Как видим, несмотря на негативные тенденции в экономике и политике последнего десятилетия, уровень жизни населения, комплексно трактуемый, в течение исследуемого периода непрерывно возрастает. Данный вывод, оговоримся, справедлив при использовании официальных статистических данных Росстата.

Итак, в данной работе предложен концептуальный подход к оценке уровня жизни как многокритериальной задаче. Представлена многовариантная реализация решения такой задачи, включая этап отбора предикторных переменных, их фильтрации, а также этапа получения интегрального показателя.

Литература

1. Индекс развития человеческого потенциала. Гуманитарная энциклопедия: Исследования. Центр гуманитарных технологий, 2006–2019 (последняя редакция: 23.04.2019) // [Электронный ресурс] – Режим доступа: https://gtmarket.ru/ratings/human-development-index/human-development-index-info (дата обращения 04.06.2019).
2. Федеральная служба государственной статистики // [Электронный ресурс] – Режим доступа: http://www.gks.ru/wps/wcm/connect/rosstat_main/rosstat/ru/statistics/publications/catalog/doc_1138623506156 (дата обращения 10.06.2019).
3. Лотов А.В., Поспелова И.И. Многокритериальные задачи принятия решений: Учебное пособие. – М.: МАКСПресс,2008. – 197 с.

УДК 519.17

АГРЕГИРОВАННЫЙ ПОДХОД ВЫДЕЛЕНИЯ СООБЩЕСТВ В СОЦИАЛЬНЫХ СЕТЯХ НА ОСНОВЕ БАЗОВЫХ АЛГОРИТМОВ

Кочкаров Азрет Ахматович (akochkar@gmail.com)
Калашников Никита Владимирович
Кочкаров Расул Ахматович
ФГБОУ ВО «Финансовый университет при Правительстве Российской Федерации»

В работе рассматривается задача выделения сообществ в социальных сетях. Рассмотрен графовый подход исследования социальных сетей. Приведены известные алгоритмы выделения сообществ в качестве базовых алгоритмов. Предложен агрегированный подход выделения сообществ в социальных сетях на основе базовых алгоритмов. Рассмотрен инструментарий динамических и предфрактальных графов для моделирования и исследования социальных сетей, в том числе для поиска сообществ.

Ключевые слова: социальные сети, сообщества, алгоритмы, агрегированный подход, динамический граф, предфрактальный граф.

Статья подготовлена по результатам исследований, выполненных за счет бюджетных средств по государственному заданию Финуниверситету.

Введение

Исследованием социальных сетей занимаются научные группы ведущих зарубежных университетов – Оксфорд, Стенфорд[1], MIT. Транснациональные компании, владеющие социальными сетями такими как Facebook и Youtube, инвестируют в разработку технологий обработки и анализа больших объемов пользовательских данных. Исследовательские центры по всему миру используют данные социальных сетей для моделирования социальных, экономических, политических и других процессов от персонального до государственного уровня с целью разработки механизмов воздействия на эти процессы, а также создания инновационных аналитических и бизнес-приложений и сервисов.

Вместе с тем, при работе с социальными данными необходимо принимать во внимание такие факторы, как нестабильность качества пользовательского контента (спам и ложные аккаунты), проблемы с обеспечением приватности личных данных пользователей при хранении и обработке, а также частые обновления пользовательской модели и функционала. Всё это требует постоянного совершенствования алгоритмов решения различных аналитических задач.

В данной работе рассматривается одна из актуальных задач анализа социальных сетей – выделение сообществ. Наличие сообществ является характерной особенностью социальных сетей и выделение структуры сообществ позволяет исследовать такие проблемы как обнаружение преступных групп, выявление ботов пропагандистов, сегментация пользователей для увлечения эффективности контекстной рекламы, рекомендательных систем и многие другие.

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

1. Алгоритмы разбиения графа на сообщества

Иерархический алгоритм. Иерархический алгоритм работает по следующему принципу – каждая вершина графа приравнивается одному сообществу. На следующей итерации пара смежных вершин с ребром максимального веса объединяется в новое сообщество. Вес ребра представляет собой меру «центральности» в графе. Существуют и другие способы задания центральности ребра. В ходе работы алгоритма строится последовательность вложенных друг в друга разбиений, результат может быть представлен в виде дендрограммы. На рисунке 1 изображена дендрограмма, полученная в результате работы иерархического алгоритма. Такой алгоритм имеет ряд недостатков, например, он склонен выделять периферийные вершины большого сообщества в отдельные сообщества [1].

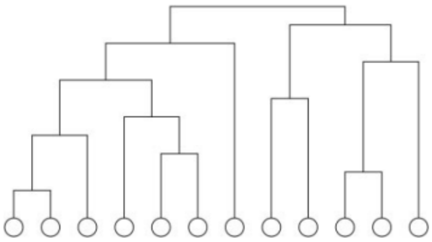

Рисунок 1. Дендограмма. Результат работы иерархического алгоритма выделения сообществ. Окружности в основании соответствуют вершинам графа, а само дерево отображает порядок объединения вершин в сообщества

Алгоритм Гирван-Ньюмана предложен Гирван М. и Ньюменом М. в 2002 году и основан на выделении «мостов» между сообществами [2]. Алгоритм осуществляет поиск ребер с наибольшей мерой центральности в графе, которые представляют собой «мосты» между сообществами. В отличие от иерархичесокго алгоритма, где сообщества пошагово объединяются, в этом алгоритме они распадаются. Последовательное удаление ребер-мостов разделяет граф на отдельные сообщества. Авторами была введена специальная величина для определения меры посредничества ребра – центральность по посредничеству для ребра, аналогично центральности по посредничеству вершины [3].

Центральность по посредничеству для вершины i определяется как количество кратчайших путей между всеми парами вершин графа, проходящих через вершину i. Данная величина отражает важность вершины сети с точки зрения востребованности вершины и объема информации, проходящего через нее. Понятие центральности по посредничеству для ребра определяется следующим образом. Пусть между парой вершин существует N кратчайших путей. Тогда каждому ребру этого пути присваивается вес $1/N$. Посчитав такие веса для каждого пути и просуммировав их, получим значения центральности по посредничеству для каждого ребра. На рисунке 2 приведен пример графа с подсчитанными значениями центральности по посредничеству для каждого ребра. Максимальное значение соответствует ребру, соединяющему два сообщества.

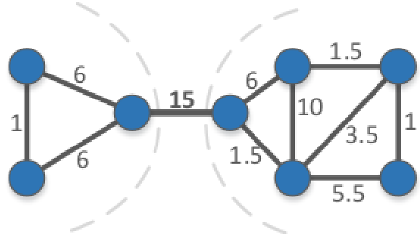

Рисунок 2. Граф с рассчитанными значениями меры центральности по посредничеству для ребер. Удаление ребра со значением 15 приводит к появлению двух связных компонент, представляющих сообщества

Основным недостатком алгоритма является его вычислительную сложность. Определение центральности для всех ребер занимает время $O(mn)$, где m и n – число ребер и вершин графа. Время работы самого алгоритма Гирван-Ньюмана составляет $O(m^2n)$. Однако, это время работы в худшем случае, в действительности не требуется на каждом шаге пересчитывать значение центральности для каждого ребра, а лишь для тех путей, которые подверглись изменению. Обычно это ребра на вершинах из одной компоненты связности.

Алгоритмы определения сообществ через модулярности. Понятие модулярности на данный момент это часто используемая и хорошо исследованная мера качества разбиения в задаче выделения сообществ. Модулярность определяется как доля ребер от всего числа ребер, которые попадают в данные группы минус ожидаемая доля ребер, которые попали бы в те же группы при случайном распределении ребер. Значение модулярности лежит в интервале $[-1,1]$. Модулярность является положительной в случае, если количество ребер в группах достигает ожидаемого числа. Для заданного разбиения вершин графа на сообщества модулярность отражает концентрацию ребер в сообществах по сравнению со случайным распределением ребер между всеми вершинами, то есть, большее значение модулярности свидетельствует о лучшем разбиении графа [5].

Таким образом, алгоритмы оптимизации модулярности в задачах выделения сообществ исследуют различные значения модулярности с целью подобрать наилучшие разбиения графа – сообщества. Полный перебор всевозможных разбиений графа с вычислением значений модулярности затруднителен, задача перебора не является полиномиальной, а относится по всей видимости к классу NP-трудных задач. Доказательство принадлежности задачи поиска максимальной модулярности графа к классу NP-трудных задач требует отдельного изучения. Однако существует ряд алгоритмов, способных находить значение модулярности приближенное к максимальному, за полиномиальное время.

Алгоритм Ньюмана. Это иерархический алгоритм кластеризации, в ходе итерации которого группы вершин объединяются в новую, если в результате такого слияния увеличивается модулярность. Алгоритм начинает работу из состояния, когда каждой вершине соответствует по одному сообществу. Ребра изначально отсутствуют. Они добавляются в ходе процедуры одна за другой. Добавление самого первого ребра снижает количество сообществ с n до $(n-1)$, образуя новое разбиение графа, где n – общее число вершин графа. Ребро выбирается таким образом, что вновь получившееся разбиение графа увеличивало бы значение модулярности по сравнению с предыдущим.

На каждой итерации алгоритма для выбора оптимального варианта объединения вычисляется величина изменения модулярности при слиянии всех пар сообществ. Слияние сообществ между которыми нет ребер не даст прироста модулярности, поэтому рассматриваются только те группы, между которыми присутствуют ребра. Число таких объединений не более m – числа ребер графа. Вычисление величины изменения модулярности выполняется за время $O(m)$, что соответствует вычислительной сложности одной итерации алгоритма. После принятия решения об объединении сообщества в новые, обновляется матрица, элементы которой равны доле ребер в графе между сообществами i и j для текущего разбиения. Вычислительная сложность этой процедуры равна $O(n)$. Алгоритм Ньюмана выполняет n итераций, тогда его вычислительная сложность равна $O\big((m+n)n\big)$. Для разряженных графов вычислительная сложность принимается за $O(n^2)$. Алгоритм Ньюмана позволяет выделять сообщества на значительно больших графах по сравнению с алгоритмом Гирван-Ньюмана.

Алгоритм Fastgreedy состоит в жадной оптимизации модулярности и фактически представляет собой модификацию алгоритма Ньюмана. Так как матрицы смежности графов обычно разряжены, в процессе обновления матриц смежности сообществ возможно сократить лишние операции. Использование специальной структуры данных – «двоичной кучи», позволило сократить вычислительную сложность алгоритма до $O(nlog^2n)$. Модификация позволила увеличить размерность исследуемых графов на порядок, вплоть до 10^6 вершин.

Алгоритм Лювена (Multilevel). Еще один алгоритм жадной оптимизации модулярности был предложен группой ученых из бельгийского университета Лювена [7], который работает также и для взвешенных графов. Алгоритм состоит из двух итеративно повторяющихся частей.

В первой части работы алгоритма каждой вершине ставится в соответствие одно сообщество. Далее для каждой вершины i перебирается список всех ее соседей j, считаются значения модулярности от их слияния. Вершина i затем перемещается в сообщество вершины j, если найдено значение модулярности максимальное и положительное. Если все значения модулярностей отрицательны, вершина остается в своем сообществе. Процесс повторяется последовательно для всех вершин снова и снова до тех пор, пока не удастся достичь увеличения ни на одном объединении.

Во второй части алгоритм строит метаграф, вершинами которого являются сообщества, полученные в предыдущей части. В метаграфе каждому ребру ставится в соответствие вес, равный количеству ребер соединяющих вершины соответствующих сообществ. Ребра между вершинами одного и того же сообщества приводят к появлению петлей у вершин метаграфа. Как только такой метаграф построен, вновь запускается на нем первая часть алгоритма. Две части – локальная оптимизация модулярности и построение метаграфа объединяются в одну итерацию. Итерация выполняются последовательно до тех пор, пока возможно увеличить значение модулярности (рис.3).

Результатом работы алгоритма является иерархическая структура разбиения графа, где вершины сообщества верхнего уровня представляют собой сообщества предыдущего уровня. Высота такой иерархии определяется количеством итераций алгоритма, и обычно является не превосходит 7-8 уровней.

Компьютерные симуляции показали, что время работы алгоритма линейно зависит от размера сети $O(m)$. Алгоритм способен обрабатывать графы размером вплоть до 10^9 вершин за приемлемое время. Алгоритм Лювена обработал сеть www 2001 года, состоящую из 118 миллионов вершин и миллиарда ребер за 152 минуты, в то время как другие алгоритмы выходили за время более суток [7].

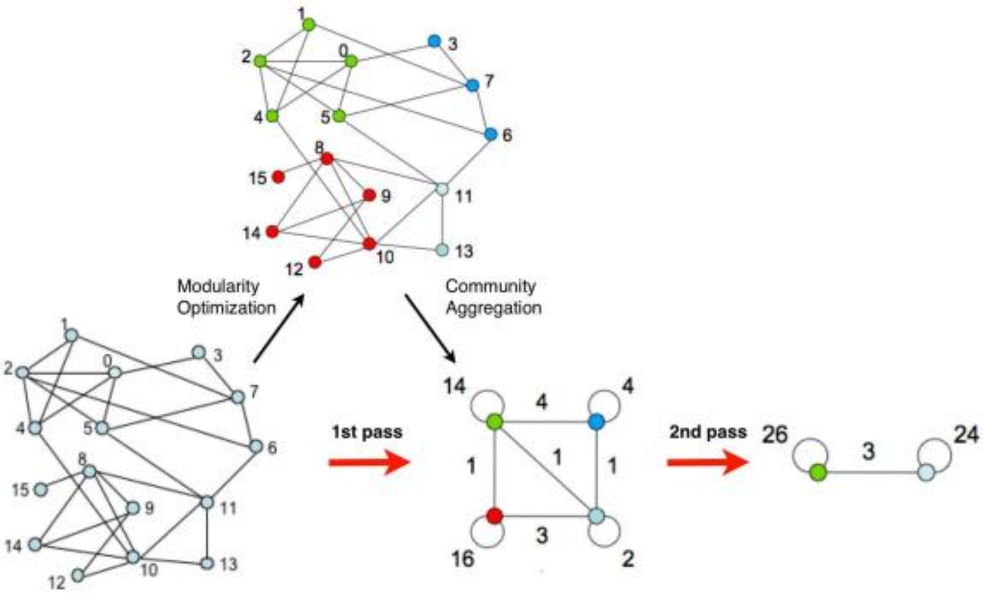

Рисунок 3. Визуализация работы алгоритма Лювена

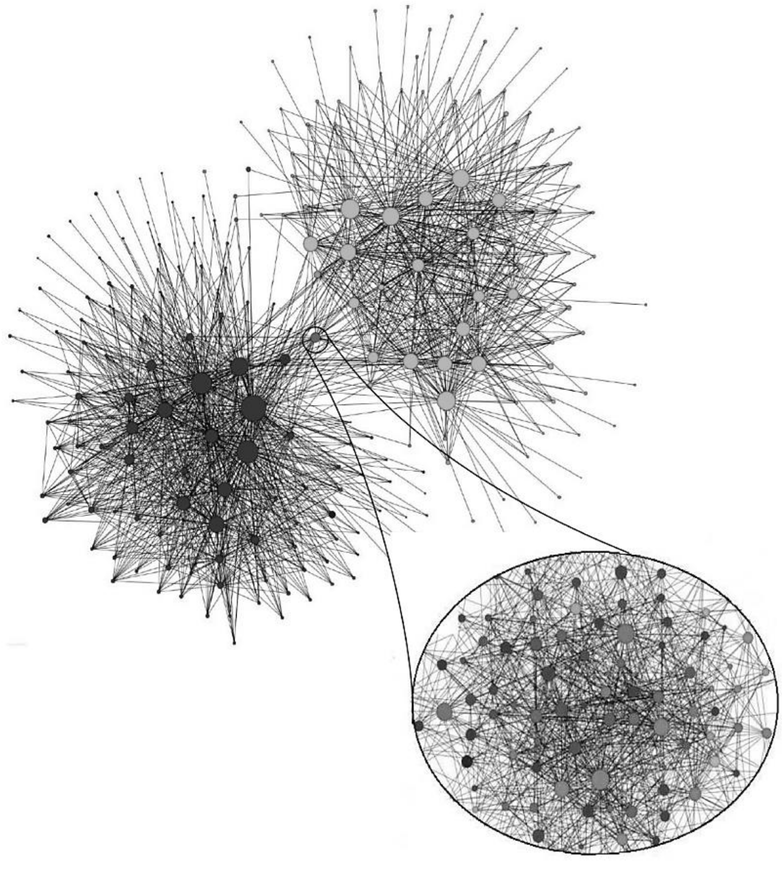

Рисунок 4. Визуализация сообществ Бельгийской мобильной сети
(https://arxiv.org/pdf/0803.0476.pdf)

На рисунке 4 представлена визуализация сообществ Бельгийской мобильной сети, состоящей из 2 миллионов пользователей. Размер круга вершины на рисунке пропорционально зависит от числа пользователей входящих в данное сообщество. Темно-серые вершины отображают сообщества пользователей, говорящих на французском языке, а светлые круги соответствуют немецко-говорящим пользователям. В качестве сообществ принимались группы с более 100 пользователей. Вершина-сообщество, соединяющее две большие компоненты, состоит из пользователей, говорящих на двух разных языках.

Алгоритм Label Propagation. Алгоритм распространения меток начинается с определения каждой вершины в качестве одного сообщества. На каждой итерации в случайном порядке обходятся все вершины графа, каждой из них ставится в соответствии метка, часто встречающаяся у соседей. Если такую метку найти не удалось, она выбирается случайным образом. При каждой последующей итерации метки распространяются в сети, большинство из них исчезает, в то время как оставшиеся будут доминировать. Алгоритм завершает свою работу, когда на очередной итерации не поменялось значение ни одной метки. Вершины с одинаковыми метками признаются сообществами. В силу наличия случайного выбора алгоритм не обязательно каждый раз сходится к одному и тому же разбиению, но такие разбиения похожи. Для алгоритма Label Propagation предложен способ агрегирования нескольких решений [8].

Для двух решений C^1 (четыре сообщества) и C^2 (три сообщества) использованы соответственно метки - $t_{1,}\ t_{2,}\ t_3,\ t_4$ и $s_{1,}\ s_{2,}\ s_3$. В агрегированном решении C вершины сообщества t_1 разбиваются на 2 сообщества с метками s_1, s_2, а вершины с метками s_3 разделяются на сообщества $t_{2,}\ t_{3,}\ t_4$ (рис. 5).

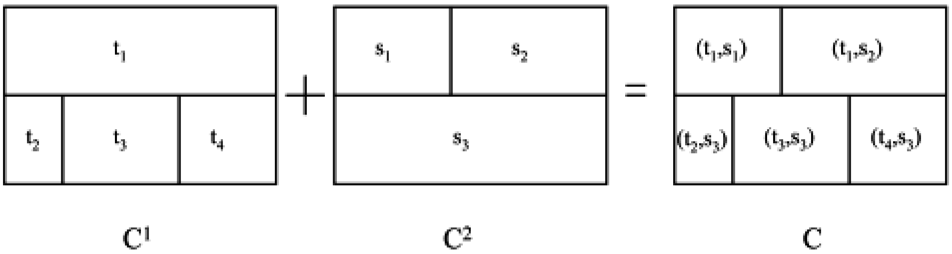

Рисунок 5. Агрегирование двух решений алгоритма Label Propagation

Преимуществом алгоритма является низкая вычислительную сложность $O(n)$. К недостаткам относится нестабильность решения, при этом алгоритм может формировать большие сообщества, не отражающие структуру сети. Рекомендуется запускать алгоритм несколько раз и далее формировать агрегированное решение. В настоящее время предлагаются различные подходы нивелирования недостатков алгоритма, что требует дополнительного исследования.

Алгоритм Walktrap. Идея алгоритма Walktrap базируется на физических принципах – сообщества в графе выделяются посредством динамических процессов, например, случайного блуждания в графе. Если структура сообществ такова, что плотность ребер внутри сообщества значительно выше плотности за ее пределами, то случайное блуждание, запущенное в одном из кластеров, будет достаточно долго блуждать среди его вершин, прежде чем покинет его и перейдет в следующий кластер.

В алгоритме Walktrap с помощью процесса случайного блуждания вычисляется мера сходства между вершинами. Сходство между вершинами i и j определяется как вероятность того, что случайный блуждатель переместится из вершины i в вершину j не более чем за t шагов. Если существует четко выраженная структура сообществ в графе, паре вершин из одного сообщества гораздо легче достичь друг друга в процессе случайного блуждания нежели паре вершин из разных сообществ. Поэтому ожидается, что сходство вершин будет значительно выше в кластерах, чем между ними. Мера сходства двух вершин i и j задается в виде: $r_{ij} = \sqrt{\left[\sum_{\kappa=1}^{n}\left(P_{ik}^t - P_{jk}^t\right)^2\right]/d_k}$, где $P_{ij} = \frac{A_{ij}}{d_i}$ – вероятность перехода из вершины i в j на одном шаге, A_{ij} – матрица смежности графа, d_i – степень вершины i, P_{ij}^t – вероятность перехода из вершины i в j за t шагов. Алгоритм Walktrap имеет вычислительную сложность $O(n^2 logn)$, а также чувствителен к выбору параметра t. Для первичного исследования социальной сети предлагается размещать параметр в интервале $3 \leq t \leq 8$ [9].

Алгоритм Leading Eigenvector. Алгоритм спектрального анализа матрицы смежности графа работает следующим образом. Задано множество n объектов $x_{1,}x_{2,}\dots, x_n$ с симметричной и неотрицательной мерой сходства S для любой пары x_i, x_j: $S(x_j, x_i) \geq 0,\ i,j = 1\dots n$. Спектральная кластеризацая включает в себя методы, которые разделяют объекты на кластеры с использованием собственных векторов матриц, таких как S, либо полученных из нее. В качестве объектов $x_{1,}x_{2,}\dots, x_n$ выступают вершины графа. Спектральная кластеризация состоит в проекции исходного набора объектов в евклидово пространство, в котором их координаты являются -мерными векторами, далее набор точек кластеризуется с помощью стандартных методов, таких как k-means [10]. В алгоритме Leading Eigenvector используется спектральная матрица $B_{ij} = A_{ij} - \frac{k_i k_j}{2m}$. Найдя два собственных вектора, соответствующих максимальным собственным значениям, можно спроецировать исходное пространство вершин графа в пространство R^2. Имея такое представление вершин графа, точки кластеризуются любым алгоритмом машинного обучения без учителя. Альтернативный подход разделения вершин графа на примере двух сообществ представлен на рис. 6. Точки разделены прямой, проходящей через начало координат таким образом, чтобы максимизировать модулярность разбиения [11]. Используя первые два собственных вектора, вершины представлены точками на плоскости. Разрезая плоскость прямой, проходящей через начало координат (на рисунке изображена пунктирной линией), получится разбиение с большим значением модулярности.

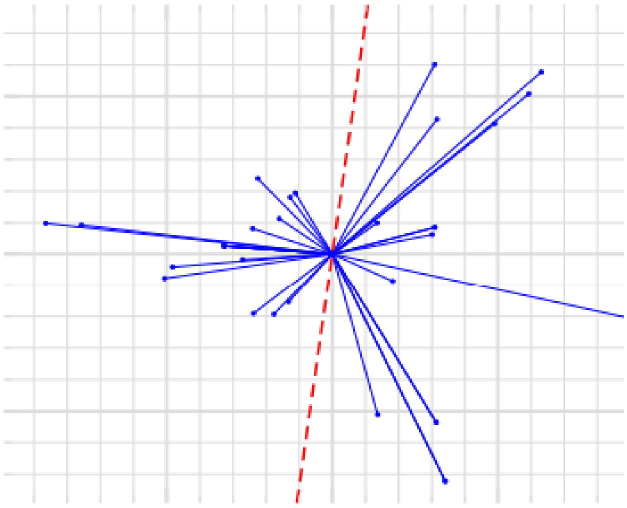

Рисунок 6. Применение спектрального алгоритма для оптимизации модулярности

2. Агрегирование базовых алгоритмов

В основе большинства базовых алгоритмов выделения сообществ лежат идеи, которые пришедшие в сетевую теорию из различных областей науки. Такие алгоритмы фактически выделяют сообщества сообществ, используя свойства сети. Для получения статистически качественного результата выделения сообществ, в данной работе предлагается объединить несколько алгоритмов в единый подход. Для реализации этой задачи использовались методы машинного обучения без учителя.

Обучение без учителя (Unsupervised learning) – один из разделов машинного обучения. Изучает широкий класс задач обработки данных, в которых известны только описания множества объектов (обучающей выборки), и требуется обнаружить внутренние взаимосвязи, зависимости, закономерности, существующие между объектами. Обучение без учителя часто противопоставляется обучению с учителем, когда для каждого обучающего объекта задаётся «правильный ответ», и требуется найти зависимость между объектами и ответами. В случае выделения сообществ в графе мы не знаем «правильных ответов», то есть к какому сообществу в действительности принадлежит данная вершина.

В нашем случае в качестве объекта выступает матрица признакового описания вершин графа, по строкам которой располагаются вершины графа, а по столбцам базовые алгоритмы, то есть, элемент матрицы равен $X_{ij} = method_j(v_i)$, где $v_i, i = 1,2,\ldots,n,$ – вершины графа социальной сети (табл. 1).

Таблица 1

Матрица признакового описания результатов работы базовых алгоритмов

№	*Fastgreedy*	*Infomap*	*Walktrap*	*Label Propogation*	*Multilevel*	*Eigenvector*
1	1	2	1	...	...	2
2	2	2	3	...	...	1
3	1	2	1	...	...	2
...	...	...		...	...	...
n	4	5	4	...	...	2

В качестве агрегирующего алгоритма был выбран метод средних. Алгоритм представляет собой версию EM-алгоритма, применяемого также для разделения смеси гауссиан. Он разбивает множество элементов векторного пространства на заранее известное число кластеров k. На каждой итерации вычисляются заново центры масс каждого кластера, полученного на предыдущем шаге, затем векторы разбиваются на кластеры вновь в соответствии с тем, какой из новых центров оказался ближе по выбранной метрике. Алгоритм завершается на итерации, когда не происходит изменения внутрикластерного расстояния. Это происходит за конечное число итераций, так как количество возможных разбиений конечного множества конечно, а на каждом шаге суммарное квадратичное отклонение уменьшается, поэтому зацикливание невозможно [12].

Алгоритм обладает рядом недостатков, в частности:

1) не гарантируется достижение глобального минимума суммарного квадратичного отклонения, а только одного из локальных минимумов;

2) Результат зависит от выбора исходных центров кластеров (их оптимальный выбор неизвестен);

3) Число кластеров следует заранее вычислить.

Проблема выбора числа кластеров может быть решена следующим образом: алгоритм последовательно запускается с числом кластеров, равным количеству сообществ, полученным из базовых алгоритмов. Проблема сходимости алгоритма в зависимости от начальных центров кластеров частично решается путем усреднения множества запусков алгоритма при различных начальных условиях. Так как алгоритм ищет евклидово расстояние между парами точек, данные в матрице признакового описания должны быть предварительно представлены в виде бинарных признаков посредством one-hot кодирования.

3. Предфрактальные и динамические графы для моделирования социальных сетей

Любая социальная сеть имеет свой жизненный цикл – от начала появления нескольких вершин, до формирования большого графа с высокой плотностью ребер. Жизненный цикл социальной сети представляет собой на первом этапе рост (набор количества пользователей), формирование связей между вершинами. На следующем этапе, в связи с очевидным ограничением количества пользователей, рост числа вершин замедляется, но при этом активно формируются новые связи – появление новых ребра, исчезают старые связи – удаление ребер. Естественно эти два этапа органично переходят один в другой и распределены во времени, как топологическом, так и в реальном.

Формально социальная сеть является динамическим графом со специальными свойствами. Многие известные алгоритмы выделения сообществ в социальных сетях используют процедуру стягивания сообщества (или группы вершин) в одну вершину. Далее выделяются новые сообщества и снова стягиваются в вершины. Эта процедура продолжается до тех пор, пока весь граф не будет стянут в простой граф (с радиусом один или два ребра). Процедура стягивания соответствует процедуре распознавания блоков и затравок предфрактального графа, а последовательность сетей разных уровней – траектории предфрактального графа. На рисунке 7 изображен предфрактальный граф третьего уровня, пунктирными окружностями выделены затравки, соответствующие сообществам. На рисунке 8 изображен результат центрирования вершин предфрактального графа [13-15].

Таким образом, представляется целесообразным определение принадлежности социальной сети к классу динамических (предфрактальных) графов. Порождение предфрактального графа соответствует росту социальной сети, а блоки и затравки предлагается рассматривать в качестве сообществ и малых групп социальной сети.

Заключение

Исследование социальных сетей, выделение сообществ, построение полиномиальных и эффективных алгоритмов является актуальным направлением и активно проводится зарубежными учеными. Решение задач в социальных сетях востребовано крупными компаниями, в том числе российскими ИТ-гигантами. Представляется, что интерес к этой проблематике будет только возрастать в связи с расширением зоны охвата пользователей в социальных сетях и переходу покупателей в виртуальные магазины. В предстоящие 10 лет рынок услуг в социальных сетях может вырасти в более чем 7 раз, в соответствии с прогнозом увеличения количества людей, которые получат доступ в глобальную сеть Интернет.

Развитие инструментальной базы моделирования, в частности использования динамических и предфрактальных графов, позволит расширить круг задач в социальных сетях, в их числе – многокритериальные (многопараметрические) задачи, задачи с множественными и нечеткими весами, разработка параллельных алгоритмов, прогнозные задачи с заданным уровнем надежности и многие другие [16-20].

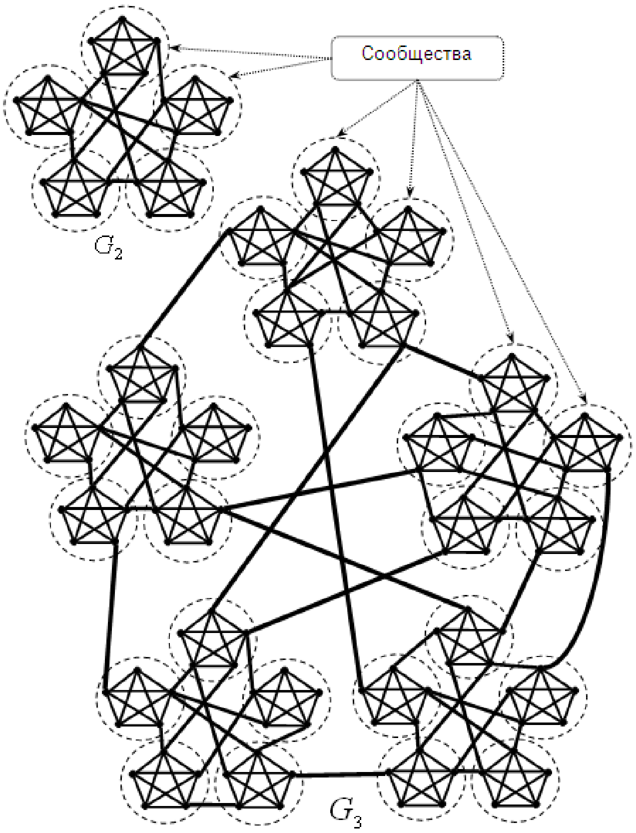

Рисунок 7. Сообщества (затравки) графов G_2 и G_3

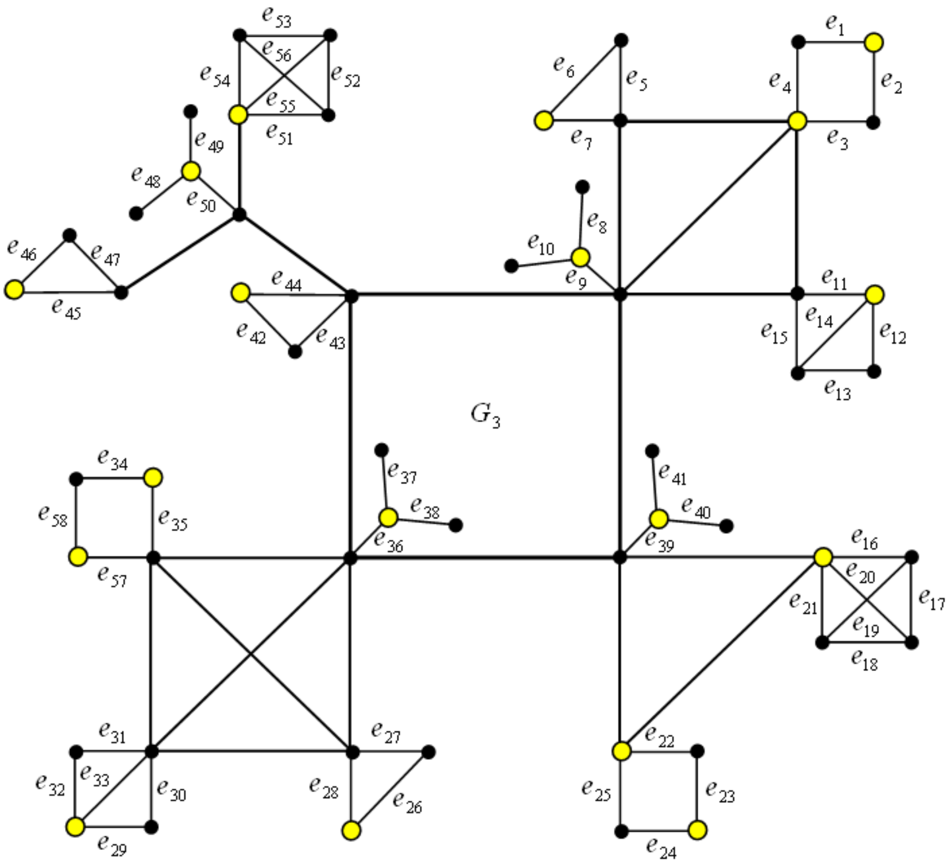

Рисунок 8. Предфрактальный граф G_3

Литература

1. Clauset A., Newman M. E. J., Moore C. Phys.Rev. 2004. 70(6).
2. Girvan M., Newman M.E.J. Community structure in social and biological networks. Proc. Natl Acad. Sci. USA 99, 7821-7826. Proceedings of the National Academy of Sciences of the United States of America. 2002. 99. 7821-6. 10.1073/pnas.122653799.
3. Bonanno G., Vandewalle N., Mantegna R.N. Phys. Rev. 2000. 62(6).
4. Newman M.E.J. Modularity and community structure in networks. Proceedings of the National Academy of Sciences of the United States of America. 2006. 103(23).
5. Li W., Schuurmans D. Modular Community Detection in Networks. IJCAI Proceedings-International Joint Conference on Artificial Intelligence. 2011. 22(1).
6. Clauset A., Newman M.E.J., Moore C. Finding community structure in very large networks. 2004.
7. Blondel V.D., Guillaume J.-L., Lambiotte R., Lefebvre E. Fast unfolding of communities in large networks. Journal of Statistical Mechanics. 2008.
8. Raghavan U.N., Albert R., Kumara S. Near linear time algorithm to detect community structures in large-scale networks. 2008.
9. Pons P., Latapy M. Computing communities in large networks using random walks. 2007.
10. Lloyd S. Least square quantization in PCM's. Bell Telephone Laboratories Paper. 1957.
11. Haut T.S., Beylkin G. Fast and accurate con-eigenvalue algorithm for optimal rational approximations. 2010.
12. Steinhaus H. Sur la division des corps materiels en parties. Bull. Acad. Polon. Sci. 1956. C1. III vol. IV.
13. Perepelitsa V.A., Kochkarov A.M., Sergienko I.V. Recognition of fractal graphs // Cybernetics and Systems Analysis. 1999. Т. 35. № 4. С. 572-585.
14. Кочкаров А.А., Кочкаров А.М., Салпагарова Л.У. Моделирование разрушения сложных сетевых систем: теоретико-графовый подход // Известия ЮФУ. Технические науки. – 2009. – № 5 (94). – С. 234-240.
15. Кочкаров А.А., Кочкаров Р.А., Малинецкий Г.Г. Некоторые аспекты динамической теории графов // Журнал вычислительной математики и математической физики. – 2015. – Т. 55. № 9. – С. 1623-1629.
16. Кочкаров А.А., Кочкаров Р.А. Параллельный алгоритм поиска кратчайшего пути на предфрактальном графе // Журнал вычисл. матем. и матем. физики. 2004. Т. 44. №6. С.1157-1162.
17. Кочкаров Р.А. Многовзвешенные предфрактальные графы с недетерминированными весами. Приложения в экономике, астрофизике и сетевых коммуникациях. М.: Ленанд. 2017. 432 с.
18. Кочкаров А.А., Салпагаров С.И., Кочкаров Р.А. О количественных оценках топологических характеристик предфрактальных графов // Известия ТРТУ. 2004. № 8(43). С. 298-301.
19. Биккузина А.И., Жуков А.О., Никольский Ю.В., Буханец Д.И. Подход к решению задачи упорядочения альтернатив в диалоговой системе моделирования принятия решений при информационно-аналитическом обеспечении оценки и прогноза экологического состояния территорий эксплуатации крупных технических комплексов // Новые исследования в разработке техники и технологий. 2014. № 1. С. 33-39.
20. Гладышев А.И., Жуков А.О. Использование в автоматизированной системе контроля полномочий биометрической идентификации // Вестник Российского нового университета. Серия: Сложные системы: модели, анализ и управление. 2013. № 4. С. 95-98.

УДК 378:001.891:330.43

НАУЧНО-ИССЛЕДОВАТЕЛЬСКАЯ ДЕЯТЕЛЬНОСТЬ ВУЗОВ: СРАВНИТЕЛЬНЫЙ АНАЛИЗ И ОЦЕНКА РЕЗУЛЬТАТОВ

Миролюбова Анастасия Александровна (mirolubowa@mail.ru)
ФГБОУ ВО «Ивановский государственный университет»
Ксенофонтова Ольга Леонидовна
ФГБОУ ВО «Ивановский государственный химико–технологический университет»

В статье проведен сравнительный анализ результатов научно-исследовательской деятельности двух региональных вузов в период реформирования системы образования. Предложена методика оценки эффективности научно-исследовательской деятельности образовательных учреждений на основе эконометрического анализа. Для каждого вуза построен комплекс моделей, позволяющий установить факторы, значимо воздействующие на активность научно-исследовательской деятельности и ее эффективность. Сделан формализованный сравнительный вывод.

Ключевые слова: анализ, вуз, количество опубликованных монографий, статей, научно-исследовательская деятельность, научно-педагогические кадры, полученные патенты, финансирование научно-исследовательской деятельности, эконометрическая модель,