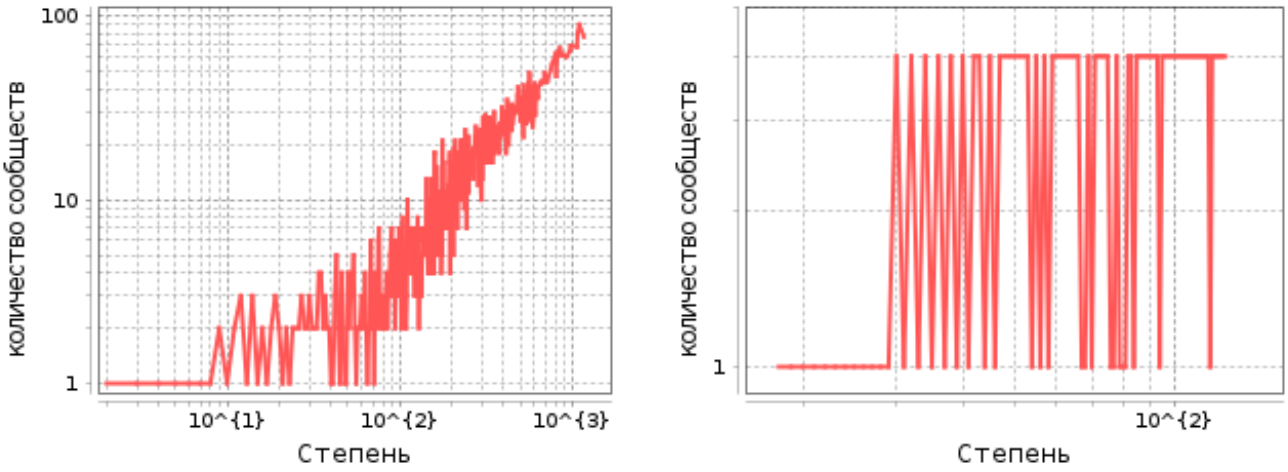


Рисунок 3.13: Зависимость количества сообществ у пользователя от его степени в шаблонных сетях CKB (слева) и LFR (справа).

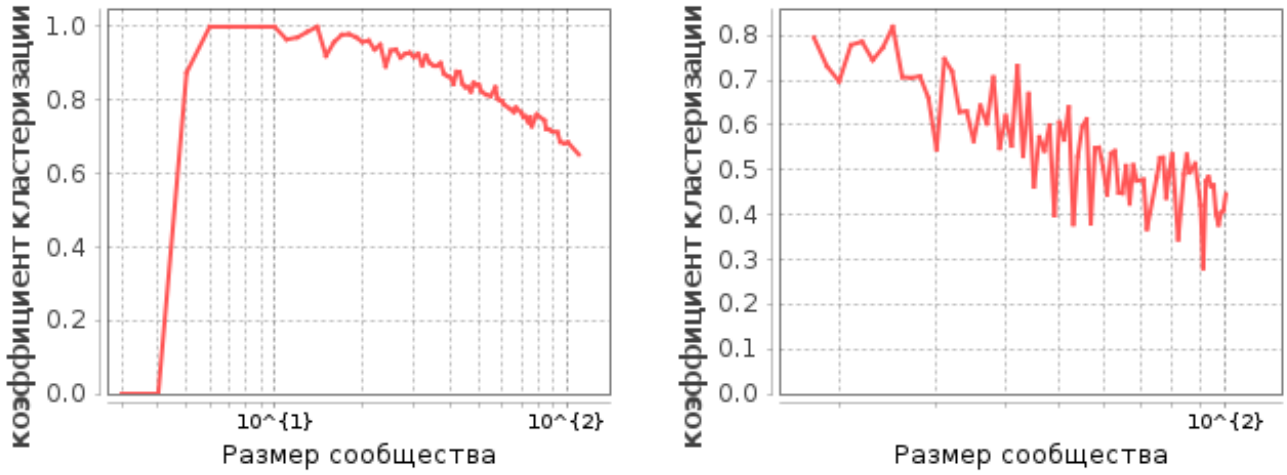


Рисунок 3.14: Зависимость среднего коэффициента кластеризации от размера сообщества в шаблонных сетях CKB (слева) и LFR (справа).

Таблица 3.4: Сравнение свойств шаблонных сетей CKB и LFR со свойствами графов Friendster, Orkut, LiveJournal и YouTube.

	Friendster	**Orkut**	**LiveJournal**	**YouTube**	**CKB**		**LFR**
Количество вершин	65,608,366	3,072,441	3,997,962	1,134,890	3M	97.5K	100K
Средняя степень	55.05	76.2	17.3	5.3	109.9	68.8	66.7
Количество сообществ	1,449,666	8,455,253	311,782	8385	62,862	875,836	2,258
Экспонента степенного распределения размеров сообществ	2.11	2.12	2.14	2.36	2.19	2.57	2.54
Экспонента степенного распределения принадлежности пользователей к сообществам	2.44	1.59	2.22	2.83	2.28	2.62	–
Экспонента степенного распределения степеней вершин	1.42	1.58	2.15	2.53	2.22	2.54	2.56
Медиана распределения размеров сообществ	2	16	2	3	5	49	40
Медиана распределения количества сообществ, в которых состоит пользователь	2	14	2	1	7	1	1
Средний коэффициент кластеризации	0.1623	0.169	0.353	0.172	0.2517	0.3826	0.226
Время генерации (с)	–	–	–	–	160	11	863

3.5.2 Оценка с помощью метрик качества

Синтезированные с помощью LFR и CKB сообщества были также исследованы с помощью описанных в разделе 1.4 метрик качества сообществ. Результаты представлены на рисунках 3.15, 3.16, 3.17, 3.18. Значения на оси ординат соответствуют кумулятивному скользящему среднему для всех сообществ, расположенных в порядке убывания значения метрики качества.

Анализ графиков показывает, что CKB-сообщества обладают в среднем более высокими показателями по всем метрикам качества сообществ. Вме-

сте с тем, минимальное значение отделимости для LFR-сообществ превышает аналогичный показатель метода СКВ. Данный факт можно объяснить большим количеством значительно пересекающихся сообществ в результатах работы предложенного метода. При этом становится сложнее провести чёткие “границы” между отдельными сообществами, что ведёт к ухудшению их отделимости в некоторых случаях. Это обуславливает низкую эффективность некоторых методов определения структуры сообществ, которые рассматривают их как хорошо отделимые плотные подграфы с незначительным пересечением множеств вершин (раздел 4.6).

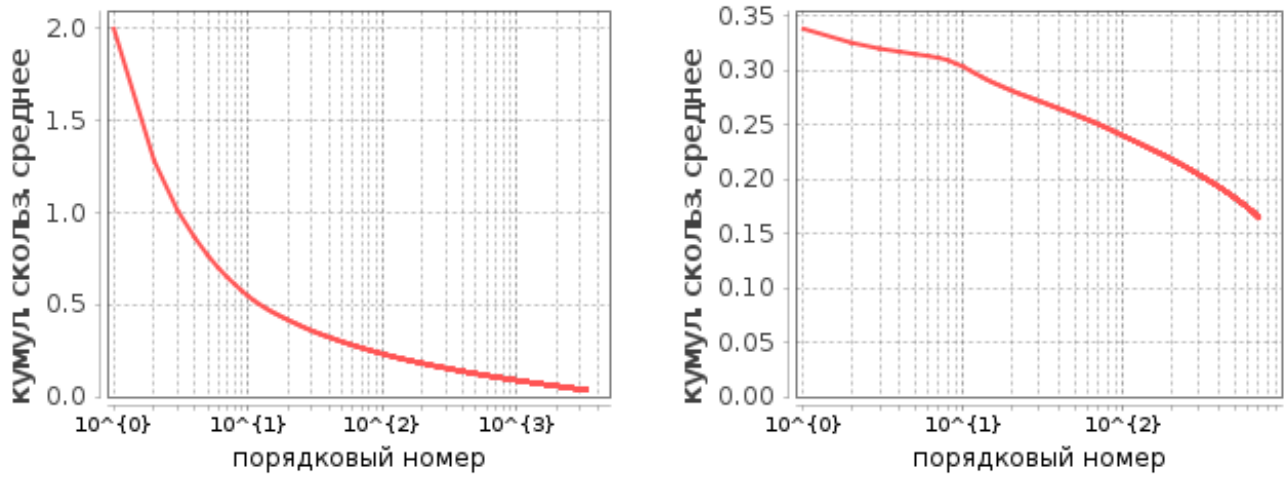


Рисунок 3.15: Отделимость сообществ шаблонных сетей СКВ (слева) и LFR (справа).

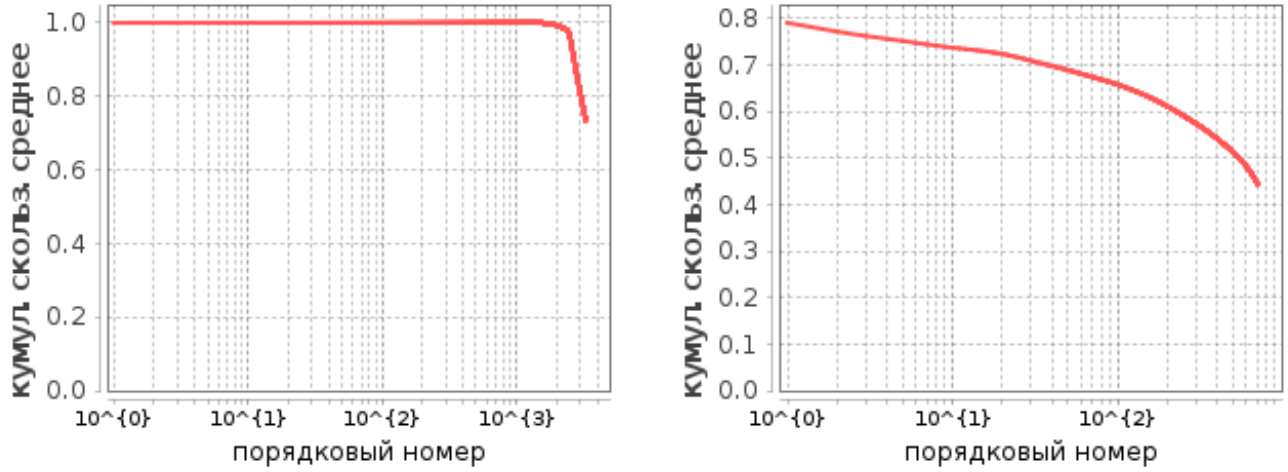


Рисунок 3.16: Плотность сообществ шаблонных сетей СКВ (слева) и LFR (справа).

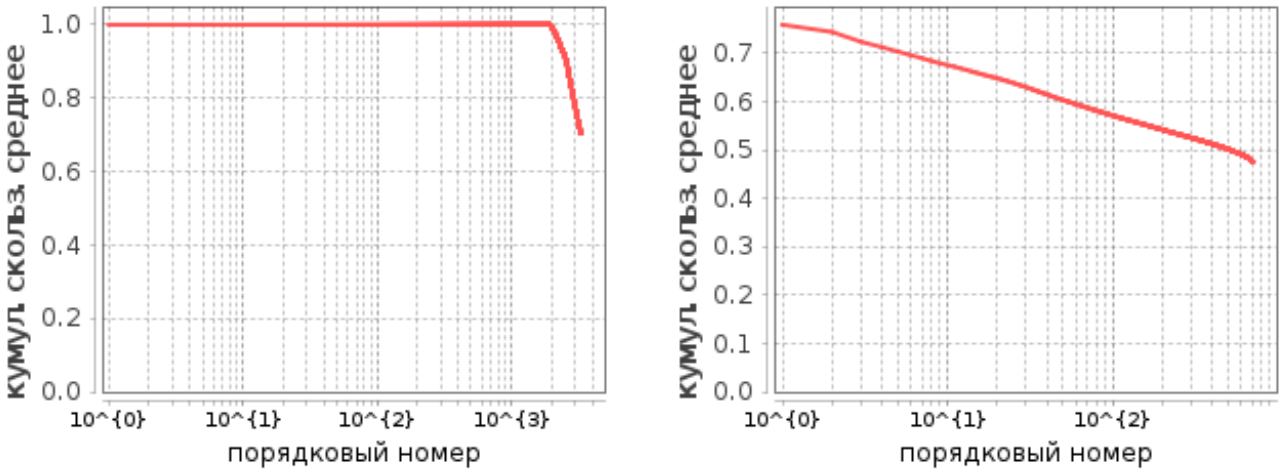


Рисунок 3.17: Сплочённость сообществ шаблонных сетей CKB (слева) и LFR (справа).

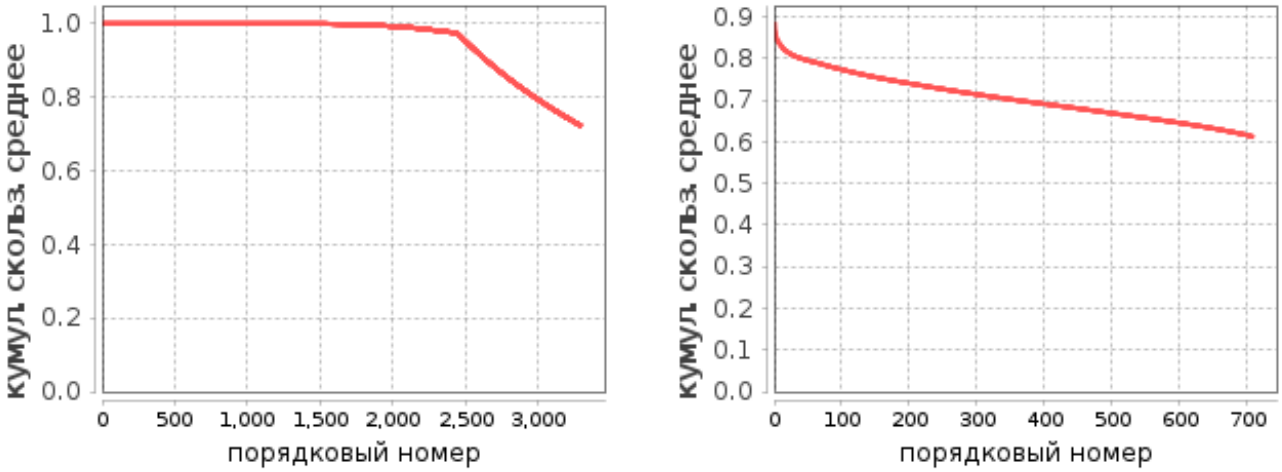


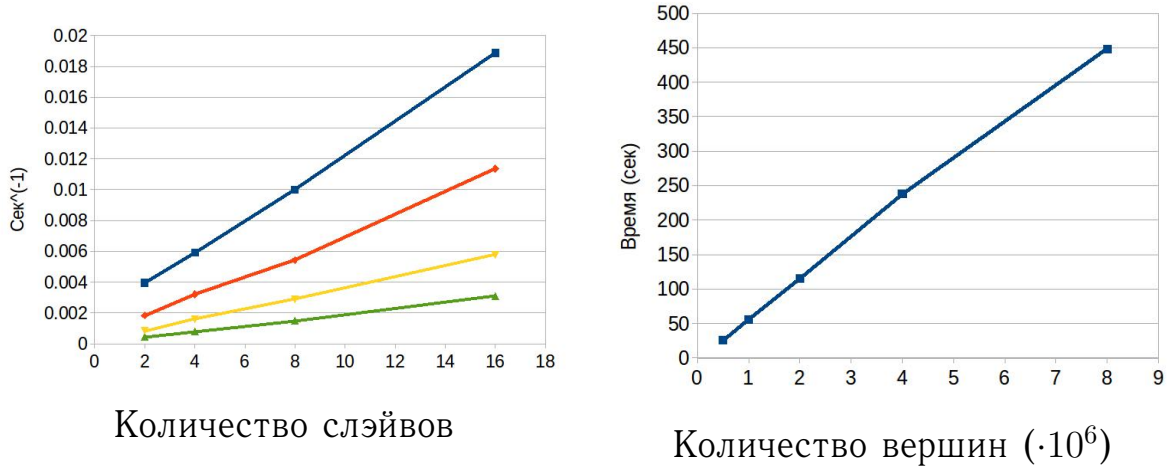
Рисунок 3.18: Коэффициент кластеризации вершин сообществ шаблонных сетей CKB (слева) и LFR (справа).

3.5.3 Производительность и масштабируемость

Для оценки масштабируемости алгоритма был использован кластер Amazon EC2 из 2, 4, 8 и 16 машин типа *m1.large*[5]. На рисунке 3.19 показано, что алгоритм имеет близкую к линейной масштабируемость, что озволяет создавать синтетические сети больших размеров за приемлемое время. Так, например, генерация графа с 1 миллиардом вершин заняла 150 минут на 150 машинах кластера Amazon EC2.

Однако в некоторых случаях может иметь место недостаточное ускорение работы генератора с ростом числа машин. Так, например, локально алгоритм отработал быстрее, чем на двух слэйвах. В данном случае это связано с

[5] http://aws.amazon.com/ec2/previous-generation/



Количество слэйвов

Количество вершин ($\cdot 10^6$)

Рисунок 3.19: **Слева**: масштабируемость на кластере Amazon EC2 из машин типа *m1.large*. Синяя линия — граф с $4 \cdot 10^6$ вершин, красная линия — $8 \cdot 10^6$ вершин, жёлтая линия — $16 \cdot 10^6$ вершин, зелёная линия — $32 \cdot 10^6$ вершин. **Справа**: время работы.

тем, что на пересылку данных по сети требуется не меньше времени, чем собственно на процесс генерации.

3.6 Выводы

В ходе работы был разработан, реализован и исследован метод `СКВ` для распределённой генерации больших шаблонных сетей с реалистичными свойствами социальных графов и структурой сообществ, учитывающей некоторые из недавно доказанных свойств модульной структуры социальных сетей. Реализация модели с помощью Apache Spark позволяет осуществлять распределённую генерацию случайных социальных графов из сотен миллионов вершин с различными наборами параметров.

Возможные направления дальнейшей работы включают:

- распределённое вычисление метрик для сравнения покрытий сообществ;
- генерация иерархических, ориентированных и взвешенных сетей;
- генерация атрибутов пользователей с поддержкой гомифилии в сообществах.

Глава 4

Распределённый метод определения структуры сообществ в социальном графе

Немногочисленные масштабируемые методы определения структуры сообществ либо обладают значительной вычислительной сложностью, либо неспособны находить пересекающиеся сообщества, либо имеют тенденцию к значительному ухудшению качества с увеличением количества сообществ у пользователей.

В данной главе приведено описание разработанного метода `EgoLP` для определения структуры сообществ пользователей в социальном графе.

Используемые в алгоритме подходы и эвристики в значительной степени мотивированы результатами экспериментального исследования простейшего алгоритма распространения меток (раздел 2.1.3). Требовалось устранить обнаруженные недостатки, повысить качество и в то же время не ухудшить масштабируемость и вычислительную сложность.

Основные результаты главы опубликованы в работе [27].

4.1 Постановка задачи

Требуется разработать метод определения структуры сообществ пользователей в графах онлайновых социальных сетей. Метод должен удовлетворять следующим критериям:

1) высокое качество восстановления заранее известной структуры сообществ;

2) высокая точность решения прикладной задачи с использованием информации о сообществах;

3) вычислительная сложность, не превышающая линейную по числу рёбер графа;

4) близкая к линейной масштабируемость;

5) способность обрабатывать социальные графы с $> 10^6$ вершин и средней степенью > 100, характерной для социальных сетей.

4.2 Общая схема метода

В основе метода лежит гипотеза о взаимосвязи мезо- и микроскопического уровней организации социальной сети (раздел 1.1), на которой основаны некоторые методы определения структуры сообществ (раздел 2.1.4). Суть гипотезы состоит в том, что эго-сообщества каждого пользователя значительно пересекаются с сообществами социального графа, в которых состоит данный пользователь (рисунок 4.1). Иными словами, объединение эго-сообществ различных пользователей ведёт к формированию глобальных сообществ, что можно рассматривать как одну из причин образования связей в социальном графе. Данная гипотеза не была проверена экспериментально в рамках диссертационной работы, однако эффективность разработанного метода в применении к реальным и синтетическим данным косвенно подтверждает её справедливость.

`EgoLP` состоит из трёх основных этапов: определение структуры эго-сообществ каждого пользователя, определение структуры сообществ путём распространения меток сообществ, определение подсообществ.

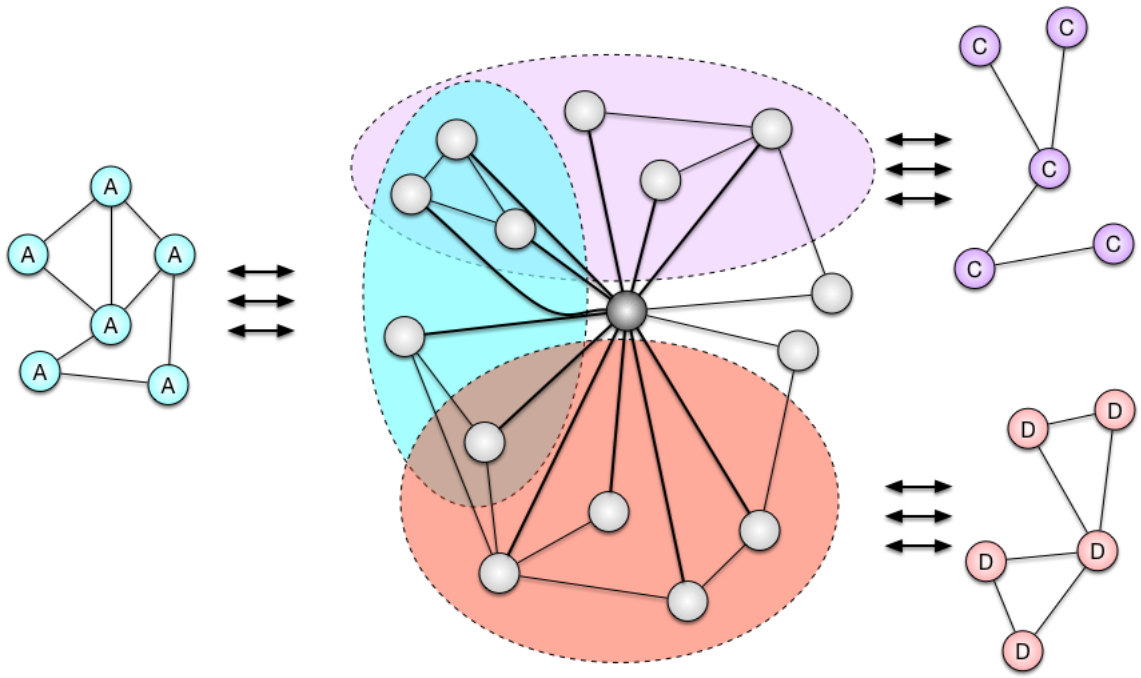


Рисунок 4.1: Связь эго-сообществ с глобальными сообществами.

На этапе предварительной обработки граф в виде списка рёбер загружается с диска, после чего из него удаляется определенная часть вершин с максимальными степенями (*хабы* от англ. *hub nodes*). По умолчанию удаляются все вершины со степенью $d_v > 2000$, но не менее 0.1% всех вершин с максимальными степенями. Эта эвристика упрощает обработку больших графов засчёт значительного уменьшения количества рёбер и снижения нагрузки на сеть и процессоры, а также исключает избыточное распространение меток хабов, предотвращая формирование очень больших сообществ.

Временная сложность данного этапа составляет $O(|E|)$, что соответствует времени, необходимому для загрузки входного графа с диска.

В конце этапа распространения меток удалённые узлы возвращаются в граф и присоединяются к сообществам.

Предложенный метод был реализован на языке программирования Scala с использованием Apache Spark - фреймворка для распределённых вычислений в распределённой среде (раздел 3.2). Основная часть реализации EgoLP основана на вычислительной парадигме Pregel [45], позволяющей оптимизировать время обработки графовых данных.

В основе Pregel лежит модель *блочно-синхронного параллелизма* (англ. *bulk-synchronous parallelism, BSP*) [73]. Вычислительный процесс BSP представляет из себя последовательность блоков асинхронных локальных вычис-

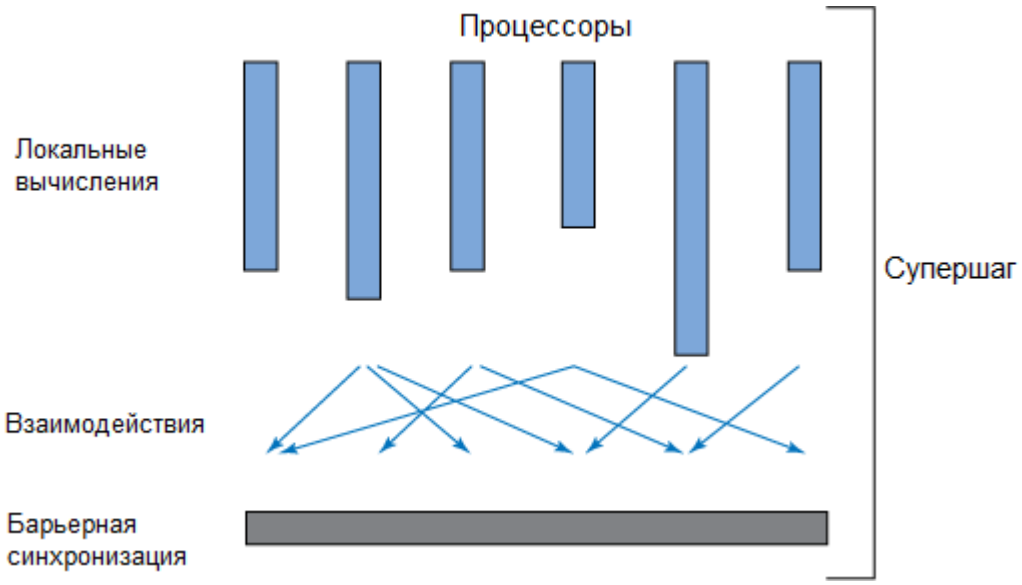


Рисунок 4.2: Модель блочно-синхронного параллелизма.

лений, чередующихся с блоками коммуникации и барьерной синхронизации (рисунок 4.2).

В модели Pregel на каждом супершаге каждая вершина выполняет все или некоторые из перечисленных элементарных операций:

- принимает сообщения, отправленные на предыдущем супершаге;
- выполняет заданную пользователем функцию;
- изменяет своё состояние или состояние исходящих рёбер;
- отправляет сообщения другим вершинам;
- изменяет топологию графа;
- посылает запрос на останов, если работы больше нет.

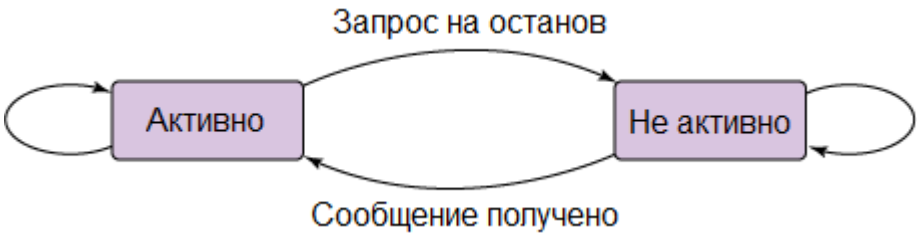


Рисунок 4.3: Процесс “голосования” вершин за останов в модели Pregel.

Условие останова (рисунок 4.3):

- все вершины одновременно неактивны;
- нет недоставленных сообщений.

Таким образом, алгоритмы распространения меток (алгоритм 1) естественным образом выражаются в виде последовательностей элементарных операций Pregel, что позволило реализовать разработанный метод в рамках данной модели на основе фреймворка Apache Spark.

4.3 Определение структуры эго-сообществ

На этом этапе для каждого узла графа строится эго-сеть и выполняется алгоритм 2 для определения структуры эго-сообществ (рисунок 4.4). Параметры алгоритма приведены в таблице 4.1.

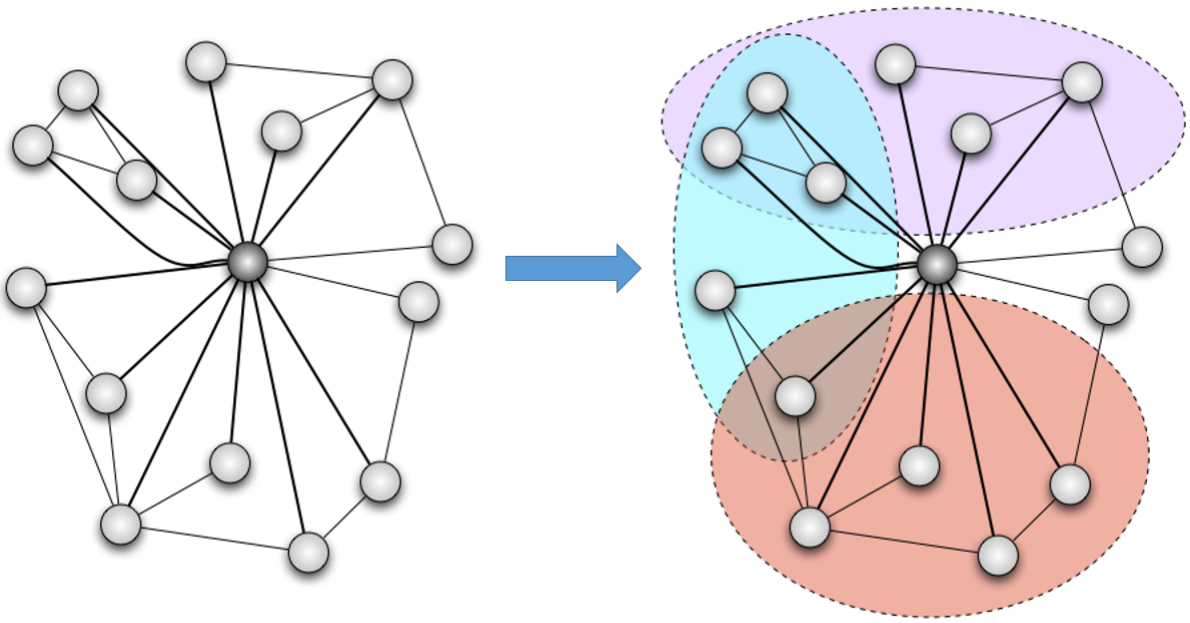


Рисунок 4.4: Определение структуры эго-сообществ в эго-сети.

После завершения обмена метками эго-сообщества размера $< minc$ удаляются исходя из предположения о том, что небольшие эго-сообщества конкретного пользователя слабо связаны с глобальной структурой сообществ в социальном графе.

Если доля соседей эго-вершины, отнесённых к эго-сообществам, не превышает порог $minu$, то считается, что структура эго-сообществ была опреде-

Исходные данные: Граф $G(V, E)$
Результат: Эго-сети $\{Ego_v(v, V_v^{ego}, E_v^{ego})\}$,
эго-сообщества $\{\mathcal{E}_{vk}(V_{vk}, E_{vk})\},\ V_{vk} \subseteq V_v^{ego}$.

для каждого $v \in V$ **выполнить**
  для каждого $n \in \{$*соседи* $v\}$ **выполнить**
    отправить n список соседей v;
    инициализировать v уникальной меткой сообщества;
  конец цикла
конец цикла
для каждого $v \in V$ **выполнить**
  построить эго-сеть $Ego_v(v, V_v^{ego}, E_v^{ego})$ вершины v;
  для каждого $i =$ *1:Te* **выполнить**
    для каждого $ev \in V_v^{ego}$ **выполнить**
      для каждого $n \in \{$*соседи* ev *из* $V_v^{ego}\}$ **выполнить**
        случайным образом выбрать k меток из множества меток вершины ev;
        отправить выбранные метки вершине n;
      конец цикла
    конец цикла
    для каждого $ev \in V_v^{ego}$ **выполнить**
      $\mathcal{L}_{vi}$:= метки, полученные вершиной ev от соседей;
      выбрать k наиболее частых меток из $\mathcal{L}_{vi}$;
      добавить выбранные метки к множеству меток вершины ev;
    конец цикла
  конец цикла
  для каждого $ev \in V_v^{ego}$ **выполнить**
    удалить из множества меток ev все метки с частотой $< r$;
  конец цикла
  преобразовать метки всех вершин в $\{\mathcal{E}_{vk}\}$;
  удалить из $\{\mathcal{E}_{vk}\}$ эго-сообщества размером $\leq minc$;
  если $\frac{|\bigcup_k V_{vk}|}{|V_v^{ego}|} < minu$ **тогда**
    $\{\mathcal{E}_{vk}\} := \{V_v^{ego}\}$;
  конец условия
конец цикла

Алгоритм 2: Определение структуры эго-сообществ в эго-сетях.

Таблица 4.1: Параметры алгоритма определения структуры эго-сообществ.

Параметр	**Описание**	**По умолчанию**
Te	Количество итераций	15
r	Пороговое значение частоты метки	0.08
$minc$	Минимальный размер эго-сообщества пользователя в сообществе	5
$minu$	Минимальный относительный размер объединения эго-сообществ	0.8

лена некорректно:

$$\frac{|\bigcup_k V_{vk}|}{|V_v^{ego}|} < minu. \tag{4.1}$$

В этом случае считается, что у вершины есть единственное эго-сообщество, состоящее из всех её соседей V_v^{ego}.

Временная сложность пересылки списка соседей вершин и формирования эго-сетей $O(d_{mean} \cdot |E|)$, а сложность пересылки меток для определения структуры эго-сообществ $O(Te \cdot |E|)$. Общая временная сложность данного этапа $O((d_{mean} + Te) \cdot |E|)$, где d_{mean} соответствует средней степени вершины входного графа.

Важно отметить, что полученные эго-сообщества для пользователей социальных сетей являются ценными сами по себе. В частности, они могут быть использованы в любом приложении, которое опирается на данные о социальных кругах среди контактов целевого пользователя (например, в качестве замены ручной группировки контактов в Facebook и Google+ для оптимизации потоков информации на персональных страницах пользователей).

В частности, в работе [24] автором демонстрируется применимость информации об эго-сообществах пользователя для решения задачи рекомендации получателей электронных сообщений[1].

4.4 Распространение меток сообществ

Следующим этапом является итеративное распространение меток сообществ для определения глобальной структуры сообществ входного графа в

[1]Веб-демонстрация разработанного прототипа доступна по адресу: http://rs.at.ispras.ru/

соответствии с общим шаблоном алгоритмов этого класса (раздел 2.1.3). На каждой итерации каждый узел поочерёдно выполняет роль “говорящего” и “слушающего узла” согласно заданным стратегиям взаимодействия узлов (алгоритм 3):

- **стратегия “говорящего узла”**: для каждого смежного ребра “говорящий узел” случайно выбирает ls элементов из своего текущего набора меток сообществ и удаляет дубликаты, после чего отправляет метки соседней вершине. Таким образом, узел с большим количеством сообществ посылает в среднем больше меток;

- **стратегия “слушающего узла”**: входящие метки распределяются между эго-сообществами в соответствии с индексами отправивших их “говорящих узлов”. Затем в каждом эго-сообществе выбираются lr наиболее частых меток и добавляются к множеству меток “слушающего узла”. Размер множества ограничен mx.

Таким образом, входящие метки сообществ для узла агрегируются не по всему множеству его соседей, а по каждому эго-сообществу. В результате в память “слушающего” узла добавляются высокочастотные метки из каждого эго-сообщества (рисунок 4.5).

Предложенная стратегия “слушающего узла”, основанная на эго-сообществах, обеспечивает следующие преимущества по сравнению с другими методами на основе распространения меток (раздел 2.1.3):

- более точно оценивается число меток, которые необходимо принять и добавить в массив меток “слушающего узла” (частично решает проблемы недостатка сообществ и наличия шумовых меток);

- наиболее популярные метки рассчитываются независимо в разных эго-сообществах, что позволяет выявить сообщества с различной силой связи с узлом (частично решает проблему доминирующего сообщества).

Временная сложность данного этапа зависит, прежде всего, от времени пересылки меток по рёбрам графа и составляет $O((T + T_2) \cdot |E|)$.

Исходные данные: Граф $G(V, E)$, эго-сообщества $\{\mathcal{E}_{vk}\}$
Результат: Покрытие сообществ $\mathbb{C} = \{Z_c(V_c, E_c)\},\ V_c \subseteq V$

для каждого $v \in V$ **выполнить**
  инициализировать v уникальной меткой сообщества;
конец цикла
для каждого $i = 1{:}T$ **выполнить**
  для каждого $v \in V$ **выполнить**
    для каждого $n \in \{\textit{соседи } v\}$ **выполнить**
      случайным образом выбрать ls меток из множества меток вершины v;
      отправить выбранные метки вершине n;
    конец цикла
  конец цикла
  для каждого $v \in V$ **выполнить**
    $\mathcal{L}_{vi}$:= метки, полученные вершиной v от соседей;
    удалить из $\mathcal{L}_{vi}$ все метки, кроме lx наиболее частых;
    для каждого $\mathcal{E}_{vk} \in \{\mathcal{E}_{vk}\}$ **выполнить**
      $\mathcal{L}_{vik}$:= метки из $\mathcal{L}_{vik}$, полученные вершиной v от соседей из $\mathcal{E}_{vk}$;
      выбрать lr наиболее частых меток из $\mathcal{L}_{vik}$;
      добавить выбранные метки к множеству меток вершины v;
    конец цикла
    удалить из множества меток вершины v все метки, кроме mx наиболее частых;
  конец цикла
конец цикла
$V := V \cup \{\text{удалённые хабы}\}$;
для каждого $v \in V$ **выполнить**
  $\{\mathcal{E}_{vk}\} := \{V_v^{ego}\}$;
конец цикла
для каждого $i = 1{:}T_2$ **выполнить**
  выполнить обмен метками аналогично циклу с T итераций;
конец цикла
удалить из множества меток v все метки с частотой $< r$;
преобразовать метки всех вершин в $\{Z_c\}$.

Алгоритм 3: Распространение меток сообществ.

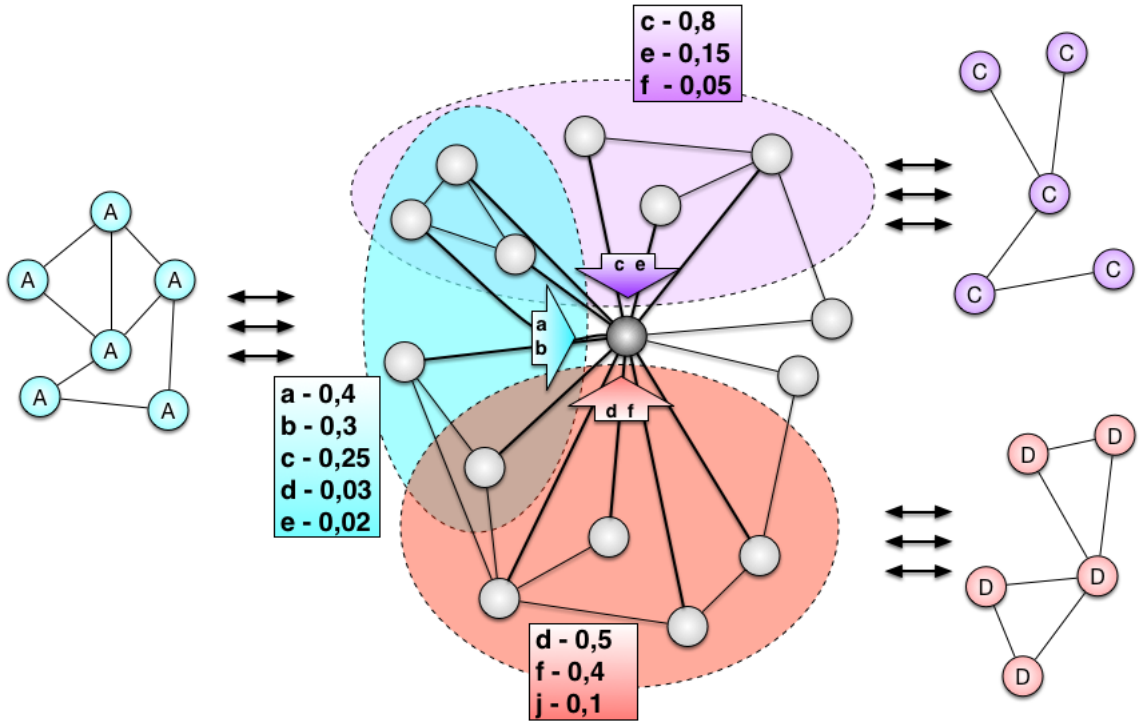


Рисунок 4.5: Распространение меток с использованием эго-сообществ. “Слушающий” узел расположен в центре и получает метки сообществ от “говорящих” узлов — всех остальных вершин своей эго-сети. На каждой итерации алгоритма каждый “слушающий” узел выбирает несколько наиболее популярных меток из каждого сообщества окружающих его “говорящих” узлов. Выбранные на данной итерации метки указывают на связь “слушающего” узла с сообществами A, C и D.

Таблица 4.2: Параметры этапа распространения меток.

Параметр	**Описание**	**По умолчанию**
T	Количество итераций	25
ls	Количество меток, отправляемых каждой вершиной соседу	5
lx	Количество меток, принимаемых каждой вершиной от соседей	20
lr	Количество меток, принимаемых каждой вершиной от каждого эго-сообщества	2
mx	Максимальное количество меток, хранящихся в памяти вершины	20
T_2	Количество дополнительных итераций после присоединения хабов	2
r	Пороговое значение частоты метки	0.06

4.5 Определение подсообществ

Большинство методов на основе распространения меток не обеспечивают связность сообществ (раздел 2.1.3). Для решения этой проблемы в качестве завершающего этапа EgoLP производится выявление возможных подсообществ в каждом сообществе, полученном на этапе распространения меток (алгоритм 4). Параметры алгоритма приведены в таблице 4.3.

Исходные данные: Граф (V, E), покрытие сообществ $\mathbb{C} = \{Z_c(V_c, E_c)\}$
Результат: Покрытие сообществ
$\mathbb{C}^{sub} = \{Z_i^{sub}(V_i^{sub}, E_i^{sub})\},\ \forall i\ \exists c : V_i^{sub} \subseteq V_c$

для каждого $Z_c \in \mathbb{C}$ **выполнить**
  если $|V_c| \leq cx$ **тогда**
    найти λ_{n-1} матрицы Лапласа L_c сообщества Z_c;
    если $\lambda_{n-1} < \lambda_x$ **тогда**
      $\mathbb{C}_c^{sub}$:= сообщества, найденные в Z_c алгоритмом SLPA(T, r, k);
      удалить из $\mathbb{C}_c^{sub}$ сообщества размером $\leq minc$;
      для каждого $Z_{c_j}^{sub} \in \mathbb{C}_c^{sub}$ **выполнить**
        $\mathbb{C}^{sub} := \mathbb{C}^{sub} \cup \{$ компоненты связности $Z_{c_j}^{sub}\}$;
      конец цикла
    иначе
      $\mathbb{C}^{sub} := \mathbb{C}^{sub} \cup\ Z_c$;
    конец условия
  конец условия
конец цикла

Алгоритм 4: Определение подсообществ.

Рассматривается матрица Лапласа L_c для каждого сообщества:

$$L_c = A_c - diag(\vec{d}_c), \tag{4.2}$$

где A_c – матрица смежности, $\vec{d}_c$ – степени узлов.

Второе наименьшее собственное значение L (также известное как *алгебраическая связность* λ_{n-1}) позволяет оценить внутреннюю связность сообщества [74]: нулевое значение означает наличие нескольких компонент связности, тогда как низкие значения ($< \lambda_x$) указывают на низкую связность. Предполагается, что для таких сообществ низкая связность обуславливается наличием подсообществ, которые не были разделены на предыдущих этапах.

Таблица 4.3: Параметры этапа определения подсообществ.

Параметр	**Описание**	**По умолчанию**
cx	Максимальный размер сообщества	10 000
λ_x	Пороговое значение алгебраической связности	10
T	Количество итераций	15
k	Количество меток, отправляемых и принимаемых каждой вершиной на каждой итерации	3
r	Пороговое значение частоты метки	0.15
$minc$	Минимальный размер подсообщества	5

С целью уточнения результатов на данном этапе к каждому сообществу с низкой связностью применяется модификация алгоритма SLPA с дополнительным параметром k, который регулирует количество меток, отправляемых и получаемых каждой вершиной на каждой итерации.

Наименьшие собственные значения рассчитываются приближённо за счёт численного решения линейного уравнения с помощью *локально-оптимального блочного метода сопряженных градиентов* (англ. *Locally Optimal Block Preconditioned Conjugate Gradient, LOBPCG*) [75]. Сложность используемого алгоритма $O(|E_c|)$.

Отметим, что сообщества размером $> cx$ удаляются из результатов работы алгоритма и не анализируются на данном этапе.

Временная сложность последнего этапа складывается, прежде всего, из $O(|E|)$ для вычисления λ_{n-1} во всех сообществах, а также $O(s \cdot T \cdot |E_{c_i}|)$ для определения подсообществ в каждом из s сообществ с $\lambda_{n-1} < \lambda_x$ с помощью SLPA(T, r, k). Итоговая сложность $O(s \cdot T \cdot |E|)$.

Поскольку количество обрабатываемых на последнем этапе сообществ, как правило, невелико, а количество итераций на каждом этапе алгоритма не превышает 30, то асимптотическая сложность метода определяется этапом определения структуры эго-сообществ и составляет $O(d_{mean} \cdot |E|)$. Однако максимальная степень вершины в графе d_{max} ограничена параметром метода, что позволяет при $|E| \to \infty$ в случае обработки больших графов пренебречь и средней степенью вершины. Таким образом, общая сложность метода равна $O(|E|)$.

4.6 Результаты экспериментов

Данный раздел содержит результаты экспериментального исследования качества результатов и производительности EgoLP с использованием различных синтетических и реальных данных. Кроме того, приводятся результаты сравнения EgoLP с другими методами определения структуры сообществ (раздел 2.1).

Для сравнения были выбраны следующие алгоритмы: OSLOM [14][2] и GCE [40][3] из класса методов локальной оптимизации, SLPA [20][4] из класса методов распространения меток, MOSES [42][5] из класса методов, основанных на вероятностных моделях. Каждый из методов является одним из наиболее популярных в своём классе и обеспечивает относительно высокую точность определения структуры сообществ.

Все эксперименты проводились с использованием шаблонных сетей, сгенерированных методами CKB и LFR с параметрами, указанными в разделе 3.5.

4.6.1 Восстановление известной структуры сообществ

Основным способом оценки качества методов в данной работе является использование реальных и синтетических графов с известным покрытием, которые затем сравниваются с результатами работы методов.

Алгоритмы получали на вход список рёбер и возвращали найденное покрытие. Для того, чтобы установить близость известного и найденного покрытий, используется NMI — *нормализованная взаимная информация* (раздел 2.2.1).

Рисунки 4.6 и 4.7 содержат сводку результатов для графов, синтезированных генераторами LFR (раздел 2.2.1) и CKB (глава 3). В случае CKB-графов была исследована зависимость NMI от параметра β_1 — экспоненты степенного распределения количества сообществ у пользователя. Уменьшение β_1 при неизменных значениях остальных параметров приводит к увеличению среднего числа сообществ у пользователя, что делает структуру сообществ более

[2] http://www.oslom.org/
[3] https://sites.google.com/site/greedycliqueexpansion/
[4] https://sites.google.com/site/communitydetectionslpa/
[5] http://www.cliquecluster.org/moses

сложной для определения. Для LFR-графов увеличение пересечения сообществ достигается за счёт увеличения параметра O_m, который регулирует количество сообществ, к которым относится каждая из O_n вершин. Отметим, что для недетерминированных методов EgoLP, SLPA, OSLOM выполнялись 3 запуска на каждом из графов с последующим усреднением.

Из экспериментальных данных следует, что на LFR-графах при $O_m \geq 4$ EgoLP показывает устойчиво более высокие значения NMI по сравнению со всеми остальными методами. На СКВ-графах разработанный метод уступает только методу MOSES. Для всех методов характерно снижение NMI с увеличением среднего количества сообществ у вершины.

Несмотря на то, что методы OSLOM, GCE и MOSES в некоторых случаях показывают лучшие результаты, чем EgoLP, они неспособны обрабатывать графы с $> 10^6$ вершин (раздел 4.6.5). Единственный метод SLPA, который наряду с EgoLP позволяет эффективную распределённую реализацию, показывает значения NMI, которые ниже аналогичных значений EgoLP на > 0.2 в большинстве экспериментов.

Кроме того, методами EgoLP и SLPA был проанализирован граф LiveJournal с референтными сообществами. Значение NMI составило 0.29 для SLPA и 0.35 для EgoLP.

Вместе с тем, остаётся открытым вопрос о том, является ли определяемая структура сообществ достаточно качественной для решения реальной практической задачи. Данный вопрос исследуется в следующем разделе.

4.6.2 Определение атрибутов пользователей

В данном разделе содержатся результаты сравнения методов с помощью предложенного Lee et al метода косвенного оценивания качества структуры сообществ с помощью задачи определения скрытых атрибутов пользователей [76].

Для эксперимента использовался набор данных `Facebook100` — коллекция социальных сетей учащихся 100 американских колледжей и университетов. Все сети являются неориентированными и содержат указанные пользователями значения атрибутов: факультет, пол, профилирующая дисциплина, общежитие, год выпуска, школа. Согласно результатам исследований Traud

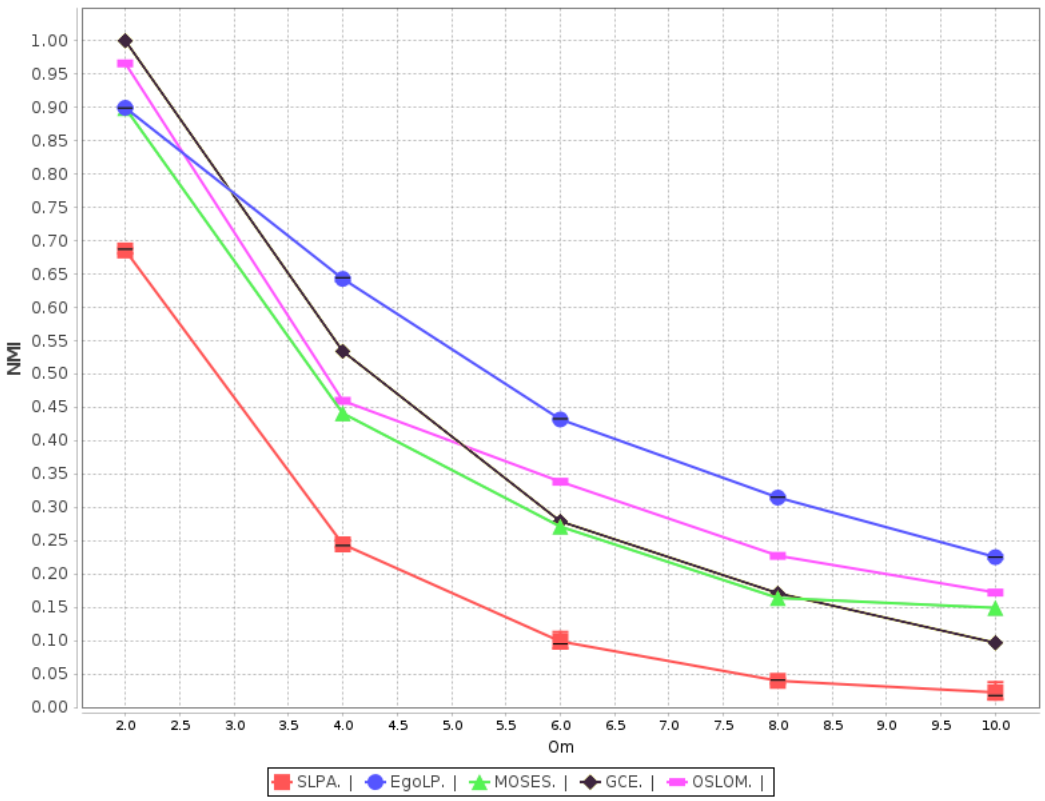


Рисунок 4.6: Сравнение качества EgoLP и других методов с помощью LFR-графов с различными значениями O_m — числа сообществ, к которым принадлежит каждая из $O_n = 0.5N$ вершин.

et al [77], год выпуска и общежитие в наибольшей степени ассоциированы со структурой сообществ.

Метки сообществ для каждого пользователя, найденные с помощью различных методов определения структуры сообществ, были использованы в качестве векторов признаков для обучения классификатора по методу *градиентного бустинга* [78][6]. Качество оценивалось по точности определения года выпуска и общежития для студентов в тестовых подвыборках. В этом приложении MOSES демонстрирует лучший результат, а EgoLP находится на второй позиции (таблица 4.4).

Таким образом, основываясь только на конфигурации рёбер социального графа, найденные EgoLP покрытия позволяют со средней точностью 0.69 определять значения атрибутов, близость по которым оказывает влияние на формирование сообществ пользователей.

[6]Использовалась реализация из библиотеки http://scikit-learn.org/ со следующими параметрами: $n\_estimators = 1000$, $learning\_rate = 0.005$, $min\_samples\_split = 5$, $subsample = 0.4$.

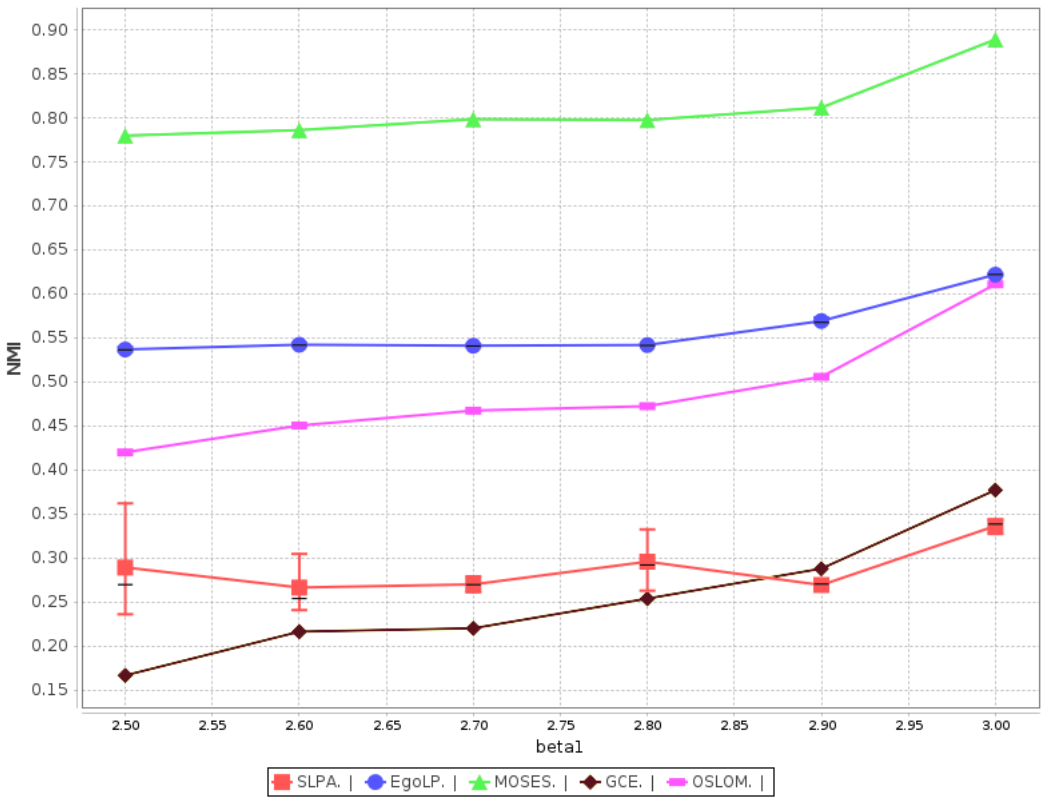


Рисунок 4.7: Сравнение качества EgoLP и других методов с помощью СКВ-графов с различными значениями β_1 — экспоненты степенного распределения количества сообществ у пользователя.

Таблица 4.4: Точность определения атрибутов “общежитие” и “год выпуска” для студентов колледжей и университетов из Facebook100.
Лучшие значения выделены полужирным начертанием, вторые лучшие значения выделены курсивом.

Название	$\|V\|$	$\|E\|$	Атрибут	EgoLP	GCE	OSLOM	MOSES	SLPA
UChicago	6,591	208,103	год выпуска	*0.64*	0.56	0.62	**0.72**	0.56
			общежитие	*0.57*	0.54	0.55	**0.66**	0.41
Caltech	769	16,656	год выпуска	*0.51*	0.44	0.42	**0.70**	0.38
			общежитие	0.79	**0.85**	*0.83*	0.78	0.70
Wellesley	2,970	94,899	год выпуска	0.71	*0.75*	*0.75*	**0.86**	*0.75*
Princeton	6,596	293,320	год выпуска	*0.82*	0.77	0.78	**0.88**	0.68
Lehigh	5,075	198,347	год выпуска	*0.74*	0.71	0.71	**0.87**	0.70
Cal	11,247	351,358	год выпуска	**0.71**	0.60	0.63	*0.70*	0.68
Среднее	≈5,541	≈193,780	–	*0.69*	0.65	0.66	**0.77**	0.61

4.6.3 Оценка свойств структуры сообществ

В приложении A содержатся результаты экспериментального исследования структурных свойств социальных графов с сообществами (раздел 1.3).

По результатам исследования можно заключить, что среди рассмотренных методов EgoLP и MOSES наиболее точно воспроизводят свойства референтных сообществ в продуцируемых покрытиях.

4.6.4 Оценка с помощью метрик качества

В приложении B содержатся результаты экспериментального исследования сообществ с помощью метрик качества (раздел 1.4).

По результатам исследования можно заключить:

- большинство методов находит сообщества с большей отделимостью по сравнению с референтными;
- большинство методов находит сообщества с большими значениями плотности, сплочённости и коэффициента кластеризации по сравнению с LFR-сообществами и с меньшими значениями — по сравнению с CKB-сообществами;
- среди всех методов GCE и SLPA показывают худшие результаты по совокупности метрик, а MOSES и EgoLP показывают лучшие результаты.

4.6.5 Производительность и масштабируемость

Для оценки скорости работы использовалось множество случайных социальных графов с 22, 50, 100, 217, 434 и 920 миллионами узлов и средней степенью 100. Эксперимент проводился на кластере с 18 узлами и 100 ядрами, на каждом узле кластера 24 Гб оперативной памяти и 4×1 Тб жестких диска. Во время каждого эксперимента были выполнены 10 итераций распространения меток. Рисунок 4.8 демонстрирует результаты.

Для исследования масштабируемости использовался граф с 25 миллионов узлов и средней степенью 100. Эксперимент проводился на кластерах с различным числом узлов: 18, 15, 12, 9 и 6 узлов, на каждом узле 12 ядер, 24 Гб оперативной памяти, 4×1 Тб жестких диска.

Рисунок 4.9 иллюстрирует результаты эксперимента для разных стадий EgoLP. Увеличение количества узлов кластера в k раз приводит к k-кратному увеличению скорости.

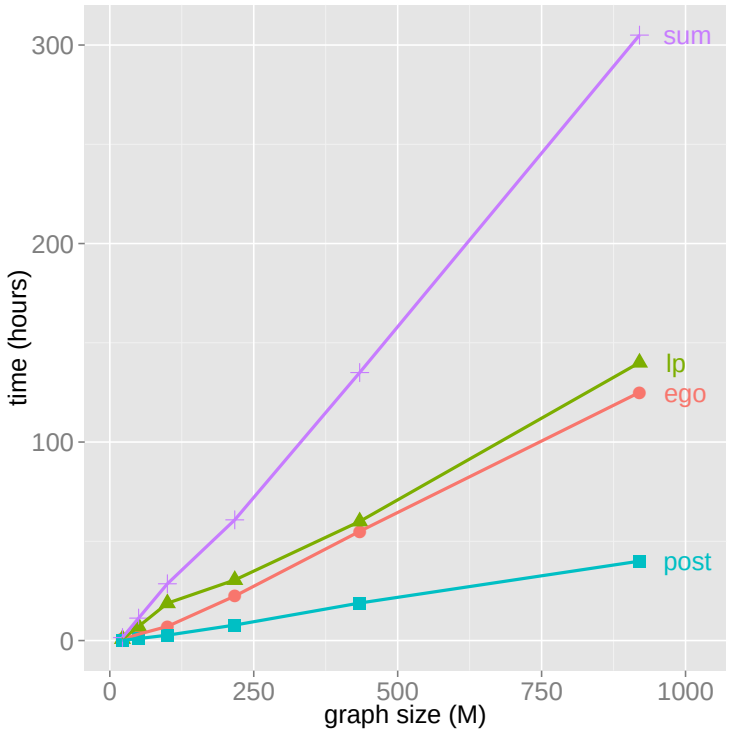


Рисунок 4.8: Время работы EgoLP в зависимости от размера графа. Обозначения на графиках соответствуют стадиям алгоритма: *ego* — определение структуры эго-сообществ, *lp* — распространение меток, *post* — определение и обработка плохо связных сообществ, *sum* — общее время.

Также была исследовано время работы EgoLP на графах небольшого размера (1000, 10000, 50000 и 100000 узлов) в сравнении с другими методами определения структуры сообществ (рисунок 4.10). Разработанная реализация EgoLP работает быстрее OSLOM, но уступает методам MOSES и GCE. Вместе с тем, для всех методов, кроме EgoLP обработка графов большего размера связана с существенной сложностью по памяти (разделы 2.1.1 и 2.1.2), что делает их неприменимыми на практике к графам размера $> 10^6$ вершин. Тогда как EgoLP позволяет обрабатывать графы из нескольких сотен миллионов вершин (рисунок 4.8).

4.7 Выводы

В ходе работы был разработан, реализован и исследован метод EgoLP для определения структуры значительно пересекающихся сообществ пользователей в социальных графах из сотен миллионов пользователей. Метод основан на итеративном распространении меток сообществ по рёбрам графа в соот-

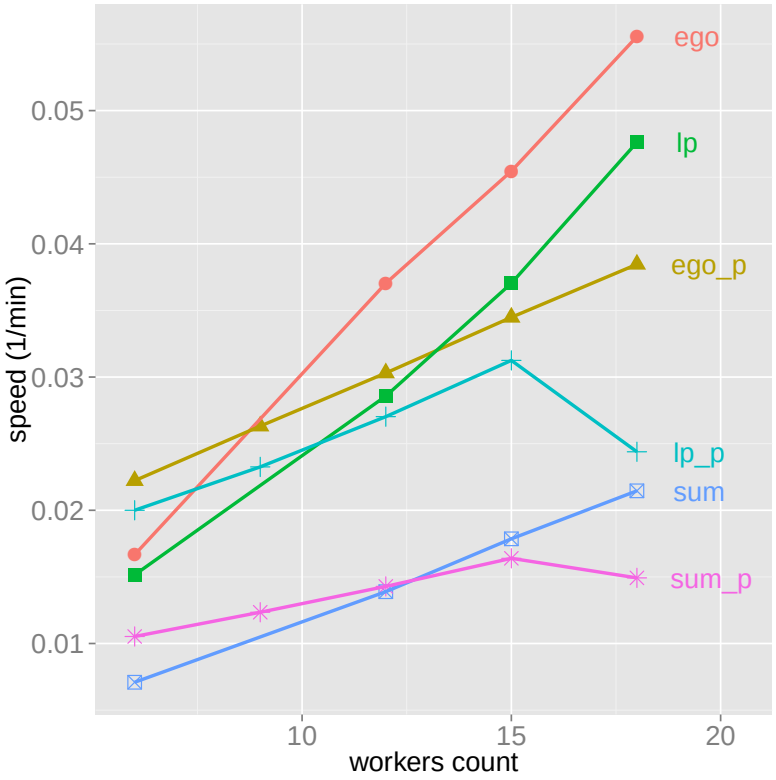


Рисунок 4.9: Скорость работы EgoLP в зависимости от количества обработчиков заданий в Spark. Обозначения на графиках соответствуют стадиям алгоритма: *ego* — определение структуры эго-сообществ, *lp* — распространение меток, *post* — определение подсообществ, *sum* — общая скорость. Постфикс *_p* соответствует запускам с фиксированным количеством независимых частей, на которые разбиваются данные, обрабатываемые на каждом узле кластера.

ветствии с заданными стратегиями взаимодействия вершин. При этом для каждой вершины графа также определяется структура эго-сообществ в сети её непосредственных контактов, что может быть использовано в различных приложениях и сервисах для персональной аналитики.

Экспериментальное исследование метода с помощью синтетических и реальных сетей показало, что EgoLP превосходит некоторые известные методы по набору критериев:

- качество восстановления заранее известной структуры сообществ;
- точность решения прикладной задачи определения скрытых атрибутов пользователей с использованием информации о сообществах;
- машстабируемость.

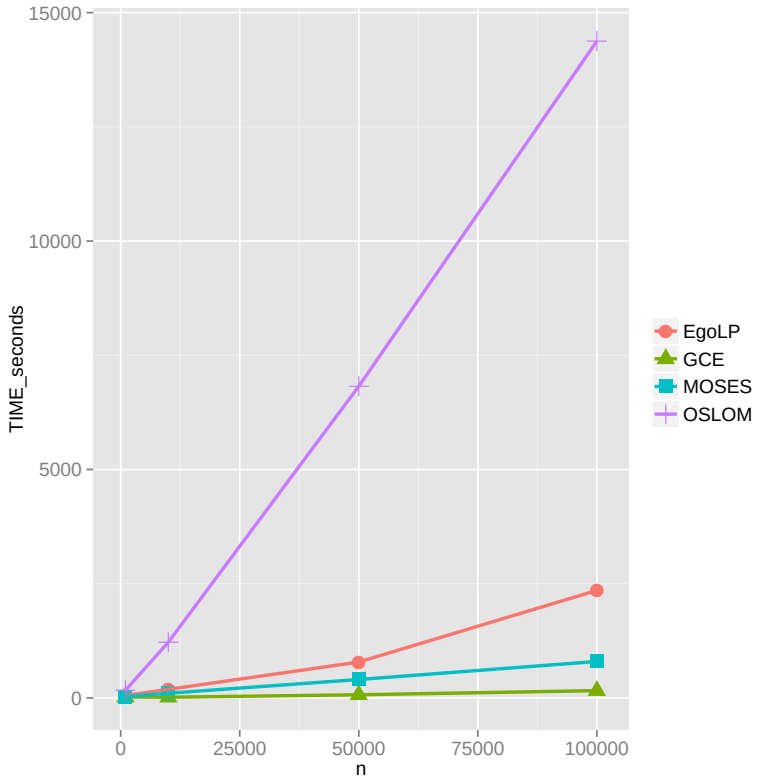


Рисунок 4.10: Время работы различных методов определения структуры сообществ.

При этом временная сложность метода линейно зависит от количества рёбер в графе, а распределённая реализация на основе фреймворка Apache Spark обладает близкой к линейной масштабируемостью. Таким образом, EgoLP является единственным из известных автору методов, позволяющих обрабатывать социальные графы в масштабе всей популяции пользователей (сотни миллионов вершин) с точностью, приемлемой для реальных приложений (на примере определения скрытых атрибутов пользователей).

Исходя из схемы работы предложенного алгоритма, можно предположить возможность создания модификаций для случаев ориентированных и взвешенных рёбер, а также для графов с категориальными атрибутами узлов и рёбер. Другим возможным направлением дальнейшей работы является выявление вложенных сообществ и построение иерархии найденных сообществ.

Заключение

Основные результаты работы заключаются в следующем:

1) проведено исследование современных методов определения структуры сообществ пользователей в графах онлайновых социальных сетей, показавшее ограниченную применимость и недостаточную эффективность большинства рассмотренных методов, особенно в приложениях, требующих определения структуры значительно пересекающихся сообществ в социальных сетях с сотнями миллионов пользователей;

2) проведено исследование современных методов генерации случайных социальных графов с заданной структурой сообществ пользователей, выявившее существенные различия между структурными свойствами синтезируемых графов и реальными социальными сетями с сообществами, а также отсутствие масштабируемых методов, позволяющих синтезировать графы из сотен миллионов вершин;

3) разработан метод `CKB` для распределённой генерации случайных социальных графов с заданной структурой сообществ, учитывающий некоторые из недавно доказанных свойств модульной структуры социальных сетей. Реализация модели с помощью Apache Spark позволяет осуществлять распределённую генерацию случайных социальных графов из сотен миллионов вершин с различными наборами параметров. Синтезированные графы с заданной структурой сообществ могут применяться для оценки применимости методов определения структуры пересекающихся сообществ к социальным графам различной природы;

4) разработан метод `EgoLP` для определения структуры значительно пересекающихся сообществ пользователей. Основой метода является итеративная пересылка меток сообществ по рёбрам графа в соответствии

с установленными правилами взаимодействия вершин. Отличительной особенностью является идентификация эго-сообществ в сети непосредственных соседей каждого пользователя. В дальнейшем с помощью особых правил взаимодействия вершин поощряется объединение эго-сообществ в глобальные. Метод имеет распределённую реализацию на основе Apache Spark с использованием парадигмы распределённых вычислений Pregel. Экспериментально продемонстрировано, что предложенный метод превосходит известные методы по совокупности критериев: а) близость определённой структуры сообществ с заранее известной; б) точность решения прикладной задачи определения скрытых атрибутов пользователей с использованием информации о сообществах; в) вычислительная сложность; г) масштабируемость. Сложность алгоритма линейно зависит от количества рёбер графа, что позволяет применять его к социальным сетям с сотнями миллионов пользователей;

5) для экспериментального подтверждения эффективности предложенных методов реализованы прототипы систем для определения структуры сообществ пользователей и генерации случайных социальных графов с заданной структурой сообществ пользователей. Реализованные прототипы позволили подтвердить высокое качество предложенных методов и соответствие экспериментальных оценок производительности теоретическим оценкам вычислительной сложности

Разработанные методы позволяют исследовать структуру сообществ социальных сетей из сотен миллионов пользователей и применять полученные знания для решения исследовательских и бизнес-задач, а также для оптимизации решения других задач анализа больших социальных графов.

Литература

1. Fortunato Santo. Community detection in graphs // Physics Reports. 2010. Т. 486, № 3. С. 75–174.

2. Xie Jierui, Kelley Stephen, Szymanski Boleslaw K. Overlapping community detection in networks: The state-of-the-art and comparative study // ACM Computing Surveys (CSUR). 2013. Т. 45, № 4. С. 43.

3. Yang Bo, Liu Dayou, Liu Jiming. Discovering communities from social networks: methodologies and applications // Handbook of Social Network Technologies and Applications. Springer, 2010. С. 331–346.

4. Tang Lei, Liu Huan. Community detection and mining in social media // Synthesis Lectures on Data Mining and Knowledge Discovery. 2010. Т. 2, № 1. С. 1–137.

5. Yang J., Leskovec J. Community-Affiliation Graph Model for Overlapping Network Community Detection // IEEE 12th International Conference on Data Mining. 2012.

6. Community Structure in Large Networks: Natural Cluster Sizes and the Absence of Large Well-Defined Clusters / J. Leskovec, K. Lang, A. Dasgupta [и др.] // Internet Mathematics. 6(1). Т. 29–123. С. 2009.

7. Statistical properties of community structure in large social and information networks / Jure Leskovec, Kevin J Lang, Anirban Dasgupta [и др.] // Proceedings of the 17th international conference on World Wide Web / ACM. 2008. С. 695–704.

8. Yang Jaewon, Leskovec Jure. Structure and Overlaps of Ground-Truth Communities in Networks // ACM Transactions on Intelligent Systems and Technology (TIST). 2014. Т. 5, № 2. С. 26.

9. Yang Jaewon, Leskovec Jure. Defining and evaluating network communities based on ground-truth // Knowledge and Information Systems. 2015. Т. 42, № 1. С. 181–213.

10. Yang Jaewon, Leskovec Jure. Community-affiliation graph model for overlapping network community detection // Data Mining (ICDM), 2012 IEEE 12th International Conference on / IEEE. 2012. С. 1170–1175.

11. Leskovec Jure, Lang Kevin J, Mahoney Michael. Empirical comparison of algorithms for network community detection // Proceedings of the 19th international conference on World wide web / ACM. 2010. С. 631–640.

12. Yang Jaewon, Leskovec Jure. Overlapping community detection at scale: a nonnegative matrix factorization approach // Proceedings of the sixth ACM international conference on Web search and data mining / ACM. 2013. С. 587–596.

13. Lancichinetti A., Fortunato S. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities // Phys. Rev. 2009. Т. 80(1).

14. Finding statistically significant communities in networks / Andrea Lancichinetti, Filippo Radicchi, José J Ramasco [и др.] // PloS one. 2011. Т. 6, № 4. С. e18961.

15. Пупырев Сергей Николаевич. Визуализация структуры сообществ в графах // Системы управления и информационные технологии. 2009. № 2. С. 36.

16. Алгоритм выделения сообществ в социальных сетях / Максим Игоревич Коломейченко, Александр Андреевич Чеповский, Андрей Михайлович Чеповский [и др.] // Фундаментальная и прикладная математика. 2014. Т. 19, № 1. С. 21–32.

17. Prat-Pérez Arnau, Dominguez-Sal David, Larriba-Pey Josep-Lluis. High quality, scalable and parallel community detection for large real graphs // Proceedings of the 23rd international conference on World wide web / International World Wide Web Conferences Steering Committee. 2014. С. 225–236.

18. Fast unfolding of communities in large networks / Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte [и др.] // Journal of Statistical Mechanics: Theory and Experiment. 2008. Т. 2008, № 10. С. P10008.

19. Raghavan Usha Nandini, Albert Réka, Kumara Soundar. Near linear time algorithm to detect community structures in large-scale networks // Physical Review E. 2007. Т. 76, № 3. С. 036106.

20. Xie Jierui, Szymanski Boleslaw K, Liu Xiaoming. SLPA: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process // 2011 IEEE 11th International Conference on Data Mining Workshops (ICDMW) / IEEE. 2011. С. 344–349.

21. Girvan M., Newman M. Community structure in social and biological networks // Proc. Natl. Acad. Sci. 2002. Т. 99.

22. Коршунов Антон. Задачи и методы определения атрибутов пользователей социальных сетей // Selected Papers of the 15th All-Russian Scientific Conference "Digital libraries: Advanced Methods and Technologies, Digital Collections Yaroslavl, Russia, October 14-17, 2013. CEUR Workshop Proceedings 1108. 2013. С. 183–193.

23. Анализ социальных сетей: методы и приложения / Антон Коршунов, Иван Белобородов, Бузун Назар [и др.] // ТРУДЫ ИНСТИТУТА СИСТЕМНОГО ПРОГРАММИРОВАНИЯ РАН. 2014. Т. 26, № 1.

24. Гомзин А.Г., Ипатов С.А., Коршунов А.В. Рекомендация получателей электронных сообщений с использованием различных типов локальных данных социальных сетей // Вестник НовГУ. 2014. № 81.

25. Distributed Generation of Billion-node Social Graphs with Overlapping Community Structure / Kyrylo Chykhradze, Anton Korshunov,

Nazar Buzun [и др.] // Complex Networks V. Springer, 2014. С. 199–208.

26. Использование модели социальной сети с сообществами пользователей для распределенной генерации случайных социальных графов / К.К. Чихрадзе, А.В. Коршунов, Н.О. Бузун [и др.] // Машинное обучение и анализ данных. 2014. Т. 1s, № 8. С. 1027–1047.

27. EgoLP: Fast and Distributed Community Detection in Billion-node Social Networks / Nazar Buzun, Anton Korshunov, Valeriy Avanesov [и др.] // Proceedings of DaMNet-2014 - The Fourth IEEE ICDM Workshop on Data Mining in Networks. December 14, 2014, Shenzhen, China. С. 533 – 540.

28. Бузун Назар, Коршунов Антон. Выявление пересекающихся сообществ в социальных сетях // Доклады Всероссийской научной конференции «Анализ изображений, сетей и текстов»-АИСТ. 2012.

29. Buzun Nazar, Korshunov Anton. Innovative methods and measures in overlapping community detection // Proceedings of International Workshop on Experimental Economics in Machine Learning. 2012. С. 20–32.

30. Prentice Deborah A, Miller Dale T, Lightdale Jenifer R. Asymmetries in attachments to groups and to their members: Distinguishing between common-identity and common-bond groups // Key Readings in Social Psychology. 1994. С. 83.

31. Ferrara Emilio. A large-scale community structure analysis in Facebook // EPJ Data Science. 2012. Т. 1, № 1. С. 1–30.

32. Measurement and analysis of online social networks / Alan Mislove, Massimiliano Marcon, Krishna P Gummadi [и др.] // Proceedings of the 7th ACM SIGCOMM conference on Internet measurement / ACM. 2007. С. 29–42.

33. Reid Fergal, McDaid Aaron, Hurley Neil. Partitioning breaks communities // Mining Social Networks and Security Informatics. Springer, 2013. С. 79–105.

34. Lazarsfeld Paul F, Merton Robert K [и др.]. Friendship as a social process: A substantive and methodological analysis // Freedom and control in modern society. 1954. Т. 18, № 1. С. 18–66.

35. McPherson Miller, Smith-Lovin Lynn, Cook James M. Birds of a feather: Homophily in social networks // Annual review of sociology. 2001. С. 415–444.

36. Watts Duncan J, Strogatz Steven H. Collective dynamics of 'small-world'networks // nature. 1998. Т. 393, № 6684. С. 440–442.

37. Granovetter Mark S. The strength of weak ties // American journal of sociology. 1973. С. 1360–1380.

38. Simmel Georg. The web of group affiliations // Conflict and the web of group affiliations. 1955. С. 125–95.

39. Feld Scott L. The focused organization of social ties // American journal of sociology. 1981. С. 1015–1035.

40. Detecting highly overlapping community structure by greedy clique expansion / Conrad Lee, Fergal Reid, Aaron McDaid [и др.] // arXiv preprint arXiv:1002.1827. 2010.

41. Balanced multi-label propagation for overlapping community detection in social networks / Zhi-Hao Wu, You-Fang Lin, Steve Gregory [и др.] // Journal of Computer Science and Technology. 2012. Т. 27, № 3. С. 468–479.

42. McDaid Aaron, Hurley Neil. Detecting highly overlapping communities with model-based overlapping seed expansion // Advances in Social Networks Analysis and Mining (ASONAM), 2010 International Conference on / IEEE. 2010. С. 112–119.

43. Morup Morten, Schmidt M. Bayesian community detection // Neural computation. 2012. Т. 24, № 9. С. 2434–2456.

44. Yang Jaewon, McAuley Julian, Leskovec Jure. Community detection in networks with node attributes // Data Mining (ICDM), 2013 IEEE 13th International Conference on / IEEE. 2013. С. 1151–1156.

45. Pregel: a system for large-scale graph processing / Grzegorz Malewicz, Matthew H Austern, Aart JC Bik [и др.] // Proceedings of the 2010 ACM SIGMOD International Conference on Management of data / ACM. 2010. С. 135–146.

46. Graphx: A resilient distributed graph system on spark / Reynold S Xin, Joseph E Gonzalez, Michael J Franklin [и др.] // First International Workshop on Graph Data Management Experiences and Systems / ACM. 2013. С. 2.

47. Gregory Steve. Finding overlapping communities in networks by label propagation // New Journal of Physics. 2010. Т. 12, № 10. С. 103018.

48. SOUNDARAJAN SUCHETA, HOPCROFT JOHN E. Use of Local Group Information to Identify Communities in Networks // ACM Transactions on Knowledge Discovery from Data (TKDD). 2014.

49. Demon: a local-first discovery method for overlapping communities / Michele Coscia, Giulio Rossetti, Fosca Giannotti [и др.] // Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining / ACM. 2012. С. 615–623.

50. Rees Bradley S, Gallagher Keith B. EgoClustering: overlapping community detection via merged friendship-groups. Springer, 2013.

51. Rees Bradley S, Gallagher Keith B. Detecting Overlapping Communities in Complex Networks Using Swarm Intelligence for Multi-threaded Label Propagation // Complex Networks. Springer, 2013. С. 111–119.

52. Kuzmin Konstantin, Shah S Yousaf, Szymanski Boleslaw K. Parallel overlapping community detection with SLPA // Social Computing (SocialCom), 2013 International Conference on / IEEE. 2013. С. 204–212.

53. Parallel community detection on large networks with propinquity dynamics / Yuzhou Zhang, Jianyong Wang, Yi Wang [и др.] // Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining / ACM. 2009. С. 997–1006.

54. Accurate spectral clustering for community detection in MapReduce / Serafeim Tsironis, Mauro Sozio, Michalis Vazirgiannis [и др.] // Frontiers of network analysis: methods, models, and applications. Lake Tahoe, NIPS workshop / Citeseer. 2013.

55. Rytsareva Inna, Chapman Timothy, Kalyanaraman Ananth. Parallel algorithms for clustering biological graphs on distributed and shared memory architectures // International Journal of High Performance Computing and Networking. 2014. Т. 7, № 4. С. 241–257.

56. Erdos P., Renyi A. On the evolution of random graphs // Bull. Inst. Int. Statist. Tokyo. 1961. Т. 38. С. 343–347.

57. Danon L., Díaz-Guilera A., Arenas A. The effect of size heterogeneity on community identification in complex networks // Journal of Statistical Mechanics: Theory and Experiment. 2006. Т. 2006, № 11. С. P11010.

58. Sawardecker E.N., Sales-Pardo M., Amaral L.A.N. Detection of node group membership in networks with group overlap // The European Physical Journal B. 2009. Т. 67, № 3. С. 277–284.

59. Arenas A., Díaz-Guilera A., Pérez-Vicente C.J. Synchronization reveals topological scales in complex networks // Physical review letters. 2006. Т. 96, № 11. С. 114102.

60. Accuracy and precision of methods for community identification in weighted networks / Y. Fan, M. Li, P. Zhang [и др.] // Physica A: Statistical Mechanics and its Applications. 2007. Т. 377, № 1. С. 363–372.

61. Seshadhri C., Kolda T. G., Pinar Ali. Community Structure and Scale-Free Collections of Erdos-Renyi Graphs // Physical Review E. 2012. Т. 85(5).

62. Aiello W., Chung F., Lu L. A Random Graph Model for Massive Graphs // 32nd Annual ACM Symposium on Theory of Computing. 2000. С. 171–180.

63. Singer K. Random intersection graphs // PhD thesis, Johns Hopkins University. 1995.

64. Deijfen M., Kets W. Random intersection graphs with tunable degree distribution and clustering // Probab. Engrg. Inform. Sci. 2009. Т. 23(4). С. 615–623.

65. Molloy M., Reed B. A Critical Point for Random Graphs with a Given Degree Sequence // Random Structures and Algorithms. 1995. Т. 6. С. 161–180.

66. Collins Linda M, Dent Clyde W. Omega: A general formulation of the rand index of cluster recovery suitable for non-disjoint solutions // Multivariate Behavioral Research. 1988. Т. 23, № 2. С. 231–242.

67. Lee Conrad, Cunningham Pádraig. Benchmarking community detection methods on social media data // arXiv preprint arXiv:1302.0739. 2013.

68. Overlapping communities in dynamic networks: their detection and mobile applications / Nam P Nguyen, Thang N Dinh, Sindhura Tokala [и др.] // Proceedings of the 17th annual international conference on Mobile computing and networking / ACM. 2011. С. 85–96.

69. UNIK: unsupervised social network spam detection / Enhua Tan, Lei Guo, Songqing Chen [и др.] // Proceedings of the 22nd ACM international conference on Conference on information & knowledge management / ACM. 2013. С. 479–488.

70. Alleviating the sparsity problem in recommender systems by exploring underlying user communities / Aline Bessa, Alberto HF Laender, Adriano Veloso [и др.]. 2012.

71. Sahebi Shaghayegh, Cohen William W. Community-based recommendations: a solution to the cold start problem // Workshop on Recommender Systems and the Social Web, RSWEB. 2011.

72. Wegner Anatol. Random graphs with motifs. Preprint // Max Planck Institute for Mathematics in the Sciences. 2011.

73. Valiant Leslie G. A bridging model for parallel computation // Communications of the ACM. 1990. Т. 33, № 8. С. 103–111.

74. Fiedler Miroslav. Algebraic connectivity of graphs // Czechoslovak Mathematical Journal. 1973. Т. 23, № 2. С. 298–305.

75. Knyazev Andrew V. Toward the optimal preconditioned eigensolver: Locally optimal block preconditioned conjugate gradient method // SIAM journal on scientific computing. 2001. Т. 23, № 2. С. 517–541.

76. Lee Conrad, Cunningham Pádraig. Benchmarking community detection methods on social media data // arXiv preprint arXiv:1302.0739. 2013.

77. Traud Amanda L, Mucha Peter J, Porter Mason A. Social structure of Facebook networks // Physica A: Statistical Mechanics and its Applications. 2012. Т. 391, № 16. С. 4165–4180.

78. Friedman Jerome H. Greedy function approximation: a gradient boosting machine // Annals of statistics. 2001. С. 1189–1232.

Приложение А

Свойства графов с сообществами

В данном приложении содержатся результаты экспериментального исследования структурных свойств социальных графов с сообществами (раздел 1.3).

Исследованы покрытия сообществ, найденные различными методами, в сравнении с референтными покрытиями, синтезированными методами CKB и LFR.

На всех графиках красная линия соответствует референтным, а синяя – алгоритмически найденным сообществам.

Выводы по результатам анализа полученных графиков содержатся в разделе 4.6.3.

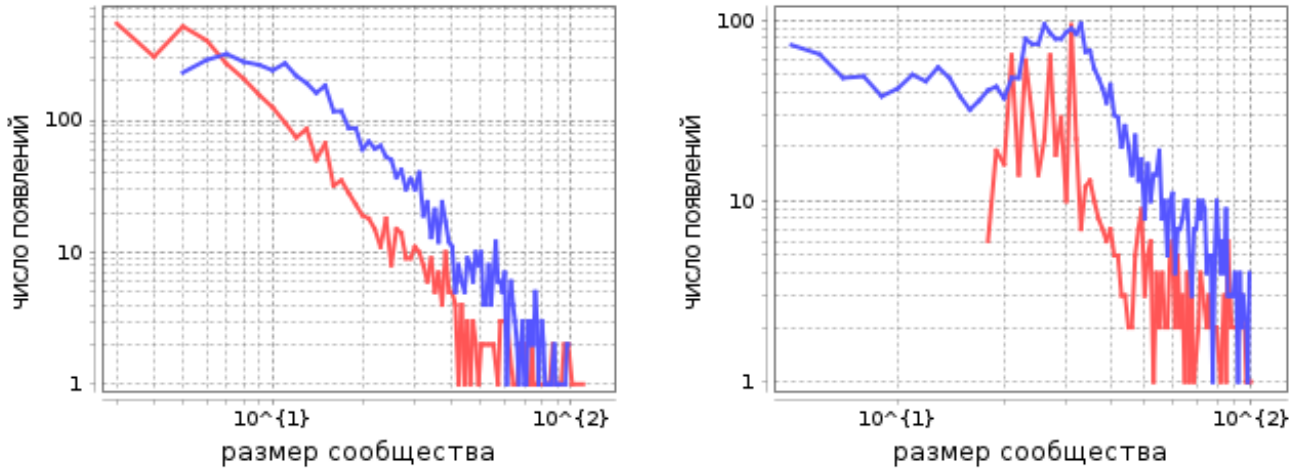


Рисунок А.1: EgoLP: распределение размеров найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

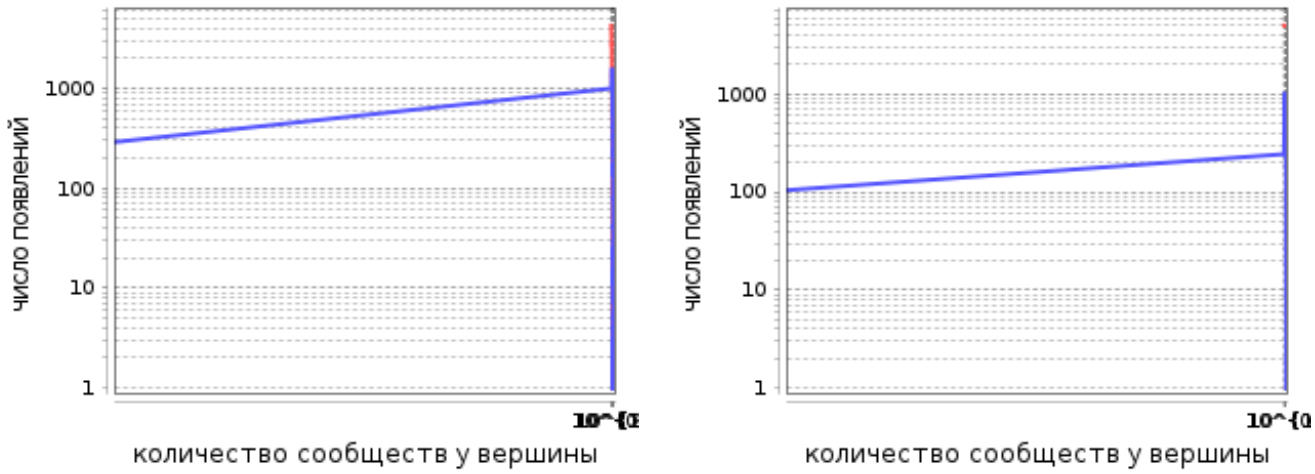


Рисунок А.2: EgoLP: распределение количества найденных (синяя линия) и референтных (красная линия) сообществ у пользователя на шаблонных сетях CKB (слева) и LFR (справа).

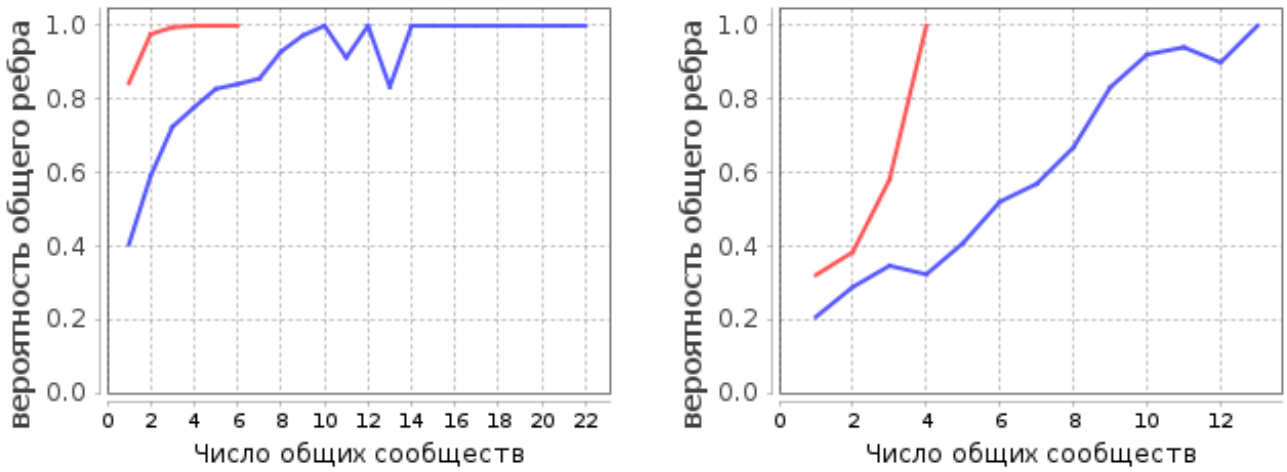


Рисунок А.3: EgoLP: зависимость вероятности ребра от количества общих сообществ у его концевых вершин для найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

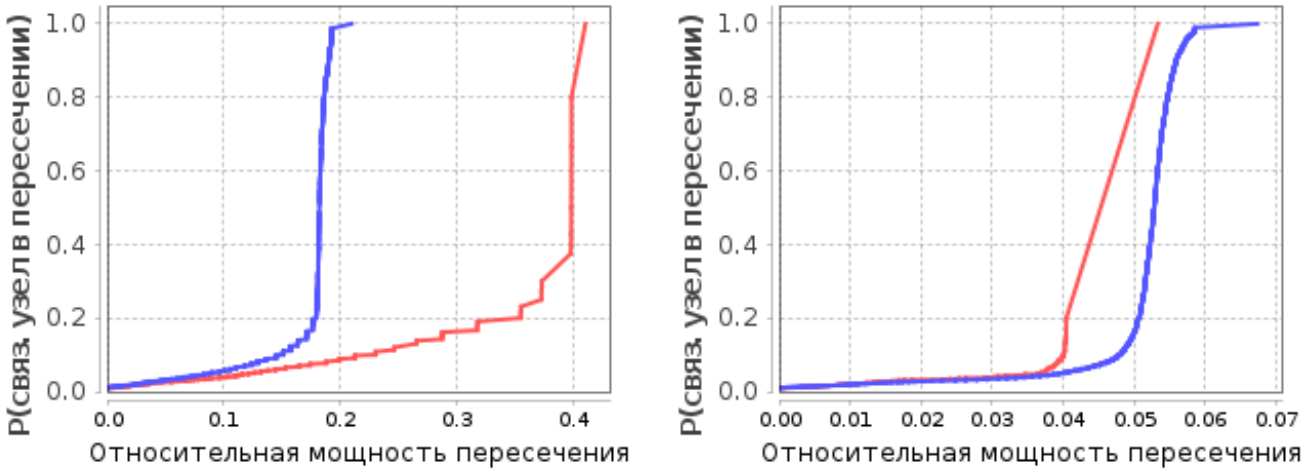


Рисунок А.4: EgoLP: зависимость вероятности появления связующей вершины в пересечении найденных (синяя линия) и референтных (красная линия) сообществ от относительной мощности пересечения на шаблонных сетях CKB (слева) и LFR (справа).

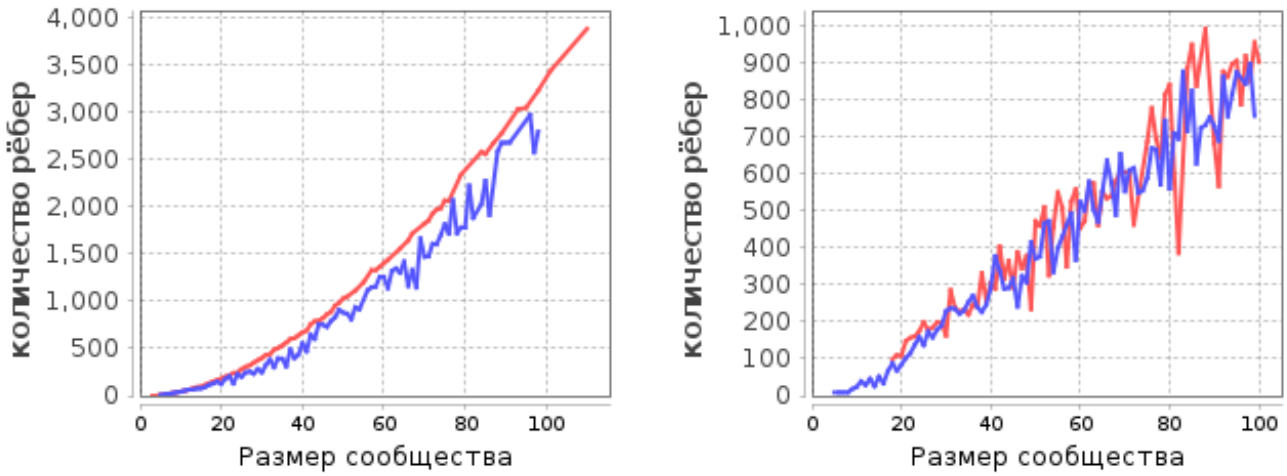


Рисунок A.5: EgoLP: зависимость количества рёбер в найденных (синяя линия) и референтных (красная линия) сообществ от их размеров на шаблонных сетях CKB (слева) и LFR (справа).

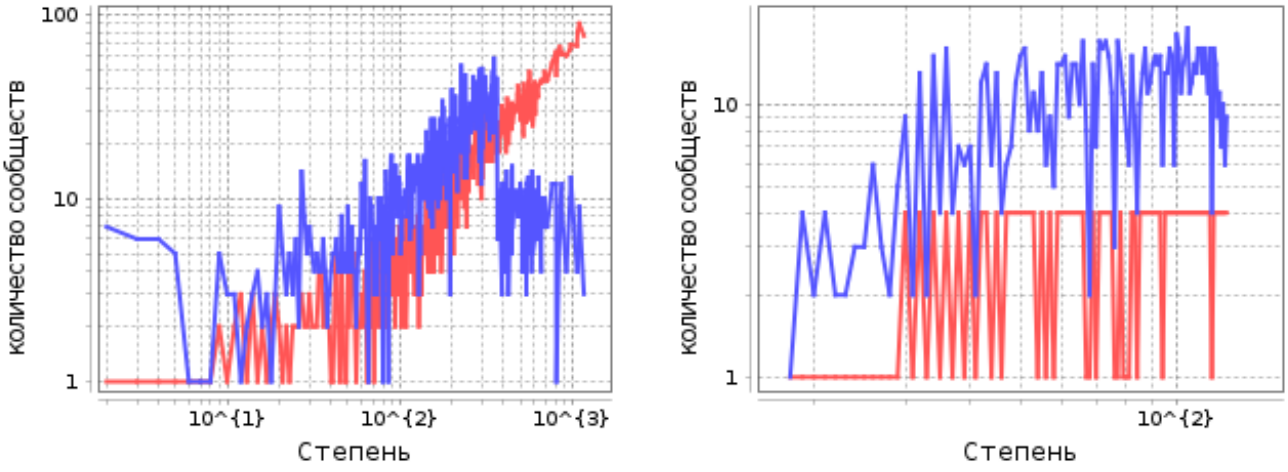


Рисунок A.6: EgoLP: зависимость количества найденных (синяя линия) и референтных (красная линия) сообществ от степени вершины на шаблонных сетях CKB (слева) и LFR (справа).

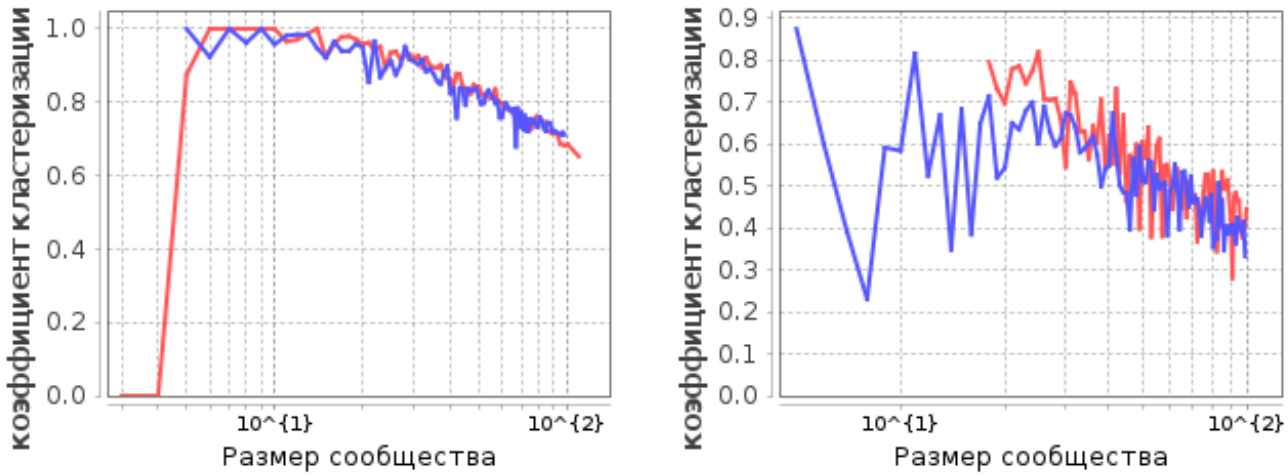


Рисунок А.7: EgoLP: зависимость среднего коэффициента кластеризации от размера найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

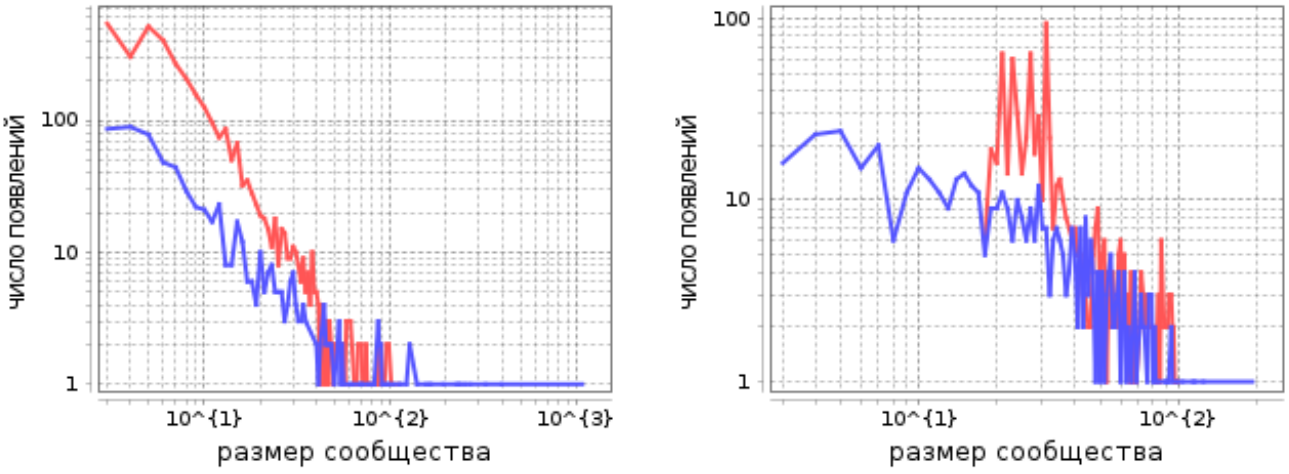


Рисунок А.8: SLPA: распределение размеров найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

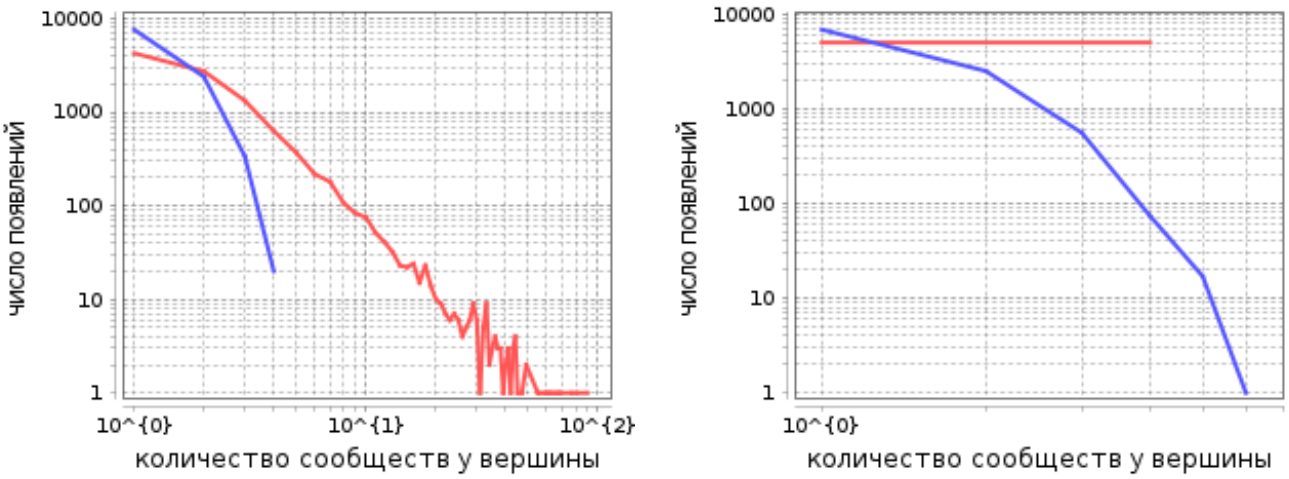


Рисунок А.9: SLPA: распределение количества найденных (синяя линия) и референтных (красная линия) сообществ у пользователя на шаблонных сетях CKB (слева) и LFR (справа).

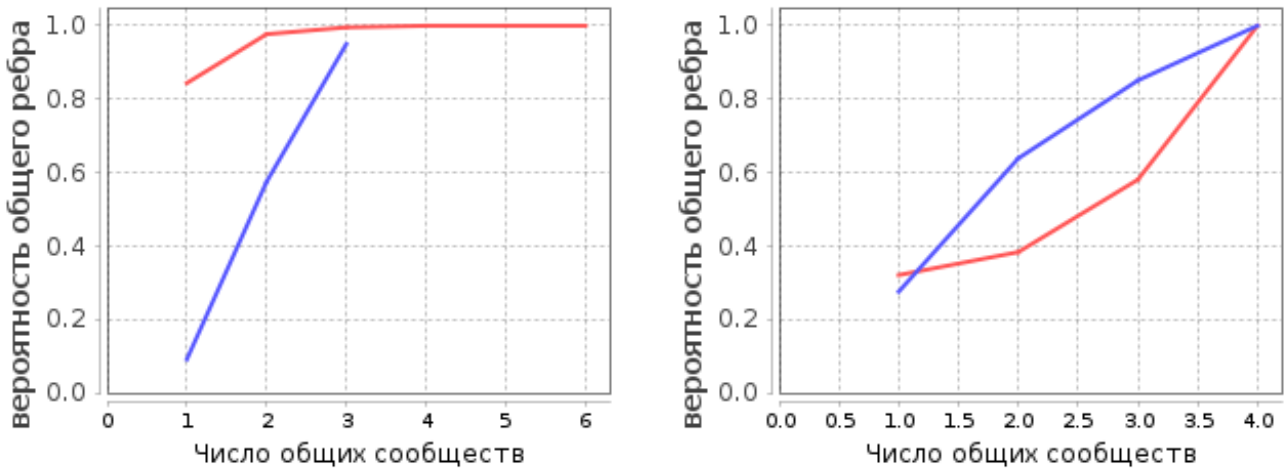


Рисунок А.10: SLPA: зависимость вероятности ребра от количества общих сообществ у его концевых вершин для найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

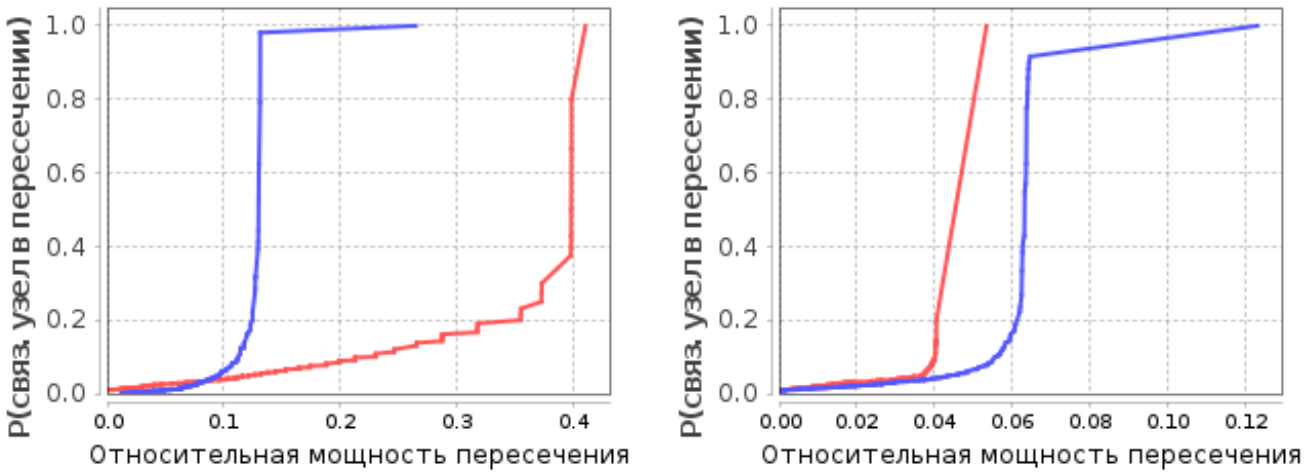


Рисунок А.11: SLPA: зависимость вероятности появления связующей вершины в пересечении найденных (синяя линия) и референтных (красная линия) сообществ от относительной мощности пересечения на шаблонных сетях CKB (слева) и LFR (справа).

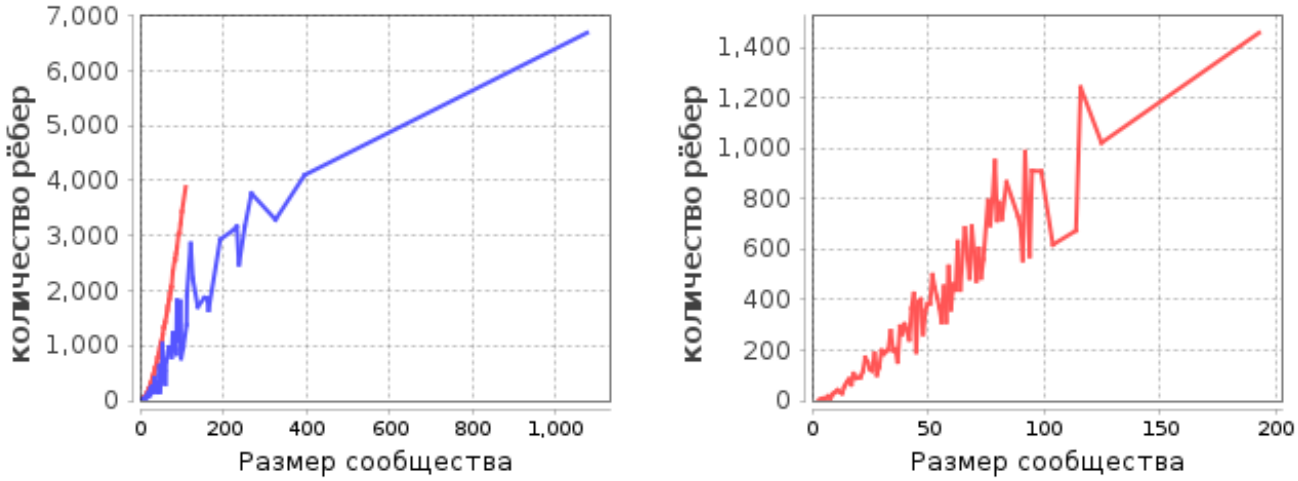


Рисунок А.12: SLPA: зависимость количества рёбер в найденных (синяя линия) и референтных (красная линия) сообществ от их размеров на шаблонных сетях CKB (слева) и LFR (справа).

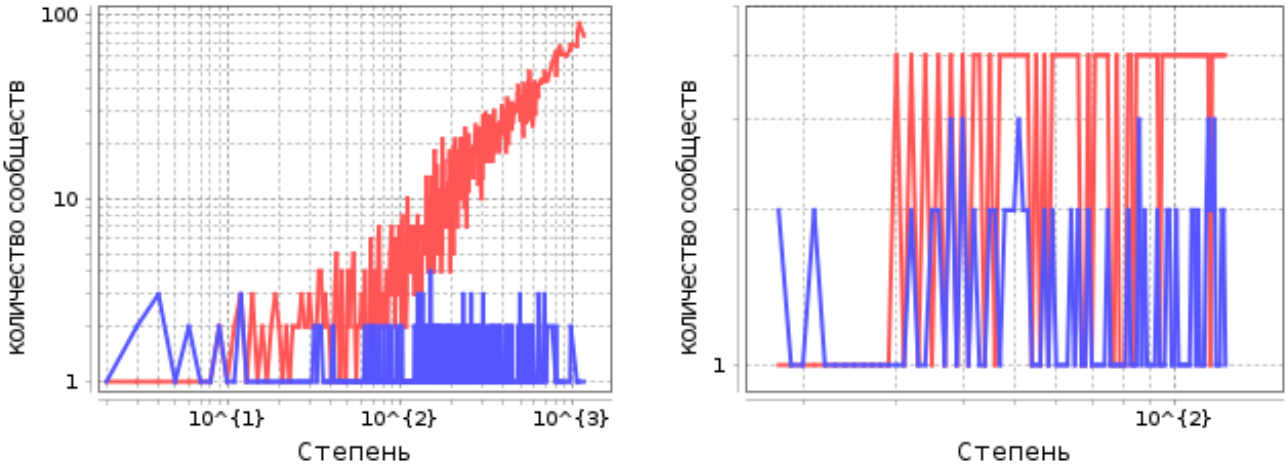


Рисунок А.13: SLPA: зависимость количества найденных (синяя линия) и референтных (красная линия) сообществ от степени вершины на шаблонных сетях CKB (слева) и LFR (справа).

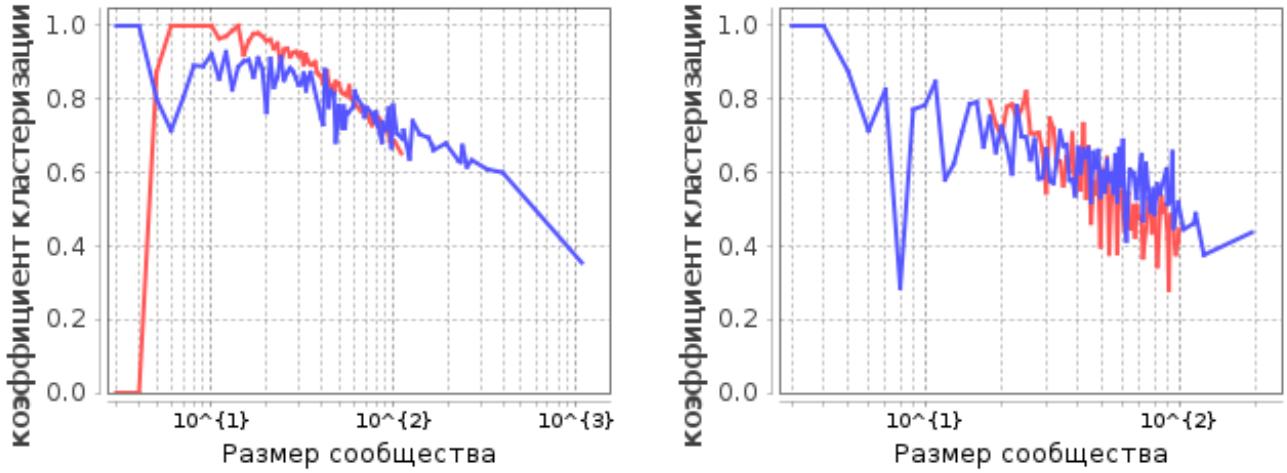


Рисунок А.14: SLPA: зависимость среднего коэффициента кластеризации от размера найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

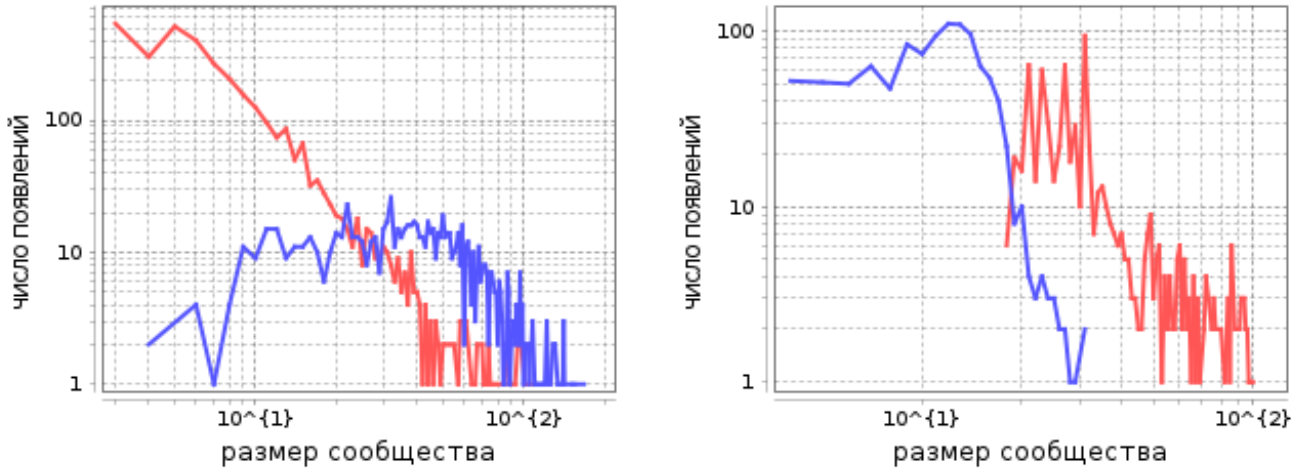


Рисунок А.15: GCE: распределение размеров найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

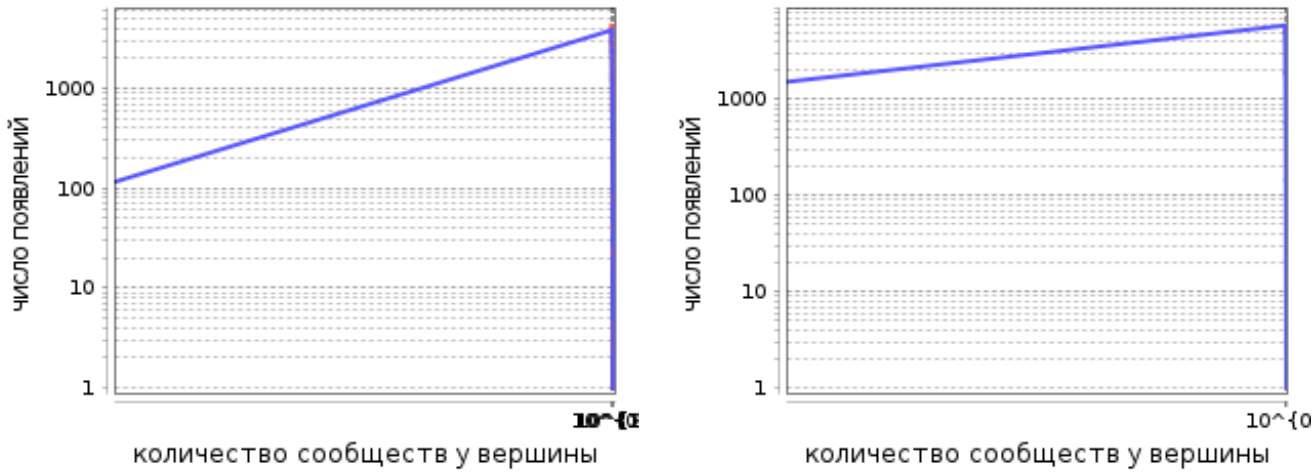


Рисунок А.16: GCE: распределение количества найденных (синяя линия) и референтных (красная линия) сообществ у пользователя на шаблонных сетях CKB (слева) и LFR (справа).

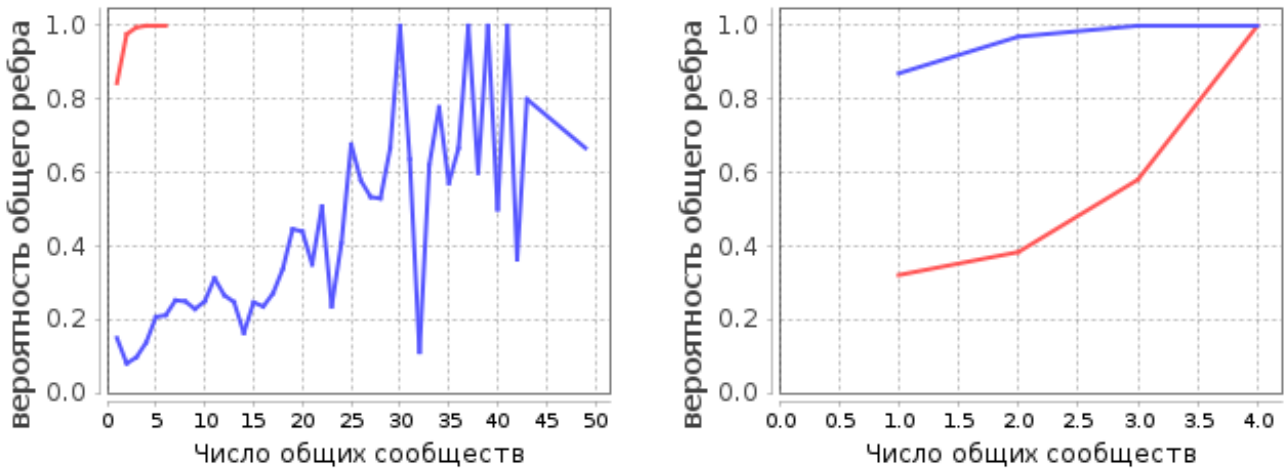


Рисунок А.17: GCE: зависимость вероятности ребра от количества общих сообществ у его концевых вершин для найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

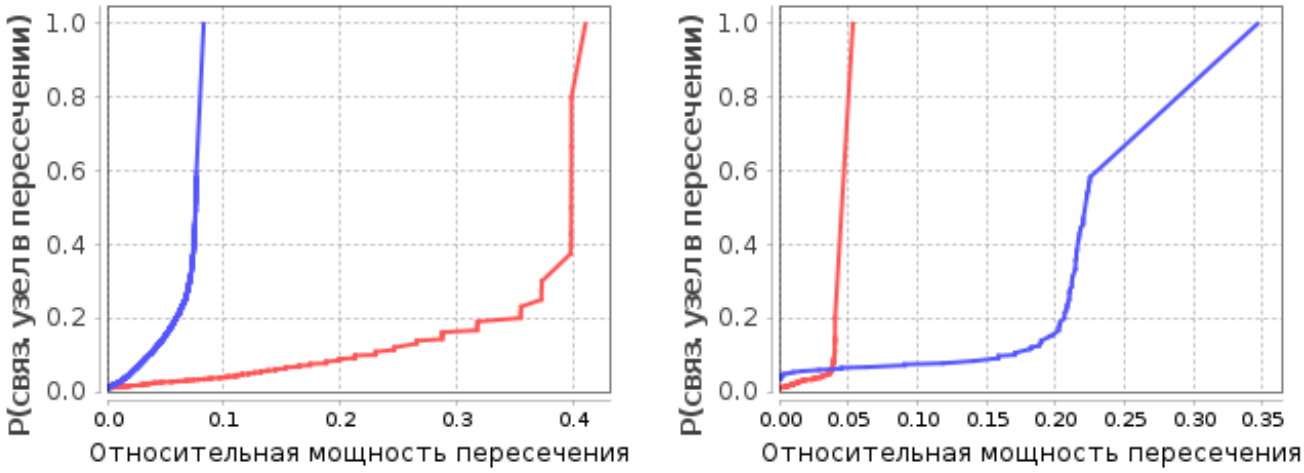


Рисунок А.18: GCE: зависимость вероятности появления связующей вершины в пересечении найденных (синяя линия) и референтных (красная линия) сообществ от относительной мощности пересечения на шаблонных сетях CKB (слева) и LFR (справа).

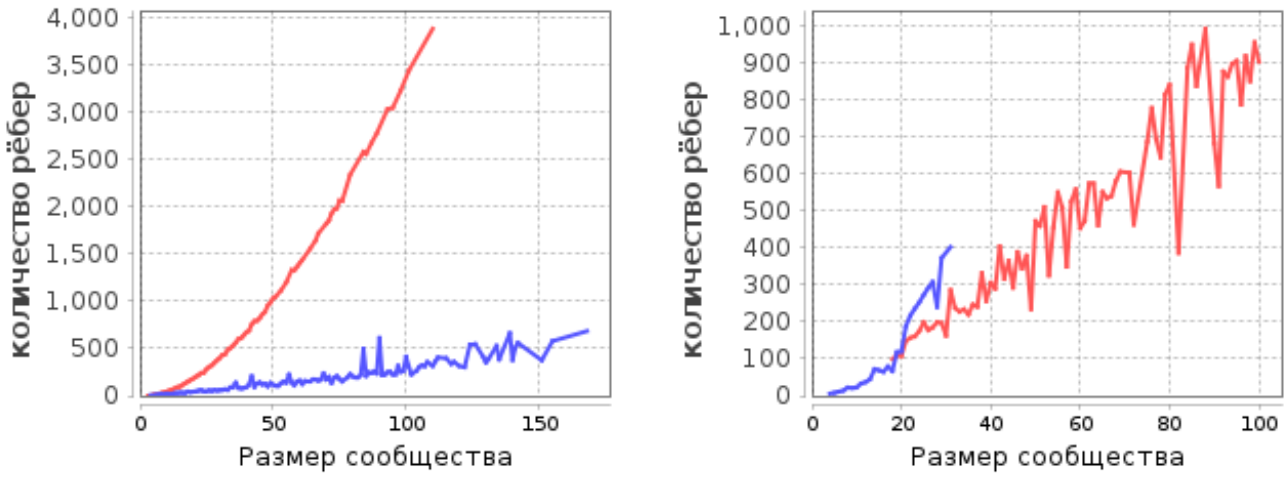


Рисунок А.19: GCE: зависимость количества рёбер в найденных (синяя линия) и референтных (красная линия) сообществ от их размеров на шаблонных сетях CKB (слева) и LFR (справа).

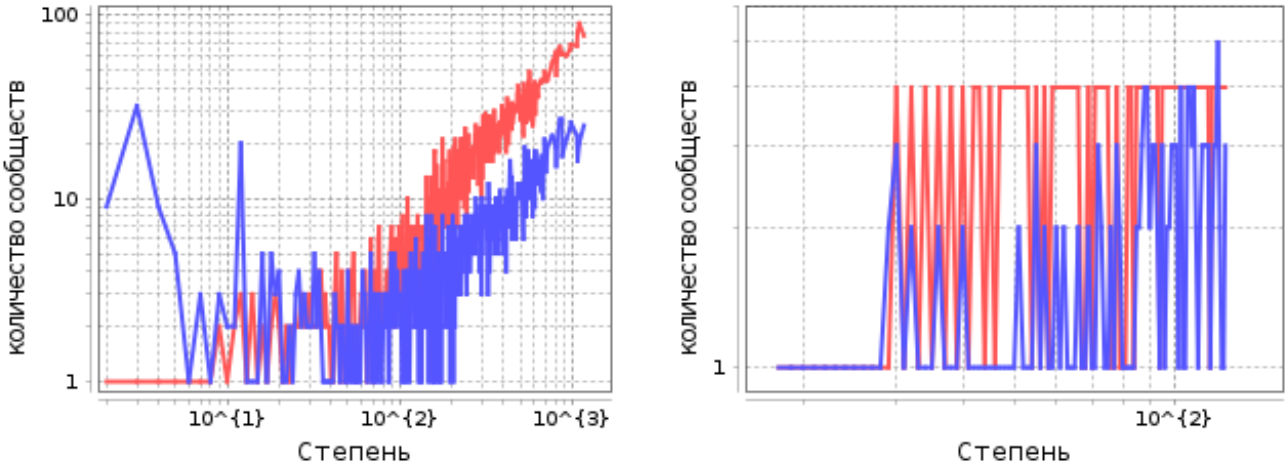


Рисунок А.20: GCE: зависимость количества найденных (синяя линия) и референтных (красная линия) сообществ от степени вершины на шаблонных сетях CKB (слева) и LFR (справа).

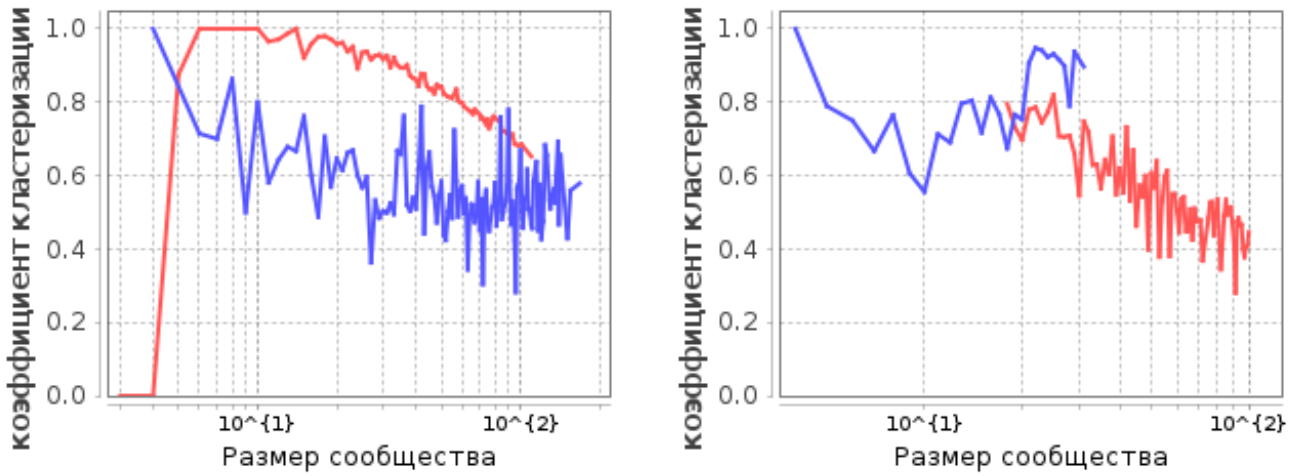


Рисунок А.21: GCE: зависимость среднего коэффициента кластеризации от размера найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

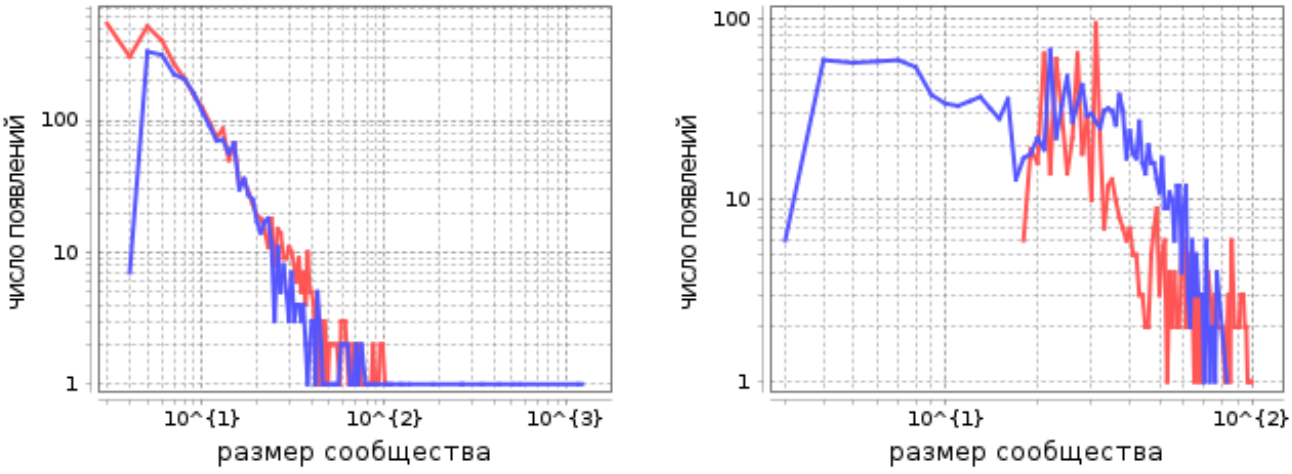


Рисунок А.22: MOSES: распределение размеров найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

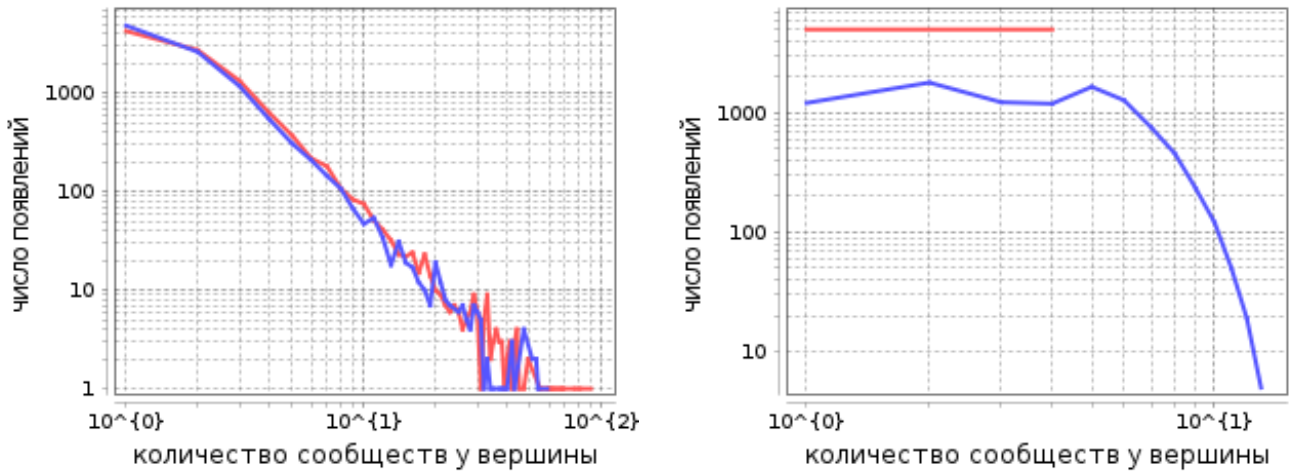


Рисунок А.23: MOSES: распределение количества найденных (синяя линия) и референтных (красная линия) сообществ у пользователя на шаблонных сетях CKB (слева) и LFR (справа).

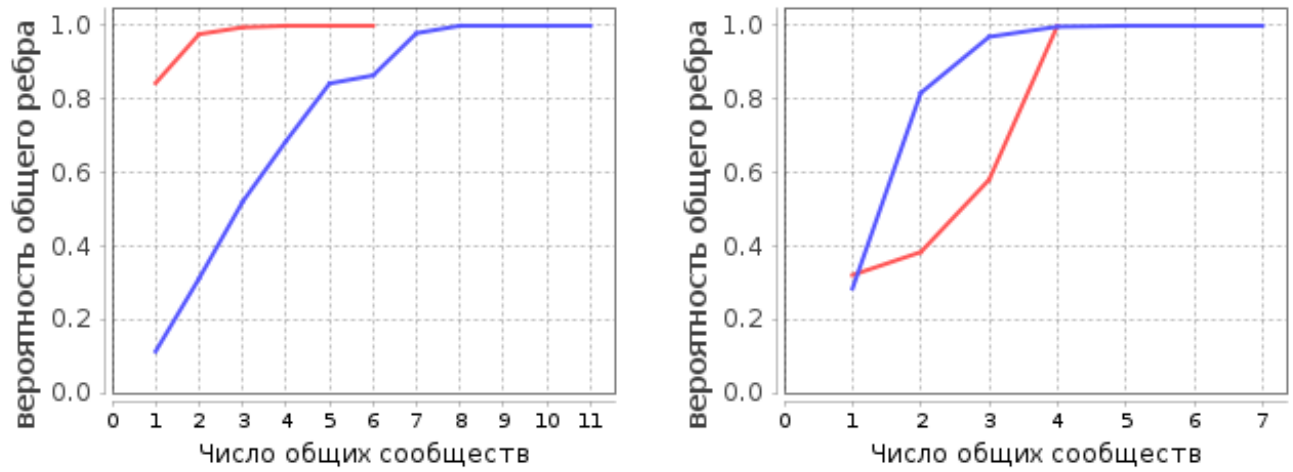


Рисунок А.24: MOSES: зависимость вероятности ребра от количества общих сообществ у его концевых вершин для найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

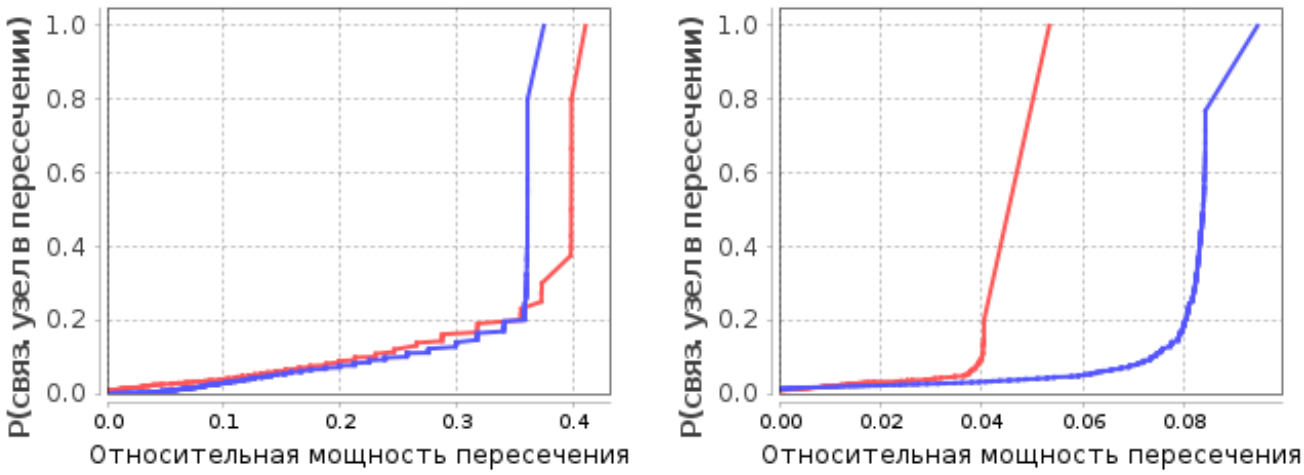


Рисунок А.25: *MOSES*: зависимость вероятности появления связующей вершины в пересечении найденных (синяя линия) и референтных (красная линия) сообществ от относительной мощности пересечения на шаблонных сетях CKB (слева) и LFR (справа).

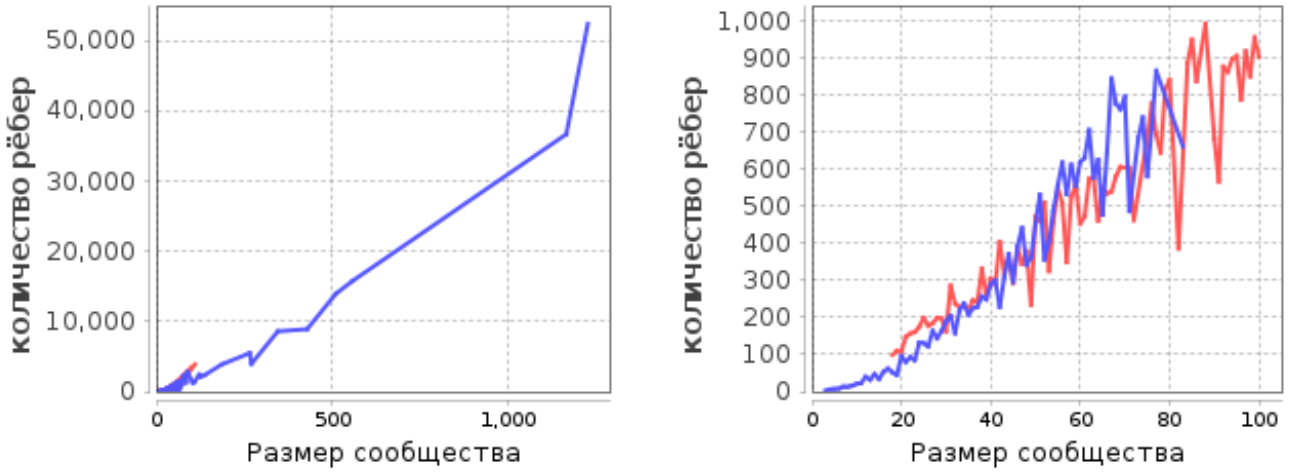


Рисунок А.26: *MOSES*: зависимость количества рёбер в найденных (синяя линия) и референтных (красная линия) сообществ от их размеров на шаблонных сетях CKB (слева) и LFR (справа).

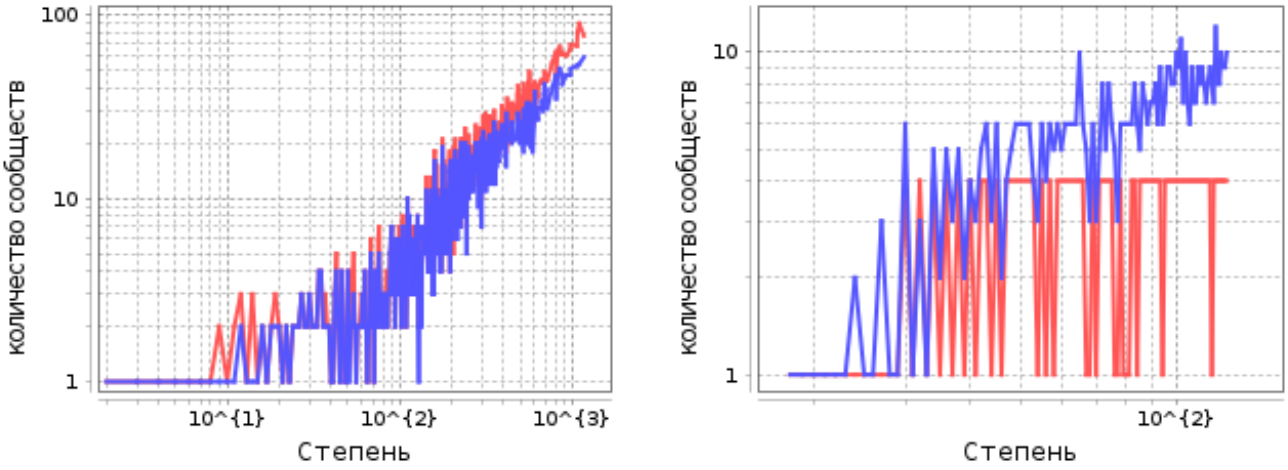


Рисунок А.27: MOSES: зависимость количества найденных (синяя линия) и референтных (красная линия) сообществ от степени вершины на шаблонных сетях CKB (слева) и LFR (справа).

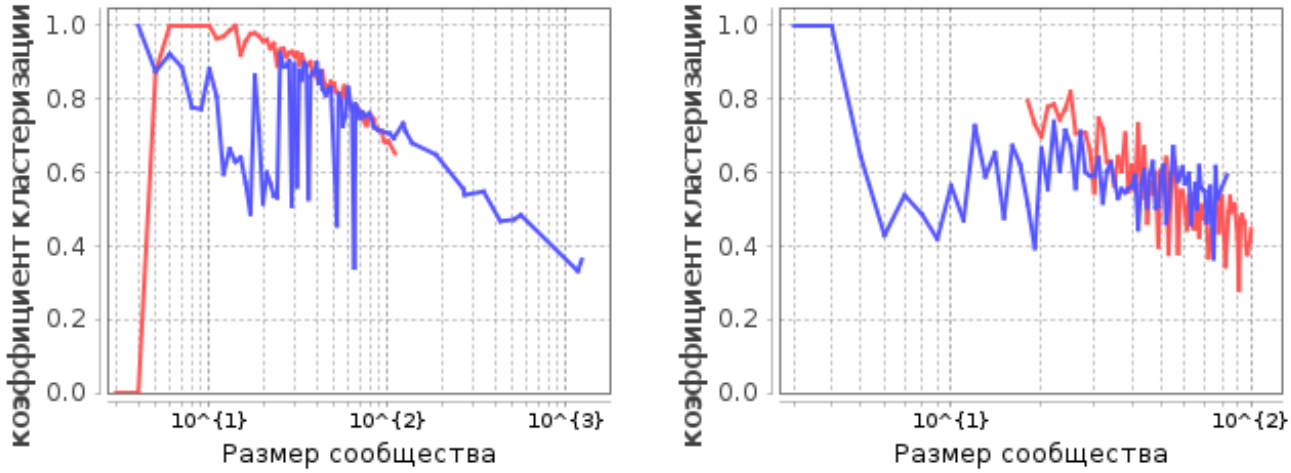


Рисунок А.28: MOSES: зависимость среднего коэффициента кластеризации от размера найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

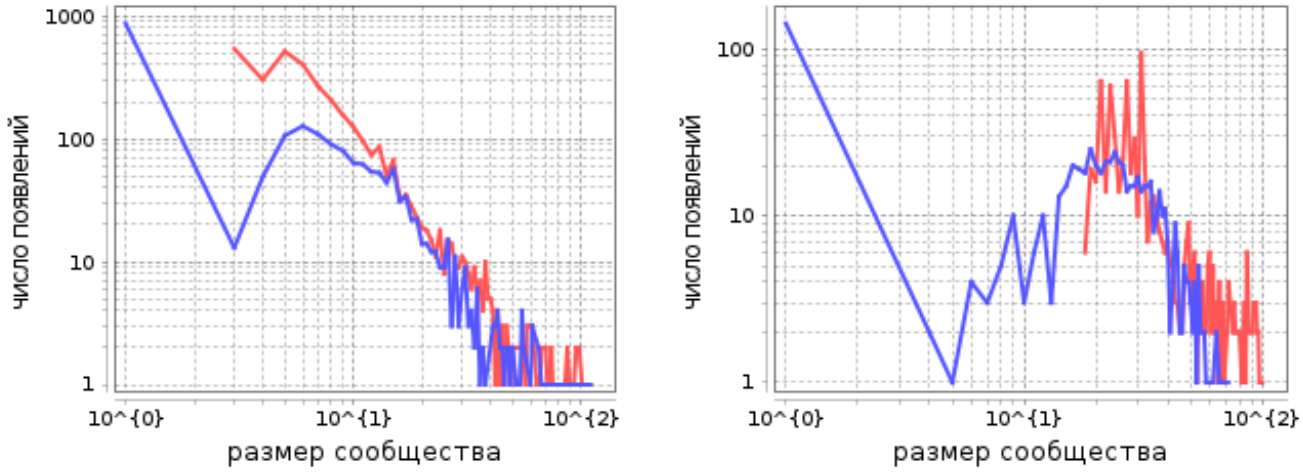


Рисунок А.29: OSLOM: распределение размеров найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

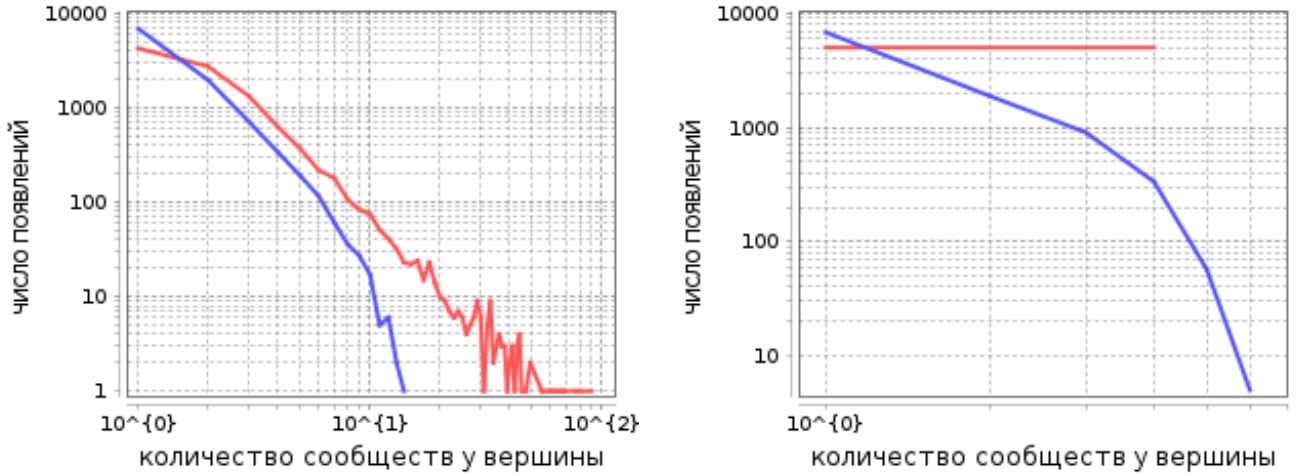


Рисунок А.30: OSLOM: распределение количества найденных (синяя линия) и референтных (красная линия) сообществ у пользователя на шаблонных сетях CKB (слева) и LFR (справа).

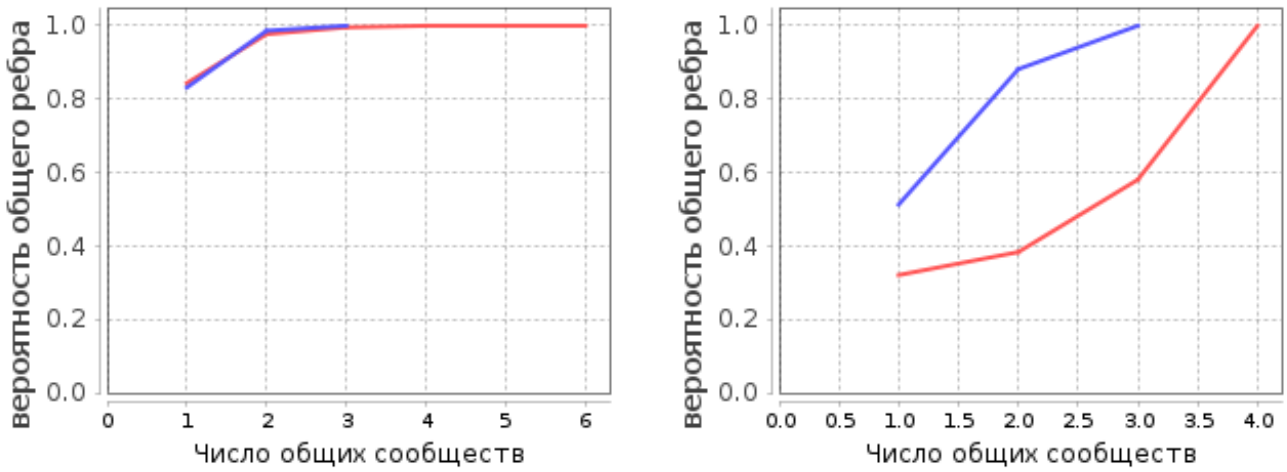


Рисунок A.31: OSLOM: зависимость вероятности ребра от количества общих сообществ у его концевых вершин для найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

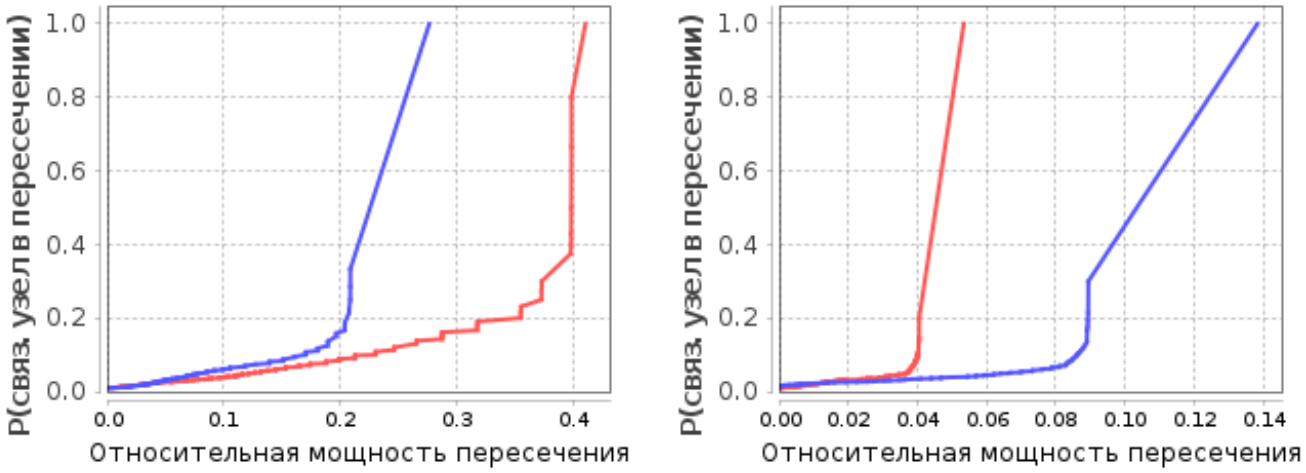


Рисунок A.32: OSLOM: зависимость вероятности появления связующей вершины в пересечении найденных (синяя линия) и референтных (красная линия) сообществ от относительной мощности пересечения на шаблонных сетях CKB (слева) и LFR (справа).

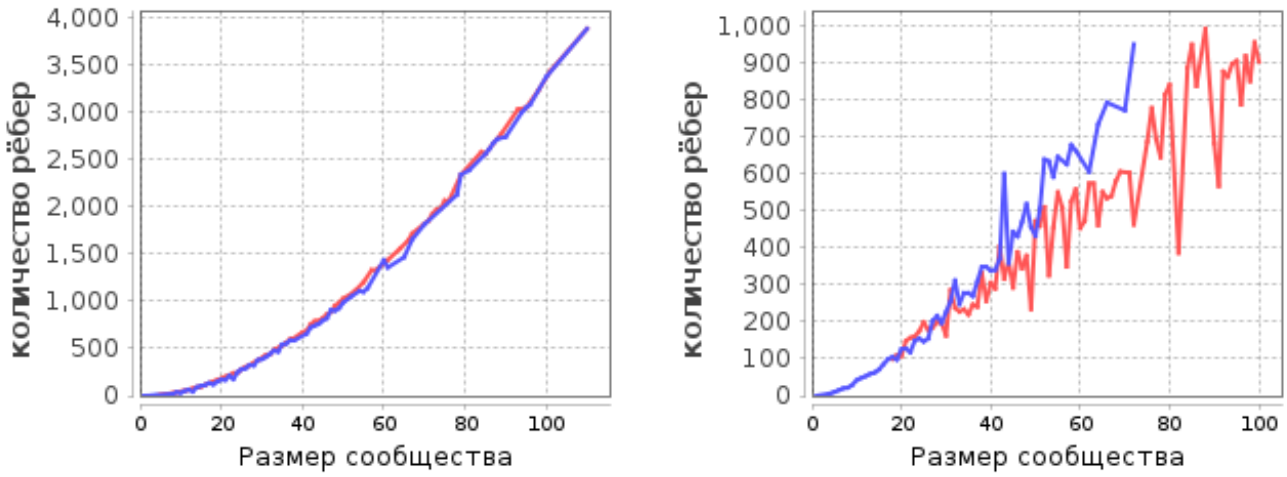


Рисунок А.33: OSLOM: зависимость количества рёбер в найденных (синяя линия) и референтных (красная линия) сообществ от их размеров на шаблонных сетях CKB (слева) и LFR (справа).

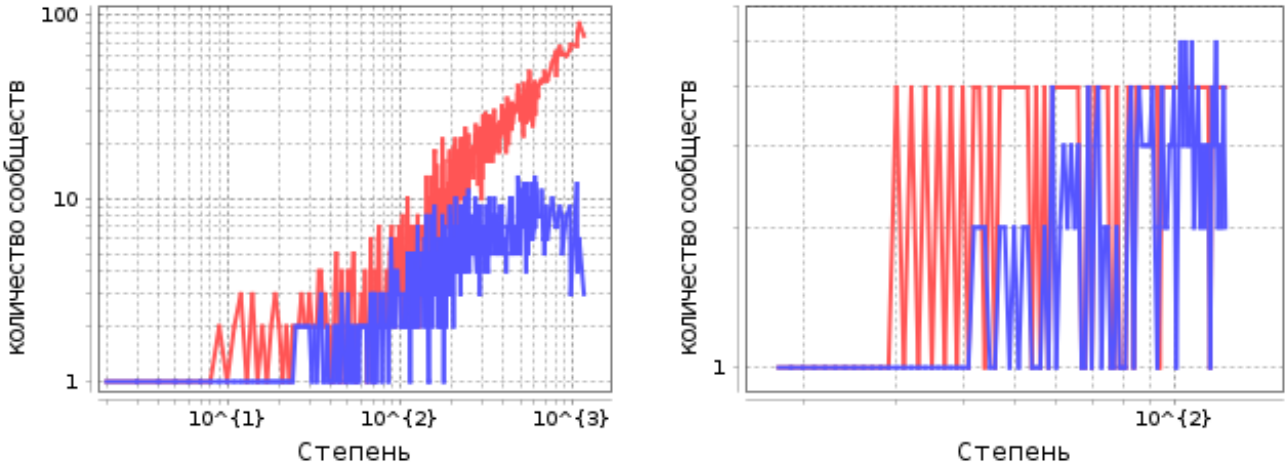


Рисунок А.34: OSLOM: зависимость количества найденных (синяя линия) и референтных (красная линия) сообществ от степени вершины на шаблонных сетях CKB (слева) и LFR (справа).

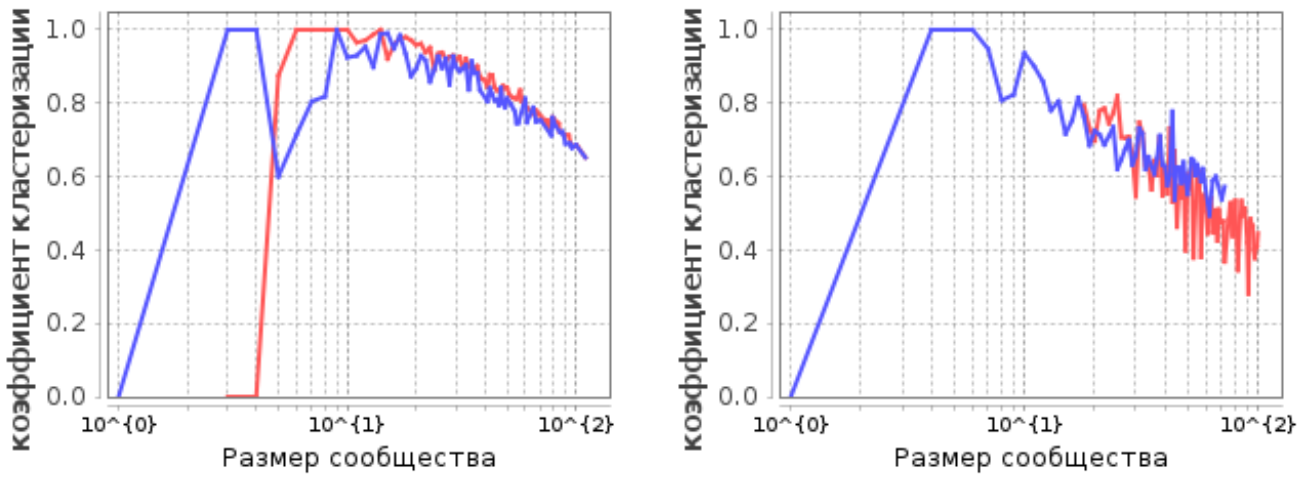


Рисунок А.35: OSLOM: зависимость среднего коэффициента кластеризации от размера найденных (синяя линия) и референтных (красная линия) сообществ на шаблонных сетях CKB (слева) и LFR (справа).

Приложение В

Метрики качества сообществ

В данном приложении содержатся результаты экспериментального исследования сообществ с помощью метрик качества (раздел 1.4).

Исследованы покрытия сообществ, найденные различными методами, в сравнении с референтными покрытиями, синтезированными методами CKB и LFR.

На всех графиках красная линия соответствует референтным, а синяя – алгоритмически найденным сообществам.

Выводы по результатам анализа полученных графиков содержатся в разделе 4.6.4.

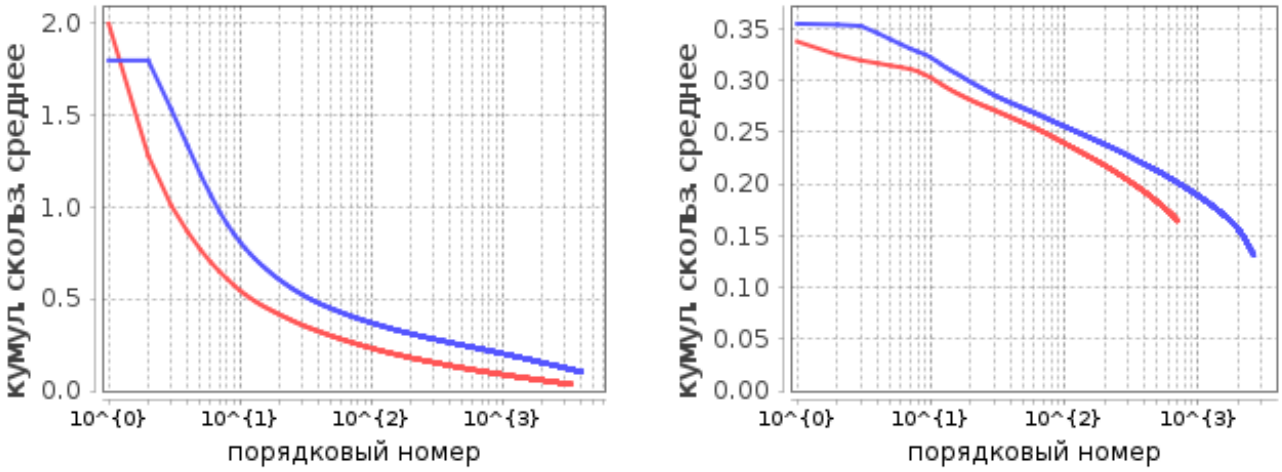


Рисунок B.1: EgoLP: сравнение отделимости найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

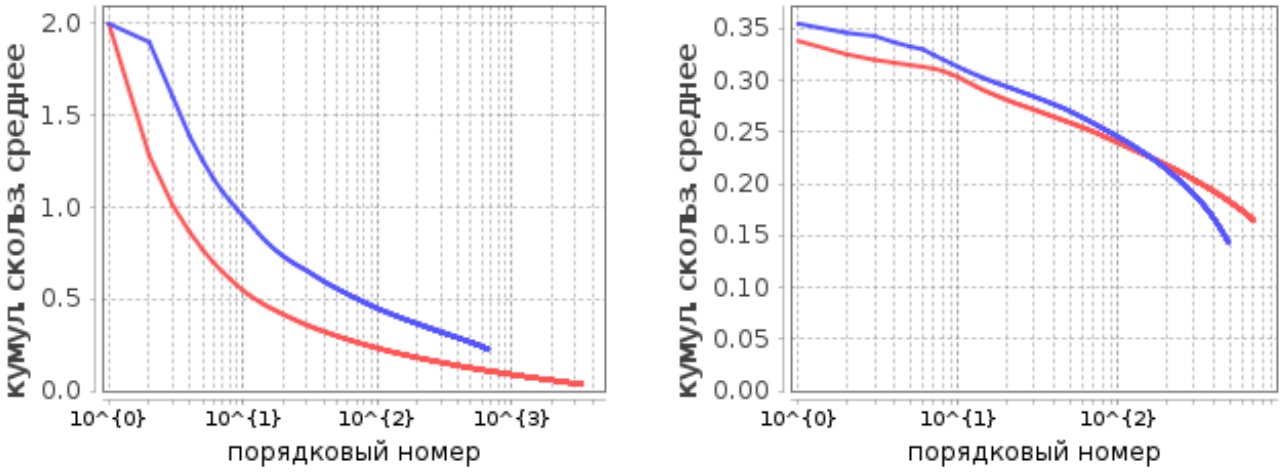


Рисунок B.2: SLPA: сравнение отделимости найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

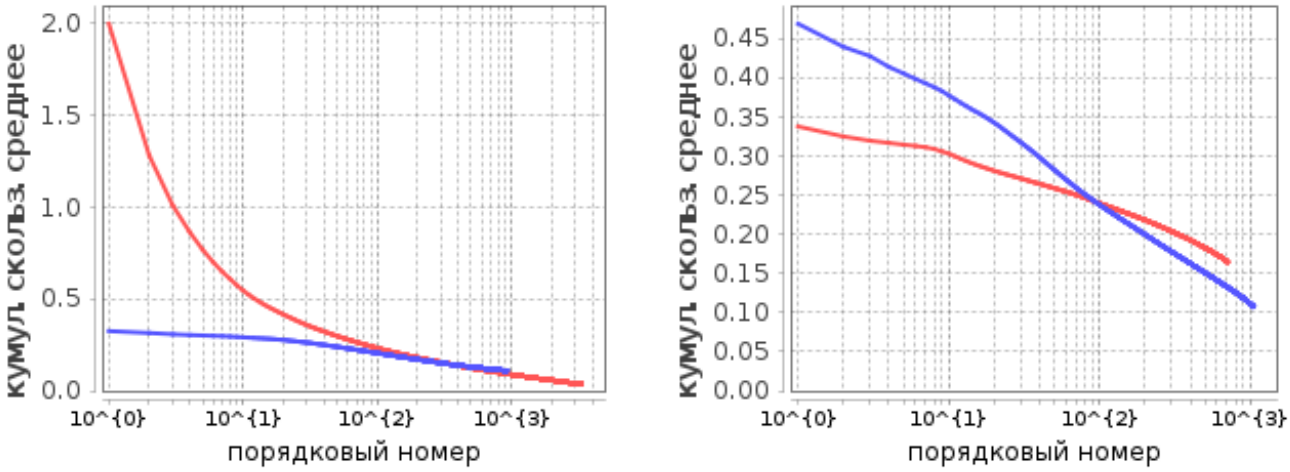


Рисунок B.3: GCE: сравнение отделимости найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

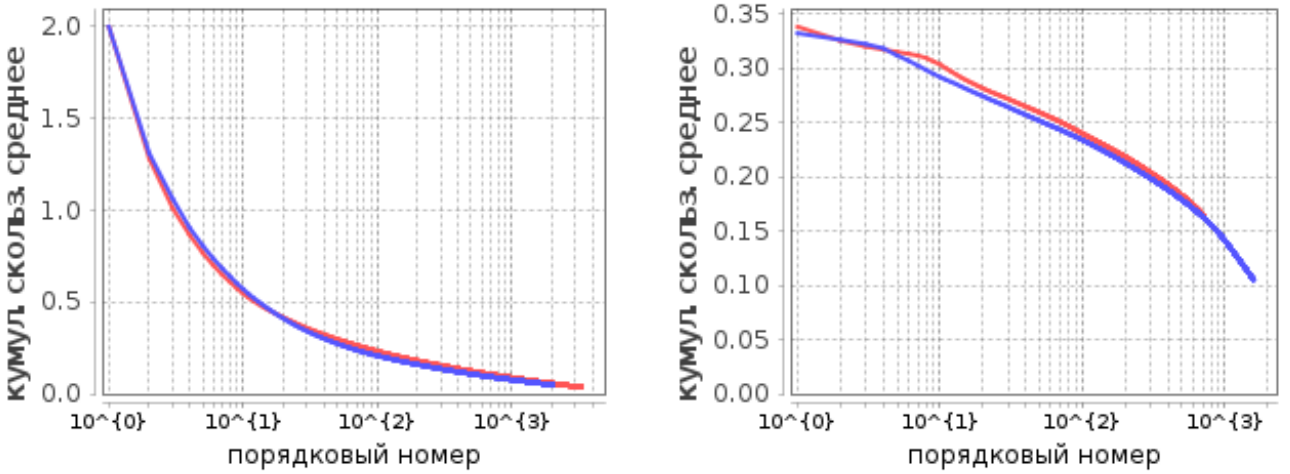


Рисунок B.4: *MOSES*: сравнение отделимости найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

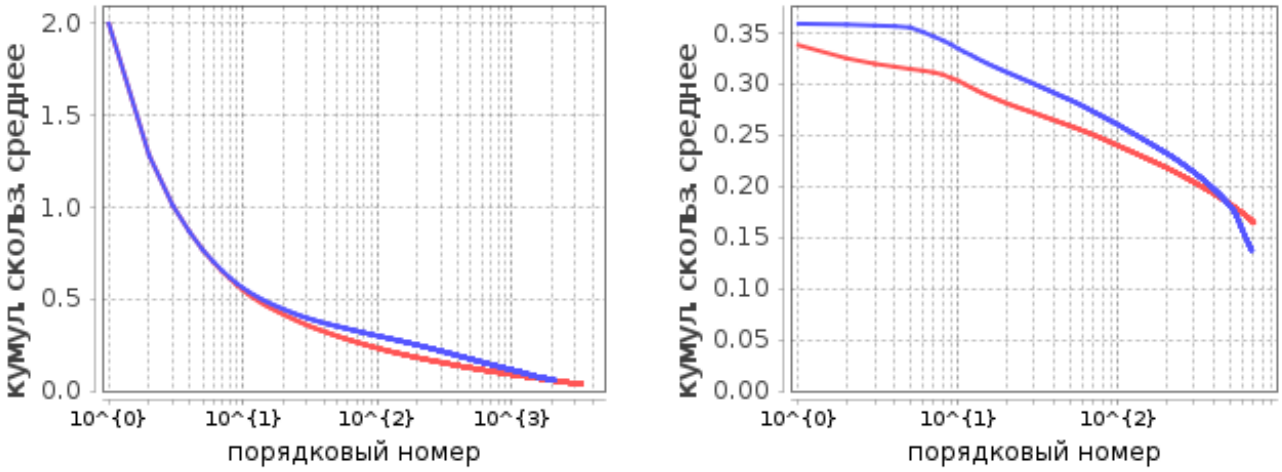


Рисунок B.5: OSLOM: сравнение отделимости найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

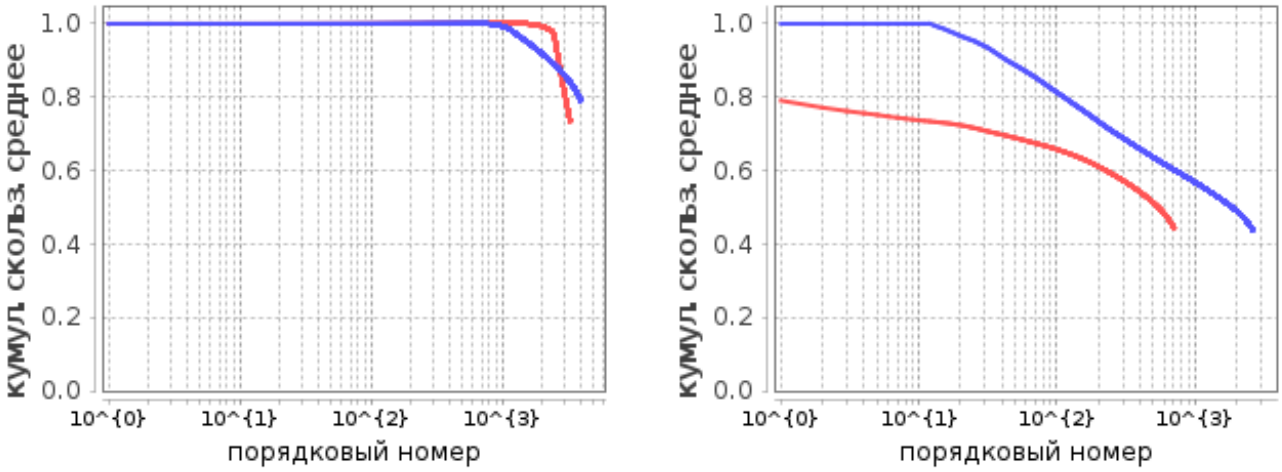


Рисунок B.6: EgoLP: сравнение плотности найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

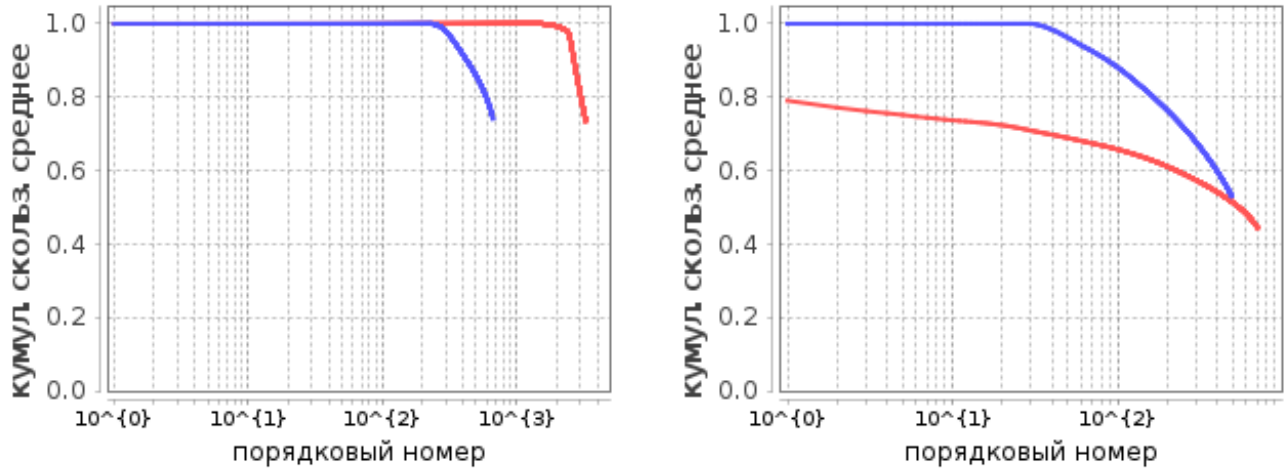


Рисунок B.7: SLPA: сравнение плотности найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

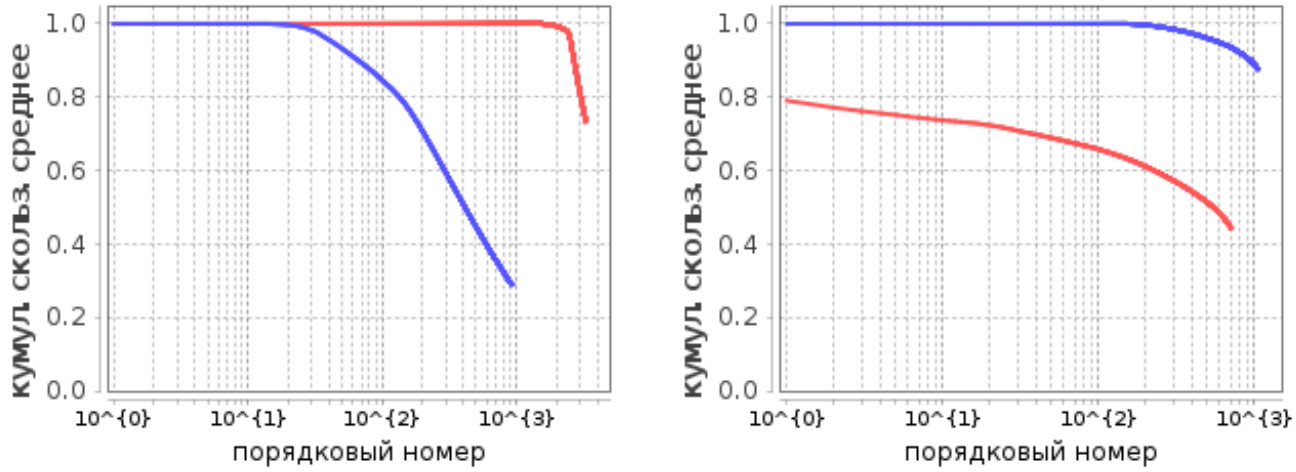


Рисунок B.8: GCE: сравнение плотности найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

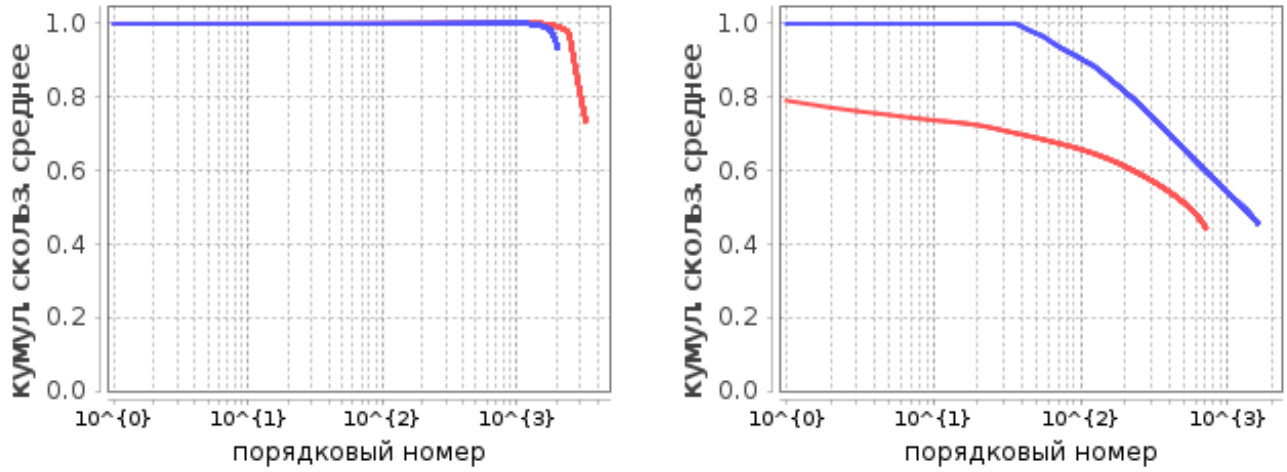


Рисунок B.9: MOSES: сравнение плотности найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

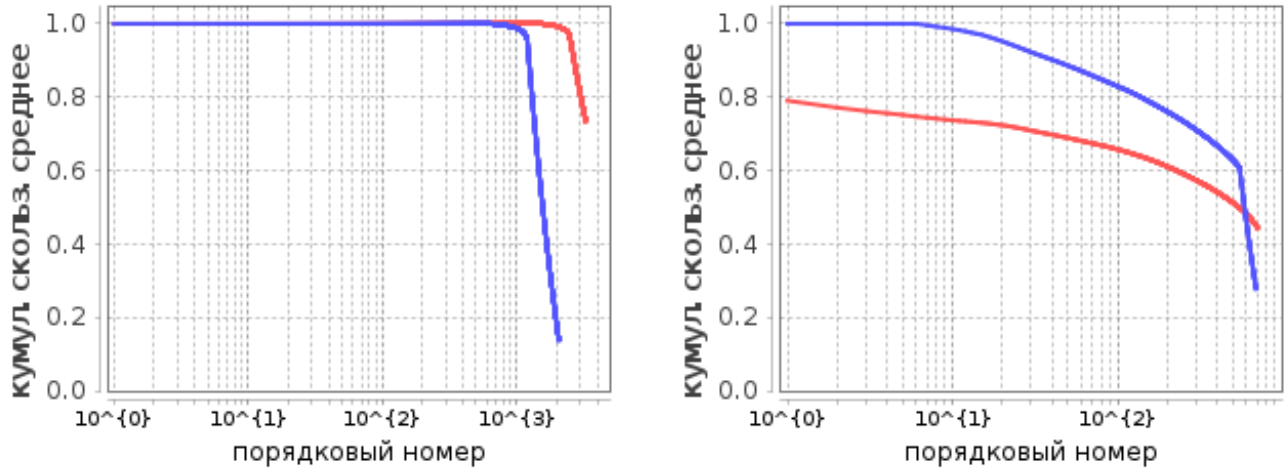


Рисунок B.10: OSLOM: сравнение плотности найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

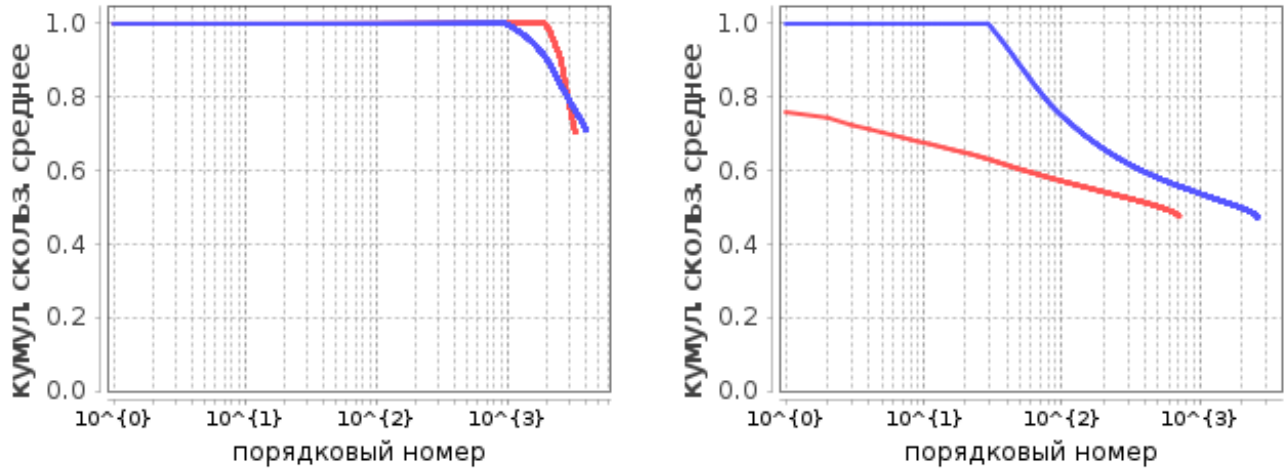


Рисунок B.11: EgoLP: сравнение сплочённости найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

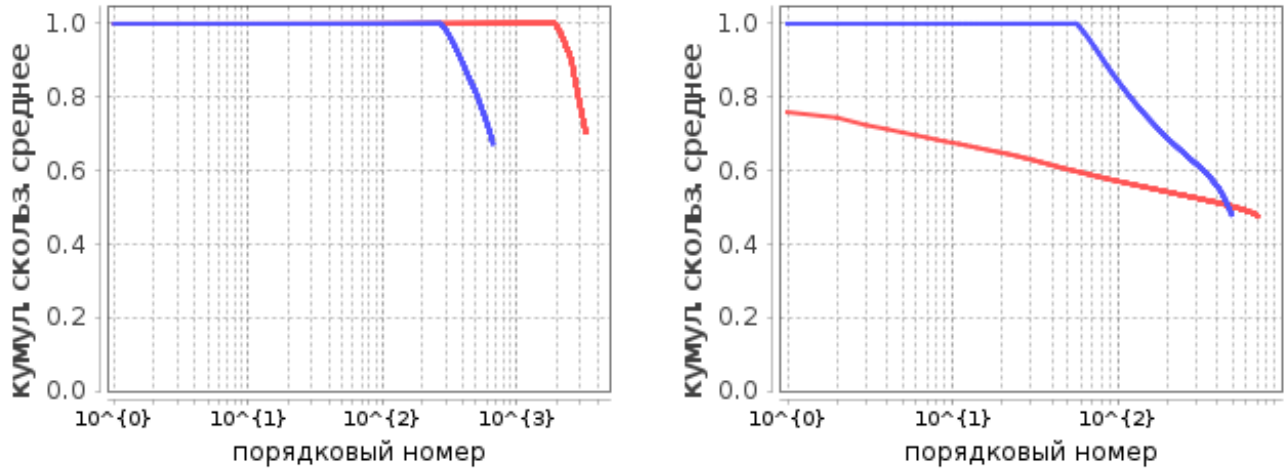


Рисунок B.12: SLPA: сравнение сплочённости найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

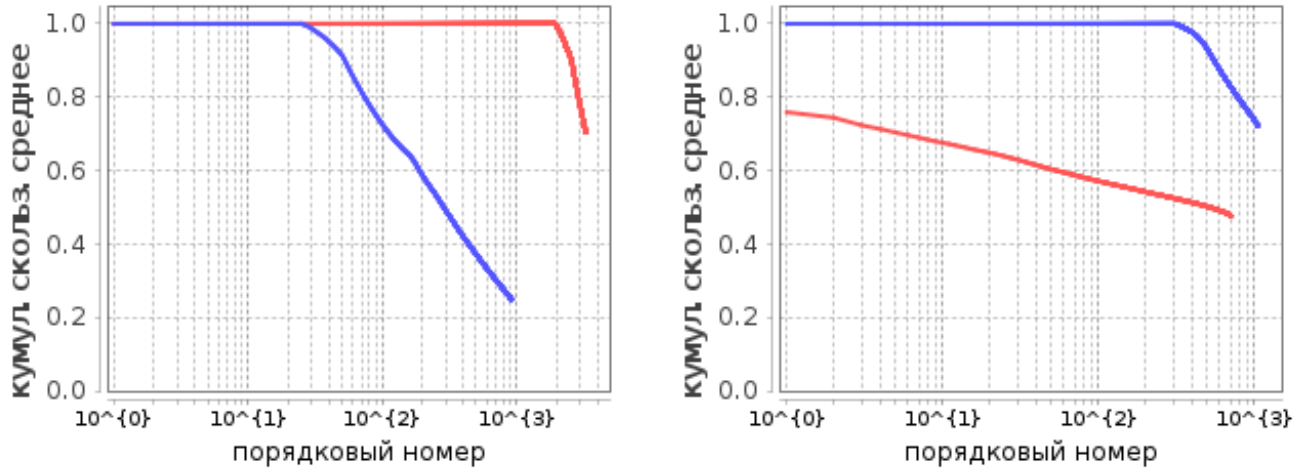


Рисунок B.13: GCE: сравнение сплочённости найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

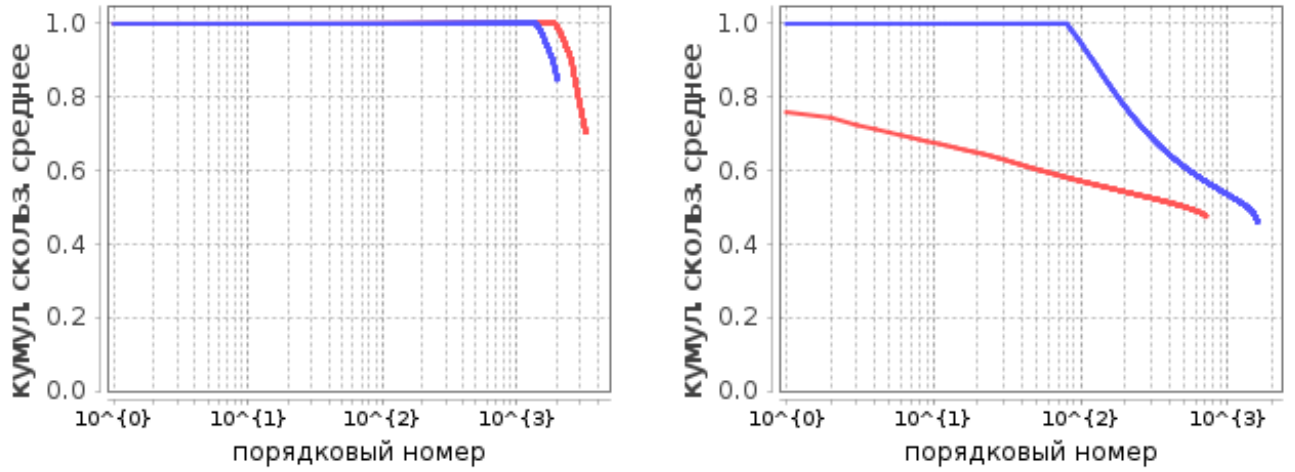


Рисунок B.14: *MOSES*: сравнение сплочённости найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

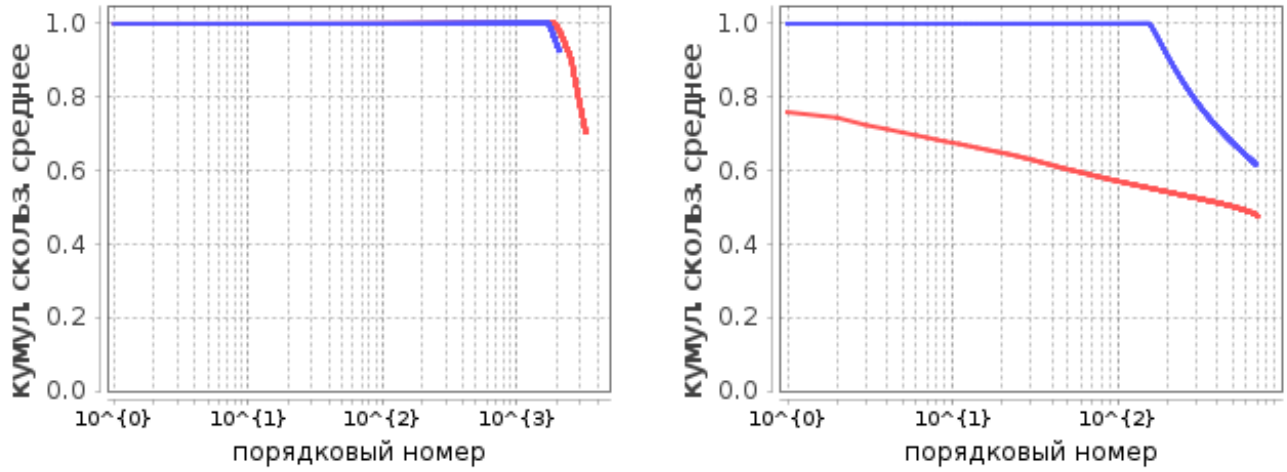


Рисунок B.15: OSLOM: сравнение сплочённости найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

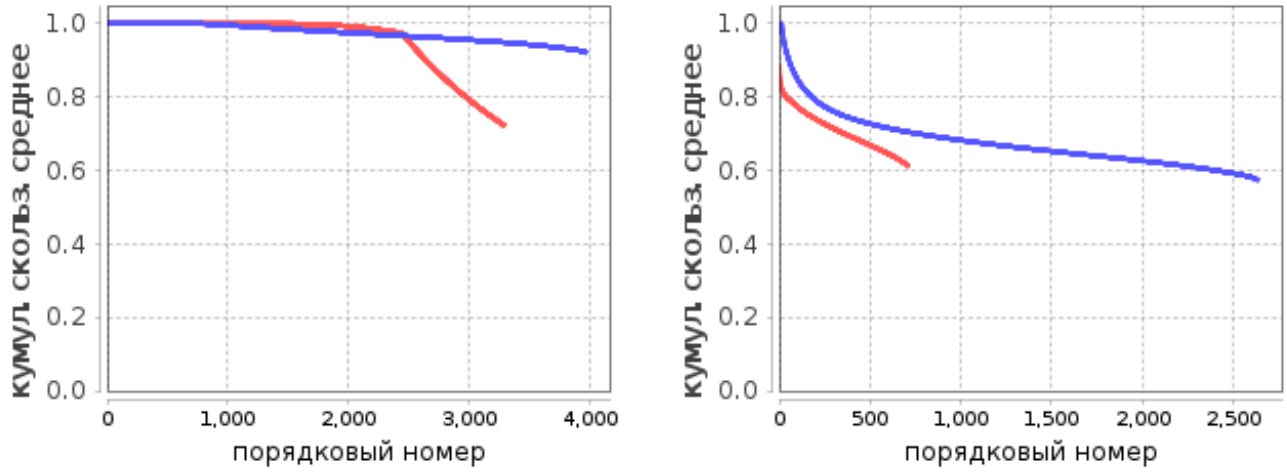


Рисунок B.16: EgoLP: сравнение коэффициента кластеризации найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

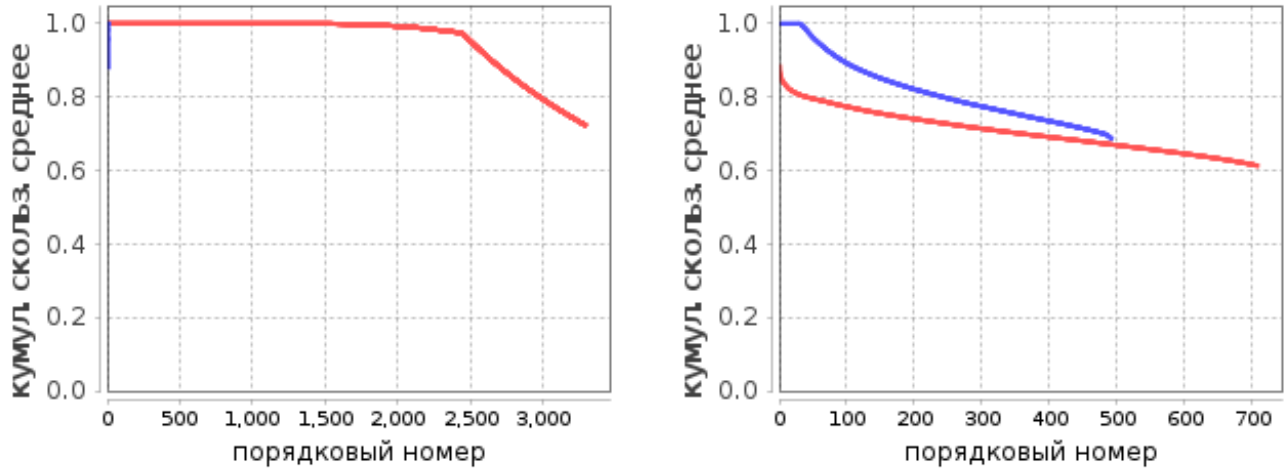


Рисунок B.17: SLPA: сравнение коэффициента кластеризации найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

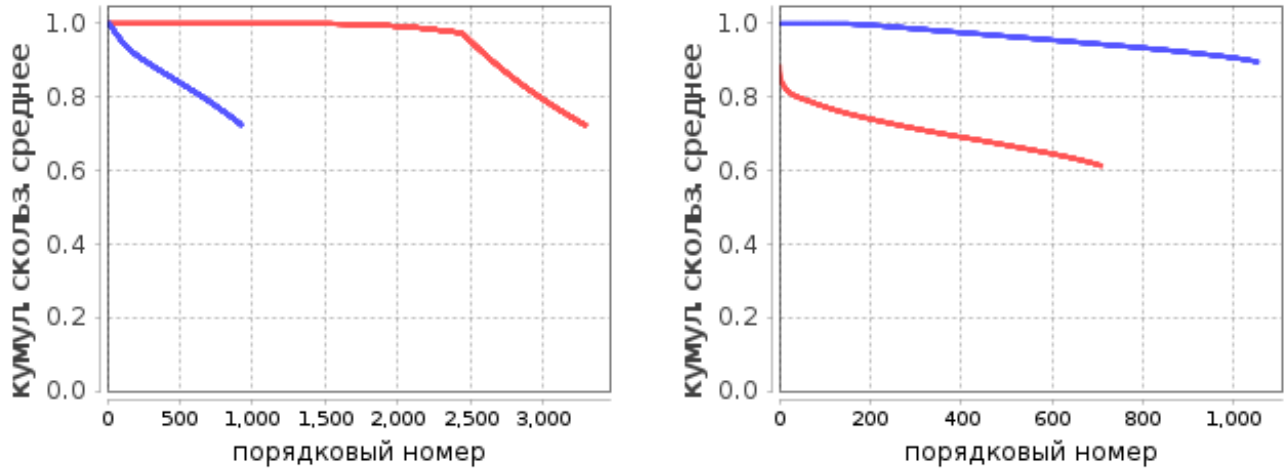


Рисунок B.18: GCE: сравнение коэффициента кластеризации найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

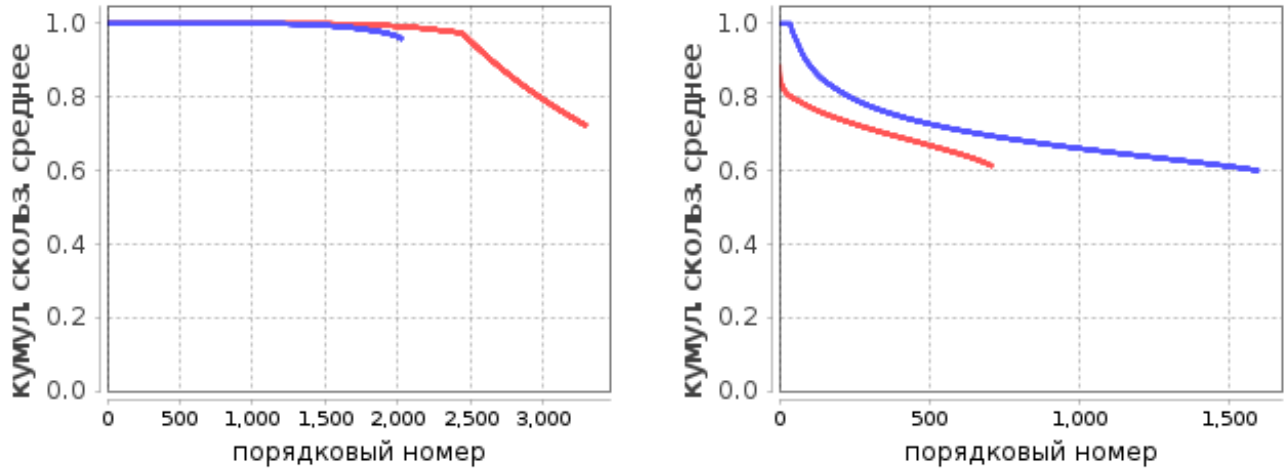


Рисунок B.19: MOSES: сравнение коэффициента кластеризации найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).

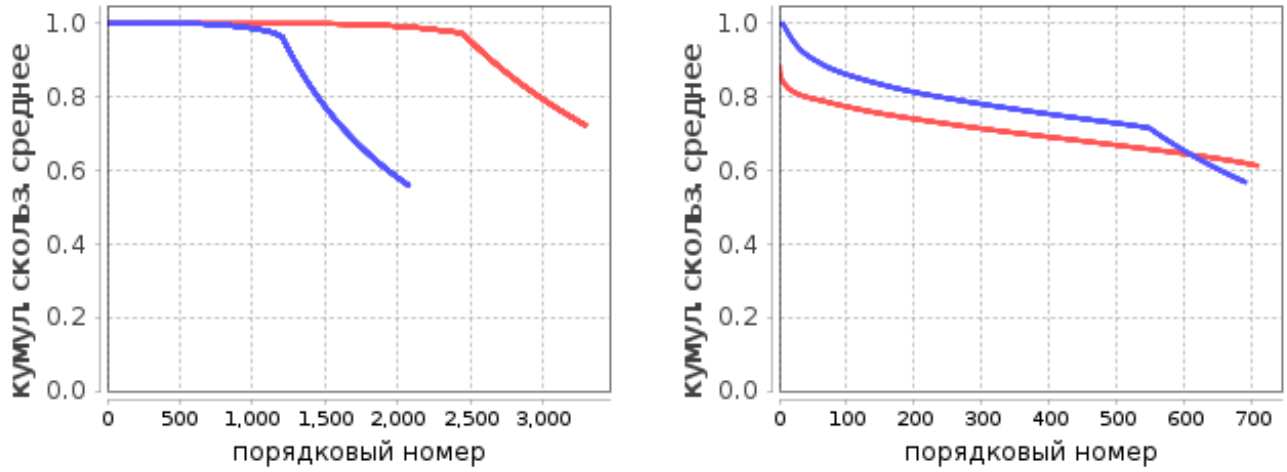


Рисунок B.20: OSLOM: сравнение коэффициента кластеризации найденных сообществ (синяя линия) с референтными (красная линия) на шаблонных сетях CKB (слева) и LFR (справа).